Supplementary_ Table_S7 : Bioinformatic scripts used in genomic analysis

These scripts were used to generate the figures in the paper, processing data from its raw form through the intermediate data structures needed to produce the final figures.

1. Colocalization: These are python script that take in processed bam files, and sets of genomic locations, and produce the colocalization plots.

2. OXBS data analysis: Sanchari/{j432,J446,J432_j446, examine_mc_hmc}: These are the scripts used to process/analyze the OXBS data.

3. ATAC-seq analysis: Sanchari/{j455,j515,j455_J515}: These are the scripts used to process/analyze the atac-seq datasets

4. Combined OXBS and ATAC-seq analysis: Sanchari/oxbs_atac: scripts to examine regions covered by oxbs and atac-seq data.

5. Correlation of RNA-seq qith ATAC-seq and hmC-seal and OXBS: (Sanchari/{peaks_rna, rnaseq}: scripts to examine rna-seq datasets and colocalization with other protocols: atac, oxbs, chip-seq.

6. Sanchari/filterCpg: scripts to filter loci based on CpG status.

7. Sanchari/homer_panc: scripts used in motif analysis

All scripts have been deposited at https://github.com/VermaLab/hmcPaper_scripts2

```
==> ./colocalization/commands1_rnaseq.sh <==


#atack seq peaks, RNA reads
########################3333
#run from cluster
#qsub runPeakProfiles.sh test2 hpneHmc_strandedPeakSummits.txt /home/kpradha1/projects/sanchari/rnaseq/HPNE.AC1FL7ACXX.lane_1_P0_I5.hg19.gsnap-rna-alignment.bam 3000
cut -f1,2,6 hpne_strandedPeakSummits.txt > hpne_atac.txt
cut -f1,2,6 jd_strandedPeakSummits.txt > jd_atac.txt


qsub runPeakProfiles.sh hpneAtacPeaks_hpneRna hpne_atac.txt /home/kpradha1/projects/sanchari/rnaseq/HPNE.AC1FL7ACXX.lane_1_P0_I5.hg19.gsnap-rna-alignment.bam 3000
qsub runPeakProfiles.sh jdAtacPeaks_jdRna jd_atac.txt /home/kpradha1/projects/sanchari/rnaseq/JRN.AD0C99ABXX.lane_8_P0_I2.hg19.gsnap-rna-alignment.bam 3000


#need to prepare the atac-seq DIFF peaks
#use the macs2 run where the other samples was used as control

#1.  need to make the stranded peak summits files for the diffs
#.../sanchari/peaks_rna/annotationPeaks.r

cut -f1,2,6 hpne-jd_strandedPeakSummits.txt > hpne-jd_atac.txt
cut -f1,2,6 jd-hpne_strandedPeakSummits.txt > jd-hpne_atac.txt

#this keeps crashing.  maybe try to
#take out all the non chr* entries
grep -v hpne-jd_atac.txt -e "^chr."

grep -v jd-hpne_atac.txt -e "^chr[^_]\+_" > jd-hpne_atac1.txt
grep -v hpne-jd_atac.txt -e "^chr[^_]\+_" > hpne-jd_atac1.txt

qsub runPeakProfiles.sh hpne-jdAtacPeaks_hpneRna hpne-jd_atac1.txt /home/kpradha1/projects/sanchari/rnaseq/HPNE.AC1FL7ACXX.lane_1_P0_I5.hg19.gsnap-rna-alignment.bam 3000
qsub runPeakProfiles.sh jd-hpneAtacPeaks_jdRna jd-hpne_atac1.txt /home/kpradha1/projects/sanchari/rnaseq/JRN.AD0C99ABXX.lane_8_P0_I2.hg19.gsnap-rna-alignment.bam 3000



#get plots of the JD gain/loss peak sets

#first, get stranded info for the peak sets
#/hpc_home/projects/sanchari/peaks_rna/annotatePeaks.r
#peaks.atac.jd.gain.1k.txt
#peaks.atac.jd.loss.1k.txt

cut -f1,2,6 peaks.atac.jd.gain.1k_stranded.txt > jdGain1k_atac.txt
cut -f1,2,6 peaks.atac.jd.loss.1k_stranded.txt > jdLoss1k_atac.txt

qsub runPeakProfiles.sh jdGain1kAtacPeaks_jdRna jdGain1k_atac.txt /home/kpradha1/projects/sanchari/rnaseq/JRN.AD0C99ABXX.lane_8_P0_I2.hg19.gsnap-rna-alignment.bam 3000
qsub runPeakProfiles.sh jdLoss1kAtacPeaks_hpneRna jdLoss1k_atac.txt /home/kpradha1/projects/sanchari/rnaseq/HPNE.AC1FL7ACXX.lane_1_P0_I5.hg19.gsnap-rna-alignment.bam 3000





#hmc sites, RNA reads
#no..have to add strandedness info.
####################################

#todo
#.../sanchari/peaks_rna/annotationPeaks.r

qsub runPeakProfiles.sh hpneHmc_hpneRna hpne_strandedHmc.txt /home/kpradha1/projects/sanchari/rnaseq/HPNE.AC1FL7ACXX.lane_1_P0_I5.hg19.gsnap-rna-alignment.bam 3000
qsub runPeakProfiles.sh jdHmc_jdRna jd_strandedHmc.txt /home/kpradha1/projects/sanchari/rnaseq/JRN.AD0C99ABXX.lane_8_P0_I2.hg19.gsnap-rna-alignment.bam 3000




#oxbs hmc sites, atac seq reads
####################################################33

#prep oxbs sites
cat hmc_sig0.01_commonSites1_jd.bed | cut -f1,2 | sed 's/.*/&\t+/' >| jd_hmc.txt
cat hmc_sig0.01_commonSites1_hpne.bed | cut -f1,2 | sed 's/.*/&\t+/' >| hpne_hmc.txt


qsub runPeakProfiles.sh hpneHmcSites_hpneAtac hpne_hmc.txt /home/kpradha1/projects/sanchari/j455/hpne_sorted_mdup.bam 3000
qsub runPeakProfiles.sh jdHmcSites_jdAtac jd_hmc.txt /home/kpradha1/projects/sanchari/j455_j515/jd_sorted_mdup_merged.bam 3000




#oxbs mc sites, atac seq reads
####################################################33
#prep mc files
cat mc_commonSites1_jd.bed | cut -f1,2 | sed 's/.*/&\t+/' >| jd_mc.txt
cat mc_commonSites1_hpne.bed | cut -f1,2 | sed 's/.*/&\t+/' >| hpne_mc.txt

qsub runPeakProfiles.sh hpneMcSites_hpneAtac_400k hpne_mc.txt /home/kpradha1/projects/sanchari/j455/hpne_sorted_mdup.bam 3000
qsub runPeakProfiles.sh jdMcSites_jdAtac_400k jd_mc.txt /home/kpradha1/projects/sanchari/j455_j515/jd_sorted_mdup_merged.bam 3000



#mc files with rna
################################
#run annotation to get strand info
```

```
#stat3 sites, atac seq reads
###########################################33

#prepare the stat3 peaks from bed files

cat jd13d_stat3_summits.bed | cut -f1,2 | sed 's/.*/&\t+/' | grep -v "hs37d5" | sed
's/.*/chr&/' >| jd_stat3.txt
cat hpne_stat3_summits.bed | cut -f1,2 | sed 's/.*/&\t+/'  | grep -v "hs37d5" | sed
's/.*/chr&/' >| hpne_stat3.txt


qsub runPeakProfiles.sh hpneStat3Sites_hpneAtac hpne_stat3.txt
/home/kpradha1/projects/sanchari/j455/hpne_sorted_mdup.bam 3000
qsub runPeakProfiles.sh jdStat3Sites_jdAtac jd_stat3.txt
/home/kpradha1/projects/sanchari/j455_j515/jd_sorted_mdup_merged.bam 3000



#prepare stranded stat3 files with rna
#add chr to the front of each row
cat jd_stat3_summits_stranded.txt | sed 's/.*/chr&/' >
jd_stat3_summits_stranded1.txt
cat hpne_stat3_summits_stranded.txt | sed 's/.*/chr&/' >
hpne_stat3_summits_stranded1.txt


qsub runPeakProfiles.sh jdstat3Peaks_jdRna jd_stat3_summits_stranded1.txt
/home/kpradha1/projects/sanchari/rnaseq/JRN.AD0C99ABXX.lane_8_P0_I2.hg19.gsnap-rna-
alignment.bam 3000
qsub runPeakProfiles.sh hpnestat3Peaks_hpneRna hpne_stat3_summits_stranded1.txt
/home/kpradha1/projects/sanchari/rnaseq/HPNE.AC1FL7ACXX.lane_1_P0_I5.hg19.gsnap-
rna-alignment.bam 3000



#move all the pics to a separate folder
ls */colocPlot.png | sed 's$\(.*\)\/\(.*\)$cp & pics1\/\1\.png$' | bash


########################################################################



bamfile1="/home/kpradhan/Desktop/data/sanchari/pancreatic/rnaseq/bam/HPNE.AC1FL7ACX
X.lane_1_P0_I5.hg19.gsnap-rna-alignment.bam"

samtools view -b $bamfile1 chr1:561524-567524 | samtools mpileup -

samtools mpileup $bamfile1 -r chr1:561524-567524 | cut -f1-5


#run from my comp
python makeColocPlots.py test2

==> ./colocalization/getPeakProfiles_preload.py <==
#from multiprocessing import Pool
```

```
from functools import partial
from itertools import chain
import time
import peakProfiles as pp
import numpy as np
import matplotlib.pyplot as plt
import os
import sys
import HTSeq as ht
import random



#reload(pp)



outFolder = "test5"
#siteFile  = "hpneHmc_strandedPeakSummits.txt"
siteFile  = "hpne_mc.txt"
siteFile  = "hpne_stat3.txt"
siteFile = "hpne_hmc_p05.bed"
#bamfile1  =
"/home/kpradha1/projects/sanchari/rnaseq/HPNE.AC1FL7ACXX.lane_1_P0_I5.hg19.gsnap-
rna-alignment.bam"
#bamfile1 =
"/home/kpradhan/Desktop/hpc_home/projects/sanchari/rnaseq/HPNE.AC1FL7ACXX.lane_1_P0
_I5.hg19.gsnap-rna-alignment.bam"
bamfile1 =
"/home/kpradhan/Desktop/hpc_home/projects/sanchari/j455/hpne_sorted_mdup.bam"
bamfile1 =
"/home/kpradhan/Desktop/hpc_home/projects/sanchari/j455/hpne_sorted_mdup_chr10.bam"
W         = 3000
#K=200000  #only look at this number of sites due to memory constraints
K=400000  #only look at this number of sites due to memory constraints

#command line arguments
outFolder = sys.argv[1]
siteFile  = sys.argv[2]
bamfile1  = sys.argv[3]
W         = int(sys.argv[4])

print ("outfolder:  %s" % outFolder)
print ("siteFile:  %s" % siteFile)
print ("bamfile:  %s" % bamfile1)
print ("W:  %d" % W)



#load the sites
sites = pp.getPeakSitesFromStrandedBed(siteFile)



#limit the number of sites to save memory
if len(sites) > K:
    ix = np.sort(random.sample(range(0, len(sites)), K))
    sites = sites[ix]
```

```
#turn any sites with * strands into .
for s in sites:
    if s['strand'] != "+" and s['strand'] != "-":
        s['strand'] = "."


#take out any sites that are lower than the halfwidth
ix = sites['site'] > W
sites = sites[ix]

#first thing...check that the sites match the notation of the bam file
sites, nMatch = pp.checkSiteNotation(sites, bamfile1)
len(sites)
print("number of sites matching bam chroms: %d" % nMatch)


#get rid of the sites that have no chromosome entry in the bam file

#save results in a folder
if not os.path.exists(outFolder):
    os.makedirs(outFolder)

#save some batch information
with open(os.path.join(outFolder, "batch_info.txt"), "w") as myfile:
    myfile.write("outfolder:  %s\n" % outFolder)
    myfile.write("siteFile:  %s\n" % siteFile)
    myfile.write("bamfile:  %s\n" % bamfile1)
    myfile.write("W:  %d\n" % W)
    myfile.write("#matching sites:  %d\n" % nMatch)


#if all the sites are facing the same way, used * strand coverage
stranded = True
if len(np.unique(sites['strand'])) == 1:
    stranded = False
    sites['strand'] = "."


#load the entire bam into a genomic array
bam = ht.BAM_Reader(bamfile1)
#maybe, if all the site strands are *, don't strand coverage
coverage = ht.GenomicArray( "auto", stranded=stranded, typecode="i" )
#coverage = ht.GenomicArray( "auto", stranded=True, typecode="i" )
for i, almnt in enumerate(bam):
    if i % 100000 == 0:
        print (i )
    #check for quality and pcr dups!
    if almnt.aligned and almnt.aQual > 0 and not almnt.pcr_or_optical_duplicate:
        coverage[ almnt.iv ] += 1


#reload(pp)
peakProfs = pp.getPeakProfilesFromCoverage(sites, coverage, halfwinwidth=3000,
stranded=stranded)

#or mabye use the mean
prof = np.sum(peakProfs, 0)
#the binned image
```

```
smallPeaks = pp.binPeakProfiles(peakProfs, K=100)

np.save(os.path.join(outFolder, "peakProfiles.npy"), peakProfs)
np.save(os.path.join(outFolder, "summedPP.npy"), prof)
np.save(os.path.join(outFolder, "binnedPP.npy"), smallPeaks)




==> ./colocalization/getPeakProfiles_noHeatmap_preload.py <==
#from multiprocessing import Pool
from functools import partial
from itertools import chain
import time
import peakProfiles as pp
import numpy as np
import matplotlib.pyplot as plt
import os
import sys
import HTSeq as ht
import random



#reload(pp)



outFolder = "test5"
#siteFile  = "hpneHmc_strandedPeakSummits.txt"
siteFile  = "hpne_mc.txt"
siteFile  = "hpne_stat3.txt"
siteFile = "hpne_hmc_p05.bed"
#bamfile1  = 
"/home/kpradha1/projects/sanchari/rnaseq/HPNE.AC1FL7ACXX.lane_1_P0_I5.hg19.gsnap-
rna-alignment.bam"
#bamfile1 = 
"/home/kpradhan/Desktop/hpc_home/projects/sanchari/rnaseq/HPNE.AC1FL7ACXX.lane_1_P0
_I5.hg19.gsnap-rna-alignment.bam"
bamfile1 = 
"/home/kpradhan/Desktop/hpc_home/projects/sanchari/j455/hpne_sorted_mdup.bam"
bamfile1 = 
"/home/kpradhan/Desktop/hpc_home/projects/sanchari/j455/hpne_sorted_mdup_chr10.bam"
W         = 3000
#K=200000  #only look at this number of sites due to memory constraints
K=400000  #only look at this number of sites due to memory constraints

#command line arguments
outFolder = sys.argv[1]
siteFile  = sys.argv[2]
bamfile1  = sys.argv[3]
W         = int(sys.argv[4])

print ("outfolder:  %s" % outFolder)
print ("siteFile:  %s" % siteFile)
print ("bamfile:  %s" % bamfile1)
print ("W:  %d" % W)
```

```
#load the sites
sites = pp.getPeakSitesFromStrandedBed(siteFile)


#turn any sites with * strands into .
for s in sites:
    if s['strand'] != "+" and s['strand'] != "-":
        s['strand'] = "."


#take out any sites that are lower than the halfwidth
ix = sites['site'] > W
sites = sites[ix]

#first thing...check that the sites match the notation of the bam file
sites, nMatch = pp.checkSiteNotation(sites, bamfile1)
len(sites)
print("number of sites matching bam chroms: %d" % nMatch)


#get rid of the sites that have no chromosome entry in the bam file

#save results in a folder
if not os.path.exists(outFolder):
    os.makedirs(outFolder)

#save some batch information
with open(os.path.join(outFolder, "batch_info.txt"), "w") as myfile:
    myfile.write("outfolder:  %s\n" % outFolder)
    myfile.write("siteFile:  %s\n" % siteFile)
    myfile.write("bamfile:  %s\n" % bamfile1)
    myfile.write("W:  %d\n" % W)
    myfile.write("#matching sites:  %d\n" % nMatch)


#if all the sites are facing the same way, used * strand coverage
stranded = True
if len(np.unique(sites['strand'])) == 1:
    stranded = False
    sites['strand'] = "."


#load the entire bam into a genomic array
bam = ht.BAM_Reader(bamfile1)
#maybe, if all the site strands are *, don't strand coverage
coverage = ht.GenomicArray( "auto", stranded=stranded, typecode="i" )
#coverage = ht.GenomicArray( "auto", stranded=True, typecode="i" )
for i, almnt in enumerate(bam):
    if i % 100000 == 0:
        print (i )
    #check for quality and pcr dups!
    if almnt.aligned and almnt.aQual > 0 and not almnt.pcr_or_optical_duplicate:
        coverage[ almnt.iv ] += 1


#reload(pp)
#todo
##just save the summed Profile
```

```
peakProf = pp.getProfileFromCoverage(sites, coverage, halfwinwidth=3000,
stranded=stranded)

np.save(os.path.join(outFolder, "peakSums.npy"), peakProf)




==> ./colocalization/runPeakProfiles_noHeatmap.sh <==
#!/bin/bash
#$ -cwd
#$ -j n
#$ -l h_vmem=10G
#$ -pe smp 1
#$ -S /bin/bash
#$ -m e
#usage:
#qsub runPeakProfiles.sh asdlk asdflka
#probably going to need more memory
module load scipy/0.14.0/python.2.7.8-atlas-3.11.30
module load python/2.7.8/gcc.4.4.7
module load HTSeq/0.6.1/python.2.7.8
module load numpy/1.9.0/python.2.7.8-atlas-3.11.30
module load matplotlib/1.4.3/python.2.7.8



outputFolder="test3"
#siteFile="hpneHmc_strandedPeakSummits.txt"
siteFile="hpne_atac.txt"
#bamFile="/home/kpradha1/projects/sanchari/rnaseq/HPNE.AC1FL7ACXX.lane_1_P0_I5.hg19
.gsnap-rna-alignment.bam"
bamFile="/home/kpradhan/Desktop/hpc_home/projects/sanchari/rnaseq/HPNE.AC1FL7ACXX.l
ane_1_P0_I5.hg19.gsnap-rna-alignment.bam"
W="3000"

outputFolder=$1
siteFile=$2
bamFile=$3
W=$4


echo $outputFolder
echo $siteFile
echo $bamFile
echo $W


#make a folder for storing intermediate resulst
#otherwise, might run out of space on default /tmp
python getPeakProfiles_noHeatmap_preload.py $outputFolder $siteFile $bamFile $W

#add a $ after the # if want email notification at job completion
# -M kith.pradhan@einstein.yu.edu



==> ./colocalization/peakProfiles.py <==
import numpy as np
import matplotlib.pyplot as plt
import glob
import os
import sys
import HTSeq as ht
import numpy.random as rd


def chunks(l, n):
    """Yield successive n-sized chunks from l."""
    for i in range(0, len(l), n):
        yield l[i:i + n]


#the bed file should have 3 columns
#chrom
#position start
#position end
#strand + or -
def getPeakSitesFromStrandedBed(pfile):
    peaks = np.loadtxt(pfile, dtype=object)
    #regions lists will all be the same format
    dt_sites = np.dtype([("chrom", "S5"), ("strand", "S1"), ("site", "int")])
    #x = [(str(a[0]), str(a[3]), int(a[2])) for a in peaks]
    x = [(str(a[0]), str(a[2]), int(a[1])) for a in peaks]
    return np.unique(np.array(x, dtype=dt_sites))



def getPeakSitesFromBed(pfile):
    peaks = np.loadtxt(pfile, dtype=object)
    #regions lists will all be the same format
    dt_sites = np.dtype([("chrom", "S5"), ("strand", "S1"), ("start", "int"),
("stop", "int")])
    x = [(str(a[1]), str(a[2]), int(a[3])-PROM_DIST, int(a[3])+PROM_DIST) for a in
peaks]
    sites_tss = np.unique(np.array(x, dtype=dt_sites))


#pfile = "peaks_jd13d_stat3Input.txt"
def getPeakSitesFromMacs(pfile):
    peaks = np.loadtxt(pfile, dtype=object)
    #regions lists will all be the same format
    dt_sites = np.dtype([("chrom", "S5"), ("strand", "S1"), ("start", "int"),
("stop", "int")])
    x = []
    for a in peaks:
        #mid = (int(a[2]) + int(a[1])) / 2
        #x = [(str(a[0]), "+", mid-200, mid+200) for a in peaks]
        x.append( (str(a[0]), "+", int(a[1]), int(a[2])) )
    #only retain proper chromosomes
    x = np.unique(np.array(x, dtype=dt_sites))
    ix = np.array([a[:3] == "chr" or a == "X" or a == "Y" for a in x["chrom"]] )
    return x[ix]


#maybe only look at the peaks that pass a threshold
def getPeakMidsFromMacs(pfile, thresh=0):
    peaks = np.loadtxt(pfile, dtype=object)
    #regions lists will all be the same format
    dt_sites = np.dtype([("chrom", "S5"), ("strand", "S1"), ("site", "int")])
    x = []
    for a in peaks:
        if thresh == 0 or float(a[5]) > thresh:
            mid = (int(a[2]) + int(a[1])) / 2
            x.append((str(a[0]), "+", mid))
        #x.append( (str(a[0]), "+", int(a[1]), int(a[2])) )
    #only retain proper chromosomes
    x = np.unique(np.array(x, dtype=dt_sites))
    ix = np.array([a[:3] == "chr" or a == "X" or a == "Y" for a in x["chrom"]] )
    return x[ix]


def truncChr(c):
    if c.isdigit():
        return c
    elif c == "X" or c == "Y":
        return c
    else:
        return c[3:]


#how many sites have thee same notation as the bam file.
def checkSiteNotation(sites, bamfile):
    bam = ht.BAM_Reader(bamfile)
    #make sure the sites and bam files have same naming convention
    #
    #retrict sites to those that have an entry in the bam file
    #probably has bug dealing with X and Y.
    bamChroms = [x["SN"] for x in bam.get_header_dict()["SQ"]]
    sites = sites[np.in1d(sites["chrom"], bamChroms)]
    return sites, np.sum(np.in1d(sites["chrom"], bamChroms))


def getProfileFromCoverage(sites, coverage, halfwinwidth=3000, stranded=False):
    nSites = 0
    peakSums = np.zeros(2*halfwinwidth)
    #collect the sites as genomic intervals
    for i, pos in enumerate(sites):
        #print "%d of %d" %(i, len(sites))
        peakProfile = np.zeros(2*halfwinwidth)
        if i % 1000 == 0:
            print "%d of %d" %(i, len(sites))
        leftSide = pos["site"] - halfwinwidth
        if leftSide < 1:
            leftSide = 1
        rightSide = pos["site"] + halfwinwidth
        if (stranded):
            window = ht.GenomicInterval( str(pos["chrom"]), leftSide, rightSide,
str(pos["strand"]) )
        else:
            window = ht.GenomicInterval( str(pos["chrom"]), leftSide, rightSide,
".")
        #print (pos)
        #try this out 1-17-17
        wincvg = np.array(list(coverage[window]))
        #if the iterator has nothing, it'll throw an error
        #wincvg = np.fromiter( coverage[window], dtype='i', count=2*halfwinwidth )
        if (stranded):
            if pos["strand"] == "+":
                peakProfile += wincvg
            elif pos["strand"] == "-":
                peakProfile += wincvg[::-1]
        else:
            peakProfile += wincvg
        if (np.sum(peakProfile) > 0):
            peakSums += peakProfile
    return peakSums


def getPeakProfilesFromCoverage(sites, coverage, halfwinwidth=3000,
stranded=False):
    peakProfs = []
    #collect the sites as genomic intervals
    for i, pos in enumerate(sites):
        #print "%d of %d" %(i, len(sites))
        peakProfile = np.zeros(2*halfwinwidth)
        if i % 1000 == 0:
            print "%d of %d" %(i, len(sites))
        leftSide = pos["site"] - halfwinwidth
        if leftSide < 1:
            leftSide = 1
        rightSide = pos["site"] + halfwinwidth
        if (stranded):
            window = ht.GenomicInterval( str(pos["chrom"]), leftSide, rightSide,
str(pos["strand"]) )
        else:
            window = ht.GenomicInterval( str(pos["chrom"]), leftSide, rightSide,
".")
        #print (pos)
        #try this out 1-17-17
        wincvg = np.array(list(coverage[window]))
        #if the iterator has nothing, it'll throw an error
        #wincvg = np.fromiter( coverage[window], dtype='i', count=2*halfwinwidth )
        if (stranded):
            if pos["strand"] == "+":
                peakProfile += wincvg
            elif pos["strand"] == "-":
                peakProfile += wincvg[::-1]
        else:
            peakProfile += wincvg
        if (np.sum(peakProfile) > 0):
            peakProfs.append(peakProfile)
    return np.array(peakProfs)



#need a way to determine if the bam file has notation as "chr10" or "10"

bamfile = "/home/kpradhan/Desktop/data/pancreas/test1/test3.bam"
def getPeakProfiles(sites, bamfile,halfwinwidth=3000):
    bam = ht.BAM_Reader(bamfile)
    #make sure the sites and bam files have same naming convention
    #
    #retrict sites to those that have an entry in the bam file
    #probably has bug dealing with X and Y.
    bamChroms = [x["SN"] for x in bam.get_header_dict()["SQ"]]
    bamChroms = ["chr"+c if c.isdigit() else c for c in bamChroms]
    sites = sites[np.in1d(sites["chrom"], bamChroms)]
    sites.shape
    #
    peakProfs = []
    #collect the sites as genomic intervals
    for i, pos in enumerate(sites):
        print "%d of %d" %(i, len(sites))
        peakProfile = np.zeros(2*halfwinwidth)
        if i % 1000 == 0:
            print "%d of %d" %(i, len(sites))
        #don't change the site notation here
        #sitechr = truncChr(pos["chrom"]) if hasChrPrefix else pos["chrom"]
        sitechr = pos["chrom"]
        window = ht.GenomicInterval( str(sitechr), pos["site"] - halfwinwidth,
pos["site"] + halfwinwidth, str(pos["strand"]) )
        #if (list(bam[window])):
        if next(bam[window], None) is not None:
            for almnt in bam[window]:
                if pos["strand"] == "+":
                    a = almnt.iv.start - pos["site"] + halfwinwidth
                    b = almnt.iv.end - pos["site"] + halfwinwidth
                if pos["strand"] == "-":
                    a = pos["site"] + halfwinwidth - almnt.iv.end
                    b = pos["site"] + halfwinwidth - almnt.iv.start
                peakProfile[a:b] += 1
        if (np.sum(peakProfile) > 0):
            peakProfs.append(peakProfile)
    return np.array(peakProfs)

def binPeakProfiles(peakProfs, K=100):
    sums = [np.sum(p) for p in peakProfs]
    ix = np.argsort(sums)
    #
    #bin every 100 rows together
    smallX = []
    for i in range(0, len(peakProfs)-1, K):
        smallX.append(np.sum(np.array([peakProfs[j] for j in ix[i:(i+K)]]),0))
    return np.array(smallX)[::-1]


#prof is the aveage profile
#smallPeaks is the binned peak image
#cuts off 500 from either side
def savePeakPlots(outfile, prof, smallPeaks):
    #half width of the window
    W = len(prof)/2
    x = smallPeaks.ravel()
    vmax = np.sort(x)[np.round(len(x)*.80)]
    #
    plt.close()
    fig = plt.figure()
    fig.set_size_inches(18.5, 10.5)
    fig.suptitle(outfile)
    ax = plt.subplot(121)
    xran = np.arange(len(prof))-W
    plt.plot(xran[500:(-500)], prof[500:(-500)])
    ax = plt.subplot(122)
    ax.set_xticklabels([])
    ax.set_yticklabels([])
    plt.imshow(smallPeaks[:, 500:(-500),],  aspect = "auto", vmax=vmax)
    #plt.colorbar(orientation="horizontal")
    plt.colorbar(orientation="vertical")
    plt.savefig(outfile, dpi=100)
    plt.close()



def savePeakPlots_part(outfile, prof, smallPeaks, throwPortion=0.20):
    rowStart = int(smallPeaks.shape[0]*throwPortion )
    rowEnd = smallPeaks.shape[0]
    #
    #half width of the window
    W = len(prof)/2
    x = smallPeaks.ravel()
    vmax = np.sort(x)[np.round(len(x)*.80)]
    #
    plt.close()
    fig = plt.figure()
    fig.set_size_inches(18.5, 10.5)
    fig.suptitle(outfile)
    ax = plt.subplot(121)
    xran = np.arange(len(prof))-W
    #
    test1 = np.sum(smallPeaks[rowStart:rowEnd, 500:(-500),],0)
    plt.plot(xran[500:(-500)], test1)
    np.sum(smallPeaks[1000:4000, 500:(-500),],0)
    #
    #plt.plot(xran[500:(-500)], prof[500:(-500)])
    ax = plt.subplot(122)
    ax.set_xticklabels([])
    ax.set_yticklabels([])
    #
    x = smallPeaks[rowStart:rowEnd, 500:(-500),].ravel()
    vmax = np.sort(x)[np.round(len(x)*.80)]
    plt.imshow(smallPeaks[rowStart:rowEnd, 500:(-500),],  aspect = "auto",
vmax=vmax)
    #plt.colorbar(orientation="horizontal")
    plt.colorbar(orientation="vertical")
    plt.savefig(outfile, dpi=100)
    plt.close()


#test


#peakmids_jd13d = getPeakMidsFromMacs(pfile)
#sites=peakmids_jd13d
def makePeakPlots(sites, bamfile, outfile, noChr=False):
    peakProfs = getPeakProfiles(bamfile, sites, 3000, noChr)
    #
    prof = np.sum(peakProfs, 0)
    smallPeaks = binPeakProfiles(peakProfs, K=100)
    x = smallPeaks.ravel()
    vmax = np.sort(x)[np.round(len(x)*.80)]
    #
    plt.close()
    fig = plt.figure()
    fig.suptitle(outfile)
    ax = plt.subplot(121)
    xran = np.arange(len(prof))-3000
    plt.plot(xran[500:(-500)], prof[500:(-500)])
    ax = plt.subplot(122)
    ax.set_xticklabels([])
    ax.set_yticklabels([])
    plt.imshow(smallPeaks[:, 500:(-500),],  aspect = "auto", vmax=vmax)
    #plt.colorbar(orientation="horizontal")
    plt.colorbar(orientation="vertical")
    plt.savefig(outfile)
    plt.close()


def clipFilename(ffile):
    return ffile[(ffile.rfind("/")+1):]



==> ./colocalization/getPeakProfiles.py <==
#from multiprocessing import Pool
from functools import partial
from itertools import chain
import time
import peakProfiles as pp
import numpy as np
import matplotlib.pyplot as plt
import os
import sys



#$reload(pp)


outFolder = "test3"
#siteFile  = "hpneHmc_strandedPeakSummits.txt"
siteFile  = "hpne_atac.txt"
#bamfile1  =
"/home/kpradha1/projects/sanchari/rnaseq/HPNE.AC1FL7ACXX.lane_1_P0_I5.hg19.gsnap-
rna-alignment.bam"
bamfile1 =
"/home/kpradhan/Desktop/hpc_home/projects/sanchari/rnaseq/HPNE.AC1FL7ACXX.lane_1_P0
_I5.hg19.gsnap-rna-alignment.bam"
W         = 3000

#command line arguments
outFolder = sys.argv[1]
siteFile  = sys.argv[2]
bamfile1  = sys.argv[3]
W         = int(sys.argv[4])

print ("outfolder:  %s" % outFolder)
print ("siteFile:  %s" % siteFile)
print ("bamfile:  %s" % bamfile1)
print ("W:  %d" % W)
```

```
#load the sites
sites = pp.getPeakSitesFromStrandedBed(siteFile)
#first thing...check that the sites match the notation of the bam file
nMatch = pp.checkSiteNotation(sites, bamfile1)
print("number of sites matching bam chroms: %d" % nMatch)

#save results in a folder
if not os.path.exists(outFolder):
    os.makedirs(outFolder)

#save some batch information
with open(os.path.join(outFolder, "batch_info.txt"), "w") as myfile:
    myfile.write("outfolder:  %s\n" % outFolder)
    myfile.write("siteFile:  %s\n" % siteFile)
    myfile.write("bamfile:  %s\n" % bamfile1)
    myfile.write("W:  %d\n" % W)
    myfile.write("#matching sites:  %d\n" % nMatch)


#partial function that allows multiple arguments to be passed to map
getPPhelper = partial(pp.getPeakProfiles, bamfile = bamfile1, halfwinwidth=W)

#pp.getPeakProfiles(sites[:100],bamfile1, W, hasChrPrefix=False)
#getPPhelper(sites[:100])


#multiprocessing
# Make the Pool of workers
#pool = Pool(processes=4)


b = time.time()
print (b)
#results = pool.map(getPPhelper, pp.chunks(sites, 1000))
#results = pool.map(getPPhelper, pp.chunks(sites[:100], 10))
results = pp.getPeakProfiles(sites[:100,  bamfile = bamfile1, halfwinwidth=W)
#results = getPPhelper(sites[:100])
peakProfs = np.concatenate(results)
a = time.time()
print (a - b)

#close the pool and wait for the work to finish (neccesary?)
#pool.close()
#pool.join()


#or mabye use the mean
prof = np.sum(peakProfs, 0)
#the binned image
smallPeaks = pp.binPeakProfiles(peakProfs, K=100)

np.save(os.path.join(outFolder, "peakProfiles.npy"), peakProfs)
np.save(os.path.join(outFolder, "summedPP.npy"), prof)
np.save(os.path.join(outFolder, "binnedPP.npy"), smallPeaks)
```

```
==> ./colocalization/makeColocPlots.py <==
import peakProfiles as pp
import numpy as np
import os
import sys

reload(pp)

resFolder = "hpneMcSites_hpneAtac"
resFolder = 'hpne_mc_100_hpneAtac'
resFolder = 'jd_mc_100_jdAtac'
resFolder = sys.argv[1]

#peakProfs = np.load(os.path.join(resFolder, "peakProfiles.npy"))
prof = np.load(os.path.join(resFolder, "summedPP.npy"))
smallPeaks = np.load(os.path.join(resFolder, "binnedPP.npy"))

pp.savePeakPlots(os.path.join(resFolder, "colocPlot.png"), prof, smallPeaks)
pp.savePeakPlots_part(os.path.join(resFolder, "colocPlot_20.png"), prof,
smallPeaks, throwPortion = .20)
pp.savePeakPlots_part(os.path.join(resFolder, "colocPlot_50.png"), prof,
smallPeaks, throwPortion = .50)



==> ./colocalization/commands2_fixed2Sets.sh <==
#########################################
#prepare the files for correlation with atac-seq


cat jd_mc_100.txt | cut -f1,2 | sed 's/.*/&\t+/' >| jd_mc_100.bed
cat jd_mc_lt50.txt | cut -f1,2 | sed 's/.*/&\t+/' >| jd_mc_lt50.bed
cat jd_mc_gt50.txt | cut -f1,2 | sed 's/.*/&\t+/' >| jd_mc_gt50.bed
cat jd_hmc_p05.txt | cut -f1,2 | sed 's/.*/&\t+/' >| jd_hmc_p05.bed

cat hpne_mc_100.txt | cut -f1,2 | sed 's/.*/&\t+/' >|  hpne_mc_100.bed
cat hpne_mc_lt50.txt | cut -f1,2 | sed 's/.*/&\t+/' >| hpne_mc_lt50.bed
cat hpne_mc_gt50.txt | cut -f1,2 | sed 's/.*/&\t+/' >| hpne_mc_gt50.bed
cat hpne_hmc_p05.txt | cut -f1,2 | sed 's/.*/&\t+/' >| hpne_hmc_p05.bed



qsub runPeakProfiles.sh jd_mc_100_jdAtac  jd_mc_100.bed
/home/kpradha1/projects/sanchari/j455_j515/jd_sorted_mdup_merged.bam 3000
qsub runPeakProfiles.sh jd_mc_lt50_jdAtac jd_mc_lt50.bed
/home/kpradha1/projects/sanchari/j455_j515/jd_sorted_mdup_merged.bam 3000
qsub runPeakProfiles.sh jd_mc_gt50_jdAtac jd_mc_gt50.bed
/home/kpradha1/projects/sanchari/j455_j515/jd_sorted_mdup_merged.bam 3000
qsub runPeakProfiles.sh jd_hmc_p05_jdAtac jd_hmc_p05.bed
/home/kpradha1/projects/sanchari/j455_j515/jd_sorted_mdup_merged.bam 3000

qsub runPeakProfiles.sh hpne_mc_100_hpneAtac  hpne_mc_100.bed
/home/kpradha1/projects/sanchari/j455/hpne_sorted_mdup.bam 3000
qsub runPeakProfiles.sh hpne_mc_lt50_hpneAtac hpne_mc_lt50.bed
/home/kpradha1/projects/sanchari/j455/hpne_sorted_mdup.bam 3000
qsub runPeakProfiles.sh hpne_mc_gt50_hpneAtac hpne_mc_gt50.bed
/home/kpradha1/projects/sanchari/j455/hpne_sorted_mdup.bam 3000
```

```
qsub runPeakProfiles.sh hpne_hmc_p05_hpneAtac hpne_hmc_p05.bed /home/kpradha1/projects/sanchari/j455/hpne_sorted_mdup.bam 3000



python makeColocPlots.py jd_mc_100_jdAtac
python makeColocPlots.py jd_mc_lt50_jdAtac
python makeColocPlots.py jd_mc_gt50_jdAtac
python makeColocPlots.py jd_hmc_p05_jdAtac
python makeColocPlots.py hpne_mc_100_hpneAtac
python makeColocPlots.py hpne_mc_lt50_hpneAtac
python makeColocPlots.py hpne_mc_gt50_hpneAtac
python makeColocPlots.py hpne_hmc_p05_hpneAtac



####################################
#now prepare the files for RNA correlation

#prepare the stranded position info
BAMJD=/home/kpradha1/projects/sanchari/rnaseq/JRN.AD0C99ABXX.lane_8_P0_I2.hg19.gsnap-rna-alignment.bam
BAMHPNE=/home/kpradha1/projects/sanchari/rnaseq/HPNE.AC1FL7ACXX.lane_1_P0_I5.hg19.gsnap-rna-alignment.bam


qsub runPeakProfiles.sh jd_mc_100_jdRna  jd_mc_100.bed $BAMJD 3000
qsub runPeakProfiles.sh jd_mc_lt50_jdRna jd_mc_lt50.bed $BAMJD 3000
qsub runPeakProfiles.sh jd_mc_gt50_jdRna jd_mc_gt50.bed $BAMJD 3000
qsub runPeakProfiles.sh jd_hmc_p05_jdRna jd_hmc_p05.bed $BAMJD 3000

qsub runPeakProfiles.sh hpne_mc_100_hpneRna  hpne_mc_100.bed  $BAMHPNE 3000
qsub runPeakProfiles.sh hpne_mc_lt50_hpneRna hpne_mc_lt50.bed $BAMHPNE 3000
qsub runPeakProfiles.sh hpne_mc_gt50_hpneRna hpne_mc_gt50.bed $BAMHPNE 3000
qsub runPeakProfiles.sh hpne_hmc_p05_hpneRna hpne_hmc_p05.bed $BAMHPNE 3000



python makeColocPlots.py  jd_mc_100_jdRna
python makeColocPlots.py jd_mc_lt50_jdRna
python makeColocPlots.py jd_mc_gt50_jdRna
python makeColocPlots.py jd_hmc_p05_jdRna
python makeColocPlots.py  hpne_mc_100_hpneRna
python makeColocPlots.py hpne_mc_lt50_hpneRna
python makeColocPlots.py hpne_mc_gt50_hpneRna
python makeColocPlots.py hpne_hmc_p05_hpneRna


##copy all the coloc.plots and rename them to a separate folder
mkdir pics2
mkdir pics3
ls hpne_{,h}mc_*/*.png jd_{,h}mc_*/*.png | sed 's/\(.*\)\/colocPlot\(.*\.png\)/cp & pics3\/\1\2/'
```

```
#look at JUST the profile and use full MC set, (not 400k max)

qsub runPeakProfiles_noHeatmap.sh jd_mc_lt50_jdRna jd_mc_lt50.bed $BAMJD 3000
qsub runPeakProfiles_noHeatmap.sh jd_mc_gt50_jdRna jd_mc_gt50.bed $BAMJD 3000

qsub runPeakProfiles_noHeatmap.sh hpne_mc_lt50_hpneRna hpne_mc_lt50.bed $BAMHPNE
3000
qsub runPeakProfiles_noHeatmap.sh hpne_mc_gt50_hpneRna hpne_mc_gt50.bed $BAMHPNE
3000



#########################################
#correlation with stat32
#have to strip off the "chr" from the positions
cat jd_hmc_p05.bed | sed 's/^chr//' >| jd_hmc_p05_s3.bed
cat jd_mc_gt50.bed | sed 's/^chr//' >| jd_mc_gt50_s3.bed
cat hpne_hmc_p05.bed | sed 's/^chr//' >| hpne_hmc_p05_s3.bed
cat hpne_mc_gt50.bed | sed 's/^chr//' >| hpne_mc_gt50_s3.bed

qsub runPeakProfiles.sh jd_mc_gt50_jdStat3 jd_mc_gt50_s3.bed
/home/kpradha1/projects/sanchari/stat3Bams/jd13d-stat3.bam 3000
qsub runPeakProfiles.sh jd_hmc_p05_jdStat3 jd_hmc_p05_s3.bed
/home/kpradha1/projects/sanchari/stat3Bams/jd13d-stat3.bam 3000
qsub runPeakProfiles.sh hpne_mc_gt50_hpneStat3 hpne_mc_gt50_s3.bed
/home/kpradha1/projects/sanchari/stat3Bams/hpne-stat3.bam 3000
qsub runPeakProfiles.sh hpne_hmc_p05_hpneStat3 hpne_hmc_p05_s3.bed
/home/kpradha1/projects/sanchari/stat3Bams/hpne-stat3.bam 3000


qsub runPeakProfiles_noHeatmap.sh jd_mc_gt50_jdStat3 jd_mc_gt50_s3.bed
/home/kpradha1/projects/sanchari/stat3Bams/jd13d-stat3.bam 3000
qsub runPeakProfiles_noHeatmap.sh jd_hmc_p05_jdStat3 jd_hmc_p05_s3.bed
/home/kpradha1/projects/sanchari/stat3Bams/jd13d-stat3.bam 3000
qsub runPeakProfiles_noHeatmap.sh hpne_mc_gt50_hpneStat3 hpne_mc_gt50_s3.bed
/home/kpradha1/projects/sanchari/stat3Bams/hpne-stat3.bam 3000
qsub runPeakProfiles_noHeatmap.sh hpne_hmc_p05_hpneStat3 hpne_hmc_p05_s3.bed
/home/kpradha1/projects/sanchari/stat3Bams/hpne-stat3.bam 3000



python makeColocPlots.py jd_mc_gt50_jdStat3
python makeColocPlots.py jd_hmc_p05_jdStat3
python makeColocPlots.py hpne_mc_gt50_hpneStat3
python makeColocPlots.py hpne_hmc_p05_hpneStat3





##############################3
####scratch
```

```
#oxbs hmc sites, atac seq reads
###################################################33

#prep oxbs sites
cat hmc_sig0.01_commonSites1_jd.bed | cut -f1,2 | sed 's/.*/&\t+/' >| jd_hmc.txt
cat hmc_sig0.01_commonSites1_hpne.bed | cut -f1,2 | sed 's/.*/&\t+/' >|
hpne_hmc.txt


qsub runPeakProfiles.sh hpneHmcSites_hpneAtac hpne_hmc.txt
/home/kpradha1/projects/sanchari/j455/hpne_sorted_mdup.bam 3000
qsub runPeakProfiles.sh jdHmcSites_jdAtac jd_hmc.txt
/home/kpradha1/projects/sanchari/j455_j515/jd_sorted_mdup_merged.bam 3000




#oxbs mc sites, atac seq reads
###################################################33
#prep mc files
cat mc_commonSites1_jd.bed | cut -f1,2 | sed 's/.*/&\t+/' >| jd_mc.txt
cat mc_commonSites1_hpne.bed | cut -f1,2 | sed 's/.*/&\t+/' >| hpne_mc.txt

qsub runPeakProfiles.sh hpneMcSites_hpneAtac_400k hpne_mc.txt
/home/kpradha1/projects/sanchari/j455/hpne_sorted_mdup.bam 3000
qsub runPeakProfiles.sh jdMcSites_jdAtac_400k jd_mc.txt
/home/kpradha1/projects/sanchari/j455_j515/jd_sorted_mdup_merged.bam 3000



#mc files with rna
################################
#run annotation to get strand info




#stat3 sites, atac seq reads
###########################################33

#prepare the stat3 peaks from bed files

cat jd13d_stat3_summits.bed | cut -f1,2 | sed 's/.*/&\t+/' | grep -v "hs37d5" | sed
's/.*/chr&/' >| jd_stat3.txt
cat hpne_stat3_summits.bed | cut -f1,2 | sed 's/.*/&\t+/'  | grep -v "hs37d5" | sed
's/.*/chr&/' >| hpne_stat3.txt


qsub runPeakProfiles.sh hpneStat3Sites_hpneAtac hpne_stat3.txt
/home/kpradha1/projects/sanchari/j455/hpne_sorted_mdup.bam 3000
qsub runPeakProfiles.sh jdStat3Sites_jdAtac jd_stat3.txt
/home/kpradha1/projects/sanchari/j455_j515/jd_sorted_mdup_merged.bam 3000



#prepare stranded stat3 files with rna
#add chr to the front of each row
cat jd_stat3_summits_stranded.txt | sed 's/.*/chr&/' >
jd_stat3_summits_stranded1.txt
```

```
cat hpne_stat3_summits_stranded.txt | sed 's/.*/chr&/' >
hpne_stat3_summits_stranded1.txt


qsub runPeakProfiles.sh jdstat3Peaks_jdRna jd_stat3_summits_stranded1.txt
/home/kpradha1/projects/sanchari/rnaseq/JRN.AD0C99ABXX.lane_8_P0_I2.hg19.gsnap-rna-
alignment.bam 3000
qsub runPeakProfiles.sh hpnestat3Peaks_hpneRna hpne_stat3_summits_stranded1.txt
/home/kpradha1/projects/sanchari/rnaseq/HPNE.AC1FL7ACXX.lane_1_P0_I5.hg19.gsnap-
rna-alignment.bam 3000



#move all the pics to a separate folder
ls */colocPlot.png | sed 's$\(.*\)\/\(.*\)$cp & pics1\/\1\.png$' | bash


########################################################################



bamfile1="/home/kpradhan/Desktop/data/sanchari/pancreatic/rnaseq/bam/HPNE.AC1FL7ACX
X.lane_1_P0_I5.hg19.gsnap-rna-alignment.bam"

samtools view -b $bamfile1 chr1:561524-567524 | samtools mpileup -

samtools mpileup $bamfile1 -r chr1:561524-567524 | cut -f1-5


#run from my comp
python makeColocPlots.py test2

==> ./colocalization/runPeakProfiles.sh <==
#!/bin/bash
#$ -cwd
#$ -j n
#$ -l h_vmem=47G
#$ -pe smp 1
#$ -S /bin/bash
#$ -m e
#usage:
#qsub runPeakProfiles.sh asdlk asdflka
#probably going to need more memory
module load scipy/0.14.0/python.2.7.8-atlas-3.11.30
module load python/2.7.8/gcc.4.4.7
module load HTSeq/0.6.1/python.2.7.8
module load numpy/1.9.0/python.2.7.8-atlas-3.11.30
module load matplotlib/1.4.3/python.2.7.8



outputFolder="test3"
#siteFile="hpneHmc_strandedPeakSummits.txt"
siteFile="hpne_atac.txt"
#bamFile="/home/kpradha1/projects/sanchari/rnaseq/HPNE.AC1FL7ACXX.lane_1_P0_I5.hg19
.gsnap-rna-alignment.bam"
bamFile="/home/kpradhan/Desktop/hpc_home/projects/sanchari/rnaseq/HPNE.AC1FL7ACXX.l
ane_1_P0_I5.hg19.gsnap-rna-alignment.bam"
W="3000"
```

```
outputFolder=$1
siteFile=$2
bamFile=$3
W=$4


echo $outputFolder
echo $siteFile
echo $bamFile
echo $W


#make a folder for storing intermediate resulst
#otherwise, might run out of space on default /tmp
python getPeakProfiles_preload.py $outputFolder $siteFile $bamFile $W

#add a $ after the # if want email notification at job completion
# -M kith.pradhan@einstein.yu.edu



==> ./colocalization/viewPeakSums_oxbs.py <==
import numpy as np
import matplotlib.pyplot as plt
import random
import scipy.stats


ls


fols = [
    "jd_mc_lt50_jdAtac",
    "jd_mc_gt50_jdAtac",
    "hpne_mc_lt50_hpneAtac",
    "hpne_mc_gt50_hpneAtac",
    "jd_mc_lt50_jdRna",
    "jd_mc_gt50_jdRna",
    "hpne_mc_lt50_hpneRna",
    "hpne_mc_gt50_hpneRna"
]


f = fols[0]
f


for f in fols:
    plt.close()
    outFile = "pics4/peakSums_"+f+".png"
    x = np.load(f+"/peakSums.npy")
    plt.plot(np.arange(-3000, 3000, 1), x)
    plt.title(f)
    plt.savefig(outFile)
```

```
#layer the ATAC HMC and MC profiles on top of each other
#normalized by total count


x_mc = np.load("jd_mc_gt50_jdAtac/peakSums.npy")
x_mc = x_mc/np.mean(x_mc)
x_hmc = np.load("jd_hmc_p05_jdAtac/binnedPP.npy")
x_hmc = np.sum(x_hmc, 0)
x_hmc = x_hmc/np.mean(x_hmc)

plt.close()
plt.plot(x_mc/np.mean(x_mc), color="blue", label=">50% mc sites")
plt.plot(x_hmc/np.mean(x_hmc), color="red", label = "0.05 hmc sites")
plt.title("JD: coloc OxBS, atac-seq")
plt.legend()
plt.savefig("pics4/jd_oxbs_atac.png")


x_mc = np.load("hpne_mc_gt50_hpneAtac/peakSums.npy")
x_mc = x_mc/np.mean(x_mc)
x_hmc = np.load("hpne_hmc_p05_hpneAtac/binnedPP.npy")
x_hmc = np.sum(x_hmc, 0)
x_hmc = x_hmc/np.mean(x_hmc)

plt.close()
plt.plot(x_mc/np.mean(x_mc), color="blue", label=">50% mc sites")
plt.plot(x_hmc/np.mean(x_hmc), color="red", label = "0.05 hmc sites")
plt.title("HPNE: coloc OxBS, atac-seq")
plt.legend()
plt.savefig("pics4/hpne_oxbs_atac.png")


#devolop statistical test that shows whether two profiles have similar peak
#tighter or looser peak
#
#sample from distributions
#Do F test to show variances are different


#treat the profile as a probability density distribution
#sample n values from the distribution
def sampleFromProfile(pos, prof, n=1000):
    #step one, contruct prob dist
    csum = np.cumsum(prof)
    #normalize so last element is prob 1.0
    prob = csum/csum[-1]
    #randomly sample 1000 from uniform dist
    ix = [min(np.where(prob > random.random())[0]) for i in np.arange(n)]
    samps = pos[ix]
    return samps


def kurtosisPerm(samps_mc, samps_hmc, K=10000):
    #get stat for difference in kurtosis
    kmc = scipy.stats.kurtosis(samps_mc)
    khmc = scipy.stats.kurtosis(samps_hmc)
    kstat = khmc - kmc
    #
    #combine the samples
    both = np.array([samps_mc, samps_hmc]).ravel()
    res = []
    for i in range(K):
        if i % 100 == 0:
            print i
        #shuffle the samples
        np.random.shuffle(both)
        #compute kurtosis on the two halves of the array
        k1 = scipy.stats.kurtosis(both[:len(samps_mc)])
        k2 = scipy.stats.kurtosis(both[len(samps_hmc):])
        #save the stat of difference of kurtosis
        res.append(k2-k1)
    #
    res = np.array(res)
    hits = np.sum(abs(kstat ) < np.abs(res))
    pval = hits / res.shape[0]
    plt.figure()
    plt.hist(res, 100)
    plt.axvline(x=kstat, color="red")
    plt.xlabel("Null Distribution (K=%d)"%K)
    return [kmc, khmc, pval]



mcFile = "jd_mc_gt50_jdAtac/peakSums.npy"
hmcFile = "jd_hmc_p05_jdAtac/binnedPP.npy"

#is the hmc profile tighter than mc?
def testTightness(mcFile, hmcFile):
    #prep the profiles, divide by mean so they on same scale
    x_mc = np.load(mcFile)
    x_mc_sum = np.sum(x_mc)
    x_mc = x_mc/np.mean(x_mc)
    x_hmc = np.load(hmcFile)
    #hmc file is the binned collection, not the sum profile
    x_hmc_sum = np.sum(x_hmc)
    x_hmc = np.sum(x_hmc, 0)
    x_hmc = x_hmc/np.mean(x_hmc)
    #get the samples
    pos = np.arange(-3000,3000,1)
    samps_mc = sampleFromProfile(pos, x_mc, 10000)
    samps_hmc = sampleFromProfile(pos, x_hmc, 10000)
    #plot the figures to make sure data makes sense
    #the orginal read sum profile
    plt.figure()
    plt.plot(pos, x_mc, label="mc", color="red")
    ax = plt.plot(pos, x_hmc, label="hmc", color="blue")
    plt.ylim(ymin = 0)
    plt.legend()
    #
    #histogram of the sampled read sum profile
    plt.figure()
    plt.hist(samps_mc, 100, alpha=0.5, label="mc")
    plt.hist(samps_hmc, 100, alpha=0.5, label="hmc")
    plt.legend()
    #
    #kurtosis test
    return kurtosisPerm(samps_mc, samps_hmc)
    #
    #variance test
    #mc_var = np.var(samps_mc)
    #hmc_var = np.var(samps_hmc)
    ##perform the levene's test on variance to see which
    ##has less (tighter), and which has more (looser)
    #res = scipy.stats.levene(samps_hmc, samps_mc)
    #return [mc_var, hmc_var, res.pvalue]



jd_tight = testTightness("jd_mc_gt50_jdAtac/peakSums.npy",
"jd_hmc_p05_jdAtac/binnedPP.npy")
hpne_tight = testTightness("hpne_mc_gt50_hpneAtac/peakSums.npy",
"hpne_hmc_p05_hpneAtac/binnedPP.npy")





jd_tight
hpne_tight


kmc = scipy.stats.kurtosis(samps_mc)
khmc = scipy.stats.kurtosis(samps_hmc)
abs(kmc ) - abs(khmc)

both = np.array([samps_mc, samps_hmc]).ravel()
res = []
for i in range(10000):
    if i % 100 == 0:
        print i
    np.random.shuffle(both)
    k1 = scipy.stats.kurtosis(both[:10000])
    k2 = scipy.stats.kurtosis(both[10000:])
    res.append(abs(k1 ) - abs(k2))

plt.hist(res, 100)

#F = min(np.var(samps_hmc) / np.var(samps_mc), np.var(samps_mc) / np.var(samps_hmc)
)
#2*scipy.stats.f.cdf(F, len(samps_mc)-1, len(samps_hmc)-1)
#p_value = scipy.stats.f.cdf(F, len(samps_mc)-1, len(samps_hmc)-1)


==> ./colocalization/getPeakProfiles_setImpl.py <==
#from multiprocessing import Pool
from functools import partial
from itertools import chain
import time
import peakProfiles as pp
import numpy as np
import matplotlib.pyplot as plt
import os
import sys
import HTSeq as ht
```

```
#$reload(pp)


outFolder = "test4"
#siteFile  = "hpneHmc_strandedPeakSummits.txt"
siteFile  = "hpne_atac.txt"
#bamfile1  =
"/home/kpradha1/projects/sanchari/rnaseq/HPNE.AC1FL7ACXX.lane_1_P0_I5.hg19.gsnap-
rna-alignment.bam"
bamfile1 =
"/home/kpradhan/Desktop/hpc_home/projects/sanchari/rnaseq/HPNE.AC1FL7ACXX.lane_1_P0
_I5.hg19.gsnap-rna-alignment.bam"
W         = 3000

#command line arguments
outFolder = sys.argv[1]
siteFile  = sys.argv[2]
bamfile1  = sys.argv[3]
W         = int(sys.argv[4])

print ("outfolder:  %s" % outFolder)
print ("siteFile:  %s" % siteFile)
print ("bamfile:  %s" % bamfile1)
print ("W:  %d" % W)



#load the sites
sites = pp.getPeakSitesFromStrandedBed(siteFile)
#first thing...check that the sites match the notation of the bam file
nMatch = pp.checkSiteNotation(sites, bamfile1)
print("number of sites matching bam chroms: %d" % nMatch)


#create a genomic array of sets to hold the site info
gas = ht.GenomicArrayOfSets("auto", stranded=True)
?ht.GenomicInterval()

def site2GI(site):
    return ht.GenomicInterval(""+site[0], site[2]-W, site[2]+W, ""+site[1])

#fill the arary of sets
for i, site in enumerate(sites[:10]):
    iv = site2GI(site)
    gas[iv] += str(i)

for g in gas.steps():
    print g


gas['1']

for iv, val in gas[ iv1 ].steps():
    print iv, val

site = sites[1]
iv1 = ht.GenomicInterval(""+site[0], site[2]-W, site[2]+W, ""+site[1])

#get regions spanned by site 1
```

```
site-W,


#read the bam file read by read filling in genomic array of sets


#look at coverages of sites
```

```
==> ./sanchari/rnaseq/commands_ngs.sh <==

#tss, tes, genebody, exon, cgi, enhancer, dhs
jdbam=JRN.AD0C99ABXX.lane_8_P0_I2.hg19.gsnap-rna-alignment.bam
hpnebam=HPNE.AC1FL7ACXX.lane_1_P0_I5.hg19.gsnap-rna-alignment.bam

#ngs.plot.r -P 4 -G hg19 -R genebody -C $jdbam -O jdplot_genebody -T jdRna -L 5000
#ngs.plot.r -P 4 -G hg19 -R genebody -C $hpnebam -O hpneplot_genebody -T hpneRna -L
5000

#
ngs.plot.r -P 4 -G hg19 -R tss -C $jdbam -O jdplot_rna_tss -T jdRna -L 5000
ngs.plot.r -P 4 -G hg19 -R tes -C $jdbam -O jdplot_rna_tes -T jdRna -L 5000
ngs.plot.r -P 4 -G hg19 -R exon -C $jdbam -O jdplot_rna_exon -T jdRna -L 5000
ngs.plot.r -P 4 -G hg19 -R cgi -C $jdbam -O jdplot_rna_cgi -T jdRna -L 5000
#
ngs.plot.r -P 4 -G hg19 -R tss -C $hpnebam -O hpneplot_rna_tss -T hpneRna -L 5000
ngs.plot.r -P 4 -G hg19 -R tes -C $hpnebam -O hpneplot_rna_tes -T hpneRna -L 5000
ngs.plot.r -P 4 -G hg19 -R exon -C $hpnebam -O hpneplot_rna_exon -T hpneRna -L 5000
ngs.plot.r -P 4 -G hg19 -R cgi -C $hpnebam -O hpneplot_rna_cgi -T hpneRna -L 5000

##run these two again
#ngs.plot.r -P 4 -G hg19 -R enhancer -C $jdbam -O jdplot_enhancer -T jdAtac -L 5000
#ngs.plot.r -P 4 -G hg19 -R dhs -C $jdbam -O jdplot_dhs -T jdAtac -L 5000
#ngs.plot.r -P 4 -G hg19 -R enhancer -C $hpnebam -O hpneplot_enhancer -T hpneAtac
-L 5000
#ngs.plot.r -P 4 -G hg19 -R dhs -C $hpnebam -O hpneplot_dhs -T hpneAtac -L 5000


==> ./sanchari/atac_hmc_rna/annotatePeaks.r <==
library("ChIPpeakAnno")
#source("https://bioconductor.org/biocLite.R")
#biocLite("EnsDb.Hsapiens.v75")
library(EnsDb.Hsapiens.v75)

options(stringsAsFactors=F)
```

```r
#load in all the gene info from hg19
knownGenes <- genes(EnsDb.Hsapiens.v75)




#peaksFile must be the peak summit file from macs2
peaksFile = "hpne_stat3_summits.bed"
addStrandToPeaks <- function(peaksFile, newFile){
      #read in summits and convert to ranges
      peaks = read.table(peaksFile)
      #if the 3rd column is not the peak ends
      if (is.na(as.numeric(peaks[1,3]))){
            #use the peaks starts + 1
            ends = peaks[,2] + 1
      }else{
            ends = peaks[,3]
      }
      rd <- RangedData(IRanges(start = peaks[,2],
            end = ends), space = peaks[,1])
      ranges.peaks = toGRanges(rd, format="RangedData")

      #annotate ranges with the knownGenes database
      anno <- annotatePeakInBatch(ranges.peaks, AnnotationData=knownGenes)

      #make sure to only take 1 result
      x = anno[!(duplicated(anno$peak))]
      #append the strand info to last column
      y = cbind(peaks, strand=as.character(x$feature_strand))
      y$strand = as.character(x$feature_strand)
      y$strand[is.na(y$strand)] = "*"
      #write the new stranded peak file
      write.table(newFile, x=y, col.names=F, row.names=F, quote=F, sep="\t")
}


#stat3
addStrandToPeaks("test1_jdmc_top100k.txt", "test1_jdmc_top100k_stranded.txt")
addStrandToPeaks("test1_jdhmc.txt", "test1_jdhmc_stranded.txt")

addStrandToPeaks("hpne_stat3_summits.bed", "hpne_stat3_summits_stranded.txt")
addStrandToPeaks("jd13d_stat3_summits.bed", "jd_stat3_summits_stranded.txt")

peaksFile = "peaks.atac.jd.gain.1k.txt"
addStrandToPeaks("peaks.atac.jd.gain.1k.txt", "peaks.atac.jd.gain.1k_stranded.txt")
addStrandToPeaks("peaks.atac.jd.loss.1k.txt", "peaks.atac.jd.loss.1k_stranded.txt")


addStrandToPeaks("/home/kpradhan/Desktop/hpc_home/projects/sanchari/j455_j515/macs_
Chpne_sorted_mdup_Tjd_sorted_mdup_merged/Chpne_sorted_mdup_Tjd_sorted_mdup_merged_s
ummits.bed", "jd-hpne_strandedPeakSummits.txt")
addStrandToPeaks("/home/kpradhan/Desktop/hpc_home/projects/sanchari/j455_j515/macs_
Cjd_sorted_mdup_merged_Thpne_sorted_mdup/Cjd_sorted_mdup_merged_Thpne_sorted_mdup_s
ummits.bed", "hpne-jd_strandedPeakSummits.txt")


addStrandToPeaks("p3.txt", "p3_stranded.txt")
addStrandToPeaks("hmc_sig0.001_commonSites1_hpne.bed",
"hpneHmc_strandedPeakSummits_0.001.txt")
addStrandToPeaks("hmc_sig0.001_commonSites1_jd.bed",
"jdHmc_strandedPeakSummits_0.001.txt")

addStrandToPeaks("hmc_sig0.01_commonSites1_hpne.bed",
"hpneHmc_strandedPeakSummits.txt")
addStrandToPeaks("hmc_sig0.01_commonSites1_jd.bed",
"jdHmc_strandedPeakSummits.txt")

addStrandToPeaks("/home/kpradhan/Desktop/hpc_home/projects/sanchari/j455/macs_hpne_
sorted_mdup/hpne_sorted_mdup_summits.bed", "hpne_strandedPeakSummits.txt")
addStrandToPeaks("/home/kpradhan/Desktop/hpc_home/projects/sanchari/j455_j515/macs_
jd_sorted_mdup_merged/jd_sorted_mdup_merged_summits.bed",
"jd_strandedPeakSummits.txt")


addStrandToPeaks("/home/kpradhan/mnt/hpc_home/projects/colocalization/jd_hmc.txt",
"jd_strandedHmc.txt")
addStrandToPeaks("/home/kpradhan/mnt/hpc_home/projects/colocalization/hpne_hmc.txt"
, "hpne_strandedHmc.txt")



#######################################################33
#scratch









rd <- RangedData(IRanges(start = peaks.hpne[,2],
      end = peaks.hpne[,3]), space = peaks.hpne[,1])
ranges.hpne = toGRanges(rd, format="RangedData")

anno.hpne <- annotatePeakInBatch(ranges.hpne, AnnotationData=knownGenes)

x = anno.hpne[!(duplicated(anno.hpne$peak))]
#write the peaks with strand info
x$feature_strand
slot(strand(x), "values")
str(ranges(x))
y = cbind(peaks.hpne, strand=as.character(x$feature_strand))
y$strand = as.character(y$strand)
class(y$strand)
sum(is.na(y$strand))
y$strand[is.na(y$strand)] = "*"

write.table("peaks_test1.bed", x=y, col.names=F, row.names=F, quote=F, sep="\t")

head(y)

start(ranges(x))
end(ranges(x))
head(x)
```

```
str(x)
names(x)
x$ranges
x["ranges"]
ranges(x)
dim(peaks.hpne)



hmc_sig0.01_commonSites1_hpne.bed

hmc.hpne = read.table("hmc_sig0.01_commonSites1_hpne.bed")

peaks.hpne =
read.table("/home/kpradhan/Desktop/hpc_home/projects/sanchari/j455/macs_hpne_sorted
_mdup/hpne_sorted_mdup_summits.bed")
peaks.hpne =
read.table("/home/kpradhan/Desktop/hpc_home/projects/sanchari/j455/macs_hpne_sorted
_mdup/hpne_sorted_mdup_peaks.xls", header=T)
head(peaks.hpne)

peaks.jd =
read.table("/home/kpradhan/Desktop/hpc_home/projects/sanchari/j455_j515/macs_jd_sor
ted_mdup_merged/jd_sorted_mdup_merged_summits.bed")

peaks.jdMinushpne =
read.table("/home/kpradhan/Desktop/hpc_home/projects/sanchari/j455_j515/macs_Chpne_
sorted_mdup_Tjd_sorted_mdup_merged/Chpne_sorted_mdup_Tjd_sorted_mdup_merged_summits
.bed")
peaks.hpneMinusjd =
read.table("/home/kpradhan/Desktop/hpc_home/projects/sanchari/j455_j515/macs_Cjd_so
rted_mdup_merged_Thpne_sorted_mdup/Cjd_sorted_mdup_merged_Thpne_sorted_mdup_summits
.bed")


==> ./sanchari/atac_hmc_rna/overlays/plotOverlays.r <==
#1.  jd atac hmc profile
#       overlayed with
#     jd atac no hmc profile
#
#
#2.  hpne atac hmc profile
#       overlayed with
#     hpne atac no hmc profile
#



#plot(regcovMat, type='l', xaxt="n")
#axis(side=1, at=xticks$pos, labels = xticks$lab)


#load in the jd peak sets

load("jd_strandedAtacPeaks_hmc/avgprof.RData")
jd.hmc.regcovMat = regcovMat
jd.hmc.xticks = xticks

load("jd_strandedAtacPeaks_nohmc/avgprof.RData")
```

```
jd.nohmc.regcovMat = regcovMat
jd.nohmc.xticks = xticks


png("jd_atac_rna_hmc.png")
plot(jd.hmc.regcovMat, type='l', xaxt="n", xlab="Genomic Region (5' -> 3')",
ylab="Read count Per Million mapped reads", col="blue", lwd=3)
lines(jd.nohmc.regcovMat, type='l', xaxt="n", col="red", lw=3)
axis(side=1, at=jd.hmc.xticks$pos+1, labels = jd.hmc.xticks$lab)
abline(v=floor(pts/2)+1, col=rgb(.5,.5,.5))
legend("topleft", c("jd.hmc.atac.rna", "jd.nohmc.atac.rna"), col=c("blue", "red"),
pch=16)
dev.off()


load("hpne_strandedAtacPeaks_hmc/avgprof.RData")
hpne.hmc.regcovMat = regcovMat
hpne.hmc.xticks = xticks

load("hpne_strandedAtacPeaks_nohmc/avgprof.RData")
hpne.nohmc.regcovMat = regcovMat
hpne.nohmc.xticks = xticks


png("hpne_atac_rna_hmc.png")
plot(hpne.hmc.regcovMat, type='l', xaxt="n", xlab="Genomic Region (5' -> 3')",
ylab="Read count Per Million mapped reads", col="blue", lwd=3)
lines(hpne.nohmc.regcovMat, type='l', xaxt="n", col="red", lw=3)
axis(side=1, at=hpne.hmc.xticks$pos+1, labels = hpne.hmc.xticks$lab)
abline(v=floor(pts/2)+1, col=rgb(.5,.5,.5))
legend("topleft", c("hpne.hmc.atac.rna", "hpne.nohmc.atac.rna"), col=c("blue",
"red"), pch=16)
dev.off()

==> ./sanchari/atac_hmc_rna/examineSites.r <==
options(stringsAsFactors=F)

#look at the atac-seq sites
#     separate into 2 sets, those that have hmc sites within
#     those that don't
#     maybe break into sets based on the percentage of hmc sites

#compare these two site sites to rna expression
#ngs.plot


#will need to add strand info




#load in the atac seq peaks
atac.jd = read.table("jd_strandedAtacPeakSummits.txt")
atac.hpne = read.table("hpne_strandedAtacPeakSummits.txt")
head(atac.jd)
head(atac.hpne)


#load in the hmc sites
hmc.jd = read.table("sites_fixed2_jdHmc_p05.txt")
hmc.hpne = read.table("sites_fixed2_hpneHmc_p05.txt")
head(hmc.jd)
head(hmc.hpne)


#how many sites have hmc sites within them?
x = atac.jd
y = hmc.jd

#work chrom by chrom
w = 3000
unique(x[,1])

#x are the atac peak summits
#y are the hmc sites
#w is the width
#for each atac peak, return the number of hmc sites within w bases
getOverlapPeaks <- function(x, y, w =3000){
      peaks = sapply(1:nrow(x), function(i){
            if (i %% 1000 == 0){
                  print (i)
            }
            ix = y[,1] == x[i,1]
            sum((y[ix,2] > x[i,2]-w) & (y[ix,2] < x[i,3]+w))
      })
      peaks
}

hmcPeaks.jd = getOverlapPeaks(atac.jd, hmc.jd, 3000)
hmcPeaks.hpne = getOverlapPeaks(atac.hpne, hmc.hpne, 3000)

write.table(file="jd_strandedAtacPeaks_hmc.txt", x=atac.jd[hmcPeaks.jd > 0,],
row.names=F, col.names=F, quote=F, sep="\t")
write.table(file="jd_strandedAtacPeaks_nohmc.txt", x=atac.jd[hmcPeaks.jd == 0,],
row.names=F, col.names=F, quote=F, sep="\t")
write.table(file="hpne_strandedAtacPeaks_hmc.txt", x=atac.hpne[hmcPeaks.hpne > 0,],
row.names=F, col.names=F, quote=F, sep="\t")
write.table(file="hpne_strandedAtacPeaks_nohmc.txt", x=atac.hpne[hmcPeaks.hpne ==
0,], row.names=F, col.names=F, quote=F, sep="\t")

dim(x)
dim(y)


#separate the atac sites into those with/without hmc sites


==> ./sanchari/atac_hmc_rna/commands1.sh <==

bam1=/home/kpradhan/Desktop/data/sanchari/pancreatic/rnaseq/bam/JRN.AD0C99ABXX.lane
_8_P0_I2.hg19.gsnap-rna-alignment.bam
bam2=/home/kpradhan/Desktop/data/sanchari/pancreatic/rnaseq/bam/HPNE.AC1FL7ACXX.lan
e_1_P0_I5.hg19.gsnap-rna-alignment.bam



#atacseq peaks with strand information
#peak1=jd_strandedAtacPeakSummits.txt
#peak2=hpne_strandedAtacPeakSummits.txt
```

```
#all
#time ngs.plot.r -SS same -G hg19 -R bed -E $peak1 -C $bam1 -O peaks.atac.jd -T
peaks.atac.jd -P 2 -L 3000
#time ngs.plot.r -SS same -G hg19 -R bed -E $peak2 -C $bam2 -O peaks.atac.hpne -T
peaks.atac.hpne -P 2 -L 3000

#separated by hmc
time ngs.plot.r -SS same -G hg19 -R bed -E jd_strandedAtacPeaks_hmc.txt -C $bam1 -O
jd_strandedAtacPeaks_hmc -T peaks.atac.jd.hmc -P 2 -L 3000
time ngs.plot.r -SS same -G hg19 -R bed -E jd_strandedAtacPeaks_nohmc.txt -C $bam1
-O jd_strandedAtacPeaks_nohmc -T peaks.atac.jd.nohmc -P 2 -L 3000
time ngs.plot.r -SS same -G hg19 -R bed -E hpne_strandedAtacPeaks_hmc.txt -C $bam2
-O hpne_strandedAtacPeaks_hmc -T peaks.atac.hpne.hmc -P 2 -L 3000
time ngs.plot.r -SS same -G hg19 -R bed -E hpne_strandedAtacPeaks_nohmc.txt -C
$bam2 -O hpne_strandedAtacPeaks_nohmc -T peaks.atac.hpne.nohmc -P 2 -L 3000

==> ./sanchari/homer_panc/runHomer.sh <==
#!/bin/bash
#$ -cwd
#$ -j n
#$ -l h_vmem=4G
#$ -pe smp 4
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

#add thep path to homer and seqlogo
#PATH=$PATH:/home/kpradha1/programs/homer/bin
PATH=$PATH:/home/kpradha1/programs/weblogo
module load blatSuite/34
module load HOMER/4.7

outname1=$1
peaks1=$2


findMotifsGenome.pl $peaks1 hg19 $outname1 -len 0 -preparsedDir
/home/kpradha1/programs/homer_preparse -preparse -p 8


==> ./sanchari/homer_panc/commands1.sh <==


./runHomer.sh test1 atac_hpne_only.bed
qsub ./runHomer.sh test1 atac_hpne_only.bed
qsub ./runHomer.sh test2 atac_hpne_only.bed
qsub ./runHomer.sh test3 atac_hpne_only.bed

==> ./sanchari/filterCpg/calcRunningAve.pl <==
#!/usr/bin/perl

use strict;
use warnings;

my $N = 1;
my $curAve = 0;
while (<STDIN>){
      my @fields = split '#';
      my ($a, $b) = split ' ', $fields[1];
      #TODO:  Can we not look at less than 100 percent methylation.
      #TODO:  Like maybe 50 and above?
      $curAve = $curAve + (($a/$b) - $curAve)/$N;
      $N++;
      if ($N % 100000 == 0){
            #print "$a\t$b\t$curAve\t$N\n";
            print STDERR "$curAve\t$N\n";
      }
}

print "$curAve\t$N\n";

==> ./sanchari/filterCpg/commands2_mcPercCpg.sh <==
#get the average methylation percentage
#regular C(not CpG) sites
#CpG sites



#run the bedgraph output through cpg filter
fol=~/mnt/hpc_home/projects/sanchari/j432_j446/conversionCounts/
ls $fol


file=${fol}JD_BS_22.txt.gz
echo $file


#separate the C's from CpG's
for file in `ls ${fol}*JD_BS*.gz`
do
    echo $file
    #The CpG sites
    echo "cpg filtering..."
    zcat $file | python ../filterCpg.py | awk '{printf "%s\t%d#%d\t%d\n",
$1,$2,$3,$4}' >| temp_cpg.txt
    #all sites
    zcat $file | awk '{printf "%s\t%d#%d\t%d\n", $1,$2,$3,$4}' >| temp_allC.txt
    #append just the C sites
    echo "writing C's..."
    join -v1 -t"#" <(sort -k1,1 -t"#" temp_allC.txt) <(sort -k1,1 -t"#"
temp_cpg.txt) >> JD_BS_C.txt
    #append the CpG sites
    echo "writing CpG's..."
    cat temp_cpg.txt >> JD_BS_CpG.txt
done

#now compute average methylation between C's and CpG's


cat Cpgfiles/JD_BS_CpG.txt | ./calcRunningAve.pl >| meanCpg.txt
cat Cpgfiles/JD_BS_C.txt | ./calcRunningAve.pl >| meanC.txt


==> ./sanchari/filterCpg/filterCpg.py <==
#!/usr/bin/python
import numpy as np
```

```
from itertools import islice
import os
import sys

faFol = "/home/kpradhan/Desktop/hpc_home/projects/hg19/chroms"
faFol = "/home/kpradha1/projects/hg19/chroms"
posFile =
"/home/kpradhan/Desktop/hpc_home/projects/sanchari/j432_j446/commonSites/common4.tx
t"
outFile = "test1.txt"

def loadFasta(faFile):
    with open(faFile) as fp:
        fa = ""
        #load the first line which should start with >
        line = fp.readline()
        #print (line)
        fa = "".join([line.rstrip("\n") for line in fp.readlines()])
        return fa




#open the output file
#out = open(outFile, "w")

curChrom = ""
#for line in islice(open(posFile), 1000):
#process chromosomal positions line by line
#for line in open(posFile):
for line in sys.stdin:
    line = line.rstrip("\n")
    #print line
    #first two fields are the chromosome and position
    #chrom, pos = line.split("\t")[0:2]
    chrom, pos = line.split()[0:2] #split on whitespace
    pos = int(filter(str.isalnum, pos)) #convert to integer
    #make sure the proper fastq is loaded
    if (chrom != curChrom):
        #load the entire fastq chrom
        faFile = os.path.join(faFol, chrom+".fa")
        ref = loadFasta(faFile)
        #current
        curChrom = chrom
    #if the current position is a cpg, then print it
    #print ref[pos:(pos+2)]
    b2 = ref[pos:(pos+2)].lower()
    if b2 == "cg":
        #out.write("%s\t%s\n"%(line, b2))
        sys.stdout.write("%s\t%s\n"%(line, b2))

#out.close()




==> ./sanchari/filterCpg/commands1.sh <==
##sort the files first
#sort -k1,1 -k2,2n
```

```
/home/kpradhan/Desktop/hpc_home/projects/sanchari/j432_j446/commonSites/common4.txt
> sorted_common4.txt
#
#sort  -k1,1 -k2,2n
/home/kpradhan/Desktop/hpc_home/projects/sanchari/j432_j446/hpc_mcSites/common4_mcS
ites_HPNE.txt > sorted_common4_mcSites_HPNE.txt
#sort  -k1,1 -k2,2n
/home/kpradhan/Desktop/hpc_home/projects/sanchari/j432_j446/hpc_mcSites/common4_mcS
ites_JD.txt > sorted_common4_mcSites_JD.txt
#
#sort  -k1,1 -k2,2n
/home/kpradhan/Desktop/hpc_home/projects/sanchari/j432_j446/hmcSites/common4_hmcSit
es_HPNE.txt > sorted_common4_hmcSites_HPNE.txt
#sort  -k1,1 -k2,2n
/home/kpradhan/Desktop/hpc_home/projects/sanchari/j432_j446/hmcSites/common4_hmcSit
es_JD.txt > sorted_common4_hmcSites_JD.txt




#
#time cat sorted_common4.txt | ./filterCpg.py >| common4cpg.txt
#
#time cat sorted_common4_mcSites_HPNE.txt | ./filterCpg.py >|
common4cpg_mcSites_HPNE.txt
#time cat sorted_common4_mcSites_JD.txt | ./filterCpg.py >|
common4cpg_mcSites_JD.txt
#
#time cat sorted_common4_hmcSites_HPNE.txt | ./filterCpg.py >|
common4cpg_hmcSites_HPNE.txt
#time cat sorted_common4_hmcSites_JD.txt | ./filterCpg.py >|
common4cpg_hmcSites_JD.txt


#get the diff sets
#sites in hpne but no jd

##mc
#comm -23 --check-order <(cut -f1-2 common4cpg_mcSites_JD.txt | sort) <(cut -f1-2
common4cpg_mcSites_HPNE.txt | sort) >| common4cpg_mcSites_JD-HPNE.txt
#comm -13 --check-order <(cut -f1-2 common4cpg_mcSites_JD.txt | sort) <(cut -f1-2
common4cpg_mcSites_HPNE.txt | sort) >| common4cpg_mcSites_HPNE-JD.txt
#
##hmc
#comm -23 --check-order <(cut -f1-2 common4cpg_hmcSites_JD.txt | sort) <(cut -f1-2
common4cpg_hmcSites_HPNE.txt | sort) >| common4cpg_hmcSites_JD-HPNE.txt
#comm -13 --check-order <(cut -f1-2 common4cpg_hmcSites_JD.txt | sort) <(cut -f1-2
common4cpg_hmcSites_HPNE.txt | sort) >| common4cpg_hmcSites_HPNE-JD.txt



#get #c's  #cpgs in common for JD
#get #meth's for c's and cpgs

#merge with common, and cpg
mc_sites=/home/kpradhan/mnt/hpc_home/projects/sanchari/j432_j446/hpc_mcSites/mcSite
s_JD_all.txt
common_cpg_sites=sorted_JDcommon4cpg.txt
common_c_sites=/home/kpradhan/mnt/hpc_home/projects/sanchari/j432_j446/commonSites/
```

```
sorted_common4_JD.txt

ls $common_c_sites
ls $common_cpg_sites

head $mc_sites
head $common_c_sites
head $common_cpg_sites

cat $mc_sites | awk '{printf("%s\t%d#%s\n", $1,$2,$0)}' | head


cat $mc_sites | awk '{printf("%s\t%d#%s\n", $1,$2,$0)}' | head

cat $common_cpg_sites | cut -f1-2

cat $mc_sites | awk '{printf("%s\t%d#%s\n", $1,$2,$0)}' | head

#default sort doesn't work properly!
sort -t"#" -k1b,1 mc1.txt > mc2.txt
cat $common_cpg_sites | cut -f1-2 | sort >| cpg1.txt
cat $common_c_sites | cut -f1-2 | sort >| c1.txt

#cpg sites
join mc2.txt cpg1.txt -t'#' >| JdMcSites_JDcommon4Cpg.txt
join mc2.txt c1.txt -t'#' >| JdMcSites_JDcommon4C.txt

wc -l JdMcSites_JDcommon4Cpg.txt
wc -l JdMcSites_JDcommon4C.txt

join mc1.txt <(cat $common_cpg_sites | cut -f1-2 | sort) -t "#" >|
JdMcSites_JDcommon4Cpg.txt

#c sites
join <(cat $mc_sites | awk '{printf("%s\t%d#%s\n", $1,$2,$0)}' | sort -t'#' -k1)
<(cat $common_c_sites | cut -f1-2 | sort) -t "#"  >| JdMcSites_JDcommon4C.txt


==> ./sanchari/filterCpg/filterCpg.sh <==
#!/bin/bash
#$ -cwd
#$ -j n
#$ -l h_vmem=4G
#$ -pe smp 1
#$ -S /bin/bash
#$ -m e
#   #$ -M kith.pradhan@einstein.yu.edu
module load python/2.7.8/gcc.4.4.7
module load numpy/1.9.0/python.2.7.8

inFile=$1
outFile=$2


python /home/kpradha1/projects/sanchari/filterCpg/filterCpg.py < $inFile > $outFile
```

```
==> ./sanchari/j455/markDuplicates.sh <==
#!/bin/bash
#$ -cwd
#$ -j n
#$ -l h_vmem=4G
#$ -pe smp 4
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

bam=$1
bname="${bam%.*}"

java -jar $(which BuildBamIndex.jar) INPUT=${bam}
java -jar $(which MarkDuplicates.jar) INPUT=${bam} OUTPUT=${bname}_mdup.bam
METRICS_FILE=${bname}_mdMetric.txt
java -jar $(which BuildBamIndex.jar) INPUT=${bname}_mdup.bam



==> ./sanchari/j455/se_macs_jd_sorted_mdup/jd_sorted_mdup_model.r <==
# R script for Peak Model
#  -- generated by MACS
p <-
c(0.0242523627067,0.0284508514944,0.0325633644024,0.0368478290696,0.0409890006042,0
.0416911369543,0.0419347352799,0.0417627835207,0.0417986068039,0.0416409843579,0.04
16409843579,0.0415693377916,0.0418917473401,0.0414403739721,0.0413687274058,0.04169
11369543,0.0417197955809,0.0418201007738,0.0416768076411,0.0417699481773,0.04180577
14605,0.0420565344427,0.0420636990993,0.0424004379611,0.0425365664372,0.04235028536
47,0.0422141568886,0.0422929681116,0.0423861086479,0.0423144620815,0.0422213215453,
0.0420493697861,0.0417842774906,0.0416553136712,0.0416051610747,0.0415908317615,0.0
415693377916,0.0417699481773,0.0418129361171,0.0418774180268,0.0423144620815,0.0424
290965877,0.0423359560514,0.0426082130035,0.0426297069734,0.0426153776602,0.0428016
587326,0.042579554377,0.0423144620815,0.042364614678,0.042500743154,0.0425938836903
,0.0427300121663,0.0429592811786,0.0428159880459,0.0430165984317,0.043009433775,0.0
432960200404,0.0433175140103,0.0436972408119,0.0434894657695,0.0433318433236,0.0434
751364563,0.0431455622511,0.0433318433236,0.0433676666067,0.0434679717996,0.0438405
339446,0.0443062366258,0.0438691925711,0.0440339796737,0.0440196503605,0.0441701081
498,0.0443778831922,0.0445999875478,0.044342059909,0.0442990719692,0.0438978511977,
0.0434823011129,0.0435037950828,0.0434751364563,0.04412712021,0.0438763572278,0.043
9838270773,0.0437115701252,0.0439336744808,0.043969497764,0.0438476986012,0.0443635
538789,0.0446214815177,0.0443205659391,0.0442274254028,0.0444925176983,0.0448579151
867,0.045438252374,0.0454740756572,0.0457105093261,0.0460400835313,0.0459827662782,
0.0464627982727,0.0470861233999,0.0468568543876,0.0470861233999,0.0469499949238,0.0
466490793452,0.0466777379717,0.0467565491947,0.0471004527132,0.0470861233999,0.0469
85818207,0.046692067285,0.0466992319416,0.0465487741523,0.0465344448391,0.046577432
7789,0.0468855130141,0.0471004527132,0.0468998423274,0.0470861233999,0.046985818207
,0.0468711837009,0.0470073121769,0.0473010630989,0.0474228622617,0.0474443562316,0.
0475374967678,0.047437191575,0.0474658502015,0.0473942036352,0.047494508828,0.04811
06692986,0.0482539624313,0.0482253038047,0.0480676813588,0.0480963399853,0.04813216
32685,0.0486766771727,0.0489775927513,0.0490993919141,0.0492856729866,0.04907789794
42,0.0493859781795,0.0494361307759,0.0494862833723,0.0498230222342,0.05000213865,0.
050080949873,0.0499806446801,0.0500594559031,0.0498373515474,0.0502099136924,0.0496
367411617,0.0498373515474,0.0498158575775,0.049787198951,0.0494791187157,0.04901341
60345,0.049199697107,0.0492068617636,0.0491137212274,0.0493143316131,0.049873174830
6,0.0497800342943,0.0500952791862,0.0504033594215,0.0511341543982,0.0514637286034,0
```

```
.0518004674652,0.051922266628,0.0523019934296,0.0515783631095,0.0520655597607,0.051
8147967785,0.0521658649535,0.0519867485377,0.0514923872299,0.0518721140315,0.051621
3510493,0.0517216562422,0.0520655597607,0.0518362907484,0.0513849173804,0.051607021
7361,0.0512487889044,0.0514279053202,0.0517789734953,0.0515067165432,0.051599857079
4,0.051549704483,0.0514207406636,0.0521228770137,0.0521730296102,0.0523879693092,0.
0528536719904,0.0533193746716,0.0532477281053,0.0530614470328,0.0531689168823,0.053
391021238,0.0532190694788,0.0533695272681,0.0536346195635,0.0538065713228,0.0535128
204008,0.0533193746716,0.0531044349726,0.0533265393283,0.0532763867318,0.0535558083
406,0.053469832461,0.0533480332982,0.0532119048221,0.053176081539,0.0537779126962,0
.053763583383,0.0539641937687,0.0542507800341,0.054193462781,0.0543295912571,0.0548
741051613,0.0553326431858,0.0559273096865,0.0559774622829,0.0555404182282,0.0555475
828849,0.0555189242583,0.0554401130354,0.0559201450298,0.0557696872405,0.0558556631
201,0.0561852373253,0.0560061209095,0.0563142011447,0.0567082572596,0.0568300564224
,0.0573817349832,0.0577256385017,0.0573530763567,0.0572456065072,0.0572814297903,0.
0570091728382,0.0568443857357,0.0570163374949,0.0570091728382,0.0573602410133,0.057
5680160557,0.0573960642965,0.0577757910981,0.057876096291,0.0578474376644,0.0578904
256042,0.0581196946165,0.0584349395084,0.0589078068463,0.0588576542499,0.0586212205
809,0.0588075016534,0.0589651240994,0.0595168026602,0.0596242725097,0.0595812845699
,0.0596027785398,0.0596959190761,0.0592087224249,0.0593878388408,0.0597389070159,0.
0602834209201,0.0603407381731,0.0607562882579,0.060892416734,0.0612219909391,0.0614
79918578,0.0614727539213,0.0620745850786,0.0622107135547,0.0622966894343,0.06201726
78255,0.0624471472236,0.0621892195848,0.0627265688323,0.0629988257844,0.06293434387
47,0.0628626973084,0.0631492835737,0.0632639180799,0.063378552586,0.0633713879294,0
.0637726087009,0.0641953234423,0.0642669700087,0.0645463916174,0.0653703271303,0.06
57142306488,0.0662085919565,0.0666098127281,0.0670970093792,0.0674552422109,0.06740
50896145,0.067512559464,0.0678994509222,0.0684869527662,0.0689168321643,0.069970036
6895,0.0704070807442,0.0714602852694,0.0723988552885,0.0736741641694,0.074619898845
1,0.0757089266535,0.0769985648476,0.0771203640104,0.078166403879,0.0781807331923,0.
0785246367107,0.0783956728913,0.0785604599939,0.0787324117531,0.0789401867955,0.079
2697610007,0.0791694558078,0.0798071102482,0.079635158489,0.0801940017065,0.0804949
172851,0.0807528449239,0.0810322665327,0.0815481218103,0.0818562020456,0.0823004107
569,0.082142788311,0.0825296797692,0.0828520893178,0.0832461454326,0.083238980776,0
.0840127636925,0.0842277033915,0.0840055990359,0.0845644422533,0.0843709965242,0.08
50731328744,0.0850301449346,0.0853238958566,0.0852450846336,0.0857322812847,0.08626
24658756,0.0865132288578,0.0869789315391,0.0872368591779,0.0876595739193,0.08825424
04199,0.0888704008905,0.0886626258481,0.0892644570054,0.0892214690656,0.08922863372
22,0.0895797018973,0.0897946415963,0.090231685651,0.0901170511448,0.0898447941927,0
.0900883925183,0.0906830590189,0.0904609546633,0.0911630910134,0.0910699504772,0.09
10914444471,0.0910556211639,0.0905469305429,0.0910627858205,0.0911845849833,0.09123
47375798,0.0915069945319,0.0919368739299,0.0922664481351,0.0925387050872,0.09339129
92267,0.0939859657273,0.0947669133004,0.0950964875056,0.0953544151444,0.09519679269
85,0.0955406962169,0.0961138687476,0.096185515314,0.0963001498201,0.0962714911936,0
.0960207282114,0.0960063988981,0.0961425273742,0.0962499972237,0.0971384146463,0.09
71670732729,0.096694205935,0.0965222541758,0.0966727119651,0.0966010653988,0.097102
5913632,0.0973963422852,0.097496647478,0.0983349123042,0.0986716511661,0.0991660124
738,0.0998466548541,0.100584614487,0.101136293048,0.10151601985,0.1022181562,0.1022
8263811,0.101867088025,0.102024710471,0.101573337103,0.101480196567,0.102039039784,
0.101938734591,0.102719682164,0.10230413208,0.101967393218,0.102368613989,0.1037513
9272,0.103894685852,0.104926396408,0.105506733595,0.105427922372,0.106359327735,0.1
0658143209,0.108021528074,0.108952933436,0.109504611997,0.109776868949,0.1105219932
39,0.110507663926,0.110923214011,0.111654008987,0.111876113343,0.1116683383,0.11170
4161584,0.112420627247,0.112685719543,0.112449285874,0.112936482525,0.113961028423,
0.114792128593,0.115429783033,0.115709204642,0.116540304812,0.117149300626,0.117822
778349,0.118818665621,0.119965010683,0.120602665123,0.121648704992,0.121756174841,0
.122372335312,0.123339563958,0.123862583892,0.12490862376,0.125274021249,0.12604780
4165,0.126542165473,0.126721281889,0.126556494786,0.127516558775,0.128390646885,0.1
2942235744,0.1307836422,0.131442790611,0.132546147732,0.133456059125,0.13459523953,
0.135913536351,0.136988234846,0.138019945401,0.138557294649,0.138879704197,0.139667
816427,0.140799832175,0.14127986417,0.14189602464,0.14255517305,0.143894963841,0.14
```

```
5263413258,0.145915397012,0.147076071387,0.147971653466,0.148616472563,0.1491896450
94,0.150106721143,0.151790415452,0.153345145942,0.15366755549,0.154126093515,0.1557
23811944,0.156483265547,0.15741467091,0.159105529875,0.160882364721,0.162028709782,
0.163024597054,0.164414540441,0.166220033913,0.167194427215,0.168326442964,0.169702
057037,0.171722490208,0.172947646493,0.174115485524,0.176028448845,0.177024336118,0
.178106199269,0.179137909825,0.180714134284,0.181946455225,0.182834872648,0.1838880
77173,0.18525652659,0.186316895772,0.187133666628,0.188867513534,0.190952428614,0.1
93037343695,0.194850001823,0.196712812548,0.199206113057,0.200402610715,0.202279750
753,0.20432884255,0.206034030829,0.206972600848,0.208247909729,0.209408584104,0.212
217129505,0.21333481594,0.215233449948,0.218808613608,0.220320356158,0.223014267052
,0.225170828699,0.227578153328,0.229612915812,0.232163533574,0.233488995052,0.23598
229556,0.238045716671,0.239428495401,0.241391611319,0.242194052862,0.24417866275,0.
246313730427,0.247803979007,0.24935154484,0.250770146853,0.252439511849,0.253908266
459,0.255147752057,0.257519253403,0.260084200478,0.261947011203,0.263652199482,0.26
6768825117,0.268416696143,0.269928438693,0.271139265664,0.271913048581,0.2731883574
62,0.273403297161,0.273267168685,0.274005128318,0.274306043897,0.273202686775,0.272
428903858,0.271877225298,0.271891554611,0.272027683087,0.271647956285,0.27051594053
7,0.270035908543,0.269398254102,0.268925386764,0.267807700329,0.267771877046,0.2675
06784751,0.266338945719,0.264920343706,0.262806769999,0.261166063629,0.258572457928
,0.255685101304,0.253177471482,0.250619689064,0.247961601453,0.24553994751,0.243354
727237,0.242394663248,0.242330181338,0.24202926576,0.242702743483,0.243935064424,0.
243662807472,0.24413567481,0.241957619193,0.239786728233,0.23613275335,0.2302362409
4,0.223823873252,0.216616228678,0.209508889297,0.201183558288,0.194484604335,0.1878
07144352,0.182727402798,0.177919918197,0.174717316681,0.172281333426,0.170819743472
,0.170576145147,0.169472788025,0.168734828392,0.168362266247,0.167237415155,0.16644
9302925,0.165990764901,0.164543504261,0.165066524195,0.163547616989,0.162788163385,
0.162236484825,0.16175645283,0.160165899057,0.159986782641,0.159377786828,0.1589192
48803,0.158553851315,0.156891650975,0.156762687156,0.156067715463,0.155480213619,0.
154641948792,0.154204904738,0.152528375085,0.151503829187,0.14994193404,0.149211139
064,0.148380038894,0.14700442482,0
.
145743445253,0.144145726823,0.143257309401,0.141888859983,0.141007607217,0.14045592
8657,0.139230772372,0.138442660142,0.137504090123,0.13687360034,0.135806066501,0.13
4494934337,0.132581971016,0.131292332822,0.131055899153,0.130095835164,0.1295871445
43,0.128068237336,0.126964880214,0.126212591268,0.125245362622,0.124772495284,0.124
321121916,0.123855419235,0.122701909517,0.121455259263,0.12065281772,0.120251596948
,0.119234215706,0.117901589572,0.117292593758,0.116712256571,0.116167742667,0.11528
6489901,0.113975357737,0.113187245507,0.112406297934,0.111102330426,0.110550651866,
0.109905832768,0.109175037792,0.107949881507,0.106996982175,0.106051247499,0.105642
862071,0.104589657546,0.10370840478,0.103371665918,0.103142396906,0.102168003604,0.
101286750838,0.100570285174,0.0999183014204,0.0991660124738,0.098399394214,0.098342
0769609,0.0974894828214,0.0966655473085,0.0957842945425,0.095304262548,0.0947310900
172,0.0942797166493,0.0936993794619,0.0930259017383,0.0924885524907,0.0917864161406
,0.091249066893,0.0915571471283,0.0911630910134,0.0908335168082,0.090740376272,0.09
01815330545,0.089644183807,0.0893719268549,0.0888274129507,0.088311557673,0.0878028
67052,0.0867281685569,0.0863197831287,0.0855316708989,0.0845572775967,0.08431367927
11,0.0840987395721,0.0834897437582,0.0829094065708,0.0828950772576,0.0826156556488,
0.0821212943411,0.0814191579909,0.0814621459307,0.0809892785929,0.0803946120922,0.0
804949172851,0.0803659534657,0.0804662586586,0.0793843955068,0.0788971988557,0.0787
252470965,0.0782667090719,0.0776863718845,0.0779084762402,0.0778439943305,0.0767979
544619,0.0767191432389,0.0764182276603,0.0766546613292,0.0766689906425,0.0762892638
408,0.0758665490994,0.0755369748943,0.0749566377069,0.0745410876221,0.0747058747247
,0.0741971841037,0.0736741641694,0.0729791924758,0.0725708070477,0.0727141001804,0.
0726066303309,0.0728287346865,0.0724848311681,0.0723272087221,0.0720191284869,0.071
6465663419,0.0718328474144,0.0715677551189,0.0715749197755,0.0710518998413,0.070901
4420519,0.070371257461,0.0703282695212,0.0699628720329,0.0693682055322,0.0694183581
287,0.0687663743749,0.0686302458989,0.0678134750426,0.0675842060303,0.0670755154093
,0.0671328326624,0.0667960938005,0.0669967041863,0.0671614912889,0.066982374873,0.0
66767435174,0.0669895395297,0.0671041740358,0.0670253628128,0.0671471619756,0.06665
```

```
28006679,0.0668390817403,0.0659864876009,0.0654992909498,0.0650479175818,0.06489029
51359,0.064496239021,0.0645678855873,0.0640018777132,0.0643314519184,0.064130841532
6,0.063908737177,0.0638084319841,0.0641093475627,0.0644675803944,0.0644890743643,0.
0645248976475,0.0639947130566,0.0639015725203,0.0632782473932,0.0632495887666,0.062
6262636394,0.0629988257844,0.062576111043,0.0620889143919,0.0617879988132,0.0613652
840718,0.0612578142223,0.0612004969692,0.0610643684932,0.0611431797162,0.0612076616
259,0.0608709227641,0.0606129951252,0.0605126899324,0.0605556778722,0.0603980554262
,0.0608207701676,0.0608422641375,0.0606703123783,0.0602117743537,0.0600756458777,0.
0602690916068,0.0604267140528,0.0601687864139,0.060011163968,0.0600254932812,0.0594
23662124,0.0590725939489,0.0586068912677,0.0585782326411,0.0584421041651,0.05782594
36945,0.0575178634593,0.0575966746822,0.0573459117,0.0572312771939,0.0568228917658,
0.0568443857357,0.0568945383321,0.0563786830544,0.0560491088493,0.0561207554156,0.0
558986510599,0.0557481932706,0.0553613018124,0.0552251733363,0.055289655246,0.05526
81612761,0.0551320328001,0.0553039845593,0.0552609966195,0.0552180086797,0.05520367
93664,0.0548454465347,0.0553039845593,0.0548669405046,0.0546161775224,0.05434392057
03,0.0545803542393,0.0543295912571,0.0540931575881,0.053820900636,0.0532978807017,0
.0532978807017,0.0525957443516,0.052803519394,0.0528321780205,0.0523879693092,0.051
7933028086,0.0514279053202,0.0516141863927,0.0514135760069,0.0517574795254,0.051549
704483,0.05147089326,0.0511484837115,0.0512416242477,0.0515210458565,0.051621351049
3,0.0517216562422,0.0513992466937,0.0513061061574,0.0508905560726,0.0504964999578,0
.0502887249154,0.050374700795,0.0497728696377,0.0492856729866,0.0492140264202,0.048
669512516,0.0484760667869,0.0482467977746,0.0479100589128,0.04778825975,0.047644966
6174,0.0472724044724,0.0474515208882,0.0472365811892,0.0469499949238,0.047387038978
5,0.0474730148581,0.0476879545572,0.0477452718102,0.0479100589128,0.0476592959306,0
.0475948140209,0.0475876493643,0.0477166131837,0.0474730148581,0.0472079225627,0.04
70001475203,0.0463123404834,0.0461690473507,0.0458251438323,0.0458752964287,0.04584
66378022,0.0454167584041,0.045223312675,0.044929561753,0.0448005979336,0.0448077625
902,0.044771939307,0.0444566944151,0.0446143168611,0.0445068470116,0.0445999875478,
0.0446214815177,0.044635810831,0.0448005979336,0.0448292565601,0.0446429754876,0.04
4635810831,0.0448292565601,0.0445999875478,0.0447934332769,0.0445426702948,0.044492
5176983,0.0444925176983,0.0443205659391,0.0440554736436,0.0436900761553,0.043740228
7517,0.0438047106614,0.0434966304262,0.0433461726368,0.0432315381307,0.043410654546
6,0.0428804699556,0.042794494076,0.0431097389679,0.0431885501909,0.0432888553838,0.
0429736104919,0.0431455622511,0.0432315381307,0.0430452570582,0.0430022691184,0.043
2028795042,0.0431813855343,0.0428231527025,0.0431169036245,0.0429019639255,0.043274
5260705,0.0435826063058,0.043088244998,0.0430165984317,0.0428661406424,0.0429234578
954,0.043167056221,0.0433103493537,0.0432530321006,0.043303184697,0.0430524217148,0
.0425508957504,0.0426225423168,0.0426941888831,0.0426726949132,0.0428016587326,0.04
24577552142,0.0422714741417,0.0421926629187,0.0421998275754,0.0420923577259,0.04246
49198708,0.0425079078106,0.0425150724673,0.0422786387983,0.042070863756,0.042300132
7683,0.0423287913948,0.0420207111595,0.0417484542074,0.041834430087,0.0416624783278
,0.0418201007738,0.0420135465029,0.0419490645932,0.0417986068039,0.0416051610747,0.
0415335145084,0.0412111049599,0.0415478438217,0.0416266550446,0.0417341248942,0.041
5263498518,0.0414905265686,0.0416624783278,0.0421711689488,0.0421783336055,0.041963
3939064,0.0426153776602,0.0425150724673,0.0424649198708,0.0425294017805,0.043181385
5343,0.0429879398051,0.0425723897204,0.0417484542074,0.0418845826835,0.042307297424
9,0.0418487594003,0.0417484542074,0.0415693377916,0.0412182696165,0.0406737557123,0
.040738237622,0.0413758920624,0.0418487594003,0.0417341248942,0.0414403739721,0.041
2970808395,0.04118961099,0.0412325989298,0.0411322937369,0.0409030247246,0.04095317
7321,0.0407454022786,0.0406236031159,0.0407597315919,0.040580615176,0.0408242135016
,0.0406092738026,0.0404516513566,0.0405161332663,0.0407454022786,0.0405877798327,0.
0404301573867,0.040580615176,0.0405519565495,0.0407024143388,0.0404874746398,0.0405
304625796,0.0405662858628,0.0403800047903,0.0402653702841,0.0400504305851,0.0401937
237178,0.0399429607356,0.0398713141693,0.0397136917233,0.0399501253922,0.0401077478
382,0.0398999727958,0.0399787840188,0.0398713141693,0.0397638443197,0.0395560692774
,0.0395775632473,0.0392551536987,0.039111860566,0.0389972260599,0.0385960052884,0.0
38818109644,0.0388109449874,0.0388109449874,0.0391046959094,0.0388539329272,0.03917
63424757,0.0391333545359,0.0391476838492,0.0390617079696,0.0393196356084,0.03941277
61447,0.0395345753075,0.0395417399641,0.0391620131625,0.0395847279039,0.03918350713
```

```
24,0.0393196356084,0.0394629287411,0.039269483012,0.0391046959094,0.038896920867,0.
0385673466618,0.0381446319204,0.0384312181858,0.0380658206974,0.0379870094745,0.037
6717645826,0.0375929533596,0.0376789292392,0.037721917179,0.0374783188534,0.0375356
361065,0.0376359412994,0.03750697748,0.0373851783172,0.0374138369437,0.037557130076
4,0.0376001180162,0.0376072826729,0.0378938689382,0.0380299974143,0.0379870094745,0
.0380514913842,0.0381159732939,0.0381876198602,0.0380729853541,0.0380013387877,0.03
78723749683,0.0378867042816,0.037349355034,0.0369123109793,0.037055604112,0.0371057
567085,0.037112921365 1,0.0370197808289,0.036704535937,0.0367117005936,0.03654691349
1,0.036683041967,0.036919475636,0.0373780136606,0.0373063670942,0.0371559093049,0.0
368621583829,0.0372633791544,0.0377290818356,0.0380299974143,0.0380228327576,0.0376
43105956,0.0374138369437,0.037349355034,0.0374926481667,0.0378150577152,0.037994174
1311,0.0374854835101,0.037213226558,0.0366472186839,0.0363534677619,0.036546913491,
0.0366615479971,0.0365540781476,0.0363534677619,0.0362316685991,0.0358519417975,0.0
362030099726,0.036145692719 5,0.0365182548645,0.0370699334253,0.0367546885334,0.0370
842627386,0.0371845679314,0.0371057567085,0.0367260299069,0.0371630739615,0.0367403
592201,0.0369338049492,0.0369481342625,0.0368693230395,0.0371702386182,0.0365755721
175,0.0365254195211,0.0369552989191,0.0368764876962,0.0367833471599,0.0368478290696
,0.0370341101421,0.0367260299069,0.0366328893706,0.0365612428043,0.0367690178467,0.
0366615479971,0.0363176444787,0.0366615479971,0.0363677970751,0.0365039255512,0.036
1815160027,0.0363534677619,0.0362245039425,0.036253162569,0.0361313634062,0.0325275
411192,0.0288162489827,0.0251049568462,0.0216300983786,0.0)
m <-
c(0.0220204840679,0.0258195910197,0.0296330342241,0.0335898399551,0.0373172656437,0
.0371810712435,0.0372455843804,0.0371810712435,0.0371810712435,0.0373172656437,0.03
75179731807,0.0376254950756,0.0376254950756,0.0375251413071,0.0374032831596,0.03708
0717475,0.0370878856013,0.0374104512859,0.0375753181913,0.0372169118751,0.037295761
2647,0.0371237262329,0.037037708717,0.0369015143169,0.0370878856013,0.0375609819387
,0.0372455843804,0.0371022218539,0.0369086824432,0.0369660274538,0.0369301868222,0.
0370305405907,0.0370520449697,0.0369015143169,0.0370090362117,0.0371595668645,0.037
4749644228,0.0376541675809,0.0378835476233,0.0377401850968,0.037611158823,0.0375896
54444,0.0374032831596,0.0374391237912,0.0372742568857,0.0370448768433,0.03684416930
63,0.036571780506,0.0368943461905,0.0375108050544,0.0373746106543,0.0369301868222,0
.0369875318328,0.0371523987382,0.0373746106543,0.0374677962965,0.0375108050544,0.03
7854875118,0.0377688576021,0.0377473532231,0.0382491220658,0.0385931921294,0.038901
4215613,0.0387508909085,0.0383709802133,0.0381631045499,0.0382132814342,0.038198945
1815,0.0380269101498,0.0378046982337,0.037546645686,0.0374677962965,0.0375179731807
,0.0373531062753,0.0374247875385,0.0373817787806,0.0374032831596,0.0371739031172,0.
0369946999591,0.0371452306119,0.0373387700226,0.0376756719599,0.0373961150332,0.037
6828400862,0.0383351395817,0.038607528382,0.0384498296029,0.0385788558767,0.0389372
621929,0.0387652271611,0.0387508909085,0.0386863777716,0.0387867315401,0.0390304478
351,0.0384068208449,0.0384713339819,0.0386720415189,0.0388440765507,0.0389659346982
,0.0392239872459,0.0392239872459,0.0391666422353,0.0393243410144,0.0393386772671,0.
0396038979411,0.0395465529305,0.0395537210568,0.0393960222777,0.0391953147406,0.039
3315091408,0.0391451378563,0.0392669960038,0.0392096509932,0.039123633477 4,0.038879
9171823,0.0386648733926,0.0387222184032,0.0393100047618,0.0394103585303,0.039130801
6037,0.0393960222777,0.0393243410144,0.0393315091408,0.0395752254358,0.039360181646
1,0.0395465529305,0.0396038979411,0.0390949609721,0.0390662884668,0.039159474109,0.
0392956685091,0.0394103585303,0.0395895616884,0.0394390310356,0.0399192954993,0.039
9192954993,0.0401056667838,0.0399694723836,0.0399981448889,0.0403207105735,0.040335
0468261,0.039890622994,0.0400626580258,0.0400626580258,0.0403565512051,0.0406146037
528,0.0405572587422,0.0407364619003,0.0412310626167,0.0409228331847,0.0410948682165
,0.041481947038,0.0411092044692,0.0411378769745,0.0406934531424,0.0409084969321,0.0
409658419427,0.0410590275849,0.0407651344056,0.0410303550796,0.0407938069109,0.0413
744251431,0.0415751326802,0.0417543358383,0.0421557509125,0.0419048664911,0.0418618
577332,0.0420840696492,0.0420769015229,0.0419837158807,0.0420840696492,0.0415392920
486,0.041747167712,0.0418260171016,0.0419837158807,0.0421342465335,0.0421915915441,
0.0418260171016,0.0418331852279,0.0417184952067,0.0418690258595,0.0420052202597,0.0
422704409337,0.0421055740282,0.0418331852279,0.0416109733118,0.0417830083436,0.0420
769015229,0.0421127421545,0.0420697333966,0.0420052202597,0.042256104681,0.04168265
```

```
45751,0.0416253095645,0.0416109733118,0.0416181414382,0.041524955796,0.041675486448
8,0.0420769015229,0.0424066353338,0.0425499978603,0.0424066353338,0.0425858384919,0
.0429227404292,0.0433886686402,0.0437040661985,0.0437685793354,0.0436252168089,0.04
32381379874,0.0430876073346,0.0431091117136,0.0432094654821,0.0433241555033,0.04309
4775461,0.0427937141553,0.0428940679239,0.0430230941977,0.0430230941977,0.043316987
377,0.0434531817772,0.0436108805563,0.0432811467454,0.0429155723028,0.043051766703,
0.0434746861561,0.0432381379874,0.0431736248505,0.0433743323876,0.0432524742401,0.0
433313236296,0.0432381379874,0.0435248630404,0.0439836231251,0.0440266318831,0.0435
750399247,0.0441556581569,0.0440194637568,0.0439334462409,0.0443922063256,0.0445857
457364,0.0444065425783,0.0445355688521,0.0445140644731,0.0444065425783,0.0448222939
051,0.0442488437991,0.0442201712938,0.0441484900306,0.0437112343248,0.0434961905351
,0.0432883148717,0.0433026511244,0.043316987377,0.0435750399247,0.0432883148717,0.0
433671642613,0.0435105267878,0.0437972518407,0.0444065425783,0.0442918525571,0.0447
936213998,0.0444495513362,0.0449728245579,0.0450373376948,0.0450803464527,0.0453168
946214,0.045639460306,0.0457469822009,0.0453527352531,0.0457326459482,0.04558211529
54,0.0458903447274,0.0455247702848,0.0454530890216,0.04590468098,0.0456036196744,0.
0454315846426,0.0456968053166,0.045639460306,0.0458258315905,0.0458401678431,0.0461
268928961,0.0463347685595,0.0467863605179,0.0467290155073,0.0469512274234,0.0470874
218235,0.0469798999287,0.047252288729,0.0471734393394,0.0474314918871,0.04761069504
52,0.04771821694,0.0470945899498,0.0472666249816,0.0474314918871,0.0477540575717,0.
0479691013614,0.0481769770248,0.0480837913826,0.0481554726458,0.0479332607298,0.048
1124638879,0.048513878962,0.0483346758039,0.048270162667,0.0484350295725,0.04814830
45195,0.0481984814038,0.0484780383304,0.0486429052359,0.0489081259099,0.04892963028
88,0.0493812222473,0.0497037879318,0.0501123711323,0.0500120173638,0.0502413974062,
0.0503417511747,0.0504994499538,0.0504851137012,0.0507073256172,0.0508936969017,0.0
5085785627,0.0511589175757,0.0512162625862,0.0510800681861,0.0515101557655,0.051854
2258291,0.0523273221665,0.0527430734933,0.0528362591355,0.0530728073042,0.053051302
9252,0.0527932503776,0.0531301523148,0.0538613011999,0.0538397968209,0.053646257410
1,0.0535243992626,0.0534813905047,0.053467054252,0.0533523642308,0.0537824518103,0.
0542913887793,0.0544204150532,0.0543343975373,0.0543630700426,0.0544849281901,0.054
7286444851,0.0552949264647,0.0556533327809,0.0563558091607,0.0566640385926,0.056735
7198559,0.0570941261721,0.0572733293302,0.057689080657,0.0580331507205,0.0585707601
948,0.0585420876896,0.0585420876896,0.0584704064263,0.0587356271003,0.0589291665111
,0.0588001402372,0.0588431489952,0.0585277514369,0.0587356271003,0.0585850964475,0.
0586424414581,0.0592660684483,0.0589076621321,0.058728458974,0.0588789896268,0.0592
947409536,0.0596961560277,0.059990049207,0.0603914642812,0.0605491630603,0.06064951
68289,0.0604631455444,0.0607928793553,0.0611871263032,0.0611656219242,0.06131615257
7,0.0611011087873,0.0612444713138,0.0612086306821,0.0613233207033,0.0617677445354,0
.0619326114409,0.0625060615468,0.0628071228524,0.0633088916951,0.0628788041157,0.06
30723435264,0.0632802191898,0.0631153522844,0.0632945554425,0.0627282734629,0.06269
24328313,0.0626780965786,0.0622336727465,0.0623196902624,0.0627641140945,0.06342358
17163,0.0640185362012,0.0639611911906,0.0643554381385,0.0644199512754,0.06474251696
,0.0651439320341,0.0655023383503,0.0654091527081,0.0658464084139,0.0656958777611,0.
0649145519917,0.0655668514872,0.0652586220553,0.0657890634033,0.0665058760357,0.066
2764959933,0.0666635748148,0.0669646361205,0.0669001229835,0.0670936623943,0.068082
863827,0.0678964925426,0.0682333944798,0.0689143664806,0.0690720652598,0.0693874628
18,0.0691007377651,0.0692154277863,0.0696598516184,0.0698390547765,0.0700899391978,
0.0706418849248,0.070978786862,0.0713156887992,0.0709286099777,0.0715665732206,0.07
15522369679,0.0715665732206,0.071731440126,0.0720540058106,0.0719966608,0.072448252
7585,0.0729285172222,0.0726919690535,0.0731865697698,0.0728711722116,0.07381019676,
0.0737456836231,0.074139930571,0.0744409918766,0.0749642650982,0.0749355925929,0.07
52796626565,0.0759176258994,0.0760538202995,0.0761828465734,0.0767921373109,0.07815
40813125,0.0786128413972,0.0789425752082,0.0791934596295,0.0795662021984,0.07957337
03247,0.0798744316303,0.0806055805154,0.0812077031266,0.0813295612741,0.08117903062
13,0.0813797381584,0.0817381444746,0.082168232054,0.0823832758438,0.0823546033385,0
.0824621252333,0.0827631865389,0.0835588485609,0.0840104405194,0.0852361901208,0.08
59171621216,0.0865336209855,0.0869780448176,0.087960078124,0.0893650308835,0.090110
5160212,0.0904115773268,0.0903542323163,0.091035204317,0.0907269748851,0.0908416649
063,0.091343433749,0.0919025476023,0.0917950257074,0.0917448488231,0.0924831658345,
0.093400686004,0.093909622973,0.0946049312265,0.0952715669746,0.0960672289966,0.096
7338647448,0.0973001467244,0.0980599681147,0.0987911169998,0.0990921783054,0.099493
5933796,0.100683502349,0.101027572413,0.101937924456,0.102454029551,0.102841108373,
0.103815973553,0.10464030808,0.105643845766,0.106403667156,0.107005789767,0.1075505
67368,0.108066672463,0.10875481259,0.109579147118,0.110338968508,0.110898082361,0.1
11629231246,0.111966133184,0.113127369648,0.114252765481,0.115170285651,0.116904972
221,0.117127184137,0.118209571212,0.119593019593,0.120682574794,0.12174345749,0.122
897525828,0.12355699345,0.123944072271,0.124954778083,0.124739734293,0.126331058337
,0.127019198464,0.127836364865,0.12899760133,0.129922289626,0.13067494289,0.1318576
83733,0.133613874683,0.134373696073,0.136201568286,0.136438116454,0.138072449256,0.
138653067489,0.138975633173,0.140050852122,0.140868018523,0.141857219955,0.14197190
9977,0.143161818946,0.143756773431,0.145613318149,0.145957388213,0.14740534973,0.14
8258356763,0.150100565228,0.151376491714,0.152415870031,0.153892504054,0.1546523254
44,0.155978428814,0.15647302953,0.157777628521,0.158236388606,0.159791872019,0.1594
97978839,0.160293640861,0.160945940357,0.162171689958,0.163053369496,0.163261245159
,0.164121420318,0.163992394045,0.165060444867,0.165103453625,0.166035310047,0.16678
0795185,0.167117697122,0.168092562302,0.169275303145,0.170178487062,0.171124679737,
0.172013527401,0.172472287486,0.173941753382,0.175303697384,0.177489975913,0.180149
350779,0.184063147752,0.189202694327,0.194277727764,0.200786386467,0.208965218602,0
.216993520085,0.224541557105,0.231100392691,0.236748876235,0.241651874641,0.2439958
51949,0.245078239024,0.246110449214,0.246827261847,0.245694697887,0.244999389634,0.
244583638307,0.244999389634,0.246239475488,0.246906111236,0.249543981724,0.25243990
4759,0.256117153563,0.258289095839,0.260439533736,0.263070236097,0.26427448132,0.26
5937486627,0.267492970039,0.269414027894,0.270367388695,0.271550129539,0.2720447302
55,0.273571541162,0.275420917754,0
.
276187907271,0.277478170009,0.27794409822,0.277635868788,0.278675247105,0.278295336
41,0.277822240072,0.277005073671,0.275628793417,0.274997998301,0.27306977232,0.2729
19241667,0.272245437792,0.270560928106,0.269206152231,0.267880048861,0.266654299259
,0.264740409531,0.263263775508,0.261321213274,0.259651039841,0.256769453058,0.25536
4500299,0.253923706908,0.251737428379,0.249802034271,0.246705403699,0.244676823949,
0.242691252958,0.239902851818,0.238024802721,0.236211266761,0.233508883136,0.231508
975892,0.228476858457,0.226699163128,0.225358723506,0.224111469525,0.221903686618,0
.219437851162,0.218068739034,0.216040159284,0.214104765177,0.213172908755,0.2121550
34817,0.209337961171,0.207581770222,0.20518761603,0.203710982007,0.20364646887,0.20
0908244614,0.199374265581,0.197782941537,0.195811706798,0.193030473784,0.1919767592
14,0.190134550749,0.189776144433,0.18843570481,0.186335443797,0.185468100512,0.1843
35536553,0.183461025141,0.181862532971,0.181031030317,0.179919970737,0.178851919915
,0.177131569597,0.174651397889,0.173762550224,0.172013527401,0.170171318936,0.16842
9464239,0.166078318805,0.165046108614,0.163426112065,0.161677089242,0.160250632103,
0.159347448186,0.157598425363,0.156702409573,0.155720376266,0.154616484812,0.154430
113528,0.153419407716,0.152430206284,0.151412332346,0.150423130913,0.149591628259,0
.148365878658,0.147261987204,0.146072078234,0.145111549307,0.143928808463,0.1428249
17009,0.142093768124,0.141075894186,0.140423594691,0.139735454564,0.138817934394,0.
137871741719,0.136982894055,0.13638793957,0.135133517463,0.134337855441,0.133298477
124,0.132416797587,0.131692816828,0.131441932406,0.1301875103,0.129298662636,0.1291
05123225,0.127434949791,0.126940349075,0.1261303508,0.125270175641,0.125090972483,0
.12396557665,0.123134073997,0.122962038965,0.12299071147,0.12147106869,0.1209406273
42,0.120230982835,0.119951425909,0.119628860224,0.118775853192,0.11790134178,0.1167
25769063,0.115643381988,0.114374623629,0.114281437986,0.11378683727,0.113750996638,
0.112711618321,0.112095159457,0.111772593773,0.111191975541,0.110776224214,0.110747
551709,0.11056834855,0.109980562192,0.109485961475,0.108661626948,0.10834622939,0.1
07787115537,0.106855259115,0.106905435999,0.106217295872,0.10561517326,0.1050990681
65,0.104403759912,0.104317742396,0.10395933608,0.104117034859,0.103622434142,0.1035
57921005,0.10290562151,0.102776595236,0.102848276499,0.102497038309,0.102346507657,
0.101966596961,0.101493500624,0.100812528623,0.100396777296,0.100289255402,0.099665
6284114,0.099436248369,0.098884302642,0.0987194357366,0.0985904094627,0.09832518878
87,0.0984470469362,0.0984470469362,0.0989774882842,0.0984255425573,0.0989631520316,
0.0987337719892,0.0987266038629,0.0989989926632,0.0987839488735,0.0988197895051,0.0
```

```
982821800308,0.0974148367456,0.0965044847024,0.0961317421335,0.0955582920276,0.0955
00947017,0.095278735101,0.0945905949738,0.094483073079,0.0945475862159,0.0947339575
003,0.0947913025109,0.0947411256266,0.0947984706372,0.0948988244058,0.0943683830578
,0.0944042236894,0.0940386492469,0.094239356784,0.0937877648255,0.0934508628883,0.0
933505091198,0.0932788278565,0.0927842271401,0.0922322814132,0.0923326351817,0.0915
083006544,0.0917878575811,0.091078213075,0.0910208680644,0.0904760904638,0.09059794
86113,0.0900961797686,0.0902897191793,0.0897162690734,0.0894295440204,0.08917865959
91,0.0890854739569,0.0889206070514,0.0885837051142,0.0883901657034,0.0881034406505,
0.0882396350506,0.0880532637662,0.0881607856611,0.0881392812821,0.0878382199765,0.0
875013180392,0.0868705229227,0.0862468959325,0.0861107015323,0.0855444195527,0.0850
784913417,0.0848347750466,0.0839029186245,0.08322911475,0.0833223003922,0.082583983
3808,0.0829352215707,0.0827631865389,0.0823330989595,0.0817739851062,0.081179030621
3,0.0807417749155,0.0809209780736,0.0804765542415,0.0801253160516,0.0800249622831,0
.0792077958822,0.0784623107444,0.078297443839,0.0784408063655,0.0785268238814,0.077
7311618594,0.0775519587013,0.0770071811006,0.0768996592058,0.0767347923003,0.076713
2879213,0.076483907879,0.0755878920884,0.0749212563403,0.0741901074552,0.0738101967
6,0.0728711722116,0.0726776328008,0.0714590513257,0.0703336554928,0.0691724190283,0
.068591800796,0.0684197657643,0.0680470231954,0.0674233962052,0.0673445468156,0.067
0793261416,0.0666349023095,0.0663911860145,0.066577557299,0.0662549916144,0.0658033
996559,0.0653661439502,0.0646278269388,0.0643984468964,0.0642837568752,0.0636672980
113,0.063638625506,0.0636314573797,0.0630078303895,0.0626494240733,0.0625060615468,
0.0622910177571,0.0619756201988,0.0621189827253,0.0615311963667,0.0622336727465,0.0
616960632722,0.0614093382192,0.0611727900505,0.0613734975876,0.0611011087873,0.0609
434100082,0.0609434100082,0.0602911105127,0.0599398723228,0.0597750054173,0.0596244
747645,0.0594882803643,0.059258900322,0.0590080159006,0.0588288127425,0.05872129084
77,0.0587212908477,0.0589219983847,0.0590008477743,0.0586424414581,0.0583127076472,
0.0580331507205,0.0580259825942,0.0579399650783,0.0579686375836,0.0579471332046,0.0
574955412462,0.0570081086562,0.0569364273929,0.0565421804451,0.0565995254557,0.0564
131541713,0.0557035096652,0.0553952802332,0.0550440420434,0.0548003257483,0.0547931
57622,0.0549365201485,0.054771653243,0.0547214763588,0.0546928038535,0.054599618211
3,0.0549365201485,0.055302094591,0.0558253678127,0.0559472259602,0.0560260753498,0.
0556318284019,0.0556174921493,0.0554454571175,0.055079882675,0.0555744833913,0.0553
952802332,0.0547788213694,0.0542913887793,0.0540763449896,0.054018999979,0.05415519
43792,0.0536820980418,0.0535889123995,0.0539401505894,0.0533523642308,0.05358174427
32,0.0534957267573,0.0534455498731,0.0533021873466,0.0532376742097,0.0532591785886,
0.0532305060833,0.0534025411151,0.0529007722724,0.0527932503776,0.0528864360198,0.0
531803291991,0.0530369666726,0.0528720997671,0.0527860822512,0.0525997109668,0.0528
290910092,0.052226968398,0.0525495340825,0.0526068790931,0.0520405971135,0.05221980
02716,0.051961747724,0.0517825445659,0.0514743151339,0.0513739613654,0.051180421954
6,0.0513811294917,0.0511302450704,0.0509295375333,0.0514814832602,0.0509367056596,0
.0510227231755,0.0510298913018,0.0511087406914,0.0513596251127,0.0515029876392,0.05
14958195129,0.0514384745023,0.0511015725651,0.0505782993434,0.0507073256172,0.05068
58212383,0.0509653781649,0.0509152012806,0.050614139975,0.0505066180802,0.050033521
7428,0.0500621942481,0.0505711312171,0.0504851137012,0.0502557336588,0.05037042368,
0.0499475042269,0.0497324604371,0.0491805147102,0.0488794534046,0.0493238772367,0.0
493668859946,0.0491231696996,0.0493955584999,0.0494313991315,0.0489511346678,0.0490
65824689,0.0493955584999,0.0497611329424,0.049596266037,0.0489511346678,0.048449365
8251,0.0480766232563,0.0477468894453,0.0473669787502,0.0475318456556,0.047510341276
6,0.0472164480974,0.0471017580762,0.0471519349604,0.0473884831291,0.0472737931079,0
.0471949437184,0.0471447668341,0.0471949437184,0.0471089262025,0.0469153867917,0.04
65283079702,0.0465641486019,0.0462559191699,0.046478131086,0.0464279542017,0.046442
2904543,0.0464494585807,0.0464924673386,0.0469727318023,0.0469942361813,0.047087421
8235,0.0473598106238,0.0477038806874,0.0475103412766,0.0474673325187,0.047381315002
8,0.0472451206027,0.0472809612343,0.0467720242653,0.0465211398439,0.0464279542017,0
.0460265391275,0.0454315846426,0.0448724707893,0.0449441520526,0.0450301695685,0.04
48509664104,0.044621586368,0.0441914987886,0.0439621187462,0.0440553043884,0.043768
5793354,0.0441413219043,0.0441628262833,0.043847428725,0.0440553043884,0.0441413219
043,0.0445212325995,0.0446072501154,0.0448581345367,0.0446932676312,0.0447362763892
,0.0447434445155,0.0449154795473,0.0451663639686,0.0449298157999,0.0444280469572,0.
```

```
0440338000094,0.043825924346,0.0434675180298,0.0432381379874,0.0429585810608,0.0427
435372711,0.0426431835025,0.0424281397128,0.0421557509125,0.0422704409337,0.0421629
190388,0.0424138034602,0.0426360153762,0.0428510591659,0.0429585810608,0.0427937141
553,0.0426933603868,0.0427292010184,0.0427220328921,0.0429872535661,0.0425356616077
,0.0423851309549,0.041990884007,0.0419765477544,0.042012388386,0.0419263708701,0.04
20553971439,0.042033892765,0.0419407071228,0.0416754864488,0.0419765477544,0.042098
4059019,0.042256104681,0.042564334113,0.0426288472499,0.0425715022393,0.04242813971
28,0.0421700871651,0.041747167712,0.0418188489752,0.0419048664911,0.0416181414382,0
.0415607964276,0.0411378769745,0.0409801781953,0.0410590275849,0.040994514448,0.041
1235407218,0.0414317701537,0.0412740713746,0.0410733638375,0.0411665494797,0.041223
8944903,0.0410805319639,0.0410303550796,0.0408941606794,0.0412310626167,0.041381593
2695,0.0417615039647,0.0419693796281,0.0420769015229,0.0420482290176,0.041811680848
9,0.0418403533542,0.0418475214805,0.0417543358383,0.0416754864488,0.0413887613958,0
.040973010069,0.04095150569,0.04095150569,0.04095150569,0.040994514448,0.0408583200
478,0.0406361081318,0.0407507981529,0.0405429224895,0.0405715949948,0.0406074356265
,0.0363567367162,0.0321490465639,0.0279915332959,0.024034727565,0.0)
ycorr <- c(-8.1216508887e-05,-8.15131308575e-05,-8.24040133671e-05,-8.38939801839e-
05,-8.59905292342e-05,-8.8700354574e-05,-9.20310633918e-05,-9.53768228354e-05,-
9.87428648573e-05,-0.000102134044883,-0.000105559348007,-0.000109021778322,-
0.000112524403615,-0.000116068435705,-0.000119653433928,-0.000123276381377,-
0.000126931735478,-0.000130614525992,-0.000134320413202,-0.000138048366144,-
0.000141794518837,-0.000145556703807,-0.000149333789014,-0.000153121279139,-
0.000156918214166,-0.000160721449371,-0.000164531077607,-0.00016834648795,-
0.0001721645223,-0.000175981681937,-0.000179793251992,-0.000183599302976,-
0.000187395543164,-0.00019117923022,-0.00019494944614,-0.000198702221682,-
0.000202435380993,-0.000206143496954,-0.000209823792955,-0.000213474880448,-
0.000217094158208,-0.000220678306148,-0.000224225262525,-0.000227734513956,-
0.000231205074446,-0.000234636544059,-0.000238026849783,-0.000241375901164,-
0.000244685360361,-0.000247953273305,-0.0002511817581,-0.000254371793883,-
0.000257525044617,-0.000260639815754,-0.000263713019356,-0.000266746581323,-
0.000269740345723,-0.000272696891157,-0.000275614701319,-0.000278493314378,-
0.000281334390271,-0.000284136227213,-0.000286896692127,-0.000289618113164,-
0.000292302264929,-0.000294948347414,-0.000297556095733,-0.000300124784058,-
0.000302654984509,-0.000305150315594,-0.000307609141046,-0.000310031823428,-
0.000312419258524,-0.000314772023817,-0.000317087046083,-0.00031936416514,-
0.000321605950915,-0.000323813278316,-0.000325987029399,-0.000328126256028,-
0.000330230770266,-0.000332304066689,-0.000334347291601,-0.000336360309921,-
0.000338346135535,-0.000340306827822,-0.00034224205923,-0.000344152643936,-
0.000346035823345,-0.000347892226583,-0.000349721368726,-0.000351521126722,-
0.000353290136075,-0.000355026108175,-0.00035673019265,-0.000358401489835,-
0.00036003930274,-0.000361643156391,-0.000363214908647,-0.000364755587916,-
0.00036626545196,-0.00036774544604,-0.000369197024131,-0.000370623883228,-
0.00037202723315,-0.000373407260308,-0.000374765221432,-0.000376101962176,-
0.000377417489273,-0.000378710845109,-0.000379983521572,-0.000381235819637,-
0.000382468263702,-0.000383680061952,-0.000384869590255,-0.000386037672031,-
0.000387183261303,-0.00038830671665,-0.000389409406722,-0.000390492344298,-
0.000391555479009,-0.000392598585394,-0.00039362273521,-0.000394627556815,-
0.000395613921012,-0.000396581207935,-0.000397528974204,-0.000398457715519,-
0.000399367633416,-0.000400257167125,-0.000401126483054,-0.000401976563249,-
0.00040280566279,-0.000403613605435,-0.000404400627915,-0.000405167817119,-
0.000405917528148,-0.000406650061801,-0.000407365571879,-0.000408064567593,-
0.000408748949313,-0.000409417302541,-0.000410069958648,-0.000410709061084,-
0.000411335482829,-0.000411949816669,-0.000412550983059,-0.000413136690866,-
0.0004137064867,-0.000414260275487,-0.000414796188088,-0.000415314707837,-
0.000415817480181,-0.000416304888945,-0.000416775704511,-0.000417228215575,-
0.000417662997089,-0.000418081364036,-0.000418484992432,-0.000418875709744,-
0.000419255136525,-0.000419624799243,-0.000419984391192,-0.000420333126165,-
0.000420670965606,-0.000420999692076,-0.000421318919149,-0.00042162948213,-
0.00042193000539,-0.000422219396949,-0.000422496075698,-0.000422759160469,-
```

```
0.000423008510939,-0.000423244633986,-0.000423469290161,-0.000423681282163,-
0.000423881545383,-0.000424069880012,-0.000424246070303,-0.000424412696232,-
0.000424571650559,-0.000424724379859,-0.000424871496095,-0.000425012441702,-
0.000425146605706,-0.000425274253422,-0.000425396881311,-0.000425513693114,-
0.000425626578642,-0.000425733751961,-0.000425833546078,-0.000425924793605,-
0.000426006971264,-0.000426080021847,-0.000426144635902,-0.000426201989738,-
0.00042624992612,-0.000426289720668,-0.000426321130543,-0.000426344146055,-
0.000426361293319,-0.000426372708985,-0.000426379694412,-0.000426383159369,-
0.000426382250048,-0.000426375402253,-0.000426361759807,-0.000426342331123,-
0.000426316207048,-0.000426284167816,-0.000426245498784,-0.000426199357774,-
0.000426146206168,-0.000426085078458,-0.000426015834036,-0.00042593966217,-
0.000425858288032,-0.000425771783612,-0.000425679205175,-0.000425579876197,-
0.000425472245323,-0.000425357491364,-0.000425234599448,-0.000425102748716,-
0.000424962020109,-0.000424812592774,-0.000424653630306,-0.00042448408466,-
0.000424305097114,-0.000424117100542,-0.000423921153352,-0.000423717572804,-
0.000423505516683,-0.000423285655384,-0.000423058641179,-0.000422823990909,-
0.000422582805259,-0.000422335111676,-0.000422081573002,-0.000421820786161,-
0.00042155365827,-0.000421279277108,-0.000420998209112,-0.000420709896119,-
0.000420414385788,-0.000420110916689,-0.000419798916402,-0.000419477009284,-
0.000419145248206,-0.000418802926355,-0.000418449456501,-0.000418083616513,-
0.000417704274638,-0.000417312155393,-0.000416907203377,-0.000416489462502,-
0.000416058978898,-0.00041561593984,-0.000415159522956,-0.000414691189645,-
0.000414211031385,-0.000413719961728,-0.000413219180517,-0.000412708755565,-
0.000412187313265,-0.000411654677364,-0.000411110488375,-0.000410555260128,-
0.000409988724031,-0.000409410749545,-0.000408819614675,-0.000408214671773,-
0.000407594486407,-0.000406958629608,-0.000406308475343,-0.000405644206347,-
0.000404965787827,-0.000404273156517,-0.00040356529646,-0.000402843194687,-
0.000402107715056,-0.000401359765743,-0.000400600231703,-0.000399829014378,-
0.00039904477655,-0.000398246231102,-0.000397432516894,-0.000396602443322,-
0.000395755708943,-0.000394890517747,-0.000394005061078,-0.000393097296142,-
0.000392166311282,-0.00039121180978,-0.000390234477891,-0.000389234916009,-
0.000388213584426,-0.000387171623253,-0.000386110635987,-0.00038503041651,-
0.000383933303841,-0.000382819868425,-0.000381690902059,-0.000380545990564,-
0.00037938398824,-0.000378205011464,-0.000377010300484,-0.000375798788054,-
0.000374570671259,-0.00037332672925,-0.000372066797193,-0.000370791149357,-
0.000369500795659,-0.000368196970263,-0.000366880388406,-0.000365553107629,-
0.00036421389272,-0.000362864151821,-0.000361503472881,-0.000360131749994,-
0.000358748132344,-0.000357353393759,-0.000355946288074,-0.000354526703274,-
0.000353094357195,-0.000351648018246,-0.00035018709912,-0.000348712780025,-
0.000347223496772,-0.000345720430476,-0.000344203318502,-0.000342672452666,-
0.00034112850523,-0.000339572793249,-0.000338005744946,-0.000336428130548,-
0.000334841311231,-0.000333244427956,-0.000331637325737,-0.000330019508159,-
0.000328391861231,-0.000326754663082,-0.000325107565481,-0.000323449460033,-
0.000321779259341,-0.000320096076589,-0.000318399476591,-0.000316688784783,-
0.000314965092384,-0.000313228909941,-0.000311479577645,-0.00030971640018,-
0.000307939401563,-0.000306148981613,-0.000304346244635,-0.000302532974499,-
0.00030070980585,-0.00029887694306,-0.000297035603862,-0.000295185526171,-
0.000293326705096,-0.000291460622046,-0.000289587790694,-0.000287708058655,-
0.000285821848537,-0.000283927710438,-0.000282025030135,-0.000280114008157,-
0.000278193541821,-0.000276264378936,-0.000274326861022,-0.000272380378285,-
0.000270424924805,-0.00026846042458,-0.00026648683215,-0.000264503924609,-
0.00026251143432,-0.000260509395699,-0.000258498657701,-0.000256478868688,-
0.000254449225939,-0.000252409763397,-0.000250360549457,-0.000248301133516,-
0.000246231649314,-0.000244152708083,-0.000242064523252,-0.000239968538257,-
0.000237864050874,-0.000235750238763,-0.000233629184328,-0.000231501512062,-
0.00022936760906,-0.000227228112928,-0.000225083135459,-0.00022293230202,-
0.000220776297888,-0.000218613659482,-0.000216444827302,-0.000214270609048,-
0.000212090061511,-0.000209902278882,-0.000207705209293,-0.000205498107352,-
0.000203280530567,-0.000201051790091,-0.000198812075808,-0.000196561201432,-
```

```
0.000194300113627,-0.000192028445581,-0.000189746632944,-0.000187454442578,-
0.000185152919711,-0.000182842864958,-0.000180525443991,-0.000178201702392,-
0.000175871906355,-0.000173535655951,-0.000171192019602,-0.000168840056248,-
0.000166479028462,-0.000164108574279,-0.000161728957105,-0.000159339674431,-
0.000156939790865,-0.000154528784201,-0.000152105292107,-0.000149668599843,-
0.000147218951511,-0.000144756819502,-0.000142282368162,-0.000139795247233,-
0.000137295076071,-0.000134781985749,-0.000132254717973,-0.000129713237191,-
0.000127157170593,-0.000124588097947,-0.000122006534291,-0.000119412445097,-
0.000116805860244,-0.000114187307234,-0.000111556064048,-0.000108911832928,-
0.000106255835159,-0.000103588274597,-0.00010090957614,-9.82181770678e-05,-
9.55122263664e-05,-9.27913456625e-05,-9.00546448789e-05,-8.73006139112e-05,-
8.45297946636e-05,-8.17430333329e-05,-7.89393663435e-05,-7.61178805709e-05,-
7.32784976412e-05,-7.04214090296e-05,-6.75471895495e-05,-6.46555804377e-05,-
6.1746996244e-05,-5.88219236719e-05,-5.58810527999e-05,-5.29236276427e-05,-
4.99491279353e-05,-4.69579066643e-05,-4.39491816114e-05,-4.09238518408e-05,-
3.78817259987e-05,-3.48234237469e-05,-3.17492507767e-05,-2.86589712595e-05,-
2.55526400009e-05,-2.24289694376e-05,-1.92873521185e-05,-1.61281126e-05,-
1.2951439268e-05,-9.75644416101e-06,-6.54172408737e-06,-3.30642942178e-06,-
4.94365479844e-08,3.22923615188e-06,6.53027075292e-06,9.85363217976e-
06,1.31980981873e-05,1.65630554065e-05,1.99488808697e-05,2.33559277144e-
05,2.67827339695e-05,3.02287648903e-05,3.36934313434e-05,3.7175785856e-
05,4.0676325344e-05,4.4195267115e-05,4.77329866714e-05,5.12914489489e-
05,5.48709981866e-05,5.84717540096e-05,6.20946176467e-05,6.57400145454e-
05,6.94083881382e-05,7.31000969696e-05,7.68159109644e-05,8.05554023694e-
05,8.43180241647e-05,8.81034250202e-05,9.19104619449e-05,9.57395176157e-
05,9.95901098825e-
05,0.000103461859945,0.00010735599296,0.000111272634665,0.000115211924715,0.0001191
73662778,0.000123159015499,0.000127168397215,0.000131202942855,0.000135262001519,0.
000139345068833,0.000143452915756,0.000147584679154,0.00015174003689,0.000155918959
982,0.000160121309998,0.000164346842188,0.000168594736611,0.000172864178284,0.00017
7155559064,0.000181469660121,0.000185806765916,0.000190165792724,0.000
194547894742,0.00019895188028,0.000203378090787,0.000207825838887,0.000212294621702
,0.000216784836721,0.000221296345775,0.00022582868264,0.00023038154903,0.0002349555
35658,0.000239550491471,0.000244165692895,0.000248801618363,0.000253457357159,0.000
258133536722,0.000262830491575,0.000267547141042,0.000272283527145,0.00027703942742
5,0.000281813862479,0.000286605410807,0.000291415022117,0.000296241815849,0.0003010
86067027,0.00030594829648,0.000310827846152,0.000315724762344,0.000320639069659,0.0
00325570264853,0.000330517851093,0.000335482160318,0.000340462586802,0.000345457393
392,0.000350466088915,0.000355486740962,0.00036051779592,0.000365557378064,0.000370
603582726,0.000375654078561,0.000380707233207,0.000385761299794,0.000390815135468,0
.000395868344893,0.00040092031494,0.000405969973695,0.000411016678824,0.00041605945
0828,0.000421098831981,0.000426134966405,0.000431168360232,0.000436198907483,0.0004
41225576564,0.000446247751145,0.000451264909939,0.000456277439904,0.000461285479878
,0.000466290268311,0.000471292961951,0.000476295199613,0.000481306455196,0.00048633
919796,0.000491406365154,0.000496520750243,0.000501725188809,0.000506820300042,0.00
0511843799899,0.000516805993249,0.000521716495751,0.000526584778827,0.0005314152396
17,0.000536198331685,0.00054092058444,0.000545568260687,0.000550126575567,0.0005545
51272334,0.000559039130015,0.000563550607789,0.000568073046118,0.000572594780043,0.
000577105128296,0.000581597746218,0.000586073539157,0.000590534580146,0.00059498358
0029,0.000599424164073,0.000603858183416,0.000608288351658,0.000612715888147,0.0006
17142094475,0.00062156668117,0.000625988854174,0.000630405803739,0.000634814028335,
0.000639210094011,0.000643589141382,0.000647948381455,0.000652283804329,0.000656592
390851,0.000660871048484,0.000665117434947,0.000669328424476,0.000673502016767,0.00
0677635838806,0.000681726423906,0.000685771008447,0.000689766557277,0.0006937091815
87,0.000697596208494,0.000701423634624,0.000705187269252,0.000708882598639,0.000712
504984938,0.000716050097306,0.000719516256947,0.000722900815706,0.000726200605444,0
.000729413154134,0.000732535410123,0.000735564924065,0.000738499909916,0.0007413390
52654,0.000744082181481,0.000746728970984,0.000749277147129,0.000751724011976,0.000
754068275064,0.000756309393139,0.000758446263764,0.000760477664656,0.00076240139126
```

```
2,0.000764215710654,0.00076591886489,0.0007675086016,0.000768983492738,0.0007703432
86916,0.000771587136481,0.00077271459031,0.000773724335036,0.00077461618763,0.00077
53907765,0.000776050192775,0.000776594805937,0.000777025627642,0.000777343751399,0.
00077755055022,0.00077764689124,0.000777633375396,0.000777511053072,0.0007772814952
76,0.000776946008656,0.00077650454978,0.000775957442797,0.000775306559356,0.0007745
52563537,0.00077369688293,0.000772741227948,0.000771687700476,0.000770538891872,0.0
00769296037375,0.00076796126318,0.000766536496804,0.000765024306792,0.0007634263543
7,0.000761744568883,0.000759980519943,0.000758135387987,0.000756209387803,0.0007542
03324817,0.000752118652892,0.000749957123999,0.000747719553038,0.000745407388729,0.
000743021969612,0.00074056509957,0.000738038769088,0.00073544534694,0.0007327875878
36,0.00073006868741,0.000727290421043,0.000724454409417,0.000721562727077,0.0007186
18314009,0.000715623063673,0.000712578902616,0.000709487504616,0.000706350126626,0.
000703167962813,0.00069994209975,0.000696673641937,0.000693363915145,0.000690014928
687,0.000686627718084,0.000683202750551,0.000679742269378,0.000676247755756,0.00067
2721192352,0.000669164254809,0.000665578863174,0.000661966413043,0.000658328556473,
0.000654666816458,0.000650981117192,0.000647273156982,0.000643543775689,0.000639793
465947,0.000636022193181,0.000632230577762,0.000628419007335,0.00062458892073,0.000
620741406381,0.000616877058634,0.000612997549229,0.000609105250369,0.00060520123120
4,0.000601286790946,0.000597363491152,0.000593433461682,0.000589497260399,0.0005855
55008765,0.000581607204376,0.00057765445198,0.0005736970085,0.000569734445208,0.000
565766305878,0.000561792465344,0.000557813856353,0.000553829974223,0.00054984162848
9,0.000545850164041,0.000541856519787,0.000537861444228,0.000533865987843,0.0005298
72140507,0.000525881640279,0.000521895428389,0.00051791351323,0.000513935916306,0.0
00509964213816,0.00050599936338,0.000502041665253,0.000498091673959,0.0004941495919
35,0.000490214058254,0.000486283621321,0.000482358868449,0.000478440532783,0.000474
530928537,0.000470629025136,0.000466733755832,0.00046284449545,0.000458961925409,0.
000455086035546,0.000451217713963,0.000447358687659,0.000443509845862,0.00043967099
4054,0.000435841611174,0.000432021807532,0.000428213115037,0.000424416354495,0.0004
20631183956,0.000416857607894,0.000413096773593,0.000409348757806,0.000405613575698
,0.000401891739428,0.0003981827751,0.000394487150043,0.000390805354638,0.0003871376
77827,0.000383484711562,0.00037984801505,0.00037622698005,0.000372620938852,0.00036
9029932884,0.000365454313345,0.00036189465936,0.000358351238711,0.000354823070767,0
.000351309466443,0.000347810311358,0.000344324846317,0.000340852831336,0.0003373950
75074,0.000333951864092,0.000330522472268,0.000327105813899,0.000323701279633,0.000
320310157741,0.000316933830885,0.000313571939081,0.000310224865045,0.00030689230656
5,0.000303573212655,0.000300266179029,0.000296971291141,0.000293688603125,0.0002904
19039517,0.000287161429469,0.000283914054596,0.000280676427307,0.000277447593508,0.
000274227896977,0.0002710179685,0.000267817727937,0.000264627801824,0.0002614482961
25,0.000258279090746,0.000255119468394,0.000251970263561,0.000248831343935,0.000245
702288472,0.000242582619093,0.000239470558902,0.000236365987782,0.000233268590551,0
.000230177837635,0.00022709290154,0.000224013410042,0.000220938620497,0.00021786792
0611,0.000214801577035,0.000211740076535,0.000208684070891,0.000205633947485,0.0002
02588097788,0.000199547224732,0.00019651056106,0.000193478064849,0.000190449096544,
0.00018742381839,0.000184402619549,0.000181384191621,0.000178367651954,0.0001753525
32306,0.00017233858056,0.000169325791511,0.000166313283887,0.000163301246765,0.0001
60289285875,0.000157277093603,0.000154264271642,0.000151250285527,0.000148235808221
,0.000145220507511,0.000142205320321,0.00013918982558,0.000136174505243,0.000133159
615759,0.000130145124054,0.000127132052962,0.000124120846938,0.00012111203243,0.000
11810592129,0.000115102162305,0.000112100819346,0.000109100875086,0.000106101806707
,0.000103102688851,0.000100102535006,9.71003671797e-05,9.40940685418e-
05,9.10815304298e-05,8.8061113353e-05,8.50316440039e-05,8.19912759664e-
05,7.89396712003e-05,7.58764766419e-05,7.280136707e-05,6.97149295144e-
05,6.66177371055e-05,6.35102981159e-05,6.03942873567e-05,5.72717879723e-
05,5.41445102748e-05,5.10141321806e-05,4.78820168752e-05,4.47489868852e-
05,4.16160868079e-05,3.8484853976e-05,3.53551125593e-05,3.22273239761e-
05,2.91016624598e-05,2.5977212461e-05,2.28531712134e-05,1.97289987157e-
05,1.66046366318e-05,1.34806240354e-05,1.03573097331e-05,7.23386274742e-
06,4.11006727192e-06,9.86553930425e-07,-2.13618884061e-06,-5.25628189377e-06,-
8.37206013265e-06,-1.14824632601e-05,-1.45876399986e-05,-1.76870063742e-05,-
```

```
2.07806909352e-05,-2.38681198238e-05,-2.69474557225e-05,-3.0018295695e-05,-
3.30800281115e-05,-3.61335315097e-05,-3.9179429674e-05,-4.22182084849e-05,-
4.52487696627e-05,-4.82704229861e-05,-5.12823268693e-05,-5.42830921415e-05,-
5.7272912296e-05,-6.02520360613e-05,-6.32204164118e-05,-6.61775276466e-05,-
6.91216589564e-05,-7.20524568779e-05,-7.49689472947e-05,-7.78707802699e-05,-
8.07574732297e-05,-8.36294786518e-05,-8.64872452939e-05,-8.93302524195e-05,-
9.21574233732e-05,-9.49678012377e-05,-9.77618233408e-05,-0.000100539400241,-
0.000103300596098,-0.000106045278179,-0.00010877376917,-0.000111485634078,-
0.000114180187584,-0.000116856877475,-0.000119515805622,-0.000122157348256,-
0.000124781357384,-0.00012738574686,-0.000129971293059,-0.000132537907316,-
0.000135085747811,-0.000137614751028,-0.000140125025715,-0.000142617394608,-
0.000145091002403,-0.000147545762587,-0.000149982155134,-0.000152401803971,-
0.00015480592965,-0.000157193977669,-0.00015956615253,-0.000161922695446,-
0.000164263746087,-0.000166589577192,-0.000168900824794,-0.000171198582486,-
0.000173483095721,-0.000175754576803,-0.000178011186183,-0.000180252989244,-
0.000182479118057,-0.000184689164689,-0.000186884102073,-0.000189062804253,-
0.000191225445587,-0.000193370338803,-0.000195497804435,-0.000197607124382,-
0.000199698210046,-0.000201772446151,-0.00020383077,-0.000205874499606,-
0.000207903415691,-0.00020991624229,-0.000211914315139,-0.000213897947143,-
0.000215868104029,-0.00021782481267,-0.000219768558884,-0.000221699403534,-
0.000223617005355,-0.000225520530954,-0.000227408887769,-0.000229284315758,-
0.000231146158707,-0.000232995258055,-0.000234830435777,-0.00023665149851,-
0.000238458769443,-0.00024025217276,-0.000242031069699,-0.00024379578387,-
0.000245547284018,-0.000247285204838,-0.000249008829149,-0.00025071887038,-
0.000252414049073,-0.000254094029933,-0.000255759599454,-0.000257409778738,-
0.000259045984968,-0.000260668162271,-0.000262275748994,-0.000263868674246,-
0.000265447530079,-0.000267011597979,-0.00026856104085,-0.000270097590238,-
0.000271621694995,-0.000273134353385,-0.000274636029823,-0.000276125971337,-
0.000277605044955,-0.000279073476076,-0.000280531096061,-0.000281978369501,-
0.000283416205348,-0.000284844475899,-0.000286261955159,-0.000287667951777,-
0.000289060970295,-0.000290441365844,-0.000291808888455,-0.00029316278728,-
0.000294503569875,-0.000295830936685,-0.00029714489922,-0.000298445332162,-
0.000299734108729,-0.000301011215231,-0.000302277696156,-0.000303532742577,-
0.000304776243582,-0.000306009489414,-0.000307232964554,-0.000308446353371,-
0.000309650064556,-0.00031084402331,-0.00031202578511,-0.000313194953624,-
0.000314352332817,-0.000315499098329,-0.000316636247364,-0.000317762951858,-
0.000318878888159,-0.000319984336506,-0.000321079076117,-0.000322162914097,-
0.000323236452859,-0.000324300934368,-0.000325354293012,-0.000326395787652,-
0.00032742470214,-0
.000328440799757,-0.000329443409248,-0.000330430875,-0.000331401907171,-
0.000332356505218,-0.000333295001093,-0.000334216753345,-0.000335123100497,-
0.000336015623591,-0.000336893885709,-0.000337757859658,-0.000338608935334,-
0.000339448655072,-0.000340278952885,-0.000341100650431,-0.000341913428267,-
0.000342716641353,-0.000343510313349,-0.000344292844428,-0.000345063942346,-
0.000345823943206,-0.000346572596844,-0.000347308430896,-0.000348030831031,-
0.000348738253055,-0.000349431184705,-0.000350109812135,-0.000350773590726,-
0.000351422800956,-0.000352057914035,-0.00035267799813,-0.000353281871712,-
0.000353869368535,-0.000354441435499,-0.000354997909025,-0.000355538616248,-
0.00035606404525,-0.000356573848746,-0.000357068220492,-0.00035754619172,-
0.000358007637152,-0.00035845322808,-0.000358884813365,-0.000359300745766,-
0.000359699644065,-0.000360081342119,-0.000360444020413,-0.000360787017851,-
0.000361110305514,-0.00036141349764,-0.000361697383087,-0.000361960629317,-
0.000362202035185,-0.000362421182933,-0.000362620033505,-0.000362798122256,-
0.00036295591569,-0.00036309473516,-0.000363214610732,-0.000363316071968,-
0.00036339877262,-0.000363462491588,-0.000363508055774,-0.000363534831871,-
0.000363541562594,-0.000363527207197,-0.000363492641558,-0.000363437691779,-
0.000363360432424,-0.000363259852417,-0.000363135343286,-0.000362987630301,-
0.000362816520788,-0.000362622097729,-0.00036240604404,-0.000362169160647,-
0.00036191225068,-0.000361633576159,-0.000361333733379,-0.000361013672289,-
```

```
0.00036067358738,-0.000360311458306,-0.000359926636466,-0.000359519282498,-
0.000359087426894,-0.000358630448342,-0.000358145602316,-0.000357632883084,-
0.000357092833541,-0.000356524670199,-0.000355928517666,-0.00035530693302,-
0.000354660072933,-0.000353989020851,-0.000353293336448,-0.000352574842809,-
0.000351835781838,-0.000351076905312,-0.000350299089145,-0.000349503520486,-
0.000348691150757,-0.000347861548137,-0.000347014776901,-0.000346152073457,-
0.000345273298414,-0.000344378403958,-0.000343466561421,-0.000342536529395,-
0.000341590405003,-0.000340627273863,-0.000339648859109,-0.000338655070879,-
0.000337647271812,-0.000336722794338,-0.000335981907802,-0.000335423803254,-
0.000335049972578,-0.000334863106957)
xcorr <- c(-600.0,-598.999165972,-597.998331943,-596.997497915,-595.996663887,-
594.995829858,-593.99499583,-592.994161802,-591.993327773,-590.992493745,-
589.991659716,-588.990825688,-587.98999166,-586.989157631,-585.988323603,-
584.987489575,-583.986655546,-582.985821518,-581.98498749,-580.984153461,-
579.983319433,-578.982485405,-577.981651376,-576.980817348,-575.979983319,-
574.979149291,-573.978315263,-572.977481234,-571.976647206,-570.975813178,-
569.974979149,-568.974145121,-567.973311093,-566.972477064,-565.971643036,-
564.970809008,-563.969974979,-562.969140951,-561.968306922,-560.967472894,-
559.966638866,-558.965804837,-557.964970809,-556.964136781,-555.963302752,-
554.962468724,-553.961634696,-552.960800667,-551.959966639,-550.959132611,-
549.958298582,-548.957464554,-547.956630525,-546.955796497,-545.954962469,-
544.95412844,-543.953294412,-542.952460384,-541.951626355,-540.950792327,-
539.949958299,-538.94912427,-537.948290242,-536.947456214,-535.946622185,-
534.945788157,-533.944954128,-532.9441201,-531.943286072,-530.942452043,-
529.941618015,-528.940783987,-527.939949958,-526.93911593,-525.938281902,-
524.937447873,-523.936613845,-522.935779817,-521.934945788,-520.93411176,-
519.933277731,-518.932443703,-517.931609675,-516.930775646,-515.929941618,-
514.92910759,-513.928273561,-512.927439533,-511.926605505,-510.925771476,-
509.924937448,-508.92410342,-507.923269391,-506.922435363,-505.921601334,-
504.920767306,-503.919933278,-502.919099249,-501.918265221,-500.917431193,-
499.916597164,-498.915763136,-497.914929108,-496.914095079,-495.913261051,-
494.912427023,-493.911592994,-492.910758966,-491.909924937,-490.909090909,-
489.908256881,-488.907422852,-487.906588824,-486.905754796,-485.904920767,-
484.904086739,-483.903252711,-482.902418682,-481.901584654,-480.900750626,-
479.899916597,-478.899082569,-477.89824854,-476.897414512,-475.896580484,-
474.895746455,-473.894912427,-472.894078399,-471.89324437,-470.892410342,-
469.891576314,-468.890742285,-467.889908257,-466.889074229,-465.8882402,-
464.887406172,-463.886572143,-462.885738115,-461.884904087,-460.884070058,-
459.88323603,-458.882402002,-457.881567973,-456.880733945,-455.879899917,-
454.879065888,-453.87823186,-452.877397832,-451.876563803,-450.875729775,-
449.874895746,-448.874061718,-447.87322769,-446.872393661,-445.871559633,-
444.870725605,-443.869891576,-442.869057548,-441.86822352,-440.867389491,-
439.866555463,-438.865721435,-437.864887406,-436.864053378,-435.863219349,-
434.862385321,-433.861551293,-432.860717264,-431.859883236,-430.859049208,-
429.858215179,-428.857381151,-427.856547123,-426.855713094,-425.854879066,-
424.854045038,-423.853211009,-422.852376981,-421.851542952,-420.850708924,-
419.849874896,-418.849040867,-417.848206839,-416.847372811,-415.846538782,-
414.845704754,-413.844870726,-412.844036697,-411.843202669,-410.842368641,-
409.841534612,-408.840700584,-407.839866555,-406.839032527,-405.838198499,-
404.83736447,-403.836530442,-402.835696414,-401.834862385,-400.834028357,-
399.833194329,-398.8323603,-397.831526272,-396.830692244,-395.829858215,-
394.829024187,-393.828190158,-392.82735613,-391.826522102,-390.825688073,-
389.824854045,-388.824020017,-387.823185988,-386.82235196,-385.821517932,-
384.820683903,-383.819849875,-382.819015847,-381.818181818,-380.81734779,-
379.816513761,-378.815679733,-377.814845705,-376.814011676,-375.813177648,-
374.81234362,-373.811509591,-372.810675563,-371.809841535,-370.809007506,-
369.808173478,-368.80733945,-367.806505421,-366.805671393,-365.804837364,-
364.804003336,-363.803169308,-362.802335279,-361.801501251,-360.800667223,-
359.799833194,-358.798999166,-357.798165138,-356.797331109,-355.796497081,-
```

```
354.795663053,-353.794829024,-352.793994996,-351.793160967,-350.792326939,-
349.791492911,-348.790658882,-347.789824854,-346.788990826,-345.788156797,-
344.787322769,-343.786488741,-342.785654712,-341.784820684,-340.783986656,-
339.783152627,-338.782318599,-337.78148457,-336.780650542,-335.779816514,-
334.778982485,-333.778148457,-332.777314429,-331.7764804,-330.775646372,-
329.774812344,-328.773978315,-327.773144287,-326.772310259,-325.77147623,-
324.770642202,-323.769808173,-322.768974145,-321.768140117,-320.767306088,-
319.76647206,-318.765638032,-317.764804003,-316.763969975,-315.763135947,-
314.762301918,-313.76146789,-312.760633862,-311.759799833,-310.758965805,-
309.758131776,-308.757297748,-307.75646372,-306.755629691,-305.754795663,-
304.753961635,-303.753127606,-302.752293578,-301.75145955,-300.750625521,-
299.749791493,-298.748957465,-297.748123436,-296.747289408,-295.746455379,-
294.745621351,-293.744787323,-292.743953294,-291.743119266,-290.742285238,-
289.741451209,-288.740617181,-287.739783153,-286.738949124,-285.738115096,-
284.737281068,-283.736447039,-282.735613011,-281.734778982,-280.733944954,-
279.733110926,-278.732276897,-277.731442869,-276.730608841,-275.729774812,-
274.728940784,-273.728106756,-272.727272727,-271.726438699,-270.725604671,-
269.724770642,-268.723936614,-267.723102585,-266.722268557,-265.721434529,-
264.7206005,-263.719766472,-262.718932444,-261.718098415,-260.717264387,-
259.716430359,-258.71559633,-257.714762302,-256.713928274,-255.713094245,-
254.712260217,-253.711426188,-252.71059216,-251.709758132,-250.708924103,-
249.708090075,-248.707256047,-247.706422018,-246.70558799,-245.704753962,-
244.703919933,-243.703085905,-242.702251877,-241.701417848,-240.70058382,-
239.699749791,-238.698915763,-237.698081735,-236.697247706,-235.696413678,-
234.69557965,-233.694745621,-232.693911593,-231.693077565,-230.692243536,-
229.691409508,-228.69057548,-227.689741451,-226.688907423,-225.688073394,-
224.687239366,-223.686405338,-222.685571309,-221.684737281,-220.683903253,-
219.683069224,-218.682235196,-217.681401168,-216.680567139,-215.679733111,-
214.678899083,-213.678065054,-212.677231026,-211.676396997,-210.675562969,-
209.674728941,-208.673894912,-207.673060884,-206.672226856,-205.671392827,-
204.670558799,-203.669724771,-202.668890742,-201.668056714,-200.667222686,-
199.666388657,-198.665554629,-197.664720601,-196.663886572,-195.663052544,-
194.662218515,-193.661384487,-192.660550459,-191.65971643,-190.658882402,-
189.658048374,-188.657214345,-187.656380317,-186.655546289,-185.65471226,-
184.653878232,-183.653044204,-182.652210175,-181.651376147,-180.650542118,-
179.64970809,-178.648874062,-177.648040033,-176.647206005,-175.646371977,-
174.645537948,-173.64470392,-172.643869892,-171.643035863,-170.642201835,-
169.641367807,-168.640533778,-167.63969975,-166.638865721,-165.638031693,-
164.637197665,-163.636363636,-162.635529608,-161.63469558,-160.633861551,-
159.633027523,-158.632193495,-157.631359466,-156.630525438,-155.62969141,-
154.628857381,-153.628023353,-152.627189324,-151.626355296,-150.625521268,-
149.624687239,-148.623853211,-147.623019183,-146.622185154,-145.621351126,-
144.620517098,-143.619683069,-142.618849041,-141.618015013,-140.617180984,-
139.616346956,-138.615512927,-137.614678899,-136.613844871,-135.613010842,-
134.612176814,-133.611342786,-132.610508757,-131.609674729,-130.608840701,-
129.608006672,-128.607172644,-127.606338616,-126.605504587,-125.604670559,-
124.60383653,-123.603002502,-122.602168474,-121.601334445,-120.600500417,-
119.599666389,-118.59883236,-117.597998332,-116.597164304,-115.596330275,-
114.595496247,-113.594662219,-112.59382819,-111.592994162,-110.592160133,-
109.591326105,-108.590492077,-107.589658048,-106.58882402,-105.587989992,-
104.587155963,-103.586321935,-102.585487907,-101.584653878,-100.58381985,-
99.5829858215,-98.5821517932,-97.5813177648,-96.5804837364,-95.5796497081,-
94.5788156797,-93.5779816514,-92.577147623,-91.5763135947,-90.5754795663,-
89.5746455379,-88.5738115096,-87.5729774812,-86.5721434529,-85.5713094245,-
84.5704753962,-83.5696413678,-82.5688073394,-81.5679733111,-80.5671392827,-
79.5663052544,-78.565471226,-77.5646371977,-76.5638031693,-75.562969141,-
74.5621351126,-73.5613010842,-72.5604670559,-71.5596330275,-70.5587989992,-
69.5579649708,-68.5571309425,-67.5562969141,-66.5554628857,-65.5546288574,-
64.553794829,-63.5529608007,-62.5521267723,-61.551292744,-60.5504587156,-
```

```
59.5496246872,-58.5487906589,-57.5479566305,-56.5471226022,-55.5462885738,-
54.5454545455,-53.5446205171,-52.5437864887,-51.5429524604,-50.542118432,-
49.5412844037,-48.5404503753,-47.539616347,-46.5387823186,-45.5379482902,-
44.5371142619,-43.5362802335,-42.5354462052,-41.5346121768,-40.5337781485,-
39.5329441201,-38.5321100917,-37.5312760634,-36.530442035,-35.5296080067,-
34.5287739783,-33.52793995,-32.5271059216,-31.5262718932,-30.5254378649,-
29.5246038365,-28.5237698082,-27.5229357798,-26.5221017515,-25.5212677231,-
24.5204336947,-23.5195996664,-22.518765638,-21.5179316097,-20.5170975813,-
19.516263553,-18.5154295246,-17.5145954962,-16.5137614679,-15.5129274395,-
14.5120934112,-13.5112593828,-12.5104253545,-11.5095913261,-10.5087572977,-
9.50792326939,-8.50708924103,-7.50625521268,-6.50542118432,-5.50458715596,-
4.50375312761,-3.50291909925,-2.50208507089,-1.50125104254,-
0.500417014179,0.500417014178,1.50125104254,2.50208507089,3.50291909925,4.503753127
61,5.50458715596,6.50542118432,7.50625521268,8.50708924103,9.50792326939,10.5087572
977,11.5095913261,12.5104253545,13.5112593828,14.5120934112,15.5129274395,16.513761
4679,17.5145954962,18.5154295246,19.516263553,20.5170975813,21.5179316097,22.518765
638,23.5195996664,24.5204336947,25.5212677231,26.5221017515,27.5229357798,28.523769
8082,29.5246038365,30.5254378649,31.5262718932,32.5271059216,33.52793995,34.5287739
783,35.5296080067,36.530442035,37.5312760634,38.5321100917,39.5329441201,40.5337781
485,41.5346121768,42.5354462052,43.5362802335,44.5371142619,45.5379482902,46.538782
3186,47.539616347,48.5404503753,49.5412844037,50.542118432,51.5429524604,52.5437864
887,53.5446205171,54.5454545455,55.5462885738,56.5471226022,57.5479566305,58.548790
6589,59.5496246872,60.5504587156,61.551292744,62.5521267723,63.5529608007,64.553794
829,65.5546288574,66.5554628857,67.5562969141,68.5571309425,69.5579649708,70.558798
9992,71.5596330275,72.5604670559,73.5613010842,74.5621351126,75.562969141,76.5638
031693,77.5646371977,78.565471226,79.5663052544,80.5671392827,81.5679733111,82.5688
073394,83.5696413678,84.5704753962,85.5713094245,86.5721434529,87.5729774812,88.573
8115096,89.5746455379,90.5754795663,91.5763135947,92.577147623,93.5779816514,94.578
8156797,95.5796497081,96.5804837364,97.5813177648,98.5821517932,99.5829858215,100.5
8381985,101.584653878,102.585487907,103.586321935,104.587155963,105.587989992,106.5
8882402,107.589658048,108.590492077,109.591326105,110.592160133,111.592994162,112.5
9382819,113.594662219,114.595496247,115.596330275,116.597164304,117.597998332,118.5
9883236,119.599666389,120.600500417,121.601334445,122.602168474,123.603002502,124.6
0383653,125.604670559,126.605504587,127.606338616,128.607172644,129.608006672,130.6
08840701,131.609674729,132.610508757,133.611342786,134.612176814,135.613010842,136.
613844871,137.614678899,138.615512927,139.616346956,140.617180984,141.618015013,142
.618849041,143.619683069,144.620517098,145.621351126,146.622185154,147.623019183,14
8.623853211,149.624687239,150.625521268,151.626355296,152.627189324,153.628023353,1
54.628857381,155.62969141,156.630525438,157.631359466,158.632193495,159.633027523,1
60.633861551,161.63469558,162.635529608,163.636363636,164.637197665,165.638031693,1
66.638865721,167.63969975,168.640533778,169.641367807,170.642201835,171.643035863,1
72.643869892,173.64470392,174.645537948,175.646371977,176.647206005,177.648040033,1
78.648874062,179.64970809,180.650542118,181.651376147,182.652210175,183.653044204,1
84.653878232,185.65471226,186.655546289,187.656380317,188.657214345,189.658048374,1
90.658882402,191.65971643,192.660550459,193.661384487,194.662218515,195.663052544,1
96.663886572,197.664720601,198.665554629,199.666388657,200.667222686,201.668056714,
202.668890742,203.669724771,204.670558799,205.671392827,206.672226856,207.673060884
,208.673894912,209.674728941,210.675562969,211.676396997,212.677231026,213.67806505
4,214.678899083,215.679733111,216.680567139,217.681401168,218.682235196,219.6830692
24,220.683903253,221.684737281,222.685571309,223.686405338,224.687239366,225.688073
394,226.688907423,227.689741451,228.69057548,229.691409508,230.692243536,231.693077
565,232.693911593,233.694745621,234.69557965,235.696413678,236.697247706,237.698081
735,238.698915763,239.699749791,240.70058382,241.701417848,242.702251877,243.703085
905,244.703919933,245.704753962,246.70558799,247.706422018,248.707256047,249.708090
075,250.708924103,251.709758132,252.71059216,253.711426188,254.712260217,255.713094
245,256.713928274,257.714762302,258.71559633,259.716430359,260.717264387,261.718098
415,262.718932444,263.719766472,264.7206005,265.721434529,266.722268557,267.7231025
85,268.723936614,269.724770642,270.725604671,271.726438699,272.727272727,273.728106
756,274.728940784,275.729774812,276.730608841,277.731442869,278.732276897,279.73311
```

```
0926,280.733944954,281.734778982,282.735613011,283.736447039,284.737281068,285.7381
15096,286.738949124,287.739783153,288.740617181,289.741451209,290.742285238,291.743
119266,292.743953294,293.744787323,294.745621351,295.746455379,296.747289408,297.74
8123436,298.748957465,299.749791493,300.750625521,301.75145955,302.752293578,303.75
3127606,304.753961635,305.754795663,306.755629691,307.75646372,308.757297748,309.75
8131776,310.758965805,311.759799833,312.760633862,313.76146789,314.762301918,315.76
3135947,316.763969975,317.764804003,318.765638032,319.76647206,320.767306088,321.76
8140117,322.768974145,323.769808173,324.770642202,325.77147623,326.772310259,327.77
3144287,328.773978315,329.774812344,330.775646372,331.7764804,332.777314429,333.778
148457,334.778982485,335.779816514,336.780650542,337.78148457,338.782318599,339.783
152627,340.783986656,341.784820684,342.785654712,343.786488741,344.787322769,345.78
8156797,346.788990826,347.789824854,348.790658882,349.791492911,350.792326939,351.7
93160967,352.793994996,353.794829024,354.795663053,355.796497081,356.797331109,357.
798165138,358.798999166,359.799833194,360.800667223,361.801501251,362.802335279,363
.803169308,364.804003336,365.804837364,366.805671393,367.806505421,368.80733945,369
.808173478,370.809007506,371.809841535,372.810675563,373.811509591,374.81234362,375
.813177648,376.814011676,377.814845705,378.815679733,379.816513761,380.81734779,381
.818181818,382.819015847,383.819849875,384.820683903,385.821517932,386.82235196,387
.823185988,388.824020017,389.824854045,390.825688073,391.826522102,392.82735613,393
.828190158,394.829024187,395.829858215,396.830692244,397.831526272,398.8323603,399.
833194329,400.834028357,401.834862385,402.835696414,403.836530442,404.83736447,405.
838198499,406.839032527,407.839866555,408.840700584,409.841534612,410.842368641,411
.843202669,412.844036697,413.844870726,414.845704754,415.846538782,416.847372811,41
7.848206839,418.849040867,419.849874896,420.850708924,421.851542952,422.852376981,4
23.853211009,424.854045038,425.854879066,426.855713094,427.856547123,428.857381151,
429.858215179,430.859049208,431.859883236,432.860717264,433.861551293,434.862385321
,435.863219349,436.864053378,437.864887406,438.865721435,439.866555463,440.86738949
1,441.86822352,442.869057548,443.869891576,444.870725605,445.871559633,446.87239366
1,447.87322769,448.874061718,449.874895746,450.875729775,451.876563803,452.87739783
2,453.87823186,454.879065888,455.879899917,456.880733945,457.881567973,458.88240200
2,459.88323603,460.884070058,461.884904087,462.885738115,463.886572143,464.88740617
2,465.8882402,466.889074229,467.889908257,468.890742285,469.891576314,470.892410342
,471.89324437,472.894078399,473.894912427,474.895746455,475.896580484,476.897414512
,477.89824854,478.899082569,479.899916597,480.900750626,481.901584654,482.902418682
,483.903252711,484.904086739,485.904920767,486.905754796,487.906588824,488.90742285
2,489.908256881,490.909090909,491.909924937,492.910758966,493.911592994,494.9124270
23,495.913261051,496.914095079,497.914929108,498.915763136,499.916597164,500.917431
193,501.918265221,502.919099249,503.919933278,504.920767306,505.921601334,506.92243
5363,507.923269391,508.92410342,509.924937448,510.925771476,511.926605505,512.92743
9533,513.928273561,514.92910759,515.929941618,516.930775646,517.931609675,518.93244
3703,519.933277731,520.93411176,521.934945788,522.935779817,523.936613845,524.93744
7873,525.938281902,526.93911593,527.939949958,528.940783987,529.941618015,530.94245
2043,531.943286072,532.9441201,533.944954128,534.945788157,535.946622185,536.947456
214,537.948290242,538.94912427,539.949958299,540.950792327,541.951626355,542.952460
384,543.953294412,544.95412844,545.954962469,546.955796497,547.956630525,548.957464
554,549.958298582,550.959132611,551.959966639,552.960800667,553.961634696,554.96246
8724,555.963302752,556.964136781,557.964970809,558.965804837,559.966638866,560.9674
72894,561.968306922,562.969140951,563.969974979,564.970809008,565.971643036,566.972
477064,567.973311093,568.974145121,569.974979149,570.975813178,571.976647206,572.97
7481234,573.978315263,574.979149291,575.979983319,576.980817348,577.981651376,578.9
82485405,579.983319433,580.984153461,581.98498749,582.985821518,583.986655546,584.9
87489575,585.988323603,586.989157631,587.98999166,588.990825688,589.991659716,590.9
92493745,591.993327773,592.994161802,593.99499583,594.995829858,595.996663887,596.9
97497915,597.998331943,598.999165972,600.0)
altd  <- c(76)
x <- seq.int((length(p)-1)/2*-1,(length(p)-1)/2)
pdf('jd_sorted_mdup_model.pdf',height=6,width=6)
plot(x,p,type='l',col=c('red'),main='Peak Model',xlab='Distance to the
middle',ylab='Percentage')
```

```
lines(x,m,col=c('blue'))
legend('topleft',c('forward tags','reverse tags'),lty=c(1,1,1),col=c('red','blue'))
plot(xcorr,ycorr,type='l',col=c('black'),main='Cross-Correlation',xlab='Lag between
+ and - tags',ylab='Correlation')
abline(v=altd,lty=2,col=c('red'))
legend('topleft','alternative lag(s)',lty=2,col='red')
legend('right','alt lag(s) : 76',bty='n')
dev.off()

==> ./sanchari/j455/runMacs2.sh <==
#!/bin/bash
#$ -cwd
#$ -N macs2
#$ -j n
#$ -l h_vmem=9.8G
#$ -pe smp 1
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

#run mac2 peak finding software on a single target bam
#usually we need a control file, igg or input
#
#usage:
#qsub runMacs.sh bamfile.bam

BAM1=$1
BAM2=$2
NAME=C${BAM1%.*}_T${BAM2%.*}
FOL=macs_${NAME}

#module load MACS2/2.1.0/python.2.7.8
module load MACS2/2.1.0-update/python.2.7.8

echo $BAM1
echo $BAM2
echo $NAME
echo $FOL

macs2 callpeak -f BAMPE -c $BAM1 -t $BAM2 -n $NAME --outdir $FOL --verbose 3

==> ./sanchari/j455/dl1.sh <==
wget http://waspsystem.einstein.yu.edu/wsfile/get/file/F8393993-BB15-4027-A8F5-
F8DAFCB5D824/J455_BC8P19ACXX_Lane4_AGGCAGAA-CTCTCTAT_HPNE.1_val_1.fq.gz
wget http://waspsystem.einstein.yu.edu/wsfile/get/file/DCEE1D23-B598-44EE-83D0-
5C5EBEC8E437/J455_BC8P19ACXX_Lane4_AGGCAGAA-CTCTCTAT_HPNE.2_val_2.fq.gz
wget http://waspsystem.einstein.yu.edu/wsfile/get/file/E584F273-7DC8-4250-B1AF-
E72F74A920D6/J455_BC8P19ACXX_Lane4_TCCTGAGC-AGAGTAGA_JD13D.1_val_1.fq.gz
wget http://waspsystem.einstein.yu.edu/wsfile/get/file/45C3D671-AE99-42BE-B947-
02F97404AADD/J455_BC8P19ACXX_Lane4_TCCTGAGC-AGAGTAGA_JD13D.2_val_2.fq.gz
wget http://waspsystem.einstein.yu.edu/wsfile/get/file/82E79983-BD9E-4047-8650-
A332197E5AEA/J455_BC8P19ACXX_Lane4_AGGCAGAA-CTCTCTAT_HPNE.1_val_1_fastqc.html
wget http://waspsystem.einstein.yu.edu/wsfile/get/file/AABF0427-112C-49A0-A87A-
0F4F1BBED9FF/J455_BC8P19ACXX_Lane4_AGGCAGAA-CTCTCTAT_HPNE.2_val_2_fastqc.html
wget http://waspsystem.einstein.yu.edu/wsfile/get/file/9352EF19-6099-4513-8E90-
FB658070C563/J455_BC8P19ACXX_Lane4_TCCTGAGC-AGAGTAGA_JD13D.1_val_1_fastqc.html
wget http://waspsystem.einstein.yu.edu/wsfile/get/file/E8FA916F-7A70-41EE-898A-
EB9A32E0FE1F/J455_BC8P19ACXX_Lane4_TCCTGAGC-AGAGTAGA_JD13D.2_val_2_fastqc.html
wget http://waspsystem.einstein.yu.edu/wsfile/get/file/A251EADA-1529-4880-B63E-
```

```
6E616A5D76D3/J455_BC8P19ACXX_Lane4_AGGCAGAA-CTCTCTAT_HPNE.1_val_1_screen.png
wget http://waspsystem.einstein.yu.edu/wsfile/get/file/2C00DBA4-EAE5-4357-877C-
8629BA783269/J455_BC8P19ACXX_Lane4_TCCTGAGC-AGAGTAGA_JD13D.1_val_1_screen.png

==> ./sanchari/j455/commands1.sh <==
qsub runBowtie.sh hpne J455_BC8P19ACXX_Lane4_AGGCAGAA-CTCTCTAT_HPNE.1_val_1.fq.gz
J455_BC8P19ACXX_Lane4_AGGCAGAA-CTCTCTAT_HPNE.2_val_2.fq.gz

qsub runBowtie.sh jd  J455_BC8P19ACXX_Lane4_TCCTGAGC-AGAGTAGA_JD13D.1_val_1.fq.gz
J455_BC8P19ACXX_Lane4_TCCTGAGC-AGAGTAGA_JD13D.2_val_2.fq.gz



SNAME=hpne
FQ1=J455_BC8P19ACXX_Lane4_AGGCAGAA-CTCTCTAT_HPNE.1_val_1.fq.gz
FQ2=J455_BC8P19ACXX_Lane4_AGGCAGAA-CTCTCTAT_HPNE.2_val_2.fq.gz
REF=../../hg19/hg19

qsub runMacs.sh hpne_sorted_mdup.bam
qsub runMacs.sh jd_sorted_mdup.bam



qsub runMacs2.sh jd_sorted_mdup.bam hpne_sorted_mdup.bam
qsub runMacs2.sh hpne_sorted_mdup.bam jd_sorted_mdup.bam



==> ./sanchari/j455/runBowtie.sh <==
#!/bin/bash
#$ -cwd
#$ -N bowtie_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 2
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

#process a pair of fastqs
#
#usage:
#qsub runBowtie.sh aligned_name reads1.fastq.gz reads2.fastq.gz

SNAME=$1
FQ1=$2
FQ2=$3
REF=../../hg19/hg19

module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20


bowtie2 -t -p 2 -q --local -x $REF -1 <(zcat $FQ1) -2 <(zcat $FQ2) 2>| summary_$
{SNAME}.txt | samtools view -bS - >| ${SNAME}.bam
java -jar $(which SortSam.jar) INPUT=${SNAME}.bam OUTPUT=${SNAME}_sorted.bam
SO=coordinate CREATE_INDEX=true
echo "done" >| $SNAME.status
```

```
==> ./sanchari/j455/se_macs_hpne_sorted_mdup/hpne_sorted_mdup_model.r <==
# R script for Peak Model
#  -- generated by MACS
p <-
c(0.0214117263579,0.0251867133745,0.0289751225671,0.0326427321752,0.0363808082076,0
.0366526072728,0.0369613173222,0.0370988946268,0.0370183615704,0.0371156723468,0.03
7159294419,0.0369411840581,0.0369143397059,0.036931117426,0.036924406338,0.03704520
59225,0.036931117426,0.0372767384595,0.0376525593892,0.03783711431,0.0381324021833,
0.0384713121288,0.038300179384,0.0383203126481,0.0382196463277,0.038286757208,0.038
1995130636,0.0380384469509,0.037840469854,0.0377062480934,0.0377599367977,0.0376794
037413,0.0379780471586,0.0380384469509,0.0382397795918,0.0379411361745,0.0381189800
072,0.0382196463277,0.0382498462238,0.0383303792802,0.0379612694385,0.0379075807343
,0.038055224671,0.0380820690231,0.0380183136868,0.0382230018717,0.0381928019756,0.0
381961575196,0.0383438014562,0.0384344011446,0.0386491559616,0.0389075328507,0.0388
337108824,0.0388504886024,0.0390417546113,0.0392766426923,0.039437708805,0.03960213
04617,0.0396155526378,0.0398034631026,0.0398772850709,0.039900773879,0.040008151287
5,0.0403302835129,0.0402598170886,0.0401960617523,0.0402262616485,0.0403034391608,0
.0403000836168,0.0403202168809,0.0403739055851,0.0404745719056,0.0403638389531,0.04
057859377,0.0407061044426,0.0408738816433,0.0412262137649,0.0411523917965,0.0413939
909656,0.0413201689973,0.0415315682702,0.0417865896153,0.0421254995608,0.0420013444
323,0.0422496546894,0.0424107208021,0.0421288551048,0.042407365258,0.0425348759306,
0.0428872080522,0.0426456088831,0.0426690976912,0.0425650758267,0.0426623866032,0.0
428804969641,0.0430147187247,0.043538183591,0.0435012726068,0.0437026052477,0.04341
73840065,0.0435012726068,0.0435549613111,0.0436791164396,0.0438435380963,0.04366569
42636,0.0438368270083,0.0436724053516,0.0438502491844,0.0438435380963,0.04415560368
97,0.0445347801634,0.0445549134275,0.0446891351881,0.0445985354997,0.0447864459645,
0.0447797348765,0.0449676453413,0.0448535568448,0.044907245549,0.0452394444065,0.04
55783354352,0.0458535089612,0.0460145750739,0.0462997963151,0.0462595297869,0.046142
0857464,0.0461219524823,0.0464273069877,0.0463534850194,0.0461756411866,0.045994441
8098,0.0459508197376,0.0460850414982,0.0461487968345,0.0464038181796,0.046407173723
6,0.0466655506127,0.0468031279173,0.047192371023,0.0473098150636,0.0476520805531,0.
0475950363048,0.047648725009,0.047631947289,0.0475145032485,0.047860124282,0.047793
0134017,0.0477762356816,0.0477057692573,0.047876902002,0.0479037463542,0.0480983679
07,0.0481856120514,0.0483601003402,0.0489842315269,0.0489607427188,0.0491385865516,
0.0493466302805,0.0494405855129,0.0498466063387,0.0500278057155,0.0500177390834,0.0
498298286186,0.0499908947313,0.0496385626098,0.0498063398105,0.0498231175306,0.0496
85540226,0.0501083387718,0.0499372060271,0.049910361675,0.0500915610518,0.050625092
5501,0.0506217370061,0.0508868249832,0.0508834694392,0.0511150019762,0.051302912441
1,0.0512592903689,0.0513264012492,0.0514740451858,0.0514572674657,0.0511686906805,0
.0513935121295,0.0514807562739,0.0518767104676,0.0521753538849,0.0521350873567,0.05
20042211401,0.0520713320204,0.0520075766842,0.0523397755416,0.0527457963674,0.05291
02180241,0.0526820410311,0.0525343970945,0.0523028645575,0.0526015079748,0.05276257
40875,0.0529605511843,0.0531987948094,0.0529941066245,0.0528397515998,0.05263170787
09,0.0529873955365,0.0532927500418,0.0533195943939,0.0535142159468,0.0536618598834,
0.0536350155313,0.0537423929398,0.0541249249575,0.0544537682709,0.054487323711,0.05
46047677516,0.0545309457832,0.0548027448484,0.0548732112727,0.0548396558326,0.05525
23877464,0.0553027209066,0.0555074090915,0.0554973424595,0.0558228302289,0.05612482
91902,0.0559872518856,0.0559302076374,0.0562154288786,0.0564436058716,0.05632616183
11,0.0563395840072,0.0561583846304,0.0566382274245,0.0567053383048,0.0568664044175,
0.0573361805795,0.0575240910444,0.0574267802679,0.0577925345655,0.0579401785022,0.0
581717110392,0.0584703544565,0.0584233768403,0.058805908858,0.059037441395,0.059248
8406679,0.0596045283335,0.0598259942385,0.0598696163107,0.0600776600396,0.060013904
7033,0.060396436721,0.0608158797228,0.0611413674922,0.0613963888374,0.0614802774377
,0.061537321686,0.0618695205434,0.0621010530804,0.0622587636491,0.062688273283,0.06
29869167003,0.0627990062355,0.0627956506915,0.0628929614679,0.0633023378377,0.06352
71592867,0.0635003149346,0.0637888917198,0.0641278016653,0.0644398672587,0.06438282
30105,0.0648291103644,0.0653257308786,0.0657384627924,0.0658491957449,0.06621159449
```

```
85,0.0667249927327,0.067251813143,0.0673659016395,0.0674095237117,0.0677316559371,0
.0679464107541,0.0680806325147,0.0680000994583,0.0685034310605,0.0690772290871,0.06
93423170642,0.0695637829692,0.0698792041066,0.0703489802686,0.0707080234782,0.07099
66002635,0.0710368667916,0.0714797986016,0.0716509313463,0.0718187085471,0.07210392
97883,0.0722415070929,0.0727481942392,0.0735635914347,0.0740300120528,0.07464407660
75,0.0756641619879,0.0762648043666,0.0768453134811,0.0774862223879,0.0778016435253,
0.0783318194797,0.0791337944992,0.0796404816454,0.0800632801912,0.0804122567688,0.0
806303671297,0.0811940985242,0.0816672302303,0.0817074967585,0.0823215613132,0.0824
893385139,0.0823886721934,0.0823584722973,0.0828081151953,0.0830631365404,0.0834490
241021,0.0835329127025,0.0834456685581,0.0837711563275,0.084120132905,0.08476775289
99,0.0855395280233,0.0861133260498,0.0862408367223,0.0868985233492,0.0873951438634,
0.0878917643776,0.0882810074833,0.0885997841647,0.0888581610538,0.0892440486155,0.0
894286035363,0.0894722256085,0.0897675134818,0.0898614687142,0.0898245577301,0.0900
795790752,0.0905493552373,0.0908312209345,0.0910426202074,0.0909016873588,0.0909385
98343,0.0912338862163,0.091861372947,0.0920526389558,0.0922204161566,0.092408326621
4,0.0927875030951,0.0931364796726,0.0934921673382,0.0940693209087,0.094129720701,0.
0945827191429,0.094589430231,0.0951565171695,0.0953343610022,0.0956028045234,0.0957
638706361,0.0962739133264,0.096418201719,0.0966027566398,0.0970725328019,0.09683428
91768,0.0970523995378,0.0967638227525,0.0971228659621,0.0972738654428,0.09731413197
09,0.0972268878265,0.0974483537315,0.0978476634693,0.0981261736225,0.0985959497846,
0.0988778154818,0.0994985911245,0.0999885005507,0.100213322,0.100813964378,0.101065
630179,0.10159245059,0.101934716079,0.102434692137,0.102740046643,0.103001779076,0.
103226600525,0.103384311094,0.103709798863,0.103884287152,0.104253396993,0.10381717
6271,0.103998375648,0.103911131504,0.104152730673,0.104578884763,0.104931216884,0.1
05159393877,0.105400993046,0.105867413664,0.106166057082,0.107051920702,0.107427741
631,0.107776718209,0.108384071675,0.108585404316,0.108840425661,0.109454490216,0.10
9645756225,0.109977955082,0.110477931141,0.111182595384,0.111498016521,0.1123369025
25,0.112940900447,0.113638853603,0.114236140437,0.114793160744,0.115393803122,0.116
014578765,0.116880309121,0.11717224145,0.117692350772,0.118356748487,0.118917124338
,0.119735877077,0.120517718833,0.121138494475,0.122423667833,0.123108198812,0.12367
8641295,0.12449403849,0.125987255577,0.126530853707,0.127316051007,0.127735494008,0
.128248892243,0.128950200942,0.129503865704,0.13028570746,0.131399748072,0.13209434
5684,0.132235278532,0.133268786089,0.134037205668,0.134963335816,0.135765310836,0.1
36758551864,0.137681326468,0.138271902215,0.139073877234,0.140328850696,0.141352291
62,0.14186233431,0.142889130779,0.143882371807,0.145284989205,0.146060119873,0.1465
26540491,0.147566759135,0.148100290634,0.148647244308,0.149734440569,0.150915592062
,0.151677300553,0.152714163654,0.153230917432,0.154469113174,0.155334843529,0.15636
1639998,0.157438769627,0.15863669884,0.159777583805,0.160596336545,0.161723799334,0
.162753951346,0.16369350367,0.16456258957,0.166089362097,0.167166491725,0.168293954
514,0.169418061759,0.170213325691,0.171696476145,0.172552139869,0.173535314265,0.17
4924509487,0.176035194556,0.177139168537,0.178316964486,0.179360538675,0.1800383585
66,0.18092757773,0.181773174821,0.183148947867,0.184585120706,0.185470984326,0.1863
87047842,0.187336666798,0.188423863059,0.189819769369,0.191641829769,0.192776003646
,0.193947088507,0.194658463838,0.195853037507,0.197517387339,0.199158248362,0.20047
0266071,0.201735306165,0.202547347817,0.203627832989,0.20507407246,0.206530378562,0
.207855818448,0.208865837196,0.210067121954,0.211788516033,0.213278377576,0.2146843
50518,0.21658358843,0.218221093909,0.219466000739,0.221049817514,0.222643700921,0.2
2382149687,0.225046270435,0.225606646286,0.226767664515,0.228331348026,0.2296970544
4,0.23076747298,0.232458667164,0.233753907154,0.23528067968,0.236901407439,0.238525
490742,0.240102596429,0.241605880148,0.242592410088,0.243042052986,0.243766850493,0
.244998335147,0.245585555349,0.246464707881,0.247229771917,0.247947858336,0.2485753
45067,0.249149143093,0.249323631382,0.250618871371,0.25110206971,0.251380579863,0.2
52279865659,0.252420798507,0.252689242029,0.252712730837,0.252877152493,0.253232840
159,0.253407328448,0.252729508557,0.253219417983,0.252461065036,0.25171277872,0.251
377224319,0.251263135822,0.25028331697,0.249320275838,0.248159257609,0.247280105077
,0.246716373682,0.244998335147,0.243870872358,0.242988364282,0.241183081602,0.23931
0688042,0.23833086919,0.236804096663,0.235260546416,0.2333076198,0.231660047688,0.2
30482251739,0.228750791028,0.227012619228,0.226452243377,0.225603290742,0.224871782
147,0.225620068462,0.226193866488,0.227603194975,0.227713927927,0.2278414386,0.2282
```

```
44103881,0.227391795702,0.226234133017,0.224405361528,0.221237727979,0.217137253193
,0.212150914787,0.207144443117,0.202738613825,0.198342851166,0.193531001049,0.18983
3191545,0.18642731437,0.183870389831,0.182558372121,0.180900733378,0.180145735974,0
.179360538675,0.178263275782,0.177494856203,0.177253257034,0.17687408056,0.17601841
6836,0.175075508968,0.174380911357,0.173099093543,0.172444762461,0.171676342881,0.1
71206566719,0.170240170043,0.169595905592,0.168495287155,0.167894644777,0.166636315
771,0.165495430806,0.165428319926,0.164626344906,0.163683437038,0.162915017459,0.16
1968754047,0.161153356851,0.160911757682,0.160119849294,0.159945361006,0.1586937430
88,0.157572991387,0.156569683727,0
.
155546242802,0.154395291205,0.153777871107,0.152519542101,0.150959214134,0.14997603
9738,0.148449267211,0.14760367012,0.14639231873,0.145365522262,0.143942771599,0.142
936108395,0.141587179701,0.141107336907,0.140590583129,0.139714786141,0.13889267785
7,0.137882659109,0.137080684089,0.135959932388,0.135486800682,0.134657981311,0.1336
27829298,0.131986968275,0.130725283726,0.129356221768,0.128406602811,0.127299273286
,0.126295965626,0.125557745943,0.124587993723,0.123416908861,0.122547822962,0.12214
515768,0.121423715717,0.120668718313,0.119745943709,0.119286234179,0.118497681336,0
.117440684971,0.116957486633,0.116199133686,0.115611913483,0.1149575824,0.114276406
965,0.113739519923,0.113212699513,0.112467768741,0.111709415794,0.111004751551,0.11
0075265859,0.109182691151,0.108444471468,0.10765256308,0.10678347718,0.106317056562
,0.105743258536,0.104625862379,0.104038642176,0.103588999278,0.102887690579,0.10269
3069027,0.102401136697,0.10155889515,0.100760275674,0.0998710565101,0.0991261257389
,0.0989818373462,0.0980557071982,0.0975456645079,0.096656445344,0.0957168930199,0.0
946666077433,0.0939518768682,0.0934821007061,0.0928411917993,0.0922002828925,0.0915
660850737,0.0912640861124,0.0905359330612,0.0902473562759,0.0899554239466,0.0895359
809448,0.0892138487194,0.0885024733883,0.0877038539128,0.0870562339179,0.0863951917
47,0.0855965722715,0.0851804848137,0.0847643973559,0.0839624223364,0.0833349356056,
0.0829389814119,0.0824658497058,0.0823047835931,0.0820061401758,0.0816873634944,0.0
814994530295,0.0809021661949,0.0804760121051,0.0803485014325,0.0802746794642,0.0796
941703496,0.0796572593655,0.0791337944992,0.0788049511857,0.0783519527437,0.0778486
211415,0.0775868887084,0.0769292020815,0.0763285597029,0.075909116701,0.07573127286
82,0.0746272988874,0.07431187775,0.0736038579629,0.0732448147533,0.0726139724786,0.
0720468855401,0.07157039829,0.0711509552881,0.0703624024447,0.0700067147791,0.07012
75143637,0.06974833789,0.0692718506399,0.0687517413176,0.0688222077419,0.0686443639
092,0.0684497423563,0.0682819651556,0.0682886762436,0.0675403899283,0.0671779911747
,0.0669229698296,0.0666847262046,0.066567282164,0.065872684553,0.0654364638311,0.06
50908427976,0.0646177110915,0.0643895340985,0.0644935559629,0.0643459120263,0.06411
77350333,0.0643626897464,0.064057335241,0.0638492915121,0.0636781587674,0.063520448
1987,0.0634936038466,0.0630607386687,0.0625406293464,0.0622822524572,0.062168163960
7,0.0616983877987,0.0614836329817,0.0615876548462,0.0612621670768,0.0611011009641,0
.060856146251,0.0605843471858,0.0605205918495,0.0604702586893,0.0602353706083,0.060
2253039762,0.0600206157913,0.059721972374,0.0595944617014,0.059487084293,0.05935286
25324,0.0592085741398,0.059044152483,0.0583898214002,0.0580005782945,0.057712001509
2,0.0573764471077,0.0570576704263,0.0570677370584,0.0568060046252,0.0565878942643,0
.0566080275283,0.0563395840072,0.0564234726075,0.0562288510547,0.0559771852536,0.05
58765189331,0.0557322305405,0.055386609507,0.0551215215298,0.0548597890967,0.054688
6563519,0.0546785897199,0.0544135017427,0.0542356579099,0.0540678807092,0.054192035
8377,0.0539269478606,0.0538866813324,0.0537692372919,0.0534034829943,0.053098128488
9,0.0525041971983,0.0523934642459,0.0522525313972,0.0518699993796,0.0515378005221,0
.0515512226982,0.0512223793847,0.051279423633,0.051279423633,0.0512727125449,0.0513
700233214,0.0515780670503,0.051514311714,0.0516284002105,0.0516888000028,0.05145726
74657,0.0515478671541,0.0512458681928,0.0512089572087,0.0508767583512,0.05073582550
26,0.0500680722437,0.0499405615711,0.0495278296573,0.0489540316308,0.0488500097663,
0.0484507000286,0.0481218567151,0.0480144793066,0.0480446792028,0.0477493913295,0.0
478265688418,0.0474339701921,0.0475044366164,0.0475715474967,0.0474306146481,0.0473
500815917,0.0474574590002,0.0472729040794,0.0470313049103,0.0470178827343,0.0468266
167254,0.0467527947571,0.0466017952764,0.0464709290599,0.046276307507,0.04620584108
27,0.0460313527939,0.0458870644013,0.0454676213995,0.0453870883431,0.0451756890702,
0.0450381117656,0.0448200014046,0.0445750466916,0.0443904917707,0.0440952038975,0.0
```

```
438066271122,0.0433938951984,0.0434979170628,0.0434207395505,0.0433536286702,0.0431
388738533,0.0428804969641,0.0430113631807,0.0428234527159,0.0426657421472,0.0422395
880573,0.0422563657774,0.0422228103372,0.0420382554164,0.0417966562474,0.0415483459
903,0.0414074131417,0.0410752142842,0.0410819253722,0.0411356140765,0.0413503688934
,0.041266480293,0.0409678368757,0.0408873038194,0.0410383033,0.0412295693089,0.0412
362803969,0.0409577702437,0.0407698597789,0.0407363043387,0.0403739055851,0.0401994
172964,0.0402430393686,0.0402497504566,0.0400685510798,0.0400081512875,0.0400383511
837,0.0400987509759,0.040112173152,0.0398605073509,0.0398403740868,0.0398403740868,
0.0398068186466,0.03965917471,0.0397195745023,0.0395182418614,0.0393504646606,0.039
1625541958,0.0391524875638,0.0391357098437,0.0390618878754,0.0390551767873,0.038837
0664264,0.0385854006253,0.0383572236323,0.0383807124404,0.0382666239439,0.038400845
7045,0.0382464906798,0.0379176473663,0.0381860908875,0.0378807363822,0.037914291822
3,0.0378639586621,0.0377297369015,0.0377330924455,0.0377297369015,0.0375351153487,0
.0375854485089,0.0377465146216,0.037602226229,0.0379277139984,0.0377733589737,0.037
9746916146,0.0379847582466,0.0378941585582,0.037830403222,0.037840469854,0.03773309
24455,0.0378203365899,0.0375686707888,0.0374545822923,0.0374914932765,0.03748813773
25,0.0372666718275,0.037145872243,0.0371290945229,0.0369613173222,0.0368841398098,0
.0366358295527,0.0368505843697,0.036921050794,0.0368304511056,0.0365250966002,0.036
464696808,0.0364076525597,0.0365083188802,0.0366794516249,0.0365955630245,0.0366156
962886,0.0365418743203,0.0364982522481,0.0365552964964,0.0364848300721,0.0366458961
848,0.0365016077921,0.0364009414717,0.0361022980544,0.0362130310069,0.0364344969118
,0.0363942303837,0.0363170528713,0.0363069862393,0.0363036306953,0.0361996088308,0.
0364110081037,0.036464696808,0.0365922074805,0.0363237639594,0.0359814984699,0.0360
11698366,0.0357566770209,0.0357197660367,0.0359412319417,0.0358741210614,0.03594123
19417,0.0358539877973,0.0359546541177,0.0361358534945,0.0363371861354,0.03619960883
08,0.0364546301759,0.0364378524559,0.0360687426142,0.0361190757745,0.0360821647903,
0.0360486093501,0.0359110320455,0.0357667436529,0.0354983001317,0.0354009893553,0.0
351895900824,0.0352600565067,0.0352936119468,0.0351291902901,0.0351392569222,0.0350
889237619,0.0349849018975,0.0352164344345,0.035330522931,0.0353506561951,0.03523321
21546,0.0349010132971,0.0350218128816,0.0351694568183,0.0348540356809,0.03492114656
12,0.0350151017936,0.0347063917443,0.0346728363041,0.0347231694643,0.0347667915365,
0.0346795473921,0.0345990143358,0.0344580814872,0.0346761918481,0.034645991952,0.03
46090809678,0.03463592532,0.0344345926791,0.0343003709185,0.0343372819026,0.0345621
033516,0.0345822366157,0.0345587478076,0.0341762157899,0.0340319273973,0.0340789050
135,0.0338809279166,0.0339513943409,0.0338842834607,0.0339446832529,0.033954749885,
0.0337668394201,0.0338037504043,0.0339178389008,0.0338842834607,0.033739995068,0.03
38943500927,0.0339044167247,0.0338943500927,0.0337735505082,0.033511818075,0.033397
7295785,0.0334480627387,0.0334950403549,0.0333876629465,0.0332400190098,0.033286996
626,0.0331963969376,0.0332668633619,0.0330890195292,0.0332333079218,0.0336493953796
,0.0334681960028,0.0334816181789,0.0335252402511,0.0336795952758,0.0336259065715,0.
0334178628426,0.033276929994,0.0335285957951,0.0334581293708,0.0333104854341,0.0333
809518584,0.0333708852264,0.033276929994,0.03327357445,0.0332198857457,0.0334044406
665,0.0332433745538,0.0329648644006,0.0329917087527,0.0326695765273,0.0325789768389
,0.0326125322791,0.0327098430555,0.0326628654393,0.0328004427439,0.0327870205678,0.
0329883532087,0.0330957306172,0.0328943979763,0.0329883532087,0.0331125083373,0.032
8675536242,0.0328641980802,0.032823931552,0.0325554880308,0.0328675536242,0.0328541
314481,0.0330084864728,0.0330118420168,0.0328876868883,0.0329380200485,0.0330084864
728,0.0329044646084,0.0329950642968,0.0329346645045,0.0327064875115,0.0327333318636
,0.032589043471,0.0327970871999,0.0328541314481,0.0326897097914,0.0324984437826,0.0
32364222022,0.0322031559093,0.0323877108301,0.0322937555977,0.032146111661,0.031948
1345642,0.0318810236839,0.0320521564286,0.0320991340448,0.0320118899004,0.032172956
0131,0.0321528227491,0.0321729560131,0.0320689341487,0.0319380679321,0.032011889900
4,0.0319615567402,0.031921290212,0.0318810236839,0.0319716233723,0.0319783344603,0.
031904512492,0.0317367352912,0.0318072017155,0.0320286676205,0.0320756452367,0.0321
528227491,0.0320253120765,0.0317971350835,0.0320689341487,0.0320118899004,0.0320622
230607,0.0319682678282,0.0320253120765,0.0320118899004,0.0320118899004,0.0318441126
997,0.0319481345642,0.032146111661,0.0319783344603,0.0318810236839,0.0319548456522,
0.031904512492,0.0318139128036,0.0318709570518,0.0286160793575,0.0254618679837,0.02
22304790975,0.0190158679314,0.0)
```

```
m <-
c(0.0189074088781,0.0221498297863,0.0253049805666,0.0285776103653,0.0315548643463,0
.0315682905198,0.0314508115014,0.0314004633507,0.031491090022,0.0315078727389,0.031
6555606478,0.031692482625,0.0315917863235,0.0315145858257,0.0317696831228,0.0318703
794243,0.0320449196803,0.0320012846163,0.0319274406619,0.0318603097942,0.0318737359
677,0.0316857695383,0.0317327611456,0.0317629700361,0.0317394742324,0.0317596134927
,0.0317831092964,0.0319173710317,0.0318770925111,0.031981145356,0.0320079977031,0.0
32182537959,0.0323772174752,0.0322664515436,0.0323671478451,0.032340295498,0.032219
4599362,0.0323604347583,0.0324477048863,0.03269944564,0.0325148357539,0.03237050438
84,0.0321623986987,0.0321556856119,0.0318871621413,0.0318200312736,0.0317025522552,
0.0315414381728,0.0315615774331,0.0313568282867,0.0313702544602,0.0313366890264,0.0
314608811316,0.0313836806338,0.0314978031088,0.0313870371771,0.0315951428669,0.0314
776638485,0.0315145858257,0.031491090022,0.0314776638485,0.031665630278,0.031689126
0816,0.0318166747302,0.0319408668354,0.0320348500501,0.0320382065935,0.032239599196
5,0.0323302258678,0.0323000169774,0.0323940001921,0.0322865908039,0.0320885547442,0
.0320851982009,0.0321489725251,0.0323235127811,0.0322630950002,0.0322395991965,0.03
20717720273,0.0321489725251,0.0321489725251,0.0322664515436,0.032414139452 4,0.03253
16184708,0.0325920362517,0.0326893760099,0.0327766461378,0.0330048910879,0.03309887
43026,0.0333136930791,0.0333707543167,0.0334580244446,0.0335083725954,0.03356879037
63,0.0335654338329,0.0333841804902,0.0334412417277,0.0332062836909,0.0334882333351,
0.0334378851843,0.0334748071615,0.0334311720976,0.0335352249424,0.0335553642027,0.0
335553642027,0.033750043719,0.0337299044587,0.0338675227374,0.0336157819836,0.03376
34698925,0.0338775923675,0.0337567568057,0.0338238876734,0.0338104614999,0.03375004
3719,0.0336761997645,0.0338675227374,0.0339312970617,0.0340152106463,0.033860809650
6,0.0338876619977,0.034109193861,0.0340387064499,0.0340185671896,0.0341125504044,0.
0343743607883,0.0342736644868,0.0345220486971,0.0345354748707,0.0347167282134,0.034
6328146288,0.0347469371038,0.0350926610723,0.0351698615701,0.0352269228076,0.035096
0176157,0.0351195134194,0.0348442768619,0.0350053909443,0.0349718255105,0.034975182
0539,0.0349516862502,0.0348610595788,0.0349214773597,0.034914764273,0.0350188171179
,0.0350691652686,0.0351094437892,0.0349483297068,0.0352269228076,0.0353477583694,0.
0354954462783,0.0352940536753,0.0356498472739,0.0358445267902,0.0359754319821,0.035
9653623519,0.0361029806307,0.0363580779278,0.036237242366,0.0362338858226,0.0363245
12494,0.0366803060926,0.0364520611425,0.0364722004028,0.036525905097,0.0367272977,0
.0368716290654,0.0369421164765,0.0367776458507,0.0368649159787,0.0367507935036,0.03
66601668323,0.0367373673301,0.0368582028919,0.0366568102889,0.0366467406588,0.03666
35233757,0.0364520611425,0.0367910720242,0.0367205846132,0.0366735930058,0.03676086
31338,0.0367944285676,0.0368347070882,0.0368716290654,0.0368078547412,0.03678100239
41,0.0367138715264,0.0366903757227,0.0368179243713,0.0368078547412,0.0367340107867,
0.0367138715264,0.0365460443573,0.0367709327639,0.0368716290654,0.0370696651251,0.0
372844839016,0.0371972137736,0.0372844839016,0.0372677011847,0.0372811273582,0.0371
602917964,0.0371636483398,0.0370663085817,0.0370226735177,0.03695554265,0.037073021
6684,0.0371065871023,0.0369018379559,0.0370126038875,0.0370562389515,0.037227422664
1,0.0373415451391,0.0374120325501,0.0374254587237,0.0373281189656,0.0372744142714,0
.0373012666185,0.0374791634178,0.0375865728061,0.0376637733039,0.037603355523,0.037
646990587,0.0378282439297,0.0379222271444,0.0378416701032,0.0378450266466,0.0377376
172583,0.037646990587,0.0376335644134,0.0376637733039,0.0377208345414,0.03763692095
68,0.0373952498332,0.0372777708148,0.0372643446413,0.0373415451391,0.0375664335458,
0.0377879654091,0.0380464192496,0.0380162103591,0.038163898268,0.0383451516107,0.03
82981600033,0.0385801096475,0.0389963210271,0.0392312790639,0.0392379921507,0.03935
54711691,0.0391540785661,0.0392916968448,0.0393454015389,0.0392984099316,0.03946959
36441,0.0395098721647,0.0394360282103,0.0393521146257,0.0393521146257,0.03949980253
46,0.0395568637721,0.0396709862471,0.0399294400876,0.0400704149097,0.0401274761472,
0.0402248159053,0.0405168351797,0.0408021413673,0.0410303863173,0.0409968208835,0.0
409632554497,0.0410941606416,0.0411042302718,0.0412149962034,0.0414499542402,0.0413
996060895,0.0412384920071,0.0411579349659,0.0410639517512,0.0410236732306,0.0412217
092902,0.0413459013954,0.0414029626329,0.0414667369572,0.0415674332587,0.0416446337
565,0.0420004273551,0.0420138535286,0.042262237739,0.0423931429309,0.0424837696023,
0.042735510356,0.0429033375252,0.0430141034569,0.0431953567996,0.0433967494026,0.04
31584348223,0.0433799666856,0.043470593357,0.0436954817637,0.0441385454903,0.044158
```

```
6847506,0.0442963030293,0.0443768600705,0.0444876260021,0.0447427232993,0.045048168
7471,0.0452361351766,0.0453166922178,0.0455885722318,0.0456355638392,0.045558363341
4,0.0458201737253,0.0462330285614,0.0465418305527,0.0466928750049,0.0469311895852,0
.0470285293433,0.0472903397272,0.0472702004669,0.0474447407228,0.0477669688876,0.04
75857155449,0.047639420239,0.047525297764,0.0478072474082,0.047797177778,0.04788444
7906,0.0480321358149,0.0479885007509,0.0480724143355,0.0480556316186,0.048438277564
3,0.04860274819,0.0488578454872,0.0488242800533,0.0491196558711,0.0494620232962,0.0
497641122007,0.0502776633383,0.0502340282743,0.0505931784163,0.0507979275627,0.0509
086934944,0.0510597379466,0.0512577740062,0.0514692362394,0.0516169241482,0.0516202
806916,0.0515262974769,0.0519492219432,0.0521606841763,0.052271450108,0.05250976468
82,0.0523922856698,0.0524527034507,0.0524627730808,0.0527279400081,0.0529628980449,
0.0531340817575,0.0533656832509,0.0533824659678,0.053348900534,0.053191142995,0.053
5603627672,0.0538758778452,0.0540336353842,0.0540134961239,0.0541477578592,0.054604
2477594,0.0545773954123,0.0546042477594,0.0549298324675,0.0553863223677,0.055503801
3861,0.0556246369479,0.0560207090671,0.0563395806885,0.0565879648989,0.056675235026
8,0.0569437584975,0.0576083540874,0.0580312785537,0.0581722533758,0.0585448296913,0
.0589207625503,0.0590113892216,0.0590248153951,0.0591959991077,0.059303408496,0.059
4309571445,0.0591322247834,0.0591456509569,0.059504801099,0.0597632549395,0.0598001
769167,0.0599881433462,0.0604345636161,0.0605621122647,0.0608071399317,0.0610655937
722,0.0614985878686,0.0616865542981,0.0619752170291,0.0620960525909,0.0621128353078
,0.0622907321071,0.0622907321071,0.0627942136146,0.0630895894323,0.0632943385787,0.
0633312605559,0.0633245474691,0.0631533637566,0.0633748956199,0.0637441153921,0.064
1703964017,0.0644590591327,0.0644456329592,0.0646570951923,0.0647880003843,0.065385
4651065,0.0659057293309,0.0663387234273,0.0666710212223,0.0668489180216,0.066996605
9305,0.0674027476798,0.067701480041,0.0678458114064,0.0682083180918,0.0685003373662
,0.0687722173802,0.0690877324583,0.069453595687,0.0698630939798,0.0700342776923,0.0
701249043637,0.0704303498116,0.0710781626845,0.0717897498818,0.0718501676627,0.0721
388303937,0.0727161558556,0.0729544704358,0.0734042472492,0.0740184946883,0.0744011
40634,0.0751530063519,0.0754148167358,0.0756061397086,0.076170038997,0.076559398029
5,0.0767507210023,0.0774824474599,0.077824814885,0.0780664860086,0.0785565413425,0.
078455845041,0.0788384909867,0.0790130312426,0.0793419724942,0.0798152451112,0.0800
334204312,0.0802851611849,0.0804966234181,0.080916191341,0.0813626116109,0.08224873
90641,0.0828629865033,0.0835712171571,0.0841854645963,0.0845479712817,0.08536025478
04,0.0855247254062,0.0862530953204,0.0870318133853,0.0874480247648,0.087746757126,0
.088132759615,0.0886295280358,0.0890323132418,0.0896197083338,0.0900090673663,0.090
549470851,0.090952256057,0.0911502921166,0.0917209044918,0.0921572551316,0.09300646
06076,0.0935737164393,0.0942148162256,0.0947417935367,0.0952184226972,0.09617168101
8,0.0972256356404,0.0985145482996,0.0994308846432,0.10031029901,0.100750006193,0.10
1387749436,0.10212618898,0.103055951497,0.103972287841,0.104445560458,0.10483827603
4,0.104979250856,0.10548944545,0.10618424993,0.106882410954,0.107694694453,0.108315
654979,0.108681518208,0.109591141464,0.110232241251,0.111172073398,0.112108549002,0
.1126422394,0.11333704388,0.114333937265,0.114978393595,0.116143114149,0.1164082810
76,0.11734475668,0.118519546864,0.119268056038,0.119882303478,0.120704656606,0.1215
53862082,0.121949934202,0.123118011299,0.124088052337,0.125511226731,0.12608519565,
0.126404067271,0.127229776944,0.12833743626,0.129683410157,0.1305997465,0.131747684
338,0.132439132274,0.133076875517,0.134463127935,0.135634561575,0.137319546354,0.13
8229169611,0.139259628429,0.140021563777,0.141001674445,0.142025420177,0.1436264913
71,0.144660306733,0.144932186747,0.145979428283,0.146294943361,0.147922866902,0.148
996960784,0.15004420232,0.151380106586,0.152219242432,0.152779785177,0.15343766768,
0.15519313987,0.155995353738,0.157290979484,0.157549433325,0.158277803239,0.1594962
28487,0.159831882825,0.161221491786,0.161912939723,0.16328912251,0.164363216393,0.1
64876767531,0.165695764116,0.166299941925,0.167632489648,0.168797210202,0.169656485
308,0.170435203373,0.171187069091,0.171935578266,0.172345076558,0.173761537866,0.17
421131468,0.175224990781,0.175946647609,0.176208457993,0.177255699528,0.17749401410
8,0.178594960338,0.178987675914,0.179977856212,0.179994638929,0.181307047392,0.1821
05904717,0.182139470151,0.183505583308,0.185613492553,0.188050343049,0.190742290842
,0.194733220925,0.198539541122,0.203487086069,0.208333934714,0.212949181866,0.21796
0501137,0.22194136159,0.224982389895,0.227761607817,0.229547288896,0.230567678085,0
.23024544992,0.229664767915,0.228828988612,0.228617526379,0.228184532283,0.22786901
```

```
7205,0.227926078442,0.228694726877,0.229500297289,0.2300272746,0.232517829791,0.234
098761724,0.235834094653,0.237619775733,0.239264481991,0.240902475162,0.24173825446
5,0.242164535474,0.243580996782,0.244849770181,0.245390173666,0.246930827078,0.2481
2239998,0.248740003962,0.249710045
,
0.250260518115,0.251787745354,0.252717507871,0.253163928141,0.25288533504,0.2532948
33333,0.253194137032,0.253033022949,0.252905474301,0.25221066982,0.252741003675,0.2
52086477715,0.252482549834,0.25173404066,0.252187174017,0.250998957659,0.2499550726
67,0.249246842013,0.248478193578,0.247709545143,0.246393780137,0.245997708018,0.244
81956129,0.243792459015,0.242550537963,0.242127613497,0.240972962573,0.240063339316
,0.239267838534,0.23805276983,0.236649734695,0.235333969689,0.234004778509,0.232749
431284,0.231490727515,0.229795673107,0.229228417275,0.227704546579,0.226039701061,0
.225274409169,0.223609563651,0.22151508058,0.22013554125,0.218443843384,0.217114652
205,0.215882800783,0.214050128096,0.212576605551,0.211301119065,0.210032345666,0.20
8877694742,0.208035202353,0.20628644325,0.205259340975,0.203722044105,0.20241634872
9,0.201254984719,0.19958006957,0.198398566299,0.197146575618,0.19597849852,0.194565
393756,0.19349801296,0.192091621282,0.190856413317,0.189893085366,0.189010314456,0.
188080551939,0.187076945468,0.185757823918,0.184421919651,0.183143076622,0.18195150
3721,0.180971393053,0.179716045828,0.178699013183,0.177426883241,0.1767119395,0.175
412957211,0.17422138431,0.173543362546,0.17245584249,0.171193782178,0.170173392989,
0.169119438367,0.167612350388,0.166343576989,0.164447129978,0.163399888442,0.162745
362482,0.161171143635,0.160479695698,0.160006423081,0.159278053167,0.158214028915,0
.157559502955,0.156884837735,0.156310868816,0.15522334876,0.153565216329,0.15266566
2702,0.151450593997,0.150158324795,0.14869487188,0.147969858509,0.146734650544,0.14
58183142,0.144771072665,0.143770822736,0.142935043434,0.142585962922,0.141965002396
,0.140914404317,0.140108833905,0.139209280279,0.138121760222,0.137118153751,0.13680
935176,0.135778892941,0.135228419826,0.13458732004,0.133422599486,0.132680803398,0.
132241096215,0.131596639885,0.131358325305,0.130509119829,0.129854593869,0.12935446
8905,0.128622742448,0.127666127583,0.126917618409,0.126165752691,0.124947327443,0.1
24343149634,0.123369752053,0.12249369423,0.121399461087,0.121013458598,0.1203790718
98,0.119526509879,0.118965967134,0.118435633279,0.118415494019,0.117687124105,0.116
985606538,0.116378072185,0.115730259312,0.114813922969,0.113961360949,0.11359885426
4,0.113202782145,0.112789927309,0.112199175673,0.111766181577,0.111145221051,0.1108
22992886,0.110124831862,0.109617993811,0.109097729587,0.108453273257,0.107792034211
,0.107563789261,0.106909263301,0.106335294383,0.106325224752,0.105653916076,0.10601
9779305,0.105731116574,0.105308192107,0.105086660244,0.104754362449,0.104375073047,
0.10415018464,0.103851452279,0.103519154484,0.103069377671,0.102163110957,0.1018408
82792,0.101565646235,0.101048738554,0.100877554841,0.100085410603,0.100028349365,0.
0997363300911,0.0993369014285,0.0993570406888,0.0993637537755,0.0991657177159,0.099
1254391953,0.0989508989394,0.0987931414004,0.0984944090393,0.0977895349288,0.097611
6381294,0.0972457749007,0.0967355803064,0.0963428647306,0.0962388118857,0.095997140
7621,0.0956480602502,0.0953493278891,0.0954131022134,0.0953426148023,0.095272127391
3,0.0950539520714,0.0950808044184,0.0948659856419,0.0943423648741,0.0937885352159,0
.093526724832,0.0935602902658,0.0929493993701,0.0928184941781,0.0923855000817,0.092
2008901956,0.0920599153735,0.0915027291719,0.0913651108931,0.0914322417608,0.091435
5983042,0.0907777158011,0.0905863928282,0.0904890530701,0.0902574515767,0.090042632
8001,0.0895794298132,0.0896834826581,0.0895089424022,0.0891262964565,0.088908121136
6,0.0889416865704,0.0889920347212,0.0887570766843,0.0883207260445,0.0879011581216,0
.0878776623179,0.0875252252627,0.0871358662302,0.0869210474537,0.0864914099006,0.08
58838755482,0.0854643076253,0.0850548093326,0.0850480962458,0.0847594335148,0.08427
60912676,0.0842324562037,0.083876662605,0.0838263144543,0.0835510778968,0.083389963
8144,0.0831113707136,0.0825608975988,0.0821346165891,0.0819533632464,0.081657987428
7,0.0811209404873,0.0810940881403,0.08062752861,0.0802717350114,0.0797279749833,0.0
792110673023,0.0788116386396,0.0784692712145,0.0780027116843,0.0774086035054,0.0768
413476737,0.075881376266,0.075092588571,0.0742702354421,0.0739815727111,0.073414316
8793,0.0727933563534,0.0719978555716,0.0715078002376,0.071108371575,0.0708163523006
,0.0706720209352,0.0703095142498,0.0701718959711,0.069987286085,0.0698731636099,0.0
700342776923,0.0696482752033,0.0692656292576,0.0686983734258,0.0684499892154,0.0681
579699411,0.0678961595572,0.0675437225019,0.0669496143231,0.0664897678796,0.0658352
```

```
419198,0.065714406358,0.0656741278374,0.0655063006683,0.0649759668137,0.06466380827
91,0.0645698250643,0.0642845188768,0.0640092823193,0.0639924996024,0.0636434190906,
0.0634655222912,0.0629016230028,0.0628109963315,0.0626935173131,0.0625793948381,0.0
623142279108,0.0620926960475,0.0620289217232,0.0616529888643,0.0616496323209,0.0615
422229326,0.061387821937,0.0611830727906,0.0610186021648,0.0608339922787,0.06053190
33742,0.0604110678124,0.0598370988939,0.0596793413549,0.0596994806152,0.05968269789
83,0.0593940351673,0.059102015893,0.0586186736458,0.0580447047272,0.0579540780559,0
.0575076577859,0.0575680755668,0.0573465437035,0.0569370454107,0.0564738424238,0.05
67088004607,0.0564906251408,0.0566785915702,0.0568363491092,0.0567322962644,0.05680
94967622,0.0567893575019,0.0567893575019,0.0567490789813,0.0565040513143,0.05628923
25378,0.0563194414282,0.0560777703046,0.0560139959803,0.0556447762082,0.05556086262
36,0.0554366705184,0.0551278685271,0.0550405983992,0.0548324927094,0.0546680220836,
0.0547015875175,0.0544330640468,0.0543457939188,0.054671378627,0.0543525070056,0.05
39396521695,0.053865808215,0.0533455439906,0.0533824659678,0.0533153351002,0.053050
1681729,0.0530031765655,0.0528051405059,0.0525970348161,0.0525836086426,0.052774931
6155,0.0528252797662,0.0529360456979,0.0527380096382,0.0525265474051,0.052395642213
2,0.0520163528108,0.0518921607057,0.0514021053717,0.0511235122709,0.0505696826126,0
.0502776633383,0.0500494183882,0.0501501146897,0.0501568277765,0.0501836801236,0.05
05629695259,0.0504018554435,0.0504018554435,0.0505159779185,0.0508516322568,0.05071
40139781,0.0509086934944,0.0505965349597,0.0504320643339,0.0504287077905,0.05003263
56713,0.0500057833242,0.0499554351735,0.0499050870228,0.0496567028124,0.04954593688
07,0.0492639872365,0.0493915358851,0.049294196127,0.0492270652593,0.0494150316888,0
.0492371348895,0.0490390988299,0.048777288446,0.0485725392996,0.0484550602812,0.048
3711466966,0.0481663975502,0.0481160493995,0.0479045871663,0.0477132641935,0.047521
9412206,0.0473641836816,0.047323905161,0.0475353673941,0.0476427767824,0.0475185846
772,0.0472836266404,0.0469479723021,0.0466593095711,0.0464780562284,0.0464176384475
,0.0465720394431,0.0466727357446,0.0464411342512,0.0463370814063,0.0464277080777,0.
0464814127718,0.0464075688174,0.0462800201688,0.0463035159725,0.046447847338,0.0461
826804107,0.0459477223739,0.0459846443511,0.04591415694,0.0457765386613,0.045598641
862,0.0454274581494,0.0454006058024,0.045440884323,0.0452864833273,0.0451723608523,
0.0449911075096,0.0447393667559,0.0444507040249,0.0443365815499,0.04434665118,0.044
7326536691,0.0448232803405,0.0446286008242,0.0442862333991,0.0442828768557,0.044202
3198145,0.0441385454903,0.0441721109241,0.0443063726594,0.0443231553763,0.043896874
3667,0.0436652728732,0.0433799666856,0.0434437410099,0.0434135321195,0.043524298051
1,0.043514228421,0.0435880723754,0.043111443215,0.0428932678951,0.0429704683929,0.0
430845908679,0.0429469725892,0.0428093543105,0.042621387881,0.0424434910817,0.04215
14718074,0.0417386169712,0.0416714861035,0.0415808594322,0.0411579349659,0.04078871
51937,0.040896124582,0.0409129072989,0.0408457764313,0.0406846623489,0.040526904809
8,0.0407148712393,0.0408155675408,0.0407585063033,0.0407282974128,0.0407517932165,0
.0405705398738,0.0406544534584,0.04053697444,0.0404732001157,0.0405638267871,0.0404
93339376,0.0404832697459,0.0403187991201,0.0403557210973,0.0401409023208,0.04016439
81244,0.0398858050236,0.0397985348957,0.0396239946397,0.039687768964,0.039664273160
3,0.0392648444977,0.0393487580823,0.0394125324066,0.0395467941419,0.0396340642699,0
.0395367245118,0.0395199417949,0.0395669334022,0.0395904292059,0.039241348694,0.039
5165852515,0.0392480617808,0.0390903042418,0.0388352069447,0.0385163353232,0.038509
6222365,0.0387647195336,0.0388352069447,0.0387747891638,0.0387311540998,0.038439134
8254,0.0387345106432,0.0388452765748,0.0390063906572,0.0391473654793,0.039013103744
,0.0387378671865,0.0385196918666,0.0380430627062,0.0381739678982,0.0382713076563,0.
0379155140576,0.0378450266466,0.0377006952811,0.0373885367465,0.0372945535317,0.037
0058908008,0.036928690303,0.037042812778,0.0370226735177,0.036928690303,0.036787715
4809,0.0365494009006,0.0367440804169,0.0367776458507,0.0369823949971,0.037066308581
7,0.0371468656229,0.0371905006868,0.0370226735177,0.0369421164765,0.0370360996912,0
.0371334394493,0.0369387599331,0.0368749856088,0.033031743435,0.0294335289281,0.025
7144788594,0.0220323507679,0.0)
ycorr <- c(-9.641783324e-05,-9.67063590239e-05,-9.75720481994e-05,-9.90179308984e-
05,-0.000101047778359,-0.000103664768662,-0.000106873524395,-0.000110083865994,-
0.000113300509028,-0.000116528330718,-0.000119772425616,-0.00012303733834,-
0.000126326897757,-0.00012964304828,-0.000132986203639,-0.000136357365723,-
0.000139756720371,-0.000143182965623,-0.00014663439946,-0.000150109883165,-
```

```
0.000153607033787,-0.000157124223235,-0.000160659694493,-0.000164213315908,-
0.000167784584968,-0.000171372049104,-0.000174972309795,-0.000178584816318,-
0.000182207444248,-0.000185837192783,-0.000189472336191,-0.000193110585176,-
0.000196750828478,-0.000200390960823,-0.000204028931967,-0.000207662996948,-
0.000211292837693,-0.000214916448196,-0.000218531010568,-0.000222135616554,-
0.000225729022386,-0.000229310310999,-0.000232878762127,-0.000236432791023,-
0.000239971270794,-0.000243491737354,-0.000246993612217,-0.000250475671337,-
0.000253938017974,-0.000257381123627,-0.000260805354353,-0.000264208728998,-
0.000267589498084,-0.000270946922353,-0.000274278846032,-0.000277584931678,-
0.000280864778782,-0.000284116824288,-0.000287341479051,-0.000290537756917,-
0.000293703622708,-0.000296839407906,-0.000299945740016,-0.000303022347223,-
0.000306069991571,-0.000309088319858,-0.00031207723721,-0.00031503695312,-
0.000317967379455,-0.000320866295073,-0.000323734372043,-0.000326572199814,-
0.000329378390265,-0.000332153455092,-0.000334897825511,-0.000337611506849,-
0.000340294580545,-0.000342946978939,-0.000345568670157,-0.000348160650292,-
0.000350723807153,-0.00035325805617,-0.000355763303023,-0.000358239219414,-
0.000360685615276,-0.000363103504829,-0.000365492854065,-0.000367855081982,-
0.000370190363613,-0.000372498614042,-0.000374780481701,-0.000377035956262,-
0.000379265353461,-0.000381469283899,-0.000383648214088,-0.000385801861381,-
0.000387930118986,-0.000390033421296,-0.000392111603687,-0.000394165721033,-
0.000396195246035,-0.000398199769836,-0.000400179972575,-0.000402135393686,-
0.000404065841833,-0.00040597218495,-0.00040785407292,-0.000409711168901,-
0.000411542864587,-0.000413347840208,-0.000415125869219,-0.000416877803648,-
0.000418603313719,-0.000420302813241,-0.000421977671331,-0.000423628294924,-
0.000425254378681,-0.000426856782452,-0.000428436029203,-0.000429992812079,-
0.000431528083361,-0.000433040734752,-0.000434530742743,-0.000435998943239,-
0.00043744412465,-0.000438865281537,-0.000440262759506,-0.000441636167408,-
0.000442986021576,-0.000444312747933,-0.000445615953018,-0.000446896810902,-
0.000448157276585,-0.000449397133271,-0.000450615915969,-0.000451814203491,-
0.000452992839784,-0.00045415108108,-0.000455289019701,-0.000456407001581,-
0.000457504611396,-0.000458582510136,-0.000459640094393,-0.000460675951085,-
0.000461690605587,-0.000462685246695,-0.000463659823403,-0.000464615578096,-
0.000465552953871,-0.000466472404495,-0.00046737347989,-0.000468256649025,-
0.000469121426916,-0.000469968324963,-0.000470798394081,-0.000471610866473,-
0.000472405827432,-0.000473182378117,-0.000473938940477,-0.000474675612164,-
0.000475393472074,-0.000476091762913,-0.000476770410693,-0.000477430026166,-
0.000478070533346,-0.00047869287298,-0.00047929778605,-0.000479884977566,-
0.000480456354221,-0.00048101198266,-0.000481551872081,-0.000482075256972,-
0.00048258336378,-0.000483076662643,-0.000483555168099,-0.00048401882647,-
0.000484466540028,-0.000484898875862,-0.000485316026024,-0.000485717381786,-
0.00048610373478,-0.000486476055367,-0.000486834457237,-0.000487178680423,-
0.000487509427515,-0.000487826336868,-0.000488129779175,-0.000488420855529,-
0.000488698595461,-0.000488964500406,-0.000489218944099,-0.000489461945316,-
0.00048969374102,-0.000489914914747,-0.000490125482988,-0.000490325225123,-
0.000490515390117,-0.000490696280154,-0.00049086808553,-0.000491031075244,-
0.000491184316483,-0.000491328077233,-0.000491462289597,-0.000491587037362,-
0.000491701692288,-0.000491806998683,-0.000491902345425,-0.000491987029018,-
0.000492061162575,-0.000492124454656,-0.000492177512595,-0.000492220401613,-
0.000492252880688,-0.000492274770971,-0.000492285621437,-0.000492285284048,-
0.000492273579519,-0.0004922506185,-0.000492216399544,-0.000492171071753,-
0.000492114830272,-0.000492048253123,-0.000491970816345,-0.000491883432706,-
0.000491786957872,-0.000491682199743,-0.000491569961266,-0.000491449954591,-
0.000491322852751,-0.000491188427261,-0.000491045889812,-0.000490894109223,-
0.000490732890471,-0.000490562311124,-0.000490381226875,-0.000490189012515,-
0.000489984725591,-0.000489768048546,-0.000489538850776,-0.000489296017414,-
0.000489040016453,-0.000488771669308,-0.000488491806093,-0.000488199705208,-
0.000487895164331,-0.00048757868544,-0.000487250400146,-0.000486910885628,-
0.000486560109473,-0.000486198570127,-0.00048582654074,-0.000485443638281,-
0.000485049644072,-0.000484644614538,-0.000484229280664,-0.000483803838998,-
```

```
0.000483367828493,-0.000482920539983,-0.000482461711519,-0.000481991173753,-
0.000481507983736,-0.000481012156008,-0.000480503346699,-0.000479980267554,-
0.000479442445347,-0.000478889223349,-0.000478320357938,-0.000477735657868,-
0.000477135220879,-0.000476519148341,-0.000475888024575,-0.000475242704607,-
0.000474583067435,-0.000473909706915,-0.000473223131003,-0.000472523028065,-
0.000471810024079,-0.000471084378708,-0.000470345899373,-0.000469594576733,-
0.000468829985265,-0.000468051025524,-0.000467257033865,-0.000466447667184,-
0.000465622230424,-0.000464780578831,-0.000463922074872,-0.000463045865099,-
0.000462151300605,-0.000461238191706,-0.00046030561868,-0.000459352643183,-
0.000458378625063,-0.000457383505826,-0.000456367289828,-0.000455329565523,-
0.000454269916488,-0.000453189095035,-0.000452087467356,-0.000450965532446,-
0.000449824173491,-0.000448664020084,-0.000447486244829,-0.000446292121384,-
0.000445081690864,-0.00044385562242,-0.000442614311571,-0.000441358326551,-
0.000440087099542,-0.000438800476179,-0.000437497957685,-0.000436179766931,-
0.000434846388577,-0.000433497061354,-0.00043213159755,-0.000430749811321,-
0.000429352039658,-0.00042793881219,-0.000426510153335,-0.000425066005629,-
0.000423606826299,-0.000422132952137,-0.000420643503027,-0.000419138143441,-
0.000417617835003,-0.000416082282127,-0.00041453146822,-0.000412964039989,-
0.000411380297295,-0.000409780333754,-0.000408164427604,-0.000406532266403,-
0.00040488393392,-0.00040322037018,-0.00040154140876,-0.000399846589921,-
0.000398136642448,-0.000396411894509,-0.000394673181058,-0.000392920553404,-
0.000391154406082,-0.000389375012989,-0.0003875826992,-0.000385776735038,-
0.000383956756598,-0.000382124009229,-0.000380278685756,-0.000378420629088,-
0.000376549539337,-0.000374664721049,-0.000372765227793,-0.000370850938449,-
0.000368921272473,-0.000366976805498,-0.000365017914467,-0.000363044065559,-
0.000361054186869,-0.000359048511991,-0.000357027248551,-0.000354991217736,-
0.000352941153108,-0.000350877611206,-0.000348800827699,-0.000346711161741,-
0.000344608437894,-0.00034249322946,-0.000340366037774,-0.000338226955303,-
0.000336075472752,-0.000333911363413,-0.00033173454439,-0.000329544471115,-
0.000327341102962,-0.000325124226005,-0.000322893798514,-0.000320649496227,-
0.000318390690293,-0.000316117557998,-0.000313830781049,-0.000311530050245,-
0.000309215729482,-0.000306887994546,-0.000304547110962,-0.000302192575837,-
0.000299824151716,-0.000297441662797,-0.000295045487579,-0.000292636061262,-
0.000290212845404,-0.000287775664292,-0.00028532451703,-0.000282858721601,-
0.000280378131893,-0.000277883728632,-0.000275376584695,-0.000272856771499,-
0.000270324345272,-0.000267778458518,-0.000265218849772,-0.000262645827692,-
0.0002600595103,-0.000257460026747,-0.000254847548569,-0.00025222063524,-
0.000249578200352,-0.000246920025799,-0.000244246628431,-0.000241558857747,-
0.000238857240293,-0.000236142354027,-0.000233413530605,-0.000230670546962,-
0.000227913647594,-0.000225143917932,-0.000222362042579,-0.000219568500022,-
0.000216762423533,-0.000213943226106,-0.000211110364335,-0.00020826280282,-
0.000205400334841,-0.000202523346876,-0.00019963177997,-0.000196724703975,-
0.000193801100896,-0.00019086104185,-0.000187905078274,-0.000184933307509,-
0.000181946020848,-0.000178943665602,-0.000175926548872,-0.000172894609573,-
0.000169847531606,-0.000166785234345,-0.000163708381595,-0.00016061665625,-
0.000157509272019,-0.000154385269874,-0.000151244427101,-0.000148085988219,-
0.000144909301056,-0.00014171381835,-0.000138499136311,-0.000135265915279,-
0.000132013952625,-0.000128743172038,-0.000125453525474,-0.000122145582534,-
0.000118819385814,-0.00011547558706,-0.000112114336326,-0.000108736461315,-
0.000105343150743,-0.000101934333842,-9.85099263334e-05,-9.50698526408e-05,-
9.16141943411e-05,-8.81427774234e-05,-8.46555145662e-05,-8.11522595607e-05,-
7.7633076901e-05,-7.4097362111e-05,-7.05439921853e-05,-6.69724496835e-05,-
6.33827202966e-05,-5.97745196588e-05,-5.61478273026e-05,-5.25024193161e-05,-
4.88383382627e-05,-4.51559337079e-05,-4.14553147401e-05,-3.77362350883e-05,-
3.3999041838e-05,-3.02437691074e-05,-2.64700392033e-05,-2.26777702334e-05,-
1.88669538973e-05,-1.50383900496e-05,-1.11925523103e-05,-7.32936095102e-06,-
3.44826377583e-06,4.50820295249e-07,4.36803969875e-06,8.30335380856e-
06,1.22565133224e-05,1.62274467752e-05,2.02164759667e-05,2.42242804483e-
05,2.82520510701e-05,3.23002943651e-05,3.63685730082e-05,4.04562296861e-
```

```
05,4.45629439629e-05,4.86884604971e-05,5.28321250854e-05,5.69943945353e-
05,6.11744514047e-05,6.5371510177e-05,6.95843957469e-05,7.38131700815e-
05,7.80576120511e-05,8.23179765839e-05,8.65946592276e-05,9.08879780885e-
05,9.51983909781e-05,9.95257457193e-
05,0.00010386982936,0.000108231595499,0.000112611736126,0.00011700983962,0.00012142
528095,0.00012585813334,0.000130307823283,0.000134774394876,0.000139257291618,0.000
143755850023,0.000148270109471,0.00015279963551,0.000157343386493,0.000161901172912
,0.000166473451887,0.000171060182757,0.000175661141754,0.000180275935052,0.00018490
4238338,0.000189546181068,0.000194201955577,0.000198871472912,0.000203554869151,0.0
00208252245549,0.000212963372177,0.000
217688283027,0.000222427294045,0.000227180030524,0.000231946705957,0.00023672747794
,0.000241521968759,0.000246329528082,0.00025114989897,0.00025598224607,0.0002608264
44029,0.000265681737942,0.00027054752919,0.000275422645353,0.000280307471832,0.0002
85201354198,0.000290103924273,0.000295015292358,0.000299935643215,0.00030486516038,
0.000309803864284,0.000314750849061,0.000319705810906,0.00032466882808,0.0003296397
80125,0.00033461754062,0.00033960147874,0.00034459050125,0.000349583600433,0.000354
579947418,0.000359578431186,0.000364578783445,0.00036958113218,0.00037458581907,0.0
00379591500368,0.0003845974571,0.000389603593676,0.000394609588181,0.00039961500904
4,0.00040461987893,0.000409623820913,0.000414626814198,0.000419627836755,0.00042462
5640437,0.000429619202849,0.000434608977439,0.000439594800115,0.000444576369731,0.0
0044955333596,0.000454524787139,0.000459489489179,0.000464446526922,0.0004693948617
12,0.000474333968639,0.000479263850844,0.000484183831245,0.000489093341691,0.000493
991951509,0.000498879779829,0.000503757889194,0.000508633345845,0.000513515678803,0
.000518414281705,0.000523338180177,0.000528322713953,0.000533170980271,0.0005379125
68455,0.000542554031027,0.000547102200938,0.000551562518687,0.000555938001519,0.000
5602204753,0.000564399175483,0.000568463253194,0.000572401556582,0.000576176963742,
0.000579984547097,0.000583792158777,0.00058759090176,0.000591372733678,0.0005951299
83014,0.00059885768027,0.000602556223033,0.00060622664317,0.000609869973966,0.00061
3488218708,0.000617082111251,0.000620653266787,0.000624203679318,0.000627734532093,
0.000631245902263,0.000634737555993,0.000638208955428,0.000641659612127,0.000645088
581624,0.000648494719833,0.000651876502066,0.000655233287947,0.00065856304102,0.000
661864171141,0.00066513519652,0.00066837460783,0.000671581529151,0.000674754661455,
0.000677892932131,0.000680994348557,0.000684057550474,0.000687081012043,0.000690062
998006,0.000693002130289,0.000695897119463,0.000698747095046,0.000701551094991,0.00
0704307481498,0.000707015140804,0.000709672939419,0.00071228058862,0.00071483765132
5,0.000717343382705,0.000719797035702,0.000722198243949,0.000724546179645,0.0007268
40058404,0.000729079214913,0.000731263011209,0.000733390784899,0.000735461652272,0.
000737474477212,0.000739428268639,0.000741322281712,0.000743155548726,0.00074492713
0553,0.000746635656835,0.000748280563399,0.000749860852957,0.000751375989933,0.0007
52825506667,0.000754208297686,0.00075552375019,0.000756771619059,0.000757951596076,
0.000759063119284,0.000760105851062,0.000761079486712,0.00076198344003,0.0007628172
4203,0.00076358059016,0.000764273100359,0.000764895553097,0.000765447919271,0.00076
5930106485,0.00076634242007,0.000766685252149,0.000766958758292,0.000767163703758,0
.000767300392916,0.000767369394237,0.000767370907307,0.0007673053337,0.000767172527
115,0.000766972244896,0.000766704456081,0.000766369103426,0.000765967092414,0.00076
5498419779,0.000764963281575,0.000764361786481,0.000763694848979,0.000762962565218,
0.000762165008627,0.000761303123205,0.000760378087537,0.000759390693885,0.000758341
167617,0.000757230007557,0.000756058431106,0.000754826905478,0.00075353610737,0.000
752186148388,0.000750778421822,0.000749313813774,0.000747792611612,0.00074621558896
7,0.000744582784501,0.000742895239354,0.000741153358699,0.000739357593486,0.0007375
08649197,0.000735607670689,0.000733655561231,0.000731653161677,0.000729601332637,0.
000727500973885,0.000725353269475,0.000723159525853,0.000720920589883,0.00071863792
0904,0.000716312727244,0.000713946103246,0.000711538227686,0.000709089449223,0.0007
06600407316,0.00070407174025,0.000701504188005,0.000698898057821,0.000696254138035,
0.000693572872304,0.000690854448983,0.000688099388656,0.000685308540679,0.000682483
047681,0.000679623867684,0.000676731512869,0.000673807006576,0.000670851082681,0.00
0667864367514,0.000664847976293,0.00066180252208,0.00065872874968,0.000655627889501
,0.000652500857237,0.000649348378263,0.00064617145078,0.000642970816148,0.000639746
972558,0.00063650054936,0.000633232401987,0.000629943143745,0.000626633309239,0.000
```

```
623303034298,0.000619952326804,0.00061658145361,0.000613191137319,0.000609781791234
,0.000606353875496,0.000602907790502,0.000599443761058,0.000595961956389,0.00059246
3090644,0.000588948442378,0.000585418303767,0.000581873423816,0.000578314134678,0.0
00574740321898,0.000571152637497,0.000567551910509,0.00056393860039,0.0005603136094
16,0.000556677350781,0.000553029686355,0.000549371230282,0.000545702745094,0.000542
024747385,0.000538338119639,0.000534643543922,0.000530941375295,0.000527231487438,0
.000523514558868,0.000519790794942,0.000516060734368,0.000512325103962,0.0005085842
51574,0.000504838839808,0.000501088956885,0.000497335073909,0.000493578078906,0.000
489818665572,0.000486057535364,0.000482295123395,0.00047853185829,0.000474768403444
,0.000471004910352,0.000467241376462,0.000463478645205,0.000459716866683,0.00045595
5603059,0.000452195023897,0.000448435013992,0.000444676573015,0.000440920137467,0.0
00437166019313,0.000433414064521,0.000429664581861,0.000425917741824,0.000422173636
239,0.000418432777269,0.000414696009974,0.00041096374459,0.000407236133979,0.000403
512701393,0.000399793645353,0.000396079610901,0.000392370958131,0.00038866790852,0.
000384970606998,0.000381279035223,0.000377593430957,0.000373914088328,0.00037024068
9704,0.000366573902165,0.000362913255896,0.000359258822831,0.000355609984389,0.0003
51967014892,0.000348330293096,0.000344700234889,0.000341076999405,0.000337460334797
,0.000333849967484,0.00033024586925,0.000326647965248,0.000323056566203,0.000319472
261973,0.000315895712529,0.000312326594575,0.000308764539856,0.000305208913717,0.00
0301659910462,0.000298118022403,0.000294583435564,0.000291056428989,0.0002875367534
75,0.000284024664998,0.000280519155022,0.000277020025268,0.000273527986227,0.000270
043222551,0.000266566116939,0.000263096625207,0.000259634476683,0.000256179348247,0
.000252731027838,0.00024928994206,0.000245856294601,0.000242430126743,0.00023901075
4718,0.000235597444572,0.000232189833404,0.000228787558682,0.000225389836623,0.0002
21996543573,0.000218607913845,0.000215223481214,0.000211842501468,0.000208464935004
,0.000205090715694,0.000201719990178,0.000198352796558,0.000194989016589,0.00019162
9262827,0.000188274010633,0.000184922669583,0.000181575041082,0.000178231160157,0.0
00174891543886,0.000171555865091,0.000168224219145,0.000164896879804,0.000161574009
584,0.000158255307725,0.000154939859357,0.00015162766031,0.000148318330459,0.000145
011593631,0.000141706815581,0.000138403399364,0.000135101501928,0.00013180042615,0.
000128499615757,0.00012519814764,0.000121896248347,0.000118594492218,0.000115292964
927,0.000111992493465,0.000108693330834,0.000105396676316,0.000102102523944,9.88107
829774e-05,9.55220829608e-05,9.22375567085e-05,8.8958411267e-05,8.56847837739e-
05,8.24165621316e-05,7.91533766306e-05,7.58951774703e-05,7.2641600974e-
05,6.93922508823e-05,6.61474193396e-05,6.29064316251e-05,5.96692832139e-
05,5.64350086676e-05,5.32030594218e-05,4.99730023432e-05,4.6744927379e-
05,4.35186451282e-05,4.02935951906e-05,3.70703653651e-05,3.38485513634e-
05,3.06285546529e-05,2.74108610178e-05,2.41954809573e-05,2.09827021433e-
05,1.77734370949e-05,1.45680672598e-05,1.13674360227e-05,8.17276834316e-
06,4.98451222703e-06,1.80366264895e-06,-1.36888936829e-06,-4.53311987486e-06,-
7.68865733222e-06,-1.0834775869e-05,-1.39714259745e-05,-1.70986202549e-05,-
2.0216385221e-05,-2.33257860973e-05,-2.64271169168e-05,-2.95202107264e-05,-
3.26063068545e-05,-3.56853645043e-05,-3.87579807359e-05,-4.18243750102e-05,-
4.48847126823e-05,-4.79388714777e-05,-5.09868102673e-05,-5.40277998684e-05,-
5.70610640956e-05,-6.00871776288e-05,-6.3105301895e-05,-6.6115110251e-05,-
6.91158347309e-05,-7.21066960743e-05,-7.50871929718e-05,-7.80571000769e-05,-
8.10163076494e-05,-8.39643312527e-05,-8.69014231737e-05,-8.9827319104e-05,-
9.2741425233e-05,-9.56435977108e-05,-9.85334007006e-05,-0.000101411150778,-
0.000104276293219,-0.000107128365424,-0.000109967499179,-0.000112793363916,-
0.000115606138356,-0.000118405532019,-0.000121191184414,-0.000123963424044,-
0.00012672233966,-0.000129467603734,-0.000132199750374,-0.000134919403024,-
0.000137626475515,-0.000140320779715,-0.000143002110763,-0.000145670409171,-
0.000148325882381,-0.000150968744254,-0.00015359851381,-0.000156215627018,-
0.000158820529995,-0.000161412253485,-0.000163990624013,-0.000166555976441,-
0.000169108099385,-0.000171646865197,-0.000174172576204,-0.000176684710285,-
0.000179183288396,-0.000181668524297,-0.000184139817662,-0.000186597664269,-
0.000189042235719,-0.000191473060518,-0.000193890051983,-0.000196293707816,-
0.000198684041742,-0.000201060993831,-0.000203425083058,-0.000205776279234,-
0.000208114067594,-0.000210438558279,-0.000212748991656,-0.000215046204335,-
```

```
0.000217330686594,-0.000219601720897,-0.000221859112309,-0.000224102745049,-
0.00022633203136,-0.000228546636071,-0.000230746692478,-0.000232931645821,-
0.000235101752087,-0.000237257255085,-0.000239397520488,-0.000241522600005,-
0.000243632225321,-0.000245726574694,-0.000247805886182,-0.000249870559419,-
0.000251920707198,-0.000253957008277,-0.000255979931179,-0.000257988674886,-
0.000259982247156,-0.000261961496375,-0.000263926461892,-0.000265877033076,-
0.000267812909812,-0.000269733824448,-0.000271639380752,-0.000273529147485,-
0.000275402160017,-0.00027725833491,-0.000279098897948,-0.000280923278541,-
0.000282730897594,-0.000284521798298,-0.000286296207458,-0.000288053527924,-
0.000289794104252,-0.000291517953397,-0.000293225599383,-0.000294917194414,-
0.000296592875365,-0.000298252537623,-0.000299896890421,-0.000301526021727,-
0.000303140186224,-0.000304740254649,-0.000306326453917,-0.000307898836861,-
0.000309457416183,-0.000311001873617,-0.000312532310928,-0.000314048705909,-
0.00031555071188,-0.000317038386894,-0.000318511082929,-0.000319968017887,-
0.000321408714191,-0.0003228330272,-0.000324240547675,-0.000325631537411,-
0.00032700556411,-0.000328362129806,-0.000329701553667,-0.000331023726524,-
0.0003323292704,-0.000333618835265,-0.00033
4892592401,-0.000336151015966,-0.000337394537954,-0.000338622766446,-
0.000339835575249,-0.000341033503214,-0.000342216864376,-0.000343385064308,-
0.00034453716613,-0.000345672019905,-0.00034678851395,-0.000347885600204,-
0.000348963264417,-0.000350021445278,-0.000351060077277,-0.000352078826884,-
0.0003530765826,-0.000354053121132,-0.000355009423397,-0.000355946147971,-
0.00035686497888,-0.000357766795986,-0.000358651773253,-0.000359519433877,-
0.000360369921791,-0.000361203482911,-0.000362020713567,-0.000362822317876,-
0.000363608299548,-0.000364377375949,-0.000365129180843,-0.000365862945559,-
0.000366578005007,-0.000367274945478,-0.000367953920409,-0.000368614012702,-
0.000369255832058,-0.000369878924395,-0.000370482673995,-0.000371067395274,-
0.000371633228083,-0.000372180065564,-0.000372708111523,-0.000373217625075,-
0.000373708241473,-0.00037417939681,-0.00037463134897,-0.000375063736395,-
0.000375476625139,-0.000375870479021,-0.000376244784344,-0.000376599159678,-
0.000376933471929,-0.000377246955106,-0.000377538952307,-0.00037780973575,-
0.000378057865243,-0.000378282783262,-0.000378483095209,-0.000378659127671,-
0.000378810317534,-0.00037893682168,-0.000379038732756,-0.000379116213991,-
0.000379168425843,-0.000379195873051,-0.00037919997127,-0.000379181536388,-
0.000379141899453,-0.000379080575169,-0.000378996789142,-0.000378890458092,-
0.000378761208526,-0.000378608551962,-0.000378432993948,-0.000378234811585,-
0.00037801205056,-0.000377763644335,-0.000377489248578,-0.00037718843362,-
0.000376862449313,-0.000376510728079,-0.000376133486504,-0.000375731138113,-
0.000375304564671,-0.000374853590238,-0.000374378643085,-0.000373881625029,-
0.000373362312563,-0.000372821835207,-0.000372259084889,-0.000371675077969,-
0.000371070191425,-0.000370444098653,-0.000369796938461,-0.000369128252725,-
0.000368438505709,-0.000367726932195,-0.000366994206871,-0.000366239866375,-
0.000365464288949,-0.000364667982796,-0.00036385000257,-0.000363011075692,-
0.000362151279327,-0.000361270973361,-0.000360371193233,-0.000359452212391,-
0.000358514289011,-0.000357557080893,-0.000356582842572,-0.000355590665603,-
0.000354580743163,-0.000353654669256,-0.000352913597313,-0.000352358119686,-
0.000351987020781,-0.000351801716806)
xcorr <- c(-600.0,-598.999165972,-597.998331943,-596.997497915,-595.996663887,-
594.995829858,-593.99499583,-592.994161802,-591.993327773,-590.992493745,-
589.991659716,-588.990825688,-587.98999166,-586.989157631,-585.988323603,-
584.987489575,-583.986655546,-582.985821518,-581.98498749,-580.984153461,-
579.983319433,-578.982485405,-577.981651376,-576.980817348,-575.979983319,-
574.979149291,-573.978315263,-572.977481234,-571.976647206,-570.975813178,-
569.974979149,-568.974145121,-567.973311093,-566.972477064,-565.971643036,-
564.970809008,-563.969974979,-562.969140951,-561.968306922,-560.967472894,-
559.966638866,-558.965804837,-557.964970809,-556.964136781,-555.963302752,-
554.962468724,-553.961634696,-552.960800667,-551.959966639,-550.959132611,-
549.958298582,-548.957464554,-547.956630525,-546.955796497,-545.954962469,-
544.95412844,-543.953294412,-542.952460384,-541.951626355,-540.950792327,-
```

```
539.949958299,-538.94912427,-537.948290242,-536.947456214,-535.946622185,-
534.945788157,-533.944954128,-532.9441201,-531.943286072,-530.942452043,-
529.941618015,-528.940783987,-527.939949958,-526.93911593,-525.938281902,-
524.937447873,-523.936613845,-522.935779817,-521.934945788,-520.93411176,-
519.933277731,-518.932443703,-517.931609675,-516.930775646,-515.929941618,-
514.92910759,-513.928273561,-512.927439533,-511.926605505,-510.925771476,-
509.924937448,-508.92410342,-507.923269391,-506.922435363,-505.921601334,-
504.920767306,-503.919933278,-502.919099249,-501.918265221,-500.917431193,-
499.916597164,-498.915763136,-497.914929108,-496.914095079,-495.913261051,-
494.912427023,-493.911592994,-492.910758966,-491.909924937,-490.909090909,-
489.908256881,-488.907422852,-487.906588824,-486.905754796,-485.904920767,-
484.904086739,-483.903252711,-482.902418682,-481.901584654,-480.900750626,-
479.899916597,-478.899082569,-477.89824854,-476.897414512,-475.896580484,-
474.895746455,-473.894912427,-472.894078399,-471.89324437,-470.892410342,-
469.891576314,-468.890742285,-467.889908257,-466.889074229,-465.8882402,-
464.887406172,-463.886572143,-462.885738115,-461.884904087,-460.884070058,-
459.88323603,-458.882402002,-457.881567973,-456.880733945,-455.879899917,-
454.879065888,-453.87823186,-452.877397832,-451.876563803,-450.875729775,-
449.874895746,-448.874061718,-447.87322769,-446.872393661,-445.871559633,-
444.870725605,-443.869891576,-442.869057548,-441.86822352,-440.867389491,-
439.866555463,-438.865721435,-437.864887406,-436.864053378,-435.863219349,-
434.862385321,-433.861551293,-432.860717264,-431.859883236,-430.859049208,-
429.858215179,-428.857381151,-427.856547123,-426.855713094,-425.854879066,-
424.854045038,-423.853211009,-422.852376981,-421.851542952,-420.850708924,-
419.849874896,-418.849040867,-417.848206839,-416.847372811,-415.846538782,-
414.845704754,-413.844870726,-412.844036697,-411.843202669,-410.842368641,-
409.841534612,-408.840700584,-407.839866555,-406.839032527,-405.838198499,-
404.83736447,-403.836530442,-402.835696414,-401.834862385,-400.834028357,-
399.833194329,-398.8323603,-397.831526272,-396.830692244,-395.829858215,-
394.829024187,-393.828190158,-392.82735613,-391.826522102,-390.825688073,-
389.824854045,-388.824020017,-387.823185988,-386.82235196,-385.821517932,-
384.820683903,-383.819849875,-382.819015847,-381.818181818,-380.81734779,-
379.816513761,-378.815679733,-377.814845705,-376.814011676,-375.813177648,-
374.81234362,-373.811509591,-372.810675563,-371.809841535,-370.809007506,-
369.808173478,-368.80733945,-367.806505421,-366.805671393,-365.804837364,-
364.804003336,-363.803169308,-362.802335279,-361.801501251,-360.800667223,-
359.799833194,-358.798999166,-357.798165138,-356.797331109,-355.796497081,-
354.795663053,-353.794829024,-352.793994996,-351.793160967,-350.792326939,-
349.791492911,-348.790658882,-347.789824854,-346.788990826,-345.788156797,-
344.787322769,-343.786488741,-342.785654712,-341.784820684,-340.783986656,-
339.783152627,-338.782318599,-337.78148457,-336.780650542,-335.779816514,-
334.778982485,-333.778148457,-332.777314429,-331.7764804,-330.775646372,-
329.774812344,-328.773978315,-327.773144287,-326.772310259,-325.77147623,-
324.770642202,-323.769808173,-322.768974145,-321.768140117,-320.767306088,-
319.76647206,-318.765638032,-317.764804003,-316.763969975,-315.763135947,-
314.762301918,-313.76146789,-312.760633862,-311.759799833,-310.758965805,-
309.758131776,-308.757297748,-307.75646372,-306.755629691,-305.754795663,-
304.753961635,-303.753127606,-302.752293578,-301.75145955,-300.750625521,-
299.749791493,-298.748957465,-297.748123436,-296.747289408,-295.746455379,-
294.745621351,-293.744787323,-292.743953294,-291.743119266,-290.742285238,-
289.741451209,-288.740617181,-287.739783153,-286.738949124,-285.738115096,-
284.737281068,-283.736447039,-282.735613011,-281.734778982,-280.733944954,-
279.733110926,-278.732276897,-277.731442869,-276.730608841,-275.729774812,-
274.728940784,-273.728106756,-272.727272727,-271.726438699,-270.725604671,-
269.724770642,-268.723936614,-267.723102585,-266.722268557,-265.721434529,-
264.7206005,-263.719766472,-262.718932444,-261.718098415,-260.717264387,-
259.716430359,-258.71559633,-257.714762302,-256.713928274,-255.713094245,-
254.712260217,-253.711426188,-252.71059216,-251.709758132,-250.708924103,-
249.708090075,-248.707256047,-247.706422018,-246.70558799,-245.704753962,-
```

```
244.703919933,-243.703085905,-242.702251877,-241.701417848,-240.70058382,-
239.699749791,-238.698915763,-237.698081735,-236.697247706,-235.696413678,-
234.69557965,-233.694745621,-232.693911593,-231.693077565,-230.692243536,-
229.691409508,-228.69057548,-227.689741451,-226.688907423,-225.688073394,-
224.687239366,-223.686405338,-222.685571309,-221.684737281,-220.683903253,-
219.683069224,-218.682235196,-217.681401168,-216.680567139,-215.679733111,-
214.678899083,-213.678065054,-212.677231026,-211.676396997,-210.675562969,-
209.674728941,-208.673894912,-207.673060884,-206.672226856,-205.671392827,-
204.670558799,-203.669724771,-202.668890742,-201.668056714,-200.667222686,-
199.666388657,-198.665554629,-197.664720601,-196.663886572,-195.663052544,-
194.662218515,-193.661384487,-192.660550459,-191.65971643,-190.658882402,-
189.658048374,-188.657214345,-187.656380317,-186.655546289,-185.65471226,-
184.653878232,-183.653044204,-182.652210175,-181.651376147,-180.650542118,-
179.64970809,-178.648874062,-177.648040033,-176.647206005,-175.646371977,-
174.645537948,-173.64470392,-172.643869892,-171.643035863,-170.642201835,-
169.641367807,-168.640533778,-167.63969975,-166.638865721,-165.638031693,-
164.637197665,-163.636363636,-162.635529608,-161.63469558,-160.633861551,-
159.633027523,-158.632193495,-157.631359466,-156.630525438,-155.62969141,-
154.628857381,-153.628023353,-152.627189324,-151.626355296,-150.625521268,-
149.624687239,-148.623853211,-147.623019183,-146.622185154,-145.621351126,-
144.620517098,-143.619683069,-142.618849041,-141.618015013,-140.617180984,-
139.616346956,-138.615512927,-137.614678899,-136.613844871,-135.613010842,-
134.612176814,-133.611342786,-132.610508757,-131.609674729,-130.608840701,-
129.608006672,-128.607172644,-127.606338616,-126.605504587,-125.604670559,-
124.60383653,-123.603002502,-122.602168474,-121.601334445,-120.600500417,-
119.599666389,-118.59883236,-117.597998332,-116.597164304,-115.596330275,-
114.595496247,-113.594662219,-112.59382819,-111.592994162,-110.592160133,-
109.591326105,-108.590492077,-107.589658048,-106.58882402,-105.587989992,-
104.587155963,-103.586321935,-102.585487907,-101.584653878,-100.58381985,-
99.5829858215,-98.5821517932,-97.5813177648,-96.5804837364,-95.5796497081,-
94.5788156797,-93.5779816514,-92.577147623,-91.5763135947,-90.5754795663,-
89.5746455379,-88.5738115096,-87.5729774812,-86.5721434529,-85.5713094245,-
84.5704753962,-83.5696413678,-82.5688073394,-81.5679733111,-80.5671392827,-
79.5663052544,-78.565471226,-77.5646371977,-76.5638031693,-75.562969141,-
74.5621351126,-73.5613010842,-72.5604670559,-71.5596330275,-70.5587989992,-
69.5579649708,-68.5571309425,-67.5562969141,-66.5554628857,-65.5546288574,-
64.553794829,-63.5529608007,-62.5521267723,-61.551292744,-60.5504587156,-
59.5496246872,-58.5487906589,-57.5479566305,-56.5471226022,-55.5462885738,-
54.5454545455,-53.5446205171,-52.5437864887,-51.5429524604,-50.542118432,-
49.5412844037,-48.5404503753,-47.539616347,-46.5387823186,-45.5379482902,-
44.5371142619,-43.5362802335,-42.5354462052,-41.5346121768,-40.5337781485,-
39.5329441201,-38.5321100917,-37.5312760634,-36.530442035,-35.5296080067,-
34.5287739783,-33.52793995,-32.5271059216,-31.5262718932,-30.5254378649,-
29.5246038365,-28.5237698082,-27.5229357798,-26.5221017515,-25.5212677231,-
24.5204336947,-23.5195996664,-22.518765638,-21.5179316097,-20.5170975813,-
19.516263553,-18.5154295246,-17.5145954962,-16.5137614679,-15.5129274395,-
14.5120934112,-13.5112593828,-12.5104253545,-11.5095913261,-10.5087572977,-
9.50792326939,-8.50708924103,-7.50625521268,-6.50542118432,-5.50458715596,-
4.50375312761,-3.50291909925,-2.50208507089,-1.50125104254,-
0.500417014179,0.500417014178,1.50125104254,2.50208507089,3.50291909925,4.503753127
61,5.50458715596,6.50542118432,7.50625521268,8.50708924103,9.50792326939,10.5087572
977,11.5095913261,12.5104253545,13.5112593828,14.5120934112,15.5129274395,16.513761
4679,17.5145954962,18.5154295246,19.516263553,20.5170975813,21.5179316097,22.518765
638,23.5195996664,24.5204336947,25.5212677231,26.5221017515,27.5229357798,28.523769
8082,29.5246038365,30.5254378649,31.5262718932,32.5271059216,33.52793995,34.5287739
783,35.5296080067,36.530442035,37.5312760634,38.5321100917,39.5329441201,40.5337781
485,41.5346121768,42.5354462052,43.5362802335,44.5371142619,45.5379482902,46.538782
3186,47.539616347,48.5404503753,49.5412844037,50.542118432,51.5429524604,52.5437864
887,53.5446205171,54.5454545455,55.5462885738,56.5471226022,57.5479566305,58.548790
```

```
6589,59.5496246872,60.5504587156,61.551292744,62.5521267723,63.5529608007,64.553794
829,65.5546288574,66.5554628857,67.5562969141,68.5571309425,69.5579649708,70.558798
9992,71.5596330275,72.5604670559,73.5613010842,74.5621351126,75.562969141,76.5638
031693,77.5646371977,78.565471226,79.5663052544,80.5671392827,81.5679733111,82.5688
073394,83.5696413678,84.5704753962,85.5713094245,86.5721434529,87.5729774812,88.573
8115096,89.5746455379,90.5754795663,91.5763135947,92.577147623,93.5779816514,94.578
8156797,95.5796497081,96.5804837364,97.5813177648,98.5821517932,99.5829858215,100.5
8381985,101.584653878,102.585487907,103.586321935,104.587155963,105.587989992,106.5
8882402,107.589658048,108.590492077,109.591326105,110.592160133,111.592994162,112.5
9382819,113.594662219,114.595496247,115.596330275,116.597164304,117.597998332,118.5
9883236,119.599666389,120.600500417,121.601334445,122.602168474,123.603002502,124.6
0383653,125.604670559,126.605504587,127.606338616,128.607172644,129.608006672,130.6
08840701,131.609674729,132.610508757,133.611342786,134.612176814,135.613010842,136.
613844871,137.614678899,138.615512927,139.616346956,140.617180984,141.618015013,142
.618849041,143.619683069,144.620517098,145.621351126,146.622185154,147.623019183,14
8.623853211,149.624687239,150.625521268,151.626355296,152.627189324,153.628023353,1
54.628857381,155.62969141,156.630525438,157.631359466,158.632193495,159.633027523,1
60.633861551,161.63469558,162.635529608,163.636363636,164.637197665,165.638031693,1
66.638865721,167.63969975,168.640533778,169.641367807,170.642201835,171.643035863,1
72.643869892,173.64470392,174.645537948,175.646371977,176.647206005,177.648040033,1
78.648874062,179.64970809,180.650542118,181.651376147,182.652210175,183.653044204,1
84.653878232,185.65471226,186.655546289,187.656380317,188.657214345,189.658048374,1
90.658882402,191.65971643,192.660550459,193.661384487,194.662218515,195.663052544,1
96.663886572,197.664720601,198.665554629,199.666388657,200.667222686,201.668056714,
202.668890742,203.669724771,204.670558799,205.671392827,206.672226856,207.673060884
,208.673894912,209.674728941,210.675562969,211.676396997,212.677231026,213.67806505
4,214.678899083,215.679733111,216.680567139,217.681401168,218.682235196,219.6830692
24,220.683903253,221.684737281,222.685571309,223.686405338,224.687239366,225.688073
394,226.688907423,227.689741451,228.69057548,229.691409508,230.692243536,231.693077
565,232.693911593,233.694745621,234.69557965,235.696413678,236.697247706,237.698081
735,238.698915763,239.699749791,240.70058382,241.701417848,242.702251877,243.703085
905,244.703919933,245.704753962,246.70558799,247.706422018,248.707256047,249.708090
075,250.708924103,251.709758132,252.71059216,253.711426188,254.712260217,255.713094
245,256.713928274,257.714762302,258.71559633,259.716430359,260.717264387,261.718098
415,262.718932444,263.719766472,264.7206005,265.721434529,266.722268557,267.7231025
85,268.723936614,269.724770642,270.725604671,271.726438699,272.727272727,273.728106
756,274.728940784,275.729774812,276.730608841,277.731442869,278.732276897,279.73311
0926,280.733944954,281.734778982,282.735613011,283.736447039,284.737281068,285.7381
15096,286.738949124,287.739783153,288.740617181,289.741451209,290.742285238,291.743
119266,292.743953294,293.744787323,294.745621351,295.746455379,296.747289408,297.74
8123436,298.748957465,299.749791493,300.750625521,301.75145955,302.752293578,303.75
3127606,304.753961635,305.754795663,306.755629691,307.75646372,308.757297748,309.75
8131776,310.758965805,311.759799833,312.760633862,313.76146789,314.762301918,315.76
3135947,316.763969975,317.764804003,318.765638032,319.76647206,320.767306088,321.76
8140117,322.768974145,323.769808173,324.770642202,325.77147623,326.772310259,327.77
3144287,328.773978315,329.774812344,330.775646372,331.7764804,332.777314429,333.778
148457,334.778982485,335.779816514,336.780650542,337.78148457,338.782318599,339.783
152627,340.783986656,341.784820684,342.785654712,343.786488741,344.787322769,345.78
8156797,346.788990826,347.789824854,348.790658882,349.791492911,350.792326939,351.7
93160967,352.793994996,353.794829024,354.795663053,355.796497081,356.797331109,357.
798165138,358.798999166,359.799833194,360.800667223,361.801501251,362.802335279,363
.803169308,364.804003336,365.804837364,366.805671393,367.806505421,368.80733945,369
.808173478,370.809007506,371.809841535,372.810675563,373.811509591,374.81234362,375
.813177648,376.814011676,377.814845705,378.815679733,379.816513761,380.81734779,381
.818181818,382.819015847,383.819849875,384.820683903,385.821517932,386.82235196,387
.823185988,388.824020017,389.824854045,390.825688073,391.826522102,392.82735613,393
.828190158,394.829024187,395.829858215,396.830692244,397.831526272,398.8323603,399.
833194329,400.834028357,401.834862385,402.835696414,403.836530442,404.83736447,405.
838198499,406.839032527,407.839866555,408.840700584,409.841534612,410.842368641,411
```

```
.843202669,412.844036697,413.844870726,414.845704754,415.846538782,416.847372811,41
7.848206839,418.849040867,419.849874896,420.850708924,421.851542952,422.852376981,4
23.853211009,424.854045038,425.854879066,426.855713094,427.856547123,428.857381151,
429.858215179,430.859049208,431.859883236,432.860717264,433.861551293,434.862385321
,435.863219349,436.864053378,437.864887406,438.865721435,439.866555463,440.86738949
1,441.86822352,442.869057548,443.869891576,444.870725605,445.871559633,446.87239366
1,447.87322769,448.874061718,449.874895746,450.875729775,451.876563803,452.87739783
2,453.87823186,454.879065888,455.879899917,456.880733945,457.881567973,458.88240200
2,459.88323603,460.884070058,461.884904087,462.885738115,463.886572143,464.88740617
2,465.8882402,466.889074229,467.889908257,468.890742285,469.891576314,470.892410342
,471.89324437,472.894078399,473.894912427,474.895746455,475.896580484,476.897414512
,477.89824854,478.899082569,479.899916597,480.900750626,481.901584654,482.902418682
,483.903252711,484.904086739,485.904920767,486.905754796,487.906588824,488.90742285
2,489.908256881,490.909090909,491.909924937,492.910758966,493.911592994,494.9124270
23,495.913261051,496.914095079,497.914929108,498.915763136,499.916597164,500.917431
193,501.918265221,502.919099249,503.919933278,504.920767306,505.921601334,506.92243
5363,507.923269391,508.92410342,509.924937448,510.925771476,511.926605505,512.92743
9533,513.928273561,514.92910759,515.929941618,516.930775646,517.931609675,518.93244
3703,519.933277731,520.93411176,521.934945788,522.935779817,523.936613845,524.93744
7873,525.938281902,526.93911593,527.939949958,528.940783987,529.941618015,530.94245
2043,531.943286072,532.9441201,533.944954128,534.945788157,535.946622185,536.947456
214,537.948290242,538.94912427,539.949958299,540.950792327,541.951626355,542.952460
384,543.953294412,544.95412844,545.954962469,546.955796497,547.956630525,548.957464
554,549.958298582,550.959132611,551.959966639,552.960800667,553.961634696,554.96246
8724,555.963302752,556.964136781,557.964970809,558.965804837,559.966638866,560.9674
72894,561.968306922,562.969140951,563.969974979,564.970809008,565.971643036,566.972
477064,567.973311093,568.974145121,569.974979149,570.975813178,571.976647206,572.97
7481234,573.978315263,574.979149291,575.979983319,576.980817348,577.981651376,578.9
82485405,579.983319433,580.984153461,581.98498749,582.985821518,583.986655546,584.9
87489575,585.988323603,586.989157631,587.98999166,588.990825688,589.991659716,590.9
92493745,591.993327773,592.994161802,593.99499583,594.995829858,595.996663887,596.9
97497915,597.998331943,598.999165972,600.0)
altd  <- c(87)
x <- seq.int((length(p)-1)/2*-1,(length(p)-1)/2)
pdf('hpne_sorted_mdup_model.pdf',height=6,width=6)
plot(x,p,type='l',col=c('red'),main='Peak Model',xlab='Distance to the 
middle',ylab='Percentage')
lines(x,m,col=c('blue'))
legend('topleft',c('forward tags','reverse tags'),lty=c(1,1,1),col=c('red','blue'))
plot(xcorr,ycorr,type='l',col=c('black'),main='Cross-Correlation',xlab='Lag between 
+ and - tags',ylab='Correlation')
abline(v=altd,lty=2,col=c('red'))
legend('topleft','alternative lag(s)',lty=2,col='red')
legend('right','alt lag(s) : 87',bty='n')
dev.off()

==> ./sanchari/j455/runMacs.sh <==
#!/bin/bash
#$ -cwd
#$ -N macs2
#$ -j n
#$ -l h_vmem=9.8G
#$ -pe smp 1
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

#run mac2 peak finding software on a single target bam
#usually we need a control file, igg or input
```

```
#
#usage:
#qsub runMacs.sh bamfile.bam

BAM=$1
NAME=${BAM%.*}
FOL=macs_${NAME}

#module load MACS2/2.1.0/python.2.7.8
module load MACS2/2.1.0-update/python.2.7.8

echo $BAM
echo $NAME
echo $FOL

macs2 callpeak -f BAMPE -t $BAM -n $NAME --outdir $FOL --verbose 3
#macs2 callpeak  -t $BAM -n $NAME --outdir $FOL --call-summits --verbose 3

==> ./sanchari/oxbs_atac/prepRandomSites.r <==
#read in a bed file
options(stringsAsFactors=F)

#sites = read.table("sites_hpne_all.bed", header=F)
sites = read.table("commonSites1_hpne.bed", header=F)

dim(sites)
head(sites)

#for each chrom
sites.all = lapply(unique(sites[,1]), function(chrom){
    ix = sites[,1] == chrom
    sites[ix,]
})
names(sites.all) = unique(sites[,1])
length(sites.all)

#for each chromosome write random sites between min and max
for (chrom in unique(sites[,1])){
    print(chrom)
    my.sites = sites.all[[chrom]]

    r1 = min(my.sites[,2])
    r2 = max(my.sites[,2])

    #range(min/max)
    rmy.sites = my.sites
    rmy.sites[,2] = sample(r1:r2, nrow(rmy.sites))
    rmy.sites[,3] = rmy.sites[,2]+1


    write.table(file="sites_rand_all.bed", x=rmy.sites, row.names=F, col.names=F,
quote=F, sep="\t", append=T)
}




#replace coordinates with random numbers in the same
r1 = min(sites.19[,2])
```

```
r2 = max(sites.19[,2])

#range(min/max)
rsites.19 = sites.19
rsites.19[,2] = sample(r1:r2, nrow(rsites.19))
rsites.19[,3] = rsites.19[,2]+1

#write new bed file
write.table(file="sites_rand_chr19.bed", x=rsites.19, row.names=F, col.names=F,
quote=F, sep="\t")



#prep sites that appear in one sample but not another.

sites.hpne = read.table("sites_hpne_all.bed", header=F)
sites.jd = read.table("sites_jd_all.bed", header=F)

sapply(paste0("chr",c(1:22,"X","Y")), function(chrom){
      in1 = (sites.hpne[sites.hpne[,1] == chrom,2] %in% sites.jd[sites.jd[,1] ==
chrom,2])
})
```

```
==> ./sanchari/oxbs_atac/plotPeakProfiles.py <==
import numpy as np
import matplotlib.pyplot as plt
import glob
import os
import sys
import HTSeq as ht
import numpy.random as rd
import multiprocessing as mp

#we want to count(and eventually plot) the stat3 peaks coinciding
#with variuos region sets
#promoter regions (TSS +/- 2kb)
#genebody regions, split by exons and introns
#pancreatic regions, peak bed files specified in the ENCODE databases



#step1.  get the regions of interest
#1a.  promoters are from the annotation table file

PROM_DIST=2000
#read in the annotation file.  this was downloaded from the ucsc genome browswer
anno = np.loadtxt("/home/kpradhan/Desktop/data/hg19/anno/annotation_hg19.gtf",
dtype=object)
anno[:5]
#regions lists will all be the same format
dt_sites = np.dtype([("chrom", "S4"), ("strand", "S1"), ("start", "int"), ("stop",
"int")])
x = [(str(a[1]), str(a[2]), int(a[3])-PROM_DIST, int(a[3])+PROM_DIST) for a in
anno]
sites_tss = np.unique(np.array(x, dtype=dt_sites))
sites_tss.shape
```

```
#1b.  can get exons/introns from the same annotaion file
#start and stop show the gene's full length
#exonStarts and exonStops will give the locations of the exons
#everything else between start and stop will be the introns
def getExonsIntrons(start, stop, exonStarts, exonStops):
    start = int(start)
    stop = int(stop)
    #split by comma
    exonStarts = map(int, exonStarts.rstrip(",").split(","))
    exonStops = map(int, exonStops.rstrip(",").split(","))
    #find all the exons
    exons = []
    for a, b in zip(exonStarts, exonStops):
        exons.append([a, b])
    #find the spots in betweent the ejdons
    introns = []
    #begin at the tss, not at the first exon
    curPos = start
    for exon in exons:
        if exon[0] - curPos > 0:
            introns.append([curPos, exon[0]])
        curPos = exon[1]
    #remember to check the last position
    if stop - curPos > 0:
        introns.append([curPos, stop])
    return exons, introns



#get a list of all the exons and introns
sites_exons = []
sites_introns = []
for a in anno:
    #get the boundaries of each exon and intron
    exons, introns = getExonsIntrons(a[3], a[4], a[8], a[9])
    #add the chromosome and strand info to the list
    for e in exons:
        sites_exons.append((str(a[1]), str(a[2]), e[0], e[1]))
    for i in introns:
        sites_introns.append((str(a[1]), str(a[2]), i[0], i[1]))
#turn them into numpy record arrays
sites_exons = np.unique(np.array(sites_exons, dtype=dt_sites))
sites_introns = np.unique(np.array(sites_introns, dtype=dt_sites))
sites_exons.shape
sites_introns.shape


#1c.  load in the 6 bed files downloaded from encode
#   #H3K27ac #H3K27me3 #H3K36me3 #H3K4me1 #H3K4me3 #H3K9me3
pancFolder = "/home/kpradhan/Desktop/data/pancreas"
pancFiles = glob.glob(pancFolder+"/*.bed.gz")

pfile= pancFiles[0]
def getPeakSitesFromBed(pfile):
    peaks = np.loadtxt(pfile, dtype=object)
    #regions lists will all be the same format
    dt_sites = np.dtype([("chrom", "S5"), ("strand", "S1"), ("start", "int"),
("stop", "int")])
```

```
    x = [(str(a[1]), str(a[2]), int(a[3])-PROM_DIST, int(a[3])+PROM_DIST) for a in
peaks]
    sites_tss = np.unique(np.array(x, dtype=dt_sites))


#pfile = "peaks_jd13d_stat3Input.txt"
def getPeakSitesFromMacs(pfile):
    peaks = np.loadtxt(pfile, dtype=object)
    #regions lists will all be the same format
    dt_sites = np.dtype([("chrom", "S5"), ("strand", "S1"), ("start", "int"),
("stop", "int")])
    x = []
    for a in peaks:
        #mid = (int(a[2]) + int(a[1])) / 2
        #x = [(str(a[0]), "+", mid-200, mid+200) for a in peaks]
        x.append( (str(a[0]), "+", int(a[1]), int(a[2])) )
    #only retain proper chromosomes
    x = np.unique(np.array(x, dtype=dt_sites))
    ix = np.array([a[:3] == "chr" or a == "X" or a == "Y" for a in x["chrom"]] )
    return x[ix]


#maybe only look at the peaks that pass a threshold
def getPeakMidsFromMacs(pfile, thresh=0):
    peaks = np.loadtxt(pfile, dtype=object)
    #regions lists will all be the same format
    dt_sites = np.dtype([("chrom", "S5"), ("strand", "S1"), ("site", "int")])
    x = []
    for a in peaks:
        if thresh == 0 or float(a[5]) > thresh:
            mid = (int(a[2]) + int(a[1])) / 2
            x.append((str(a[0]), "+", mid))
        #x.append( (str(a[0]), "+", int(a[1]), int(a[2])) )
    #only retain proper chromosomes
    x = np.unique(np.array(x, dtype=dt_sites))
    ix = np.array([a[:3] == "chr" or a == "X" or a == "Y" for a in x["chrom"]] )
    return x[ix]

#step2.  get the read count profiles
#using the stat3 peaks
pfile = "peaks_jd13d_stat3Input.txt"
peaks_jd13d = getPeakSitesFromMacs(pfile)
peakmids_jd13d = getPeakMidsFromMacs(pfile)


peaks_jd13d = getPeakSitesFromMacs("peaks_jd13d_stat3Input.txt")
peaks_hpne = getPeakSitesFromMacs("peaks_hpne_stat3Input.txt")
peaks_jd13d.shape
peaks_hpne.shape


#count the number of peaks that fall in any of the regions
peaks = peaks_jd13d
sites = sites_tss
p = peaks[0]
s = sites[0]
peaks['chrom']
sites['chrom']
```

```
#peaks: X
#sites: chrX
def countOverlap(peaks, sites):
    count = 0
    #work chromosome by chrom
    for chrom in np.unique(peaks["chrom"]):
        ix1 = peaks["chrom"] == chrom
        ix2 = (sites['chrom'] == chrom) + (sites['chrom'] == "chr"+chrom)
        #does the peak region intersect any of the sites?
        A = peaks[ix1]["start"]
        B = peaks[ix1]["stop"]
        C = sites[ix2]["start"]
        D = sites[ix2]["stop"]
        for (a, b) in zip(A, B):
            count = count + np.sum((a <= D) * (C <= b))
    return count

#find the number of peaks that lay within boundaries of
#   promoter regions
#   gene exon regions
#   gene intron regions
sites_tss.shape
sites_exons.shape
sites_introns.shape
countOverlap(peaks_jd13d, sites_tss)
countOverlap(peaks_jd13d, sites_exons)
countOverlap(peaks_jd13d, sites_introns)
countOverlap(peaks_hpne, sites_tss)
countOverlap(peaks_hpne, sites_exons)
countOverlap(peaks_hpne, sites_introns)

#for every site find the distribution of reads in
#the bam file that overlap the region centered at site
#assume the sites datastructure has a chr* name
#the bamfile might have chroms that look like "chr12" or "12"
bam = bams[0]
sites = peakmids_jd13d

class _exhausted():
    pass

def truncChr(c):
    if c.isdigit():
        return c
    elif c == "X" or c == "Y":
        return c
    else:
        return c[3:]

def getProfile(pos, bam, halfwinwidth=3000, noChr=False):
    peakProfile = np.zeros(2*halfwinwidth)
    sitechr = truncChr(pos["chrom"]) if noChr else pos["chrom"]
    window = ht.GenomicInterval( str(sitechr), pos["site"] - halfwinwidth,
pos["site"] + halfwinwidth, str(pos["strand"]) )
    if (list(bam[window])):
        for almnt in bam[window]:
            if pos["strand"] == "+":
                a = almnt.iv.start - pos["site"] + halfwinwidth
                b = almnt.iv.end - pos["site"] + halfwinwidth
            if pos["strand"] == "-":
                a = pos["site"] + halfwinwidth - almnt.iv.end
                b = pos["site"] + halfwinwidth - almnt.iv.start
            peakProfile[a:b] += 1
    if (np.sum(peakProfile) > 0):
        return peakProfile
    else:
        return []

def getPeakProfiles_mt(bamfile, sites, halfwinwidth=3000, noChr = False):
    bam = ht.BAM_Reader(bamfile)
    #make sure the sites and bam files have same naming convention
    #
    #retrict sites to those that have an entry in the bam file
    bamChroms = [x["SN"] for x in bam.get_header_dict()["SQ"]]
    bamChroms = ["chr"+c if c.isdigit() else c for c in bamChroms]
    sites = sites[np.in1d(sites["chrom"], bamChroms)]
    sites.shape
    #
    peakProfs = []
    #collect the sites as genomic intervals
    pool = mp.Pool(processes=7)
    out = [pool.apply_async(getProfile, args=(pos, bam, halfwinwidth, noChr)) for
pos in sites]
    peakProfs = [o.get() for o in out]
    #ignore the profiles with no reads
    peakProfs = [p for p in peakProfs if len(p) > 0]
    return np.array(peakProfs)



bamfile = "/home/kpradhan/Desktop/data/pancreas/test1/test3.bam"
def getPeakProfiles(bamfile, sites, halfwinwidth=3000, noChr = False):
    bam = ht.BAM_Reader(bamfile)
    #make sure the sites and bam files have same naming convention
    #
    #retrict sites to those that have an entry in the bam file
    bamChroms = [x["SN"] for x in bam.get_header_dict()["SQ"]]
    bamChroms = ["chr"+c if c.isdigit() else c for c in bamChroms]
    sites = sites[np.in1d(sites["chrom"], bamChroms)]
    sites.shape
    #
    peakProfs = []
    #collect the sites as genomic intervals
    for i, pos in enumerate(sites):
        peakProfile = np.zeros(2*halfwinwidth)
        if i % 1000 == 0:
            print "%d of %d" %(i, len(sites))
        sitechr = truncChr(pos["chrom"]) if noChr else pos["chrom"]
        window = ht.GenomicInterval( str(sitechr), pos["site"] - halfwinwidth,
pos["site"] + halfwinwidth, str(pos["strand"]) )
        if (list(bam[window])):
            for almnt in bam[window]:
                if pos["strand"] == "+":
                    a = almnt.iv.start - pos["site"] + halfwinwidth
                    b = almnt.iv.end - pos["site"] + halfwinwidth
                if pos["strand"] == "-":
                    a = pos["site"] + halfwinwidth - almnt.iv.end
                    b = pos["site"] + halfwinwidth - almnt.iv.start
                peakProfile[a:b] += 1
        if (np.sum(peakProfile) > 0):
            peakProfs.append(peakProfile)
    return np.array(peakProfs)

def binPeakProfiles(peakProfs, K=100):
    sums = [np.sum(p) for p in peakProfs]
    ix = np.argsort(sums)
    #
    #bin every 100 rows together
    smallX = []
    for i in range(0, len(peakProfs)-1, K):
        smallX.append(np.sum(np.array([peakProfs[j] for j in ix[i:(i+K)]]),0))
    return np.array(smallX)[::-1]

peakmids_jd13d = getPeakMidsFromMacs(pfile)
sites=peakmids_jd13d
def makePeakPlots(sites, bamfile, outfile, noChr=False):
    peakProfs = getPeakProfiles(bamfile, sites, 3000, noChr)
    #
    prof = np.sum(peakProfs, 0)
    smallPeaks = binPeakProfiles(peakProfs, K=100)
    x = smallPeaks.ravel()
    vmax = np.sort(x)[np.round(len(x)*.80)]
    #
    plt.close()
    fig = plt.figure()
    fig.suptitle(outfile)
    ax = plt.subplot(121)
    xran = np.arange(len(prof))-3000
    plt.plot(xran[500:(-500)], prof[500:(-500)])
    ax = plt.subplot(122)
    ax.set_xticklabels([])
    ax.set_yticklabels([])
    plt.imshow(smallPeaks[:, 500:(-500),],  aspect = "auto", vmax=vmax)
    #plt.colorbar(orientation="horizontal")
    plt.colorbar(orientation="vertical")
    plt.savefig(outfile)
    plt.close()


def clipFilename(ffile):
    return ffile[(ffile.rfind("/")+1):]



pfile = "peaks_jd13d_stat3Input.txt"
pancBamFolder = "/home/kpradhan/Desktop/data/pancreas/sorted/hacked"
sites_jd13d = getPeakMidsFromMacs(pfile)
sites_jd13d_q3 = getPeakMidsFromMacs(pfile, 3)
sites_jd13d_q3.shape

bams = glob.glob(pancBamFolder+"/*.bam")
for bamfile in bams:
    outfile = "jd13dStat3Peaks_q3_"+clipFilename(bamfile)+".png"
    makePeakPlots_mt(sites_jd13d_q3, bamfile, outfile)


bams = glob.glob(pancBamFolder+"/*.bam")
```

```
for bamfile in bams:
    outfile = "jd13dStat3Peaks_q3_"+clipFilename(bamfile)+".png"
    makePeakPlots(sites_jd13d, bamfile, outfile)

rsites = getRandomPeaks(sites_jd13d)
bams = glob.glob(pancBamFolder+"/*.bam")
for bamfile in bams:
    outfile = "randomPeaks_"+clipFilename(bamfile)+".png"
    makePeakPlots(rsites, bamfile, outfile)



def getRandomPeaks(sites):
    #get max sizes for each chromosome
    chroms = np.unique(sites["chrom"])
    nSites = {}
    maxSize = {}
    minSize = {}
    for chrom in chroms:
        nSites[chrom] = np.sum(sites["chrom"] == chrom)
        maxSize[chrom] = np.max(sites[sites["chrom"] == chrom]["site"])
        minSize[chrom] = np.min(sites[sites["chrom"] == chrom]["site"])
    #randomly sample peaks.
    rsites = sites.copy()
    for chrom in chroms:
        s = np.sort(rd.random_integers(minSize[chrom], maxSize[chrom],
size=nSites[chrom]))
        for i, ix in enumerate(np.where(rsites["chrom"] == chrom)[0]):
            rsites[ix]["site"] = s[i]
    return rsites


#######################################
#get the peak call from the h3k encode beds
pfile =
"/media/kpradhan/DATAPART1/data/pancreas/encodePeaks/pancreas/pancreas_distal.H3K4m
e1.bed"
sites_d = getPeakMidsFromMacs(pfile)
pfile =
"/media/kpradhan/DATAPART1/data/pancreas/encodePeaks/pancreas/pancreas_proximal.H3K
4me1.bed"
sites_p = getPeakMidsFromMacs(pfile)
sites.shape
bfile =
"/home/kpradhan/Desktop/data/sanchari/pancreatic/stat3_chipseq/J179/bwa/JD13D-
STAT3.bwa.sorted.mkdup.bam"


pfiles =
glob.glob("/media/kpradhan/DATAPART1/data/pancreas/encodePeaks/pancreas/*.bed")
bfiles =
glob.glob("/home/kpradhan/Desktop/data/sanchari/pancreatic/stat3_chipseq/J179/bwa/*
.bam")
for bfile in bfiles:
    for pfile in pfiles:
        sites = getPeakMidsFromMacs(pfile)
        outfile = "colocPlot_"+clipFilename(pfile)+"_"+clipFilename(bfile)+".png"
        makePeakPlots(sites, bfile, outfile, noChr=True)

#do the merged prox dist peaks
```

```
for bfile in bfiles:
    for i in range(3):
        sites = np.hstack((getPeakMidsFromMacs(pfiles[i]),
getPeakMidsFromMacs(pfiles[i+3])))
        outfile = "colocPlot_"+clipFilename(pfiles[i])+clipFilename(pfiles[i+3])
+"_"+clipFilename(bfile)+".png"
        makePeakPlots(sites, bfile, outfile, noChr=True)

peakProfs_p = getPeakProfiles(bfile, sites_p, 3000, noChr=True)
peakProfs_d = getPeakProfiles(bfile, sites_d, 3000, noChr=True)
prof = np.sum(peakProfs, 0)
profd = np.sum(peakProfs_d, 0)
plt.plot(prof, label="prox")
plt.plot(profd, label="distal")
plt.legend()





#################################################3
bamfile1 =
"/home/kpradhan/Desktop/data/sanchari/pancreatic/stat3_chipseq/J179/bwa/jd13stat3_q
10.bam"
bamfile2 =
"/home/kpradhan/Desktop/data/sanchari/pancreatic/stat3_chipseq/J179/bwa/JD13D-
STAT3.bwa.sorted.mkdup.bam"
bamfile2 =
"/home/kpradhan/Desktop/data/sanchari/pancreatic/stat3_chipseq/J179/bwa/HPNE-
Input.bwa.sorted.mkdup.bam"

sites_jd13d_p001 = getPeakMidsFromMacs(pfile, 3)
sites_jd13d = getPeakMidsFromMacs(pfile)
sites_jd13d_p001.shape
sites_jd13d.shape
peakProfs1 = getPeakProfiles(bamfile1, sites_jd13d, 3000, noChr=True)
peakProfs2 = getPeakProfiles(bamfile2, sites_jd13d, 3000, noChr=True)
prof1 = np.sum(peakProfs1, 0)
prof2 = np.sum(peakProfs2, 0)
plt.plot(prof1, label="stat3 q10")
plt.plot(prof2, label="stat3")
plt.legend()
#input has a vally
#stat3 has a mountain


bamfile =
"/home/kpradhan/Desktop/data/pancreas/sorted/hacked/hacked_sorted_GSM906419_UCSD.Pa
ncreas.Input.STL003.bam"
peakProfs = getPeakProfiles_mt(bamfile, sites_jd13d_q3, 3000)

bamfile =
"/home/kpradhan/Desktop/data/pancreas/sorted/hacked/hacked_sorted_GSM906419_UCSD.Pa
ncreas.Input.STL003.bam"
bamfile = bams[0]
peakProfs = getPeakProfiles(bamfile, sites_jd13d, 3000)
rProfs = getPeakProfiles(bamfile, rsites, 3000)

prof = np.sum(peakProfs, 0)
```

```
rprof = np.sum(rProfs, 0)
smallPeaks = binPeakProfiles(peakProfs, K=100)
rsmallPeaks = binPeakProfiles(rProfs, K=100)

plt.plot(prof)

plt.hist(smallPeaks.ravel(), 1000)
x = smallPeaks.ravel()
rx = rsmallPeaks.ravel()
vmax = np.sort(x)[np.round(len(x)*.80)]
rvmax = np.sort(rx)[np.round(len(rx)*.80)]


    plt.close()
    fig = plt.figure()
    #fig.suptitle("\n".join(map(clipFilename, [hmcfile, bamfile])))
    ax = plt.subplot(121)
    xran = np.arange(len(prof))-3000
    plt.plot(xran[500:(-500)], rprof[500:(-500)])
    #plt.plot(xran[500:(-500)], prof[500:(-500)])
    ax = plt.subplot(122)
    ax.set_xticklabels([])
    ax.set_yticklabels([])
    plt.imshow(rsmallPeaks[:, 500:(-500),],  aspect = "auto", vmax=rvmax)
    #plt.imshow(smallPeaks[:, 500:(-500),],  aspect = "auto", vmax=vmax)
    #plt.colorbar(orientation="horizontal")
    plt.colorbar(orientation="vertical")
    #plt.savefig(outfile)
    plt.close()


np
plt.imshow(smallPeaks[:, 500:(-500),],  aspect = "auto", vmax=vmax)
plt.plot(prof[500:(-500)])
plt.plot(prof)

#is it possible to load the panc bed files as a bam or wig?
wig =
ht.WiggleReader("/home/kpradhan/Desktop/data/pancreas/GSM910576_UCSD.Pancreas.H3K4m
e1.STL003.wig.gz")
window = ht.GenomicInterval( "chr1", 10000, 20000000, "+" )
list(wig[window])
        if (list(bam[window])):
            for almnt in bam[window]:




==> ./sanchari/oxbs_atac/commands1.sh <==
ngsplotdb.py list

tss, tes, genebody, exon, cgi, enhancer, dhs
testbam=/home/kpradhan/Desktop/hpc_home/projects/sanchari/j455_j515/jd_sorted_mdup_
merged.bam
ngs.plot.r -G hg19 -R tss -C $testbam -O test1plot_tss -T testplot -L 3000
ngs.plot.r -G hg19 -R genebody -C $testbam -O test1plot_genebody -T testplot -L
3000
ngs.plot.r -G hg19 -R exon -C $testbam -O test1plot_exon -T testplot -L 3000
ngs.plot.r -G hg19 -R enhancer -C $testbam -O test1plot_enhancer -T testplot -L
```

```
3000
ngs.plot.r -G hg19 -R dhs -C $testbam -O test1plot_dhs -T testplot -L 3000

#make a test bed file
#chr start stop
ls ../../

#prepare the hmc site bed files for jd and hpne
tail -n +2 ../j432_j446/hmcSites/hmcSites_HPNE_all.txt | sed 's/"//g' | awk '{print
$1,"\t",$2,"\t",$2+1}' | sed 's/ //'g >| sites_hpne_all.bed
tail -n +2 ../j432_j446/hmcSites/hmcSites_JD_all.txt | sed 's/"//g' | awk '{print
$1,"\t",$2,"\t",$2+1}' | sed 's/ //'g >| sites_jd_all.bed

#prepare a list of random sites



#some smaller single chrom beds
tail -n +2 ../j432_j446/hmcSites/hmcSites_HPNE_10.txt | sed 's/"//g' | awk '{print
$1,"\t",$2-2000,"\t",$2+2000}' | sed 's/ //'g >| s10.bed
tail -n +2 ../j432_j446/hmcSites/hmcSites_HPNE_10.txt | sed 's/"//g' | awk '{print
$1,"\t",$2,"\t",$2+1}' | sed 's/ //'g >| s10.bed
time ngs.plot.r -G hg19 -R bed -E s10.bed -C $bam -O test2plot_bed -T testplot -P 6
-L 3000

#prepare bed files for chr19
tail -n +2 ../j432_j446/hmcSites/hmcSites_HPNE_all.txt | sed 's/"//g' | awk '{print
$1,"\t",$2,"\t",$2+1}' | sed 's/ //'g | grep -e "chr19" >| sites_hpne_19.bed
tail -n +2 ../j432_j446/hmcSites/hmcSites_JD_all.txt | sed 's/"//g' | awk '{print
$1,"\t",$2,"\t",$2+1}' | sed 's/ //'g | grep -e "chr19" >| sites_jd_19.bed


bam1=/home/kpradhan/Desktop/hpc_home/projects/sanchari/j455/hpne_sorted_mdup.bam
#bam2=/home/kpradhan/Desktop/hpc_home/projects/sanchari/j455/jd_sorted_mdup.bam
bam2=/home/kpradhan/Desktop/hpc_home/projects/sanchari/j455_j515/jd_sorted_mdup_mer
ged.bam
#time ngs.plot.r -G hg19 -R bed -E sites_hpne_19.bed -C $bam1 -O hpne_19 -T hpne_19
-P 6 -L 3000
#cat test1_jdbad.txt | awk '{printf("%s\t%d\t%d\t%s\n", $1, $2, $2+1, $3)}' >
test2_jdbad.txt
#cat test1_jdmc.txt | awk '{printf("%s\t%d\t%d\t%s\n", $1, $2, $2+1, $3)}' >
test2_jdmc.txt
#cat test1_jdmc_top10k.txt | awk '{printf("%s\t%d\t%d\t%s\n", $1, $2, $2+1, $3)}' >
test2_jdmc_top10k.txt
#cat test1_jdmc_top20k-10k.txt | awk '{printf("%s\t%d\t%d\t%s\n", $1, $2, $2+1,
$3)}' >| test2_jdmc_top20k-10k.txt
#cat test1_jdmc_top30k-10k.txt | awk '{printf("%s\t%d\t%d\t%s\n", $1, $2, $2+1,
$3)}' >| test2_jdmc_top30k-10k.txt
cat test1_jdmc_top60k-10k.txt | awk '{printf("%s\t%d\t%d\t%s\n", $1, $2, $2+1,
$3)}' >| test2_jdmc_top60k-10k.txt
cat test1_jdmc_top100k.txt | awk '{printf("%s\t%d\t%d\t%s\n", $1, $2, $2+1, $3)}'
>| test2_jdmc_top100k.txt
cut -f1-3 test2_jdmc_top30k-10k.txt > test3_jdmc_top30k-10k.txt
sed test2_jdmc_top30k-10k.txt -e 's/+/-/' > test4_jdmc_top30k-10k.txt
time ngs.plot.r -G hg19 -R bed -E sites_jd_19.bed -C $bam2 -O jd_19 -T jd_19 -P 6
-L 3000
time ngs.plot.r -G hg19 -R bed -E test2_jdhmc.txt -C $bam2 -O jd_test1 -T jd_test
-P 6 -L 3000
time ngs.plot.r -G hg19 -R bed -E test2_jdmc.txt -C $bam2 -O jd_mc_test1 -T
```

```
jd_mc_test -P 6 -L 3000
time ngs.plot.r -G hg19 -R bed -E test2_jdmc_top10k.txt -C $bam2 -O jdmc_top10k.txt
-T jdmc_top10k.txt -P 6 -L 3000
time ngs.plot.r -G hg19 -R bed -E test2_jdmc_top20k-10k.txt -C $bam2 -O top20k-10k
-T jdmc_top20k-10k.txt -P 6 -L 3000
time ngs.plot.r -G hg19 -R bed -E test3_jdmc_top30k-10k.txt -C $bam2 -O top30k-
10k_v2 -T jdmc_top30k-10k_v2 -P 6 -L 3000
time ngs.plot.r -G hg19 -R bed -E test4_jdmc_top30k-10k.txt -C $bam2 -O top30k-
10k_v4 -T jdmc_top30k-10k_v4 -P 6 -L 3000
time ngs.plot.r -G hg19 -R bed -E test2_jdmc_top60k-10k.txt -C $bam2 -O
jdmc_top60k-10k -T jdmc_top60k-10k -P 6 -L 3000
time ngs.plot.r -G hg19 -R bed -E test2_jdmc_top100k.txt -C $bam2 -O jdmc_100k -T
jdmc_100k -P 6 -L 3000
time ngs.plot.r -G hg19 -R bed -E test2_jdbad.txt -C $bam2 -O jdbad_test1 -T
jdbad_test -P 6 -L 3000

time ngs.plot.r -G hg19 -R bed -E sites_rand_chr19.bed -C config1.txt -O jd_rand19
-T jd_rand19 -P 6 -L 3000
time ngs.plot.r -G hg19 -R bed -C config2.txt -O jd_batch -T jd_batch -P 6 -L 3000

#prepare the hmcSite bed file from the common4 hmcSites.
#TODO:  messed up JD common sites bed..must redo
cat ../j432_j446/hmcSites/common4_hmcSites_JD.txt | cut -f1,2 | sed 's/#//' | \
    awk '{print $1,"\t",$2-1000,"\t",$2+1000}' | sed 's/ //'g >| commonSites_jd.bed
cat ../j432_j446/hmcSites/common4_hmcSites_HPNE.txt | cut -f1,2 | sed 's/#//' | \
    awk '{print $1,"\t",$2-1000,"\t",$2+1000}' | sed 's/ //'g >|
commonSites_hpne.bed


#dont' forget to take out spaces.
#ngsplot only works with tabs
cat ../j432_j446/hmcSites/common4_hmcSites_JD.txt | cut -f1,2 | sed 's/#//' | \
    awk '{print $1,"\t",$2,"\t",$2+1}' | sed 's/ //'g >| commonSites1_jd.bed
cat ../j432_j446/hmcSites/common4_hmcSites_HPNE.txt | cut -f1,2 | sed 's/#//' | \
    awk '{print $1,"\t",$2,"\t",$2+1}' | sed 's/ //'g >| commonSites1_hpne.bed



time ngs.plot.r -S 0.1 -G hg19 -R bed -E commonSites1_hpne.bed -C $bam1 -O hpne_all
-T hpne_all -P 2 -L 3000
#time ngs.plot.r -S 0.01 -G hg19 -R bed -E commonSites1_hpne.bed -C $bam2 -O
jdBam_hpneSites_all -T jdBam_hpneSites_0.01_all -P 6 -L 3000
time ngs.plot.r -S 0.1 -G hg19 -R bed -E commonSites1_jd.bed -C $bam2 -O jd_all -T
jd_all -P 2 -L 3000

time ngs.plot.r -S 0.01 -G hg19 -R bed -C config1.txt -O atacSeqBatch -T
atacSeqBatch -P 6 -L 3000



bam1=/home/kpradhan/Desktop/hpc_home/projects/sanchari/j455/hpne_sorted_mdup.bam
bam2=/home/kpradhan/Desktop/hpc_home/projects/sanchari/j455_j515/jd_sorted_mdup_mer
ged.bam
#try it with a more significant set of hmc values
cat ../j432_j446/hmcSites/common4_hmcSites_JD.txt | cut -f1,2,8 | sed 's/#//' | \
    awk '{if ($3 < 0.01) print $1,"\t",$2,"\t",$2+1}' | sed 's/ //'g >|
hmc_sig0.01_commonSites1_jd.bed
cat ../j432_j446/hmcSites/common4_hmcSites_HPNE.txt | cut -f1,2,8 | sed 's/#//' | \
    awk '{if ($3 < 0.01) print $1,"\t",$2,"\t",$2+1}' | sed 's/ //'g >|
hmc_sig0.01_commonSites1_hpne.bed

time ngs.plot.r  -G hg19 -R bed -E hmc_sig0.01_commonSites1_hpne.bed -C $bam1 -O
hmc_sig0.01_hpne_all -T hmc_sig0.01_hpne_all -P 2 -L 3000
time ngs.plot.r  -G hg19 -R bed -E hmc_sig0.01_commonSites1_jd.bed -C $bam2 -O
hmc_sig0.01_jd_all -T hmc_sig0.01_jd_all -P 2 -L 3000


#prepare bed files for the stat3 peaks
bam1=/home/kpradhan/Desktop/hpc_home/projects/sanchari/j455/hpne_sorted_mdup.bam
bam2=/home/kpradhan/Desktop/hpc_home/projects/sanchari/j455_j515/jd_sorted_mdup_mer
ged.bam

time ngs.plot.r  -G hg19 -R bed -E hpne_stat3_summits.bed -C $bam1 -O
hpne_stat3Peaks_atac -T hpne_stat3Peaks_atac -P 2 -L 3000
time ngs.plot.r  -G hg19 -R bed -E jd13d_stat3_summits.bed -C $bam2 -O
jd_stat3Peaks_atac -T jd_stat3Peaks_atac -P 2 -L 3000



#make plots for the MC sites
###############################################


#dont' forget to take out spaces.
#ngsplot only works with tabs
cat ../j432_j446/hpc_mcSites/common4_mcSites_JD.txt | cut -f1,2 | sed 's/#//' | \
    awk '{print $1,"\t",$2,"\t",$2+1}' | sed 's/ //'g >| mc_commonSites1_jd.bed
cat ../j432_j446/hpc_mcSites/common4_mcSites_HPNE.txt | cut -f1,2 | sed 's/#//' | \
    awk '{print $1,"\t",$2,"\t",$2+1}' | sed 's/ //'g >| mc_commonSites1_hpne.bed


bam1=/home/kpradhan/Desktop/hpc_home/projects/sanchari/j455/hpne_sorted_mdup.bam
bam2=/home/kpradhan/Desktop/hpc_home/projects/sanchari/j455_j515/jd_sorted_mdup_mer
ged.bam

time ngs.plot.r -S 0.05 -G hg19 -R bed -E mc_commonSites1_hpne.bed -C $bam1 -O
mc_hpne_0.05_all -T mc_hpne_0.05_all -P 2 -L 3000
time ngs.plot.r -S 0.05 -G hg19 -R bed -E mc_commonSites1_jd.bed -C $bam2 -O
mc_jd_0.05_all -T mc_jd_0.05_all -P 2 -L 3000

time ngs.plot.r -S 0.01 -G hg19 -R bed -E mc_commonSites1_jd.bed -C $bam2 -O
mc_jd_0.01_all -T mc_jd_0.01_all -P 3 -L 3000
time ngs.plot.r -S 0.01 -G hg19 -R bed -E mc_commonSites1_hpne.bed -C $bam1 -O
mc_hpne_0.01_all -T mc_hpne_0.01_all -P 3 -L 3000

time ngs.plot.r -S 0.1 -G hg19 -R bed -E commonSites1_jd.bed -C $bam2 -O
hmc_jd_0.1_all -T hmc_jd_0.1_all -P 3 -L 3000
time ngs.plot.r -S 0.1 -G hg19 -R bed -E commonSites1_hpne.bed -C $bam1 -O
hmc_hpne_0.1_all -T hmc_hpne_0.1_all -P 3 -L 3000




#try to do the same with RNA seq data

bam1=/home/kpradhan/Desktop/data/sanchari/pancreatic/rnaseq/bam/HPNE.AC1FL7ACXX.lan
e_1_P0_I5.hg19.gsnap-rna-alignment.bam
bam2=/home/kpradhan/Desktop/data/sanchari/pancreatic/rnaseq/bam/JRN.AD0C99ABXX.lane
```

```
_8_P0_I2.hg19.gsnap-rna-alignment.bam

samtools index $bam1
samtools index $bam2

time ngs.plot.r -S 0.01 -G hg19 -R bed -E mc_commonSites1_hpne.bed -C $bam1 -O
mc_rna_hpne_0.01_all -T mc_rna_hpne_0.01_all -P 2 -L 3000
time ngs.plot.r -S 0.01 -G hg19 -R bed -E mc_commonSites1_jd.bed -C $bam2 -O
mc_rna_jd_0.01_all -T mc_rna_jd_0.01_all -P 2 -L 3000
time ngs.plot.r -RR 300 -S 0.05 -G hg19 -R bed -E mc_commonSites1_hpne.bed -C $bam1
-O mc_rna_hpne_0.05_all -T mc_rna_hpne_0.05_all -P 2 -L 3000
time ngs.plot.r -RR 300 -S 0.05 -G hg19 -R bed -E mc_commonSites1_jd.bed -C $bam2
-O mc_rna_jd_0.05_all -T mc_rna_jd_0.05_all -P 2 -L 3000





#filter out mitichondria
samtools idxstats input.bam | cut -f 1 | grep -v MT | xargs samtools view -b
input.bam > output.bam

#get the numbe of good reads

#3844
#read unmapped
#not primary alignment
#read fails platform/vendor quality checks
#read is PCR or optical duplicate
#supplementary alignment
#64
#first in pair
samtools view -f 64 -F 3844 -c $bam1
samtools view -f 64 -F 3844 -c $bam1 chrM
samtools view -f 64 -F 3844 -c $bam2
samtools view -f 64 -F 3844 -c $bam2 chrM







==> ./sanchari/j446/clipOverlaps.sh <==
#!/bin/bash
#$ -cwd
#$ -j n
#$ -l h_vmem=4G
#$ -pe smp 1
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu
module load samtools/1.2/gcc.4.4.7
module load bamUtil/1.0.13/gcc.4.4.7

bam=$1
bname="${bam%.*}"

bam clipOverlap --in $bam --out ${bname}_clip.bam
```

```
samtools index ${bname}_clip.bam



==> ./sanchari/j446/processFiles1.sh <==
#step 1:  split each file into 40M line chunks
mkdir split
NL=40000000
for fq in `ls *.fq.gz`;
do
    nam=`basename $fq .gz`
    #get name without extension
    #echo $fq
    #echo $nam
    echo "split <(zcat $fq) -d -a 3 -l $NL split/${nam}.pt"
done >| split1.sh
chmod +x split1.sh


#step 2:  run bismark on the split files
#   and sort the resulting bam
mkdir bismark
cd bismark
for fq1 in `ls ../split/*1_val_1*`;
do
    fq2=`echo $fq1 | sed 's/\(.*\)1_val_1\(.*\)/\12_val_2\2/'`
    pname=`echo $fq1 | sed 's/.*\/\(.*\).1_val_1.fq.\(p.*\)/\1_\2/'`
    #echo $fq1
    #echo $fq2
    #echo $pname
    qsub ../runBismark.sh $pname $fq1 $fq2
done

#step 3.  merge the bams together
mkdir merged
cd merged
qsub ../mergeBams.sh J446_AHLT3HBCXX_Lane1_ACAGTG_HPNE-1-BS_merged_sorted.bam
../bismark/*HPNE-1-BS_p*_sorted.bam
qsub ../mergeBams.sh J446_AHLT3HBCXX_Lane2_GCCAAT_HPNE-1-
OxBS_merged_sorted.bam ../bismark/*HPNE-1-OxBS_p*_sorted.bam
qsub ../mergeBams.sh J446_BHLYGMBCXX_Lane1_CTTGTA_JD-BS_merged_sorted.bam
../bismark/*JD-BS_p*_sorted.bam
qsub ../mergeBams.sh J446_BHLYGMBCXX_Lane2_GTGAAA_JD-OxBS_merged_sorted.bam
../bismark/*JD-OxBS_p*_sorted.bam

#step 4.  mark duplicates
qsub ../markDuplicates.sh J446_AHLT3HBCXX_Lane1_ACAGTG_HPNE-1-BS_merged_sorted.bam
qsub ../markDuplicates.sh J446_AHLT3HBCXX_Lane2_GCCAAT_HPNE-1-
OxBS_merged_sorted.bam
qsub ../markDuplicates.sh J446_BHLYGMBCXX_Lane1_CTTGTA_JD-BS_merged_sorted.bam
qsub ../markDuplicates.sh J446_BHLYGMBCXX_Lane2_GTGAAA_JD-OxBS_merged_sorted.bam

#step 5.  clip overlap
qsub ../clipOverlaps.sh J446_AHLT3HBCXX_Lane1_ACAGTG_HPNE-1-
BS_merged_sorted_mdup.bam
qsub ../clipOverlaps.sh J446_AHLT3HBCXX_Lane2_GCCAAT_HPNE-1-
OxBS_merged_sorted_mdup.bam
qsub ../clipOverlaps.sh J446_BHLYGMBCXX_Lane1_CTTGTA_JD-BS_merged_sorted_mdup.bam
qsub ../clipOverlaps.sh J446_BHLYGMBCXX_Lane2_GTGAAA_JD-OxBS_merged_sorted_mdup.bam
```

```
#step 6.  methylation report generation
qsub ../reportMeth.sh J446_AHLT3HBCXX_Lane1_ACAGTG_HPNE-1-BS_merged_sorted_mdup_clip.bam
qsub ../reportMeth.sh J446_AHLT3HBCXX_Lane2_GCCAAT_HPNE-1-OxBS_merged_sorted_mdup_clip.bam
qsub ../reportMeth.sh J446_BHLYGMBCXX_Lane1_CTTGTA_JD-BS_merged_sorted_mdup_clip.bam
qsub ../reportMeth.sh J446_BHLYGMBCXX_Lane2_GTGAAA_JD-OxBS_merged_sorted_mdup_clip.bam




==> ./sanchari/j446/runBismark.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 3
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

#process a chunk of reads
#
#usage:
#qsub runBismark.sh hpneBs reads1.fastq.gz reads2.fastq.gz

SNAME=$1
FQ1=$2
FQ2=$3
REF=/home/kpradha1/projects/hg19/

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

bismark --multicore 1 --bowtie2 -p 2 $REF -1 $FQ1 -2 $FQ2
java -jar $(which SortSam.jar) INPUT=${FQ1##*/}_bismark_bt2_pe.bam OUTPUT=${SNAME}_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > $SNAME.status


==> ./sanchari/j446/markDuplicates.sh <==
#!/bin/bash
#$ -cwd
#$ -j n
#$ -l h_vmem=4G
#$ -pe smp 4
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

bam=$1
```

```
bname="${bam%.*}"

java -jar $(which BuildBamIndex.jar) INPUT=${bam}
java -jar $(which MarkDuplicates.jar) INPUT=${bam} OUTPUT=${bname}_mdup.bam
METRICS_FILE=${bname}_mdMetric.txt
java -jar $(which BuildBamIndex.jar) INPUT=${bname}_mdup.bam
```

```
==> ./sanchari/j446/dlfiles1.sh <==
wget http://waspsystem.einstein.yu.edu/wsfile/get/file/b3469e61-ce36-4849-b95c-
36c8376171bc/J446_AHLT3HBCXX_Lane1_ACAGTG_HPNE-1-BS.1_val_1.fq.gz
wget http://waspsystem.einstein.yu.edu/wsfile/get/file/5fa39bfa-b203-4978-8d64-
5161a327791f/J446_AHLT3HBCXX_Lane1_ACAGTG_HPNE-1-BS.2_val_2.fq.gz
wget http://waspsystem.einstein.yu.edu/wsfile/get/file/b8527954-c75d-4626-9677-
aea3eb499cfc/J446_AHLT3HBCXX_Lane2_GCCAAT_HPNE-1-OxBS.1_val_1.fq.gz
wget http://waspsystem.einstein.yu.edu/wsfile/get/file/bd34b329-4b0b-4147-b1f3-
e9f9834e6989/J446_AHLT3HBCXX_Lane2_GCCAAT_HPNE-1-OxBS.2_val_2.fq.gz
wget http://waspsystem.einstein.yu.edu/wsfile/get/file/a47b6a91-3f9b-4edd-a2bd-
e2756b2dcedc/J446_BHLYGMBCXX_Lane1_CTTGTA_JD-BS.1_val_1.fq.gz
wget http://waspsystem.einstein.yu.edu/wsfile/get/file/b5390735-036d-435a-b43c-
743fa04829a7/J446_BHLYGMBCXX_Lane1_CTTGTA_JD-BS.2_val_2.fq.gz
wget http://waspsystem.einstein.yu.edu/wsfile/get/file/919729fc-5ecc-43ea-8a54-
65c694185b71/J446_BHLYGMBCXX_Lane2_GTGAAA_JD-OxBS.1_val_1.fq.gz
wget http://waspsystem.einstein.yu.edu/wsfile/get/file/606a01ed-9c1e-4e51-a3dc-
0ae2f06b9465/J446_BHLYGMBCXX_Lane2_GTGAAA_JD-OxBS.2_val_2.fq.gz
```

```
==> ./sanchari/j446/mergeBams.sh <==
#!/bin/bash
#$ -cwd
#$ -j n
#$ -l h_vmem=4G
#$ -pe smp 4
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu
#usage:
#qsub mergeBams.sh outfile.bam infile1.bam infile2.bam infile3.bam
#qsub mergeBams.sh hpne_bs.bam *hpne*bs*sorted*bam
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

#make a folder for storing intermediate resulst
#otherwise, might run out of space on default /tmp
mkdir -p tmp
chmod 777 tmp

#first argument is the name of the output file
outbam=$1
shift
#the rest of the arguments are the bam files we want to merge
```

```
java -jar $(which MergeSamFiles.jar) \
    `echo $@ | sed 's/\(\S\+\)/I=&/g'`\
    ASSUME_SORTED=true\
    TMP_DIR=tmp\
    O=$outbam


==> ./sanchari/j446/reportMeth.sh <==
#!/bin/bash
#$ -cwd
#$ -j n
#$ -l h_vmem=4G
#$ -pe smp 1
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu
module load samtools/1.2/gcc.4.4.7
module load python/2.7.8/gcc.4.4.7

REF=/home/kpradha1/projects/hg19/hg19.fa
bam=$1
bname="${bam%.*}"


samtools mpileup -d1000000 -Q0 -B -f $REF $bam | \
  ../mpileup2methylation.py -i - >| ${bname}.bedGraph




==> ./sanchari/j446/mpileup2methylation.py <==
#!/usr/bin/env python

import sys
import argparse

parser = argparse.ArgumentParser(description= """
DESCRIPTION
    Extract methylation calls from samtools mpileup. Only positions with non-zero
count
    are printed. Output has is bedGraph with additional columns:
    <chrom>  <pos-1>  <pos>  <pct meth'd>  <cnt methylated>  <tot count>  <strand>

   Memo: bedGraph is 0-based, so if the first base of chr1 is C it will have
position: `chrom 0 1 ... +`

EXAMPLE & RECIPES
    samtools mpileup -d100000 -Q0 -B -l mm9.allcpg.bed.gz -f mmu.fa myreads.bam |
mpileup2methylation.py -i -

    If you want to get only the Cs in a certain context (e.g. CpG) you need to pass
    a bed file to mpileup where these positions are. For example, to get all the
    CpGs you could use:
    fastaRegexFinder.py -f mmu.fa -r CG --noreverse > mm9.allcpg.bed
    (Possibly pipe to `cut -f1-3 | gzip` to make the file smaller)

TESTED:
   14/02/2013: The output of mpileup2methylation.py is almost (but not) identical
   to bismark_methylation_extractor (files *.genome-wide_CX|CpG_report.txt). Tested
   on synthetic strands single end, and paired-end whole genome. Differences not
investigated
   further but should affect ~1 call in 1000-10000.

""", formatter_class= argparse.RawTextHelpFormatter)

parser.add_argument('--input', '-i',
                   required= True,
                   help='''Input file as produced by samtools mpileup. Use - to
read from stdin.
                   ''')

parser.add_argument('--outfmt', '-f',
                   required= False,
                   choices= ['bedgraph', 'bismark'],
                   default= 'bedgraph',
                   help='''Output format: bedgraph (default) or bismark.
'bismark' output has columns: chrom, pos, strand, count meth'd, count unmeth'd.
The first 5 column of the genome-wide cytosine report produced by
bismark_methylation_extractor should be nearly identical to 'bismark' format.
                   ''')

args= parser.parse_args()
# --------------------------------------------------------------------------------

def cleanCallString(bases):
   """Removes from the call string in mpileup (5th column) the ^ character and
   the char next to it. Note TODO: Account for insertions/deletions.
   bases:
      String of read bases (5th column of mpileup)
   Return:
      Same string as bases but with ^ and the following char removed
   Example:
      bases= '^A....,,.,.,...,,,.,....^k.'
      cleanCallString(bases) >>> '....,,.,.,...,,,.,.....'
   """
   callString= ''
   skip= False
   for x in bases:
      if x  == '^':
         skip= True
      elif skip:
         skip= False
      else:
         callString += x
   return(callString)

def pileup2methylation(chrom, pos, callString, ref, outfmt):
   """Count methylated and unmethylated calls.
   chrom, pos:
      Chromosome (string) and position (int) on the pileup
   callString:
      String of bases obtained by cleanCallString
   ref:
      Reference base as obtained from 3nd column of mpileup

   Memo: mpileup input looks like this:
   chr7    3002089 C       2       .^~.    IA
   chr7    3002090 G       2       ..      HE
   chr7    3002114 C       2       ..      HE

   """
#   ref= mpileup[2]
#   callstring= mpileup[4]
   cnt_M= 0 ## Count methylated
   cnt_m= 0 ## Count unmethylated

   if ref.upper() == 'G':
      strand= '-'
      cnt_M += callString.count(',')
      cnt_m += callString.count('a')
   elif ref.upper() == 'C':
      strand= '+'
      cnt_M += callString.count('.')
      cnt_m += callString.count('T')
   else:
      return(None)
   if (cnt_m + cnt_M) == 0:
      return(None)
   ## Use for first version of methylation2mpileup.py
   ## methList= [mpileup[0], mpileup[1], strand, str(cnt_M), str(cnt_m), ref]
   if outfmt == 'bismark':
      methList= [chrom, str(pos), strand, str(cnt_M), str(cnt_m)]
   elif outfmt == 'bedgraph':
      totreads= cnt_M + cnt_m
      methList= [chrom, str(pos-1), str(pos),
str(round(100*(float(cnt_M)/totreads), 4)), str(cnt_M), str(totreads), strand]
   else:
      sys.exit('Unexpected keyword for outfmt "%s"' %(outfmt))
   return(methList)
# ------------------------------------------------------------------------------

if args.input == '-':
   fin= sys.stdin
else:
   fin= open(args.input)

for line in fin:
   line= line.strip().split('\t')
   callString= cleanCallString(line[4])
   methList= pileup2methylation(chrom= line[0], pos= int(line[1]), callString=
callString, ref= line[2], outfmt= args.outfmt)
   if methList is None:
      pass
   else:
      try:
         print('\t'.join(methList))
      except IOError as e:
         break
fin.close()
sys.exit()




==> ./sanchari/j446/split1.sh <==
split <(zcat J446_AHLT3HBCXX_Lane1_ACAGTG_HPNE-1-BS.1_val_1.fq.gz) -d -a 3 -l
40000000 split/J446_AHLT3HBCXX_Lane1_ACAGTG_HPNE-1-BS.1_val_1.fq.pt
```

```
split <(zcat J446_AHLT3HBCXX_Lane1_ACAGTG_HPNE-1-BS.2_val_2.fq.gz) -d -a 3 -l
40000000 split/J446_AHLT3HBCXX_Lane1_ACAGTG_HPNE-1-BS.2_val_2.fq.pt
split <(zcat J446_AHLT3HBCXX_Lane2_GCCAAT_HPNE-1-OxBS.1_val_1.fq.gz) -d -a 3 -l
40000000 split/J446_AHLT3HBCXX_Lane2_GCCAAT_HPNE-1-OxBS.1_val_1.fq.pt
split <(zcat J446_AHLT3HBCXX_Lane2_GCCAAT_HPNE-1-OxBS.2_val_2.fq.gz) -d -a 3 -l
40000000 split/J446_AHLT3HBCXX_Lane2_GCCAAT_HPNE-1-OxBS.2_val_2.fq.pt
split <(zcat J446_BHLYGMBCXX_Lane1_CTTGTA_JD-BS.1_val_1.fq.gz) -d -a 3 -l 40000000
split/J446_BHLYGMBCXX_Lane1_CTTGTA_JD-BS.1_val_1.fq.pt
split <(zcat J446_BHLYGMBCXX_Lane1_CTTGTA_JD-BS.2_val_2.fq.gz) -d -a 3 -l 40000000
split/J446_BHLYGMBCXX_Lane1_CTTGTA_JD-BS.2_val_2.fq.pt
split <(zcat J446_BHLYGMBCXX_Lane2_GTGAAA_JD-OxBS.1_val_1.fq.gz) -d -a 3 -l
40000000 split/J446_BHLYGMBCXX_Lane2_GTGAAA_JD-OxBS.1_val_1.fq.pt
split <(zcat J446_BHLYGMBCXX_Lane2_GTGAAA_JD-OxBS.2_val_2.fq.gz) -d -a 3 -l
40000000 split/J446_BHLYGMBCXX_Lane2_GTGAAA_JD-OxBS.2_val_2.fq.pt

==> ./sanchari/examine_mc_hmc/examineRegions.r <==
install.packages("Gviz")
library("Gviz")
install.packages("playwith")
library("playwith")
options(stringsAsFactors=F)


region=data.frame(name = "brd4", chr="19", start=15346301, end=15393262)
region$start = region$start - 2000
region$end  = region$end + 2000


writeGeneInfo <- function(name, chr, start, end, W = 2000){
      region=data.frame(name = name, chr=chr, start=start, end=end)
      region$start = region$start - W
      region$end  = region$end + W

      #where the conversion counts are stored
      ccfol =
"/home/kpradhan/Desktop/hpc_home/projects/sanchari/j432_j446/conversionCounts/"

      #load in the jd and hpne info for the region
      #takes about 3 minutes to load in chr19
      t1 = proc.time()
      jd.oxbs = read.table(paste0(ccfol, "JD_OxBs_", region$chr, ".txt.gz"))
      jd.bs = read.table(paste0(ccfol, "JD_BS_", region$chr, ".txt.gz"))
      hpne.oxbs = read.table(paste0(ccfol, "HPNE_OxBS_", region$chr, ".txt.gz"))
      hpne.bs = read.table(paste0(ccfol, "HPNE_BS_", region$chr, ".txt.gz"))
      t2 = proc.time()
      print(t2 - t1)

      #get the rows that lie within the genes region
      jd.oxbs.reg = jd.oxbs[head(which(jd.oxbs[,2] >
region$start),1):head(which(jd.oxbs[,2] > region$end),1),]
      jd.bs.reg = jd.bs[head(which(jd.bs[,2] >
region$start),1):head(which(jd.bs[,2] > region$end),1),]
      hpne.oxbs.reg = hpne.oxbs[head(which(hpne.oxbs[,2] >
region$start),1):head(which(hpne.oxbs[,2] > region$end),1),]
      hpne.bs.reg = hpne.bs[head(which(hpne.bs[,2] >
region$start),1):head(which(hpne.bs[,2] > region$end),1),]


      #
      #merge together bs and oxbs
      jd.reg = merge(jd.bs.reg, jd.oxbs.reg, by=2)
      hpne.reg = merge(hpne.bs.reg, hpne.oxbs.reg, by=2)

      #playwith({
      #     plot(jd.reg[,1], jd.reg[,3]/jd.reg[,4], type="l")
      #})

      write.table(paste0(region$name, "_jd_bs.bdg"), x=(cbind(jd.reg[,2], jd.reg[,
1], jd.reg[,1]+1, jd.reg[,3]/jd.reg[,4])), sep="\t", col.names=F, row.names=F,
quote=F)
      write.table(paste0(region$name, "_jd_oxbs.bdg"), x=(cbind(jd.reg[,2],
jd.reg[, 1], jd.reg[,1]+1, jd.reg[,6]/jd.reg[,7])), sep="\t", col.names=F,
row.names=F, quote=F)

      write.table(paste0(region$name, "_hpne_bs.bdg"), x=(cbind(hpne.reg[,2],
hpne.reg[, 1], hpne.reg[,1]+1, hpne.reg[,3]/hpne.reg[,4])), sep="\t", col.names=F,
row.names=F, quote=F)
      write.table(paste0(region$name, "_hpne_oxbs.bdg"), x=(cbind(hpne.reg[,2],
hpne.reg[, 1], hpne.reg[,1]+1, hpne.reg[,6]/hpne.reg[,7])), sep="\t", col.names=F,
row.names=F, quote=F)

}




#brd4: chr19:15,346,301-15,393,262
#myc:  chr8:128,746,315-128,755,680
#vegfa: chr6:43,735,946-43,756,223
#vegfb: chr11:64,000,056-64,008,736
#vegfc: chr4:177,602,689-177,715,899

writeGeneInfo("brd4", "19", 15346301, 15393262)
writeGeneInfo("myc", "8", 128746315, 128755680)
writeGeneInfo("vegfa", "6", 43735946, 43756223)
writeGeneInfo("vegfb", "11", 64000056, 64008736)
writeGeneInfo("vegfc", "4", 177602689, 177715899)




==> ./sanchari/examine_mc_hmc/commands1.sh <==
#prepare the oxbs files for a specific region.

#cut out the specific regions belonging to the gene.

#plus promoters
#chr19:15235519-15332545

#or just read each chromosome in, and use R to get the region chunk

#lets try lookin at just the cpg sites
cat hpne_bs.bdg | python ../filterCpg/filterCpg.py > hpne_bs_cpg.bdg
cat hpne_oxbs.bdg | python ../filterCpg/filterCpg.py > hpne_oxbs_cpg.bdg
cat jd_bs.bdg | python ../filterCpg/filterCpg.py > jd_bs_cpg.bdg
cat jd_oxbs.bdg | python ../filterCpg/filterCpg.py > jd_oxbs_cpg.bdg

#filter all but the cpg sites and save into new files
```

```
ls *.bdg | grep -v -e "-cpg." | sed 's/\(.*\)\.bdg/cat & \| python
\.\.\/filterCpg\/filterCpg\.py >| \1_cpg.bdg/'

#merge all the cpg files by sample type
cat *_hpne_oxbs_cpg* >| convPerc_hpne_oxbs_cpg.bdg
cat *_hpne_bs_cpg* >| convPerc_hpne_bs_cpg.bdg
cat *_jd_oxbs_cpg* >| convPerc_jd_oxbs_cpg.bdg
cat *_jd_bs_cpg* >| convPerc_jd_bs_cpg.bdg
```

```
==> ./sanchari/peaks_rna/annotatePeaks.r <==
library("ChIPpeakAnno")
#source("https://bioconductor.org/biocLite.R")
#biocLite("EnsDb.Hsapiens.v75")
library(EnsDb.Hsapiens.v75)

options(stringsAsFactors=F)

#load in all the gene info from hg19
knownGenes <- genes(EnsDb.Hsapiens.v75)




#peaksFile must be the peak summit file from macs2
peaksFile = "hpne_stat3_summits.bed"
addStrandToPeaks <- function(peaksFile, newFile){
      #read in summits and convert to ranges
      peaks = read.table(peaksFile)
      #if the 3rd column is not the peak ends
      if (is.na(as.numeric(peaks[1,3]))){
            #use the peaks starts + 1
            ends = peaks[,2] + 1
      }else{
            ends = peaks[,3]
      }
      rd <- RangedData(IRanges(start = peaks[,2],
            end = ends), space = peaks[,1])
      ranges.peaks = toGRanges(rd, format="RangedData")

      #annotate ranges with the knownGenes database
      anno <- annotatePeakInBatch(ranges.peaks, AnnotationData=knownGenes)

      #make sure to only take 1 result
      x = anno[!(duplicated(anno$peak))]
      #append the strand info to last column
      y = cbind(peaks, strand=as.character(x$feature_strand))
      y$strand = as.character(x$feature_strand)
      y$strand[is.na(y$strand)] = "*"
      #write the new stranded peak file
      write.table(newFile, x=y, col.names=F, row.names=F, quote=F, sep="\t")
}
```

```
#stat3
addStrandToPeaks("test1_jdmc_top100k.txt", "test1_jdmc_top100k_stranded.txt")
addStrandToPeaks("test1_jdhmc.txt", "test1_jdhmc_stranded.txt")

addStrandToPeaks("hpne_stat3_summits.bed", "hpne_stat3_summits_stranded.txt")
addStrandToPeaks("jd13d_stat3_summits.bed", "jd_stat3_summits_stranded.txt")

peaksFile = "peaks.atac.jd.gain.1k.txt"
addStrandToPeaks("peaks.atac.jd.gain.1k.txt", "peaks.atac.jd.gain.1k_stranded.txt")
addStrandToPeaks("peaks.atac.jd.loss.1k.txt", "peaks.atac.jd.loss.1k_stranded.txt")


addStrandToPeaks("/home/kpradhan/Desktop/hpc_home/projects/sanchari/j455_j515/macs_
Chpne_sorted_mdup_Tjd_sorted_mdup_merged/Chpne_sorted_mdup_Tjd_sorted_mdup_merged_s
ummits.bed", "jd-hpne_strandedPeakSummits.txt")
addStrandToPeaks("/home/kpradhan/Desktop/hpc_home/projects/sanchari/j455_j515/macs_
Cjd_sorted_mdup_merged_Thpne_sorted_mdup/Cjd_sorted_mdup_merged_Thpne_sorted_mdup_s
ummits.bed", "hpne-jd_strandedPeakSummits.txt")


addStrandToPeaks("p3.txt", "p3_stranded.txt")

addStrandToPeaks("hmc_sig0.001_commonSites1_hpne.bed",
"hpneHmc_strandedPeakSummits_0.001.txt")
addStrandToPeaks("hmc_sig0.001_commonSites1_jd.bed",
"jdHmc_strandedPeakSummits_0.001.txt")

addStrandToPeaks("hmc_sig0.01_commonSites1_hpne.bed",
"hpneHmc_strandedPeakSummits.txt")
addStrandToPeaks("hmc_sig0.01_commonSites1_jd.bed",
"jdHmc_strandedPeakSummits.txt")

addStrandToPeaks("/home/kpradhan/Desktop/hpc_home/projects/sanchari/j455/macs_hpne_
sorted_mdup/hpne_sorted_mdup_summits.bed", "hpne_strandedPeakSummits.txt")
addStrandToPeaks("/home/kpradhan/Desktop/hpc_home/projects/sanchari/j455_j515/macs_
jd_sorted_mdup_merged/jd_sorted_mdup_merged_summits.bed",
"jd_strandedPeakSummits.txt")


addStrandToPeaks("/home/kpradhan/mnt/hpc_home/projects/colocalization/jd_hmc.txt",
"jd_strandedHmc.txt")
addStrandToPeaks("/home/kpradhan/mnt/hpc_home/projects/colocalization/hpne_hmc.txt"
, "hpne_strandedHmc.txt")



#######################################################33
#scratch
```

```
rd <- RangedData(IRanges(start = peaks.hpne[,2],
      end = peaks.hpne[,3]), space = peaks.hpne[,1])
ranges.hpne = toGRanges(rd, format="RangedData")

anno.hpne <- annotatePeakInBatch(ranges.hpne, AnnotationData=knownGenes)

x = anno.hpne[!(duplicated(anno.hpne$peak))]
#write the peaks with strand info
x$feature_strand
slot(strand(x), "values")
str(ranges(x))
y = cbind(peaks.hpne, strand=as.character(x$feature_strand))
y$strand = as.character(y$strand)
class(y$strand)
sum(is.na(y$strand))
y$strand[is.na(y$strand)] = "*"

write.table("peaks_test1.bed", x=y, col.names=F, row.names=F, quote=F, sep="\t")

head(y)

start(ranges(x))
end(ranges(x))
head(x)
str(x)
names(x)
x$ranges
x["ranges"]
ranges(x)
dim(peaks.hpne)



hmc_sig0.01_commonSites1_hpne.bed

hmc.hpne = read.table("hmc_sig0.01_commonSites1_hpne.bed")

peaks.hpne =
read.table("/home/kpradhan/Desktop/hpc_home/projects/sanchari/j455/macs_hpne_sorted
_mdup/hpne_sorted_mdup_summits.bed")
peaks.hpne =
read.table("/home/kpradhan/Desktop/hpc_home/projects/sanchari/j455/macs_hpne_sorted
_mdup/hpne_sorted_mdup_peaks.xls", header=T)
head(peaks.hpne)

peaks.jd =
read.table("/home/kpradhan/Desktop/hpc_home/projects/sanchari/j455_j515/macs_jd_sor
ted_mdup_merged/jd_sorted_mdup_merged_summits.bed")

peaks.jdMinushpne =
read.table("/home/kpradhan/Desktop/hpc_home/projects/sanchari/j455_j515/macs_Chpne_
sorted_mdup_Tjd_sorted_mdup_merged/Chpne_sorted_mdup_Tjd_sorted_mdup_merged_summits
.bed")
peaks.hpneMinusjd =
read.table("/home/kpradhan/Desktop/hpc_home/projects/sanchari/j455_j515/macs_Cjd_so
rted_mdup_merged_Thpne_sorted_mdup/Cjd_sorted_mdup_merged_Thpne_sorted_mdup_summits
.bed")
```

```
==> ./sanchari/peaks_rna/commands2_stat3Rna.sh <==

#RNA bams
bam1=/home/kpradhan/Desktop/data/sanchari/pancreatic/rnaseq/bam/JRN.AD0C99ABXX.lane
_8_P0_I2.hg19.gsnap-rna-alignment.bam
bam2=/home/kpradhan/Desktop/data/sanchari/pancreatic/rnaseq/bam/HPNE.AC1FL7ACXX.lan
e_1_P0_I5.hg19.gsnap-rna-alignment.bam



#atacseq peaks with strand information
peak1=peaks.atac.jd.gain.1k_stranded.txt
peak2=peaks.atac.jd.loss.1k_stranded.txt



time ngs.plot.r -SS same -G hg19 -R bed -E $peak1 -C $bam1 -O peaks.atac.jd.gain.1k
-T peaks.atac.jd.gain.1k -P 2 -L 3000
time ngs.plot.r -SS same -G hg19 -R bed -E $peak2 -C $bam2 -O peaks.atac.jd.loss.1k
-T peaks.atac.jd.loss.1k -P 2 -L 3000

python makeColocPlots.py jdLoss1kAtacPeaks_hpneRna
python makeColocPlots.py jdGain1kAtacPeaks_jdRna



==> ./sanchari/peaks_rna/commands1.sh <==
#get a list of differential peaks between jd and hpne.
#the peaks in one but not the other.

#see if stat3 appears in either of the differential peaks

#want to find transciption factor bindings at these diff peaks
#MOTIF analysis?

#rna coverage plot at these diff peak summits

#prepare peak strand info for the hmc sites





#RNA bams
bam1=/home/kpradhan/Desktop/data/sanchari/pancreatic/rnaseq/bam/HPNE.AC1FL7ACXX.lan
e_1_P0_I5.hg19.gsnap-rna-alignment.bam
bam2=/home/kpradhan/Desktop/data/sanchari/pancreatic/rnaseq/bam/JRN.AD0C99ABXX.lane
_8_P0_I2.hg19.gsnap-rna-alignment.bam

#atacseq peaks
peak1=/home/kpradhan/Desktop/hpc_home/projects/sanchari/j455/macs_hpne_sorted_mdup/
hpne_sorted_mdup_summits.bed
peak2=/home/kpradhan/Desktop/hpc_home/projects/sanchari/j455_j515/macs_jd_sorted_md
up_merged/jd_sorted_mdup_merged_summits.bed

#atacseq peaks with strand information
peak1=hpne_strandedPeakSummits.txt
peak2=jd_strandedPeakSummits.txt
```

```
#hmc peaks with strand
peak1=hpneHmc_strandedPeakSummits_0.001.txt
peak2=jdHmc_strandedPeakSummits_0.001.txt

cat $peak1 | awk -v OFS='\t' '{print $1,$2,$3,"1234","5678",$4}' >| p1.txt
cat $peak2 | awk -v OFS='\t' '{print $1,$2,$3,"1234","5678",$4}' >| p2.txt
hpneHmc_strandedPeakSummits.txt


cat commonSites1_hpne.sorted.bed | awk -v OFS='\t' '{print $1,$2,$3,"1234","5678",
$4}' >| p3.txt

peak1=hpneHmc_strandedPeakSummits_0.001.txt
peak2=jdHmc_strandedPeakSummits_0.001.txt

peak1=commonSites1_hpne.sorted.bed
peak2=commonSites1_jd.sorted.bed

time ngs.plot.r -SS same -G hg19 -R bed -E p1.txt -C $bam1 -O test1 -T test1 -P 2
-L 3000
time ngs.plot.r -SS same -G hg19 -R bed -E p2.txt -C $bam2 -O test2 -T test2 -P 2
-L 3000

time ngs.plot.r -S 0.1 -SS same -G hg19 -R bed -E p3_stranded.txt -C $bam1 -O test3
-T test3 -P 2 -L 3000

time ngs.plot.r -SS same -G hg19 -R bed -E $peak1 -C $bam1 -O
hpne_strandedHmcPeaks_fisher0.001_strandedRna -T
hpne_strandedHmcPeaks_fisher0.001_strandedRna -P 2 -L 3000
time ngs.plot.r -SS same -G hg19 -R bed -E $peak2 -C $bam2 -O
jd_strandedHmcPeaks_fisher0.001_strandedRna -T
jd_strandedHmcPeaks_fisher0.001_strandedRna -P 2 -L 3000


time ngs.plot.r -S 0.1 -SS same -G hg19 -R bed -E $peak1 -C $bam1 -O
hpne_strandedHmcPeaks_0.01_strandedRna -T hpne_strandedHmcPeaks_0.1_strandedRna -P
2 -L 3000
time ngs.plot.r -S 0.1 -SS same -G hg19 -R bed -E $peak2 -C $bam2 -O
jd_strandedHmcPeaks_0.01_strandedRna -T jd_strandedHmcPeaks_0.1_strandedRna -P 2 -L
3000


#hmc use same stranded reads for the stranded peaks
time ngs.plot.r -SS same -G hg19 -R bed -E $peak1 -C $bam1 -O
hpneHmc_strandedAtacPeaks_strandedRna -T hpneHmc_strandedAtacPeaks_strandedRna -P 2
-L 3000
time ngs.plot.r -SS same -G hg19 -R bed -E $peak2 -C $bam2 -O
jdHmc_strandedAtacPeaks_strandedRna -T jdHmc_strandedAtacPeaks_strandedRna -P 2 -L
3000


#use same stranded reads for the stranded peaks
time ngs.plot.r -SS same -G hg19 -R bed -E $peak1 -C $bam1 -O
hpne_strandedAtacPeaks_strandedRna -T hpne_strandedAtacPeaks_strandedRna -P 2 -L
3000
time ngs.plot.r -SS same -G hg19 -R bed -E $peak2 -C $bam2 -O
jd_strandedAtacPeaks_strandedRna -T jd_strandedAtacPeaks_strandedRna -P 2 -L 3000
```

```
#time ngs.plot.r -G hg19 -R bed -E $peak1 -C $bam1 -O atac_rna_hpne_all -T
atac_rna_hpne_all -P 2 -L 3000
#time ngs.plot.r -G hg19 -R bed -E $peak2 -C $bam2 -O atac_rna_jd_all -T
atac_rna_jd_all -P 2 -L 3000

time ngs.plot.r -G hg19 -R bed -E peaks_test1.bed -C $bam1 -O testatac_rna_hpne_all
-T testatac_rna_hpne_all -P 2 -L 3000
time ngs.plot.r -SS same -G hg19 -R bed -E peaks_test1.bed -C $bam1 -O
test2atac_rna_hpne_all -T test2atac_rna_hpne_all -P 2 -L 3000


#####################
#new stuff
#



#try it with hmc peaks?
time ngs.plot.r -G hg19 -R bed -E jdHmc_strandedPeakSummits.txt -C $bam2 -O
jdHmc0.01_jdRna -T jdHmc0.01_jdRna -P 2 -L 3000
time ngs.plot.r -G hg19 -R bed -E hpneHmc_strandedPeakSummits.txt -C $bam1 -O
hpneHmc0.01_hpneRna -T hpneHmc0.01_hpneRna -P 2 -L 3000
sort -k1,1 -k2,2n jdHmc_strandedPeakSummits.txt | ../filterCpg/filterCpg.py | cut
-f1-4 >| jdHmc_cpg_strandedPeakSummits.txt
sort -k1,1 -k2,2n hpneHmc_strandedPeakSummits.txt | ../filterCpg/filterCpg.py | cut
-f1-4 >| hpneHmc_cpg_strandedPeakSummits.txt
time ngs.plot.r -SS same -G hg19 -R bed -E jdHmc_cpg_strandedPeakSummits.txt -C
$bam2 -O jdHmc0.01cpg_jdRna -T jdHmc0.01cpg_jdRna -P 2 -L 3000
time ngs.plot.r -SS same -G hg19 -R bed -E hpneHmc_cpg_strandedPeakSummits.txt -C
$bam1 -O hpneHmc0.01cpg_hpneRna -T hpneHmc0.01cpg_hpneRna -P 2 -L 3000

#TODO:
#instead of the 0.01 hmc sites...try will all of the hmc sites
#but use a random shuf of them.


#try atacseq diff peaks and RNA
time ngs.plot.r -G hg19 -R bed -E jd-hpne_strandedPeakSummits.txt -C $bam2 -O jd-
hpneAtac_jdRna -T jd-hpneAtac_jdRna -P 2 -L 3000
time ngs.plot.r -G hg19 -R bed -E hpne-jd_strandedPeakSummits.txt -C $bam1 -O hpne-
jdAtac_jdRna -T hpne-jdAtac_jdRna -P 2 -L 3000


#TODO
#try it with mc peaks?


#scratch

hpneHmc_strandedPeakSummits.txt
cat jdHmc_strandedPeakSummits.txt | awk -v OFS='\t' '{print $1,$2,$3,"1234","5678",
$4}' >| p1.txt
cat $peak2 | awk -v OFS='\t' '{print $1,$2,$3,"1234","5678",$4}' >| p2.txt


==> ./sanchari/peaks_rna/commands2_atacRna.sh <==

#RNA bams
```

```
bam1=/home/kpradhan/Desktop/data/sanchari/pancreatic/rnaseq/bam/JRN.AD0C99ABXX.lane
_8_P0_I2.hg19.gsnap-rna-alignment.bam
bam2=/home/kpradhan/Desktop/data/sanchari/pancreatic/rnaseq/bam/HPNE.AC1FL7ACXX.lan
e_1_P0_I5.hg19.gsnap-rna-alignment.bam



#atacseq peaks with strand information
peak1=peaks.atac.jd.gain.1k_stranded.txt
peak2=peaks.atac.jd.loss.1k_stranded.txt



time ngs.plot.r -SS same -G hg19 -R bed -E $peak1 -C $bam1 -O peaks.atac.jd.gain.1k
-T peaks.atac.jd.gain.1k -P 2 -L 3000
time ngs.plot.r -SS same -G hg19 -R bed -E $peak2 -C $bam2 -O peaks.atac.jd.loss.1k
-T peaks.atac.jd.loss.1k -P 2 -L 3000

python makeColocPlots.py jdLoss1kAtacPeaks_hpneRna
python makeColocPlots.py jdGain1kAtacPeaks_jdRna


#test out the new atac peaks
bam1=/home/kpradhan/Desktop/data/sanchari/pancreatic/rnaseq/bam/JRN.AD0C99ABXX.lane
_8_P0_I2.hg19.gsnap-rna-alignment.bam
cut -f1,2,4 test1_jdmc_top100k_stranded.txt | awk '{printf("%s\t%d\t%d\t%s\n", $1,
$2, $2+1, $3)}' > test2_jdmc_top100k_stranded.txt
cut -f1,2,4 test1_jdhmc_stranded.txt | awk '{printf("%s\t%d\t%d\t%s\n", $1, $2,
$2+1, $3)}' > test2_jdhmc_stranded.txt

#cat test1_jdmc.txt | awk '{printf("%s\t%d\t%d\t%s\n", $1, $2, $2+1, $3)}' >
test2_jdmc.txt
time ngs.plot.r -SS same -G hg19 -R bed -E test2_jdmc_top100k_stranded.txt -C $bam1
-O test_jdmc_top100k -T test_jdmc_top100k -P 6 -L 3000
time ngs.plot.r -SS same -G hg19 -R bed -E test2_jdhmc_stranded.txt -C $bam1 -O
test_jdhmc -T test_jdhmc -P 6 -L 3000


==> ./sanchari/j432/test1/makeMergeCommands.sh <==
SNAME=$1
FQ1=$2
FQ2=$3
REF=/home/kpradha1/projects/hg19/

cat <<PROGRAM_END
#!/bin/bash
#$ -cwd
#$ -j n
#$ -l h_vmem=40G
#$ -pe smp 17
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20
```

```
==> ./sanchari/j432/test1/markDuplicates.sh <==
#!/bin/bash
#$ -cwd
#$ -j n
#$ -l h_vmem=4G
#$ -pe smp 4
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

bam=$1
bname="${bam%.*}"

java -jar $(which BuildBamIndex.jar) INPUT=${bam}
java -jar $(which MarkDuplicates.jar) INPUT=${bam} OUTPUT=${bname}_mdup.bam
METRICS_FILE=${bname}_mdMetric.txt
java -jar $(which BuildBamIndex.jar) INPUT=${bname}_mdup.bam



==> ./sanchari/j432/test1/batch1.sh <==
#!/bin/bash
#$ -cwd
#$ -j n
#$ -l h_vmem=40G
#$ -pe smp 1
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu
module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

FNAME=test
SNAME=foo
FQ1=${FNAME}_1.fastq.gz
FQ2=${FNAME}_2.fastq.gz
REF=/home/kpradha1/projects/hg19/
#can't use basename with multicore
#bismark --multicore 2 --bowtie2 -p 4 $REF -1 $FQ1 -2 $FQ2  --basename $SNAME
bismark --multicore 4 --bowtie2 -p 4 $REF -1 $FQ1 -2 $FQ2
#java -jar $(which SortSam.jar) INPUT=${FQ1##*/}_bismark_bt2_pe.bam OUTPUT=$
{SNAME}_sorted.bam SO=coordinate
java -jar $(which SortSam.jar) INPUT=${FQ1}_bismark_bt2_pe.bam OUTPUT=$
{SNAME}_sorted.bam SO=coordinate
java -jar $(which BuildBamIndex.jar) INPUT=${SNAME}_sorted.bam
java -jar $(which MarkDuplicates.jar) INPUT=${SNAME}_sorted.bam OUTPUT=$
{SNAME}_sorted_md.bam METRICS_FILE=${SNAME}_mdMetric.txt
java -jar $(which BuildBamIndex.jar) INPUT=${SNAME}_sorted_md.bam



==> ./sanchari/j432/test1/mergeBams_old.sh <==
#!/bin/bash
#$ -cwd
#$ -j n
#$ -l h_vmem=4G
#$ -pe smp 4
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

mkdir -p tmp
chmod 777 tmp
java -jar $(which MergeSamFiles.jar) \
    `ls *HPNE-1_BS*pe.bam | sed 's/.*/I=&/' | tr '\n' ' '`\
    ASSUME_SORTED=false\
    TMP_DIR=tmp\
    O=test.bam


==> ./sanchari/j432/test1/mergeBams.sh <==
#!/bin/bash
#$ -cwd
#$ -j n
#$ -l h_vmem=4G
#$ -pe smp 4
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu
#usage:
#qsub mergeBams.sh outfile.bam infile1.bam infile2.bam infile3.bam
#qsub mergeBams.sh hpne_bs.bam *hpne*bs*sorted*bam
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

#make a folder for storing intermediate resulst
#otherwise, might run out of space on default /tmp
mkdir -p tmp
chmod 777 tmp

#first argument is the name of the output file
outbam=$1
shift
#the rest of the arguments are the bam files we want to merge

java -jar $(which MergeSamFiles.jar) \
    `echo $@ | sed 's/\(\S\+\)/I=&/g'`\
    ASSUME_SORTED=true\
    TMP_DIR=tmp\
    O=$outbam


==> ./sanchari/j432/test1/makeBismarkBatch.sh <==
SNAME=$1
FQ1=$2
FQ2=$3
REF=/home/kpradha1/projects/hg19/

cat <<PROGRAM_END
#!/bin/bash
#$ -cwd
#$ -j n
#$ -l h_vmem=40G
#$ -pe smp 17
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=$1
FQ1=$2
FQ2=$3
REF=/home/kpradha1/projects/hg19/
bismark --multicore 4 --bowtie2 -p 4 $REF -1 $FQ1 -2 $FQ2
java -jar \$(which SortSam.jar) INPUT=${FQ1}_bismark_bt2_pe.bam OUTPUT=$
{SNAME}_sorted.bam SO=coordinate CREATE_INDEX=true

PROGRAM_END


==> ./sanchari/j432/test1/testSetup1.sh <==
#change this to ls *.1_val_1*
for fq1 in `ls ../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.1_val_1*`;
do
    fq2=`echo $fq1 | sed 's/\(.*\)1_val_1\(.*\)/\12_val_2\2/'`
    pname=`echo $fq1 | sed 's/.*\/\(.*\).1_val_1.fq.\(p.*\)/\1_\2/'`
    #echo $fq1
    #echo $fq2
    #echo $pname
    echo "./makeBismarkBatch.sh $pname $fq1 $fq2 | qsub"
done

==> ./sanchari/j432/test1/do_bismark1.sh <==
#!/bin/bash
#$ -cwd
#$ -j n
#$ -l h_vmem=15G
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu
#
# do_bismark.sh
# Brent Calder <brent.calder@einstein.yu.edu>
#
# Convienience script to take fastq.gz input, run trim_galore,
# fastqc, split input, run Bismark, then merge, sort, and index
# output into BAM.
#


set -o errexit -o nounset -o allexport -o pipefail

# user options

GENOME_FOLDER=/home/kpradha1/projects/hg19/
GENOME=mm9.fa # not used, this is the original
# genome name at ../fasta, relative to GENOME_FOLDER
```

```
TRIM_GALORE_OPTS="" # mainly for new adapter sequence or clips
FASTQC_OPTS="" # optional FASTQC options
BISMARK_OPTS="" # see documentation, 'non-directional' set below (DIRECTIONS)
BOWTIE2_OPTS="" # bowtie2 options for bismark

# number of reads per split work file
READS_PER_FASTQ=4000000
# DIRECTIONS must be 2 or 4 (eg. 4 = "--non_directional")
DIRECTIONS=4
# NTHREADS is the number of threads per DIRECTION
NTHREADS=2
# DIRECTIONS*THREADS = num_proc

########################
# DO NOT EDIT BELOW HERE

USAGE="Usage:\t\tqsub do_bismark.sh forward_read.fastq.gz
[reverse_read.fastq.gz]\n"
ADMON="reads must be in format: name.fastq.gz\n"
PAIRED=0
REVERSE_READ=""
MYID=$$-${RANDOM}

if [ "$DIRECTIONS" -eq "4" ]; then
    BISMARK_OPTS="${BISMARK_OPTS} --non_directional"
else
    # default back to directional
    DIRECTIONS=2
fi

if [ "$#" -eq "2" ]; then
    PAIRED=1
    REVERSE_READ=$2
    REVERSE=`basename ${REVERSE_READ/.fastq.gz/}`
    TRIM_GALORE_OPTS="${TRIM_GALORE_OPTS} --paired"
    if [ "${#REVERSE_READ}" -eq "${#REVERSE}" ]; then
        USAGE="${USAGE}${ADMON}"
        echo -e "$USAGE" >&2
        exit 2
    fi
else
    if [ "$#" -ne "1" ]; then
        echo -e "$USAGE" >&2
        exit 1
    fi
fi

FORWARD_READ=$1
FORWARD=`basename ${FORWARD_READ/.fastq.gz/}`

if [ "${#FORWARD_READ}" -eq "${#FORWARD}" ]; then
    USAGE="${USAGE}${ADMON}"
    echo -e "$USAGE" >&2
    exit 2
fi

##################
# REQUIRED MODULES
```

```
##################
#source /apps1/modules/init/bash

module load bowtie2/2.2.3/gcc.4.4.7
module load trim_galore/0.3.7/gcc.4.4.7
module load bismark/0.14.5
module load FastQC/0.11.4/java.1.8.0_20
module load picard-tools/1.92/java.1.8.0_20
# might be necessary to set up qsub
if [ -e /etc/profile.d/sge-binaries.sh ]; then
    source /etc/profile.d/sge-binaries.sh
fi

# output dirs
mkdir -p fastqc split bismark trimmed tmp

# trim illumina adapters and -q 20 by default
trim_galore --dont_gzip -o trimmed --fastqc --fastqc_args "${FASTQC_OPTS} -o
./fastqc" $TRIM_GALORE_OPTS $FORWARD_READ $REVERSE_READ

# split files based on READS_PER_FASTQ
if [ "$PAIRED" -eq "0" ]; then
    split -a 3 -l $((${READS_PER_FASTQ}*4)) trimmed/${FORWARD}_trimmed.fq split/$
{FORWARD}.split.
else
    if [ "$PAIRED" -eq "1" ]; then
        split -a 3 -l $((${READS_PER_FASTQ}*4)) trimmed/${FORWARD}_val_1.fq split/$
{FORWARD}.split.
        split -a 3 -l $((${READS_PER_FASTQ}*4)) trimmed/${REVERSE}_val_2.fq split/$
{REVERSE}.split.
    fi
fi

# run bismark on the split files
for x in `ls -1 split/${FORWARD}.split.*`; do
    READSTR=${x}
    suf=${x/*split./}
    if [ "$PAIRED" -eq "1" ]; then
        second=split/${REVERSE}.split.${suf}
        READSTR="-1 ${x} -2 ${second}"
    fi
    qsub -cwd -l h_vmem=30G -V -N bismark-${MYID} -j y -b y bismark --temp_dir tmp
-o bismark --bowtie2 -p $((${NTHREADS}*${DIRECTIONS})) $BISMARK_OPTS $BOWTIE2_OPTS
$GENOME_FOLDER $READSTR
done

# use picard to merge, sort, create bam and index split SAM upon completion.
SAMS=""
if [ "$PAIRED" -eq "0" ]; then
    SAMS=`ls -1 split/${FORWARD}.split.* | sed 's|^split|I=bismark|g' | sed 's/
$/_bismark_bt2.sam/g' | tr "\n" " "`
else
    SAMS=`ls -1 split/${FORWARD}.split.* | sed 's|^split|I=bismark|g' | sed 's/
$/_bismark_bt2_pe.sam/g' | tr "\n" " "`
fi
BAM=${FORWARD}.bam

qsub -S /bin/bash -hold_jid bismark-${MYID} -cwd -V -l h_vmem=4G -j y -N merge -b y
java -jar $(which MergeSamFiles.jar) $SAMS O=$BAM AS=false SO=coordinate
```

```
TMP_DIR=./tmp CREATE_INDEX=true


==> ./sanchari/j432/test3/do_bismark.sh <==
#!/bin/bash
#$ -S /bin/bash
#$ -cwd
#$ -j y
#$ -l p=2
#$ -l mf=30G
#
# do_bismark.sh
# Brent Calder <brent.calder@einstein.yu.edu>
#
# Convienience script to take fastq.gz input, run trim_galore,
# fastqc, split input, run Bismark, then merge, sort, and index
# output into BAM.
#


set -o errexit -o nounset -o allexport -o pipefail

# user options

GENOME_FOLDER=/rio1/indexes/hg19/bowtie2/
GENOME=mm9.fa # not used, this is the original
# genome name at ../fasta, relative to GENOME_FOLDER

TRIM_GALORE_OPTS="" # mainly for new adapter sequence or clips
FASTQC_OPTS="" # optional FASTQC options
BISMARK_OPTS="" # see documentation, 'non-directional' set below (DIRECTIONS)
BOWTIE2_OPTS="" # bowtie2 options for bismark

# number of reads per split work file
READS_PER_FASTQ=4000000
# DIRECTIONS must be 2 or 4 (eg. 4 = "--non_directional")
DIRECTIONS=4
# NTHREADS is the number of threads per DIRECTION
NTHREADS=2
# DIRECTIONS*THREADS = num_proc

########################
# DO NOT EDIT BELOW HERE

USAGE="Usage:\t\tqsub do_bismark.sh forward_read.fastq.gz
[reverse_read.fastq.gz]\n"
ADMON="reads must be in format: name.fastq.gz\n"
PAIRED=0
REVERSE_READ=""
MYID=$$-${RANDOM}

if [ "$DIRECTIONS" -eq "4" ]; then
    BISMARK_OPTS="${BISMARK_OPTS} --non_directional"
else
    # default back to directional
    DIRECTIONS=2
fi

if [ "$#" -eq "2" ]; then
```

```
    PAIRED=1
    REVERSE_READ=$2
    REVERSE=`basename ${REVERSE_READ/.fastq.gz/}`
    TRIM_GALORE_OPTS="${TRIM_GALORE_OPTS} --paired"
    if [ "${#REVERSE_READ}" -eq "${#REVERSE}" ]; then
        USAGE="${USAGE}${ADMON}"
        echo -e "$USAGE" >&2
        exit 2
    fi
else
    if [ "$#" -ne "1" ]; then
        echo -e "$USAGE" >&2
        exit 1
    fi
fi

FORWARD_READ=$1
FORWARD=`basename ${FORWARD_READ/.fastq.gz/}`

if [ "${#FORWARD_READ}" -eq "${#FORWARD}" ]; then
    USAGE="${USAGE}${ADMON}"
    echo -e "$USAGE" >&2
    exit 2
fi

##################
# REQUIRED MODULES
##################
source /apps1/modules/init/bash

module load trim_galore/0.3.3
module load bismark/0.10.1
module load fastqc/0.10.1
module load picard/1.104

# might be necessary to set up qsub
if [ -e /etc/profile.d/sge-binaries.sh ]; then
    source /etc/profile.d/sge-binaries.sh
fi

# output dirs
mkdir -p fastqc split bismark trimmed tmp

# trim illumina adapters and -q 20 by default
trim_galore --dont_gzip -o trimmed --fastqc --fastqc_args "${FASTQC_OPTS} -o
./fastqc" $TRIM_GALORE_OPTS $FORWARD_READ $REVERSE_READ

# split files based on READS_PER_FASTQ
if [ "$PAIRED" -eq "0" ]; then
    split -a 3 -l $((${READS_PER_FASTQ}*4)) trimmed/${FORWARD}_trimmed.fq split/$
{FORWARD}.split.
else
    if [ "$PAIRED" -eq "1" ]; then
        split -a 3 -l $((${READS_PER_FASTQ}*4)) trimmed/${FORWARD}_val_1.fq split/$
{FORWARD}.split.
        split -a 3 -l $((${READS_PER_FASTQ}*4)) trimmed/${REVERSE}_val_2.fq split/$
{REVERSE}.split.
    fi
fi
```

```
# run bismark on the split files
for x in `ls -1 split/${FORWARD}.split.*`; do
    READSTR=${x}
    suf=${x/*split./}
    if [ "$PAIRED" -eq "1" ]; then
        second=split/${REVERSE}.split.${suf}
        READSTR="-1 ${x} -2 ${second}"
    fi
    qsub -cwd -l p=$((${NTHREADS}*${DIRECTIONS})) -l mf=30g -V -N bismark-${MYID}
-j y -b y bismark --temp_dir tmp -o bismark --bowtie2 -p $((${NTHREADS}*$
{DIRECTIONS})) $BISMARK_OPTS $BOWTIE2_OPTS $GENOME_FOLDER $READSTR
done

# use picard to merge, sort, create bam and index split SAM upon completion.
SAMS=""
if [ "$PAIRED" -eq "0" ]; then
    SAMS=`ls -1 split/${FORWARD}.split.* | sed 's|^split|I=bismark|g' | sed 's/
$/_bismark_bt2.sam/g' | tr "\n" " "`
else
    SAMS=`ls -1 split/${FORWARD}.split.* | sed 's|^split|I=bismark|g' | sed 's/
$/_bismark_bt2_pe.sam/g' | tr "\n" " "`
fi
BAM=${FORWARD}.bam

qsub -S /bin/bash -hold_jid bismark-${MYID} -cwd -V -l mf=4g -j y -N merge -b y
java -jar $PICARD_ROOT/MergeSamFiles.jar $SAMS O=$BAM AS=false SO=coordinate
TMP_DIR=./tmp CREATE_INDEX=true


==> ./sanchari/j432/test3/batch3.sh <==
#!/bin/bash
#$ -cwd
#$ -j n
#$ -l h_vmem=15G
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu
module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

FNAME=J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS
SNAME=hpne_bs
FQ1=${FNAME}.1_val_1.fq.gz
FQ2=${FNAME}.2_val_2.fq.gz
REF=/home/kpradha1/projects/hg19/
bismark --bowtie2 $REF -1 $FQ1 -2 $FQ2  --basename $SNAME
java -jar $(which SortSam.jar) INPUT=${SNAME}_pe.bam OUTPUT=${SNAME}_sorted.bam
SO=coordinate
java -jar $(which BuildBamIndex.jar) INPUT=${SNAME}_sorted.bam
java -jar $(which MarkDuplicates.jar) INPUT=${SNAME}_sorted.bam OUTPUT=$
{SNAME}_sorted_md.bam METRICS_FILE=${SNAME}_mdMetric.txt
java -jar $(which BuildBamIndex.jar) INPUT=${SNAME}_sorted_md.bam



==> ./sanchari/j432/test3/do_bismark1.sh <==
#!/bin/bash
#$ -cwd
#$ -j n
#$ -l h_vmem=15G
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu
#
# do_bismark.sh
# Brent Calder <brent.calder@einstein.yu.edu>
#
# Convienience script to take fastq.gz input, run trim_galore,
# fastqc, split input, run Bismark, then merge, sort, and index
# output into BAM.
#


set -o errexit -o nounset -o allexport -o pipefail

# user options

GENOME_FOLDER=/home/kpradha1/projects/hg19/
GENOME=mm9.fa # not used, this is the original
# genome name at ../fasta, relative to GENOME_FOLDER

TRIM_GALORE_OPTS="" # mainly for new adapter sequence or clips
FASTQC_OPTS="" # optional FASTQC options
BISMARK_OPTS="" # see documentation, 'non-directional' set below (DIRECTIONS)
BOWTIE2_OPTS="" # bowtie2 options for bismark

# number of reads per split work file
READS_PER_FASTQ=4000000
# DIRECTIONS must be 2 or 4 (eg. 4 = "--non_directional")
DIRECTIONS=4
# NTHREADS is the number of threads per DIRECTION
NTHREADS=2
# DIRECTIONS*THREADS = num_proc

########################
# DO NOT EDIT BELOW HERE

USAGE="Usage:\t\tqsub do_bismark.sh forward_read.fastq.gz
[reverse_read.fastq.gz]\n"
ADMON="reads must be in format: name.fastq.gz\n"
PAIRED=0
REVERSE_READ=""
MYID=$$-${RANDOM}

if [ "$DIRECTIONS" -eq "4" ]; then
    BISMARK_OPTS="${BISMARK_OPTS} --non_directional"
else
    # default back to directional
    DIRECTIONS=2
fi

if [ "$#" -eq "2" ]; then
    PAIRED=1
    REVERSE_READ=$2
    REVERSE=`basename ${REVERSE_READ/.fastq.gz/}`
    TRIM_GALORE_OPTS="${TRIM_GALORE_OPTS} --paired"
    if [ "${#REVERSE_READ}" -eq "${#REVERSE}" ]; then
        USAGE="${USAGE}${ADMON}"
        echo -e "$USAGE" >&2
        exit 2
    fi
else
    if [ "$#" -ne "1" ]; then
        echo -e "$USAGE" >&2
        exit 1
    fi
fi

FORWARD_READ=$1
FORWARD=`basename ${FORWARD_READ/.fastq.gz/}`

if [ "${#FORWARD_READ}" -eq "${#FORWARD}" ]; then
    USAGE="${USAGE}${ADMON}"
    echo -e "$USAGE" >&2
    exit 2
fi

##################
# REQUIRED MODULES
##################
#source /apps1/modules/init/bash

module load trim_galore/0.3.7/gcc.4.4.7
module load bismark/0.14.5
module load FastQC/0.11.4/java.1.8.0_20
module load picard-tools/1.92/java.1.8.0_20

# might be necessary to set up qsub
if [ -e /etc/profile.d/sge-binaries.sh ]; then
    source /etc/profile.d/sge-binaries.sh
fi

# output dirs
mkdir -p fastqc split bismark trimmed tmp

# trim illumina adapters and -q 20 by default
trim_galore --dont_gzip -o trimmed --fastqc --fastqc_args "${FASTQC_OPTS} -o
./fastqc" $TRIM_GALORE_OPTS $FORWARD_READ $REVERSE_READ

# split files based on READS_PER_FASTQ
if [ "$PAIRED" -eq "0" ]; then
    split -a 3 -l $((${READS_PER_FASTQ}*4)) trimmed/${FORWARD}_trimmed.fq split/$
{FORWARD}.split.
else
    if [ "$PAIRED" -eq "1" ]; then
        split -a 3 -l $((${READS_PER_FASTQ}*4)) trimmed/${FORWARD}_val_1.fq split/$
{FORWARD}.split.
        split -a 3 -l $((${READS_PER_FASTQ}*4)) trimmed/${REVERSE}_val_2.fq split/$
{REVERSE}.split.
    fi
fi

# run bismark on the split files
for x in `ls -1 split/${FORWARD}.split.*`; do
```

```
    READSTR=${x}
    suf=${x/*split./}
    if [ "$PAIRED" -eq "1" ]; then
        second=split/${REVERSE}.split.${suf}
        READSTR="-1 ${x} -2 ${second}"
    fi
    qsub -cwd -l p=$((${NTHREADS}*${DIRECTIONS})) -l mf=30g -V -N bismark-${MYID}
-j y -b y bismark --temp_dir tmp -o bismark --bowtie2 -p $((${NTHREADS}*$
{DIRECTIONS})) $BISMARK_OPTS $BOWTIE2_OPTS $GENOME_FOLDER $READSTR
done

# use picard to merge, sort, create bam and index split SAM upon completion.
SAMS=""
if [ "$PAIRED" -eq "0" ]; then
    SAMS=`ls -1 split/${FORWARD}.split.* | sed 's|^split|I=bismark|g' | sed 's/
$/_bismark_bt2.sam/g' | tr "\n" " "`
else
    SAMS=`ls -1 split/${FORWARD}.split.* | sed 's|^split|I=bismark|g' | sed 's/
$/_bismark_bt2_pe.sam/g' | tr "\n" " "`
fi
BAM=${FORWARD}.bam

qsub -S /bin/bash -hold_jid bismark-${MYID} -cwd -V -l mf=4g -j y -N merge -b y
java -jar $PICARD_ROOT/MergeSamFiles.jar $SAMS O=$BAM AS=false SO=coordinate
TMP_DIR=./tmp CREATE_INDEX=true


==> ./sanchari/j432/bismark_aligned/clipOverlaps.sh <==
#!/bin/bash
#$ -cwd
#$ -j n
#$ -l h_vmem=4G
#$ -pe smp 1
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu
module load samtools/1.2/gcc.4.4.7
module load bamUtil/1.0.13/gcc.4.4.7

bam=$1
bname="${bam%.*}"

bam clipOverlap --in $bam --out ${bname}_clip.bam
samtools index ${bname}_clip.bam



==> ./sanchari/j432/bismark_aligned/sortBam.sh <==
#!/bin/bash
#$ -cwd
#$ -j n
#$ -l h_vmem=4G
#$ -pe smp 1
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu
module load samtools/1.2/gcc.4.4.7

bam=$1
```

```
bname="${bam%.*}"

samtools sort $bam ${bname}_sorted
samtools index ${bname}_sorted.bam



==> ./sanchari/j432/bismark_aligned/markDuplicates.sh <==
#!/bin/bash
#$ -cwd
#$ -j n
#$ -l h_vmem=4G
#$ -pe smp 4
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

bam=$1
bname="${bam%.*}"

java -jar $(which BuildBamIndex.jar) INPUT=${bam}
java -jar $(which MarkDuplicates.jar) INPUT=${bam} OUTPUT=${bname}_mdup.bam
METRICS_FILE=${bname}_mdMetric.txt
java -jar $(which BuildBamIndex.jar) INPUT=${bname}_mdup.bam



==> ./sanchari/j432/bismark_aligned/nameSort.sh <==
#!/bin/bash
#$ -cwd
#$ -j n
#$ -l h_vmem=4G
#$ -pe smp 4
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu
module load samtools/1.2/gcc.4.4.7

bam=$1
bname="${bam%.*}"

samtools sort -n $bam ${bname}_nsorted
samtools index ${bname}_nsorted.bam



==> ./sanchari/j432/bismark_aligned/qscripts/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-
OxBS_p011.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu
```

```
module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS_p011
FQ1=../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.1_val_1.fq.p011
FQ2=../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.2_val_2.fq.p011
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.1_val_1.fq.p011 -2
../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.2_val_2.fq.p011
java -jar $(which SortSam.jar) INPUT=J432_BHKVCGBCXX_Lane2_GTGAAA_JD-
OxBS.1_val_1.fq.p011_bismark_bt2_pe.bam OUTPUT=J432_BHKVCGBCXX_Lane2_GTGAAA_JD-
OxBS_p011_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS_p011.status

==> ./sanchari/j432/bismark_aligned/qscripts/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-
OxBS_p014.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS_p014
FQ1=../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.1_val_1.fq.p014
FQ2=../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.2_val_2.fq.p014
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.1_val_1.fq.p014 -2
../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.2_val_2.fq.p014
java -jar $(which SortSam.jar) INPUT=J432_BHKVCGBCXX_Lane2_GTGAAA_JD-
OxBS.1_val_1.fq.p014_bismark_bt2_pe.bam OUTPUT=J432_BHKVCGBCXX_Lane2_GTGAAA_JD-
OxBS_p014_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS_p014.status

==> ./sanchari/j432/bismark_aligned/qscripts/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-
BS_p000.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu
```

```
module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS_p000
FQ1=../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.1_val_1.fq.p000
FQ2=../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.2_val_2.fq.p000
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.1_val_1.fq.p000 -2
../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.2_val_2.fq.p000
java -jar $(which SortSam.jar) INPUT=J432_BHKVCGBCXX_Lane1_CTTGTA_JD-
BS.1_val_1.fq.p000_bismark_bt2_pe.bam OUTPUT=J432_BHKVCGBCXX_Lane1_CTTGTA_JD-
BS_p000_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS_p000.status

==> ./sanchari/j432/bismark_aligned/qscripts/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-
1_BS_p000.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS_p000
FQ1=../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.1_val_1.fq.p000
FQ2=../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.2_val_2.fq.p000
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.1_val_1.fq.p000 -2
../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.2_val_2.fq.p000
java -jar $(which SortSam.jar) INPUT=J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-
1_BS.1_val_1.fq.p000_bismark_bt2_pe.bam OUTPUT=J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-
1_BS_p000_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS_p000.status

==> ./sanchari/j432/bismark_aligned/qscripts/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-
1OxBS_p008.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS_p008
FQ1=../trimmed/split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.1_val_1.fq.p008
FQ2=../trimmed/split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.2_val_2.fq.p008
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.1_val_1.fq.p008 -2
../trimmed/split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.2_val_2.fq.p008
java -jar $(which SortSam.jar) INPUT=J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-
1OxBS.1_val_1.fq.p008_bismark_bt2_pe.bam OUTPUT=J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-
1OxBS_p008_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS_p008.status

==> ./sanchari/j432/bismark_aligned/qscripts/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-
BS_p001.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS_p001
FQ1=../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.1_val_1.fq.p001
FQ2=../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.2_val_2.fq.p001
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.1_val_1.fq.p001 -2
../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.2_val_2.fq.p001
java -jar $(which SortSam.jar) INPUT=J432_BHKVCGBCXX_Lane1_CTTGTA_JD-
BS.1_val_1.fq.p001_bismark_bt2_pe.bam OUTPUT=J432_BHKVCGBCXX_Lane1_CTTGTA_JD-
BS_p001_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS_p001.status

==> ./sanchari/j432/bismark_aligned/qscripts/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-
OxBS_p006.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS_p006
FQ1=../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.1_val_1.fq.p006
FQ2=../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.2_val_2.fq.p006
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.1_val_1.fq.p006 -2
../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.2_val_2.fq.p006
java -jar $(which SortSam.jar) INPUT=J432_BHKVCGBCXX_Lane2_GTGAAA_JD-
OxBS.1_val_1.fq.p006_bismark_bt2_pe.bam OUTPUT=J432_BHKVCGBCXX_Lane2_GTGAAA_JD-
OxBS_p006_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS_p006.status

==> ./sanchari/j432/bismark_aligned/qscripts/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-
1OxBS_p013.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS_p013
FQ1=../trimmed/split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.1_val_1.fq.p013
FQ2=../trimmed/split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.2_val_2.fq.p013
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.1_val_1.fq.p013 -2
../trimmed/split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.2_val_2.fq.p013
java -jar $(which SortSam.jar) INPUT=J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-
1OxBS.1_val_1.fq.p013_bismark_bt2_pe.bam OUTPUT=J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-
1OxBS_p013_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS_p013.status

==> ./sanchari/j432/bismark_aligned/qscripts/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-
1OxBS_p001.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS_p001
FQ1=../trimmed/split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.1_val_1.fq.p001
FQ2=../trimmed/split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.2_val_2.fq.p001
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.1_val_1.fq.p001 -2
../trimmed/split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.2_val_2.fq.p001
java -jar $(which SortSam.jar) INPUT=J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-
1OxBS.1_val_1.fq.p001_bismark_bt2_pe.bam OUTPUT=J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-
1OxBS_p001_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS_p001.status

==> ./sanchari/j432/bismark_aligned/qscripts/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-
1_BS_p010.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS_p010
FQ1=../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.1_val_1.fq.p010
FQ2=../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.2_val_2.fq.p010
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.1_val_1.fq.p010 -2
../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.2_val_2.fq.p010
java -jar $(which SortSam.jar) INPUT=J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-
1_BS.1_val_1.fq.p010_bismark_bt2_pe.bam OUTPUT=J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-
1_BS_p010_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS_p010.status

==> ./sanchari/j432/bismark_aligned/qscripts/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-
1_BS_p012.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20
```

```
SNAME=J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS_p012
FQ1=../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.1_val_1.fq.p012
FQ2=../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.2_val_2.fq.p012
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.1_val_1.fq.p012 -2
../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.2_val_2.fq.p012
java -jar $(which SortSam.jar) INPUT=J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-
1_BS.1_val_1.fq.p012_bismark_bt2_pe.bam OUTPUT=J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-
1_BS_p012_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS_p012.status

==> ./sanchari/j432/bismark_aligned/qscripts/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-
BS_p008.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS_p008
FQ1=../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.1_val_1.fq.p008
FQ2=../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.2_val_2.fq.p008
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.1_val_1.fq.p008 -2
../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.2_val_2.fq.p008
java -jar $(which SortSam.jar) INPUT=J432_BHKVCGBCXX_Lane1_CTTGTA_JD-
BS.1_val_1.fq.p008_bismark_bt2_pe.bam OUTPUT=J432_BHKVCGBCXX_Lane1_CTTGTA_JD-
BS_p008_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS_p008.status

==> ./sanchari/j432/bismark_aligned/qscripts/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-
1_BS_p002.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20
```

```
SNAME=J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS_p002
FQ1=../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.1_val_1.fq.p002
FQ2=../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.2_val_2.fq.p002
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.1_val_1.fq.p002 -2
../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.2_val_2.fq.p002
java -jar $(which SortSam.jar) INPUT=J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-
1_BS.1_val_1.fq.p002_bismark_bt2_pe.bam OUTPUT=J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-
1_BS_p002_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS_p002.status

==> ./sanchari/j432/bismark_aligned/qscripts/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-
BS_p009.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS_p009
FQ1=../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.1_val_1.fq.p009
FQ2=../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.2_val_2.fq.p009
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.1_val_1.fq.p009 -2
../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.2_val_2.fq.p009
java -jar $(which SortSam.jar) INPUT=J432_BHKVCGBCXX_Lane1_CTTGTA_JD-
BS.1_val_1.fq.p009_bismark_bt2_pe.bam OUTPUT=J432_BHKVCGBCXX_Lane1_CTTGTA_JD-
BS_p009_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS_p009.status

==> ./sanchari/j432/bismark_aligned/qscripts/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-
1OxBS_p012.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS_p012
```

```
FQ1=../trimmed/split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.1_val_1.fq.p012
FQ2=../trimmed/split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.2_val_2.fq.p012
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.1_val_1.fq.p012 -2
../trimmed/split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.2_val_2.fq.p012
java -jar $(which SortSam.jar) INPUT=J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-
1OxBS.1_val_1.fq.p012_bismark_bt2_pe.bam OUTPUT=J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-
1OxBS_p012_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS_p012.status

==> ./sanchari/j432/bismark_aligned/qscripts/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-
BS_p014.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS_p014
FQ1=../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.1_val_1.fq.p014
FQ2=../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.2_val_2.fq.p014
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.1_val_1.fq.p014 -2
../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.2_val_2.fq.p014
java -jar $(which SortSam.jar) INPUT=J432_BHKVCGBCXX_Lane1_CTTGTA_JD-
BS.1_val_1.fq.p014_bismark_bt2_pe.bam OUTPUT=J432_BHKVCGBCXX_Lane1_CTTGTA_JD-
BS_p014_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS_p014.status

==> ./sanchari/j432/bismark_aligned/qscripts/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-
1_BS_p009.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS_p009
FQ1=../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.1_val_1.fq.p009
```

```
FQ2=../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.2_val_2.fq.p009
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.1_val_1.fq.p009 -2
../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.2_val_2.fq.p009
java -jar $(which SortSam.jar) INPUT=J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-
1_BS.1_val_1.fq.p009_bismark_bt2_pe.bam OUTPUT=J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-
1_BS_p009_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS_p009.status

==> ./sanchari/j432/bismark_aligned/qscripts/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-
1OxBS_p000.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS_p000
FQ1=../trimmed/split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.1_val_1.fq.p000
FQ2=../trimmed/split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.2_val_2.fq.p000
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.1_val_1.fq.p000 -2
../trimmed/split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.2_val_2.fq.p000
java -jar $(which SortSam.jar) INPUT=J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-
1OxBS.1_val_1.fq.p000_bismark_bt2_pe.bam OUTPUT=J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-
1OxBS_p000_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS_p000.status

==> ./sanchari/j432/bismark_aligned/qscripts/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-
1_BS_p014.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS_p014
FQ1=../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.1_val_1.fq.p014
FQ2=../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.2_val_2.fq.p014
```

```
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.1_val_1.fq.p014 -2
../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.2_val_2.fq.p014
java -jar $(which SortSam.jar) INPUT=J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-
1_BS.1_val_1.fq.p014_bismark_bt2_pe.bam OUTPUT=J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-
1_BS_p014_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS_p014.status

==> ./sanchari/j432/bismark_aligned/qscripts/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-
1_BS_p006.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS_p006
FQ1=../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.1_val_1.fq.p006
FQ2=../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.2_val_2.fq.p006
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.1_val_1.fq.p006 -2
../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.2_val_2.fq.p006
java -jar $(which SortSam.jar) INPUT=J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-
1_BS.1_val_1.fq.p006_bismark_bt2_pe.bam OUTPUT=J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-
1_BS_p006_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS_p006.status

==> ./sanchari/j432/bismark_aligned/qscripts/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-
BS_p006.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS_p006
FQ1=../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.1_val_1.fq.p006
FQ2=../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.2_val_2.fq.p006
REF=/home/kpradha1/projects/hg19/
```

```
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.1_val_1.fq.p006 -2
../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.2_val_2.fq.p006
java -jar $(which SortSam.jar) INPUT=J432_BHKVCGBCXX_Lane1_CTTGTA_JD-
BS.1_val_1.fq.p006_bismark_bt2_pe.bam OUTPUT=J432_BHKVCGBCXX_Lane1_CTTGTA_JD-
BS_p006_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS_p006.status

==> ./sanchari/j432/bismark_aligned/qscripts/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-
1OxBS_p009.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS_p009
FQ1=../trimmed/split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.1_val_1.fq.p009
FQ2=../trimmed/split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.2_val_2.fq.p009
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.1_val_1.fq.p009 -2
../trimmed/split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.2_val_2.fq.p009
java -jar $(which SortSam.jar) INPUT=J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-
1OxBS.1_val_1.fq.p009_bismark_bt2_pe.bam OUTPUT=J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-
1OxBS_p009_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS_p009.status

==> ./sanchari/j432/bismark_aligned/qscripts/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-
1_BS_p005.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS_p005
FQ1=../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.1_val_1.fq.p005
FQ2=../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.2_val_2.fq.p005
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
```

```
../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.1_val_1.fq.p005 -2
../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.2_val_2.fq.p005
java -jar $(which SortSam.jar) INPUT=J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-
1_BS.1_val_1.fq.p005_bismark_bt2_pe.bam OUTPUT=J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-
1_BS_p005_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS_p005.status

==> ./sanchari/j432/bismark_aligned/qscripts/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-
OxBS_p012.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS_p012
FQ1=../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.1_val_1.fq.p012
FQ2=../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.2_val_2.fq.p012
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.1_val_1.fq.p012 -2
../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.2_val_2.fq.p012
java -jar $(which SortSam.jar) INPUT=J432_BHKVCGBCXX_Lane2_GTGAAA_JD-
OxBS.1_val_1.fq.p012_bismark_bt2_pe.bam OUTPUT=J432_BHKVCGBCXX_Lane2_GTGAAA_JD-
OxBS_p012_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS_p012.status

==> ./sanchari/j432/bismark_aligned/qscripts/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-
1_BS_p008.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS_p008
FQ1=../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.1_val_1.fq.p008
FQ2=../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.2_val_2.fq.p008
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.1_val_1.fq.p008 -2
```

```
../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.2_val_2.fq.p008
java -jar $(which SortSam.jar) INPUT=J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-
1_BS.1_val_1.fq.p008_bismark_bt2_pe.bam OUTPUT=J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-
1_BS_p008_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS_p008.status

==> ./sanchari/j432/bismark_aligned/qscripts/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-
BS_p002.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS_p002
FQ1=../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.1_val_1.fq.p002
FQ2=../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.2_val_2.fq.p002
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.1_val_1.fq.p002 -2
../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.2_val_2.fq.p002
java -jar $(which SortSam.jar) INPUT=J432_BHKVCGBCXX_Lane1_CTTGTA_JD-
BS.1_val_1.fq.p002_bismark_bt2_pe.bam OUTPUT=J432_BHKVCGBCXX_Lane1_CTTGTA_JD-
BS_p002_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS_p002.status

==> ./sanchari/j432/bismark_aligned/qscripts/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-
OxBS_p013.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS_p013
FQ1=../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.1_val_1.fq.p013
FQ2=../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.2_val_2.fq.p013
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.1_val_1.fq.p013 -2
../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.2_val_2.fq.p013
```

```
java -jar $(which SortSam.jar) INPUT=J432_BHKVCGBCXX_Lane2_GTGAAA_JD-
OxBS.1_val_1.fq.p013_bismark_bt2_pe.bam OUTPUT=J432_BHKVCGBCXX_Lane2_GTGAAA_JD-
OxBS_p013_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS_p013.status

==> ./sanchari/j432/bismark_aligned/qscripts/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-
1OxBS_p010.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS_p010
FQ1=../trimmed/split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.1_val_1.fq.p010
FQ2=../trimmed/split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.2_val_2.fq.p010
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.1_val_1.fq.p010 -2
../trimmed/split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.2_val_2.fq.p010
java -jar $(which SortSam.jar) INPUT=J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-
1OxBS.1_val_1.fq.p010_bismark_bt2_pe.bam OUTPUT=J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-
1OxBS_p010_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS_p010.status

==> ./sanchari/j432/bismark_aligned/qscripts/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-
OxBS_p003.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS_p003
FQ1=../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.1_val_1.fq.p003
FQ2=../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.2_val_2.fq.p003
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.1_val_1.fq.p003 -2
../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.2_val_2.fq.p003
java -jar $(which SortSam.jar) INPUT=J432_BHKVCGBCXX_Lane2_GTGAAA_JD-
```

```
OxBS.1_val_1.fq.p003_bismark_bt2_pe.bam OUTPUT=J432_BHKVCGBCXX_Lane2_GTGAAA_JD-
OxBS_p003_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS_p003.status

==> ./sanchari/j432/bismark_aligned/qscripts/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-
BS_p007.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS_p007
FQ1=../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.1_val_1.fq.p007
FQ2=../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.2_val_2.fq.p007
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.1_val_1.fq.p007 -2
../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.2_val_2.fq.p007
java -jar $(which SortSam.jar) INPUT=J432_BHKVCGBCXX_Lane1_CTTGTA_JD-
BS.1_val_1.fq.p007_bismark_bt2_pe.bam OUTPUT=J432_BHKVCGBCXX_Lane1_CTTGTA_JD-
BS_p007_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS_p007.status

==> ./sanchari/j432/bismark_aligned/qscripts/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-
BS_p005.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS_p005
FQ1=../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.1_val_1.fq.p005
FQ2=../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.2_val_2.fq.p005
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.1_val_1.fq.p005 -2
../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.2_val_2.fq.p005
java -jar $(which SortSam.jar) INPUT=J432_BHKVCGBCXX_Lane1_CTTGTA_JD-
BS.1_val_1.fq.p005_bismark_bt2_pe.bam OUTPUT=J432_BHKVCGBCXX_Lane1_CTTGTA_JD-
```

```
BS_p005_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS_p005.status

==> ./sanchari/j432/bismark_aligned/qscripts/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-
BS_p011.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS_p011
FQ1=../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.1_val_1.fq.p011
FQ2=../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.2_val_2.fq.p011
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.1_val_1.fq.p011 -2
../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.2_val_2.fq.p011
java -jar $(which SortSam.jar) INPUT=J432_BHKVCGBCXX_Lane1_CTTGTA_JD-
BS.1_val_1.fq.p011_bismark_bt2_pe.bam OUTPUT=J432_BHKVCGBCXX_Lane1_CTTGTA_JD-
BS_p011_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS_p011.status

==> ./sanchari/j432/bismark_aligned/qscripts/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-
1OxBS_p002.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS_p002
FQ1=../trimmed/split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.1_val_1.fq.p002
FQ2=../trimmed/split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.2_val_2.fq.p002
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.1_val_1.fq.p002 -2
../trimmed/split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.2_val_2.fq.p002
java -jar $(which SortSam.jar) INPUT=J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-
1OxBS.1_val_1.fq.p002_bismark_bt2_pe.bam OUTPUT=J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-
1OxBS_p002_sorted.bam SO=coordinate CREATE_INDEX=true
```

```
echo "done" > J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS_p002.status

==> ./sanchari/j432/bismark_aligned/qscripts/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-
BS_p013.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS_p013
FQ1=../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.1_val_1.fq.p013
FQ2=../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.2_val_2.fq.p013
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.1_val_1.fq.p013 -2
../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.2_val_2.fq.p013
java -jar $(which SortSam.jar) INPUT=J432_BHKVCGBCXX_Lane1_CTTGTA_JD-
BS.1_val_1.fq.p013_bismark_bt2_pe.bam OUTPUT=J432_BHKVCGBCXX_Lane1_CTTGTA_JD-
BS_p013_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS_p013.status

==> ./sanchari/j432/bismark_aligned/qscripts/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-
BS_p015.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS_p015
FQ1=../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.1_val_1.fq.p015
FQ2=../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.2_val_2.fq.p015
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.1_val_1.fq.p015 -2
../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.2_val_2.fq.p015
java -jar $(which SortSam.jar) INPUT=J432_BHKVCGBCXX_Lane1_CTTGTA_JD-
BS.1_val_1.fq.p015_bismark_bt2_pe.bam OUTPUT=J432_BHKVCGBCXX_Lane1_CTTGTA_JD-
BS_p015_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS_p015.status
```

```
==> ./sanchari/j432/bismark_aligned/qscripts/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-
BS_p012.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS_p012
FQ1=../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.1_val_1.fq.p012
FQ2=../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.2_val_2.fq.p012
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.1_val_1.fq.p012 -2
../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.2_val_2.fq.p012
java -jar $(which SortSam.jar) INPUT=J432_BHKVCGBCXX_Lane1_CTTGTA_JD-
BS.1_val_1.fq.p012_bismark_bt2_pe.bam OUTPUT=J432_BHKVCGBCXX_Lane1_CTTGTA_JD-
BS_p012_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS_p012.status

==> ./sanchari/j432/bismark_aligned/qscripts/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-
1_BS_p013.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS_p013
FQ1=../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.1_val_1.fq.p013
FQ2=../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.2_val_2.fq.p013
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.1_val_1.fq.p013 -2
../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.2_val_2.fq.p013
java -jar $(which SortSam.jar) INPUT=J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-
1_BS.1_val_1.fq.p013_bismark_bt2_pe.bam OUTPUT=J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-
1_BS_p013_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS_p013.status
```

```
==> ./sanchari/j432/bismark_aligned/qscripts/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-
OxBS_p010.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS_p010
FQ1=../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.1_val_1.fq.p010
FQ2=../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.2_val_2.fq.p010
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.1_val_1.fq.p010 -2
../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.2_val_2.fq.p010
java -jar $(which SortSam.jar) INPUT=J432_BHKVCGBCXX_Lane2_GTGAAA_JD-
OxBS.1_val_1.fq.p010_bismark_bt2_pe.bam OUTPUT=J432_BHKVCGBCXX_Lane2_GTGAAA_JD-
OxBS_p010_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS_p010.status

==> ./sanchari/j432/bismark_aligned/qscripts/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-
1OxBS_p014.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS_p014
FQ1=../trimmed/split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.1_val_1.fq.p014
FQ2=../trimmed/split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.2_val_2.fq.p014
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.1_val_1.fq.p014 -2
../trimmed/split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.2_val_2.fq.p014
java -jar $(which SortSam.jar) INPUT=J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-
1OxBS.1_val_1.fq.p014_bismark_bt2_pe.bam OUTPUT=J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-
1OxBS_p014_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS_p014.status

==> ./sanchari/j432/bismark_aligned/qscripts/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-
```

```
1OxBS_p005.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS_p005
FQ1=../trimmed/split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.1_val_1.fq.p005
FQ2=../trimmed/split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.2_val_2.fq.p005
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.1_val_1.fq.p005 -2
../trimmed/split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.2_val_2.fq.p005
java -jar $(which SortSam.jar) INPUT=J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-
1OxBS.1_val_1.fq.p005_bismark_bt2_pe.bam OUTPUT=J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-
1OxBS_p005_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS_p005.status

==> ./sanchari/j432/bismark_aligned/qscripts/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-
1OxBS_p006.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS_p006
FQ1=../trimmed/split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.1_val_1.fq.p006
FQ2=../trimmed/split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.2_val_2.fq.p006
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.1_val_1.fq.p006 -2
../trimmed/split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.2_val_2.fq.p006
java -jar $(which SortSam.jar) INPUT=J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-
1OxBS.1_val_1.fq.p006_bismark_bt2_pe.bam OUTPUT=J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-
1OxBS_p006_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS_p006.status

==> ./sanchari/j432/bismark_aligned/qscripts/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-
1OxBS_p011.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS_p011
FQ1=../trimmed/split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.1_val_1.fq.p011
FQ2=../trimmed/split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.2_val_2.fq.p011
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.1_val_1.fq.p011 -2
../trimmed/split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.2_val_2.fq.p011
java -jar $(which SortSam.jar) INPUT=J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-
1OxBS.1_val_1.fq.p011_bismark_bt2_pe.bam OUTPUT=J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-
1OxBS_p011_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS_p011.status

==> ./sanchari/j432/bismark_aligned/qscripts/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-
BS_p003.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS_p003
FQ1=../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.1_val_1.fq.p003
FQ2=../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.2_val_2.fq.p003
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.1_val_1.fq.p003 -2
../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.2_val_2.fq.p003
java -jar $(which SortSam.jar) INPUT=J432_BHKVCGBCXX_Lane1_CTTGTA_JD-
BS.1_val_1.fq.p003_bismark_bt2_pe.bam OUTPUT=J432_BHKVCGBCXX_Lane1_CTTGTA_JD-
BS_p003_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS_p003.status

==> ./sanchari/j432/bismark_aligned/qscripts/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-
OxBS_p001.sh <==
#!/bin/bash
```

```
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS_p001
FQ1=../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.1_val_1.fq.p001
FQ2=../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.2_val_2.fq.p001
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.1_val_1.fq.p001 -2
../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.2_val_2.fq.p001
java -jar $(which SortSam.jar) INPUT=J432_BHKVCGBCXX_Lane2_GTGAAA_JD-
OxBS.1_val_1.fq.p001_bismark_bt2_pe.bam OUTPUT=J432_BHKVCGBCXX_Lane2_GTGAAA_JD-
OxBS_p001_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS_p001.status

==> ./sanchari/j432/bismark_aligned/qscripts/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-
OxBS_p005.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS_p005
FQ1=../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.1_val_1.fq.p005
FQ2=../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.2_val_2.fq.p005
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.1_val_1.fq.p005 -2
../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.2_val_2.fq.p005
java -jar $(which SortSam.jar) INPUT=J432_BHKVCGBCXX_Lane2_GTGAAA_JD-
OxBS.1_val_1.fq.p005_bismark_bt2_pe.bam OUTPUT=J432_BHKVCGBCXX_Lane2_GTGAAA_JD-
OxBS_p005_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS_p005.status

==> ./sanchari/j432/bismark_aligned/qscripts/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-
1OxBS_p007.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS_p007
FQ1=../trimmed/split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.1_val_1.fq.p007
FQ2=../trimmed/split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.2_val_2.fq.p007
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.1_val_1.fq.p007 -2
../trimmed/split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.2_val_2.fq.p007
java -jar $(which SortSam.jar) INPUT=J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-
1OxBS.1_val_1.fq.p007_bismark_bt2_pe.bam OUTPUT=J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-
1OxBS_p007_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS_p007.status

==> ./sanchari/j432/bismark_aligned/qscripts/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-
OxBS_p004.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS_p004
FQ1=../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.1_val_1.fq.p004
FQ2=../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.2_val_2.fq.p004
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.1_val_1.fq.p004 -2
../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.2_val_2.fq.p004
java -jar $(which SortSam.jar) INPUT=J432_BHKVCGBCXX_Lane2_GTGAAA_JD-
OxBS.1_val_1.fq.p004_bismark_bt2_pe.bam OUTPUT=J432_BHKVCGBCXX_Lane2_GTGAAA_JD-
OxBS_p004_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS_p004.status

==> ./sanchari/j432/bismark_aligned/qscripts/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-
OxBS_p000.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS_p000
FQ1=../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.1_val_1.fq.p000
FQ2=../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.2_val_2.fq.p000
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.1_val_1.fq.p000 -2
../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.2_val_2.fq.p000
java -jar $(which SortSam.jar) INPUT=J432_BHKVCGBCXX_Lane2_GTGAAA_JD-
OxBS.1_val_1.fq.p000_bismark_bt2_pe.bam OUTPUT=J432_BHKVCGBCXX_Lane2_GTGAAA_JD-
OxBS_p000_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS_p000.status

==> ./sanchari/j432/bismark_aligned/qscripts/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-
1_BS_p004.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS_p004
FQ1=../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.1_val_1.fq.p004
FQ2=../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.2_val_2.fq.p004
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.1_val_1.fq.p004 -2
../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.2_val_2.fq.p004
java -jar $(which SortSam.jar) INPUT=J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-
1_BS.1_val_1.fq.p004_bismark_bt2_pe.bam OUTPUT=J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-
1_BS_p004_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS_p004.status

==> ./sanchari/j432/bismark_aligned/qscripts/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-
1OxBS_p003.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS_p003
FQ1=../trimmed/split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.1_val_1.fq.p003
FQ2=../trimmed/split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.2_val_2.fq.p003
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.1_val_1.fq.p003 -2
../trimmed/split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.2_val_2.fq.p003
java -jar $(which SortSam.jar) INPUT=J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-
1OxBS.1_val_1.fq.p003_bismark_bt2_pe.bam OUTPUT=J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-
1OxBS_p003_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS_p003.status

==> ./sanchari/j432/bismark_aligned/qscripts/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-
BS_p010.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS_p010
FQ1=../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.1_val_1.fq.p010
FQ2=../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.2_val_2.fq.p010
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.1_val_1.fq.p010 -2
../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.2_val_2.fq.p010
java -jar $(which SortSam.jar) INPUT=J432_BHKVCGBCXX_Lane1_CTTGTA_JD-
BS.1_val_1.fq.p010_bismark_bt2_pe.bam OUTPUT=J432_BHKVCGBCXX_Lane1_CTTGTA_JD-
BS_p010_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS_p010.status

==> ./sanchari/j432/bismark_aligned/qscripts/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-
BS_p004.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS_p004
FQ1=../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.1_val_1.fq.p004
FQ2=../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.2_val_2.fq.p004
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.1_val_1.fq.p004 -2
../trimmed/split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.2_val_2.fq.p004
java -jar $(which SortSam.jar) INPUT=J432_BHKVCGBCXX_Lane1_CTTGTA_JD-
BS.1_val_1.fq.p004_bismark_bt2_pe.bam OUTPUT=J432_BHKVCGBCXX_Lane1_CTTGTA_JD-
BS_p004_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS_p004.status

==> ./sanchari/j432/bismark_aligned/qscripts/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-
1_BS_p001.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS_p001
FQ1=../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.1_val_1.fq.p001
FQ2=../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.2_val_2.fq.p001
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.1_val_1.fq.p001 -2
../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.2_val_2.fq.p001
java -jar $(which SortSam.jar) INPUT=J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-
1_BS.1_val_1.fq.p001_bismark_bt2_pe.bam OUTPUT=J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-
1_BS_p001_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS_p001.status

==> ./sanchari/j432/bismark_aligned/qscripts/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-
1_BS_p011.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS_p011
FQ1=../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.1_val_1.fq.p011
FQ2=../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.2_val_2.fq.p011
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.1_val_1.fq.p011 -2
../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.2_val_2.fq.p011
java -jar $(which SortSam.jar) INPUT=J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-
1_BS.1_val_1.fq.p011_bismark_bt2_pe.bam OUTPUT=J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-
1_BS_p011_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS_p011.status

==> ./sanchari/j432/bismark_aligned/qscripts/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-
OxBS_p007.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS_p007
FQ1=../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.1_val_1.fq.p007
FQ2=../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.2_val_2.fq.p007
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.1_val_1.fq.p007 -2
../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.2_val_2.fq.p007
java -jar $(which SortSam.jar) INPUT=J432_BHKVCGBCXX_Lane2_GTGAAA_JD-
OxBS.1_val_1.fq.p007_bismark_bt2_pe.bam OUTPUT=J432_BHKVCGBCXX_Lane2_GTGAAA_JD-
OxBS_p007_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS_p007.status

==> ./sanchari/j432/bismark_aligned/qscripts/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-
1_BS_p007.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS_p007
FQ1=../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.1_val_1.fq.p007
FQ2=../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.2_val_2.fq.p007
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.1_val_1.fq.p007 -2
../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.2_val_2.fq.p007
java -jar $(which SortSam.jar) INPUT=J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-
1_BS.1_val_1.fq.p007_bismark_bt2_pe.bam OUTPUT=J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-
1_BS_p007_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS_p007.status

==> ./sanchari/j432/bismark_aligned/qscripts/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-
OxBS_p002.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS_p002
FQ1=../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.1_val_1.fq.p002
FQ2=../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.2_val_2.fq.p002
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.1_val_1.fq.p002 -2
../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.2_val_2.fq.p002
java -jar $(which SortSam.jar) INPUT=J432_BHKVCGBCXX_Lane2_GTGAAA_JD-
OxBS.1_val_1.fq.p002_bismark_bt2_pe.bam OUTPUT=J432_BHKVCGBCXX_Lane2_GTGAAA_JD-
OxBS_p002_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS_p002.status

==> ./sanchari/j432/bismark_aligned/qscripts/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-
OxBS_p008.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS_p008
FQ1=../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.1_val_1.fq.p008
FQ2=../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.2_val_2.fq.p008
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.1_val_1.fq.p008 -2
../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.2_val_2.fq.p008
java -jar $(which SortSam.jar) INPUT=J432_BHKVCGBCXX_Lane2_GTGAAA_JD-
OxBS.1_val_1.fq.p008_bismark_bt2_pe.bam OUTPUT=J432_BHKVCGBCXX_Lane2_GTGAAA_JD-
OxBS_p008_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS_p008.status

==> ./sanchari/j432/bismark_aligned/qscripts/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-
OxBS_p009.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS_p009
FQ1=../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.1_val_1.fq.p009
FQ2=../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.2_val_2.fq.p009
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.1_val_1.fq.p009 -2
../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.2_val_2.fq.p009
java -jar $(which SortSam.jar) INPUT=J432_BHKVCGBCXX_Lane2_GTGAAA_JD-
OxBS.1_val_1.fq.p009_bismark_bt2_pe.bam OUTPUT=J432_BHKVCGBCXX_Lane2_GTGAAA_JD-
OxBS_p009_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS_p009.status

==> ./sanchari/j432/bismark_aligned/qscripts/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-
1OxBS_p004.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu
```

```
module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS_p004
FQ1=../trimmed/split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.1_val_1.fq.p004
FQ2=../trimmed/split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.2_val_2.fq.p004
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.1_val_1.fq.p004 -2
../trimmed/split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.2_val_2.fq.p004
java -jar $(which SortSam.jar) INPUT=J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-
1OxBS.1_val_1.fq.p004_bismark_bt2_pe.bam OUTPUT=J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-
1OxBS_p004_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS_p004.status

==> ./sanchari/j432/bismark_aligned/qscripts/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-
1_BS_p003.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS_p003
FQ1=../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.1_val_1.fq.p003
FQ2=../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.2_val_2.fq.p003
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.1_val_1.fq.p003 -2
../trimmed/split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.2_val_2.fq.p003
java -jar $(which SortSam.jar) INPUT=J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-
1_BS.1_val_1.fq.p003_bismark_bt2_pe.bam OUTPUT=J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-
1_BS_p003_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS_p003.status

==> ./sanchari/j432/bismark_aligned/qscripts/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-
OxBS_p015.sh <==
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu
```

```
module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS_p015
FQ1=../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.1_val_1.fq.p015
FQ2=../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.2_val_2.fq.p015
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 /home/kpradha1/projects/hg19/ -1
../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.1_val_1.fq.p015 -2
../trimmed/split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.2_val_2.fq.p015
java -jar $(which SortSam.jar) INPUT=J432_BHKVCGBCXX_Lane2_GTGAAA_JD-
OxBS.1_val_1.fq.p015_bismark_bt2_pe.bam OUTPUT=J432_BHKVCGBCXX_Lane2_GTGAAA_JD-
OxBS_p015_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS_p015.status

==> ./sanchari/j432/bismark_aligned/submitClipOverlaps_mycomp.sh <==
#qsub clipOverlaps.sh J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-
1_BS_merged_sorted_mdup_nsorted.bam
#qsub clipOverlaps.sh J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-
1OxBS_merged_sorted_mdup_nsorted.bam
#qsub clipOverlaps.sh J432_BHKVCGBCXX_Lane1_CTTGTA_JD-
BS_merged_sorted_mdup_nsorted.bam
#qsub clipOverlaps.sh J432_BHKVCGBCXX_Lane2_GTGAAA_JD-
OxBS_merged_sorted_mdup_nsorted.bam

./clipOverlaps_mycomp.sh J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-
1_BS_merged_sorted_mdup.bam
./clipOverlaps_mycomp.sh J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-
1OxBS_merged_sorted_mdup.bam
./clipOverlaps_mycomp.sh J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS_merged_sorted_mdup.bam
./clipOverlaps_mycomp.sh J432_BHKVCGBCXX_Lane2_GTGAAA_JD-
OxBS_merged_sorted_mdup.bam


==> ./sanchari/j432/bismark_aligned/clipOverlaps_mycomp.sh <==
bam=$1
bname="${bam%.*}"

bam clipOverlap --in $bam --out ${bname}_clip.bam
samtools index ${bname}_clip.bam


==> ./sanchari/j432/bismark_aligned/submitMD.sh <==
qsub markDuplicates.sh J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS_merged_sorted.bam
qsub markDuplicates.sh J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS_merged_sorted.bam
qsub markDuplicates.sh J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS_merged_sorted.bam
qsub markDuplicates.sh J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS_merged_sorted.bam

==> ./sanchari/j432/bismark_aligned/submitMerges.sh <==

qsub mergeBams.sh J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS_merged_sorted.bam
J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS_p*_sorted.bam
qsub mergeBams.sh J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS_merged_sorted.bam
J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS_p*_sorted.bam
qsub mergeBams.sh J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS_merged_sorted.bam
J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS_p*_sorted.bam
qsub mergeBams.sh J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS_merged_sorted.bam
```

```
J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS_p*_sorted.bam



==> ./sanchari/j432/bismark_aligned/mergeBams.sh <==
#!/bin/bash
#$ -cwd
#$ -j n
#$ -l h_vmem=4G
#$ -pe smp 4
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu
#usage:
#qsub mergeBams.sh outfile.bam infile1.bam infile2.bam infile3.bam
#qsub mergeBams.sh hpne_bs.bam *hpne*bs*sorted*bam
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

#make a folder for storing intermediate resulst
#otherwise, might run out of space on default /tmp
mkdir -p tmp
chmod 777 tmp

#first argument is the name of the output file
outbam=$1
shift
#the rest of the arguments are the bam files we want to merge

java -jar $(which MergeSamFiles.jar) \
    `echo $@ | sed 's/\(\S\+\)/I=&/g'`\
    ASSUME_SORTED=true\
    TMP_DIR=tmp\
    O=$outbam


==> ./sanchari/j432/bismark_aligned/reportMeth.sh <==
#!/bin/bash
#$ -cwd
#$ -j n
#$ -l h_vmem=4G
#$ -pe smp 1
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu
module load samtools/1.2/gcc.4.4.7
module load python/2.7.8/gcc.4.4.7

REF=/home/kpradha1/projects/hg19/hg19.fa
bam=$1
bname="${bam%.*}"


samtools mpileup -d1000000 -Q0 -B -f $REF $bam | \
  ./mpileup2methylation.py -i - >| ${bname}.bedGraph
```

```
==> ./sanchari/j432/bismark_aligned/mpileup2methylation.py <==
#!/usr/bin/env python

import sys
import argparse

parser = argparse.ArgumentParser(description= """
DESCRIPTION
    Extract methylation calls from samtools mpileup. Only positions with non-zero
count
    are printed. Output has is bedGraph with additional columns:
    <chrom>  <pos-1>  <pos>  <pct meth'd>  <cnt methylated>  <tot count>  <strand>

   Memo: bedGraph is 0-based, so if the first base of chr1 is C it will have
position: `chrom 0 1 ... +`

EXAMPLE & RECIPES
    samtools mpileup -d100000 -Q0 -B -l mm9.allcpg.bed.gz -f mmu.fa myreads.bam |
mpileup2methylation.py -i -

    If you want to get only the Cs in a certain context (e.g. CpG) you need to pass
    a bed file to mpileup where these positions are. For example, to get all the
    CpGs you could use:
    fastaRegexFinder.py -f mmu.fa -r CG --noreverse > mm9.allcpg.bed
    (Possibly pipe to `cut -f1-3 | gzip` to make the file smaller)

TESTED:
   14/02/2013: The output of mpileup2methylation.py is almost (but not) identical
   to bismark_methylation_extractor (files *.genome-wide_CX|CpG_report.txt). Tested
   on synthetic strands single end, and paired-end whole genome. Differences not
investigated
   further but should affect ~1 call in 1000-10000.

""", formatter_class= argparse.RawTextHelpFormatter)

parser.add_argument('--input', '-i',
                   required= True,
                   help='''Input file as produced by samtools mpileup. Use - to
read from stdin.
                   ''')

parser.add_argument('--outfmt', '-f',
                   required= False,
                   choices= ['bedgraph', 'bismark'],
                   default= 'bedgraph',
                   help='''Output format: bedgraph (default) or bismark.
'bismark' output has columns: chrom, pos, strand, count meth'd, count unmeth'd.
The first 5 column of the genome-wide cytosine report produced by
bismark_methylation_extractor should be nearly identical to 'bismark' format.
                   ''')

args= parser.parse_args()
# -------------------------------------------------------------------------------

def cleanCallString(bases):
   """Removes from the call string in mpileup (5th column) the ^ character and
   the char next to it. Note TODO: Account for insertions/deletions.
   bases:
      String of read bases (5th column of mpileup)
   Return:
      Same string as bases but with ^ and the following char removed
   Example:
      bases= '^A...,,.,.,...,,,.,....^k.'
      cleanCallString(bases) >>> '...,,.,.,...,,,.,.....'
   """
   callString= ''
   skip= False
   for x in bases:
      if x  == '^':
         skip= True
      elif skip:
         skip= False
      else:
         callString += x
   return(callString)

def pileup2methylation(chrom, pos, callString, ref, outfmt):
   """Count methylated and unmethylated calls.
   chrom, pos:
      Chromosome (string) and position (int) on the pileup
   callString:
      String of bases obtained by cleanCallString
   ref:
      Reference base as obtained from 3nd column of mpileup

   Memo: mpileup input looks like this:
   chr7    3002089 C       2       .^~.    IA
   chr7    3002090 G       2       ..      HE
   chr7    3002114 C       2       ..      HE

   """
#   ref= mpileup[2]
#   callstring= mpileup[4]
   cnt_M= 0 ## Count methylated
   cnt_m= 0 ## Count unmethylated

   if ref.upper() == 'G':
      strand= '-'
      cnt_M += callString.count(',')
      cnt_m += callString.count('a')
   elif ref.upper() == 'C':
      strand= '+'
      cnt_M += callString.count('.')
      cnt_m += callString.count('T')
   else:
      return(None)
   if (cnt_m + cnt_M) == 0:
      return(None)
   ## Use for first version of methylation2mpileup.py
   ## methList= [mpileup[0], mpileup[1], strand, str(cnt_M), str(cnt_m), ref]
   if outfmt == 'bismark':
      methList= [chrom, str(pos), strand, str(cnt_M), str(cnt_m)]
   elif outfmt == 'bedgraph':
      totreads= cnt_M + cnt_m
      methList= [chrom, str(pos-1), str(pos), 
str(round(100*(float(cnt_M)/totreads), 4)), str(cnt_M), str(totreads), strand]
   else:
      sys.exit('Unexpected keyword for outfmt "%s"' %(outfmt))
   return(methList)
# --------------------------------------------------------------------------------

if args.input == '-':
   fin= sys.stdin
else:
   fin= open(args.input)

for line in fin:
   line= line.strip().split('\t')
   callString= cleanCallString(line[4])
   methList= pileup2methylation(chrom= line[0], pos= int(line[1]), callString=
callString, ref= line[2], outfmt= args.outfmt)
   if methList is None:
      pass
   else:
      try:
         print('\t'.join(methList))
      except IOError as e:
         break
fin.close()
sys.exit()
```

```
==> ./sanchari/j432/bismark_aligned/makeBismarkBatch.sh <==
#!/bin/bash
#constructs the qsub script to process a chunk of reads
#writes the script to stdout
#pipe it into qsub if you want to submit it
#
#usage:
#makeBismarkBatch.sh hpneBs reads1.fastq.gz reads2.fastq.gz | qsub

SNAME=$1
FQ1=$2
FQ2=$3
REF=/home/kpradha1/projects/hg19/

cat <<PROGRAM_END
#!/bin/bash
#$ -cwd
#$ -N bismark_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 5
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

SNAME=$1
FQ1=$2
FQ2=$3
```

```
REF=/home/kpradha1/projects/hg19/
bismark --multicore 1 --bowtie2 -p 4 $REF -1 $FQ1 -2 $FQ2
java -jar \$(which SortSam.jar) INPUT=${FQ1##*/}_bismark_bt2_pe.bam OUTPUT=$
{SNAME}_sorted.bam SO=coordinate CREATE_INDEX=true
echo "done" > $SNAME.status
PROGRAM_END


==> ./sanchari/j432/bismark_aligned/submitReportMeths.sh <==

#accidently name the bams .bam.bam
#fix that here
#ls *.bam.bam | sed 's/\(.*\.bam\)\.bam/mv & \1/'

#qsub reportMeth.sh J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS_merged_sorted.bam
#qsub reportMeth.sh J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS_merged_sorted.bam
#qsub reportMeth.sh J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS_merged_sorted.bam
#qsub reportMeth.sh J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS_merged_sorted.bam

#qsub sortBam.sh J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-
1_BS_merged_sorted_mdup_nsorted_clip.bam
#qsub sortBam.sh J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-
1OxBS_merged_sorted_mdup_nsorted_clip.bam
#qsub sortBam.sh J432_BHKVCGBCXX_Lane1_CTTGTA_JD-
BS_merged_sorted_mdup_nsorted_clip.bam
#qsub sortBam.sh J432_BHKVCGBCXX_Lane2_GTGAAA_JD-
OxBS_merged_sorted_mdup_nsorted_clip.bam

qsub reportMeth.sh J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-
1_BS_merged_sorted_mdup_clip.bam
qsub reportMeth.sh J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-
1OxBS_merged_sorted_mdup_clip.bam
qsub reportMeth.sh J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS_merged_sorted_mdup_clip.bam
qsub reportMeth.sh J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS_merged_sorted_mdup_clip.bam







==> ./sanchari/j432/bismark_aligned/testSetup1.sh <==
for fq1 in `ls ../trimmed/split/*1_val_1*`;
do
    fq2=`echo $fq1 | sed 's/\(.*\)1_val_1\(.*\)/\12_val_2\2/'`
    pname=`echo $fq1 | sed 's/.*\/\(.*\).1_val_1.fq.\(p.*\)/\1_\2/'`
    #echo $fq1
    #echo $fq2
    #echo $pname
    ./makeBismarkBatch.sh $pname $fq1 $fq2 > ${pname}.sh
done

==> ./sanchari/j432/bismark_aligned/submitClipOverlaps.sh <==
#qsub clipOverlaps.sh J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-
1_BS_merged_sorted_mdup_nsorted.bam
#qsub clipOverlaps.sh J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-
1OxBS_merged_sorted_mdup_nsorted.bam
#qsub clipOverlaps.sh J432_BHKVCGBCXX_Lane1_CTTGTA_JD-
```

```
BS_merged_sorted_mdup_nsorted.bam
#qsub clipOverlaps.sh J432_BHKVCGBCXX_Lane2_GTGAAA_JD-
OxBS_merged_sorted_mdup_nsorted.bam

qsub clipOverlaps.sh J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS_merged_sorted_mdup.bam
qsub clipOverlaps.sh J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS_merged_sorted_mdup.bam
qsub clipOverlaps.sh J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS_merged_sorted_mdup.bam
qsub clipOverlaps.sh J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS_merged_sorted_mdup.bam


==> ./sanchari/j432/bismark_aligned/submitNameSorts.sh <==
qsub nameSort.sh J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS_merged_sorted_mdup.bam
qsub nameSort.sh J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS_merged_sorted_mdup.bam
qsub nameSort.sh J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS_merged_sorted_mdup.bam
qsub nameSort.sh J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS_merged_sorted_mdup.bam


==> ./sanchari/j432/test2/batch2.sh <==
#!/bin/bash
#$ -cwd
#$ -j n
#$ -l h_vmem=15G
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu
module load bismark/0.14.5
module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

FNAME=J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS
SNAME=hpne_bs
FQ1=${FNAME}.1_val_1.fq.gz
FQ2=${FNAME}.2_val_2.fq.gz
REF=/home/kpradha1/projects/hg19/
bismark --bowtie2 $REF -1 $FQ1 -2 $FQ2  --basename $SNAME
java -jar $(which SortSam.jar) INPUT=${SNAME}_pe.bam OUTPUT=${SNAME}_sorted.bam
SO=coordinate
java -jar $(which BuildBamIndex.jar) INPUT=${SNAME}_sorted.bam
java -jar $(which MarkDuplicates.jar) INPUT=${SNAME}_sorted.bam OUTPUT=$
{SNAME}_sorted_md.bam METRICS_FILE=${SNAME}_mdMetric.txt
java -jar $(which BuildBamIndex.jar) INPUT=${SNAME}_sorted_md.bam



==> ./sanchari/j432/test2/do_bismark1.sh <==
#!/bin/bash
#$ -cwd
#$ -j n
#$ -l h_vmem=15G
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu
#
# do_bismark.sh
# Brent Calder <brent.calder@einstein.yu.edu>
#
# Convienience script to take fastq.gz input, run trim_galore,
# fastqc, split input, run Bismark, then merge, sort, and index
# output into BAM.
#


set -o errexit -o nounset -o allexport -o pipefail

# user options

GENOME_FOLDER=/home/kpradha1/projects/hg19/
GENOME=mm9.fa # not used, this is the original
# genome name at ../fasta, relative to GENOME_FOLDER

TRIM_GALORE_OPTS="" # mainly for new adapter sequence or clips
FASTQC_OPTS="" # optional FASTQC options
BISMARK_OPTS="" # see documentation, 'non-directional' set below (DIRECTIONS)
BOWTIE2_OPTS="" # bowtie2 options for bismark

# number of reads per split work file
READS_PER_FASTQ=4000000
# DIRECTIONS must be 2 or 4 (eg. 4 = "--non_directional")
DIRECTIONS=4
# NTHREADS is the number of threads per DIRECTION
NTHREADS=2
# DIRECTIONS*THREADS = num_proc

########################
# DO NOT EDIT BELOW HERE

USAGE="Usage:\t\tqsub do_bismark.sh forward_read.fastq.gz
[reverse_read.fastq.gz]\n"
ADMON="reads must be in format: name.fastq.gz\n"
PAIRED=0
REVERSE_READ=""
MYID=$$-${RANDOM}

if [ "$DIRECTIONS" -eq "4" ]; then
    BISMARK_OPTS="${BISMARK_OPTS} --non_directional"
else
    # default back to directional
    DIRECTIONS=2
fi

if [ "$#" -eq "2" ]; then
    PAIRED=1
    REVERSE_READ=$2
    REVERSE=`basename ${REVERSE_READ/.fastq.gz/}`
    TRIM_GALORE_OPTS="${TRIM_GALORE_OPTS} --paired"
    if [ "${#REVERSE_READ}" -eq "${#REVERSE}" ]; then
        USAGE="${USAGE}${ADMON}"
        echo -e "$USAGE" >&2
        exit 2
    fi
else
    if [ "$#" -ne "1" ]; then
        echo -e "$USAGE" >&2
        exit 1
    fi
fi
```

```
FORWARD_READ=$1
FORWARD=`basename ${FORWARD_READ/.fastq.gz/}`

if [ "${#FORWARD_READ}" -eq "${#FORWARD}" ]; then
    USAGE="${USAGE}${ADMON}"
    echo -e "$USAGE" >&2
    exit 2
fi

##################
# REQUIRED MODULES
##################
#source /apps1/modules/init/bash

module load trim_galore/0.3.7/gcc.4.4.7
module load bismark/0.14.5
module load FastQC/0.11.4/java.1.8.0_20
module load picard-tools/1.92/java.1.8.0_20

# might be necessary to set up qsub
if [ -e /etc/profile.d/sge-binaries.sh ]; then
    source /etc/profile.d/sge-binaries.sh
fi

# output dirs
mkdir -p fastqc split bismark trimmed tmp

# trim illumina adapters and -q 20 by default
trim_galore --dont_gzip -o trimmed --fastqc --fastqc_args "${FASTQC_OPTS} -o
./fastqc" $TRIM_GALORE_OPTS $FORWARD_READ $REVERSE_READ

# split files based on READS_PER_FASTQ
if [ "$PAIRED" -eq "0" ]; then
    split -a 3 -l $((${READS_PER_FASTQ}*4)) trimmed/${FORWARD}_trimmed.fq split/$
{FORWARD}.split.
else
    if [ "$PAIRED" -eq "1" ]; then
        split -a 3 -l $((${READS_PER_FASTQ}*4)) trimmed/${FORWARD}_val_1.fq split/$
{FORWARD}.split.
        split -a 3 -l $((${READS_PER_FASTQ}*4)) trimmed/${REVERSE}_val_2.fq split/$
{REVERSE}.split.
    fi
fi

# run bismark on the split files
for x in `ls -1 split/${FORWARD}.split.*`; do
    READSTR=${x}
    suf=${x/*split./}
    if [ "$PAIRED" -eq "1" ]; then
        second=split/${REVERSE}.split.${suf}
        READSTR="-1 ${x} -2 ${second}"
    fi
    qsub -cwd -l p=$((${NTHREADS}*${DIRECTIONS})) -l mf=30g -V -N bismark-${MYID}
-j y -b y bismark --temp_dir tmp -o bismark --bowtie2 -p $((${NTHREADS}*$
{DIRECTIONS})) $BISMARK_OPTS $BOWTIE2_OPTS $GENOME_FOLDER $READSTR
done

# use picard to merge, sort, create bam and index split SAM upon completion.
SAMS=""
```

```
if [ "$PAIRED" -eq "0" ]; then
    SAMS=`ls -1 split/${FORWARD}.split.* | sed 's|^split|I=bismark|g' | sed 's/
$/_bismark_bt2.sam/g' | tr "\n" " "`
else
    SAMS=`ls -1 split/${FORWARD}.split.* | sed 's|^split|I=bismark|g' | sed 's/
$/_bismark_bt2_pe.sam/g' | tr "\n" " "`
fi
BAM=${FORWARD}.bam

qsub -S /bin/bash -hold_jid bismark-${MYID} -cwd -V -l mf=4g -j y -N merge -b y
java -jar $PICARD_ROOT/MergeSamFiles.jar $SAMS O=$BAM AS=false SO=coordinate
TMP_DIR=./tmp CREATE_INDEX=true


==> ./sanchari/j432/trimmed/splitfiles.sh <==
#split each file into 40M line chunks


NL=40000000
for fq in `ls *.fq.gz`;
do
    nam=`basename $fq .gz`
    #get name without extension
    #echo $fq
    #echo $nam
    echo "split <(zcat $fq) -d -a 3 -l $NL split/${nam}.pt"
done >| split1.sh
chmod +x split1.sh




==> ./sanchari/j432/trimmed/dlfiles1.sh <==
wget http://waspsystem.einstein.yu.edu/wsfile/get/file/c62d758b-0c20-41b8-a64d-
b84f575f8449/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.1_val_1.fq.gz
wget http://waspsystem.einstein.yu.edu/wsfile/get/file/b68c7385-632e-45c8-ae31-
00e99c7eb8c0/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.2_val_2.fq.gz
wget http://waspsystem.einstein.yu.edu/wsfile/get/file/71c87f75-ad4b-4624-b304-
6b3a572915f3/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.1_val_1.fq.gz
wget http://waspsystem.einstein.yu.edu/wsfile/get/file/a9768b55-8443-4df6-aa3a-
4690eb498630/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.2_val_2.fq.gz
wget http://waspsystem.einstein.yu.edu/wsfile/get/file/4766ecab-cfb5-4159-9a74-
9773771ffa29/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.1_val_1.fq.gz
wget http://waspsystem.einstein.yu.edu/wsfile/get/file/9b8c42f3-40ad-4301-ac1d-
1e463f8ecd7d/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.2_val_2.fq.gz
wget http://waspsystem.einstein.yu.edu/wsfile/get/file/13acec40-db3f-4de3-ba13-
394e8c1a4455/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.1_val_1.fq.gz
wget http://waspsystem.einstein.yu.edu/wsfile/get/file/5619af56-20ea-4324-aa98-
ccb443039dbd/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.2_val_2.fq.gz


==> ./sanchari/j432/trimmed/split1.sh <==
split <(zcat J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.1_val_1.fq.gz) -d -a 3 -l
40000000 split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.1_val_1.fq.p
split <(zcat J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.2_val_2.fq.gz) -d -a 3 -l
40000000 split/J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS.2_val_2.fq.p
split <(zcat J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.1_val_1.fq.gz) -d -a 3 -l
40000000 split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.1_val_1.fq.p
split <(zcat J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.2_val_2.fq.gz) -d -a 3 -l
```

```
40000000 split/J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS.2_val_2.fq.p
split <(zcat J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.1_val_1.fq.gz) -d -a 3 -l 40000000
split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.1_val_1.fq.p
split <(zcat J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.2_val_2.fq.gz) -d -a 3 -l 40000000
split/J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS.2_val_2.fq.p
split <(zcat J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.1_val_1.fq.gz) -d -a 3 -l
40000000 split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.1_val_1.fq.p
split <(zcat J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.2_val_2.fq.gz) -d -a 3 -l
40000000 split/J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS.2_val_2.fq.p

==> ./sanchari/mpileup2methylation.py <==
#!/usr/bin/env python

import sys
import argparse

parser = argparse.ArgumentParser(description= """
DESCRIPTION
    Extract methylation calls from samtools mpileup. Only positions with non-zero
count
    are printed. Output has is bedGraph with additional columns:
    <chrom>  <pos-1>  <pos>  <pct meth'd>  <cnt methylated>  <tot count>  <strand>

   Memo: bedGraph is 0-based, so if the first base of chr1 is C it will have
position: `chrom 0 1 ... +`

EXAMPLE & RECIPES
    samtools mpileup -d100000 -Q0 -B -l mm9.allcpg.bed.gz -f mmu.fa myreads.bam |
mpileup2methylation.py -i -

    If you want to get only the Cs in a certain context (e.g. CpG) you need to pass
    a bed file to mpileup where these positions are. For example, to get all the
    CpGs you could use:
    fastaRegexFinder.py -f mmu.fa -r CG --noreverse > mm9.allcpg.bed
    (Possibly pipe to `cut -f1-3 | gzip` to make the file smaller)

TESTED:
   14/02/2013: The output of mpileup2methylation.py is almost (but not) identical
   to bismark_methylation_extractor (files *.genome-wide_CX|CpG_report.txt). Tested
   on synthetic strands single end, and paired-end whole genome. Differences not
investigated
   further but should affect ~1 call in 1000-10000.

""", formatter_class= argparse.RawTextHelpFormatter)

parser.add_argument('--input', '-i',
                   required= True,
                   help='''Input file as produced by samtools mpileup. Use - to
read from stdin.
                   ''')

parser.add_argument('--outfmt', '-f',
                   required= False,
                   choices= ['bedgraph', 'bismark'],
                   default= 'bedgraph',
                   help='''Output format: bedgraph (default) or bismark.
'bismark' output has columns: chrom, pos, strand, count meth'd, count unmeth'd.
The first 5 column of the genome-wide cytosine report produced by
bismark_methylation_extractor should be nearly identical to 'bismark' format.
```

```
                   ''')

args= parser.parse_args()
# ------------------------------------------------------------------------------

def cleanCallString(bases):
   """Removes from the call string in mpileup (5th column) the ^ character and
   the char next to it. Note TODO: Account for insertions/deletions.
   bases:
      String of read bases (5th column of mpileup)
   Return:
      Same string as bases but with ^ and the following char removed
   Example:
      bases= '^A....,,.,.,...,,,.,....^k.'
      cleanCallString(bases) >>> '....,,.,.,...,,,.,.....'
   """
   callString= ''
   skip= False
   for x in bases:
      if x  == '^':
         skip= True
      elif skip:
         skip= False
      else:
         callString += x
   return(callString)

def pileup2methylation(chrom, pos, callString, ref, outfmt):
   """Count methylated and unmethylated calls.
   chrom, pos:
      Chromosome (string) and position (int) on the pileup
   callString:
      String of bases obtained by cleanCallString
   ref:
      Reference base as obtained from 3nd column of mpileup

   Memo: mpileup input looks like this:
   chr7    3002089 C       2       .^~.    IA
   chr7    3002090 G       2       ..      HE
   chr7    3002114 C       2       ..      HE

   """
#   ref= mpileup[2]
#   callstring= mpileup[4]
   cnt_M= 0 ## Count methylated
   cnt_m= 0 ## Count unmethylated

   if ref.upper() == 'G':
      strand= '-'
      cnt_M += callString.count(',')
      cnt_m += callString.count('a')
   elif ref.upper() == 'C':
      strand= '+'
      cnt_M += callString.count('.')
      cnt_m += callString.count('T')
   else:
      return(None)
   if (cnt_m + cnt_M) == 0:
      return(None)
   ## Use for first version of methylation2mpileup.py
   ## methList= [mpileup[0], mpileup[1], strand, str(cnt_M), str(cnt_m), ref]
   if outfmt == 'bismark':
      methList= [chrom, str(pos), strand, str(cnt_M), str(cnt_m)]
   elif outfmt == 'bedgraph':
      totreads= cnt_M + cnt_m
      methList= [chrom, str(pos-1), str(pos),
str(round(100*(float(cnt_M)/totreads), 4)), str(cnt_M), str(totreads), strand]
   else:
      sys.exit('Unexpected keyword for outfmt "%s"' %(outfmt))
   return(methList)
# ------------------------------------------------------------------------------

if args.input == '-':
   fin= sys.stdin
else:
   fin= open(args.input)

for line in fin:
   line= line.strip().split('\t')
   callString= cleanCallString(line[4])
   methList= pileup2methylation(chrom= line[0], pos= int(line[1]), callString=
callString, ref= line[2], outfmt= args.outfmt)
   if methList is None:
      pass
   else:
      try:
         print('\t'.join(methList))
      except IOError as e:
         break
fin.close()
sys.exit()
```

```
==> ./sanchari/j515/markDuplicates.sh <==
#!/bin/bash
#$ -cwd
#$ -j n
#$ -l h_vmem=4G
#$ -pe smp 4
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

bam=$1
bname="${bam%.*}"

java -jar $(which BuildBamIndex.jar) INPUT=${bam}
java -jar $(which MarkDuplicates.jar) INPUT=${bam} OUTPUT=${bname}_mdup.bam
METRICS_FILE=${bname}_mdMetric.txt
java -jar $(which BuildBamIndex.jar) INPUT=${bname}_mdup.bam
```

```
==> ./sanchari/j515/dl1.sh <==
wget http://waspsystem.einstein.yu.edu/wsfile/get/file/FA194627-F178-4366-993C-
519448DFF5EA/J515_BHMJHHBCXX_Lane2_TCCTGA_JD13D.1.fastq.gz
wget http://waspsystem.einstein.yu.edu/wsfile/get/file/F0F9499C-7774-4EAE-B497-
353FE5E9CA15/J515_BHMJHHBCXX_Lane2_TCCTGA_JD13D.2.fastq.gz
wget http://waspsystem.einstein.yu.edu/wsfile/get/file/E331804B-2100-40FF-B04B-
D17A802047FA/J515_BHMJHHBCXX_Lane2_TCCTGA_JD13D.1_val_1.fq.gz
wget http://waspsystem.einstein.yu.edu/wsfile/get/file/72DC4E0A-1EB1-457E-A528-
EE502EE3CBCC/J515_BHMJHHBCXX_Lane2_TCCTGA_JD13D.2_val_2.fq.gz
wget http://waspsystem.einstein.yu.edu/wsfile/get/file/FA706ED4-C816-4362-A959-
4B63BDF906AD/J515_BHMJHHBCXX_Lane2_TCCTGA_JD13D.1_val_1_fastqc.html
wget http://waspsystem.einstein.yu.edu/wsfile/get/file/DDF55E49-DE3D-48CB-B736-
503044E10D3E/J515_BHMJHHBCXX_Lane2_TCCTGA_JD13D.2_val_2_fastqc.html
wget http://waspsystem.einstein.yu.edu/wsfile/get/file/CCB2D19E-81F2-4B08-8C99-
033F38C23B44/J515_BHMJHHBCXX_Lane2_TCCTGA_JD13D.1_val_1_screen.png

==> ./sanchari/j515/commands1.sh <==

qsub runBowtie.sh jd  J515_BHMJHHBCXX_Lane2_TCCTGA_JD13D.1_val_1.fq.gz
J515_BHMJHHBCXX_Lane2_TCCTGA_JD13D.2_val_2.fq.gz


#first remove duplicates
qsub markDuplicates.sh jd_sorted.bam



#run peak finding on this sample alone



qsub ~/projects/qsub_scripts/runMacs.sh jd_sorted_mdup.bam
#can't have .. in file names.
#make a link instead
#qsub ~/projects/qsub_scripts/runMacs2.sh jd_sorted_mdup.bam
../j455/hpne_sorted_mdup.bam

ln -s ../j455/hpne_sorted_mdup.bam .
qsub ~/projects/qsub_scripts/runMacs2.sh jd_sorted_mdup.bam hpne_sorted_mdup.bam



==> ./sanchari/j515/runBowtie.sh <==
#!/bin/bash
#$ -cwd
#$ -N bowtie_alignment
#$ -j n
#$ -l h_vmem=5.8G
#$ -pe smp 4
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

#process a pair of fastqs
#
#usage:
#qsub runBowtie.sh aligned_name reads1.fastq.gz reads2.fastq.gz

SNAME=$1
FQ1=$2
```

```
FQ2=$3
REF=../../hg19/hg19

module load bowtie2/2.2.3/gcc.4.4.7
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20


#bowtie2 -t -p 2 -q --local -x $REF -1 <(zcat $FQ1) -2 <(zcat $FQ2) 2>| summary_$
{SNAME}.txt | samtools view -bS - >| ${SNAME}.bam
java -jar $(which SortSam.jar) INPUT=${SNAME}.bam OUTPUT=${SNAME}_sorted.bam
SO=coordinate CREATE_INDEX=true
echo "done" >| $SNAME.status


==> ./sanchari/j455_j515/markDuplicates.sh <==
#!/bin/bash
#$ -cwd
#$ -j n
#$ -l h_vmem=4G
#$ -pe smp 4
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

bam=$1
bname="${bam%.*}"

java -jar $(which BuildBamIndex.jar) INPUT=${bam}
java -jar $(which MarkDuplicates.jar) INPUT=${bam} OUTPUT=${bname}_mdup.bam
METRICS_FILE=${bname}_mdMetric.txt
java -jar $(which BuildBamIndex.jar) INPUT=${bname}_mdup.bam



==> ./sanchari/j455_j515/runMacs2.sh <==
#!/bin/bash
#$ -cwd
#$ -N macs2
#$ -j n
#$ -l h_vmem=9.8G
#$ -pe smp 1
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

#run mac2 peak finding software on a single target bam
#usually we need a control file, igg or input
#
#usage:
#qsub runMacs.sh bamfile.bam

BAM1=$1
BAM2=$2
NAME=C${BAM1%.*}_T${BAM2%.*}
FOL=macs_${NAME}
```

```
#module load MACS2/2.1.0/python.2.7.8
module load MACS2/2.1.0-update/python.2.7.8

echo $BAM1
echo $BAM2
echo $NAME
echo $FOL

macs2 callpeak -f BAMPE -c $BAM1 -t $BAM2 -n $NAME --outdir $FOL --verbose 3

==> ./sanchari/j455_j515/ngsplot/commands_ngs.sh <==

#tss, tes, genebody, exon, cgi, enhancer, dhs
jdbam=/home/kpradhan/Desktop/hpc_home/projects/sanchari/j455_j515/jd_sorted_mdup_me
rged.bam
hpnebam=/home/kpradhan/Desktop/hpc_home/projects/sanchari/j455_j515/hpne_sorted_mdu
p.bam

#
#ngs.plot.r -P 4 -G hg19 -R tss -C $jdbam -O jdplot_tss -T jdAtac -L 5000
#ngs.plot.r -P 4 -G hg19 -R tes -C $jdbam -O jdplot_tes -T jdAtac -L 5000
#ngs.plot.r -P 4 -G hg19 -R genebody -C $jdbam -O jdplot_genebody -T jdAtac -L 5000
#ngs.plot.r -P 4 -G hg19 -R exon -C $jdbam -O jdplot_exon -T jdAtac -L 5000
#ngs.plot.r -P 4 -G hg19 -R cgi -C $jdbam -O jdplot_cgi -T jdAtac -L 5000
#
#ngs.plot.r -P 4 -G hg19 -R tss -C $hpnebam -O hpneplot_tss -T hpneAtac -L 5000
#ngs.plot.r -P 4 -G hg19 -R tes -C $hpnebam -O hpneplot_tes -T hpneAtac -L 5000
#ngs.plot.r -P 4 -G hg19 -R genebody -C $hpnebam -O hpneplot_genebody -T hpneAtac
-L 5000
#ngs.plot.r -P 4 -G hg19 -R exon -C $hpnebam -O hpneplot_exon -T hpneAtac -L 5000
#ngs.plot.r -P 4 -G hg19 -R cgi -C $hpnebam -O hpneplot_cgi -T hpneAtac -L 5000

#run these two again
ngs.plot.r -P 4 -G hg19 -R enhancer -C $jdbam -O jdplot_enhancer -T jdAtac -L 5000
ngs.plot.r -P 4 -G hg19 -R dhs -C $jdbam -O jdplot_dhs -T jdAtac -L 5000
ngs.plot.r -P 4 -G hg19 -R enhancer -C $hpnebam -O hpneplot_enhancer -T hpneAtac -L
5000
ngs.plot.r -P 4 -G hg19 -R dhs -C $hpnebam -O hpneplot_dhs -T hpneAtac -L 5000


==> ./sanchari/j455_j515/mergeBams.sh <==
#!/bin/bash
#$ -cwd
#$ -j n
#$ -l h_vmem=4G
#$ -pe smp 4
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu
#usage:
#qsub mergeBams.sh outfile.bam infile1.bam infile2.bam infile3.bam
#qsub mergeBams.sh hpne_bs.bam *hpne*bs*sorted*bam
module load samtools/1.2/gcc.4.4.7
module load picard-tools/1.92/java.1.8.0_20

#make a folder for storing intermediate resulst
#otherwise, might run out of space on default /tmp
mkdir -p tmp
chmod 777 tmp
```

```
#first argument is the name of the output file
outbam=$1
shift
#the rest of the arguments are the bam files we want to merge

java -jar $(which MergeSamFiles.jar) \
    `echo $@ | sed 's/\(\S\+\)/I=&/g'`\
    ASSUME_SORTED=true\
    TMP_DIR=tmp\
    O=$outbam


==> ./sanchari/j455_j515/commands1.sh <==

ls ../j515/jd_sorted.bam
ls ../j455/jd_sorted.bam

#first remove duplicates
qsub markDuplicates.sh jd_sorted.bam


#then merge the bams
qsub mergeBams.sh jd_sorted_mdup_merged.bam ../j515/jd_sorted_mdup.bam
../j455/jd_sorted_mdup.bam


#now run peak finding

qsub ~/projects/qsub_scripts/runMacs.sh jd_sorted_mdup_merged.bam

ln -s ../j455/hpne_sorted_mdup.bam .
qsub ~/projects/qsub_scripts/runMacs2.sh jd_sorted_mdup_merged.bam
hpne_sorted_mdup.bam
qsub ~/projects/qsub_scripts/runMacs2.sh hpne_sorted_mdup.bam
jd_sorted_mdup_merged.bam

==> ./sanchari/j455_j515/addAnnotationToResults.r <==

library("ChIPpeakAnno")
library(EnsDb.Hsapiens.v75)

library("biomaRt")
library("VariantAnnotation")
library(TxDb.Hsapiens.UCSC.hg19.knownGene)
txdb_hg19 <- TxDb.Hsapiens.UCSC.hg19.knownGene

library(GenomicFeatures)
hg19.refseq.db <- makeTxDbFromUCSC(genome="hg19", table="refGene")
refseq.genes<- genes(hg19.refseq.db)

options(stringsAsFactors=F)



getSiteCounts <- function(x){
      input = GRanges(seqnames=x[,1], ranges=IRanges(x[,2], x[,2]+1), strand="*")
      loc_hg19 <- locateVariants(input, txdb_hg19, AllVariants())
      loc_hg19 = loc_hg19[!duplicated(loc_hg19)]
      table(loc_hg19$LOCATION)
}


#read in the peak lists
atac.hpne.only =
read.table("macs_Cjd_sorted_mdup_merged_Thpne_sorted_mdup/Cjd_sorted_mdup_merged_Th
pne_sorted_mdup_peaks.xls", header=T)
atac.jd.only =
read.table("macs_Chpne_sorted_mdup_Tjd_sorted_mdup_merged/Chpne_sorted_mdup_Tjd_sor
ted_mdup_merged_peaks.xls", header=T)
#atac.hpne.only =
read.table("macs_Cjd_sorted_mdup_merged_Thpne_sorted_mdup/Cjd_sorted_mdup_merged_Th
pne_sorted_mdup_summits.bed")
#atac.jd.only =
read.table("macs_Chpne_sorted_mdup_Tjd_sorted_mdup_merged/Chpne_sorted_mdup_Tjd_sor
ted_mdup_merged_summits.bed")
head(atac.hpne.only)
head(atac.jd.only)

#this is how it's done with oxbs
sCounts.hpne = getSiteCounts(atac.hpne.only)
sCounts.jd = getSiteCounts(atac.jd.only)


sCounts.hpne
sCounts.jd

sCounts.hpne/sum(sCounts.hpne)
sCounts.jd /sum(sCounts.jd)






chunk <- function(x, n){
      split(x, ceiling(seq_along(x)/n))
}

getGenesFromPeaks_chunks <- function(peakValue_chunks){
      results = list()
      #process in chunks
      for (i in 1:length(peakValue_chunks)){
            chunk = peakValue_chunks[[i]]
            print (chunk[1])
            res = NA
            try({
                  G = getBM(attributes = c("hgnc_symbol", "chromosome_name",
"start_position", "end_position"), filters = c("chromosomal_region"), values =
chunk, mart=grch37)
                  res = unique(G[,'hgnc_symbol'])
            }, silent = T)
            results[[i]] = res
      }

      results
}
```

```
getGenesFromPeaks <- function(peaks, chunkSize=1000){
      values = paste0(gsub(peaks[,1], pat="chr(.*)", rep="\\1", perl=T),":", 
peaks[,2], ":", peaks[,3])
      peakValue_chunks = chunk(values, chunkSize)

      results = getGenesFromPeaks_chunks(peakValue_chunks)

      unique(unlist(results))
      #or maybe do a while loop
      #if there were any errors, try one more time
      #check results for NA entries
      #ix = which(sapply(results, function(x){any(is.na(x))}))
      #if (length(ix) > 0){
      #     results2 = getGenesFromPeaks_chunks(peakValue_chunks[ix])
      #}

}

filterPeaks <- function(peaks, grange){
      #intersect the peaks with the TSS locations
      #the chromosomal locations of the peaks
      #granges requires format of chr1:pos1-pos2
      x = GRanges(paste0(gsub(peaks[,1], pat="chr(.*)", rep="\\1", perl=T),":", 
peaks[,2], "-", peaks[,2]+1))
      #which peaks overlap with Tss's
      res = findOverlaps(x, grange)
      ix = unique(queryHits(res))
      peaks[ix,]
}

#get a list of TSS's +/- bp for all all genes
all.genes <- unique( getBM(attributes = "hgnc_symbol", values = "*", mart = grch37) 
)
bp = 2000
tss = getBM(attributes=c("chromosome_name", "transcript_start", "transcript_end"), 
      filters="hgnc_symbol", values=all.genes$hgnc_symbol, mart=grch37)
tss.2k =  GRanges(paste0(tss$chromosome_name,":", tss$transcript_start-bp, "-", 
tss$transcript_start+bp))



#filter away all the sites that are more than 2kb from a TSS
atac.jd.tss2k = filterPeaks(atac.jd.only, tss.2k)
atac.hpne.tss2k = filterPeaks(atac.hpne.only, tss.2k)
dim(atac.jd.tss2k)
dim(atac.hpne.tss1k)



appendClosestGeneInfo <- function(x, pos = 2){
      res = lapply(1:nrow(x), function(i){
            if (i %% 1000 == 0){
                  print(i)
            }
            genes = findClosestGene(x[i,1],x[i, pos],"hg19")
            #if there's more than one, just take the first
            data.frame(genes[1,])
      })
      x.res = do.call(rbind, res)
      cbind(x.res, x)
}

atac.jd.tss2k.genes = appendClosestGeneInfo(atac.jd.tss2k, 5)
atac.hpne.tss2k.genes = appendClosestGeneInfo(atac.hpne.tss2k, 5)

head(atac.jd.tss2k.genes)
head(atac.hpne.tss2k.genes)

#filter away the peaks further than 2k from a gene tss
filterOnTss <- function(x, D){
      x[abs(x$abs_summit - x$txStart) <= D,]
}


atac.jd.tss2k.genes[abs(atac.jd.tss2k.genes$Distance) <= 2000,]


x.jd  = filterOnTss(atac.jd.tss2k.genes, 2000)
x.hpne = filterOnTss(atac.hpne.tss2k.genes, 2000)
write.table(x = x.jd, file="geneAnnoSets/atac_jd_tss2k_genes.txt", row.names=F,
sep="\t")
write.table(x = x.hpne, file="geneAnnoSets/atac_hpne_tss2k_genes.txt", row.names=F,
sep="\t")


#redo the region type counts
#on just the TSS close peaks
sCounts.hpne = getSiteCounts(x.jd)
sCounts.jd = getSiteCounts(x.hpne)

sCounts.hpne
sCounts.jd

sCounts.hpne/sum(sCounts.hpne)
sCounts.jd /sum(sCounts.jd)



#######################################


x = atac.jd.tss2k


res = lapply(1:nrow(x), function(i){
      if (i %% 1000 == 0){
            print(i)
      }
      genes = findClosestGene(x[i,1],x[i, 2],"hg19")
      #if there's more than one, just take the first
      data.frame(genes[1,])
})

x.res = do.call(rbind, res)
head(x.res)
```

```
#load in all the gene info from hg19
knownGenes <- genes(EnsDb.Hsapiens.v75)
rd <- RangedData(IRanges(start = x[,2],
      end = x[,3]), space = x[,1])
ranges.peaks = toGRanges(rd, format="RangedData")

#annotate ranges with the knownGenes database
anno <- annotatePeakInBatch(ranges.peaks, AnnotationData=knownGenes)


#make sure to only take 1 result
x = anno[!(duplicated(anno$peak))]
x$feature
```

```
tssgenes.jd = getGenesFromPeaks(atac.jd.tss2k)
tssgenes.hpne = getGenesFromPeaks(atac.hpne.tss2k)
```

```
lis

grch37 = useMart(biomart="ENSEMBL_MART_ENSEMBL",
host="feb2014.archive.ensembl.org", path="/biomart/martservice",
dataset="hsapiens_gene_ensembl")
head(atac.hpne.only)
pos = paste0(substr(atac.hpne.only[,1], 4, 99), ":", atac.hpne.only[,2], ":",
atac.hpne.only[,3])
#how to get gene id info
G = getBM(attributes=c("hgnc_symbol", "chromosome_name", "start_position",
"end_position", "gene_biotype"), filters = "chromosomal_region", values =
pos[1:1000], mart=grch37)

==> ./sanchari/j455_j515/runMacs.sh <==
#!/bin/bash
#$ -cwd
#$ -N macs2
#$ -j n
#$ -l h_vmem=9.8G
#$ -pe smp 1
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

#run mac2 peak finding software on a single target bam
#usually we need a control file, igg or input
#
#usage:
#qsub runMacs.sh bamfile.bam
```

```
BAM=$1
NAME=${BAM%.*}
FOL=macs_${NAME}

#module load MACS2/2.1.0/python.2.7.8
module load MACS2/2.1.0-update/python.2.7.8

echo $BAM
echo $NAME
echo $FOL

macs2 callpeak -f BAMPE -t $BAM -n $NAME --outdir $FOL --verbose 3
#macs2 callpeak  -t $BAM -n $NAME --outdir $FOL --call-summits --verbose 3

==> ./sanchari/j432_j446/commands2_mc.sh <==
#prepare files for mc discovery.

#both BS and OxBS should have no converted bases
#1.  write an r script that goes through each chromosome
#   writes the site position if both Bs and oxbs have 0% conversino

#run the stream based version of the prog
for chr in Y {22..1} X
do
    #echo $chr
    echo "./findMcSites_stream.r conversionCounts/HPNE_BS_${chr}.txt.gz
conversionCounts/HPNE_OxBS_${chr}.txt.gz mcSites/stream_mcSites_HPNE_${chr}.txt"
    echo "./findMcSites_stream.r conversionCounts/JD_BS_${chr}.txt.gz
conversionCounts/JD_OxBs_${chr}.txt.gz mcSites/stream_mcSites_JD_${chr}.txt"
done | parallel -j 6


#messed up conversinoCountsHPNE_BS_22
#either redo....or skip it and take it from results above.


#do the same using the hpc
chr=21
echo $chr
qsub runFindMcSites.sh test1/HPNE_BS_${chr}.txt.gz test1/HPNE_OxBS_${chr}.txt.gz
hpc_mcSites/stream_mcSites_HPNE_${chr}.txt

#editted this 12-12-16
for chr in Y {22..1} X
do
    echo $chr
    qsub runFindMcSites.sh conversionCounts/HPNE_BS_${chr}.txt.gz
conversionCounts/HPNE_OxBS_${chr}.txt.gz mcSites_fixed/mcSites_HPNE_${chr}.txt
    qsub runFindMcSites.sh conversionCounts/JD_BS_${chr}.txt.gz
conversionCounts/JD_OxBs_${chr}.txt.gz mcSites/mcSites_JD_${chr}.txt
done


#ran this 12-12-16
#the new faster way using unix tools
for chr in Y {22..1} X
do
    echo $chr
    time join <(zcat conversionCounts/HPNE_BS_${chr}.txt.gz | cut -f2,3,4 | awk
'{ printf "%010i %i %i\n" , $1 , $2 , $3 }') <(zcat conversionCounts/HPNE_OxBS_$
{chr}.txt.gz | cut -f2,3,4 | awk '{ printf "%010i %i %i\n" , $1 , $2 , $3 }') | awk
'{if ($2 == $3 && $3 >= 4 && $4 == $5 && $5 >= 4) print $0}' | sed 's/^0*//' | sed
-e "s/.*/chr$chr &/" > mcSites_fixed/mcSites_HPNE_${chr}.txt
    time join <(zcat conversionCounts/JD_BS_${chr}.txt.gz | cut -f2,3,4 | awk
'{ printf "%010i %i %i\n" , $1 , $2 , $3 }') <(zcat conversionCounts/JD_OxBs_$
{chr}.txt.gz | cut -f2,3,4 | awk '{ printf "%010i %i %i\n" , $1 , $2 , $3 }') | awk
'{if ($2 == $3 && $3 >= 4 && $4 == $5 && $5 >= 4) print $0}' | sed 's/^0*//' | sed
-e "s/.*/chr$chr &/" > mcSites_fixed/mcSites_JD_${chr}.txt
done

#join the BS and OXBS files
    #unzip
    #take out the chr* field
    #pad the position with zeros so it's lexigraphical sorted
#retain the rows that have 0 conversion and at least 4 reads in both
#unpad the position
#tack on the chr* field

#merge all the chromos into one file per sample
cd hpc_mcSites
for chr in {1..22} X Y
do
    echo $chr
    cat stream_mcSites_HPNE_${chr}.txt >> mcSites_HPNE_all.txt
    cat stream_mcSites_JD_${chr}.txt >> mcSites_JD_all.txt
done




#join with the list of common4 sites
#1.  sort mcsites by chr pos alphabetically
#       do this by adding a # between the 2nd and 3rd columns
#       effectively merging chr/pos into 1 field, and rest into another field
#    turn spaces into tabs
#2.  join with commons sites
time sort -k 1,2 ../commonSites/common4.txt > ../commonSites/sorted_common4.txt
time join -t# ../commonSites/sorted_common4.txt <(cat mcSites_HPNE_all.txt | sort
-k 1,2 | sed 's/\(\w*\W*\w*\)\(.*\)/\1#\2/' | sed 's/ /\t/g') >|
common4_mcSites_HPNE.txt
time join -t# ../commonSites/sorted_common4.txt <(cat mcSites_JD_all.txt | sort -k
1,2 | sed 's/\(\w*\W*\w*\)\(.*\)/\1#\2/' | sed 's/ /\t/g') >|
common4_mcSites_JD.txt


#12-13-16
#now filter all the cpg sites
for chr in Y {22..1} X
do
    #mcSites are separated by spaces, not tabs
    echo $chr
    qsub ../filterCpg/filterCpg.sh mcSites_fixed/mcSites_JD_${chr}.txt
mcSites_fixed/cpg/mcSites_JD_${chr}_cpg.txt
    qsub ../filterCpg/filterCpg.sh mcSites_fixed/mcSites_HPNE_${chr}.txt
mcSites_fixed/cpg/mcSites_HPNE_${chr}_cpg.txt
done
```

```
#############
#scratch##########
join -t# ../commonSites/common4.txt <(cat mcSites_HPNE_all.txt | sort -k 1,2 | sed
's/\(\w*\W*\w*\)\(.*\)/\1#\2/')


cat mcSites_HPNE_all.txt | sort -k 1,2 | sed 's/\(\w*\W*\w*\)\(.*\)/\1#\2/' | head
-1000 > t1.txt
head t1.txt | sed 's/ /\t/g'


join -t# ../commonSites/sorted_common4.txt <(cat t1.txt | sed 's/ \t/g')

head -1000 ../commonSites/common4.txt
head -10 ../commonSites/common4.txt




==> ./sanchari/j432_j446/runHmcSites.sh <==


./findHmcSites.r conversionCounts/HPNE_BS_22.txt.gz
conversionCounts/HPNE_OxBS_22.txt.gz hmcSites_HPNE_22.txt




#for chr in {1..22} X Y
for chr in Y {22..1} X
do
    echo $chr
    ./findHmcSites.r conversionCounts/HPNE_BS_${chr}.txt.gz
conversionCounts/HPNE_OxBS_${chr}.txt.gz hmcSites_HPNE_${chr}.txt
    ./findHmcSites.r conversionCounts/JD_BS_${chr}.txt.gz
conversionCounts/JD_OxBs_${chr}.txt.gz hmcSites_JD_${chr}.txt
done

for chr in Y {22..1} X
do
    echo $chr
    ./findHmcSites.r conversionCounts/JD_BS_${chr}.txt.gz
conversionCounts/JD_OxBs_${chr}.txt.gz hmcSites_JD_${chr}.txt
done

#run the stream based version of the prog
for chr in 1 2 3
do
    echo $chr
    ./findHmcSites_stream.r conversionCounts/HPNE_BS_${chr}.txt.gz
conversionCounts/HPNE_OxBS_${chr}.txt.gz stream_hmcSites_HPNE_${chr}.txt&
    ./findHmcSites_stream.r conversionCounts/JD_BS_${chr}.txt.gz
conversionCounts/JD_OxBs_${chr}.txt.gz stream_hmcSites_JD_${chr}.txt&
done


#run the python based version
```

```
for chr in Y {22..1} X
do
    echo $chr
    ./findHmcSites.py conversionCounts/HPNE_BS_${chr}.txt.gz
conversionCounts/HPNE_OxBS_${chr}.txt.gz hmcSites_fixed/hmcSites_HPNE_${chr}.txt
    ./findHmcSites.py conversionCounts/JD_BS_${chr}.txt.gz
conversionCounts/JD_OxBs_${chr}.txt.gz hmcSites_fixed/hmcSites_JD_${chr}.txt
done


#12-12-16
#run on hpc cluster
#12-15-16
#reran, because a bug was found in findHmcSites.  Manmy sites were missing
#change script to only output sites with at least 4 reads in both bs and oxbs
for chr in Y {22..1} X
do
    echo $chr
    qsub findHmcSites.sh conversionCounts/HPNE_BS_${chr}.txt.gz
conversionCounts/HPNE_OxBS_${chr}.txt.gz hmcSites_fixed2/hmcSites_HPNE_${chr}.txt
    qsub findHmcSites.sh conversionCounts/JD_BS_${chr}.txt.gz
conversionCounts/JD_OxBs_${chr}.txt.gz hmcSites_fixed2/hmcSites_JD_${chr}.txt
done

#why is chr17 so small in JD?
qsub findHmcSites.sh conversionCounts/JD_BS_17.txt.gz
conversionCounts/JD_OxBs_17.txt.gz test1/hmcSites_JD_17.txt

#now filter them all for cpg sites
chr="Y"

qsub ../filterCpg/filterCpg.sh hmcSites_fixed/hmcSites_JD_${chr}.txt
hmcSites_fixed/cpg/hmcSites_JD_${chr}_cpg.txt
qsub ../filterCpg/filterCpg.sh test1/hmcSites_JD_${chr}.txt test1/hmcSites_JD_$
{chr}_cpg.txt


#12-13-16
for chr in Y {22..1} X
do
    echo $chr
    qsub ../filterCpg/filterCpg.sh hmcSites_fixed2/hmcSites_JD_${chr}.txt
hmcSites_fixed2/cpg/hmcSites_JD_${chr}_cpg.txt
    qsub ../filterCpg/filterCpg.sh hmcSites_fixed2/hmcSites_HPNE_${chr}.txt
hmcSites_fixed2/cpg/hmcSites_HPNE_${chr}_cpg.txt
done


cd cpg



#join the jd and hpne sites into one
#run mergeSites.r


#combine all the chromsomes into one file
```

```
for chr in {1..22} X Y
do
    echo $chr
    cat hmcSites_BOTH_${chr}_cpg_common4.txt >> hmcSites_BOTH_all_cpg_common4.txt
done



==> ./sanchari/j432_j446/commands.sh <==
#testing
tail -50 J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS_merged_sorted_mdup_clip.bedGraph >>
file1.txt
tail -50 J446_AHLT3HBCXX_Lane1_ACAGTG_HPNE-1-BS_merged_sorted_mdup_clip.bedGraph >>
file2.txt
./mergeCounts.pl <(cut -f1,2,5,6 file1.txt) <(cut -f1,2,5,6 file2.txt) chr10



function mergeCounts {
    file1=$1
    file2=$2
    outfile=$3

    for chr in {1..22} X Y
    do
        echo $chr
        time ./mergeCounts.pl <(cut -f1,2,5,6 $file1 | grep -e "^chr$chr\s") <(cut
-f1,2,5,6 $file2 | grep -e "^chr$chr\s") chr$chr | gzip > ${outfile}_${chr}.txt.gz
    done
}

mergeCounts\
    file1.txt\
    file2.txt\
    test

mergeCounts\
    J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS_merged_sorted_mdup_clip.bedGraph\
    J446_AHLT3HBCXX_Lane1_ACAGTG_HPNE-1-BS_merged_sorted_mdup_clip.bedGraph\
    HPNE_BS

mergeCounts\
    J432_AHHH5NBCXX_Lane2_GCCAAT_HPNE-1OxBS_merged_sorted_mdup_clip.bedGraph\
    J446_AHLT3HBCXX_Lane2_GCCAAT_HPNE-1-OxBS_merged_sorted_mdup_clip.bedGraph\
    HPNE_OxBS

mergeCounts\
    J432_BHKVCGBCXX_Lane1_CTTGTA_JD-BS_merged_sorted_mdup_clip.bedGraph\
    J446_BHLYGMBCXX_Lane1_CTTGTA_JD-BS_merged_sorted_mdup_clip.bedGraph\
    JD_BS

mergeCounts\
    J432_BHKVCGBCXX_Lane2_GTGAAA_JD-OxBS_merged_sorted_mdup_clip.bedGraph\
    J446_BHLYGMBCXX_Lane2_GTGAAA_JD-OxBS_merged_sorted_mdup_clip.bedGraph\
    JD_OxBs
```

```
#########################################

file1=J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS_merged_sorted_mdup_clip.bedGraph
file2=J446_AHLT3HBCXX_Lane1_ACAGTG_HPNE-1-BS_merged_sorted_mdup_clip.bedGraph
outfile=HPNE_BS
chr=22
echo $chr
time ./mergeCounts.pl <(cut -f1,2,5,6 $file1 | grep -e "^chr$chr\s") <(cut
-f1,2,5,6 $file2 | grep -e "^chr$chr\s") chr$chr | gzip > ${outfile}_${chr}.txt.gz



#file1=file1.txt
#file2=file2.txt
#outfile=test
for chr in {1..22} X Y
do
    echo $chr
    time ./mergeCounts.pl <(cut -f1,2,5,6 $file1 | grep -e "^chr$chr\s") <(cut
-f1,2,5,6 $file2 | grep -e "^chr$chr\s") chr$chr | gzip > ${outfile}_${chr}.txt.gz
done



#create set of "common sites".
#sites that have at least 4 reads in all samples
ls conversionCounts/JD_BS_*.txt.gz
zcat conversionCounts/JD_BS_*.txt.gz | ./filterSitesByCount.pl 4 >|
commonSites/common4_JD_BS.txt
zcat conversionCounts/HPNE_BS_*.txt.gz | ./filterSitesByCount.pl 4 >|
commonSites/common4_HPNE_BS.txt
zcat conversionCounts/JD_OxBs_*.txt.gz | ./filterSitesByCount.pl 4 >|
commonSites/common4_JD_OxBS.txt
zcat conversionCounts/HPNE_OxBS_*.txt.gz | ./filterSitesByCount.pl 4 >
commonSites/common4_HPNE_OxBS.txt

#use my own computer to do the sorting
time sort --parallel=6 -T tmp commonSites/common4_JD_BS.txt >|
commonSites/sorted_common4_JD_BS.txt
time sort --parallel=6 -T tmp commonSites/common4_HPNE_BS.txt >|
commonSites/sorted_common4_HPNE_BS.txt
time sort --parallel=6 -T tmp commonSites/common4_JD_OxBS.txt >|
commonSites/sorted_common4_JD_OxBS.txt
time sort --parallel=6 -T tmp commonSites/common4_HPNE_OxBS.txt >|
commonSites/sorted_common4_HPNE_OxBS.txt


#join all the files, retaining only those present in all
cd commonSites
time join -t# sorted_common4_JD_OxBS.txt <(join -t# sorted_common4_JD_BS.txt <(join
-t# sorted_common4_HPNE_BS.txt sorted_common4_HPNE_OxBS.txt)) >| common4.txt

#get the common sites only looking at a single sample
cd commonSites
time join -t# sorted_common4_JD_OxBS.txt sorted_common4_JD_BS.txt  >|
common4_JD.txt&
time join -t# sorted_common4_HPNE_OxBS.txt sorted_common4_HPNE_BS.txt  >|
common4_HPNE.txt&
```

```
#$ wc -l common4.txt
#527710019 common4.txt


#now lets filter away the hmcsites that aren't common.
#1.  sort hmcsites by chr pos alphabetically
#2.  join with commons sites
time join -t# ../commonSites/common4.txt <(tail -n +2 hmcSites_HPNE_all.txt | sort
-k 1,2 | sed 's/\(\w*\W*\w*\)\(.*\)/\1#\2/') >| common4_hmcSites_HPNE.txt
time join -t# ../commonSites/common4.txt <(tail -n +2 hmcSites_JD_all.txt | sort -k
1,2 | sed 's/\(\w*\W*\w*\)\(.*\)/\1#\2/') >| common4_hmcSites_JD.txt

#wc -l common4_hmcSites_*
#  4657389 common4_hmcSites_HPNE.txt
#  6067629 common4_hmcSites_JD.txt
# 10725018 total




###############
#scratch
time join -t# ../commonSites/common4.txt <(head ../commonSites/common4.txt )
time join -t# ../commonSites/common4.txt <(tail -n +2 hmcSites_HPNE_all.txt | head
-5000 | sed 's/\(\w*\W*\w*\)\(.*\)/\1#\2/' | sort -k 1 -t#)
time join -t# ../commonSites/common4.txt <(tail -n +2 hmcSites_HPNE_all.txt | grep
-e "chr10" | head -5000 | sed 's/\(\w*\W*\w*\)\(.*\)/\1#\2/' | sort -k 1 -t#)

tail -n +2 hmcSites_HPNE_all.txt | grep -e "chr10" | sed 's/\(\w*\W*\w*\)\
(.*\)/\1#\2/' | sort -k 1 -t# >| test1.txt
tail -n +2 hmcSites_HPNE_all.txt | grep -e "chr10" | sort -k 1,2  >| test1a.txt
tail -n +2 hmcSites_HPNE_all.txt | grep -e "chr10" | sort -k 1,2 | sed 's/\
(\w*\W*\w*\)\(.*\)/\1#\2/' >| test1b.txt

join -t# ../commonSites/common4.txt test1.txt > test2.txt
#sort by chr then pos numerically




==> ./sanchari/j432_j446/examineFixedOxbs.r <==
library("VariantAnnotation")
library(TxDb.Hsapiens.UCSC.hg19.knownGene)
txdb_hg19 <- TxDb.Hsapiens.UCSC.hg19.knownGene






#1.  Want to look at the methylation percentages of
#the jd and hpne samples at cpg sites.
#amit said the meth percentages looked a little high
#for the jd sample(over half had 100%)
#the mc histograms of the hmcSites should be high
#in the BS sample, and low in the OxBS sample


options(stringsAsFactors=F)
```

```
#load in the merged data
x = read.table("hmcSites_fixed2/cpg/hmcSites_BOTH_all_cpg_common4.txt")

colnames(x) = c("chr", "pos",
      "jd.bs.meth", "jd.bs.tot", "jd.oxbs.meth", "jd.oxbs.tot", "jd.or", "jd.pval",
      "hpne.bs.meth", "hpne.bs.tot", "hpne.oxbs.meth", "hpne.oxbs.tot", "hpne.or",
"hpne.pval"
)

head(x)
tail(x)
dim(x)
mean(x$jd.bs.tot)
mean(x$jd.oxbs.tot)
mean(x$hpne.bs.tot)
mean(x$hpne.oxbs.tot)

#for jd and hpne
#how many sites have at least 4 reads
#amongst those sites, what is avearge depth?





#try to get measures of overall methylation between samples
#what percentage of cpg sites have methylation less than 50%
ix.jd.lt50 = (x[,"jd.oxbs.meth"]/x[,"jd.oxbs.tot"] <= .50)
ix.hpne.lt50 = (x[,"hpne.oxbs.meth"]/x[,"hpne.oxbs.tot"] <= .50)
mean(x[,"jd.oxbs.meth"]/x[,"jd.oxbs.tot"] <= .50)
mean(x[,"hpne.oxbs.meth"]/x[,"hpne.oxbs.tot"] <= .50)


#what percentage of cpg sites have methylation greater than 50%
ix.jd.gt50 = (x[,"jd.oxbs.meth"]/x[,"jd.oxbs.tot"] > .50)
ix.hpne.gt50 = (x[,"hpne.oxbs.meth"]/x[,"hpne.oxbs.tot"] > .50)
mean(x[,"jd.oxbs.meth"]/x[,"jd.oxbs.tot"] > .50)
mean(x[,"hpne.oxbs.meth"]/x[,"hpne.oxbs.tot"] > .50)

#what percentage of cpg sites have total methylation
ix.jd.100 = (x[,"jd.oxbs.meth"]/x[,"jd.oxbs.tot"] == 1)
ix.hpne.100 = (x[,"hpne.oxbs.meth"]/x[,"hpne.oxbs.tot"] == 1)
mean(ix.jd.100)
mean(ix.hpne.100)

mean(x[,"jd.pval"] < 0.05)
mean(x[,"hpne.pval"] < 0.05)


#get indices of the signif hmc sites
jd.hmc.ix = which(x[,"jd.pval"] < 0.05)
hpne.hmc.ix = which(x[,"hpne.pval"] < 0.05)

length(jd.hmc.ix) / nrow(x)
length(hpne.hmc.ix) / nrow(x)
length(jd.hmc.ix)
length(hpne.hmc.ix)
```

```
#write the site lists
write.table(x[jd.hmc.ix,1:2], file="jd_hmc_p05.txt", col.names=F, row.names=F,
quote=F, sep="\t")
write.table(x[hpne.hmc.ix,1:2], file="hpne_hmc_p05.txt", col.names=F, row.names=F,
quote=F, sep="\t")

write.table(x[ix.jd.gt50,1:2], file="jd_mc_gt50.txt", col.names=F, row.names=F,
quote=F, sep="\t")
write.table(x[ix.hpne.gt50,1:2], file="hpne_mc_gt50.txt", col.names=F, row.names=F,
quote=F, sep="\t")

write.table(x[ix.jd.lt50,1:2], file="jd_mc_lt50.txt", col.names=F, row.names=F,
quote=F, sep="\t")
write.table(x[ix.hpne.lt50,1:2], file="hpne_mc_lt50.txt", col.names=F, row.names=F,
quote=F, sep="\t")

write.table(x[ix.jd.100,1:2], file="jd_mc_100.txt", col.names=F, row.names=F,
quote=F, sep="\t")
write.table(x[ix.hpne.100,1:2], file="hpne_mc_100.txt", col.names=F, row.names=F,
quote=F, sep="\t")





#get percentage of sites that lie within exons introns promoters etc

getSiteCounts <- function(x){
      input = GRanges(seqnames=x[,1], ranges=IRanges(x[,2], x[,2]+1), strand="*")
      loc_hg19 <- locateVariants(input, txdb_hg19, AllVariants())
      loc_hg19 = loc_hg19[!duplicated(loc_hg19)]
      table(loc_hg19$LOCATION)
}

sCounts.jd.hmc = getSiteCounts(x[jd.hmc.ix,])
sCounts.hpne.hmc = getSiteCounts(x[hpne.hmc.ix,])

sCounts.jd.hmc
sCounts.hpne.hmc

sCounts.jd.hmc/sum(sCounts.jd.hmc)
sCounts.hpne.hmc/sum(sCounts.hpne.hmc)


#meth > 50%
sCounts.jd.mc = getSiteCounts(x[ix.jd.gt50,])
sCounts.hpne.mc = getSiteCounts(x[ix.hpne.gt50,])
sCounts.jd.mc
sCounts.hpne.mc
sCounts.jd.mc/sum(sCounts.jd.mc)
sCounts.hpne.mc/sum(sCounts.hpne.mc)



#histograms of methylation across samples
png("methPerc_allCpg.png")
par(mfcol=c(2,2))
```

```
hist(x[,"jd.bs.meth"]/x[,"jd.bs.tot"], 100,         main="all cpg\njd bs meth %"
)
hist(x[,"jd.oxbs.meth"]/x[,"jd.oxbs.tot"], 100,     main="all cpg\njd oxbs meth %"
)
hist(x[,"hpne.bs.meth"]/x[,"hpne.bs.tot"], 100,     main="all cpg\nhpne bs meth %"
)
hist(x[,"hpne.oxbs.meth"]/x[,"hpne.oxbs.tot"], 100, main="all cpg\nhpne oxbs meth
%")
dev.off()


png("methPerc_hmcCpg.png")
par(mfcol=c(2,2))
hist(x[jd.hmc.ix,"jd.bs.meth"]/x[jd.hmc.ix,"jd.bs.tot"], 100             ,
main="hmc cpg\njd bs meth %"    )
hist(x[jd.hmc.ix,"jd.oxbs.meth"]/x[jd.hmc.ix,"jd.oxbs.tot"], 100         ,
main="hmc cpg\njd oxbs meth %"  )
hist(x[hpne.hmc.ix,"hpne.bs.meth"]/x[hpne.hmc.ix,"hpne.bs.tot"], 100     ,
main="hmc cpg\nhpne bs meth %"  )
hist(x[hpne.hmc.ix,"hpne.oxbs.meth"]/x[hpne.hmc.ix,"hpne.oxbs.tot"],  100,
main="hmc cpg\nhpne oxbs meth %")
dev.off()


#get hydroxy meth percentages in and out of hmcsites
#hmc% is meth% bs - meth% oxbs

jd.hmcPerc = x[,"jd.bs.meth"]/x[,"jd.bs.tot"] -
x[,"jd.oxbs.meth"]/x[,"jd.oxbs.tot"]
jd.hmcPerc.hmcSites = x[jd.hmc.ix,"jd.bs.meth"]/x[jd.hmc.ix,"jd.bs.tot"] -
x[jd.hmc.ix,"jd.oxbs.meth"]/x[jd.hmc.ix,"jd.oxbs.tot"]
hpne.hmcPerc = x[,"hpne.bs.meth"]/x[,"hpne.bs.tot"] -
x[,"hpne.oxbs.meth"]/x[,"hpne.oxbs.tot"]
hpne.hmcPerc.hmcSites = x[hpne.hmc.ix,"hpne.bs.meth"]/x[hpne.hmc.ix,"hpne.bs.tot"]
- x[hpne.hmc.ix,"hpne.oxbs.meth"]/x[hpne.hmc.ix,"hpne.oxbs.tot"]

#how many sites with hmc > 50%?
mean(jd.hmcPerc > .50)
mean(hpne.hmcPerc > .50)
mean(x[,"jd.pval"] < 0.05)
mean(x[,"jd.pval"] < 0.01)

sum(jd.hmcPerc > 0 )
sum(jd.hmcPerc < 0 )
length(jd.hmc.ix)


jd.mcPerc.hmcSites

png("hmcPerc_cpg.png", 1000, 1000)
par(mfcol=c(2,2))
hist(jd.hmcPerc, 100)
hist(jd.hmcPerc.hmcSites, 100)
hist(hpne.hmcPerc, 100)
hist(hpne.hmcPerc.hmcSites, 100)
dev.off()


#overlay histograms of MC percentages
```

```
#with HMC percentages (of significant sites)

jd.mcPerc = x[,"jd.oxbs.meth"]/x[,"jd.oxbs.tot"]
hpne.mcPerc = x[,"hpne.oxbs.meth"]/x[,"hpne.oxbs.tot"]
jd.mcPerc.hmcSites = x[jd.hmc.ix,"jd.oxbs.meth"]/x[jd.hmc.ix,"jd.oxbs.tot"]
hpne.mcPerc.hmcSites =
x[hpne.hmc.ix,"hpne.oxbs.meth"]/x[hpne.hmc.ix,"hpne.oxbs.tot"]


mean(jd.mcPerc)
sd(jd.mcPerc)
mean(hpne.mcPerc)
sd(hpne.mcPerc)

mean(jd.hmcPerc.hmcSites)
sd(jd.hmcPerc.hmcSites)
mean(hpne.hmcPerc.hmcSites)
sd(hpne.hmcPerc.hmcSites)


#examine the effect of smoothing on density plots
png("test1.png")
par(mfcol=c(3, 1))
plot(density(jd.mcPerc), col="red")
lines(density(jd.hmcPerc.hmcSites), col="blue")

plot(density(jd.mcPerc), col="red", ylim=c(0, 4))
lines(density(jd.hmcPerc.hmcSites), col="blue")

plot(density(jd.mcPerc, width=0.1),ylim=c(0, 4),  col="red")
lines(density(jd.hmcPerc.hmcSites, width=0.1), col="blue")
dev.off()



png("jd_histMCHmc_smooth.png")
plot(density(jd.mcPerc, width=0.1), ylim=c(0, 4), col="red", main="Methylation
Percentage Densities:  JD", xlab="Methylation Percentage")
lines(density(jd.hmcPerc.hmcSites, width=0.1), col="blue")
legend("topleft", c("MC", "HMC"), pch=16, col=c("red", "blue"))
dev.off()

png("hpne_histMCHmc_smooth.png")
plot(density(hpne.mcPerc, width=0.1), ylim=c(0, 4), col="red", main="Methylation
Percentage Densities:  HPNE", xlab="Methylation Percentage")
lines(density(hpne.hmcPerc.hmcSites, width=0.1), col="blue")
legend("topleft", c("MC", "HMC"), pch=16, col=c("red", "blue"))
dev.off()

png("hists_McPerc_smooth.png")
plot(density(jd.mcPerc, width=0.1), col="red", main="MC Percentage Densities",
xlab="Methylation Percentage")
lines(density(hpne.mcPerc, width=.1), col="blue")
legend("topleft", c("JD", "HPNE"), pch=16, col=c("red", "blue"))
dev.off()

png("hists_HmcPerc_smooth.png")
plot(density(jd.hmcPerc.hmcSites, width=0.10), col="red", ylim=c(0, 4), main="HMC
Percentage Densities", xlab="Hydroxy-Methylation Percentage")
```

```
lines(density(hpne.hmcPerc.hmcSites, width=.10), col="blue")
legend("topleft", c("JD", "HPNE"), pch=16, col=c("red", "blue"))
dev.off()


#JD
png("jd_histMcHmc.png")
#methylation histogram over all cpg sites
hist.mc = hist(x[,"jd.oxbs.meth"]/x[,"jd.oxbs.tot"], plot=F , 100 )
#hmc histogram over all significant cpg sites
hist.hmc.sig = hist(jd.hmcPerc.hmcSites, plot=F, 100)
xlim <- range(hist.mc$breaks,hist.hmc.sig$breaks)
ylim <- range(0,hist.mc$density, hist.hmc.sig$density)
## plot the first graph
plot(hist.mc,xlim = xlim, ylim = ylim,
     col = rgb(1,0,0,0.4),xlab = 'Lengths',
     freq = FALSE, ## relative, not absolute frequency
     main = 'JD:  Methylation')
## plot the second graph on top of this
opar <- par(new = FALSE)
plot(hist.hmc.sig,xlim = xlim, ylim = ylim,
     xaxt = 'n', yaxt = 'n', ## don't add axes
     col = rgb(0,0,1,0.4), add = TRUE,
     freq = FALSE) ## relative, not absolute frequency
## add a legend in the corner
legend('topleft',c('MC','HMC'),
       fill = rgb(1:0,0,0:1,0.4), bty = 'n',
       border = NA)
par(opar)
dev.off()



#HPNE
png("hpne_histMcHmc.png")
#methylation histogram over all cpg sites
hist.mc = hist(x[,"hpne.oxbs.meth"]/x[,"hpne.oxbs.tot"], plot=F , 100 )
#hmc histogram over all significant cpg sites
hist.hmc.sig = hist(hpne.hmcPerc.hmcSites, plot=F, 100)
xlim <- range(hist.mc$breaks,hist.hmc.sig$breaks)
ylim <- range(0,hist.mc$density, hist.hmc.sig$density)
## plot the first graph
plot(hist.mc,xlim = xlim, ylim = ylim,
     col = rgb(1,0,0,0.4),xlab = 'Lengths',
     freq = FALSE, ## relative, not absolute frequency
     main = 'HPNE:  Methylation')
## plot the second graph on top of this
opar <- par(new = FALSE)
plot(hist.hmc.sig,xlim = xlim, ylim = ylim,
     xaxt = 'n', yaxt = 'n', ## don't add axes
     col = rgb(0,0,1,0.4), add = TRUE,
     freq = FALSE) ## relative, not absolute frequency
## add a legend in the corner
legend('topleft',c('MC','HMC'),
       fill = rgb(1:0,0,0:1,0.4), bty = 'n',
       border = NA)
par(opar)
dev.off()
```

```
#overlay hmc jd and hpne
png("hists_HmcPerc.png")
hist.hmc.jd= hist(jd.hmcPerc.hmcSites, plot=F, 100)
hist.hmc.hpne= hist(hpne.hmcPerc.hmcSites, plot=F, 100)
xlim <- range(hist.hmc.jd$breaks,hist.hmc.hpne$breaks)
ylim <- range(0,hist.hmc.jd$density, hist.hmc.hpne$density)
## plot the first graph
plot(hist.hmc.jd,xlim = xlim, ylim = ylim,
     col = rgb(1,0,0,0.4),xlab = 'Lengths',
     freq = FALSE, ## relative, not absolute frequency
     main = 'HMC: JD, HPNE')
## plot the second graph on top of this
opar <- par(new = FALSE)
plot(hist.hmc.hpne,xlim = xlim, ylim = ylim,
     xaxt = 'n', yaxt = 'n', ## don't add axes
     col = rgb(0,0,1,0.4), add = TRUE,
     freq = FALSE) ## relative, not absolute frequency
## add a legend in the corner
legend('topleft',c('JD','HPNE'),
       fill = rgb(1:0,0,0:1,0.4), bty = 'n',
       border = NA)
par(opar)
dev.off()
```

```
#overlay mc: jd, hpne
png("hists_McPerc.png")
hist.mc.jd = hist(x[,"jd.oxbs.meth"]/x[,"jd.oxbs.tot"], plot=F, 100  )
hist.mc.hpne = hist(x[,"hpne.oxbs.meth"]/x[,"hpne.oxbs.tot"], plot=F, 100)
xlim <- range(hist.mc.jd$breaks,hist.mc.hpne$breaks)
ylim <- range(0,hist.mc.jd$density, hist.mc.hpne$density)
## plot the first graph
plot(hist.mc.jd,xlim = xlim, ylim = ylim,
     col = rgb(1,0,0,0.4),xlab = 'Lengths',
     freq = FALSE, ## relative, not absolute frequency
     main = 'MC: JD, HPNE')
## plot the second graph on top of this
opar <- par(new = FALSE)
plot(hist.mc.hpne,xlim = xlim, ylim = ylim,
     xaxt = 'n', yaxt = 'n', ## don't add axes
     col = rgb(0,0,1,0.4), add = TRUE,
     freq = FALSE) ## relative, not absolute frequency
## add a legend in the corner
legend('topleft',c('JD','HPNE'),
       fill = rgb(1:0,0,0:1,0.4), bty = 'n',
       border = NA)
par(opar)
dev.off()
```

```
# there are a total of 27,999,538 cpg sites in genome
# we have data for 11,296,328 of them



#brd promoter region
#look at hmc on chr19 between these points
p1 = 15391262 - 1500
p2 = 15443342 + 1500
head(x)
x.19 = x[x[,1] == "chr19",]
ix.brd4 = which((x.19$pos >= p1) & (x.19$pos < p2))
length(ix.brd4)
#plot out mc and hmc over this region
x.brd4 = x.19[ix.brd4,]
head(x.brd4)
jd.mc = x.brd4$jd.oxbs.meth/x.brd4$jd.oxbs.tot
jd.hmc = x.brd4[,"jd.bs.meth"]/x.brd4[,"jd.bs.tot"] -
x.brd4[,"jd.oxbs.meth"]/x.brd4[,"jd.oxbs.tot"]
hpne.mc = x.brd4$hpne.oxbs.meth/x.brd4$hpne.oxbs.tot
hpne.hmc = x.brd4[,"hpne.bs.meth"]/x.brd4[,"hpne.bs.tot"] -
x.brd4[,"hpne.oxbs.meth"]/x.brd4[,"hpne.oxbs.tot"]
hmc
stem(hmc)

#draw the less confidence dots with transparency
blues = sapply(1:100, function(i){
      rgb(0,0,1,i/100.0)
})
jd.hmc.cols =blues[101 - round(x.brd4$jd.pval, 2)*100]
hpne.hmc.cols =blues[101 - round(x.brd4$hpne.pval, 2)*100]


reds = sapply(1:100, function(i){
      rgb(1,0,0,i/100.0)
})
#more reads -> more confidences
n = max(x.brd4$jd.oxbs.tot)
jd.mc.cols = reds[round(x.brd4$jd.oxbs.tot / n * 100)]
hpne.mc.cols = reds[round(x.brd4$hpne.oxbs.tot / n * 100)]

png("jd_brd4_mc_hmc_conf.png")
plot(x.brd4$pos, jd.hmc, col=jd.hmc.cols, ylim=c(-1, 1), ylab="methylation")
points(x.brd4$pos, jd.mc, col=jd.mc.cols)
dev.off()

png("hpne_brd4_mc_hmc_conf.png")
plot(x.brd4$pos, hpne.hmc, col=hpne.hmc.cols, ylim=c(-1, 1), ylab="methylation")
points(x.brd4$pos, hpne.mc, col=hpne.mc.cols)
dev.off()

png("jd_brd4_mc_hmc.png")
plot(x.brd4$pos, hmc, col="blue", ylim=c(-1, 1), ylab="methylation")
points(x.brd4$pos, mc, col="red")
dev.off()

x11()
plot(x.brd4$pos, hpne.hmc, col="blue", ylim=c(-1, 1), ylab="methylation")
points(x.brd4$pos, hpne.mc, col="red")
```

```
plot(x.brd4$pos, ylim=c(-1, 1), xlim=c(min(x.brd4$pos), max(x.brd4$pos)))
segments(x0=x.brd4$pos, y0=x.brd4$pos*0, y1=mc, col="red")
segments(x0=x.brd4$pos, y0=x.brd4$pos*0, y1=hmc, col="blue")

#find places where methylation is significantly different between
#hpne and jd
diffmeth.pvals = NA*numeric(nrow(x))
diffmeth.or = NA*numeric(nrow(x))
for (i in 1:nrow(x)){
      if (i %% 10000 == 0){
            print(i)
      }

      #a = 10
      #b = 20
      #c = 30
      #d = 40
      a = x[i,"jd.bs.meth"] #num converted reads
      b = x[i,"jd.bs.tot"] - a #num of converted reads
      c = x[i,"hpne.bs.meth"]
      d = x[i,"hpne.bs.tot"] - c

      #see if there's a significant different in the ratios
      #of methylated reads between jd and hpne
      res = fisher.test(matrix(c(a, b ,c, d), ncol=2))
      p = res$p.value
      est = res$estimate
      meth.pvals[i] = p
      meth.or[i] =est
}

save.image("work_diffMeth.rData")
qvals = p.adjust(meth.pvals)

ix.sig =  qvals < 0.05

#is this hmc loss and gain?
#more meth in JD, sig hmc in HPNE
sum(ix.sig & meth.or > 1 & x[,"hpne.pval"] < 0.05)
head(x[ix.sig & meth.or > 1 & x[,"hpne.pval"] < 0.05,])

#more meth in hpne, sig hmc in JD
sum(ix.sig & meth.or < 1 & x[,"jd.pval"] < 0.05)
x[ix.sig & meth.or < 1 & x[,"jd.pval"] < 0.05,]









#collect the four sets
#sites with significant HMC in JD
#sites with significant HMC in HPNE
#sites with significantly more MC in JD than HPNE
```

```
#sites with significantly more MC in HPNE than JD



#get site distributions for these four sets
sum(x[,"jd.pval"] < 0.05)
sum(x[,"jd.pval"] < 0.01)
sum(x[,"hpne.pval"] < 0.01)
sum(x[,"hpne.pval"] < 0.05)

png("methSigDiff_hist.png")
par(mfcol=c(2,1))
hist(log(meth.or), 1000)
hist(log(meth.or[qvals < 0.2]), 1000)
dev.off()

qvals = p.adjust(meth.pvals)
sum(meth.or < 1 & qvals < 0.2)
sum(meth.or > 1 & qvals < 0.2)
sum(qvals < 0.2)
ix = meth.or > 1 & qvals < 0.2
head(x[ix,])


jd.mc.sites = x[meth.or > 1 & qvals < 0.2,1:2]
hpne.mc.sites = x[meth.or < 1 & qvals < 0.2,1:2]
jd.hmc.sites = x[(x[,"jd.or"] > 1) & (x[,"jd.pval"] < 0.05),1:2]
hpne.hmc.sites = x[(x[,"hpne.or"] > 1) & (x[,"hpne.pval"] < 0.05),1:2]
dim(jd.mc.sites)
dim(hpne.mc.sites)
dim(jd.hmc.sites)
dim(hpne.hmc.sites)


write.table(file="sites_fixed2_jdMc.txt", jd.mc.sites, col.names=F, row.names=F,
quote=F, sep="\t")
write.table(file="sites_fixed2_hpneMc.txt", hpne.mc.sites, col.names=F,
row.names=F, quote=F, sep="\t")
write.table(file="sites_fixed2_jdHmc_p05.txt", jd.hmc.sites, col.names=F,
row.names=F, quote=F, sep="\t")
write.table(file="sites_fixed2_hpneHmc_p05.txt", hpne.hmc.sites, col.names=F,
row.names=F, quote=F, sep="\t")


#what if we only look at the sites with significant meth
x[,"jd.meth"]

i = 1

a = x[i,"jd.bs.meth"]
b = x[i,"jd.bs.tot"] - a
c = x[i,"hpne.bs.meth"]
d = x[i,"hpne.bs.tot"] - c

matrix(c(a,b,c,d), 2)

ix = which((qvals < 0.05) & (x[,"jd.pval"] < 0.05))
a = x[ix, "jd.pval"]
b = p.adjust(a)
```

```
c = which(b < 0.05)
x[ix[c],]
which(p.adjust(x[(meth.pvals < 0.05) & (x[,"jd.pval"] < 0.05), "jd.pval"]) < 0.05)
x[c(19880, 24423),]
sum((meth.pvals > 0.05) & (x[,"jd.pval"] < 0.05))
sum((qvals >= 0.05) & (x[,"jd.pval"] < 0.05))

sum((qvals < 0.05) & (x[qvals < 0.05,"hpne.pval"] < 0.05))
q1 = p.adjust(x[qvals < 0.05,"hpne.pval"])
sum(q1 < 0.15)

sum(meth.pvals < 0.05)
fisher.test(








######################################
##scratch
#view the histograms of odds ratios.

#write out the places with significant difference between jd and hpne

n = 320000
hist(meth.pvals, 100)
hist(meth.pvals, 100)

hist(log(meth.or[meth.or != 1]), 1000)
hist(log(meth.or[meth.or != 1]), 100)

hist(log(meth.or[1:n][meth.or[1:n] != 1]), 1000)
hist(log(meth.or[1:n][meth.or[1:n] != 1]), 100)
sum(meth.pvals[1:n] < 0.05, na.rm=T)
qvals = p.adjust(meth.pvals)
mean(qvals < 0.05)
sum(qvals < 0.05)
hist(log(meth.or[qvals < 0.05]), 100)
ix = which(qvals < 0.05)
length(ix)
x[ix,]

==> ./sanchari/j432_j446/hmcSites/mergeSites.sh <==
#add the header to the merged file
head -1 hmcSites_HPNE_22.txt >| hmcSites_HPNE_all.txt
head -1 hmcSites_JD_22.txt >| hmcSites_JD_all.txt
for chr in {1..22} X Y
do
    echo $chr
    #process the HPNE files
    echo `ls *hmcSites_HPNE_${chr}.txt`
    file=`ls *hmcSites_HPNE_${chr}.txt`
    if echo $file | grep -e "stream"; then
        cat $file >> hmcSites_HPNE_all.txt
    else
        echo "nostream"
        #skip the first line and take out quotations
        tail -n +2 $file  | sed 's/"//g' >> hmcSites_HPNE_all.txt
    fi
    #process the JD files
    echo `ls *hmcSites_JD_${chr}.txt`
    file=`ls *hmcSites_JD_${chr}.txt`
    if echo $file | grep -e "stream"; then
        cat $file >> hmcSites_JD_all.txt
    else
        echo "nostream"
        #skip the first line and take out quotations
        tail -n +2 $file  | sed 's/"//g' >> hmcSites_JD_all.txt
    fi
done


==> ./sanchari/j432_j446/hmcSites/getDiffSites.sh <==

#get the sites in JD but not in HPNE
comm -23 --check-order <(cut -f1-2 common4_hmcSites_JD.txt | sort) <(cut -f1-2
common4_hmcSites_HPNE.txt | sort) >| common4_hmcSites_JD-HPNE.txt
comm -13 --check-order <(cut -f1-2 common4_hmcSites_JD.txt | sort) <(cut -f1-2
common4_hmcSites_HPNE.txt | sort) >| common4_hmcSites_HPNE-JD.txt





==> ./sanchari/j432_j446/mergeCounts.pl <==
#!/usr/bin/perl

use strict;
use warnings;

#load in the command line arguments
my ($bed1, $bed2, $chr) = @ARGV;

#print "file1:  $bed1\n";
#print "file2:  $bed2\n";
#print "chr:  $chr\n";



#open file handles to each file
open my $in1, '<', $bed1 or die;
open my $in2, '<', $bed2 or die;


#read the first line in each file and init the fields
my ($chr1, $pos1, $nMeth1, $nTot1) = split ' ', <$in1>;
my ($chr2, $pos2, $nMeth2, $nTot2) = split ' ', <$in2>;

#print "$chr1, $pos1, $nMeth1, $nTot1\n";
#print "$chr2, $pos2, $nMeth2, $nTot2\n";

#keep going while there's still stuff
#being read by both files
while (!eof $in1 and !eof $in2){
      #same position
      #add counts from both lines
      #move both file handles up one
      if ($pos1 == $pos2){
            print "$chr1\t$pos1\t".($nMeth1+$nMeth2)."\t".($nTot1+$nTot2)."\n";
            ($chr1, $pos1, $nMeth1, $nTot1) = split ' ', <$in1>;
            ($chr2, $pos2, $nMeth2, $nTot2) = split ' ', <$in2>;
      }
      elsif ($pos1 > $pos2){
            #print 2nd file, move it's handle up one
            print "$chr2\t$pos2\t$nMeth2\t$nTot2\n";
            ($chr2, $pos2, $nMeth2, $nTot2) = split ' ', <$in2>;
      }
      elsif ($pos2 > $pos1){
            #print 1st file, move it's handle up one
            print "$chr1\t$pos1\t$nMeth1\t$nTot1\n";
            ($chr1, $pos1, $nMeth1, $nTot1) = split ' ', <$in1>;
      }
      #print "1:".<$in1>;
      #print "2:".<$in2>;
}

#one of the files finished, now write out the rest
if (eof $in1){
      while(<$in2>){
            print "$chr2\t$pos2\t$nMeth2\t$nTot2\n";
            ($chr2, $pos2, $nMeth2, $nTot2) = split ' ', <$in2>;
      }
}
else{
      while(<$in1>){
            print "$chr1\t$pos1\t$nMeth1\t$nTot1\n";
            ($chr1, $pos1, $nMeth1, $nTot1) = split ' ', <$in1>;
      }
}

==> ./sanchari/j432_j446/hmcSites_fixed2/runMerge.sh <==

for chr in Y {22..1} X
do
    echo $chr
    Rscript mergeSites.r $chr
done

==> ./sanchari/j432_j446/hmcSites_fixed2/cpg/countsToBedgraph.r <==
options(stringsAsFactors=F)

x = read.table("hmcSites_BOTH_all_cpg_common4.txt")
head(x)


colnames(x) = c("chr", "pos",
      "jd.bs.meth", "jd.bs.tot", "jd.oxbs.meth", "jd.oxbs.tot", "jd.or", "jd.pval",
      "hpne.bs.meth", "hpne.bs.tot", "hpne.oxbs.meth", "hpne.oxbs.tot", "hpne.or",
"hpne.pval"
)

head(x)
```

```
#map the methylated reads to positive(blue)
#unmethylation reads to negative(red)

#do this for BS and OxBS



#positive methylated blue

descr="jd.bs"
col1="jd.bs.meth"
col2="jd.bs.tot"
#show raw counts of converted and non-converted reads
writeBedgraph_v1 <- function(descr, x, col1, col2){
      #turn off scientific notation
      options(scipen=999)

      ofile = paste0(descr, "_meth.bedGraph")
      trackinfo = paste0("track type=bedGraph name=\"",ofile,"\" visibility=full
color=0,0,255 altColor=255,255,255")
      x.meth = cbind(x[,"chr"], x[,"pos"], x[,"pos"]+1, x[,col1])
      write(trackinfo, file = ofile)
      write.table(file= ofile, x.meth, col.names=F, row.names=F, quote=F, sep="\t",
append=T)

      #negative non-methylation red
      ofile = paste0(descr, "_nometh.bedGraph")
      trackinfo = paste0("track type=bedGraph name=\"",ofile,"\" visibility=full
altColor=255,0,0 color=255,255,255")
      x.nometh = cbind(x[,"chr"], x[,"pos"], x[,"pos"]+1, -(x[,col2] - x[,col1]))
      write(trackinfo, file = ofile)
      write.table(file= ofile, x.nometh, col.names=F, row.names=F, quote=F,
sep="\t", append=T)

}


writeBedgraph_v1("jd.bs", x, "jd.bs.meth", "jd.bs.tot")
writeBedgraph_v1("jd.oxbs", x, "jd.oxbs.meth", "jd.oxbs.tot")
writeBedgraph_v1("hpne.bs", x, "hpne.bs.meth", "hpne.bs.tot")
writeBedgraph_v1("hpne.oxbs", x, "hpne.oxbs.meth", "hpne.oxbs.tot")





#now show the hmc and mc estimates as a bedgraph
samp = "jd"
writeBedgraph_v2 <- function(samp, x){
      #separate mc/hmc into two graphs

      #ratio of meth reads to total reads is estiate of methylation
      r.bs = x[,paste0(samp, ".bs.meth")]/x[,paste0(samp, ".bs.tot")]
      r.oxbs = x[,paste0(samp, ".oxbs.meth")]/x[,paste0(samp, ".oxbs.tot")]

      #hist(r.bs, 100)
      #hist(r.oxbs - r.bs, 100)

      #difference of BS ratio and OxBS ratio is estimate for hmc%
      #shown in blue
      ofile = paste0(samp, "_hmc.bedGraph")
      trackinfo = paste0("track type=bedGraph name=\"",ofile,"\" visibility=full
color=0,0,255 altColor=255,255,255")
      x.hmc = cbind(x[,"chr"], x[,"pos"], x[,"pos"]+1, r.bs - r.oxbs)
      write(trackinfo, file = ofile)
      write.table(file= ofile, x.hmc, col.names=F, row.names=F, quote=F, sep="\t",
append=T)


      #ratio of oxBS is estimate for mc% shown in red
      ofile = paste0(samp, "_mc.bedGraph")
      trackinfo = paste0("track type=bedGraph name=\"",ofile,"\" visibility=full
altColor=255,0,0 color=255,255,255")
      x.mc = cbind(x[,"chr"], x[,"pos"], x[,"pos"]+1, -r.oxbs)
      write(trackinfo, file = ofile)
      write.table(file= ofile, x.mc, col.names=F, row.names=F, quote=F, sep="\t",
append=T)

}

writeBedgraph_v2("jd", x)
writeBedgraph_v2("hpne", x)



#take a look at chr19, 15391500 - 15391600
x.19 = x[x[,"chr"] == "chr19",]

ix = (x.19[,"pos"] > 15391500) & (x.19[,"pos"] < 15391600)
x.19[ix,]


trackinfo = paste0("track type=bedGraph name=\"",bname,"\" visibility=full
color=0,0,255 altColor=255,255,255")
head(x[,c("chr", "pos", "jd.bs.meth")])

#negative non-methylation red
trackinfo = paste0("track type=bedGraph name=\"",bname,"\" visibility=full
altColor=255,0,0 color=255,255,255")

head(cbind(x[,c("chr", "pos")], -(x[,"jd.bs.tot"] - x[,"jd.bs.meth"])))

head(x)



==> ./sanchari/j432_j446/hmcSites_fixed2/cpg/mergeSites_cpg.r <==
options(stringsAsFactors=F)

COMMON_THRESH = 4
#combine the hpne and jd samples

jd.files = paste0("hmcSites_JD_",c(1:22, "X", "Y"), "_cpg.txt")
hpne.files = paste0("hmcSites_HPNE_",c(1:22, "X", "Y"), "_cpg.txt")
out.files = paste0("hmcSites_BOTH_",c(1:22, "X", "Y"), "_cpg_common",
COMMON_THRESH, ".txt")
```

```
for (i in 1:length(jd.files)){
    x.jd = read.table(jd.files[i])
    x.hpne = read.table(hpne.files[i])
    #head(x.jd)
    x.both = merge(x.jd, x.hpne, by=2)
    #head(x.both)

    #column names
    # chr
    # pos
    # #meth BS JD
    # #tot BS JD
    # #meth OXBS JD
    # #tot OXBS JD
    # odd ratio JD
    # fisher p-val JD
    # #meth BS HPNE
    # #tot BS HPNE
    # #meth OXBS HPNE
    # #tot OXBS HPNE
    # odd ratio HPNE
    # fisher p-val HPNE
    y = x.both[,c(2, 1, 3:8, 11:16)]
    head(y)
        #  V1.x    V2 V3.x V4.x V5.x V6.x V7.x V8.x V3.y V4.y V5.y V6.y V7.y V8.y
        #1 chr1 12264    3    3    0    1  Inf 0.25    1    1    3    3  NaN  1.0
        #2 chr1 12270    3    3    1    1  NaN 1.00    1    1    3    4  Inf  0.8
        #3 chr1 12277    3    3    1    1  NaN 1.00    0    1    4    4    0  1.0

    #the read counts for the four files
    col.ix = c(4, 6, 10, 12)
    #all four files must meet criteria
    row.ix = apply(y[,col.ix] >= COMMON_THRESH, MARGIN=1, all)

    #head(y[row.ix,])
    #dim(y[row.ix,])
    write.table(y[row.ix,], file=out.files[i], row.names=F, col.names=F, quote=F,
sep="\t")
}
```

```
==> ./sanchari/j432_j446/hmcSites_fixed2/mergeSites.r <==
library("data.table")
options(stringsAsFactors=F)

#read the chrom from command line
```

```
args = commandArgs(trailingOnly=TRUE)

# test if there is at least one argument: if not, return an error
if (length(args)==0) {
  stop("One argument must be supplied (chrom)", call.=FALSE)
} else if (length(args)==1) {
  # default output file
  chrom = args[1]
}


COMMON_THRESH = 4
#combine the hpne and jd samples


jd.file = paste0("hmcSites_JD_",chrom, ".txt")
hpne.file = paste0("hmcSites_HPNE_",chrom, ".txt")
out.file = paste0("hmcSites_BOTH_",chrom, "_common", COMMON_THRESH, ".txt")


xdt.jd = data.table(read.table(jd.file), key="V2")
xdt.hpne = data.table(read.table(hpne.file), key="V2")
xdt.both = merge(xdt.jd, xdt.hpne, by="V2")


y = xdt.both[,c(2, 1, 3:8, 10:15), with=F]
col.ix = c(4, 6, 10, 12)
row.ix = apply(y[,col.ix, with=F] >= COMMON_THRESH, MARGIN=1, all)

write.table(y[row.ix,], file=out.file, row.names=F, col.names=F, quote=F, sep="\t")



##########################3
#jd.files = paste0("hmcSites_JD_",c(1:22, "X", "Y"), ".txt")
#hpne.files = paste0("hmcSites_HPNE_",c(1:22, "X", "Y"), ".txt")
#out.files = paste0("hmcSites_BOTH_",c(1:22, "X", "Y"), "_common", COMMON_THRESH,
".txt")
#
#
#for (i in 1:length(jd.files)){
#    #data.table's should be able to handle the huge matrices
#    #x.jd = read.table(jd.files[i])
#    #x.hpne = read.table(hpne.files[i])
#    #x.both = merge(x.jd, x.hpne, by=2)
#    xdt.jd = data.table(read.table(jd.files[i]), key="V2")
#    xdt.hpne = data.table(read.table(hpne.files[i]), key="V2")
#    xdt.both = merge(xdt.jd, xdt.hpne, by="V2")
#     #dim(xdt.both)
#     #head(xdt.both)
#    #head(x.jd)
#    #head(x.both)
#
#    #column names
#    # chr
#    # pos
#    # #meth BS JD
#    # #tot BS JD
#    # #meth OXBS JD
#    # #tot OXBS JD
#    # odd ratio JD
#    # fisher p-val JD
#    # #meth BS HPNE
#    # #tot BS HPNE
#    # #meth OXBS HPNE
#    # #tot OXBS HPNE
#    # odd ratio HPNE
#    # fisher p-val HPNE
#
#    y = xdt.both[,c(2, 1, 3:8, 10:15), with=F]
#    #y = x.both[,c(2, 1, 3:8, 10:15)]
#    head(y)
#        #  V1.x    V2 V3.x V4.x V5.x V6.x V7.x V8.x V3.y V4.y V5.y V6.y V7.y V8.y
#        #1 chr1 12264    3    3    0    1  Inf 0.25    1    1    3    3  NaN  1.0
#        #2 chr1 12270    3    3    1    1  NaN 1.00    1    1    3    4  Inf  0.8
#        #3 chr1 12277    3    3    1    1  NaN 1.00    0    1    4    4    0  1.0
#
#    #the read counts for the four files
#    col.ix = c(4, 6, 10, 12)
#    #all four files must meet criteria
#    #row.ix = apply(y[,col.ix] >= COMMON_THRESH, MARGIN=1, all)
#    row.ix = apply(y[,col.ix, with=F] >= COMMON_THRESH, MARGIN=1, all)
#
#    #head(y[row.ix,])
#    #dim(y[row.ix,])
#    write.table(y[row.ix,], file=out.files[i], row.names=F, col.names=F, quote=F,
sep="\t")
#}
#
#
#
#
#
#
#
#
#
#
#
#
#
#

==> ./sanchari/j432_j446/hpc_mcSites/getDiffSites.sh <==

#get the sites in JD but not in HPNE
comm -23 --check-order <(cut -f1-2 common4_mcSites_JD.txt | sort) <(cut -f1-2
common4_mcSites_HPNE.txt | sort) >| common4_mcSites_JD-HPNE.txt
comm -13 --check-order <(cut -f1-2 common4_mcSites_JD.txt | sort) <(cut -f1-2
common4_mcSites_HPNE.txt | sort) >| common4_mcSites_HPNE-JD.txt





==> ./sanchari/j432_j446/findHmcSites_stream.r <==
#!/usr/bin/env Rscript
```

```
options(stringsAsFactors=F)

#command line arguments
args = commandArgs(trailingOnly=TRUE)
f.bs = args[1]
f.oxbs = args[2]
f.hmc = args[3]
#f.bs = "conversionCounts/HPNE_BS_22.txt.gz"
#f.oxbs = "conversionCounts/HPNE_OxBS_22.txt.gz"
#f.hmc = "hmcSites_HPNE_22.txt"

print(f.bs)
print(f.oxbs)
print(f.hmc)

#BS
#c -> t
#5mc -> c
#5hmc -> c
#
#oxBS
#c -> t
#5mc -> c
#5hmc -> t

#we want to find the sites with hmc
#low conversion in BS
#high conversion in oxBS


#candidate sites should have
#conversion rate in Bs  below 50%
#conversion rate in oxBs above 50%



#instead of loading everything in at once
#read and process line by line
con1 = pipe(paste0("zcat ", f.bs))
open(con1, "r")
con2 = pipe(paste0("zcat ", f.oxbs))
open(con2, "r")

#start the file pointer at the begining
x1 = read.table(textConnection(readLines(con1, n=1)))
x2 = read.table(textConnection(readLines(con2, n=1)))

#keep processing line by line until there's an error
try({while(T){
      #we should only be working with a single chrom
      if (x1[1,1] != x2[1,1]){
            break
      }

      #matching site, process and print
      if (x1[1,2] == x2[1,2]){
            x = merge(x1, x2, by.x=1, by.y=1)
            v =as.numeric(x[1, c("V3.x", "V4.x", "V3.y", "V4.y")])
            m = matrix((c(v[1], v[3], v[2]-v[1], v[4]-v[3])), ncol=2)
            res = fisher.test(m)

            if (res$estimate < 1 && res$p.value < 0.05){
                  cat(paste(c(as.character(x[1,-5]),
c(odds=res$estimate,pval=res$p.value), "\n"), sep="", collapse="\t"), file=f.hmc,
append=T)
            }
            #move both pointers up
            x1 = read.table(textConnection(readLines(con1, 1)))
            x2 = read.table(textConnection(readLines(con2, 1)))
      }else if(x1[1,2] < x2[1,2]){
            #move the bs file pointer up a spot
            x1 = read.table(textConnection(readLines(con1, 1)))
      }else if (x1[1,2] > x2[1,2]){
            #move the oxbs file pointer up a spot
            x2 = read.table(textConnection(readLines(con2, 1)))
      }
}}, silent=T)

close(con1)
close(con2)




==> ./sanchari/j432_j446/runFindMcSites.sh <==
#!/bin/bash
#$ -cwd
#$ -N findMcSites
#$ -j n
#$ -l h_vmem=1G
#$ -pe smp 1
#$ -S /bin/bash
#$ -m e
#$ -M kith.pradhan@einstein.yu.edu

#run the streaming r script that finds mc sites
#
#usage:
#qsub runFindMcSites.sh HPNE_BS_1.txt.gz HPNE_OxBS_1.txt.gz
mcSites/stream_mcSites_HPNE_1.txt

BSFILE=$1
OXBSFILE=$2
OUTFILE=$3

module load R/3.3.0/gcc.4.4.7

echo $BSFILE
echo $OXBSFILE
echo $OUTFILE

./findMcSites_stream.r $BSFILE $OXBSFILE $OUTFILE

==> ./sanchari/j432_j446/findHmcSites.r <==
#!/usr/bin/env Rscript

options(stringsAsFactors=F)
```

```
#command line arguments
args = commandArgs(trailingOnly=TRUE)
f.bs = args[1]
f.oxbs = args[2]
f.hmc = args[3]
#f.bs = "conversionCounts/HPNE_BS_22.txt.gz"
#f.oxbs = "conversionCounts/HPNE_OxBS_22.txt.gz"
#f.hmc = "hmcSites_HPNE_22.txt"

print(f.bs)
print(f.oxbs)
print(f.hmc)

#BS
#c -> t
#5mc -> c
#5hmc -> c
#
#oxBS
#c -> t
#5mc -> c
#5hmc -> t

#we want to find the sites with hmc
#low conversion in BS
#high conversion in oxBS


#candidate sites should have
#conversion rate in Bs  below 50%
#conversion rate in oxBs above 50%



#read in the count info from the gz files
print(paste0("loading bs:  ", f.bs))
x1 = read.table(pipe(paste0("zcat ", f.bs)))
print(paste0("loading oxbs:  ", f.oxbs))
x2 = read.table(pipe(paste0("zcat ", f.oxbs)))

#head(x1)
#merge the bs and oxbs together
print("merging bs/oxbs")
x = merge(x1, x2, by.x=2, by.y=2)


#for each position calculate probability that
#oxbs conversion is more than bs conversion
#these are the hmc sites

#store data like this will make access faster
vals = t(data.matrix(x[,c("V3.x", "V4.x", "V3.y", "V4.y")]))

print ("testing...")
t1 = Sys.time()
#res.sites = sapply(1:20000, function(i){
res.sites = sapply(1:nrow(x), function(i){
      if (i %% 1000 == 0){
            print(i)
      }
      v = vals[,i]
      #v = unlist(x[i,c("V3.x", "V4.x", "V3.y", "V4.y")])
      m = matrix((c(v[1], v[3], v[2]-v[1], v[4]-v[3])), ncol=2)

      res = fisher.test(m)
      c(odds=res$estimate,pval=res$p.val)
})
t2 = Sys.time()
t2 - t1
dim(res.sites)

#lets just look at the sites with pval < 0.05
#and where oxbs has more conversion than bs
ix = which((res.sites[1,] < 1) & (res.sites[2,] < 0.05))
res = cbind(x[ix, 2:1], x[ix,c("V3.x", "V4.x", "V3.y", "V4.y")], t(res.sites[,ix]))
print (paste0("writing:  ", f.hmc))
write.table(x=res, file=f.hmc, row.names=F, sep="\t")




==> ./sanchari/j432_j446/filterSitesByCount.pl <==
#!/usr/bin/perl
#print the chromosomal locations that have at least k reads
#read a conversionCount file from stdin
#write locations to stdout
#usage:
#   zcat HPNE_BS_3.txt.gz | ./filterSitesByCount.pl 4 > common4_HPNE_BS_3.txt
use strict;
use warnings;

my $k = 0;
if (scalar @ARGV > 0){
      $k = $ARGV[0];
      #print "k:  $k\n";
}else{
      die "must supply read count threshold argument\n";
}

while(<STDIN>){
      #my $line = $_;
      my @vals = split;
      if ($vals[3] >= $k){
            #print $line;
            print "$vals[0]\t$vals[1]\n";
      }
}

==> ./sanchari/j432_j446/hmcSites_fixed/cpg/mergeSites_cpg.r <==
options(stringsAsFactors=F)

COMMON_THRESH = 4
#combine the hpne and jd samples

jd.files = paste0("hmcSites_JD_",c(1:22, "X", "Y"), "_cpg.txt")
hpne.files = paste0("hmcSites_HPNE_",c(1:22, "X", "Y"), "_cpg.txt")
out.files = paste0("hmcSites_BOTH_",c(1:22, "X", "Y"), "_cpg_common",
COMMON_THRESH, ".txt")


for (i in 1:length(jd.files)){
    x.jd = read.table(jd.files[i])
    x.hpne = read.table(hpne.files[i])
    #head(x.jd)
    x.both = merge(x.jd, x.hpne, by=2)
    #head(x.both)

    #column names
    # chr
    # pos
    # #meth BS JD
    # #tot BS JD
    # #meth OXBS JD
    # #tot OXBS JD
    # odd ratio JD
    # fisher p-val JD
    # #meth BS HPNE
    # #tot BS HPNE
    # #meth OXBS HPNE
    # #tot OXBS HPNE
    # odd ratio HPNE
    # fisher p-val HPNE
    y = x.both[,c(2, 1, 3:8, 11:16)]
    head(y)
        #  V1.x    V2 V3.x V4.x V5.x V6.x V7.x V8.x V3.y V4.y V5.y V6.y V7.y V8.y
        #1 chr1 12264    3    3    0    1  Inf 0.25    1    1    3    3  NaN  1.0
        #2 chr1 12270    3    3    1    1  NaN 1.00    1    1    3    4  Inf  0.8
        #3 chr1 12277    3    3    1    1  NaN 1.00    0    1    4    4    0  1.0

    #the read counts for the four files
    col.ix = c(4, 6, 10, 12)
    #all four files must meet criteria
    row.ix = apply(y[,col.ix] >= COMMON_THRESH, MARGIN=1, all)

    #head(y[row.ix,])
    #dim(y[row.ix,])
    write.table(y[row.ix,], file=out.files[i], row.names=F, col.names=F, quote=F,
sep="\t")
}
```

```
==> ./sanchari/j432_j446/hmcSites_fixed/cpg/examineSiteDistributions.r <==
library("VariantAnnotation")
library(TxDb.Hsapiens.UCSC.hg19.knownGene)
txdb_hg19 <- TxDb.Hsapiens.UCSC.hg19.knownGene

options(stringsAsFactors=F)



x = read.table("hmcSites_BOTH_all_cpg_common4.txt")


#look at the hmcsites with pval < 0.05
head(x)
png("jd_methylation_hist.png")
hist(x[,3]/x[,4], 100, xlab="methylation percentage")
dev.off()
mean(x[,3]/x[,4] > .50)
mean(x[,3]/x[,4] == 1)
dim(x)
#take the top 10000 with the most methylatiojh

ix = order(x[,3]/x[,4], x[,4])
#ix.mc2 = tail(ix, 10000)
ix.mc = tail(ix, 100000)
head(x[ix,])
ix.mc = sort(sample(which(x[,3]/x[,4] > .50), 10000 ))

head(x[ix.mc,])
tail(x[ix.mc,])
y = x[ix.mc,1:2]
y[,3] = "+"
write.table(y, file="test1_jdmc_top100k.txt", col.names=F, row.names=F, sep="\t",
quote=F)

jd.hmc.ix = which(x[,8] < 0.01)
hpne.hmc.ix = which(x[,14] < 0.01)
length(jd.hmc.ix)
length(hpne.hmc.ix)

ov.hmc.ix = jd.hmc.ix %in% hpne.hmc.ix

sum(ov.hmc.ix)
head(x[jd.hmc.ix,])

y = x[jd.hmc.ix,1:2]
y[,3] = "+"
write.table(y, file="test1_jdhmc.txt", col.names=F, row.names=F, sep="\t", quote=F)

#write a set of random sites

jd.bad.ix = which(x[,8] > 0.91)
length(jd.bad.ix)
jd.bad.ix = sample(jd.bad.ix, length(jd.hmc.ix))
length(jd.bad.ix)
y = x[jd.bad.ix,1:2]
y[,3] = "+"
write.table(y, file="test1_jdbad.txt", col.names=F, row.names=F, sep="\t", quote=F)
```

```
#look at sites with high methylation





getSiteCounts <- function(x){
    input = GRanges(seqnames=x[,1], ranges=IRanges(x[,2], x[,2]+1), strand="*")
    loc_hg19 <- locateVariants(input, txdb_hg19, AllVariants())
    loc_hg19 = loc_hg19[!duplicated(loc_hg19)]
    table(loc_hg19$LOCATION)
}


sCounts.1 = getSiteCounts(x[jd.hmc.ix,1:2])
sCounts.1a = getSiteCounts(x[jd.bad.ix,1:2])
sCounts.2 = getSiteCounts(x[hpne.hmc.ix,1:2])

barplot(sCounts.1)
x11(); barplot(sCounts.1a)
x11(); barplot(sCounts.2)



==> ./sanchari/j432_j446/hmcSites_fixed/mergeSites.r <==
library("data.table")
options(stringsAsFactors=F)

COMMON_THRESH = 4
#combine the hpne and jd samples

jd.files = paste0("hmcSites_JD_",c(1:22, "X", "Y"), ".txt")
hpne.files = paste0("hmcSites_HPNE_",c(1:22, "X", "Y"), ".txt")
out.files = paste0("hmcSites_BOTH_",c(1:22, "X", "Y"), "_common", COMMON_THRESH,
".txt")


for (i in 1:length(jd.files)){
    #data.table's should be able to handle the huge matrices
    x.jd = read.table(jd.files[i])
    x.hpne = read.table(hpne.files[i])
    x.both = merge(x.jd, x.hpne, by=2)
    xdt.jd = data.table(read.table(jd.files[i]), key="V2")
    xdt.hpne = data.table(read.table(hpne.files[i]), key="V2")
    xdt.both = merge(xdt.jd, xdt.hpne, by="V2")
    #head(x.jd)
    #head(x.both)

    #column names
    # chr
    # pos
    # #meth BS JD
    # #tot BS JD
    # #meth OXBS JD
    # #tot OXBS JD
    # odd ratio JD
    # fisher p-val JD
    # #meth BS HPNE
    # #tot BS HPNE
    # #meth OXBS HPNE
    # #tot OXBS HPNE
    # odd ratio HPNE
    # fisher p-val HPNE
    head(x.both)
    dim(x.both)

    x.both[1:5, 2, with=F]
    y = x.both[,c(2, 1, 3:8, 10:15), with=F]
    y = x.both[,c(2, 1, 3:8, 10:15)]
    head(y)
        #  V1.x    V2 V3.x V4.x V5.x V6.x V7.x V8.x V3.y V4.y V5.y V6.y V7.y V8.y
        #1 chr1 12264    3    3    0    1  Inf 0.25    1    1    3    3  NaN  1.0
        #2 chr1 12270    3    3    1    1  NaN 1.00    1    1    3    4  Inf  0.8
        #3 chr1 12277    3    3    1    1  NaN 1.00    0    1    4    4    0  1.0

    #the read counts for the four files
    col.ix = c(4, 6, 10, 12)
    #all four files must meet criteria
    row.ix = apply(y[,col.ix] >= COMMON_THRESH, MARGIN=1, all)

    #head(y[row.ix,])
    #dim(y[row.ix,])
    write.table(y[row.ix,], file=out.files[i], row.names=F, col.names=F, quote=F,
sep="\t")
}
```

```
==> ./sanchari/j432_j446/annotateSites.r <==
library(GenomicFeatures)
library(BSgenome.Hsapiens.UCSC.hg19)
library(VariantAnnotation)
library(TxDb.Hsapiens.UCSC.hg19.knownGene) # for annotation
library(org.Hs.eg.db)
library("biomaRt")
library("stringr")

options(stringsAsFactors=F)

#load up the database utlility, find the chrom pos and ids for the significant diff
expressed locatinos
grch37 = useMart(biomart="ENSEMBL_MART_ENSEMBL", host="grch37.ensembl.org",
path="/biomart/martservice", dataset="hsapiens_gene_ensembl")
#grch37 = useMart(biomart="ENSEMBL_MART_ENSEMBL",
host="feb2014.archive.ensembl.org", path="/biomart/martservice",
dataset="hsapiens_gene_ensembl")


#find the sites with gain/loss hmc from hpne to jd
#are there any matching sites?
#Where do these occur?  exons, introns, promoters.

#in depth look at brd4

#over chr19
#> dim(sites.hpne)
#[1] 173149      8
#> dim(sites.jd)
#[1] 200834      8
#> dim(peaks.jd)
#[1] 117180      2
#> dim(peaks.hpne)
#[1] 66119     2




#load in the hmcsites
sites.hpne = read.table("hmcSites/common4_hmcSites_HPNE.txt", header=F)
sites.jd = read.table("hmcSites/common4_hmcSites_JD.txt", header=F)



#sites.hpne = read.table(paste0("hmcSites/hmcSites_HPNE_",chrom, ".txt"), header=T)
#sites.jd = read.table(paste0("hmcSites/hmcSites_JD_", chrom, ".txt"), header=T)
#dim(sites.hpne)
#head(sites.hpne)
#dim(sites.jd)
#
#sites.hpne = read.table("hmcSites/hmcSites_HPNE_all.txt", header=T)
#sites.jd = read.table("hmcSites/hmcSites_JD_all.txt", header=T)


#load in the peak sites from the atac-seq run
peaks.hpne =
read.table("~/hpc_home/projects/sanchari/j455/macs_Chpne_sorted_mdup_Tjd_sorted_mdu
p/Chpne_sorted_mdup_Tjd_sorted_mdup_summits.bed", header=F)
peaks.jd =
read.table("~/hpc_home/projects/sanchari/j455/macs_Cjd_sorted_mdup_Thpne_sorted_mdu
p/Cjd_sorted_mdup_Thpne_sorted_mdup_summits.bed", header=F)
peaks.hpne =
read.table("~/hpc_home/projects/sanchari/j455/macs_hpne_sorted_mdup/hpne_sorted_mdu
p_summits.bed", header=F)
peaks.jd =
read.table("~/hpc_home/projects/sanchari/j455/macs_jd_sorted_mdup/jd_sorted_mdup_su
mmits.bed", header=F)
peaks.hpne = peaks.hpne[,1:2]
peaks.jd = peaks.jd[,1:2]
dim(peaks.jd)
dim(peaks.hpne)

#https://adairama.wordpress.com/2013/02/15/functionally-annotate-snps-and-indels-
in-bioconductor/
annotateList <- function(sites){
      #get the chromosomal location of the sites
      input = sites[, 1:2]
      colnames(input) <- c("chr", "pos")
      input$pos       <- as.numeric(as.character(input$pos))

      #construct a range for each site.
      target <- with(input, GRanges(
            seqnames = Rle(chr),
            ranges   = IRanges(
                  pos,
                end=pos,
                names=paste0("rand", 1:nrow(input))
            ),
            strand   = Rle(strand("*"))
      ))

      #find variants for each site
      loc <- locateVariants(target, TxDb.Hsapiens.UCSC.hg19.knownGene, 
AllVariants())
      names(loc) <- NULL
      out <- as.data.frame(loc)
      out$names <- names(target)[ out$QUERYID ]
      out <- out[ , c("names", "seqnames", "start", "end", "LOCATION", "GENEID")]
      out <- unique(out)

      #convert gene id to gene symbol
      Symbol2id <- as.list( org.Hs.egSYMBOL2EG )
      id2Symbol <- rep( names(Symbol2id), sapply(Symbol2id, length) )
      names(id2Symbol) <- unlist(Symbol2id)

      #make sure all entries have been found
      x <- unique( with(out, c(levels(GENEID))) )
      table( x %in% names(id2Symbol) ) # good, all found

      #assign the gene symbol
      out$GENESYMBOL <- id2Symbol[ as.character(out$GENEID) ]
      out
}


#look at a more conservative list of hmc events
head(sites.hpne)
sum(sites.hpne$pval < 0.001)
sites.hpne.sig  = sites.hpne[sites.hpne$pval < 0.001,]
sites.jd.sig  = sites.jd[sites.jd$pval < 0.001,]


#look at a random distribution of sites taken from min/max range of chromosome
sites.rand = sites.jd
max(sites.rand[,2])
sites.rand[,2] = runif(
      nrow(sites.rand),
      min(sites.rand[,2]),
      max(sites.rand[,2]) - min(sites.rand[,2])
)


panno.hpne = annotateList(peaks.hpne)
panno.jd = annotateList(peaks.jd)



#annoation the hmcSites
#break up long list into multiple queries
#takes a long time
chunk <- function(x,n) split(x, factor(sort(rank(x)%%n)))
ix = chunk(1:nrow(sites.hpne), 10)
anno.hpne = lapply(ix, function(i){
      print (i[1])
      annotateList(sites.hpne[i,])
})
x.hpne=do.call(rbind, anno.hpne)

ix = chunk(1:nrow(sites.jd), 10)
anno.jd = lapply(ix, function(i){
      print (i[1])
      annotateList(sites.jd[i,])
})
x.jd=do.call(rbind, anno.jd)

#save.image("anno_v1.RData")
makePie <- function(x, descr){
      mytable <- table(x[,5])
      #lbls <- paste(names(mytable), "\n", mytable, sep="")
      lbls <- paste(mytable, " (", format(round(100*mytable/sum(mytable), 4),
nsmall=4), "%)", sep="")
      cols = rainbow(length(lbls))
      pie(mytable, labels = lbls, col=cols, main=descr)
      legend("topright", names(mytable), col=cols, pch=15)
      mytable
}

png("pie_hmcHpne_all.png")
makePie(x.hpne, "hmc sites: HPNE")
dev.off()
png("pie_hmcJD_all.png")
makePie(x.jd, "hmc sites: JD")
dev.off()

#how many genes are affected?
head(x.jd)
ugenes.jd = unique(x.jd$GENESYMBOL)
length(ugenes.jd)
ugenes.hpne = unique(x.hpne$GENESYMBOL)
length(ugenes.hpne)


#genes with hmc in one sample, but not the other
length(ugenes.hpne[!(ugenes.hpne %in% ugenes.jd)])
length(ugenes.jd[!(ugenes.jd %in% ugenes.hpne)])






anno.hpne = annotateList(sites.hpne)
anno.jd = annotateList(sites.jd)
anno.rand = annotateList(sites.rand)
anno.hpne.sig = annotateList(sites.hpne.sig)
anno.jd.sig = annotateList(sites.jd.sig)
dim(sites.hpne)
dim(sites.jd)
dim(sites.hpne.sig)
dim(sites.jd.sig)


#briefly look at hmc events at brd4
#save in bed format, view in igv
ix.1 = which(anno.hpne$GENESYMBOL == "BRD4")
ix.2 = which(anno.jd$GENESYMBOL == "BRD4")
length(ix.1)
length(ix.2)

anno.hpne[ix.1,]
x = anno.hpne[ix.1,2:4]
x = cbind(x, 1, 1)
write.table(file="hmc_brd4_hpne.bed", x=x, col.names=F, row.names=F, quote=F)
x = anno.jd[ix.2,2:4]
x = cbind(x, 1, 1)
write.table(file="hmc_brd4_jd.bed", x=x, col.names=F, row.names=F, quote=F)
#add a header to each bed file
# track name=test1 description="hmc" visibility=2


png("atacseq_hpne.png")
mytable <- table(panno.hpne[,5])
lbls <- paste(names(mytable), "\n", mytable, sep="")
pie(mytable, labels = lbls,
   main="ATACseq: HPNE")
dev.off()

png("atacseq_jd.png")
mytable <- table(panno.jd[,5])
lbls <- paste(names(mytable), "\n", mytable, sep="")
pie(mytable, labels = lbls,
   main="ATACseq: JD")
dev.off()


#par(mfcol=c(1, 2))
png("hmc_chr19_hpne.png")
mytable <- table(anno.hpne[,5])
lbls <- paste(names(mytable), "\n", mytable, sep="")
pie(mytable, labels = lbls,
   main="chr19 HMC: HPNE")
dev.off()

png("hmc_chr19_jd.png")
mytable <- table(anno.jd[,5])
lbls <- paste(names(mytable), "\n", mytable, sep="")
pie(mytable, labels = lbls,
   main="chr19 HMC: JD")
dev.off()

png("hmc_chr19_hpneSig.png")
mytable <- table(anno.hpne.sig[,5])
```

```
lbls <- paste(names(mytable), "\n", mytable, sep="")
pie(mytable, labels = lbls,
   main="chr19 HMC: HPNEsig")
dev.off()

png("hmc_chr19_jdSig.png")
mytable <- table(anno.jd.sig[,5])
lbls <- paste(names(mytable), "\n", mytable, sep="")
pie(mytable, labels = lbls,
   main="chr19 HMC: JDsig")
dev.off()

png("hmc_chr19_random.png")
mytable <- table(anno.rand[,5])
lbls <- paste(names(mytable), "\n", mytable, sep="")
pie(mytable, labels = lbls,
   main="chr19 HMC: random")
dev.off()

#doesn't look like there's much differenc in percentage
#on exonic/intron/promoter regions between hpne, jd
#random sites have slighter more intergenic regions


#might have to filter out some of the hmc sites based
#on direction of the gene
#wehther the reference base is C->T or G->A


#maybe look at the mutation impact factors?
#with snpEff?
#or with VEP

#Lets look at the hmc sites at brd4
#prepare teh vep file

sites = anno.hpne[ix.1,2:4]


anno.hpne[ix.1,]
x = anno.hpne[ix.1,2:4]




x = cbind(x, 1, 1)
write.table(file="hmc_brd4_hpne.bed", x=x, col.names=F, row.names=F, quote=F)
x = anno.jd[ix.2,2:4]


dat =
GRanges(seqnames=rep('chr17',3),ranges=IRanges(start=c(45229228,45229234,45234706),
width=1))
varallele = DNAStringSet(c('C','C','C'))
db = TxDb.Hsapiens.UCSC.hg19.knownGene

results = predictCoding(dat,db,Hsapiens,varallele)


sites = sites.hpne
```

```
input = sites[, 1:2]
colnames(input) <- c("chr", "pos")
input$pos       <- as.numeric(as.character(input$pos))
input = input[1:3000,]

target <- with(input, GRanges(
      seqnames = Rle(chr),
      ranges   = IRanges(
            pos,
            width=1,
            names=paste0("rand", 1:nrow(input))
      ),
      strand   = Rle(strand("*"))
))
length(target)
varallele = DNAStringSet(rep('C',length(target)))
results = predictCoding(target,db,Hsapiens,varallele)






#############################################
"chromosome_name", "start", "end"
x = list(chromosome_name="19", start="15236836", end="15332543")

x = head(sites.hpne[,c(1,2,2)], 300)
x = list(chromosome_name=gsub(x[,1], pat="chr(.*)", rep="\\1", perl=T),
start=x[,2], end=x[,3])
x = getBM(attributes=c("ensembl_gene_id", "hgnc_symbol", "chromosome_name",
"start_position", "end_position", "go_biological_process_linkage_type"), filters =
c("chromosome_name", "start", "end"), values = x, mart=grch37)


==> ./sanchari/j432_j446/commands3_percHist.sh <==
#find the percetages of hmc at each selected site

cat hmcSites/hmcSites_JD_all.txt | awk '{print $5/$6 - $3/$4}' > hmcPerc_JD.txt
cat hmcSites/hmcSites_HPNE_all.txt | awk '{print $5/$6 - $3/$4}' > hmcPerc_HPNE.txt


#R code
x = read.table("hmcPerc_JD.txt")
x = x[,1]
png("hmcPerc_JD.png")
hist(x, 100, main="hmC Percentage of Selected Sites: JD")
dev.off()

x = read.table("hmcPerc_HPNE.txt")
x = x[,1]
png("hmcPerc_HPNE.png")
hist(x, 100, main="hmC Percentage of Selected Sites: HPNE")
dev.off()



join <(zcat conversionCounts/HPNE_BS_22.txt.gz | cut -f2-4) <(zcat
conversionCounts/HPNE_OxBS_22.txt.gz | cut -f2-4) \
 | awk '{if ($3 >= 4 && $5 >= 4) print $4/$5 - $2/$3}' > commonPerc_HPNE22.txt


#r code
x = read.table("commonPerc_HPNE22.txt")
x = x[,1]
png("commonPerc_HPNE22.png")
hist(x, 100, main="hmC Percentage of Common Sites: HPNE chr22")
dev.off()
```

```
==> ./sanchari/j432_j446/findHmcSites.sh <==
#!/bin/bash
#$ -cwd
#$ -j n
#$ -l h_vmem=1G
#$ -pe smp 1
#$ -S /bin/bash
#$ -m e
#   #$ -M kith.pradhan@einstein.yu.edu
module load python/2.7.8/gcc.4.4.7
module load scipy/0.14.0/python.2.7.8

oxFile=$1
oxbsFile=$2
outFile=$3


python findHmcSites.py $oxFile $oxbsFile $outFile
```

```
==> ./sanchari/j432_j446/annotateSites_mc.r <==
library(GenomicFeatures)
library(BSgenome.Hsapiens.UCSC.hg19)
library(VariantAnnotation)
library(TxDb.Hsapiens.UCSC.hg19.knownGene) # for annotation
library(org.Hs.eg.db)
library("biomaRt")
library("stringr")

options(stringsAsFactors=F)

#load up the database utlility, find the chrom pos and ids for the significant diff
expressed locatinos
grch37 = useMart(biomart="ENSEMBL_MART_ENSEMBL", host="grch37.ensembl.org",
path="/biomart/martservice", dataset="hsapiens_gene_ensembl")
#grch37 = useMart(biomart="ENSEMBL_MART_ENSEMBL",
host="feb2014.archive.ensembl.org", path="/biomart/martservice",
dataset="hsapiens_gene_ensembl")


#find the sites with gain/loss hmc from hpne to jd
```

```
#are there any matching sites?
#Where do these occur?  exons, introns, promoters.

#in depth look at brd4

#over chr19
#> dim(sites.hpne)
#[1] 173149      8
#> dim(sites.jd)
#[1] 200834      8
#> dim(peaks.jd)
#[1] 117180      2
#> dim(peaks.hpne)
#[1] 66119     2




#load in the hmcsites
sites.hpne = read.table("hpc_mcSites/common4_mcSites_HPNE.txt", header=F)
sites.jd = read.table("hpc_mcSites/common4_mcSites_JD.txt", header=F)



#sites.hpne = read.table(paste0("hmcSites/hmcSites_HPNE_",chrom, ".txt"), header=T)
#sites.jd = read.table(paste0("hmcSites/hmcSites_JD_", chrom, ".txt"), header=T)
#dim(sites.hpne)
#head(sites.hpne)
#dim(sites.jd)
#
#sites.hpne = read.table("hmcSites/hmcSites_HPNE_all.txt", header=T)
#sites.jd = read.table("hmcSites/hmcSites_JD_all.txt", header=T)



#https://adairama.wordpress.com/2013/02/15/functionally-annotate-snps-and-indels-
in-bioconductor/
annotateList <- function(sites){

      #get the chromosomal location of the sites
      input = sites[, 1:2]
      colnames(input) <- c("chr", "pos")
      input$pos       <- as.numeric(as.character(input$pos))

      #construct a range for each site.
      target <- with(input, GRanges(
            seqnames = Rle(chr),
            ranges   = IRanges(
                  pos,
                end=pos,
                names=paste0("rand", 1:nrow(input))
            ),
            strand   = Rle(strand("*"))
      ))

      #find variants for each site
      loc <- locateVariants(target, TxDb.Hsapiens.UCSC.hg19.knownGene,
AllVariants())
      names(loc) <- NULL
      out <- as.data.frame(loc)
      out$names <- names(target)[ out$QUERYID ]
      out <- out[ , c("names", "seqnames", "start", "end", "LOCATION", "GENEID")]
      out <- unique(out)

      #convert gene id to gene symbol
      Symbol2id <- as.list( org.Hs.egSYMBOL2EG )
      id2Symbol <- rep( names(Symbol2id), sapply(Symbol2id, length) )
      names(id2Symbol) <- unlist(Symbol2id)

      #make sure all entries have been found
      x <- unique( with(out, c(levels(GENEID))) )
      table( x %in% names(id2Symbol) ) # good, all found

      #assign the gene symbol
      out$GENESYMBOL <- id2Symbol[ as.character(out$GENEID) ]
      out
}



#look at a random distribution of sites taken from min/max range of chromosome
sites.rand = sites.jd
max(sites.rand[,2])
sites.rand[,2] = runif(
      nrow(sites.rand),
      min(sites.rand[,2]),
      max(sites.rand[,2]) - min(sites.rand[,2])
)




#break this up...use just 10% of the sites
#to speed things up
#annoation the mcSites
#break up long list into multiple queries
#takes a long time
chunk <- function(x,n) split(x, factor(sort(rank(x)%%n)))
P = 0.1
ix.small = sort(sample(1:nrow(sites.hpne), nrow(sites.hpne)*P))
smallsites.hpne = sites.hpne[ix.small,]
ix = chunk(1:nrow(smallsites.hpne), 10)
anno.hpne = lapply(ix, function(i){
      print (i[1])
      annotateList(smallsites.hpne[i,])
})
x.hpne=do.call(rbind, anno.hpne)

ix.small = sort(sample(1:nrow(sites.jd), nrow(sites.jd)*P))
smallsites.jd = sites.jd[ix.small,]
ix = chunk(1:nrow(smallsites.jd), 10)
anno.jd = lapply(ix, function(i){
      print (i[1])
      annotateList(smallsites.jd[i,])
})
x.jd=do.call(rbind, anno.jd)

save.image("mc_anno_v1.RData")
```

```
makePie <- function(x, descr){
      mytable <- table(x[,5])
      #lbls <- paste(names(mytable), "\n", mytable, sep="")
      lbls <- paste(mytable, " (", format(round(100*mytable/sum(mytable), 4),
nsmall=4), "%)", sep="")
      cols = rainbow(length(lbls))
      pie(mytable, labels = lbls, col=cols, main=descr)
      legend("topright", names(mytable), col=cols, pch=15)
      mytable
}

png("pie_mc_Hpne_all.png")
makePie(x.hpne, "HPNE mc sites")
dev.off()
png("pie_mc_JD_all.png")
makePie(x.jd, "JD mc sites")
dev.off()

#how many genes are affected?
head(x.jd)
ugenes.jd = unique(x.jd$GENESYMBOL)
length(ugenes.jd)
ugenes.hpne = unique(x.hpne$GENESYMBOL)
length(ugenes.hpne)


#genes with hmc in one sample, but not the other
length(ugenes.hpne[!(ugenes.hpne %in% ugenes.jd)])
length(ugenes.jd[!(ugenes.jd %in% ugenes.hpne)])






anno.hpne = annotateList(sites.hpne)
anno.jd = annotateList(sites.jd)
anno.rand = annotateList(sites.rand)
anno.hpne.sig = annotateList(sites.hpne.sig)
anno.jd.sig = annotateList(sites.jd.sig)
dim(sites.hpne)
dim(sites.jd)
dim(sites.hpne.sig)
dim(sites.jd.sig)


#briefly look at hmc events at brd4
#save in bed format, view in igv
ix.1 = which(anno.hpne$GENESYMBOL == "BRD4")
ix.2 = which(anno.jd$GENESYMBOL == "BRD4")
length(ix.1)
length(ix.2)

anno.hpne[ix.1,]
x = anno.hpne[ix.1,2:4]
x = cbind(x, 1, 1)
write.table(file="hmc_brd4_hpne.bed", x=x, col.names=F, row.names=F, quote=F)
x = anno.jd[ix.2,2:4]
x = cbind(x, 1, 1)
```

```
write.table(file="hmc_brd4_jd.bed", x=x, col.names=F, row.names=F, quote=F)
#add a header to each bed file
# track name=test1 description="hmc" visibility=2


png("atacseq_hpne.png")
mytable <- table(panno.hpne[,5])
lbls <- paste(names(mytable), "\n", mytable, sep="")
pie(mytable, labels = lbls,
   main="ATACseq: HPNE")
dev.off()

png("atacseq_jd.png")
mytable <- table(panno.jd[,5])
lbls <- paste(names(mytable), "\n", mytable, sep="")
pie(mytable, labels = lbls,
   main="ATACseq: JD")
dev.off()


#par(mfcol=c(1, 2))
png("hmc_chr19_hpne.png")
mytable <- table(anno.hpne[,5])
lbls <- paste(names(mytable), "\n", mytable, sep="")
pie(mytable, labels = lbls,
   main="chr19 HMC: HPNE")
dev.off()

png("hmc_chr19_jd.png")
mytable <- table(anno.jd[,5])
lbls <- paste(names(mytable), "\n", mytable, sep="")
pie(mytable, labels = lbls,
   main="chr19 HMC: JD")
dev.off()

png("hmc_chr19_hpneSig.png")
mytable <- table(anno.hpne.sig[,5])
lbls <- paste(names(mytable), "\n", mytable, sep="")
pie(mytable, labels = lbls,
   main="chr19 HMC: HPNEsig")
dev.off()

png("hmc_chr19_jdSig.png")
mytable <- table(anno.jd.sig[,5])
lbls <- paste(names(mytable), "\n", mytable, sep="")
pie(mytable, labels = lbls,
   main="chr19 HMC: JDsig")
dev.off()

png("hmc_chr19_random.png")
mytable <- table(anno.rand[,5])
lbls <- paste(names(mytable), "\n", mytable, sep="")
pie(mytable, labels = lbls,
   main="chr19 HMC: random")
dev.off()

#doesn't look like there's much differenc in percentage
#on exonic/intron/promoter regions between hpne, jd
#random sites have slighter more intergenic regions
```

```
#might have to filter out some of the hmc sites based
#on direction of the gene
#wehther the reference base is C->T or G->A


#maybe look at the mutation impact factors?
#with snpEff?
#or with VEP

#Lets look at the hmc sites at brd4
#prepare teh vep file

sites = anno.hpne[ix.1,2:4]


anno.hpne[ix.1,]
x = anno.hpne[ix.1,2:4]



x = cbind(x, 1, 1)
write.table(file="hmc_brd4_hpne.bed", x=x, col.names=F, row.names=F, quote=F)
x = anno.jd[ix.2,2:4]


dat =
GRanges(seqnames=rep('chr17',3),ranges=IRanges(start=c(45229228,45229234,45234706),
width=1))
varallele = DNAStringSet(c('C','C','C'))
db = TxDb.Hsapiens.UCSC.hg19.knownGene

results = predictCoding(dat,db,Hsapiens,varallele)


sites = sites.hpne
input = sites[, 1:2]
colnames(input) <- c("chr", "pos")
input$pos       <- as.numeric(as.character(input$pos))
input = input[1:3000,]

target <- with(input, GRanges(
      seqnames = Rle(chr),
      ranges   = IRanges(
            pos,
            width=1,
            names=paste0("rand", 1:nrow(input))
      ),
      strand   = Rle(strand("*"))
))
length(target)
varallele = DNAStringSet(rep('C',length(target)))
results = predictCoding(target,db,Hsapiens,varallele)
```

```
#############################################
"chromosome_name", "start", "end"
x = list(chromosome_name="19", start="15236836", end="15332543")

x = head(sites.hpne[,c(1,2,2)], 300)
x = list(chromosome_name=gsub(x[,1], pat="chr(.*)", rep="\\1", perl=T),
start=x[,2], end=x[,3])
x = getBM(attributes=c("ensembl_gene_id", "hgnc_symbol", "chromosome_name",
"start_position", "end_position", "go_biological_process_linkage_type"), filters =
c("chromosome_name", "start", "end"), values = x, mart=grch37)


==> ./sanchari/j432_j446/annotateSites_fig1.r <==
library(GenomicFeatures)
library(BSgenome.Hsapiens.UCSC.hg19)
library(VariantAnnotation)
library(TxDb.Hsapiens.UCSC.hg19.knownGene) # for annotation
library(org.Hs.eg.db)
library("biomaRt")
library("stringr")

options(stringsAsFactors=F)

#load up the database utlility, find the chrom pos and ids for the significant diff
expressed locatinos
grch37 = useMart(biomart="ENSEMBL_MART_ENSEMBL", host="grch37.ensembl.org",
path="/biomart/martservice", dataset="hsapiens_gene_ensembl")
#grch37 = useMart(biomart="ENSEMBL_MART_ENSEMBL",
host="feb2014.archive.ensembl.org", path="/biomart/martservice",
dataset="hsapiens_gene_ensembl")






#https://adairama.wordpress.com/2013/02/15/functionally-annotate-snps-and-indels-
in-bioconductor/
annotateList <- function(sites){

      #get the chromosomal location of the sites
      input = sites[, 1:2]
      colnames(input) <- c("chr", "pos")
      input$pos       <- as.numeric(as.character(input$pos))

      #construct a range for each site.
      target <- with(input, GRanges(
            seqnames = Rle(chr),
            ranges   = IRanges(
                  pos,
                end=pos,
                names=paste0("rand", 1:nrow(input))
            ),
            strand   = Rle(strand("*"))
      ))

      #find variants for each site
      loc <- locateVariants(target, TxDb.Hsapiens.UCSC.hg19.knownGene,
AllVariants())
      names(loc) <- NULL
      out <- as.data.frame(loc)
      out$names <- names(target)[ out$QUERYID ]
      out <- out[ , c("names", "seqnames", "start", "end", "LOCATION", "GENEID")]
      out <- unique(out)

      #convert gene id to gene symbol
      Symbol2id <- as.list( org.Hs.egSYMBOL2EG )
      id2Symbol <- rep( names(Symbol2id), sapply(Symbol2id, length) )
      names(id2Symbol) <- unlist(Symbol2id)

      #make sure all entries have been found
      x <- unique( with(out, c(levels(GENEID))) )
      table( x %in% names(id2Symbol) ) # good, all found

      #assign the gene symbol
      out$GENESYMBOL <- id2Symbol[ as.character(out$GENEID) ]
      out
}


makePie <- function(x, descr){
      mytable <- table(x[,5])
      #lbls <- paste(names(mytable), "\n", mytable, sep="")
      lbls <- paste(mytable, " (", format(round(100*mytable/sum(mytable), 4),
nsmall=4), "%)", sep="")
      cols = rainbow(length(lbls))
      pie(mytable, labels = lbls, col=cols, main=descr)
      legend("topright", names(mytable), col=cols, pch=15)
      mytable
}



#load in the sites
fol = "/home/kpradhan/Desktop/projects/sanchari/homer_20160708"
atac.hpne = read.table(paste0(fol, "/atac_hpne.bed"), header=F)
atac.jd = read.table(paste0(fol, "/atac_jd.bed"), header=F)
atac.hpne.only = read.table(paste0(fol, "/atac_hpne_only.bed"), header=F)
atac.jd.only = read.table(paste0(fol, "/atac_jd_only.bed"), header=F)

hmc.hpne.only = read.table(paste0(fol, "/hmc_hpne_only.bed"), header=F)
hmc.jd.only = read.table(paste0(fol, "/hmc_jd_only.bed"), header=F)

mc.hpne.only = read.table(paste0(fol, "/mc_hpne_only.bed"), header=F)
mc.jd.only = read.table(paste0(fol, "/mc_jd_only.bed"), header=F)


atac.hpne.anno = annotateList(atac.hpne)
atac.jd.anno = annotateList(atac.jd)

atac.hpne.only.anno = annotateList(atac.hpne.only)
atac.jd.only.anno = annotateList(atac.jd.only)
hmc.hpne.only.anno = annotateList(hmc.hpne.only)
hmc.jd.only.anno = annotateList(hmc.jd.only)
mc.hpne.only.anno = annotateList(mc.hpne.only)
mc.jd.only.anno = annotateList(mc.jd.only)
```

```
png("pie_atac_hpne.png")
makePie(atac.hpne.anno, "atac_hpne")
dev.off()
png("pie_atac_jd.png")
makePie(atac.jd.anno, "atac_jd")
dev.off()

png("pie_atac_hpne_only.png")
makePie(atac.hpne.only.anno, "atac_hpne_only")
dev.off()
png("pie_atac_jd_only.png")
makePie(atac.jd.only.anno, "atac_jd_only")
dev.off()

png("pie_hmc_hpne_only.png")
makePie(hmc.hpne.only.anno, "hmc_hpne_only")
dev.off()
png("pie_hmc_jd_only.png")
makePie(hmc.jd.only.anno, "hmc_jd_only")
dev.off()

png("pie_mc_hpne_only.png")
makePie(mc.hpne.only.anno, "mc_hpne_only")
dev.off()
png("pie_mc_jd_only.png")
makePie(mc.jd.only.anno, "mc_jd_only")
dev.off()
```

```
#look at a random distribution of sites taken from min/max range of chromosome
sites.rand = sites.jd
max(sites.rand[,2])
sites.rand[,2] = runif(
      nrow(sites.rand),
      min(sites.rand[,2]),
      max(sites.rand[,2]) - min(sites.rand[,2])
)
```

```
#break this up...use just 10% of the sites
#to speed things up
#annoation the mcSites
#break up long list into multiple queries
#takes a long time
chunk <- function(x,n) split(x, factor(sort(rank(x)%%n)))
```

```
P = 0.1
ix.small = sort(sample(1:nrow(sites.hpne), nrow(sites.hpne)*P))
smallsites.hpne = sites.hpne[ix.small,]
ix = chunk(1:nrow(smallsites.hpne), 10)
anno.hpne = lapply(ix, function(i){
      print (i[1])
      annotateList(smallsites.hpne[i,])
})
x.hpne=do.call(rbind, anno.hpne)

ix.small = sort(sample(1:nrow(sites.jd), nrow(sites.jd)*P))
smallsites.jd = sites.jd[ix.small,]
ix = chunk(1:nrow(smallsites.jd), 10)
anno.jd = lapply(ix, function(i){
      print (i[1])
      annotateList(smallsites.jd[i,])
})
x.jd=do.call(rbind, anno.jd)

save.image("mc_anno_v1.RData")


png("pie_mc_Hpne_all.png")
makePie(x.hpne, "HPNE mc sites")
dev.off()
png("pie_mc_JD_all.png")
makePie(x.jd, "JD mc sites")
dev.off()

#how many genes are affected?
head(x.jd)
ugenes.jd = unique(x.jd$GENESYMBOL)
length(ugenes.jd)
ugenes.hpne = unique(x.hpne$GENESYMBOL)
length(ugenes.hpne)


#genes with hmc in one sample, but not the other
length(ugenes.hpne[!(ugenes.hpne %in% ugenes.jd)])
length(ugenes.jd[!(ugenes.jd %in% ugenes.hpne)])






anno.hpne = annotateList(sites.hpne)
anno.jd = annotateList(sites.jd)
anno.rand = annotateList(sites.rand)
anno.hpne.sig = annotateList(sites.hpne.sig)
anno.jd.sig = annotateList(sites.jd.sig)
dim(sites.hpne)
dim(sites.jd)
dim(sites.hpne.sig)
dim(sites.jd.sig)


#briefly look at hmc events at brd4
#save in bed format, view in igv
ix.1 = which(anno.hpne$GENESYMBOL == "BRD4")
ix.2 = which(anno.jd$GENESYMBOL == "BRD4")
length(ix.1)
length(ix.2)

anno.hpne[ix.1,]
x = anno.hpne[ix.1,2:4]
x = cbind(x, 1, 1)
write.table(file="hmc_brd4_hpne.bed", x=x, col.names=F, row.names=F, quote=F)
x = anno.jd[ix.2,2:4]
x = cbind(x, 1, 1)
write.table(file="hmc_brd4_jd.bed", x=x, col.names=F, row.names=F, quote=F)
#add a header to each bed file
# track name=test1 description="hmc" visibility=2


png("atacseq_hpne.png")
mytable <- table(panno.hpne[,5])
lbls <- paste(names(mytable), "\n", mytable, sep="")
pie(mytable, labels = lbls,
   main="ATACseq: HPNE")
dev.off()

png("atacseq_jd.png")
mytable <- table(panno.jd[,5])
lbls <- paste(names(mytable), "\n", mytable, sep="")
pie(mytable, labels = lbls,
   main="ATACseq: JD")
dev.off()


#par(mfcol=c(1, 2))
png("hmc_chr19_hpne.png")
mytable <- table(anno.hpne[,5])
lbls <- paste(names(mytable), "\n", mytable, sep="")
pie(mytable, labels = lbls,
   main="chr19 HMC: HPNE")
dev.off()

png("hmc_chr19_jd.png")
mytable <- table(anno.jd[,5])
lbls <- paste(names(mytable), "\n", mytable, sep="")
pie(mytable, labels = lbls,
   main="chr19 HMC: JD")
dev.off()

png("hmc_chr19_hpneSig.png")
mytable <- table(anno.hpne.sig[,5])
lbls <- paste(names(mytable), "\n", mytable, sep="")
pie(mytable, labels = lbls,
   main="chr19 HMC: HPNEsig")
dev.off()

png("hmc_chr19_jdSig.png")
mytable <- table(anno.jd.sig[,5])
lbls <- paste(names(mytable), "\n", mytable, sep="")
pie(mytable, labels = lbls,
   main="chr19 HMC: JDsig")
dev.off()
```

```
png("hmc_chr19_random.png")
mytable <- table(anno.rand[,5])
lbls <- paste(names(mytable), "\n", mytable, sep="")
pie(mytable, labels = lbls,
   main="chr19 HMC: random")
dev.off()

#doesn't look like there's much differenc in percentage
#on exonic/intron/promoter regions between hpne, jd
#random sites have slighter more intergenic regions


#might have to filter out some of the hmc sites based
#on direction of the gene
#wehther the reference base is C->T or G->A


#maybe look at the mutation impact factors?
#with snpEff?
#or with VEP

#Lets look at the hmc sites at brd4
#prepare teh vep file

sites = anno.hpne[ix.1,2:4]


anno.hpne[ix.1,]
x = anno.hpne[ix.1,2:4]



x = cbind(x, 1, 1)
write.table(file="hmc_brd4_hpne.bed", x=x, col.names=F, row.names=F, quote=F)
x = anno.jd[ix.2,2:4]


dat =
GRanges(seqnames=rep('chr17',3),ranges=IRanges(start=c(45229228,45229234,45234706),
width=1))
varallele = DNAStringSet(c('C','C','C'))
db = TxDb.Hsapiens.UCSC.hg19.knownGene

results = predictCoding(dat,db,Hsapiens,varallele)


sites = sites.hpne
input = sites[, 1:2]
colnames(input) <- c("chr", "pos")
input$pos       <- as.numeric(as.character(input$pos))
input = input[1:3000,]

target <- with(input, GRanges(
      seqnames = Rle(chr),
      ranges   = IRanges(
            pos,
            width=1,
            names=paste0("rand", 1:nrow(input))
      ),
      strand   = Rle(strand("*"))
))
length(target)
varallele = DNAStringSet(rep('C',length(target)))
results = predictCoding(target,db,Hsapiens,varallele)






#############################################
"chromosome_name", "start", "end"
x = list(chromosome_name="19", start="15236836", end="15332543")

x = head(sites.hpne[,c(1,2,2)], 300)
x = list(chromosome_name=gsub(x[,1], pat="chr(.*)", rep="\\1", perl=T),
start=x[,2], end=x[,3])
x = getBM(attributes=c("ensembl_gene_id", "hgnc_symbol", "chromosome_name",
"start_position", "end_position", "go_biological_process_linkage_type"), filters =
c("chromosome_name", "start", "end"), values = x, mart=grch37)


==> ./sanchari/j432_j446/findMcSites_stream.r <==
#!/usr/bin/env Rscript

options(stringsAsFactors=F)

#command line arguments
args = commandArgs(trailingOnly=TRUE)
f.bs = args[1]
f.oxbs = args[2]
f.hmc = args[3]
#f.bs = "conversionCounts/HPNE_BS_22.txt.gz"
#f.oxbs = "conversionCounts/HPNE_OxBS_22.txt.gz"
#f.hmc = "hmcSites_HPNE_22.txt"
#f.hmc = "test.txt"

print(f.bs)
print(f.oxbs)
print(f.hmc)

#BS
#c -> t
#5mc -> c
#5hmc -> c
#
#oxBS
#c -> t
#5mc -> c
#5hmc -> t

#we want to find the sites with hmc
#low conversion in BS
#high conversion in oxBS


#candidate sites should have
#conversion rate in Bs  below 50%
```

```
#conversion rate in oxBs above 50%



#instead of loading everything in at once
#read and process line by line
con1 = pipe(paste0("zcat ", f.bs))
open(con1, "r")
con2 = pipe(paste0("zcat ", f.oxbs))
open(con2, "r")

#start the file pointer at the begining
x1 = read.table(textConnection(readLines(con1, n=1)))
x2 = read.table(textConnection(readLines(con2, n=1)))

#keep processing line by line until there's an error
try({while(T){
      #we should only be working with a single chrom
      if (x1[1,1] != x2[1,1]){
            break
      }

      #matching site, process and print
      if (x1[1,2] == x2[1,2]){
            x = merge(x1, x2, by.x=1, by.y=1)
            v =as.numeric(x[1, c("V3.x", "V4.x", "V3.y", "V4.y")])

            #if both files have at least 4 reads
            #and both have a 0 conversion rate
            #if (v[1] == 0 && v[2] >= 4 && v[3] == 0 && v[4] >= 4){ #this is
incorrect
                #v1 is the # of unconverted reads of file1, v2 total file1
                #v3 is the # of unconverted reads of file2, v4 total file2
            if (v[1] == v[2] && v[2] >= 4 && v[3] == v[4] && v[4] >= 4){
                  cat(paste(c(as.character(x[1,1:2]), v)), file=f.hmc, append=T)
                  cat("\n", file=f.hmc, append=T)
            }
            #move both pointers up
            x1 = read.table(textConnection(readLines(con1, 1)))
            x2 = read.table(textConnection(readLines(con2, 1)))
      }else if(x1[1,2] < x2[1,2]){
            #move the bs file pointer up a spot
            x1 = read.table(textConnection(readLines(con1, 1)))
      }else if (x1[1,2] > x2[1,2]){
            #move the oxbs file pointer up a spot
            x2 = read.table(textConnection(readLines(con2, 1)))
      }
}}, silent=T)

close(con1)
close(con2)




==> ./sanchari/j432_j446/findHmcSites.py <==
#!/usr/bin/python

import gzip
import sys
from scipy.stats import fisher_exact

#ignore sites that have fewer than this many reads in both BS and OxBS
COMMON_THRESH = 4

#file are actually the unconverted counts, NOT converted counts
#   they show for each position
#   chr pos #meth #total
#  here, #meth are the number of unconverted reads!!

file1 = "conversionCounts/JD_BS_17.txt.gz"
file2 = "conversionCounts/JD_OxBs_17.txt.gz"
out = "hmcSites_py/hmcSites_HPNE_22.txt"
out = "hmcSites_py/test_22.txt"
out = "test1/test_17.txt"

file1 = sys.argv[1]
file2 = sys.argv[2]
out = sys.argv[3]

print (file1)
print (file2)
print (out)


fp1 = iter(gzip.open(file1, 'r'))
fp2 = iter(gzip.open(file2, "r"))
fp3 = open(out, "w")


x1 = next(fp1, "").rstrip("\n").split("\t")
x2 = next(fp2, "").rstrip("\n").split("\t")


while(True and len(x1) != 1 and len(x2) != 1):
    if (x1[0] != x2[0]):
        break
    #matching site, process and print
    if (int(x1[1]) == int(x2[1])):
        a = int(x1[2])
        b = int(x1[3]) - a
        c = int(x2[2])
        d = int(x2[3]) - c
        if (a+b >= COMMON_THRESH and c+b >= COMMON_THRESH):
            #r uses conditional MLE method to calc odds ratio
            #so it won't be exactly the same
            #odds, pval = fisher_exact([[a, b], [c, d]])
            #odds, pval = fisher_exact([[a, c], [b, d]], alternative="less")  #this
one is incorrect!!
            #we're looking for high methylation in BS
            #and low methylation in OxBS
            odds, pval = fisher_exact([[a, c], [b, d]], alternative="greater")
            #process fisher test pvalue
            #
            #print result
            fp3.write("\t".join(map(str, (x1+x2[2:4] + [odds, pval])))+"\n")
            #if (odds < 1 and pval < 0.05):
            #    print odds, pval
            #    print x1
            #    print x2
            #    sigCount = sigCount + 1
            #
        #move both pointers up
        x1 = next(fp1, "").rstrip("\n").split("\t")
        x2 = next(fp2, "").rstrip("\n").split("\t")
    elif(int(x1[1]) < int(x2[1])):
        #move the bs file pointer up a spot
        x1 = next(fp1, "").rstrip("\n").split("\t")
    elif (int(x1[1]) > int(x2[1])):
        #move the oxbs file pointer up a spot
        x2 = next(fp2, "").rstrip("\n").split("\t")

fp1.close()
fp2.close()
fp3.close()

==> ./sanchari/j432_j446/mergeCounts.r <==
list.files(pattern="*.bedGraph")

list.files(pattern="*HPNE-1[-_]BS*")

chrom = "chr10"

x1 = read.table(pipe("grep J432_AHHH5NBCXX_Lane1_ACAGTG_HPNE-1_BS_merged_sorted_mdup_clip.bedGraph -e \"^chr17\\s\" | cut -f1,2,5,6"))
```