

R script:

```
##### Reads single probe map
probe<-read.csv('MoGene-1_0-st-
v1.mm9.probe.csv',header=FALSE,colClasses=c('character','integer','integer','character','integer','character','integer'))

# 'MoGene-1_0-st-v1.mm9.probe.csv': no header; Col1: chromosome name (chr10); Col2: probe 5' site on the genome (3515979);
Col3: probe 3' site on the genome (3516002); Col4: probe ID (10367546_23); Col5: "0"; Col6: direction on the genome (-); Col7:
probeset ID (10367546)

##### Reads transcript map
structure<-
read.csv('features_mm9.csv',header=FALSE,colClasses=c('character','integer','integer','character','integer','character','character','inte
ger'))

# 'features_mm9.csv': no header; Col1: chromosome name (chr1); Col2: exon 5' site on the genome (134227135); Col3: exon 3' site
on the genome (134227268); Col4: transcript Refseq (NM_028778); Col5: "0"; Col6: direction on the genome (+); Col7: "exon"; Col8:
exon number in the transcript (6)

structure<-rbind(structure,structure[nrow(structure),])
structure[nrow(structure),4]<-"FAKE"

##### Reads Aire WT/KO MoGeneST1.0 dataset
dat<-
read.csv('ST1_F1Aire.csv',colClasses=c('numeric','numeric','numeric','numeric','numeric','numeric','numeric','numeric','numeric','num
eric','numeric','numeric','numeric','numeric','numeric','character'))

# 'ST1_F1Aire.csv': with header; Col1 [probeset]: probeset ID (10350977); Col2 [indice]: probe indice in the probeset from 0 to n
(11); Col3 [strand]: "1" for +, "2" for -, "0" for unknown; Col4 [KO_116] probe expression value in Aire KO REP1 (579.05); Col5
[KO_118] probe expression value in Aire KO REP2 (425.11); Col6 [KO_124] probe expression value in Aire KO REP3 (551.82); Col7
[WT_126] probe expression value in Aire WT REP1 (465.77); Col8 [WT_128] probe expression value in Aire WT REP2 (627.11);
Col9 [WT_132] probe expression value in Aire WT REP3 (601.38); Col10 [KO] average KO probe expression value (518.66); Col11
[WT] average WT probe expression value (564.75); Col12 [FC_probe] probe fold change (1.089); Col13 [] empty; Col14 [FC_set]
probeset fold change (1.028); Col15 [rank] probeset FC rank (12507); Col16 [] empty; Col17 [probenam] probe ID (10350977_12)

##### Mis-hybridization filtering
dat<-dat[dat$KO_116>=20|dat$KO_118>=20|dat$KO_124>=20|dat$WT_126>=20|dat$WT_128>=20|dat$WT_132>=20,]

##### Core algorithm
testWT<-vector(mode="numeric",length=nrow(structure))
testWTnorm<-vector(mode="numeric",length=nrow(structure))
testKO<-vector(mode="numeric",length=nrow(structure))
testKOnorm<-vector(mode="numeric",length=nrow(structure))
testFCnorm<-vector(mode="numeric",length=nrow(structure))
testP<-vector(mode="numeric",length=nrow(structure))
FCmed<-vector(mode="numeric",length=nrow(structure))
Refseq<-vector(mode="character",length=nrow(structure))
ex<-vector(mode="character",length=nrow(structure))
exnb<-vector(mode="integer",length=nrow(structure))
dir<-vector(mode="character",length=nrow(structure))

ini<-1
WTglob<-1
KOGlob<-1
store<-NA
test<-0
adj<-1
```

```

nbr<-nrow(structure)-1
#nbr<-191

```

```

for (i in ini:(nbr+1)) {

```

```

  chr<-structure[i,1]
  start<-structure[i,2]
  end<-structure[i,3]

```

```

  temp<-probe[(probe[,1]==chr & probe[,2]>=start & probe[,3]<=end),c(4,7)]

```

```

  if (nrow(temp)>0){
    findWT<-as.matrix(log2(dat[match(temp[,1],dat$probename),c('WT_126','WT_128','WT_132')]))
    findKO<-as.matrix(log2(dat[match(temp[,1],dat$probename),c('KO_116','KO_118','KO_124')]))
    general<-

```

```

cbind(structure[i,c(4,7,8,6)],feat_start=start,feat_end=end,probename=temp[,1],probe[match(temp[,1],probe[,4]),c(2,3)],2^findWT,2^fi
ndKO,row.names=NULL)

```

```

  if (test==0){
    store<-general
    test<-1
  }else{store<-rbind(store,general)}
}

```

```

if (nrow(temp)>=2|i==(nbr+1)) {

```

```

  print(i)
  if (length(unique(na.omit((match(temp[,1],dat$probename))))>=2)
    {

```

```

    WT<-cbind(findWT,i)
    KO<-cbind(findKO,i)

```

```

    if (i==ini|(is.na(WTglob[1])&WTglob[1]==1)){
      WTglob<-WT
      KOglob<-KO
    } else {

```

```

      if ((structure[i,4]==(structure[i-adj,4]))){
        WTglob<-rbind(WTglob,WT)
        KOglob<-rbind(KOglob,KO)

```

```

      } else {

```

```

        WTmedian<-median(na.omit(WTglob[,1:3]))
        KOmedian<-median(na.omit(KOglob[,1:3]))

```

```

        WTcn<-cbind(WTglob[,1:3]-WTmedian,WTglob[,4])
        KOcn<-cbind(KOglob[,1:3]-KOmedian,KOglob[,4])

```

```

        for (j in WTglob[1,4]:WTglob[nrow(WTglob),4]){

```

```

          Refseq[j]<-structure[j,4]
          ex[j]<-structure[j,7]
          exnb[j]<-structure[j,8]
          dir[j]<-structure[j,6]

```

```

          FCmed[j]<-2^(WTmedian-KOmedian)

```

```

          if (nrow(na.omit(WTcn[WTcn[,4]==j,1:3]))>=2){
            if

```

```

(!identical(WTcn[WTcn[,4]==j,1:3][1,],WTcn[WTcn[,4]==j,1:3][2,])&!identical(KOcn[KOcn[,4]==j,1:3][1,],KOcn[KOcn[,4]==j,1:3][2,]))==T
RUE){

```

```

          testWT[j]<-median(WTglob[WTglob[,4]==j,1:3],na.rm=TRUE)
          testKO[j]<-median(KOglob[KOglob[,4]==j,1:3],na.rm=TRUE)
          testWTnorm[j]<-median(WTcn[WTcn[,4]==j,1:3],na.rm=TRUE)
          testKOnorm[j]<-median(KOcn[KOcn[,4]==j,1:3],na.rm=TRUE)
          testFCnorm[j]<-testWTnorm[j]-testKOnorm[j]

```

```

ttest<-t.test(as.numeric(na.omit(WTcn[WTcn[,4]==j,1:3])),as.numeric(na.omit(KOcn[KOcn[,4]==j,1:3])),var.equal=TRUE)

```

