

HMBscript

Model Fitting

This code will run the 7 models reported in the paper and generate all the relevant parameters.

```
library(hBayesDM)
library(knitr)
library("vioplot")
attach(mtcars)

setwd("C:/Users/orobinson/OneDrive - University College
London/OneDrive_Oli/BanditStudy/BanditR")
output1_ANXT = bandit4arm_4par("BanditDataforRThreatANXAnon.txt", nchain=3,
ncore=3)
output1_ANXS = bandit4arm_4par("BanditDataforRSafeANXAnon.txt", nchain=3,
ncore=3)
output1_HCT = bandit4arm_4par("BanditDataforRThreatHCAnon.txt", nchain=3,
ncore=3)
output1_HCS = bandit4arm_4par("BanditDataforRSafeHCAnon.txt", nchain=3,
ncore=3)
output2_ANXT = bandit4arm_lapse("BanditDataforRThreatANXAnon.txt",
nchain=3, ncore=3)
output2_ANXS = bandit4arm_lapse("BanditDataforRSafeANXAnon.txt", nchain=3,
ncore=3)
output2_HCT = bandit4arm_lapse("BanditDataforRThreatHCAnon.txt", nchain=3,
ncore=3)
output2_HCS = bandit4arm_lapse("BanditDataforRSafeHCAnon.txt", nchain=3,
ncore=3)
output3_ANXT = igt_pvl_decay("BanditDataforRThreatANXAnon.txt", nchain=3,
ncore=3)
output3_ANXS = igt_pvl_decay("BanditDataforRSafeANXAnon.txt", nchain=3,
ncore=3)
output3_HCT = igt_pvl_decay("BanditDataforRThreatHCAnon.txt", nchain=3,
ncore=3)
output3_HCS = igt_pvl_decay("BanditDataforRSafeHCAnon.txt", nchain=3,
ncore=3)
output4_ANXT = igt_pvl_delta("BanditDataforRThreatANXAnon.txt", nchain=3,
ncore=3)
output4_ANXS = igt_pvl_delta("BanditDataforRSafeANXAnon.txt", nchain=3,
ncore=3)
output4_HCT = igt_pvl_delta("BanditDataforRThreatHCAnon.txt", nchain=3,
ncore=3)
output4_HCS = igt_pvl_delta("BanditDataforRSafeHCAnon.txt", nchain=3,
ncore=3)
output5_ANXT = bandit4arm_2par_lapse("BanditDataforRThreatANXAnon.txt",
nchain=3, ncore=3)
output5_ANXS = bandit4arm_2par_lapse("BanditDataforRSafeANXAnon.txt",
nchain=3, ncore=3)
output5_HCT = bandit4arm_2par_lapse("BanditDataforRThreatHCAnon.txt",
nchain=3, ncore=3)
output5_HCS = bandit4arm_2par_lapse("BanditDataforRSafeHCAnon.txt",
nchain=3, ncore=3)
output6_ANXT = bandit4arm_singleA_lapse("BanditDataforRThreatANXAnon.txt",
nchain=3, ncore=3)
output6_ANXS = bandit4arm_singleA_lapse("BanditDataforRSafeANXAnon.txt",
nchain=3, ncore=3)
```

```

output6_HCT = bandit4arm_singleA_lapse("BanditDataforRThreatHCAnon.txt",
nchain=3, ncore=3)
output6_HCS = bandit4arm_singleA_lapse("BanditDataforRSafeHCAnon.txt",
nchain=3, ncore=3)
output7_ANXT = bandit4arm_lapse_decay("BanditDataforRThreatANXAnon.txt",
nchain=3, ncore=3)
output7_ANXS = bandit4arm_lapse_decay("BanditDataforRSafeANXAnon.txt",
nchain=3, ncore=3)
output7_HCT = bandit4arm_lapse_decay("BanditDataforRThreatHCAnon.txt",
nchain=3, ncore=3)
output7_HCS = bandit4arm_lapse_decay("BanditDataforRSafeHCAnon.txt",
nchain=3, ncore=3)

```

Model Comparison

This code will generate a model comparison table (N.b. exact values will likely differ due to nature of fitting).

```

fits = printFit(output1_ANXT,output1_ANXS,
output1_HCT,output1_HCS,output2_ANXT,output2_ANXS,
output2_HCT,output2_HCS,output3_ANXT,output3_ANXS, output3_HCT,output3_HCS,
output4_ANXT,output4_ANXS,
output4_HCT,output4_HCS,output5_ANXT,output5_ANXS,
output5_HCT,output5_HCS,output6_ANXT,output6_ANXS,
output6_HCT,output6_HCS,output7_ANXT,output7_ANXS, output7_HCT,output7_HCS)
bandit4arm_4parLOOIC <- sum(fits[1:4,2])
bandit4arm_lapseLOOIC <- sum(fits[5:8,2])
igt_pvl_decayLOOIC <- sum(fits[9:12,2])
igt_pvl_deltaLOOIC <- sum(fits[13:16,2])
bandit4arm_2par_lapseLOOIC <- sum(fits[17:20,2])
bandit4arm_singleA_lapseLOOIC <- sum(fits[21:24,2])
bandit4arm_lapse_decayLOOIC <- sum(fits[25:28,2])

LOOICtable<-
matrix(c(bandit4arm_4parLOOIC,bandit4arm_lapseLOOIC,igt_pvl_decayLOOIC,igt_
pvl_deltaLOOIC,bandit4arm_2par_lapseLOOIC,bandit4arm_singleA_lapseLOOIC,ban
dit4arm_lapse_decayLOOIC),ncol=1,byrow=TRUE)
rownames(LOOICtable)<-
c("bandit4arm_4par","bandit4arm_lapse","igt_pvl_decay","igt_pvl_delta","ban
dit4arm_2par_lapseLOOIC","bandit4arm_singleA_lapseLOOIC","bandit4arm_lapse_
decayLOOIC")
colnames(LOOICtable)<-c("LOOIC")
LOOICtable <- as.table(LOOICtable)

kable(LOOICtable) # %>%

```

Inference on bandit4arm_lapse model

The below will extract and compare parameters for one of the reported models (tweak if you want to do the same for other models)

```

#reward Sensitivity
diffDistRdiagT = output2_ANXT$parVals$mu_R-
output2_HCT$parVals$mu_R
diffDistRdiagS = output2_ANXS$parVals$mu_R -
output2_HCS$parVals$mu_R

```

```

diffDistRcondA = output2_ANXT$parVals$mu_R -
  output2_ANXS$parVals$mu_R
diffDistRcondH = output2_HCT$parVals$mu_R -
  output2_HCS$parVals$mu_R
# Rlot 95% Highest Density Interval (HDI)
RdiagT = HDIofMCMC(diffDistRdiagT)
RdiagS = HDIofMCMC(diffDistRdiagS)
RcondA = HDIofMCMC(diffDistRcondA)
RcondH = HDIofMCMC(diffDistRcondH)

#punishment Sensitivity
diffDistPdiagT = output2_ANXT$parVals$mu_P -
  output2_HCT$parVals$mu_P
diffDistPdiagS = output2_ANXS$parVals$mu_P -
  output2_HCS$parVals$mu_P
diffDistPcondA = output2_ANXT$parVals$mu_P -
  output2_ANXS$parVals$mu_P
diffDistPcondH = output2_HCT$parVals$mu_P -
  output2_HCS$parVals$mu_P
PdiagT = HDIofMCMC(diffDistPdiagT)
PdiagS = HDIofMCMC(diffDistPdiagS)
PcondA = HDIofMCMC(diffDistPcondA)
PcondH = HDIofMCMC(diffDistPcondH)

#reward LR
diffDistRAdiagT = output2_ANXT$parVals$mu_Arew -
  output2_HCT$parVals$mu_Arew
diffDistRAdiagS = output2_ANXS$parVals$mu_Arew -
  output2_HCS$parVals$mu_Arew
diffDistRAcondA = output2_ANXT$parVals$mu_Arew -
  output2_ANXS$parVals$mu_Arew
diffDistRAcondH = output2_HCT$parVals$mu_Arew -
  output2_HCS$parVals$mu_Arew
# Rlot 95% Highest Density Interval (HDI)
RAdiagT = HDIofMCMC(diffDistRAdiagT)
RAdiagS = HDIofMCMC(diffDistRAdiagS)
RAcondA = HDIofMCMC(diffDistRAcondA)
RAcondH = HDIofMCMC(diffDistRAcondH)

#punishment LR
diffDistPAdiagT = output2_ANXT$parVals$mu_Apun -
  output2_HCT$parVals$mu_Apun
diffDistPAdiagS = output2_ANXS$parVals$mu_Apun -
  output2_HCS$parVals$mu_Apun
diffDistPAcondA = output2_ANXT$parVals$mu_Apun -
  output2_ANXS$parVals$mu_Apun
diffDistPAcondH = output2_HCT$parVals$mu_Apun -
  output2_HCS$parVals$mu_Apun
PAdiagT = HDIofMCMC(diffDistPAdiagT)
PAdiagS = HDIofMCMC(diffDistPAdiagS)
PAcondA = HDIofMCMC(diffDistPAcondA)
PAcondH = HDIofMCMC(diffDistPAcondH)

#Lapse
diffDistLapseddiagT = output2_ANXT$parVals$mu_xi -
  output2_HCT$parVals$mu_xi
diffDistLapseddiagS = output2_ANXS$parVals$mu_xi -
  output2_HCS$parVals$mu_xi
diffDistLapsecondA = output2_ANXT$parVals$mu_xi -
  output2_ANXS$parVals$mu_xi
diffDistLapsecondH = output2_HCT$parVals$mu_xi -

```

```

output2_HCS$parVals$mu_xi
LapseddiagT = HDIofMCMC(diffDistLapseddiagT)
LapseddiagS = HDIofMCMC(diffDistLapseddiagS)
LapsecondA = HDIofMCMC(diffDistLapsecondA)
LapsecondH = HDIofMCMC(diffDistLapsecondH)

HDItable<-matrix(c(RdiagT,RdiagS,RcondA,RcondH,
PdiagT,PdiagS,PcondA,PcondH, RAdiagT,RAdiagS,RAcondA,RAcondH,
PAdiagT,PAdiagS,PAcondA,PAcondH,
LapseddiagT,LapseddiagS,LapsecondA,LapsecondH),ncol=8,byrow=TRUE)
rownames(HDItable)<-c("Reward Sensitivity","Punishment Sensitivity","Reward
Learning Rate","Punishment Learning Rate", "Lapse")
colnames(HDItable)<-c("Anx-HC Thr LB", "Anx-HC Thr UB", "Anx-HC Saf LB",
"Anx-HC Saf UB", "Thr-Saf Anx LB", "Thr-Saf Anx UB","Thr-Saf HC LB", "Thr-
Saf HC UB")
HDItable <- as.table(HDItable)
kable(HDItable)

PTANX = output2_ANXT$parVals$mu_Apun
PSANX = output2_ANXS$parVals$mu_Apun
PTHC = output2_HCT$parVals$mu_Apun
PSHC = output2_HCS$parVals$mu_Apun

XTANX = output2_ANXT$parVals$mu_xi
XSANX = output2_ANXS$parVals$mu_xi
XTHC = output2_HCT$parVals$mu_xi
XSHC = output2_HCS$parVals$mu_xi

par(mfrow=c(1,2))
plot(1, 2, xlim = c(0, 8),ylim = range(c(PTANX, PSANX, PTHC, PSHC)), type =
'n', xlab = '', ylab = 'Punishment Learning Rate', xaxt = 'n')
vioplot(PSANX, at = 1, add = T, col = 'palegreen2')
vioplot(PTANX, at = 3, add = T, col = 'palegreen4')
vioplot(PSHC, at =5, add = T, col = 'paleturquoise2')
vioplot(PTHC, at = 7, add = T, col = 'paleturquoise4')
legend("topright", inset=.02, legend=c('ANX Safe', 'ANX Threat','HC Safe',
'HC Threat'), fill = c("palegreen2", "palegreen4","paleturquoise2",
"paleturquoise4"), cex=0.8)

plot(2, 2, xlim = c(0, 8),ylim = range(c(XTANX, XSANX, XTHC, XSHC)), type =
'n', xlab = '', ylab = 'Lapse', xaxt = 'n')
vioplot(XSANX, at = 1, add = T, col = 'palegreen2')
vioplot(XTANX, at = 3, add = T, col = 'palegreen4')
vioplot(XSHC, at =5, add = T, col = 'paleturquoise2')
vioplot(XTHC, at = 7, add = T, col = 'paleturquoise4')
legend("topright", inset=.02, legend=c('ANX Safe', 'ANX Threat','HC Safe',
'HC Threat'), fill = c("palegreen2", "palegreen4","paleturquoise2",
"paleturquoise4"), cex=0.8)

####the following section compares the different priors on the winning
model. The text files can be made by combining the provided anon data.

output2_ANX = bandit4arm_lapse("BanditDataforRANX.txt", nchain=3, ncore=3)
output2_HC = bandit4arm_lapse("BanditDataforRHC.txt", nchain=3, ncore=3)
output2_S = bandit4arm_lapse("BanditDataforRSafe.txt", nchain=3, ncore=3)
output2_T = bandit4arm_lapse("BanditDataforRThreat.txt", nchain=3, ncore=3)
output2_All = bandit4arm_lapse("BanditDataforRAll.txt", nchain=3, ncore=3)

fits = printFit(output2_ANX, output2_HC, output2_S, output2_T, output2_All)

```

```

bandit4arm_lapseDiagPriorLOOIC <- sum(fits[1:2,2])
bandit4arm_lapseCondPriorLOOIC <- sum(fits[3:4,2])
bandit4arm_lapseSinglePriorLOOIC <- fits[5,2]

LOOICtable<-
matrix(c(bandit4arm_lapseDiagPriorLOOIC,bandit4arm_lapseCondPriorLOOIC,band
it4arm_lapseSinglePriorLOOIC),ncol=1,byrow=TRUE)
rownames(LOOICtable)<-c("Diagnosis_Prior","Condition_Prior","Single_Prior")
colnames(LOOICtable)<-c("LOOIC")
LOOICtable <- as.table(LOOICtable)

kable(LOOICtable)
write.csv(LOOICtable, file = "PriorComp.csv")

#reward Sensitivity
diffDistRdiag = output2_ANX$parVals$mu_R -
  output2_HC$parVals$mu_R
# Rlot 95% Highest Density Interval (HDI)
Rdiag = HDIofMCMC(diffDistRdiag)

#punishment Sensitivity
diffDistPdiag = output2_ANX$parVals$mu_P -
  output2_HC$parVals$mu_P
Pdiag = HDIofMCMC(diffDistPdiag)

#reward LR
diffDistRAdiag = output2_ANX$parVals$mu_Arew -
  output2_HC$parVals$mu_Arew
# Rlot 95% Highest Density Interval (HDI)
RAdiag = HDIofMCMC(diffDistRAdiag)

#punishment LR
diffDistPAdiag = output2_ANX$parVals$mu_Apun -
  output2_HC$parVals$mu_Apun
PAdiag = HDIofMCMC(diffDistPAdiag)

#Lapse
diffDistLapsediag = output2_ANX$parVals$mu_xi -
  output2_HC$parVals$mu_xi
Lapsediag = HDIofMCMC(diffDistLapsediag)

HDItable<-matrix(c(Rdiag, Pdiag, RAdiag, PAdiag,
Lapsediag),ncol=2,byrow=TRUE)
rownames(HDItable)<-c("Reward Sensitivity","Punishment Sensitivity","Reward
Learning Rate","Punishment Learning Rate", "Lapse")
colnames(HDItable)<-c("Anx-HC LB", "Anx-HC UB")
HDItable <- as.table(HDItable)
kable(HDItable)
write.csv(HDItable, file = "ParamsDiagPrior.csv")

rawApunANX = output2_ANX$allIndPars$Apun
rawApunHC = output2_HC$allIndPars$Apun
rawArewANX = output2_ANX$allIndPars$Arew
rawArewHC = output2_HC$allIndPars$Arew
rawXiANX = output2_ANX$allIndPars$xi
rawXiHC = output2_HC$allIndPars$xi
rawPANX = output2_ANX$allIndPars$P
rawPHC = output2_HC$allIndPars$P
rawRANX = output2_ANX$allIndPars$R
rawRHC = output2_HC$allIndPars$R

```

```
Alldata <- data.frame(group=c(rep("HC", times=length(rawPHC)),
                             rep("ANX", times=length(rawPANX))),
                    SubID=c(output2_HC$allIndPars$subjID,
                             output2_ANX$allIndPars$subjID),
                    Rew_LR=c(rawArewHC, rawArewANX),
                    Punish_LR=c(rawApunHC, rawApunANX),
                    Rew_Sens=c(rawRHC, rawRANX),
                    Punish_Sens=c(rawPHC, rawPANX),
                    Noise=c(rawXiHC, rawXiANX))
```