```
# ------------------------------------------------------------------------------
# Program: Age moderation of depressive symptoms for IGEMS consortium
# Univariate age moderation with two slopes (age 40-75 and 75-90)
# Depression Variable: CAMDEX and Harmonized CAMDEX
#  Author: Drew Petkus
# Original Date: 9 16 2015
# Final Modification data 9 18 2016 (add confidence interval esimates)

# -------|---------|---------|---------|---------|---------|---------|---------|

####clear workspace

remove(list=ls())

###load required programs

require(OpenMx)
require(psych)
source("http://www.vipbg.vcu.edu/~vipbg/Tc24/GenEpiHelperFunctions.R")
source("GenEpiHelperFunctions.R")
#=============================================================================#
#   PREPARE DATA  AND CREATE SLOPE VARIABLES
#
#=============================================================================#

###set working memory

setwd("/Users/Drew/Documents/IGEMS dep x age moderation analyses/depXage")


# LOAD AND PREPARE DATA

# Read Twin Data
data<- read.csv(file="depcirs_23sep15.csv", header=TRUE)
describe(data)

#TRANSFORM DATA to be centered and divided by 10
#AGE
data$ageC1<-data$age1-75
data$old<-ifelse(data$ageC1>=0,c(1),c(0))
data$slope2<-ifelse(data$ageC1>=0,c(data$ageC1),c(0))

data$slope1<-data$ageC1
data$slope1<-ifelse(data$ageC1>=0,c(0),(data$ageC1))

table(data$slope1)
table(data$old)

#=============================================================================#
#   SELECT DATA FOR TWIN MODELS
#
#=============================================================================#

selVars <- c('tlndepC1','tlndepC2') # phenotype variables to be analyzed
```

```
modVars <- c('slope1','old','slope2') # moderator variable
nv <- 1 # number of phenotype variables per twin
ntv <- nv*2 # total number of phenotype variables
na <- 1 # number of A factors (per twin)
nc <- 1 # number of C factors (per twin)
ne <- 1 # number of E factors (per twin)
zygos <-c('zygos1','zygos2')

#select data for male and female MZ and DZ

data<-subset(data,age1<=90 &age1>=40)  ##SELECT ONLY THOSE BETWEEN 40 AND 90
mzMData <- subset(data, sex1==1 & sex2==1 & zygos1==1, c(selVars,modVars))
###SELECT MZ MALE DATA
dzMData <- subset(data, sex1==1 & sex2==1 & zygos1==2, c(selVars,modVars)) ####
SELECT DZ MALE DATA
mzFData <- subset(data, sex1==2 & sex2==2 & zygos1==1, c(selVars,modVars))
####SELECT MZ FEMALE DATA
dzFData <- subset(data, sex1==2 & sex2==2 & zygos1==2, c(selVars,modVars)) #####
SELECT DZ FEMALE DATA
dzOData <- subset(data, zygos1==3, c(selVars,modVars)) ###SELECT OPPOSITE SEX
TWIN PAIR DATA


# Select cases with no missings on the definition variable
mzMData <- mzMData[!is.na(mzMData$slope1),]  ###REMOVE PARTICIPANTS WITH MISSING
ON DEFINITION VARIABLES
dzMData <- dzMData[!is.na(dzMData$slope1),]
mzFData <- mzFData[!is.na(mzFData$slope1),]
dzFData <- dzFData[!is.na(dzFData$slope1),]
dzOData <- dzOData[!is.na(dzOData$slope1),]


# Generate descriptive statistics for each group
colMeans(mzMData[,selVars],na.rm=TRUE)
colMeans(dzMData[,selVars],na.rm=TRUE)
colMeans(mzFData[,selVars],na.rm=TRUE)
colMeans(dzFData[,selVars],na.rm=TRUE)
colMeans(dzOData[,selVars],na.rm=TRUE)

#Generate covariance matrix for each group
cov(mzMData[,selVars],use="complete")
cov(dzMData[,selVars],use="complete")
cov(mzFData[,selVars],use="complete")
cov(dzFData[,selVars],use="complete")
cov(dzOData[,selVars],use="complete")

# Generate correlations for each group
cor(dzMData[,selVars],use="complete")
cor(mzMData[,selVars],use="complete")
cor(dzFData[,selVars],use="complete")
cor(mzFData[,selVars],use="complete")
cor(dzOData[,selVars],use="complete")
```

```
# SET UP ACE Model


# Matrices declared to store a, c, and e Path Coefficients for males and females
pathAm     <- mxMatrix( type="Full", nrow=nv, ncol=nv, free=TRUE, values=4,
label="am11", name="aM" )
pathCm     <- mxMatrix( type="Full", nrow=nv, ncol=nv, free=TRUE, values=1,
label="cm11", name="cM" )
pathEm     <- mxMatrix( type="Full", nrow=nv, ncol=nv, free=TRUE, values=7,
label="em11", name="eM" )
pathAf     <- mxMatrix( type="Full", nrow=nv, ncol=nv, free=TRUE, values=5.5,
label="af11", name="aF" )
pathCf     <- mxMatrix( type="Full", nrow=nv, ncol=nv, free=TRUE, values=1,
label="cf11", name="cF" )
pathEf     <- mxMatrix( type="Full", nrow=nv, ncol=nv, free=TRUE, values=8.3,
label="ef11", name="eF" )


# Algebra for expected Mean Matrices in MZ & DZ twins and opposite sex twin
pairs
meanm    <- mxMatrix( "Full", nrow=1, ncol=1, free=TRUE,
                      values= -0.5, label="meanm", name="expMeanm" )
meanf    <- mxMatrix( "Full", nrow=1, ncol=1, free=TRUE,
                      values= 1, label="meanf", name="expMeanf" )


# Matrix for moderating variables
defAge    <- mxMatrix( type="Full", nrow=1, ncol=1, free=FALSE,
labels="data.slope1", name="age")
defold <-mxMatrix(type="Full", nrow=1, ncol=1, free=F, labels="data.old",
name="old")
oldage<-mxMatrix(type="Full", nrow=1, ncol=1, free=F, labels="data.slope2",
name="age2" )

# Matrices declared to store moderated a, c, and e Path Coefficients
pathASm    <- mxMatrix( type="Full", nrow=nv, ncol=nv, free=TRUE, values=-0.1,
label="aS1m", name="aL1m" )
pathCSm    <- mxMatrix( type="Full", nrow=nv, ncol=nv, free=TRUE, values=-
0.43,label="cS1m", name="cL1m" )
pathESm    <- mxMatrix( type="Full", nrow=nv, ncol=nv, free=TRUE, values=-
0.02,label="eS1m", name="eL1m" )

pathASf    <- mxMatrix( type="Full", nrow=nv, ncol=nv, free=TRUE,
values=0.142,label="aS1f", name="aL1f" )
pathCSf    <- mxMatrix( type="Full", nrow=nv, ncol=nv, free=TRUE, values=0.08,
label="cS1f", name="cL1f" )
pathESf    <- mxMatrix( type="Full", nrow=nv, ncol=nv, free=TRUE, values=0.04,
label="eS1f", name="eL1f" )


###second slope spline coefficient
# Matrices declared to store moderated a, c, and e Path Coefficients
```

```
pathAS2m    <- mxMatrix( type="Full", nrow=nv, ncol=nv, free=TRUE, values=0.35,
label="aS2m", name="aL2m" )
pathCS2m    <- mxMatrix( type="Full", nrow=nv, ncol=nv, free=TRUE, values=1.49,
label="cS2m", name="cL2m" )
pathES2m    <- mxMatrix( type="Full", nrow=nv, ncol=nv, free=TRUE, values=-0.05,
label="eS2m", name="eL2m" )

pathAS2f    <- mxMatrix( type="Full", nrow=nv, ncol=nv, free=TRUE, values=-1,
label="aS2f", name="aL2f" )
pathCS2f    <- mxMatrix( type="Full", nrow=nv, ncol=nv, free=TRUE, values=1.9,
label="cS2f", name="cL2f" )
pathES2f    <- mxMatrix( type="Full", nrow=nv, ncol=nv, free=TRUE, values=0.4,
label="eS2f", name="eL2f" )



# Matrices declared to store linear and quadratic Regression Coefficients for
mean moderation
pathBm     <- mxMatrix( type="Full", nrow=1, ncol=2, free=TRUE, values=
c(0.03,0.002), label=c("lsm","ls2m"), name="bm" )
pathBf     <- mxMatrix( type="Full", nrow=1, ncol=2, free=TRUE, values=
c(0.05,0.002), label=c("lsf","ls2f"), name="bf" )


# Matrices generated to hold A, C, and E computed Variance Components


covAMmod1<- mxAlgebra(name = "AM1", expression = (aM+ age%x%aL1m+
old%x%age2%x%aL2m) %*% t(aM+ age%x%aL1m+ old%x%age2%x%aL2m))
covCMmod1<- mxAlgebra(name = "CM1", expression = (cM+ age%x%cL1m+
old%x%age2%x%cL2m) %*% t(cM+ age%x%cL1m+ old%x%age2%x%cL2m))
covEMmod1<- mxAlgebra(name = "EM1", expression = (eM+ age%x%eL1m+
old%x%age2%x%eL2m) %*% t(eM+ age%x%eL1m+ old%x%age2%x%eL2m))


covAMmod2<- mxAlgebra(name = "AM2", expression = (aM+ age%x%aL1m+
old%x%age2%x%aL2m) %*% t(aM+ age%x%aL1m+ old%x%age2%x%aL2m))
covCMmod2<- mxAlgebra(name = "CM2", expression = (cM+ age%x%cL1m+
old%x%age2%x%cL2m) %*% t(cM+ age%x%cL1m+ old%x%age2%x%cL2m))
covEMmod2<- mxAlgebra(name = "EM2", expression = (eM+ age%x%eL1m+
old%x%age2%x%eL2m) %*% t(eM+ age%x%eL1m+ old%x%age2%x%eL2m))


covAFmod1<- mxAlgebra(name = "AF1", expression = (aF+ age%x%aL1f+
old%x%age2%x%aL2f) %*% t(aF+ age%x%aL1f+ old%x%age2%x%aL2f))
covCFmod1<- mxAlgebra(name = "CF1", expression = (cF+ age%x%cL1f+
old%x%age2%x%cL2f) %*% t(cF+ age%x%cL1f+ old%x%age2%x%cL2f))
covEFmod1<- mxAlgebra(name = "EF1", expression = (eF+ age%x%eL1f+
old%x%age2%x%eL2f) %*% t(eF+ age%x%eL1f+ old%x%age2%x%eL2f))


covAFmod2<- mxAlgebra(name = "AF2", expression = (aF+ age%x%aL1f+
old%x%age2%x%aL2f) %*% t(aF+ age%x%aL1f+ old%x%age2%x%aL2f))
```

```
covCFmod2<- mxAlgebra(name = "CF2", expression = (cF+ age%x%cL1f+
old%x%age2%x%cL2f) %*% t(cF+ age%x%cL1f+ old%x%age2%x%cL2f))
covEFmod2<- mxAlgebra(name = "EF2", expression = (eF+ age%x%eL1f+
old%x%age2%x%eL2f) %*% t(eF+ age%x%eL1f+ old%x%age2%x%eL2f))


covAM12<-mxAlgebra(name= "AM12", expression = (aM+ age%x%aL1m+
old%x%age2%x%aL2m) %*% t(aM+ age%x%aL1m+ old%x%age2%x%aL2m))
covCM12<-mxAlgebra(name= "CM12", expression = (cM+ age%x%cL1m+
old%x%age2%x%cL2m) %*% t(cM+ age%x%cL1m+ old%x%age2%x%cL2m))
covEM12<-mxAlgebra(name= "EM12", expression = (eM+ age%x%eL1m+
old%x%age2%x%eL2m) %*% t(eM+ age%x%eL1m+ old%x%age2%x%eL2m))

covAM21<-mxAlgebra(name= "AM21", expression = (aM+ age%x%aL1m+
old%x%age2%x%aL2m) %*% t(aM+ age%x%aL1m+ old%x%age2%x%aL2m))
covCM21<-mxAlgebra(name= "CM21", expression = (cM+ age%x%cL1m+
old%x%age2%x%cL2m) %*% t(cM+ age%x%cL1m+ old%x%age2%x%cL2m))
covEM21<-mxAlgebra(name= "EM21", expression = (eM+ age%x%eL1m+
old%x%age2%x%eL2m) %*% t(eM+ age%x%eL1m+ old%x%age2%x%eL2m))

covAF12<-mxAlgebra(name= "AF12", expression = (aF+ age%x%aL1f+
old%x%age2%x%aL2f) %*% t(aF+ age%x%aL1f+ old%x%age2%x%aL2f))
covCF12<-mxAlgebra(name= "CF12", expression = (cF+ age%x%cL1f+
old%x%age2%x%cL2f) %*% t(cF+ age%x%cL1f+ old%x%age2%x%cL2f))
covEF12<-mxAlgebra(name= "EF12", expression = (eF+ age%x%eL1f+
old%x%age2%x%eL2f) %*% t(eF+ age%x%eL1f+ old%x%age2%x%eL2f))

covAF21<-mxAlgebra(name= "AF21", expression = (aF+ age%x%aL1f+
old%x%age2%x%aL2f) %*% t(aF+ age%x%aL1f+ old%x%age2%x%aL2f))
covCF21<-mxAlgebra(name= "CF21", expression = (cF+ age%x%cL1f+
old%x%age2%x%cL2f) %*% t(cF+ age%x%cL1f+ old%x%age2%x%cL2f))
covEF21<-mxAlgebra(name= "EF21", expression = (eF+ age%x%eL1f+
old%x%age2%x%eL2f) %*% t(eF+ age%x%eL1f+ old%x%age2%x%eL2f))

covAO12<-mxAlgebra(name= "AO12", expression = (aM+ age%x%aL1m+
old%x%age2%x%aL2m) %*% t(aF+ age%x%aL1f+ old%x%age2%x%aL2f))
covCO12<-mxAlgebra(name= "CO12", expression = (cM+ age%x%cL1m+
old%x%age2%x%cL2m) %*% t(cF+ age%x%cL1f+ old%x%age2%x%cL2f))
covEO12<-mxAlgebra(name= "EO12", expression = (eM+ age%x%eL1m+
old%x%age2%x%eL2m) %*% t(eF+ age%x%eL1f+ old%x%age2%x%eL2f))

covAO21<-mxAlgebra(name= "AO21", expression = (aF+ age%x%aL1f+
old%x%age2%x%aL2f) %*% t(aM+ age%x%aL1m+ old%x%age2%x%aL2m))
covCO21<-mxAlgebra(name= "CO21", expression = (cF+ age%x%cL1f+
old%x%age2%x%cL2f) %*% t(cM+ age%x%cL1m+ old%x%age2%x%cL2m))
covEO21<-mxAlgebra(name= "EO21", expression = (eF+ age%x%eL1f+
old%x%age2%x%eL2f) %*% t(eM+ age%x%eL1m+ old%x%age2%x%eL2m))

pathrG    <- mxMatrix( type="Full", nrow=1, ncol=1, free=TRUE,
values=0.50,label="rGo", name="rG",ubound=0.50)
pathrE           <-mxMatrix( type="Full", nrow=1, ncol=1, free=TRUE,
values=1.00, label="rEo", name="rE", ubound=1.00)
```

```
# Algebra to compute total variances and standard deviations (diagonal only)
#MALES
covPm      <- mxAlgebra( expression=Am+Cm+Em, name="Vm" )
matIm      <- mxMatrix( type="Iden", nrow=nv, ncol=nv, name="Im")
invSDm     <- mxAlgebra( expression=solve(sqrt(Im*Vm)), name="iSDm")

#FEMALES
covPf      <- mxAlgebra( expression=Af+Cf+Ef, name="Vf" )
matIf      <- mxMatrix( type="Iden", nrow=nv, ncol=nv, name="If")
invSDf     <- mxAlgebra( expression=solve(sqrt(If*Vf)), name="iSDf")

# Algebras generated to create summary Table of Derived Variance Components
#males
rowVarsm   <- rep('varsm',nv)
colVarsm   <- rep(c('Am','Cm','Em','SAm','SCm','SEm'),each=nv)
estVarsm   <- mxAlgebra( expression=cbind(Am,Cm,Em,Am/Vm,Cm/Vm,Em/Vm),
name="Varsm", dimnames=list(rowVarsm,colVarsm) )

#females
rowVarsf   <- rep('varsf',nv)
colVarsf   <- rep(c('Af','Cf','Ef','SAf','SCf','SEf'),each=nv)
estVarsf   <- mxAlgebra( expression=cbind(Af,Cf,Ef,Af/Vf,Cf/Vf,Ef/Vf),
name="Varsf", dimnames=list(rowVarsf,colVarsf) )


###algebras generated to create estimated means for men and women

rowMeanm <-rep('means',nv)
colMeanm <-rep('meanm',each=nv)
estMeanm <-mxAlgebra( expression=cbind(meanm),"estMeanm",
dimnames=list(rowMeanm,colMeanm))

rowMeanf <-rep('means',nv)
colMeanf <-rep('meanf',each=nv)
estMeanf <-mxAlgebra( expression=cbind(meanf),"estMeanf",
dimnames=list(rowMeanf,colMeanf))

# Algebra for expected Mean and Variance/Covariance Matrices in MZ & DZ twins
#Males MZ and DZ
meanAgem   <- mxAlgebra( expression= bm%*%rbind(age,old*age2), name="AgeRm")
meanGIm    <- mxAlgebra( expression= cbind((expMeanm + AgeRm),(expMeanm +
AgeRm)), name="expMeanGm")

 expCovMZM <- mxAlgebra(name = "expCovMZM",
                      expression = rbind (cbind(AM1+CM1+EM1, AM12+CM12),
                                          cbind(AM21+CM21,   AM2+CM2+EM2)))

expCovDZM <- mxAlgebra(name = "expCovDZM",
                      expression = rbind (cbind(AM1+CM1+EM1, 0.5%x%AM12+CM12),
                                          cbind(0.5%x%AM21+CM21,
AM2+CM2+EM2)))
```

```
##Femal MZ and DZ

meanAgef   <- mxAlgebra( expression= bf%*%rbind(age,old*age2), name="AgeRf")
meanGIf    <- mxAlgebra( expression= cbind((expMeanf + AgeRf),(expMeanf +
AgeRf)), name="expMeanGf")
expCovMZF <- mxAlgebra(name = "expCovMZF",
                       expression = rbind (cbind(AF1+CF1+EF1, AF12+CF12),
                                           cbind(AF21+CF21,   AF2+CF2+EF2)))

expCovDZF <- mxAlgebra(name = "expCovDZF",
                       expression = rbind (cbind(AF1+CF1+EF1, 0.5%x%AF12+CF12),
                                           cbind(0.5%x%AF21+CF21,
AF2+CF2+EF2)))
##Expected means opposite sex pairs


meanGImf    <- mxAlgebra( expression= cbind((expMeanm + AgeRm),(expMeanf +
AgeRf)), name="expMeanGmf")

# Expected covariance matrix in opposite sex pairs, note use of rg and rc
expCovOZ    <- mxAlgebra(name="expCovOZ",
expression = rbind (cbind(AM1+CM1+EM1, rG%x%AO12+rE%x%CO12),
                                           cbind(rG%x%AO21+rE%x%CO21,
AF2+CF2+EF2)))


# Data objects for Multiple Groups
dataMZf   <- mxData( observed=mzFData, type="raw" )
dataDZf   <- mxData( observed=dzFData, type="raw" )
dataMZm   <- mxData( observed=mzMData, type="raw" )
dataDZm   <- mxData( observed=dzMData, type="raw" )
dataDZO   <- mxData( observed=dzODData, type="raw" )


# Objective objects for Multiple Groups
objMZf    <- mxExpectationNormal( covariance="expCovMZF", means="expMeanGf",
dimnames=selVars )
objDZf    <- mxExpectationNormal( covariance="expCovDZF", means="expMeanGf",
dimnames=selVars )
objMZm    <- mxExpectationNormal( covariance="expCovMZM", means="expMeanGm",
dimnames=selVars )
objDZm    <- mxExpectationNormal( covariance="expCovDZM", means="expMeanGm",
dimnames=selVars )
objDZo    <- mxExpectationNormal( covariance="expCovOZ", means="expMeanGmf",
dimnames=selVars )

fitFunction<-mxFitFunctionML()

parsm      <- list( pathAm, pathCm, pathEm,
pathASm, pathCSm, pathESm,
pathAS2m,pathCS2m,pathES2m,
pathBm, meanm)

parsf      <- list( pathAf, pathCf, pathEf,
```

```
pathASf, pathCSf, pathESf,
pathAS2f,pathCS2f,pathES2f,
pathBf, meanf)


defsm      <- list( defAge,defold,oldage,meanAgem,meanm)
defsf      <- list( defAge,defold,oldage,meanAgef,meanf)
defsmf     <- list(defAge, defold,oldage,meanAgem, meanAgef,meanm,meanf)

modelMZm   <- mxModel( parsm,
covAMmod1,covCMmod1,covEMmod1,
covAMmod2,covCMmod2,covEMmod2,
covAM12,covCM12,covEM12,covAM21,covCM21,covEM21,
meanm,defsm, meanGIm,  expCovMZM, dataMZm, objMZm,fitFunction, name="MZm" )


modelDZm   <- mxModel( parsm,covAMmod1,covCMmod1,covEMmod1,
covAMmod2,covCMmod2,covEMmod2,
covAM12,covCM12,covEM12,covAM21,covCM21,covEM21,
meanm,defsm, meanGIm,  expCovDZM, dataDZm, objDZm,fitFunction, name="DZm" )

modelMZf   <- mxModel( parsf,covAFmod1,covCFmod1,covEFmod1,
covAFmod2,covCFmod2,covEFmod2,
covAF12,covCF12,covEF12,covAF21,covCF21,covEF21, defsf, meanf, meanGIf,
expCovMZF, dataMZf, objMZf,fitFunction, name="MZf" )


modelDZf   <- mxModel( parsf, covAFmod1,covCFmod1,covEFmod1,
covAFmod2,covCFmod2,covEFmod2,
covAF12,covCF12,covEF12,covAF21,covCF21,covEF21, defsf, meanGIf,  expCovDZF,
dataDZf, objDZf,fitFunction, name="DZf" )

modelDZo  <- mxModel( parsf,parsm, pathrG, pathrE,
covAMmod1,covCMmod1,covEMmod1,
covAFmod2,covCFmod2,covEFmod2,
covAO12,covCO12,covEO12,covAO21,covCO21,covEO21,
defsmf,  expCovOZ, meanGImf, dataDZO, objDZo,fitFunction, name="DZo" )


minus2ll  <- mxAlgebra( MZf.fitfunction+ DZf.fitfunction+ MZm.fitfunction+
DZm.fitfunction+ DZo.fitfunction, name="m2LL" )
obj       <- mxFitFunctionAlgebra( "m2LL" )
QualAceModel  <- mxModel( "Full ACE 75 turning", parsf, parsm, modelMZf,
modelDZf, modelMZm, modelDZm, modelDZo, minus2ll, obj )


# ------------------------------------------------------------------------------
# RUN MODEL

###turn off hessian and standard error calculation

QualAceModel<- mxOption(QualAceModel, "Calculate Hessian", "No")
QualAceModel<- mxOption(QualAceModel, "Standard Errors", "No")
```

```
# Run Qualitative Sex Differences ACE model
QualAceFit    <- mxRun(QualAceModel)
QualAceFit    <- mxRun(QualAceFit)
QualAceSumm   <- summary(QualAceFit)
QualAceSumm

####REMOVE QUALITATIVE SEX DIFFERENCES##

qual<-mxModel(QualAceFit, name="drop qualitative sex differences")
qual<-omxSetParameters(qual, labels=c('rGo'), free=F, values=0.50)
qual<-omxSetParameters(qual, labels=c('rEo'), free=F, values=1.00)
qual<-
mxModel(qual,mxCI(c("aF","cF","eF","aL1f","cL1f","eL1f","aL2f","cL2f","eL2f","bf
",
"aM","cM","eM","aL1m","cL1m","eL1m","aL2m","cL2m","eL2m","bm")))
qualfit<-mxRun(qual)
summary(qualfit)
tableFitStatistics(QualAceFit,qualfit)



###REMOVE QUANTITATIVE SEX DIFFERENCESL######

spline2sex<-mxModel(qualfit, name="remove sex differences")
spline2sex<-omxSetParameters(spline2sex, labels=
c("af11","cf11","ef11","aS1f","cS1f","eS1f","aS2f","cS2f","eS2f"),
newlabels=c("a11","c11","e11","aS1","cS1","eS1","aS2","cS2","eS2"))
spline2sex<-omxSetParameters(spline2sex, labels=
c("am11","cm11","em11","aS1m","cS1m","eS1m","aS2m","cS2m","eS2m"),
newlabels=c("a11","c11","e11","aS1","cS1","eS1","aS2","cS2","eS2"))

spline2sex<-omxAssignFirstParameters(spline2sex)

spline2sexfit<-mxRun(spline2sex)
tableFitStatistics(qualfit,spline2sexfit)


###REMOVE QUANTITATIVE SEX DIFFERENCES ON A ONLY######

spline2sex<-mxModel(qualfit, name="remove sex differences on A")
spline2sex<-omxSetParameters(spline2sex, labels= c("af11","aS1f","aS2f"),
newlabels=c("a11","aS1","aS2"))
spline2sex<-omxSetParameters(spline2sex, labels= c("am11","aS1m","aS2m"),
newlabels=c("a11","aS1","aS2"))

spline2sex<-omxAssignFirstParameters(spline2sex)

spline2sexfit<-mxRun(spline2sex)
tableFitStatistics(qualfit,spline2sexfit)
summary(spline2sexfit)
```

```
###drop C all####

spline3C<-mxModel(qualfit, name="drop all C")
spline3C<-omxSetParameters(spline3C,
labels=c('cf11','cS1f','cS2f','cm11','cS1m','cS2m'), free=F, values=0)
spline3C<-mxModel(spline3C, mxCI(c("aF","eF","aL1f","eL1f","aL2f","eL2f","bf",
"aM","eM","aL1m","eL1m","aL2m","eL2m","bm")))
spline3Cfit<-mxRun(spline3C, intervals=F)
summary(spline3Cfit)
tableFitStatistics(qualfit,spline3Cfit)


####REMOVE 75-90 VARIANCE AGE MODERATION####

spline2<-mxModel(spline3Cfit, name="drop second turning point variance only")
spline2<-omxSetParameters(spline2, labels=c('aS2f','eS2f','aS2m','eS2m'),
free=F, values=0)

spline2fit<-mxRun(spline2)
summary(spline2fit)
tableFitStatistics(spline3Cfit,spline2fit)




###REMOVE 40-75 variance age moderationl#####

spline1<-mxModel(spline3Cfit, name="remove all first turning point variance
moderation")
spline1<-omxSetParameters(spline1, labels=c('aS1f','eS1f',"aS1m","eS1m"),
free=F, values=0)

spline1fit<-mxRun(spline1)
summary(spline1fit)
tableFitStatistics(spline3Cfit,spline1fit)




####get 95% CI

CI<-mxModel(qualfit,name="get CI")

CI<-mxModel(CI,
mxCI(c("aF","cF","eF","aL1f","cL1f","eL1f","aL2f","cL2f","eL2f","bf",
"aM","cM","eM","aL1m","cL1m","eL1m","aL2m","cL2m","eL2m","bm")))
CIfit<-mxRun(CI, intervals=T)
summary(CIfit)
```

```
#######TEST MODELS WITHIN MALES


###drop all AE AGE variance moderationmales#####

splineomnmale<-mxModel(spline3Cfit, name="remove all age moderation males")
splineomnmale<-omxSetParameters(splineomnmale,
labels=c('aS1m','eS1m',"aS2m","eS2m"), free=F, values=0)

splineOMNmalefit<-mxRun(splineomnmale)
summary(splineOMNmalefit)
tableFitStatistics(spline3Cfit,splineOMNmalefit)


##drop 75+ AGE SLOPE A MALES####

splineAm1<-mxModel(spline3Cfit, name="drop second slope A males")
splineAm1<-omxSetParameters(splineAm1, labels=c('aS2m'), free=F, values=0)

splineAm1fit<-mxRun(splineAm1)
summary(splineAm1fit)
tableFitStatistics(spline3Cfit,splineAm1fit)


##drop 40-75 AGE SLOPE A males####

splineAm2<-mxModel(spline3Cfit, name="drop first slope A males")
splineAm2<-omxSetParameters(splineAm2, labels=c('aS1m'), free=F, values=0)

splineAm2fit<-mxRun(splineAm2)
summary(splineAm2fit)
tableFitStatistics(spline3Cfit,splineAm2fit)



###drop both age moderation  A males ####

splineAm3<-mxModel(spline3Cfit, name="drop both slopes A males")
splineAm3<-omxSetParameters(splineAm3, labels=c('aS1m',"aS2m"), free=F,
values=0)

splineAm3fit<-mxRun(splineAm3)

summary(splineAm3fit)
tableFitStatistics(spline3Cfit,splineAm3fit)



###drop 75+ Age slope E males ####

splineEm2<-mxModel(spline3Cfit, name="drop second turning point E males")
splineEm2<-omxSetParameters(splineEm2, labels=c('eS2m'), free=F, values=0)
```

```
splineEm2fit<-mxRun(splineEm2)
summary(splineEm2fit)
tableFitStatistics(spline3Cfit,splineEm2fit)


##drop 40-75 AGE slope E males####

splineEm1<-mxModel(spline3Cfit, name="drop first slope E males")
splineEm1<-omxSetParameters(splineEm1, labels=c('eS1m'), free=F, values=0)

splineEm1fit<-mxRun(splineEm1)
summary(splineEm1fit)
tableFitStatistics(spline3Cfit,splineEm1fit)



###drop both age moderation E males ####

splineEm3<-mxModel(spline3Cfit, name="drop both slopes E males")
splineEm3<-omxSetParameters(splineEm3, labels=c('eS1m','eS2m'), free=F,
values=0)

splineEm3fit<-mxRun(splineEm3)
summary(splineEm3fit)
tableFitStatistics(spline3Cfit,splineEm3fit)



#######FEMALE
MODELS#############################################################

###drop all AE turning point females#####

splineomnfemale<-mxModel(spline3Cfit, name="remove all age moderation females")
splineomnfemale<-omxSetParameters(splineomnfemale,
labels=c('aS1f','aS2f','eS1f','eS2f'), free=F, values=0)

splineOMNfemalefit<-mxRun(splineomnfemale)
summary(splineOMNfemalefit)
tableFitStatistics(spline3Cfit,splineOMNfemalefit)


##drop first slope A females####

splineAf1<-mxModel(spline3Cfit, name="drop first slope A females")
splineAf1<-omxSetParameters(splineAf1, labels=c('aS1f'), free=F, values=0)

splineAf1fit<-mxRun(splineAf1)
summary(splineAf1)
tableFitStatistics(spline3Cfit,splineAf1fit)

###drop SECOND turning point A females ####

splineAf2<-mxModel(spline3Cfit, name="drop second turning point A females")
```

```
splineAf2<-omxSetParameters(splineAf2, labels=c('aS2f'), free=F, values=0)

splineAf2fit<-mxRun(splineAf2)
summary(splineAf2fit)
tableFitStatistics(spline3Cfit,splineAf2fit)

###drop both turning point A females ####

splineAf3<-mxModel(spline3Cfit, name="drop both A slopes females")
splineAf3<-omxSetParameters(splineAf3, labels=c('aS1f','aS2f'), free=F,
values=0)

splineAf3fit<-mxRun(splineAf3)

tableFitStatistics(spline3Cfit,splineAf3fit)
summary(splineAf3fit)


##drop first slope E females####

splineEf1<-mxModel(spline3Cfit, name="drop first slope E females")
splineEf1<-omxSetParameters(splineEf1, labels=c('eS1f'), free=F, values=0)

splineEf1fit<-mxRun(splineEf1)
summary(splineEf1fit)
tableFitStatistics(spline3Cfit,splineEf1fit)

###drop SECOND turning point E females ####

splineEf2<-mxModel(spline3Cfit, name="drop second turning point E females")
splineEf2<-omxSetParameters(splineEf2, labels=c('eS2f'), free=F, values=0)

splineEf2fit<-mxRun(splineEf2)
summary(splineEf2fit)
tableFitStatistics(spline3Cfit,splineEf2fit)


###drop both E slopes females ####

splineEf3<-mxModel(spline3Cfit, name="drop both slopes E females")
splineEf3<-omxSetParameters(splineEf3, labels=c('eS1f','eS2f'), free=F,
values=0)

splineEf3fit<-mxRun(splineEf3)
summary(splineEf3fit)

tableFitStatistics(spline3Cfit,splineEf3fit)




allmodels<-c(spline3Cfit,splineAm1fit, splineAm2fit, splineAm3fit, splineEm1fit,
splineEm2fit,
```

```
splineEm3fit, splineAf1fit, splineAf2fit,splineAf3fit,
splineEf1fit,splineEf2fit, splineEf3fit )
tableFitStatistics(spline3Cfit,allmodels)
```