

Sex-limitation and moderation models

Practical

Conor Dolan & Eveline de Zeeuw

*Thanks to Sarah Medland, Robert Kirkpatrick, Michael
Hunter and many others*

COPY FILES FROM:

/home/eveline/2020/Heterogeneity

G x Sex interaction

Example

- Measurement
 - BMI at age 12
 - Based on mother-reported height and weight
- Sample
 - Netherlands Twin Register
 - 971 MZM, 878 DZM, 1015 MZF, 835 DZF and 1715 DOS pairs
- Research question
 - Are there qualitative and/or quantitative sex differences in BMI?

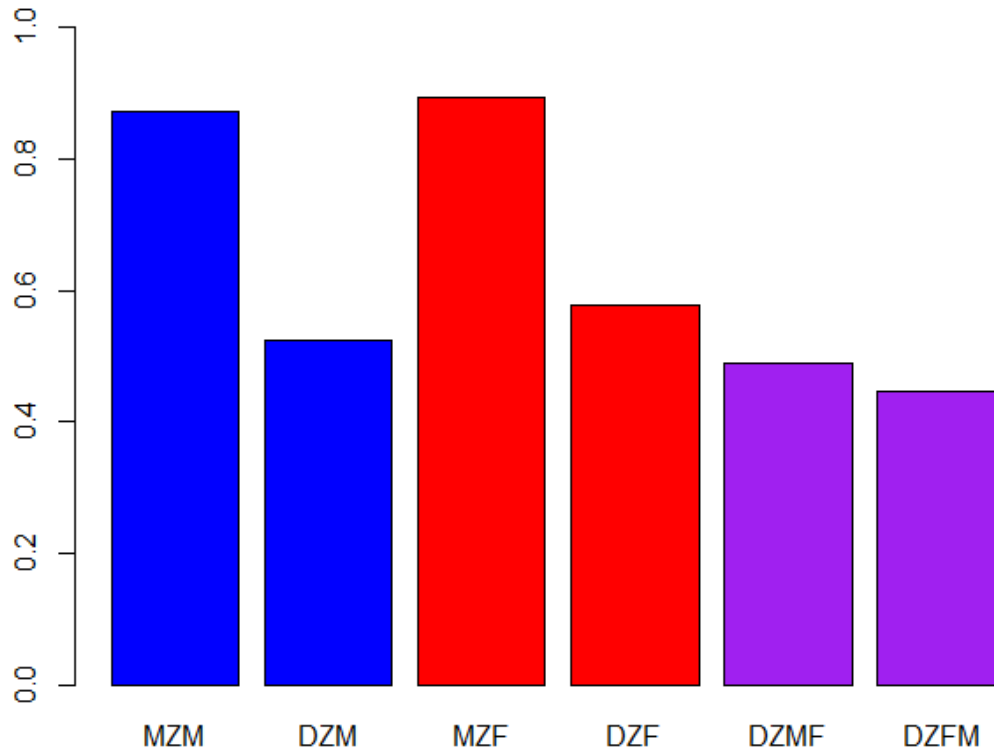
Data

```
# Load Data
BMIData <- read.table(file='dataBMI.dat',header=TRUE, na=-999)
head(BMIData)
dim(BMIData)

# select variables for Analysis
vars <- 'BMI' # list of variables names
nv <- 1 # number of variables
ntv <- nv*2 # number of total variables
selvars <- paste(vars,c(rep('_tw1',nv),rep('_tw2',nv)),sep='')

# select Data for Analysis
mzmData <- subset(BMIData, zyg==1, selvars)
dzmData <- subset(BMIData, zyg==2, selvars)
mzfData <- subset(BMIData, zyg==3, selvars)
dzfData <- subset(BMIData, zyg==4, selvars)
dzmfData <- subset(BMIData, zyg==5, selvars)
dzfmData <- subset(BMIData, zyg==6, selvars)
```

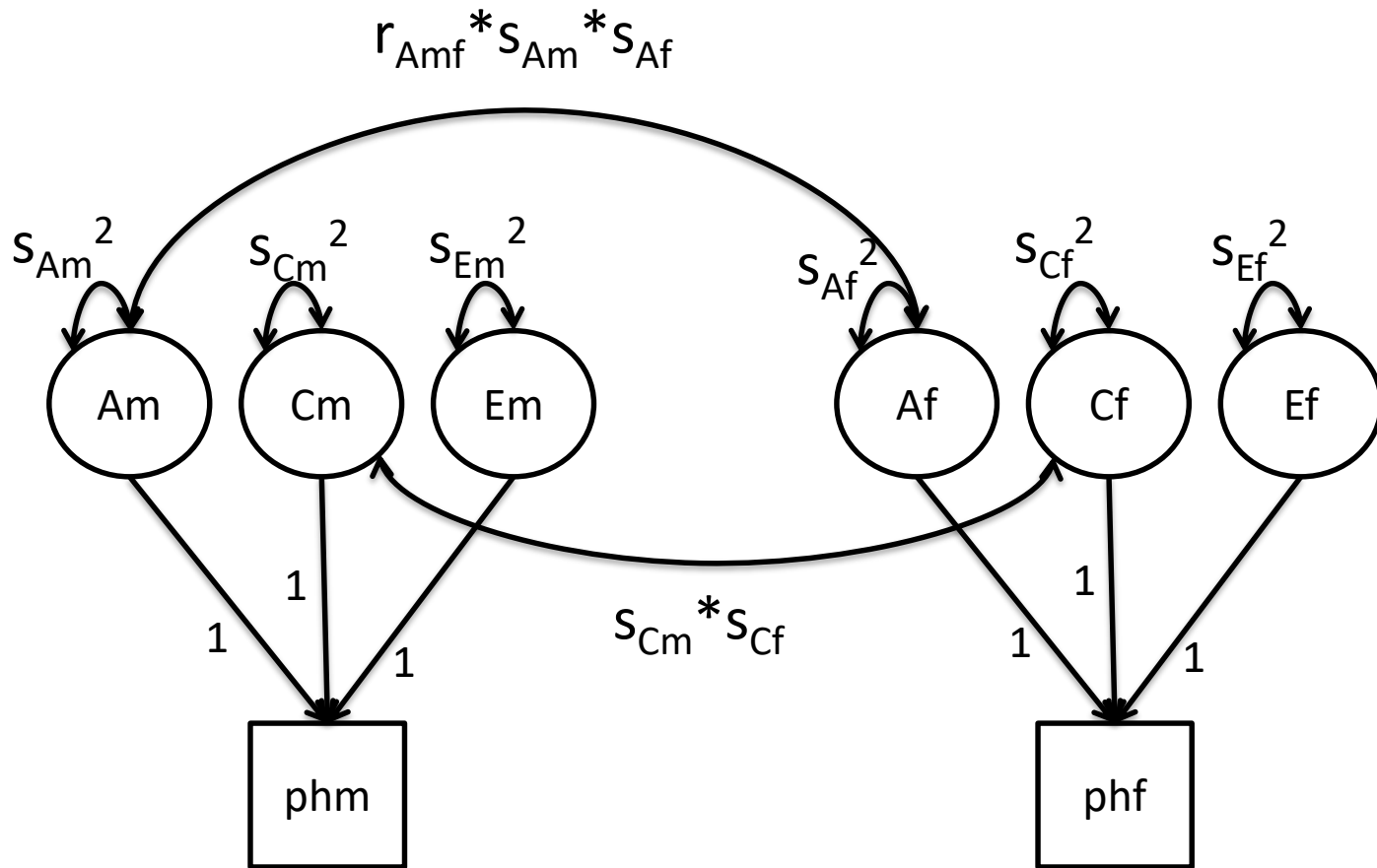
Twin Correlations



Means

```
# Create Algebra for expected Mean Matrices
```

```
meanGm      <- mxMatrix(type="Full", nrow=1, ncol=nv, free=TRUE, values=svMem, labels="meanm", name="meanGm")  
meanGf      <- mxMatrix(type="Full", nrow=1, ncol=nv, free=TRUE, values=svMef, labels="meanf", name="meanGf")  
expMeanMZm  <- mxAlgebra(expression= cbind(meanGm, meanGm), name="expMeanMZm")  
expMeanDZm  <- mxAlgebra(expression= cbind(meanGm, meanGm), name="expMeanDZm")  
expMeanMZf  <- mxAlgebra(expression= cbind(meanGf, meanGf), name="expMeanMZf")  
expMeanDZf  <- mxAlgebra(expression= cbind(meanGf, meanGf), name="expMeanDZf")  
expMeanDZmf <- mxAlgebra(expression= cbind(meanGm, meanGf), name="expMeanDZmf")  
expMeanDZfm <- mxAlgebra(expression= cbind(meanGf, meanGm), name="expMeanDZfm")
```



$S_{Am}^2 + S_{Cm}^2 + S_{Em}^2$	
$r_A * S_{Am} S_{Af} + S_{Cm} S_{Cf}$	$S_{Af}^2 + S_{Cf}^2 + S_{Ef}^2$

(Co)variances

```
# Create Matrices for Variance Components
```

```
Am    <- mxMatrix(type="Symm", nrow=nv, ncol=nv, free=TRUE, values=svPam, label="A11m", name="Am")
Af    <- mxMatrix(type="Symm", nrow=nv, ncol=nv, free=TRUE, values=svPaf, label="A11f", name="Af")
Cm    <- mxMatrix(type="Symm", nrow=nv, ncol=nv, free=TRUE, values=svPcm, label="C11m", name="Cm")
Cf    <- mxMatrix(type="Symm", nrow=nv, ncol=nv, free=TRUE, values=svPcf, label="C11f", name="Cf")
Em    <- mxMatrix(type="Symm", nrow=nv, ncol=nv, free=TRUE, values=svPem, label="E11m", name="Em")
Ef    <- mxMatrix(type="Symm", nrow=nv, ncol=nv, free=TRUE, values=svPef, label="E11f", name="Ef")
```

```
# Create Matrix for Genetic Correlation between DOS twin pairs
```

```
RAos  <- mxMatrix(type="Symm", nrow=nv, ncol=nv, free=TRUE, values=.4, label="raos", name="RAos")
```

```
# Create Matrices for Total Variances
```

```
varPm  <- mxAlgebra(expression= Am+Cm+Em, name="vm")
varPf  <- mxAlgebra(expression= Af+Cf+Ef, name="vf")
```

```
# The following lines calculate the absolute covariation between Males and Females
```

```
Aos    <- mxAlgebra((sqrt((Af))%*%t(sqrt((Am))))), name="Aos")
Cos    <- mxAlgebra((sqrt((Cf))%*%t(sqrt((Cm))))), name="Cos")
```

```
# Create Algebra for expected Variance/Covariance Matrices in MZ, DZ and DOS twins
```

```
expCovMZm <- mxAlgebra( expression= rbind(cbind(Am+Cm+Em, Am+Cm),
                                          cbind(Am+Cm, Am+Cm+Em)), name="expCovMZm")
expCovDZm <- mxAlgebra( expression= rbind(cbind(Am+Cm+Em, 0.5%x%Am+Cm),
                                          cbind(0.5%x%Am+Cm, Am+Cm+Em)), name="expCovDZm")
expCovMZf <- mxAlgebra( expression= rbind(cbind(Af+Cf+Ef, Af+Cf),
                                          cbind(Af+Cf, Af+Cf+Ef)), name="expCovMZf")
expCovDZf <- mxAlgebra( expression= rbind(cbind(Af+Cf+Ef, 0.5%x%Af+Cf),
                                          cbind(0.5%x%Af+Cf, Af+Cf+Ef)), name="expCovDZf")
expCovDZmf <- mxAlgebra( expression= rbind(cbind(Am+Cm+Em, RAos%x%Aos+Cos),
                                          cbind(RAos%x%Aos+Cos, Af+Cf+Ef)), name="expCovDZmf")
expCovDZfm <- mxAlgebra( expression= rbind(cbind(Af+Cf+Ef, RAos%x%Aos+Cos),
                                          cbind(RAos%x%Aos+Cos, Am+Cm+Em)), name="expCovDZfm")
```

Model

```
# Create Data Objects for Multiple Groups
dataMZm <- mxData(observed=mzmData, type="raw")
dataDZm <- mxData(observed=dzmData, type="raw")
dataMZf <- mxData(observed=mzfData, type="raw")
dataDZf <- mxData(observed=dzfData, type="raw")
dataDZmf <- mxData(observed=dzmfData, type="raw")
dataDZfm <- mxData(observed=dzfmData, type="raw")

# Create Expectation Objects for Multiple Groups
expMZm <- mxExpectationNormal(covariance="expCovMZm", means="expMeanMZm", dimnames=selVars)
expDZm <- mxExpectationNormal(covariance="expCovDZm", means="expMeanDZm", dimnames=selVars)
expMZf <- mxExpectationNormal(covariance="expCovMZf", means="expMeanMZf", dimnames=selVars)
expDZf <- mxExpectationNormal(covariance="expCovDZf", means="expMeanDZf", dimnames=selVars)
expDZmf <- mxExpectationNormal(covariance="expCovDZmf", means="expMeanDZmf", dimnames=selVars)
expDZfm <- mxExpectationNormal(covariance="expCovDZfm", means="expMeanDZfm", dimnames=selVars)
funML <- mxFitFunctionML()

# Create Model Objects for Multiple Groups
pars <- list(meanGm, meanGf, Af, Cf, Ef, Am, Cm, Em, Aos, Cos, varPm, varPf, RAos)
modelMZm <- mxModel(name="MZm", pars, expMeanMZm, expCovMZm, dataMZm, expMZm, funML)
modelDZm <- mxModel(name="DZm", pars, expMeanDZm, expCovDZm, dataDZm, expDZm, funML)
modelMZf <- mxModel(name="MZf", pars, expMeanMZf, expCovMZf, dataMZf, expMZf, funML)
modelDZf <- mxModel(name="DZf", pars, expMeanDZf, expCovDZf, dataDZf, expDZf, funML)
modelDZmf <- mxModel(name="DZmf", pars, expMeanDZmf, expCovDZmf, dataDZmf, expDZmf, funML)
modelDZfm <- mxModel(name="DZfm", pars, expMeanDZfm, expCovDZfm, dataDZfm, expDZfm, funML)
multi <- mxFitFunctionMultigroup(c("MZm", "DZm", "MZf", "DZf", "DZmf", "DZfm"))

# Create Algebra for Variance Components
rowVC <- rep('VC',nv)
colVCm <- rep(c('Am', 'Cm', 'Em', 'SAM', 'SCm', 'SEm'),each=nv)
colVCf <- rep(c('Af', 'Cf', 'Ef', 'SAf', 'SCf', 'SEf'),each=nv)
estVCm <- mxAlgebra( expression=cbind(Am,Cm,Em,Am/Vm,Cm/Vm,Em/Vm), name="VCm", dimnames=list(rowVC,colVCm))
estVCf <- mxAlgebra( expression=cbind(Af,Cf,Ef,Af/Vf,Cf/Vf,Ef/Vf), name="VCf", dimnames=list(rowVC,colVCf))

# Build Model
modelACERq <- mxModel("ACERq", pars, modelMZf, modelMZm, modelDZf, modelDZm, modelDZmf, modelDZfm, multi, estVCm, estVCf)

# Run Non-Scalar General Sex-Limitation ACE Model
fitACERq <- mxRun(modelACERq)
(sumACERq <- summary(fitACERq))
```

Tests

1. Are there qualitative sex differences?
2. Are there quantitative sex differences?
3. Does the common environment play a role?

Your task

1. Open the script *G x Sex interaction.R*
2. Walk through the first part of the script
 - 1 = MZM, 2 = DZM, 3 = MZF, 4 = DZF, 5 = DZMF, 6 = DZFM
3. Run the saturated model
4. Have a look at the twin correlations
5. Run the non-scalar general sex-limitation model
6. Run the submodels ($\alpha = 0.05$)
 - For each model fill out the question marks
7. Report the standardized variance components of the best model
8. Make sure that you know what you are doing
9. Raise your hand if you have a question

Results

Model		EP	-2LL	df	AIC	Δ -2LL	Δ df	P
0. SAT		30	45706.65	10753	24560.65	-	-	-
1. ACErq	0	9	45829.65	10594	24641.65	123.00	21	<.001
2. ACEq	1	8	45831.79	10595	24641.79	2.13	1	.144
3. ACE	2	5	45866.29	10598	24670.29	34.50	3	<.001
4. AEboys	2	7	45842.87	10596	24650.87	11.08	1	<.001
5. AEGirls	2	7	45850.71	10596	24658.71	18.92	1	<.001

	mean	A	C	E	SA	SC	SE
Males	17.5	5.1	0.2	0.7	.85	.04	.11
Females	17.7	4.9	1.5	0.7	.70	.21	.10

G x Age interaction

Example

- Measurement
 - BMI in adulthood (> 18 year)
 - Based on reported height and weight
- Sample
 - Only males
 - Younger and older cohort
 - Australian Twin Register
 - 1232 MZ and 750 DZ pairs
- Research question
 - Is heritability of BMI dependent on age?

Data

```
# Load Data
data(twinData)
head(twinData)
?twinData

# Select Variables for Analysis
vars      <- 'bmi'                # list of variables names
modVars   <- 'age'                # list of moderator variables names
nv        <- 1                    # number of variables
ntv       <- nv*2                 # number of total variables
selVars   <- paste(vars,c(rep('1',nv),rep('2',nv)),sep="")

# Select Data for Analysis
mzData    <- subset(twinData, zyg==1|zyg==6, c(selVars,modVars))
dzData    <- subset(twinData, zyg==3|zyg==8, c(selVars,modVars))

# Select cases with no missings on the definition variable
mzData <- mzData[!is.na(mzData$age),]
dzData <- dzData[!is.na(dzData$age),]

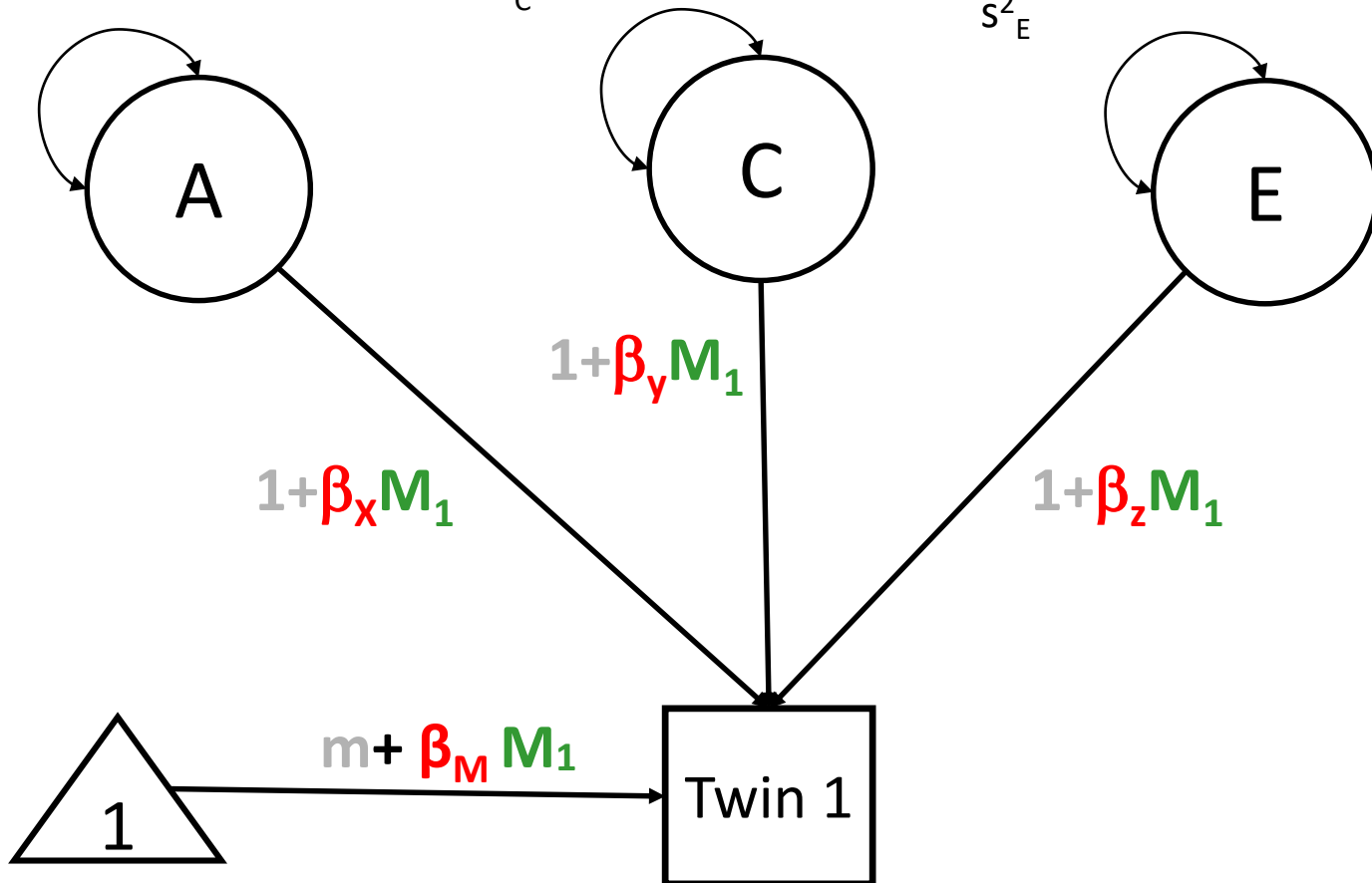
# Standardize age
mzData$age <- scale(mzData$age)
dzData$age <- scale(dzData$age)

# Generate Descriptive Statistics
round(colMeans(mzData,na.rm=TRUE),3)
round(colMeans(dzData,na.rm=TRUE),3)
round(cov(mzData,use="complete"),3)
round(cov(dzData,use="complete"),3)
round(cor(mzData,use="complete"),3)
round(cor(dzData,use="complete"),3)
```


S^2_A

S^2_C

S^2_E



Means

```
# Create Matrices for Covariates and linear Regression Coefficients
defAge    <- mxMatrix(type="Full", nrow=1, ncol=1, free=FALSE, labels=c("data.age"), name="defAge")
pathBAge  <- mxMatrix(type="Full", nrow=1, ncol=1, free=TRUE, values=.3, label="b11", name="BAge")

# Create Algebra for expected Mean Matrices
meanG     <- mxMatrix(type="Full", nrow=1, ncol=ntv, free=TRUE, values=c(21,21), labels=c("mean","mean"), name="meanG")
expMean   <- mxAlgebra(expression= meanG + cbind(defAge%%BAge,defAge%%BAge), name="expMean")
```

Covariances

```
# Matrices to store a, d, c, and e Path Coefficients (all fixed to 1)
pathA <- mxMatrix(type="Full", nrow=nv, ncol=1, free=FALSE, values=1, label="a11", name="a")
pathD <- mxMatrix(type="Full", nrow=nv, ncol=1, free=FALSE, values=1, label="d11", name="d")
pathC <- mxMatrix(type="Full", nrow=nv, ncol=1, free=FALSE, values=1, label="c11", name="c")
pathE <- mxMatrix(type="Full", nrow=nv, ncol=1, free=FALSE, values=1, label="e11", name="e")

# Matrices to compute the A, D, C, and E Variance Components
varA <- mxMatrix(type="Full", nrow=nv, ncol=1, free=TRUE, values=.6, label="VA11", name="VA")
varD <- mxMatrix(type="Full", nrow=nv, ncol=1, free=FALSE, values=0, label="VD11", name="VD")
varC <- mxMatrix(type="Full", nrow=nv, ncol=1, free=TRUE, values=.1, label="VC11", name="VC")
varE <- mxMatrix(type="Full", nrow=nv, ncol=1, free=TRUE, values=.5, label="VE11", name="VE")

# Matrices to store the moderating a, d, c, and e Path Coefficients
modPathA <- mxMatrix(type='Full', nrow=nv, ncol=1, free=TRUE, values=.1, label="aM11", name="aM")
modPathD <- mxMatrix(type='Full', nrow=nv, ncol=1, free=FALSE, values=0, label="dM11", name="dM")
modPathC <- mxMatrix(type='Full', nrow=nv, ncol=1, free=TRUE, values=.1, label="cM11", name="cM")
modPathE <- mxMatrix(type='Full', nrow=nv, ncol=1, free=TRUE, values=.1, label="eM11", name="eM")

# Matrices to compute the moderated A, D, C, and E variance components
modVarA <- mxAlgebra(expression=((a+ defAge%%aM) %% VA %% t(a+ defAge%%aM)), name="A")
modVarD <- mxAlgebra(expression=((d+ defAge%%dM) %% VD %% t(d+ defAge%%dM)), name="D")
modVarC <- mxAlgebra(expression=((c+ defAge%%cM) %% VC %% t(c+ defAge%%cM)), name="C")
modVarE <- mxAlgebra(expression=((e+ defAge%%eM) %% VE %% t(e+ defAge%%eM)), name="E")

# Algebra to compute the total variance per twin
varP <- mxAlgebra(expression=A+D+C+E, name="V")

# Algebra for the expected mean vector and expected variance/covariance matrices
covMZ <- mxAlgebra(expression= rbind (cbind(A+D+C+E , A+D+C),
                                     cbind(A+D+C, A+D+C+E)), name="expCovMZ")
covDZ <- mxAlgebra(expression= rbind (cbind(A+D+C+E , 0.5%x%A+0.25%x%D+C),
                                     cbind(0.5%x%A+0.25%x%D+C, A+D+C+E)), name="expCovDZ")
```

Model

```
# Data objects for Multiple Groups
dataMZ <- mxData(observed=mzData, type="raw")
dataDZ <- mxData(observed=dzData, type="raw")

# Objective objects for Multiple Groups
objMZ <- mxExpectationNormal(covariance="expCovMZ", means="expMean", dimnames=selVars)
objDZ <- mxExpectationNormal(covariance="expCovDZ", means="expMean", dimnames=selVars)
funML <- mxFitFunctionML()

# Combine Groups
pars <- list(pathA, pathD, pathC, pathE, modPathA, modPathD, modPathC, modPathE,
             defAge, pathBAge, modVarA, modVarD, modVarC, modVarE, varP, varA, varD, varC, varE)
modelMZ <- mxModel(pars, meanG, expMean, covMZ, dataMZ, objMZ, funML, name="MZ")
modelDZ <- mxModel(pars, meanG, expMean, covDZ, dataDZ, objDZ, funML, name="DZ")
minus2ll <- mxAlgebra( expression=MZ.objective+ DZ.objective, name="m2LL" )
obj <- mxFitFunctionAlgebra( "m2LL" )

# Build Model
modACEModel <- mxModel( "modACE", pars, modelMZ, modelDZ, minus2ll, obj, dataMZ, dataDZ )

# Run Moderation ACE model
modACEFit <- mxRun(modACEModel)
(modACESumm <- summary(modACEFit))
round(modACEFit@output$estimate,4)
```

Tests

1. Does the mean of BMI depend on age?
2. Does the common environment play a role?
3. Does the influence of genes and the environment depend on age?

Your task

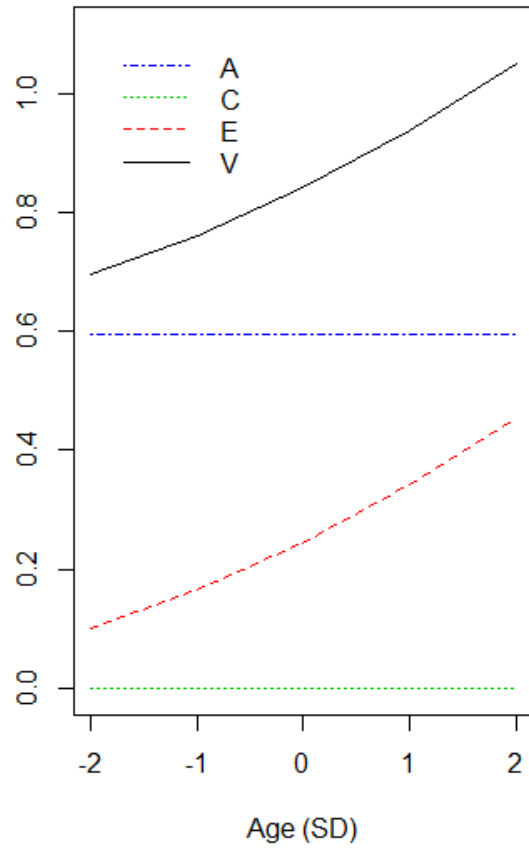
1. Open the script *G x Age interaction.R*
2. Walk through the first part of the script
 - 1 = MZm young, 3 = DZm young, 6 = MZm old, 8 = DZm old
3. Run the first part of the script
4. Run the submodels ($\alpha = 0.05$)
 - For each model fill out the question marks
5. Plot the (un)standardized variance components of the 'best' model
6. Make sure that you know what you are doing
7. Raise your hand if you have a question

Results

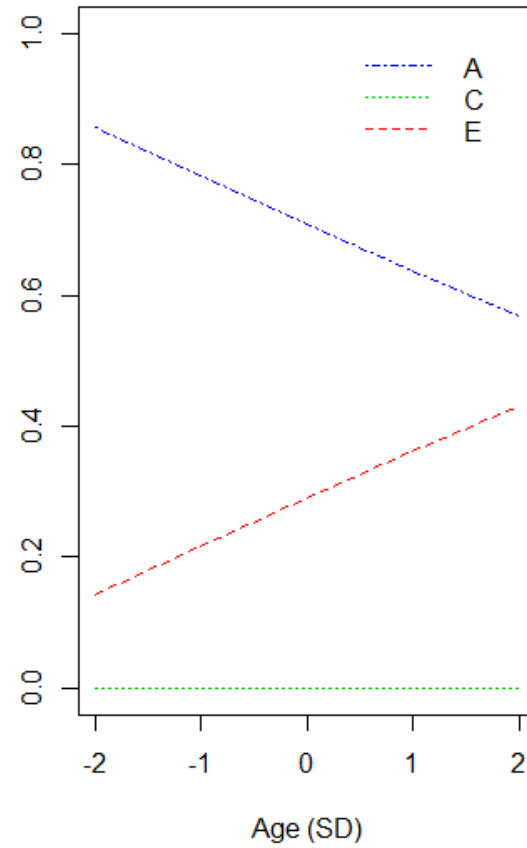
Model		EP	-2LL	df	AIC	Δ -2LL	Δ df	P
1. modACE		8	9321.68	3845	1631.68	-	-	-
2. modAE	1	6	9325.22	3847	1631.22	3.54	2	.171
3. AEnoModMean	2	5	9552.43	3848	1856.43	227.21	1	<.001
4. AEnoModAE	2	4	9395.07	3849	1697.07	69.84	2	<.001
5. AEnoModA	2	5	9328.26	3848	1632.26	3.04	1	.081
6. AEnoModE	5	4	9395.07	3849	1697.07	66.81	1	<.001

Results

Unstandardized variance components



Standardized variance components



Questions?