Re-introduction to openMx

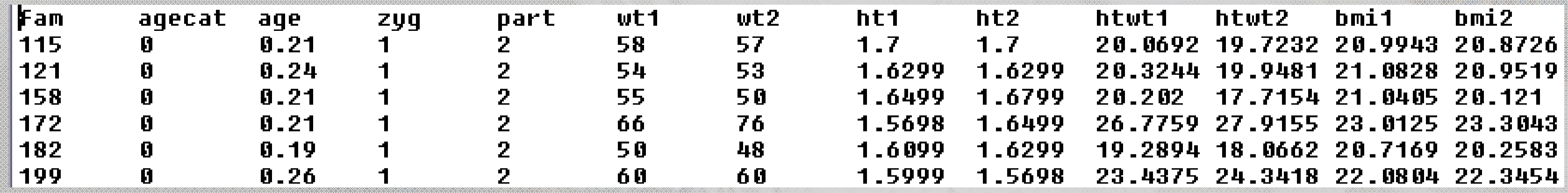
Sarah Medland

Starting at the beginning…

- Data preparation
  - The algebra style used in Mx expects 1 line per case/family
  - (Almost) limitless number of families and variables
  - Data needs to be read into R before it can be analysed
    - (the commands to read the data can be nested within the R script)
    - Default missing code is now **NA**

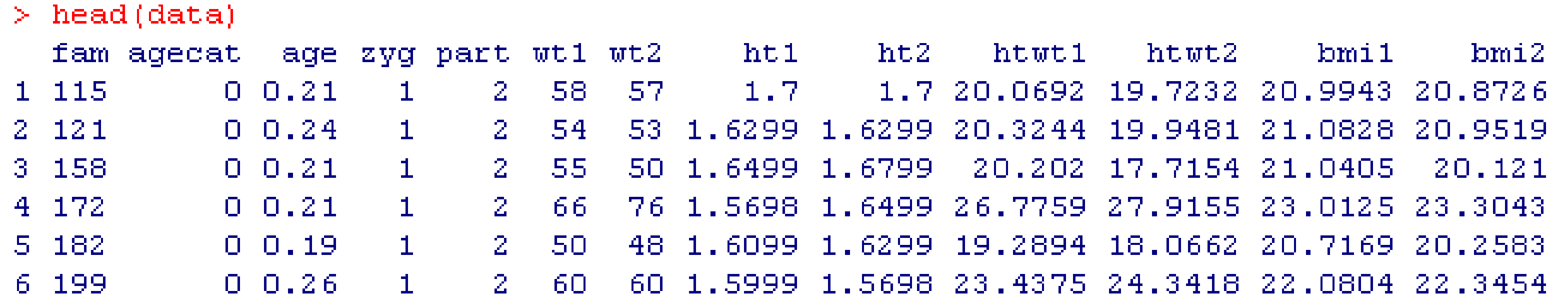
Getting your data into R

- Example data:ozbmi2.txt



```
fam     agecat  age     zyg     part    wt1     wt2     ht1     ht2     htwt1   htwt2   bmi1    bmi2
115     0       0.21    1       2       58      57      1.7     1.7     20.0692 19.7232 20.9943 20.8726
121     0       0.24    1       2       54      53      1.6299  1.6299  20.3244 19.9481 21.0828 20.9519
158     0       0.21    1       2       55      50      1.6499  1.6799  20.202  17.7154 21.0405 20.121
172     0       0.21    1       2       66      76      1.5698  1.6499  26.7759 27.9155 23.0125 23.3043
182     0       0.19    1       2       50      48      1.6099  1.6299  19.2894 18.0662 20.7169 20.2583
199     0       0.26    1       2       60      60      1.5999  1.5698  23.4375 24.3418 22.0804 22.3454
```

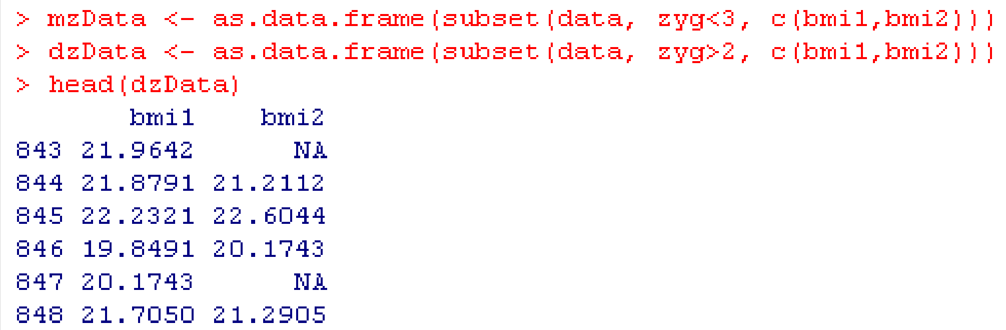
- data<-read.table("ozbmi2.txt", header=T, na.strings = "NA")
- head(data)



```
> head(data)
  fam agecat  age zyg part wt1 wt2    ht1    ht2   htwt1   htwt2    bmi1    bmi2
1 115      0 0.21   1    2  58  57    1.7    1.7 20.0692 19.7232 20.9943 20.8726
2 121      0 0.24   1    2  54  53 1.6299 1.6299 20.3244 19.9481 21.0828 20.9519
3 158      0 0.21   1    2  55  50 1.6499 1.6799  20.202 17.7154 21.0405  20.121
4 172      0 0.21   1    2  66  76 1.5698 1.6499 26.7759 27.9155 23.0125 23.3043
5 182      0 0.19   1    2  50  48 1.6099 1.6299 19.2894 18.0662 20.7169 20.2583
6 199      0 0.26   1    2  60  60 1.5999 1.5698 23.4375 24.3418 22.0804 22.3454
```

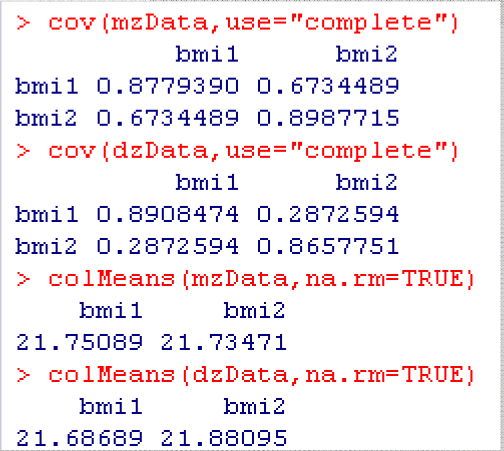
Selecting and sub-setting data

- Make separate data sets for the MZ and DZ



```
> mzData <- as.data.frame(subset(data, zyg<3, c(bmi1,bmi2)))
> dzData <- as.data.frame(subset(data, zyg>2, c(bmi1,bmi2)))
> head(dzData)
       bmi1    bmi2
843 21.9642      NA
844 21.8791 21.2112
845 22.2321 22.6044
846 19.8491 20.1743
847 20.1743      NA
848 21.7050 21.2905
```

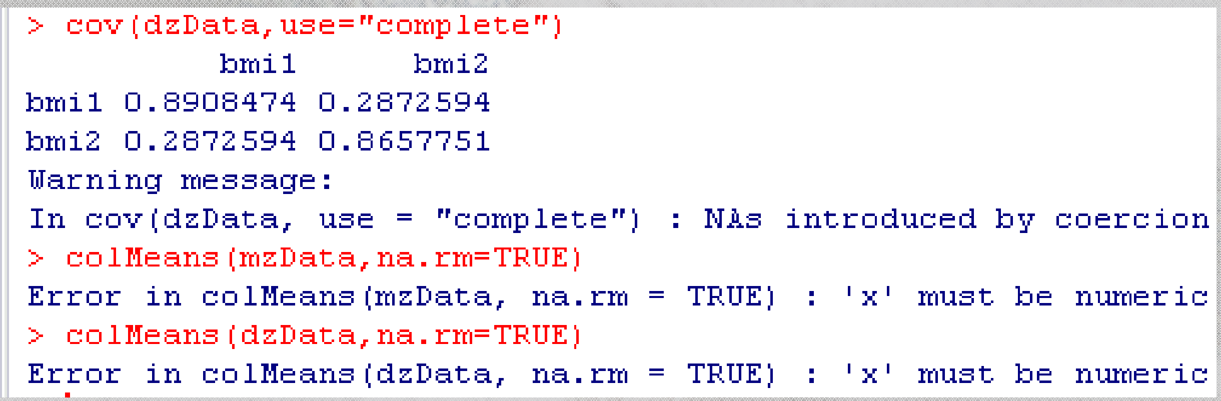
- Check data is numeric and behaves as expected



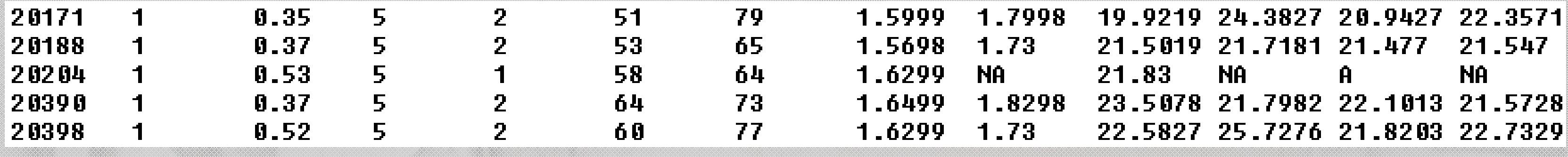
```
> cov(mzData,use="complete")
          bmi1      bmi2
bmi1 0.8779390 0.6734489
bmi2 0.6734489 0.8987715
> cov(dzData,use="complete")
          bmi1      bmi2
bmi1 0.8908474 0.2872594
bmi2 0.2872594 0.8657751
> colMeans(mzData,na.rm=TRUE)
    bmi1     bmi2
21.75089 21.73471
> colMeans(dzData,na.rm=TRUE)
    bmi1     bmi2
21.68689 21.88095
```

Common problem

- Problem: data contains a non numeric value



```
> cov(dzData,use="complete")
          bmi1      bmi2
bmi1 0.8908474 0.2872594
bmi2 0.2872594 0.8657751
Warning message:
In cov(dzData, use = "complete") : NAs introduced by coercion
> colMeans(mzData,na.rm=TRUE)
Error in colMeans(mzData, na.rm = TRUE) : 'x' must be numeric
> colMeans(dzData,na.rm=TRUE)
Error in colMeans(dzData, na.rm = TRUE) : 'x' must be numeric
```



20171	1	0.35	5	2	51	79	1.5999	1.7998	19.9219	24.3827	20.9427	22.3571
20188	1	0.37	5	2	53	65	1.5698	1.73	21.5019	21.7181	21.477	21.547
20204	1	0.53	5	1	58	64	1.6299	NA	21.83	NA	A	NA
20390	1	0.37	5	2	64	73	1.6499	1.8298	23.5078	21.7982	22.1013	21.5728
20398	1	0.52	5	2	60	77	1.6299	1.73	22.5827	25.7276	21.8203	22.7329

- Equivalent Mx Classic error - *Uh-oh... I'm having trouble reading a number in D or E format*

Important structural stuff

- openMx has a very fluid and flexible stucture
- Each code snippet is being saved as a variable
- We tend to reuse the variable names in our scripts
- This makes it very important to create a new project for each series of analyses
- Remember the project also contains the data so these files can become very large.

Matrices are the building blocks

```
mxMatrix( type="Lower", nrow=nv, ncol=nv, free=TRUE,
values=.6, label="a11", name="a" ), #X
```

- Many types eg. type="Lower"
- Denoted by names eg. name="a“
- Size eg. nrow=nv, ncol=nv
- All estimated parameters must be placed in a matrix & Mx must be told what type of matrix it is

Matrices are the building blocks

- Many types

```
mxMatrix( type="Zero", nrow=2,
ncol=3, name="a" )

0 0 0
0 0 0
```

```
mxMatrix( type="Unit", nrow=2,
ncol=3, name="a" )

1 1 1
1 1 1
```

```
mxMatrix( type="Ident", nrow=3,
ncol=3, name="a" )

1 0 0
0 1 0
0 0 1
```

```
mxMatrix( type="Diag", nrow=3,
ncol=3, free=TRUE, name="a" )

? 0 0
0 ? 0
0 0 ?
```

```
mxMatrix( type="Sdiag", nrow=3,
ncol=3, free=TRUE, name="a" )

0 0 0
? 0 0
? ? 0
```

```
mxMatrix( type="Stand", nrow=3,
ncol=3, free=TRUE, name="a" )

1 ? ?
? 1 ?
? ? 1
```

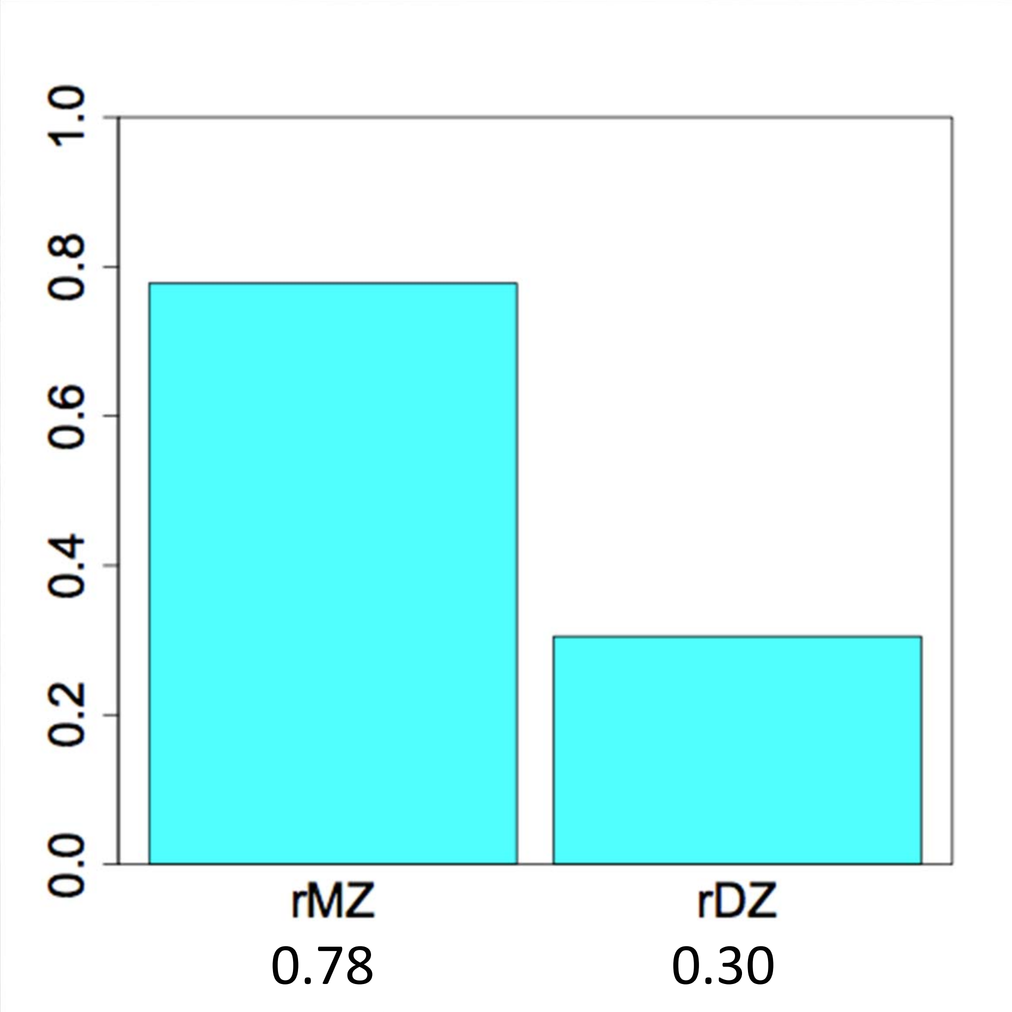
```
mxMatrix( type="Symm", nrow=3,
ncol=3, free=TRUE, name="a" )

? ? ?
? ? ?
? ? ?
```

```
mxMatrix( type="Lower", nrow=3,
ncol=3, free=TRUE, name="a" )

? 0 0
? ? 0
? ? ?
```

```
mxMatrix( type="Full", nrow=2,
ncol=4, free=TRUE, name="a" )

? ? ? ?
? ? ? ?
```

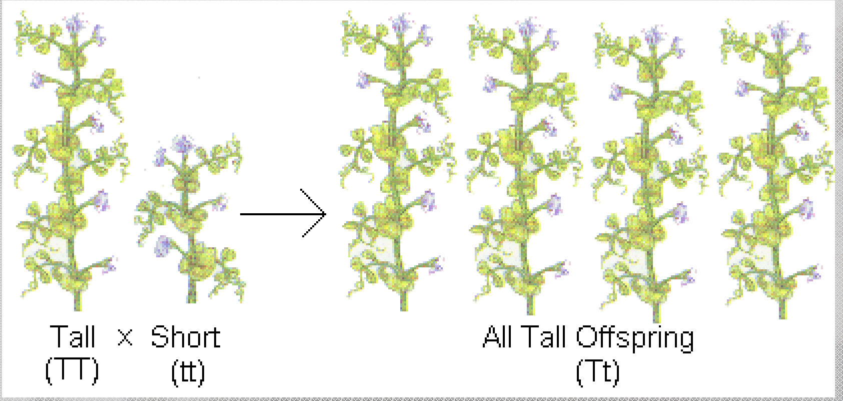
Yesterday we ran an ADE Model

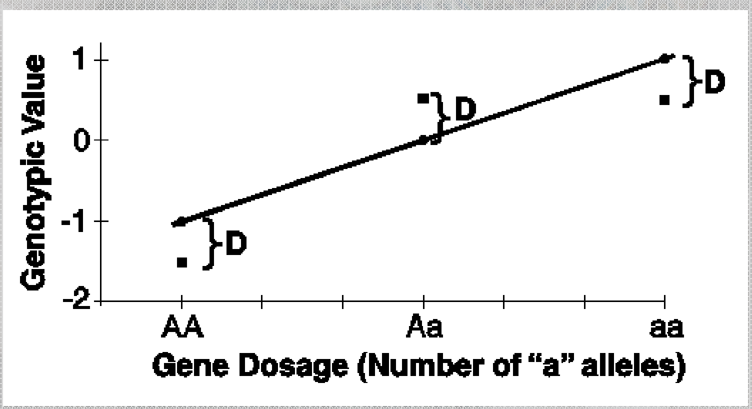
- Why?



What is D again?

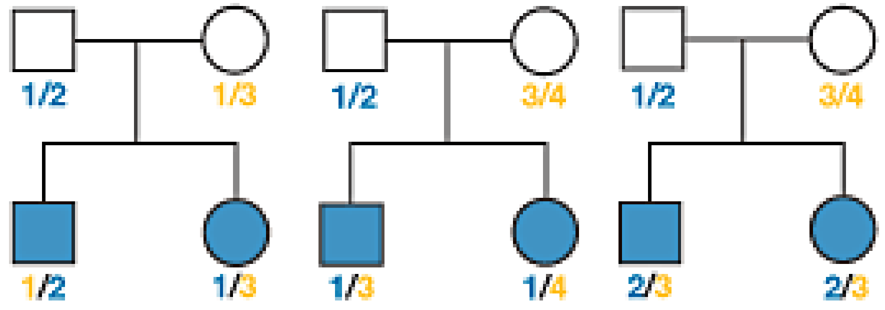
- Dominance refers to non-additive genetic effects resulting from interactions between alleles at the same locus or different loci (epistasis)





What is D again?

- DZ twins/full siblings share
  - ~50% of their segregating DNA &
  - for ~25% loci they share not only the genotype but also the parental origin of each allele



- DZ twins/full siblings share
  - ~50% of their segregating DNA &
  - for ~25% loci they share not only the genotype but also the parental origin of each allele

This is where the .5A comes from

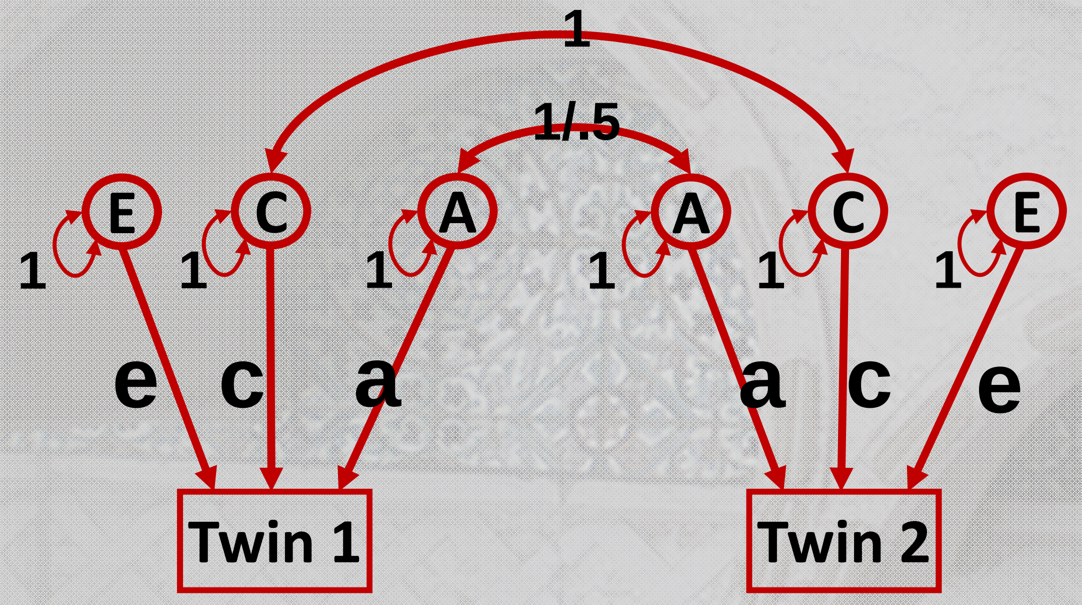
This is where the .25D comes from

Consider a mating between mother AB x father CD:

		Sib1			
		AC	AD	BC	BD
Sib 2	AC	2	1	1	0
	AD	1	2	0	1
	BC	1	0	2	1
	BD	0	1	1	2

IBD 0 : 1 : 2 = 25% : 50% : 25%

Today we will run an ACE model

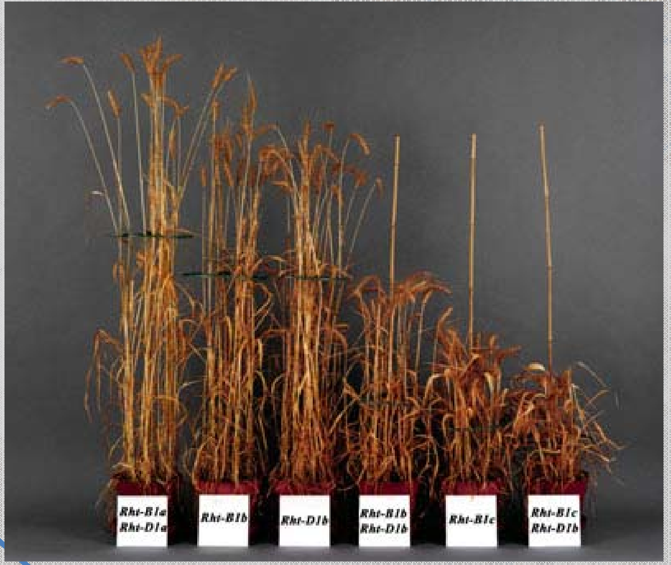


MZ		DZ	
$a^2+c^2+e^2$	a^2+c^2	$a^2+c^2+e^2$	$.5a^2+c^2$
a^2+c^2	$a^2+c^2+e^2$	$.5a^2+c^2$	$a^2+c^2+e^2$

Today we will run an ACE model

Additive genetic effects

- Why is the coefficient for DZ pairs .5?
- Average genetic sharing between siblings/DZ twins



		Sib1			
		AC	AD	BC	BD
Sib 2	AC	2	1	1	0
	AD	1	2	0	1
	BC	1	0	2	1
	BD	0	1	1	2

MZ

$$\begin{array}{cc} a^2+c^2+e^2 & a^2+c^2 \\ a^2+c^2 & a^2+c^2+e^2 \end{array}$$

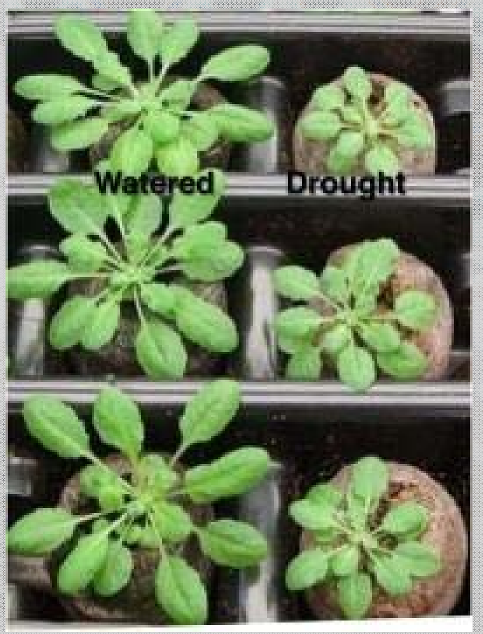
DZ

$$\begin{array}{cc} a^2+c^2+e^2 & .5a^2+c^2 \\ .5a^2+c^2 & a^2+c^2+e^2 \end{array}$$

Today we will run an ACE model

Common environmental effects

- Coefficient =1 for MZ and DZ pairs
- Equal environment assumption – for all the environmental influences THAT MATTER there is ON AVERAGE no differences in the degree of environmental sharing between MZ and DZ pairs

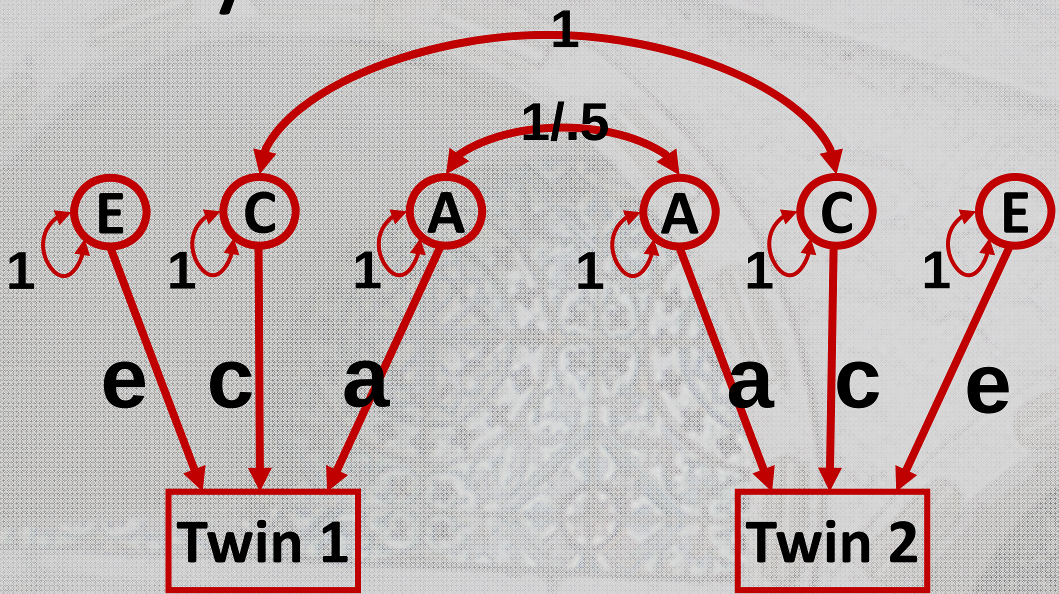


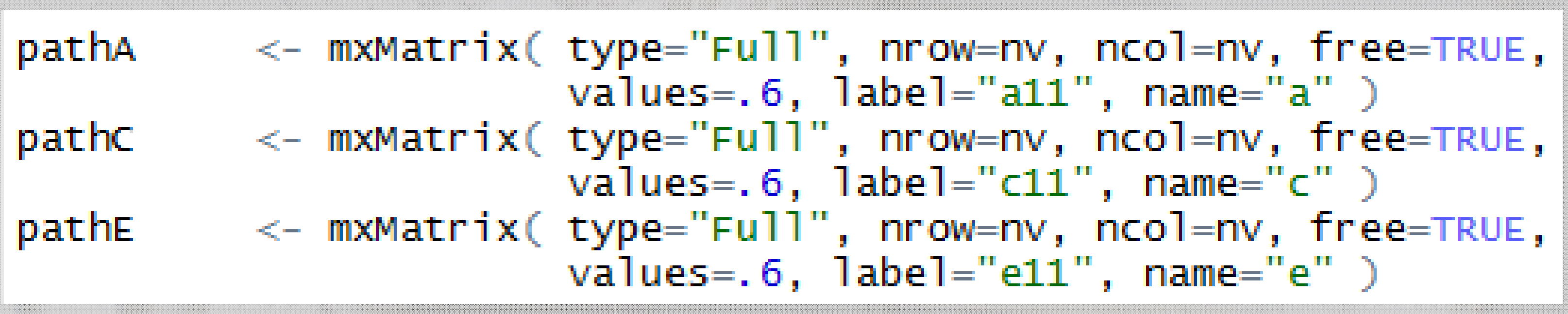
MZ		DZ	
$a^2+c^2+e^2$	a^2+c^2	$a^2+c^2+e^2$	$.5a^2+c^2$
a^2+c^2	$a^2+c^2+e^2$	$.5a^2+c^2$	$a^2+c^2+e^2$

Today we will run an ACE model

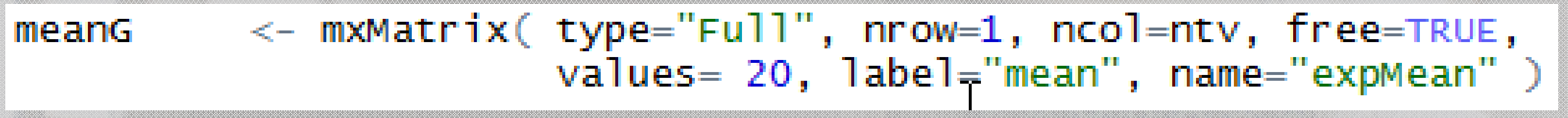
- Open RStudio
- faculty/sarah/tues_morning
- Copy everything

Today we will run an ACE model



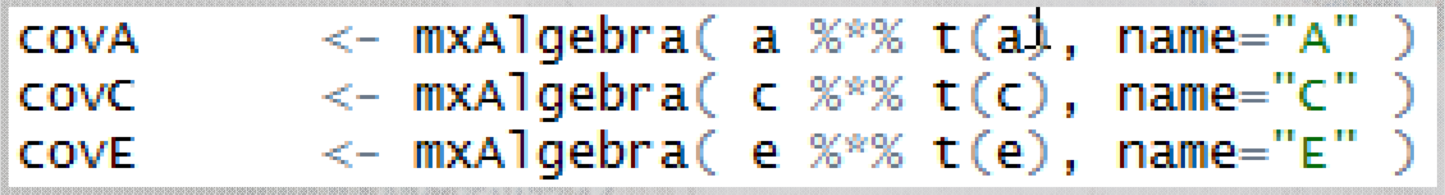


```
pathA      <- mxMatrix( type="Full", nrow=nv, ncol=nv, free=TRUE,
                        values=.6, label="a11", name="a" )
pathC      <- mxMatrix( type="Full", nrow=nv, ncol=nv, free=TRUE,
                        values=.6, label="c11", name="c" )
pathE      <- mxMatrix( type="Full", nrow=nv, ncol=nv, free=TRUE,
                        values=.6, label="e11", name="e" )
```

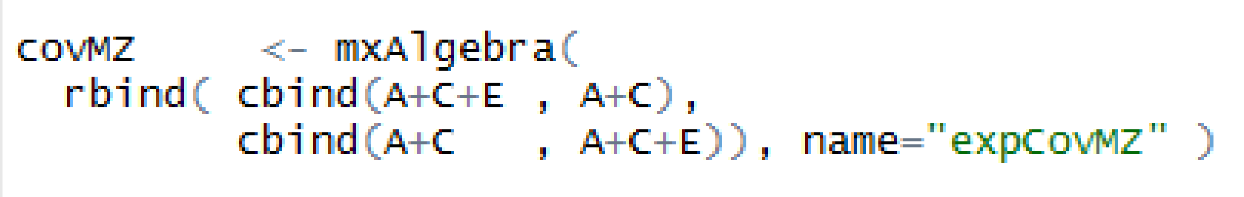


```
meanG      <- mxMatrix( type="Full", nrow=1, ncol=ntv, free=TRUE,
                        values= 20, label="mean", name="expMean" )
```

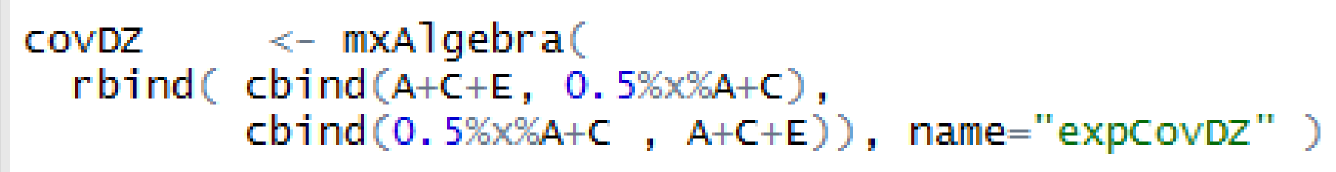
Today we will run an ACE model



```
covA        <- mxAlgebra( a %*% t(a), name="A" )
covC        <- mxAlgebra( c %*% t(c), name="C" )
covE        <- mxAlgebra( e %*% t(e), name="E" )
```

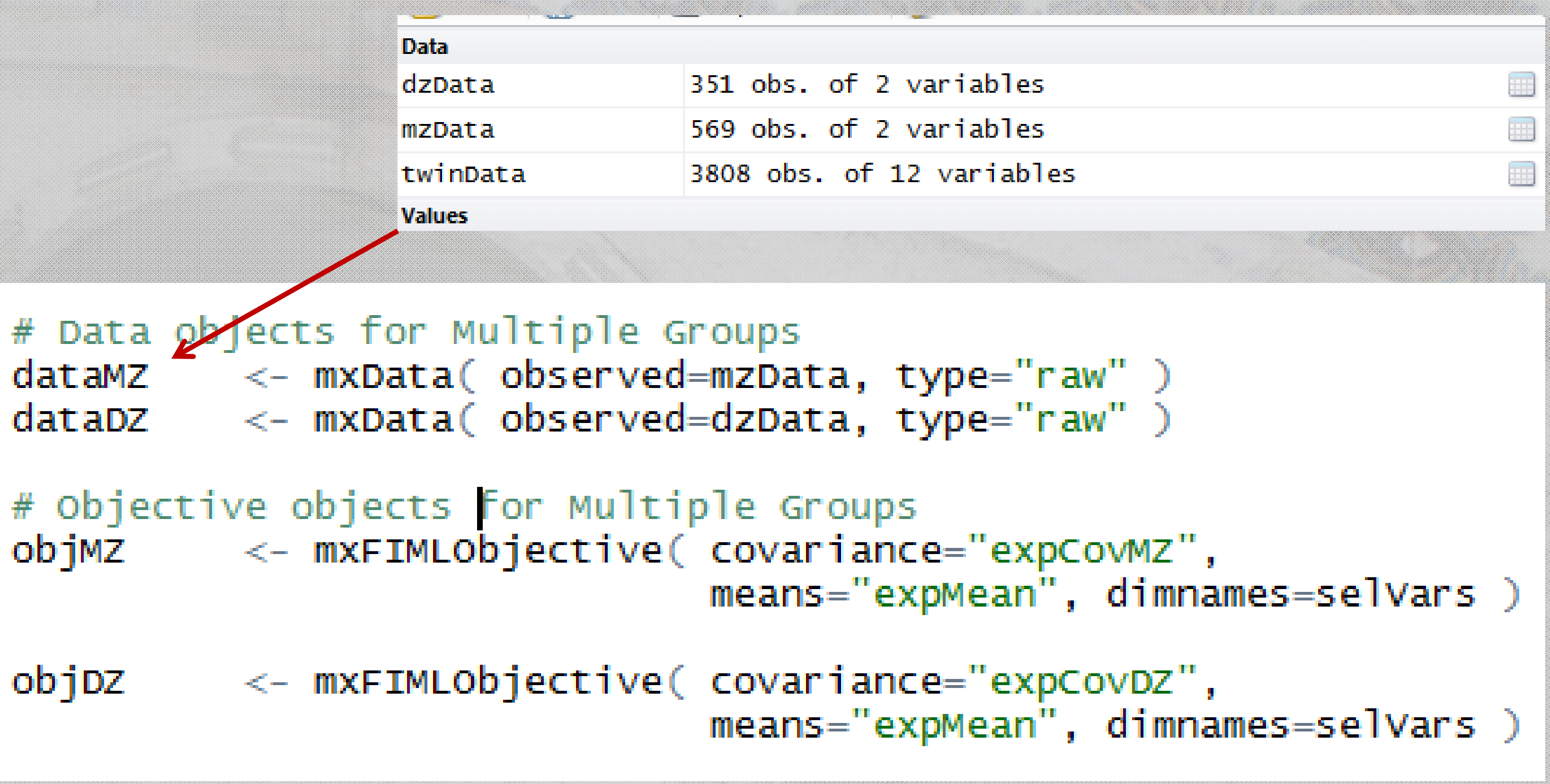


```
covMZ      <- mxAlgebra(
  rbind( cbind(A+C+E , A+C),
         cbind(A+C   , A+C+E)), name="expCovMZ" )
```



```
covDZ      <- mxAlgebra(
  rbind( cbind(A+C+E, 0.5%x%A+C),
         cbind(0.5%x%A+C , A+C+E)), name="expCovDZ" )
```

MZ		DZ	
$a^2+c^2+e^2$	a^2+c^2	$a^2+c^2+e^2$	$.5a^2+c^2$
a^2+c^2	$a^2+c^2+e^2$	$.5a^2+c^2$	$a^2+c^2+e^2$



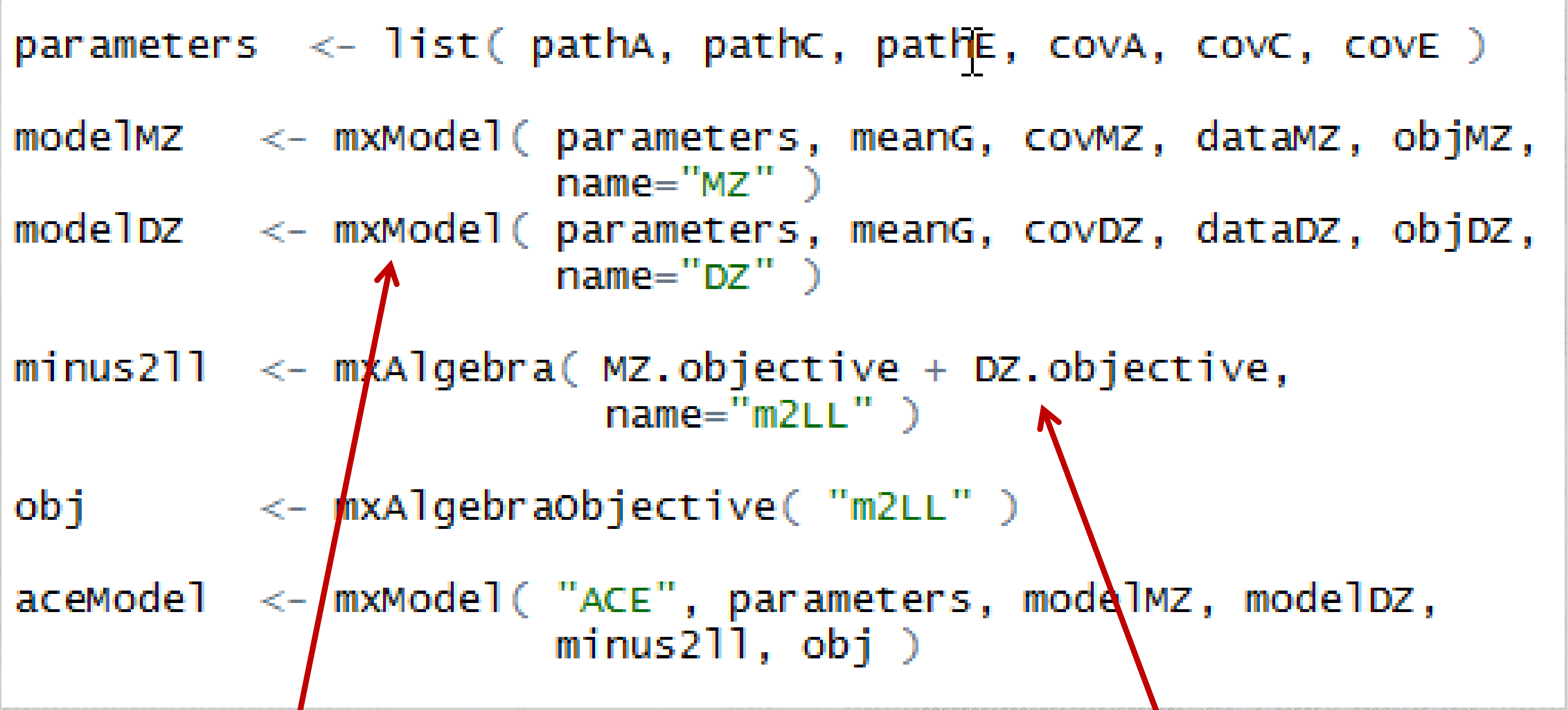
```
Data
dzData      351 obs. of 2 variables
mzData      569 obs. of 2 variables
twinData    3808 obs. of 12 variables
Values
```

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

# Objective objects for Multiple Groups
objMZ      <- mxFIMLObjective( covariance="expCovMZ",
                               means="expMean", dimnames=selVars )

objDZ      <- mxFIMLObjective( covariance="expCovDZ",
                               means="expMean", dimnames=selVars )
```

To fit a model to data, the differences between the observed covariance matrix and model-implied expected covariance matrix are minimized.

Objective functions are functions for which free parameter values are chosen such that the value of the objective function is minimized.

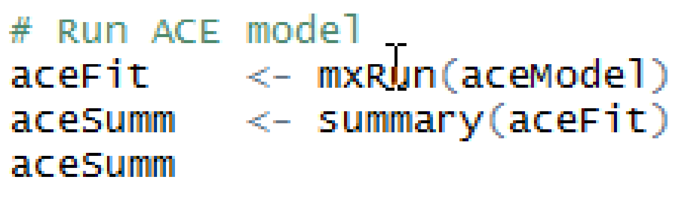
mxFIMLObjective() uses full-information maximum likelihood to provide maximum likelihood estimates of free parameters in the algebra defined by the covariance and means arguments.



```
parameters  <- list( pathA, pathC, pathE, covA, covC, covE )

modelMZ    <- mxModel( parameters, meanG, covMZ, dataMZ, objMZ,
                       name="MZ" )
modelDZ    <- mxModel( parameters, meanG, covDZ, dataDZ, objDZ,
                       name="DZ" )

minus2ll   <- mxAlgebra( MZ.objective + DZ.objective,
                         name="m2LL" )

obj        <- mxAlgebraObjective( "m2LL" )

aceModel   <- mxModel( "ACE", parameters, modelMZ, modelDZ,
                       minus2ll, obj )
```

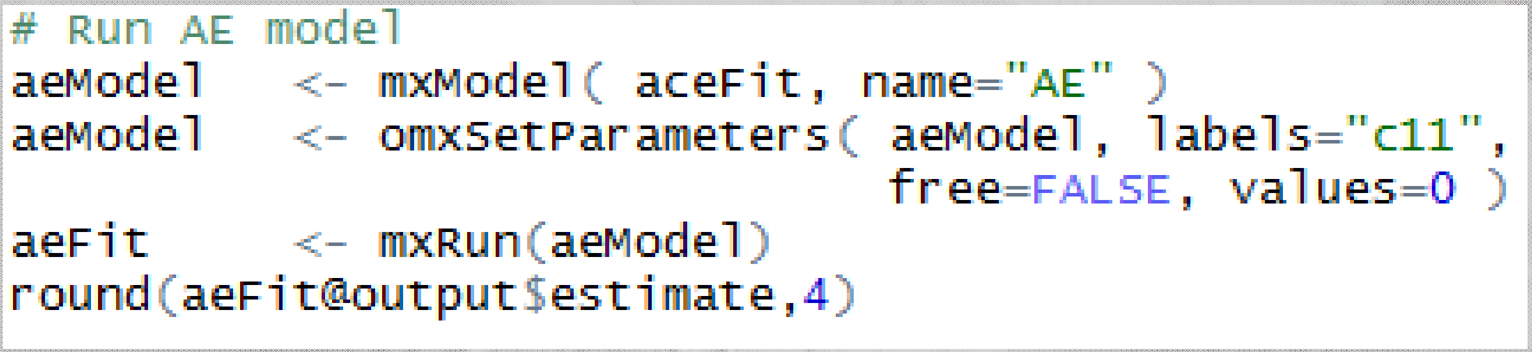
This models requires path parameters, means, covariance, data and objectives

Automatic naming – you don’t need to predefine this



```
# Run ACE model
aceFit      <- mxRun(aceModel)
aceSumm     <- summary(aceFit)
aceSumm
```

Submodels



```
# Run AE model
aeModel   <- mxModel( aceFit, name="AE" )
aeModel   <- omxSetParameters( aeModel, labels="c11",
                               free=FALSE, values=0 )
aeFit     <- mxRun(aeModel)
round(aeFit@output$estimate,4)
```

Pickup the previously prepared model
Edit as required
Rerun and compare

Saving your output

- Save the R workspace
  - On closing click yes
  - Very big
  - Saves everything
- Save the fitted model
  - Equivalent to save in classic Mx
  - save(univACEFit, file="test.omxs")
  - load("test.omxs") – need to load OpenMx first

What to report

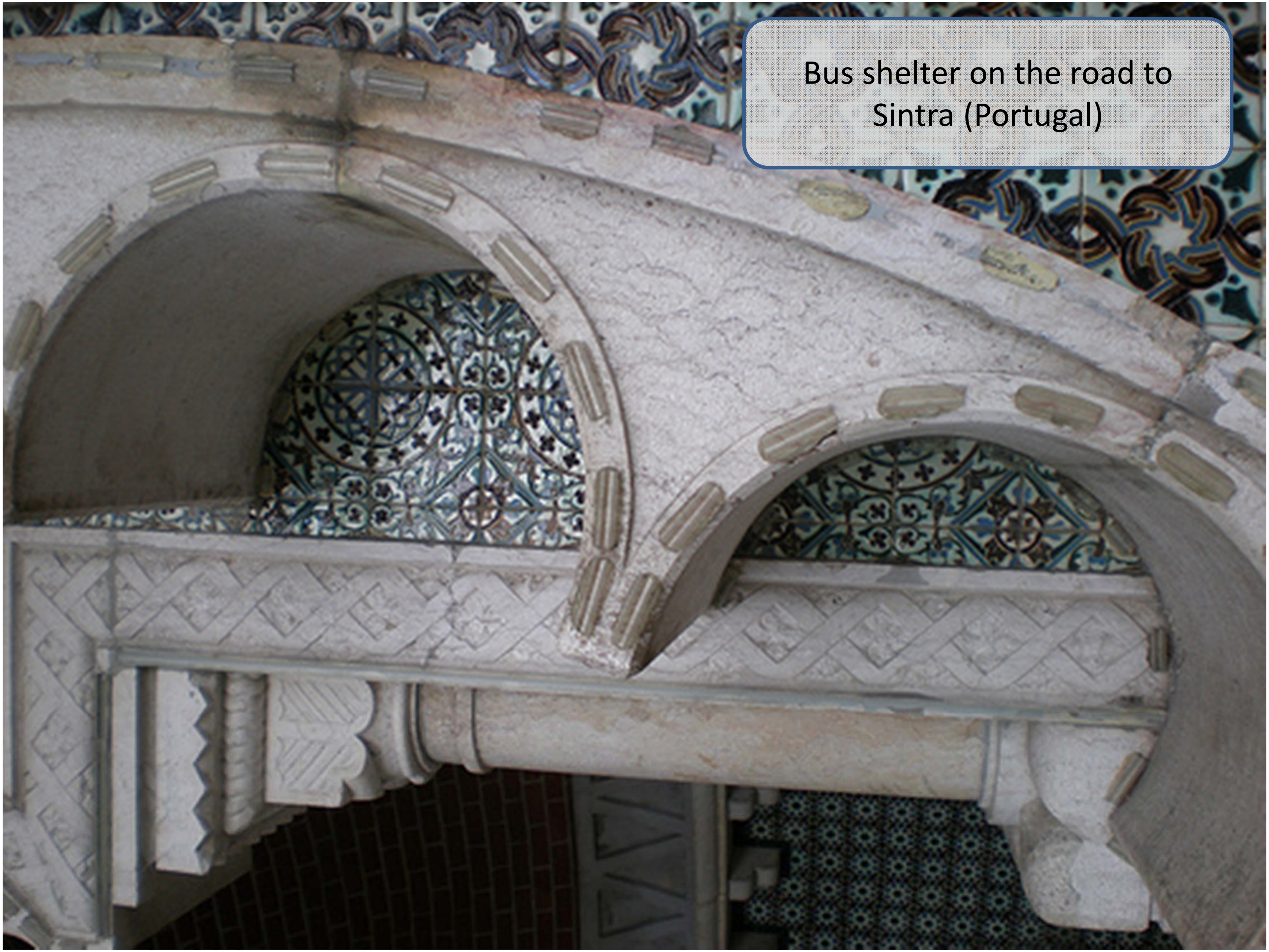
- Summary statistics
  - Usually from a simplified ‘saturated’ model
- Standardized estimates
  - Easier to conceptualise
    - ie 40% of the phenotypic variance vs a genetic effect of 2.84
    - Can easily be returned to original scale if summary statistics are provided

What to report

- Path coefficients
  - Very important in multivariate analyses
    - Gives a much clearer picture of the directionality of effects
- Variance components/proportion of variance explained
- Genetic correlations

General Advice/Problem solving

- Scripting styles differ
- Check the sample description
- Learn to love the webpage
- Comments are your friends



Bus shelter on the road to Sintra (Portugal)