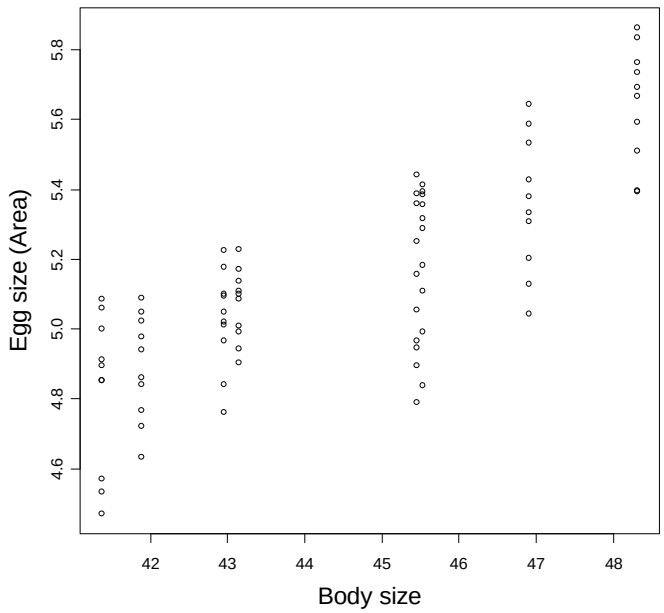
Latent random variables

Imagine that you have collected egg size data on a fish called *Austrolebias elongatus*, and the graph of egg size on body size of the mother looks as follows:



What would happen if you hadn't measured the body size of the mother? What if the covariate would be simply impossible to measure?
It is not always possible to control covariates experimentally, or to measure all relevant explanatory variables explicitly. Even when you know that some effects must exist, you often cannot measure them directly. The relevant explanation for a phenomenon then remains *latent* and difficult to access.
In biology, common latent variables are the genotypic effects that individuals carry with them. You cannot see them, you know they are there and in most cases you cannot measure their effects directly. A workable approach to take this particular problem into account already exists for more than 80 years. It is called quantitative genetics. In these exercises, we will investigate some quantitative genetic models, and analyse them using an approach which has much broader applicability.
The general idea of that approach is that such latent variables are random draws from an a priori assumed probability distribution, of which we only want to estimate the parameter(s).

This afternoon, we go undercover and investigate latent phenomena.
A helper file with R commands is available on blackboard. It is called "latent2007.txt".

♣ The Infinitesimal Model

Quantitative genetics investigates patterns of genetic and phenotypic variation. The basic idea is that quantitative phenotypes are composed from genetic and environmental contributions.

phenotype = population mean + random genotype + random environment

We don't want to (or we can't) know the genotypic and environmental contributions exactly, we're primarily interested in the relative magnitudes of genetic and phenotypic variance components.
In the so-called infinitesimal model, the assumption is that many genes with independent small effects contribute to phenotype, such that the overall genotypic contribution is a normally distributed random variable.

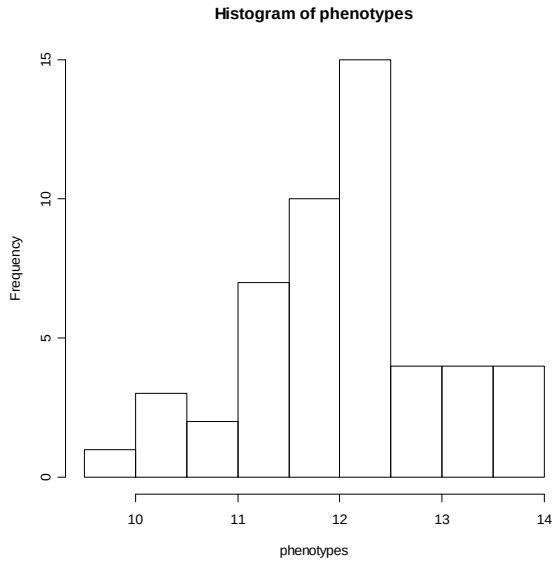
We will now simulate data for one of the simplest breeding experiments possible: several randomly sampled females are mated to a single male, and a single phenotypic trait is measured in the offspring of all females.
The example assumes 10 females, with 5 offspring each. We want to know the magnitude of the genetic variance between females. In the simulation, we assume that the genetic variance is 0.64, the environmental variance 0.25, that the overall mean trait value is 12.

```
genotypeslist<-rnorm(10,0,0.8)
genotypes <- gl(10,5,50)
genotypicvalues<-numeric(50)
for(i in 1:50)genotypicvalues[i]<-genotypeslist[floor(1+(i-1)/5)]
environments <-rnorm(50,0,0.5)
mean<-12

phenotypes<-mean + genotypicvalues + environments
```

This is what your data can look like

```
hist(phenotypes)
```



The analysis of such data usually occurs by means of mixed models, for which the "lme4" library is one of the tools which are availabe in R. You might have to install that library yourself.

```
library(lme4)
```

This code fits a linear mixed model, with a fixed effect intercept and random variation between female genotypes. Note that a random effect in a model is specified between "(" and ")". The random grouping variable, which is called *genotypes* in our data, is given behind the "|" sign. What could the "1" in the random effect specification mean?

```
mm1<-lmer(phenotypes~1+(1|genotypes))
```

The output of lmer() is the following:

```
mm1
Linear mixed-effects model fit by REML
Formula: phenotypes ~ 1 + (1 | genotypes)
   AIC  BIC    logLik MLdeviance  REMLdeviance
 103.1  107.0  -49.57  97.76          99.14
Random effects:
 Groups        Name        Variance   Std.Dev.
 genotypes     (Intercept) 0.34892    0.59069
 Residual                  0.28508    0.53393
number of obs: 50, groups: genotypes, 10

Fixed effects:
             Estimate    Std. Error    t value
(Intercept)  11.6370      0.2015       57.76
```

Does the software estimate the genotypic variance in your data accurately?
We can also inspect the random genotypic effects as they are predicted by the model. The assumption is that they are normally distributed, but is that also what the model will predict?

```
ranef(mm1)
hist(ranef(mm1)[[1]][,1])
```

Mind the [[1]][,1] here, it is necessary to get the values of the random effects out.
A normal probability plot looks like this:

```
qqnorm(ranef(mm1)[[1]][,1])
qqline(ranef(mm1)[[1]][,1])
```

One can wonder whether it is really critical for the analysis that genotypic values are from a normal distribution? Imagine that the genotypic value is determined by a major gene with two alleles and dominance-recessivity. Each female then has one of two genotypic values. Let's simulate that:

```
genotypeslist2<-c(rep(-1,5),rep(1,5))
```

Here females either have a genotypic value of -1 or +1.

```
genotypicvalues2<-numeric(50)
for(i in 1:50)genotypicvalues2[i]<-genotypeslist2[floor(1+(i-1)/5)]
phenotypes2<-mean + genotypicvalues2 + environments
hist(phenotypes2)
mm2<-lmer(phenotypes2~1+(1|genotypes))
mm2
hist(ranef(mm2)[[1]][,1])
qqnorm(ranef(mm2)[[1]][,1])
```

Well, what do the random effects look like? Are they normally distributed?
Please change the parameters of the sampling design and report whether the random effects become more normally distributed or not.

♣ Liabilities and Threshold Traits

We will now investigate another type of quantitative genetic model, but we keep the simple *many females – one male* breeding design for our simulations.
The prevalence of many diseases also have a genetic component which contributes to their appearance.

Whether the disease will appear in an individual or not, is a Bernoulli zero-one trait, not a normally distributed quantitative trait.
The idea on which the analysis of zero-one traits is based is that there is an underlying quantitative trait which is called the "liability", which is then mapped to a zero-one trait by a specific and fixed non-linear mapping.
This liability technique is used to model genetic variation for any kind of discrete trait, such as sex ratios in offspring for example. The following simulation could be on sex ratio data.

The design assumes 10 clutches per female of ten offspring individuals each.

```
genotypelist<-rnorm(10,0,2)
genotypes <- gl(10,10,100)
genotypicvalues<-numeric(100)
for(i in 1:100)genotypicvalues[i]<-genotypelist[floor(1+(i-1)/10)]
environments <-rnorm(100,0,0.02)
mean<-0
```

This is the value of the liability

```
liabilities<-mean + genotypicvalues + environments
```

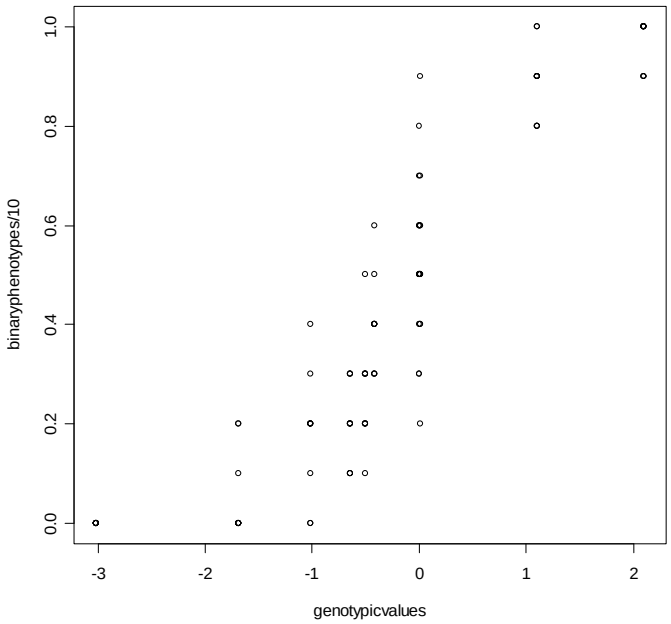
Usually the probit function is used to map a liability to a probability between zero and one. It gives the probability that an offspring will have a 'positive' response. From that probability one can then simulate responses for each offspring.

```
invprobit<-function(l)pnorm(l)
thresholdtraitp<-invprobit(liabilities)
```

Here we draw the numbers of positive responses for all the clutches of offspring.

```
binaryphenotypes<-rbinom(n=100,size=10,prob=thresholdtraitp)

plot(binaryphenotypes/10~genotypicvalues)
```



Data that need to be analysed using binomial distributions, need to be prepared for analysis in the following way:

```
binaryphenotypes<-cbind(binaryphenotypes,10-binaryphenotypes)
head(binaryphenotypes)
```

Per sample, the positive and negative responses have to be given.
We can analyse the numbers of positive responses either ignoring the random variation between females, and using a generalized linear model, or taking the variation between females into account. Then one has to fit a generalized linear mixed model.

```
glm1<-glm(binaryphenotypes~1,family=binomial(link=probit))
lmer(binaryphenotypes~1+(1|genotypes),family=binomial(link=probit))
```

A phenomenon which often occurs in generalized linear models such as glm1 is overdispersion. Look up in your book what that term means. A common estimate of overdispersion is a so-called "scale parameter" such as

```
glm1$deviance/glm1$df.residual
```

If there is overdispersion, then the scale will be larger than one. Note that in the output from the mixed model, the value of the scale is smaller than the estimate based on glm1. Generally, random effects which are not taken into account in generalized linear models are a common cause of overdispersion [Needless to say that all statistical tests are flawed when overdispersion is not taken into account].

♣ Multiple Traits per Individual

One can also measure several quantitative traits on an individual. Then one can draw random numbers from multivariate normal distributions to simulate such measurements For the case of two traits, these are the variance-covariance matrices of the genotypic values

```
G <- matrix(c(10,3,3,2),2,2)
```

and the environmental component E

```
E <-matrix(c(5,0,0,6),2,2)
```

Load the following library, it contains a function which can drawn multivariate normal random numbers.

```
library(MASS)
```

We simulate a breeding experiment as before:

```
mgenotypelist<-mvrnorm(n=10, rep(0, 2), G)
mgenotypelist
mgenotypes <- gl(10,5,50)
mgenotypicvalues1<-numeric(50)
mgenotypicvalues2<-numeric(50)
for(i in 1:50){
mgenotypicvalues1[i]<-mgenotypelist[floor(1+(i-1)/5),1]
mgenotypicvalues2[i]<-mgenotypelist[floor(1+(i-1)/5),2]
}

menvironments <-mvrnorm(n=50, rep(0, 2), E)
menvironments1<-menvironments[,1]
menvironments2<-menvironments[,2]

mmean1<-12
mmean2<-4

mphenotypes1<-mmean1 + mgenotypicvalues1 + menvironments1
mphenotypes2<-mmean2 + mgenotypicvalues2 + menvironments2
```

Inspect the data a bit:

```
plot(mphenotypes1,mphenotypes2)
plot(mgenotypicvalues1,mgenotypicvalues2)
```

```
cor.test(mphenotypes1,mphenotypes2)
cor.test(mgenotypicvalues1,mgenotypicvalues2)
```

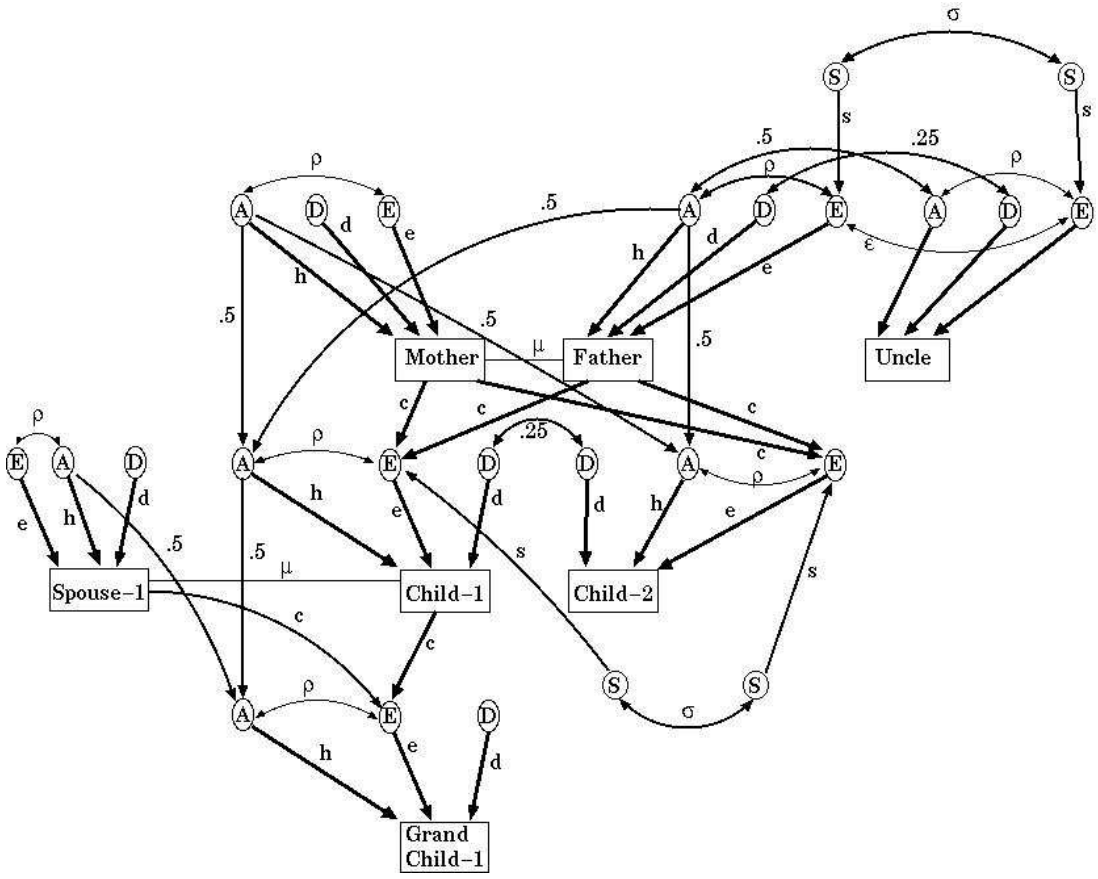
There is specific software on the market for the analysis of multivariate mixed models, but for relatively small datasets, we can coerce the data into a particular univariate specification and analyse it like that. The idea is that we introduce a new grouping variable "mtrait" which can take two values, one or two, depending on whether a measurement is on trait one or two.

```
mphenotypes<-c(mphenotypes1,mphenotypes2)
mgeno<-rep(mgenotypes,2)
mtrait<-gl(2,50,100)
```

And fit a mixed model

```
lmer(mphenotypes~mtrait+(mtrait-1|mgeno))
```

Often many correlated and uncorrelated random variables contribute to variation in a specific trait. There are specific types of analysis to investigate such networks of dependency, and the result is called a path diagram. The one below describes heritable and non-heritable contributions to individual variation in systolic blood pressure. It is from Tambs et al. [Genetic Epidemiology 9:11 – 26, 1992]. If you google for example "path analysis" and "ecology", you will find many more examples.



Quite forbidding, no? This session has presented you all the necessary tools to simulate such an entire network bit by bit, if necessary.

Tom Van Dooren 09/2007
t.j.m.van.dooren@biology.leidenuniv.nl