Lecture 1 - fixed vs random effects models

Dr Richard Wilkinson

8 February 2016

```
library(magic)
```

```
## Loading required package: abind
```

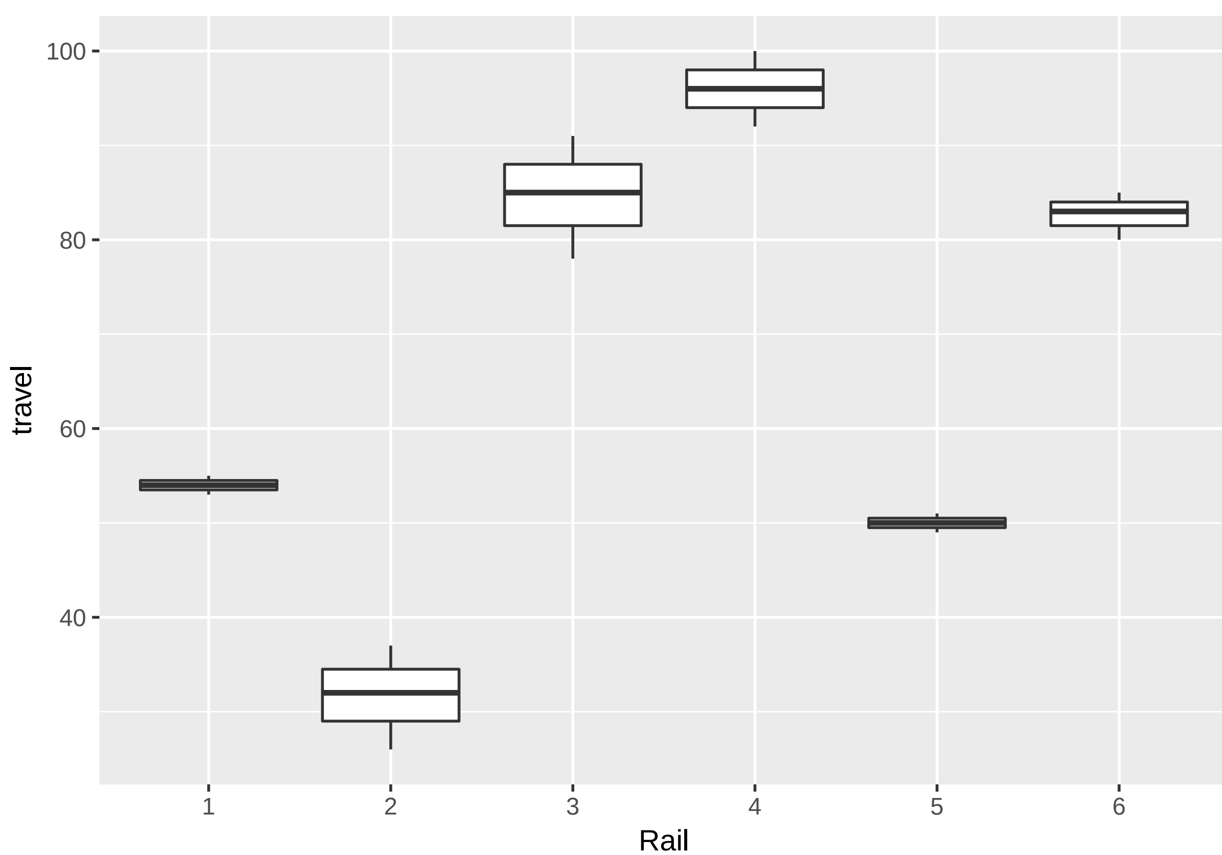
```
library(mvtnorm)
library(lme4)
```

```
## Loading required package: Matrix
```

Rail data example. Model is

$$y_{ij} = mu + b_i + epsilon_{ij}$$

```
load('MAS473.RData')
# b_i ~ N(0,psisq)
# epsilon_{ij} ~ N(0,sigmasq)
attach(raildata)
library(ggplot2)
qplot(Rail, travel, geom='boxplot')
```



Fit model using ordinary maximum likelihood

```
(fm1.ml<-lmer(travel~1+(1|Rail),raildata,REML=F))
```

```
## Linear mixed model fit by maximum likelihood  ['lmerMod']
## Formula: travel ~ 1 + (1 | Rail)
##    Data: raildata
##      AIC      BIC   logLik deviance df.resid
## 134.5600 137.2312 -64.2800 128.5600       15
## Random effects:
##  Groups   Name        Std.Dev.
##  Rail     (Intercept) 22.624
##  Residual              4.021
## Number of obs: 18, groups:  Rail, 6
## Fixed Effects:
## (Intercept)
##        66.5
```

Now try to get the same parameter estimates by numerically maximising the log likelihood

Define a function to calculate (minus) the log likelihood

```
minus.log.ordinary.likelihood.raildata<-function(theta,y){
  # theta[1] = log sigmasq
  # theta[2] = log psisq

  sigmasq<-exp(theta[1])
  psisq<-exp(theta[2]) # (force sigmasq and psisq to be positive)

  V1<-matrix(psisq,3,3)
  diag(V1)<-diag(V1)+sigmasq
  V<-adiag(V1,V1,V1,V1,V1,V1) # Variance covariance matrix of data
  X<- model.matrix(travel~1,raildata)

  -dmvnorm(t(y),X*theta[3],V,log=T) # -log likelihood
}
```

minimise - log likelihood. Compare estimates with fm1.ml

```
y<-matrix(travel,18,1)
theta<-c(log(10),log(500),50)

theta.mle<-optim(c(log(10),log(500),50), minus.log.ordinary.likelihood.raildata,y=matrix(travel,18,1))

exp(theta.mle$par[1:2]) # estimates of sigmasq and psisq
```

```
## [1]  16.16365 511.83309
```

```
theta.mle$par[3] # estimate of mu
```

```
## [1] 66.49475
```

```
summary(fm1.ml)
```

```
## Linear mixed model fit by maximum likelihood  ['lmerMod']
## Formula: travel ~ 1 + (1 | Rail)
##    Data: raildata
##
##      AIC      BIC   logLik deviance df.resid
##    134.6    137.2    -64.3    128.6       15
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -1.61098 -0.28887  0.03454  0.21373  1.62222
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  Rail     (Intercept) 511.86   22.624
##  Residual              16.17    4.021
## Number of obs: 18, groups:  Rail, 6
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)   66.500      9.285   7.162
```

likely to be small discrepancies due to optimisation routine. Can try different starting values to get global maximum

Fit model using REML

```
(fm1.reml<-lmer(travel~1+(1|Rail),raildata))
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: travel ~ 1 + (1 | Rail)
##    Data: raildata
## REML criterion at convergence: 122.177
## Random effects:
##  Groups   Name        Std.Dev.
##  Rail     (Intercept) 24.805
##  Residual              4.021
## Number of obs: 18, groups:  Rail, 6
## Fixed Effects:
## (Intercept)
##        66.5
```

```
minus.log.reml.likelihood.raildata<-function(theta,y){
  sigmasq<-exp(theta[1])
  psisq<-exp(theta[2])
  V1<-matrix(psisq,3,3)
  diag(V1)<-diag(V1)+sigmasq
  V<-adiag(V1,V1,V1,V1,V1,V1)
  X<- model.matrix(travel~1,raildata)
  Vinv<-solve(V)
  betahat<-solve(t(X)%*% Vinv %*% X)%*% t(X) %*% Vinv %*% y
 -( -0.5*log(det(V)) - 0.5*log(det(t(X)%*%Vinv%*%X)) - 0.5*(18-1)*log(2*pi) - 0.5*t(y-X%*%betahat)%*%Vi
```

```
}
```

Example to show construction of V, betahat and log REML criterion

```
sigmasq<-10
psisq<-100
V1<-matrix(psisq,3,3)
diag(V1)<-diag(V1)+sigmasq
(V<-adiag(V1,V1,V1,V1,V1,V1))
```

```
##       [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10] [,11] [,12] [,13]
##  [1,]  110  100  100    0    0    0    0    0    0     0     0     0     0
##  [2,]  100  110  100    0    0    0    0    0    0     0     0     0     0
##  [3,]  100  100  110    0    0    0    0    0    0     0     0     0     0
##  [4,]    0    0    0  110  100  100    0    0    0     0     0     0     0
##  [5,]    0    0    0  100  110  100    0    0    0     0     0     0     0
##  [6,]    0    0    0  100  100  110    0    0    0     0     0     0     0
##  [7,]    0    0    0    0    0    0  110  100  100     0     0     0     0
##  [8,]    0    0    0    0    0    0  100  110  100     0     0     0     0
##  [9,]    0    0    0    0    0    0  100  100  110     0     0     0     0
## [10,]    0    0    0    0    0    0    0    0    0   110   100   100     0
## [11,]    0    0    0    0    0    0    0    0    0   100   110   100     0
## [12,]    0    0    0    0    0    0    0    0    0   100   100   110     0
## [13,]    0    0    0    0    0    0    0    0    0     0     0     0   110
## [14,]    0    0    0    0    0    0    0    0    0     0     0     0   100
## [15,]    0    0    0    0    0    0    0    0    0     0     0     0   100
## [16,]    0    0    0    0    0    0    0    0    0     0     0     0     0
## [17,]    0    0    0    0    0    0    0    0    0     0     0     0     0
## [18,]    0    0    0    0    0    0    0    0    0     0     0     0     0
##       [,14] [,15] [,16] [,17] [,18]
##  [1,]     0     0     0     0     0
##  [2,]     0     0     0     0     0
##  [3,]     0     0     0     0     0
##  [4,]     0     0     0     0     0
##  [5,]     0     0     0     0     0
##  [6,]     0     0     0     0     0
##  [7,]     0     0     0     0     0
##  [8,]     0     0     0     0     0
##  [9,]     0     0     0     0     0
## [10,]     0     0     0     0     0
## [11,]     0     0     0     0     0
## [12,]     0     0     0     0     0
## [13,]   100   100     0     0     0
## [14,]   110   100     0     0     0
## [15,]   100   110     0     0     0
## [16,]     0     0   110   100   100
## [17,]     0     0   100   110   100
## [18,]     0     0   100   100   110
```

```
X<- model.matrix(travel~1,raildata)
y<-matrix(travel,18,1)
Vinv<-solve(V)
(betahat<-solve(t(X)%*% Vinv %*% X)%*% t(X) %*% Vinv %*% y)
```

```
##             [,1]
```

```
## (Intercept) 66.5
```

```r
-0.5*log(det(V)) - 0.5*log(det(t(X)%*%Vinv%*%X)) - 0.5*(18-1)*log(2*pi) - 0.5*t(y-X%*%betahat)%*%Vinv%
```

```
##           [,1]
## [1,] -69.94102
```

Optimise log REML likelihood

```r
theta.mle.reml<-optim(log(c(10,500)), minus.log.reml.likelihood.raildata,y=matrix(travel,18,1))
exp(theta.mle.reml$par[1:2]) # estimates of sigmasq and psisq
```

```
## [1]  16.16761 615.48251
```

```r
2*theta.mle.reml$value
```

```
## [1] 122.177
```