NARTI and Q-STEP Centre (University of Leeds): Hierarchical linear modeling in R: Random coefficients

In this session we are going to look at the rationale and computational features of random coefficient models. We shall begin with fitting a simple random interecept model and continue with more sophisticated random slopes (mixed effects) models.

The following packages will be required to execute the script. If a certain package is not intsalled use command install.packages(”) to complete the installation

```
library(lme4)
```

```
## Warning: package 'lme4' was built under R version 3.2.5

## Loading required package: Matrix

## Warning: package 'Matrix' was built under R version 3.2.5
```

```
library(psych)
```

```
## Warning: package 'psych' was built under R version 3.2.5
```

```
library(lattice)
library(sjPlot)
```

```
## Warning: package 'sjPlot' was built under R version 3.2.5

## Visit http://strengejacke.de/sjPlot for package-vignettes.
```

```
library(sjmisc)
```

```
## Warning: package 'sjmisc' was built under R version 3.2.5
```

```
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 3.2.5

##
## Attaching package: 'ggplot2'

## The following objects are masked from 'package:psych':
##
##     %+%, alpha
```

```
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 3.2.5

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

First things first, we need to upload the data. The lines below read the dataset into R, print out the names of all variables in the sample and provide very basic descriptive statistics

```
dataAbortion=read.csv('AbortionAttitudes.csv')
names(dataAbortion)
```

```
## [1] "ID"         "Respondent" "Year"       "Attitudes"  "Politics"
## [6] "Class"      "Gender"     "Age"        "Religion"
```

```
describe(dataAbortion)
```

```
##            vars    n    mean     sd median trimmed     mad min  max range
## ID            1 1055   28.03  15.09     30   28.11   19.27   1   54    53
## Respondent    2 1055 1719.40 989.75   1754 1722.59 1168.29  39 3388  3349
## Year          3 1055    2.50   1.12      3    2.50    1.48   1    4     3
## Attitudes     4 1055    4.93   1.84      5    5.04    2.97   0    7     7
## Politics      5 1055    1.97   0.97      2    1.86    1.48   1    5     4
## Class         6 1055    2.25   0.83      2    2.31    1.48   1    3     2
## Gender        7 1055    1.56   0.50      2    1.57    0.00   1    2     1
## Age           8 1055   43.49  15.66     40   42.77   17.79  18   80    62
## Religion      9 1055    2.78   0.99      3    2.81    1.48   1    4     3
##             skew kurtosis    se
## ID         -0.08    -1.20  0.46
## Respondent -0.04    -1.20 30.47
## Year        0.00    -1.36  0.03
## Attitudes  -0.30    -1.15  0.06
## Politics    0.88     0.47  0.03
## Class      -0.48    -1.38  0.03
## Gender     -0.23    -1.95  0.02
## Age         0.35    -0.84  0.48
## Religion    0.04    -1.36  0.03
```

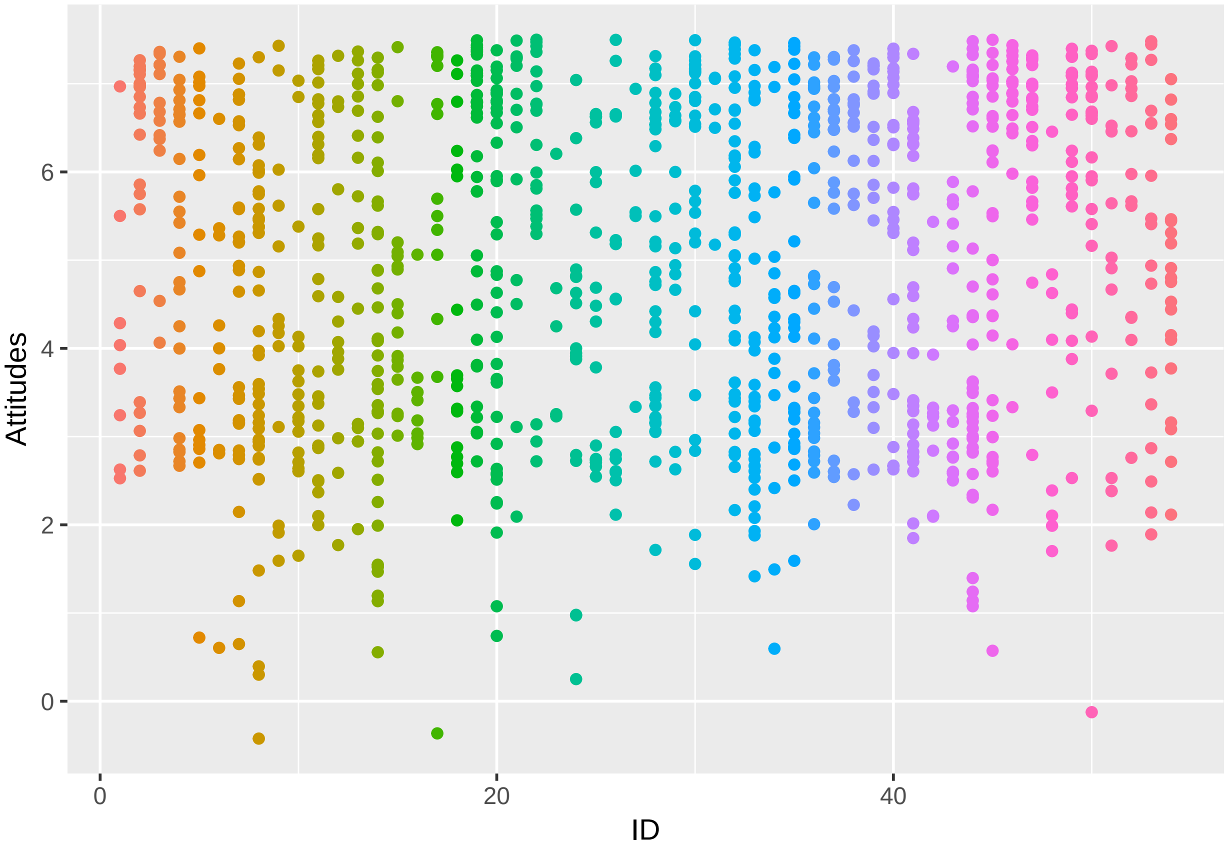
We are interested in the attitudes towards abortion, a variable measured 7-point Likert-type scale. The data were derived from the British Social Attitudes Survey, a nationally representative study wherein respondents are nested on administrative units (districts).

Let us begin with visualising group differences in relation to the attitudes variable. The command below belongs to package 'dplyr'. It creates a dataset with aggregated values of attitudes towards abortion.

```
group2 <- dataAbortion %>%
  group_by(ID) %>%
  summarise(Attitudes = mean(Attitudes))
```

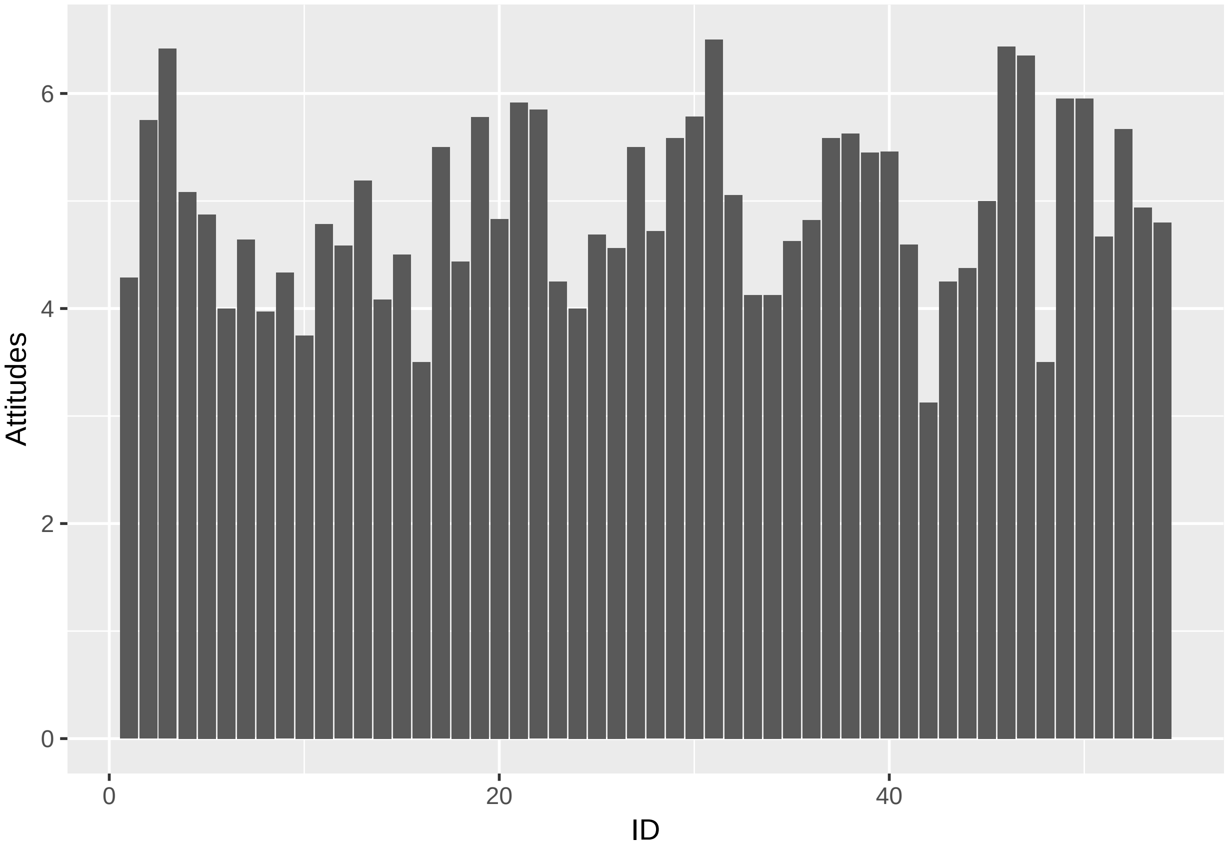
Having created the aggregated dataset, we can plot responses by groups, using different colours to signify the group effect.

```
ggplot(dataAbortion, aes(x=ID,y=Attitudes,colour = factor(ID))) +
  geom_point(position=position_jitter(w=0.00, h=0.5)) + guides(colour=FALSE) +
  geom_point(data = group2)
```



Another way of tackling the same problem is to produce a bar chart in line with the syntax below.

```
ggplot(group2, aes(x = ID, y = Attitudes)) +
  geom_bar(stat = "identity")
```



It appears that we have enough reason to fit a random intercept model. lme4 package syntax for random intercept models is as follows.

```
AbortionIntercept=lmer(Attitudes~(1|ID),REML=TRUE,
                         data=dataAbortion)
summary(AbortionIntercept)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: Attitudes ~ (1 | ID)
##    Data: dataAbortion
##
## REML criterion at convergence: 4227.8
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.2575 -0.9203  0.1371  0.8332  1.6449
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  ID       (Intercept) 0.3876   0.6226
##  Residual             3.0249   1.7392
## Number of obs: 1055, groups:  ID, 54
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)   4.9368     0.1031   47.89
```

ICC1, the variance partitioning coefficient, can be easliy derived from the output

```r
0.3876/(0.3876+3.0249)
```

```
## [1] 0.1135824
```

Roughly 11.3 per cent of the total variation in attitudes towards abortion is due to level two (i.e. variation between the districts). This is good enough to justify the use of multilevel modeling.

Using deviance statistics we can ensure that the random intercept model is superior to the OLS linear regression.

```r
AbortionLinear=lm(Attitudes~1,data=dataAbortion)
summary(AbortionLinear)
```

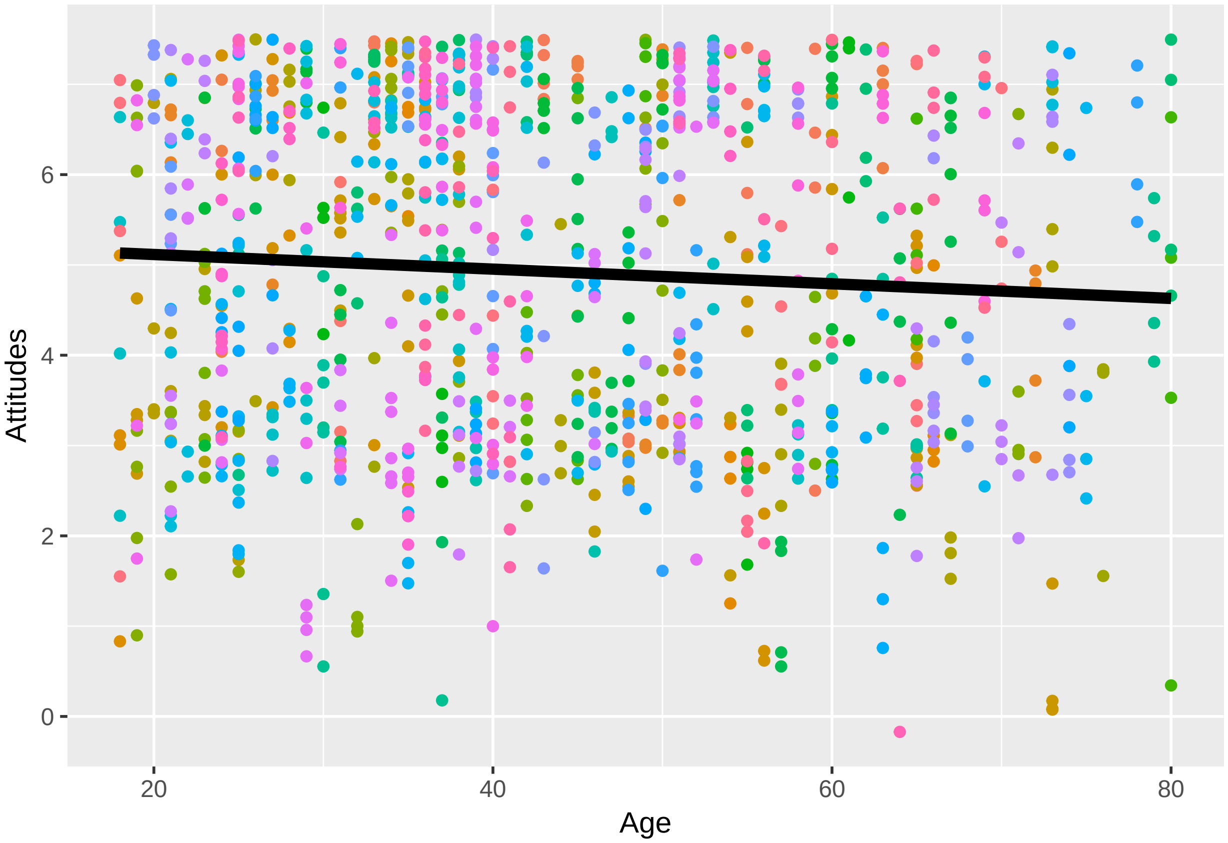
```
## 
## Call:
## lm(formula = Attitudes ~ 1, data = dataAbortion)
## 
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -4.9251 -1.9251  0.0749  2.0749  2.0749 
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept)  4.92512    0.05654    87.1   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 1.837 on 1054 degrees of freedom
```

```r
logLin=logLik(AbortionLinear)*-2
LogInt=logLik(AbortionIntercept)*-2
test=logLin-LogInt
pchisq(test,1,lower=FALSE)
```

```
## 'log Lik.' 4.619714e-12 (df=2)
```

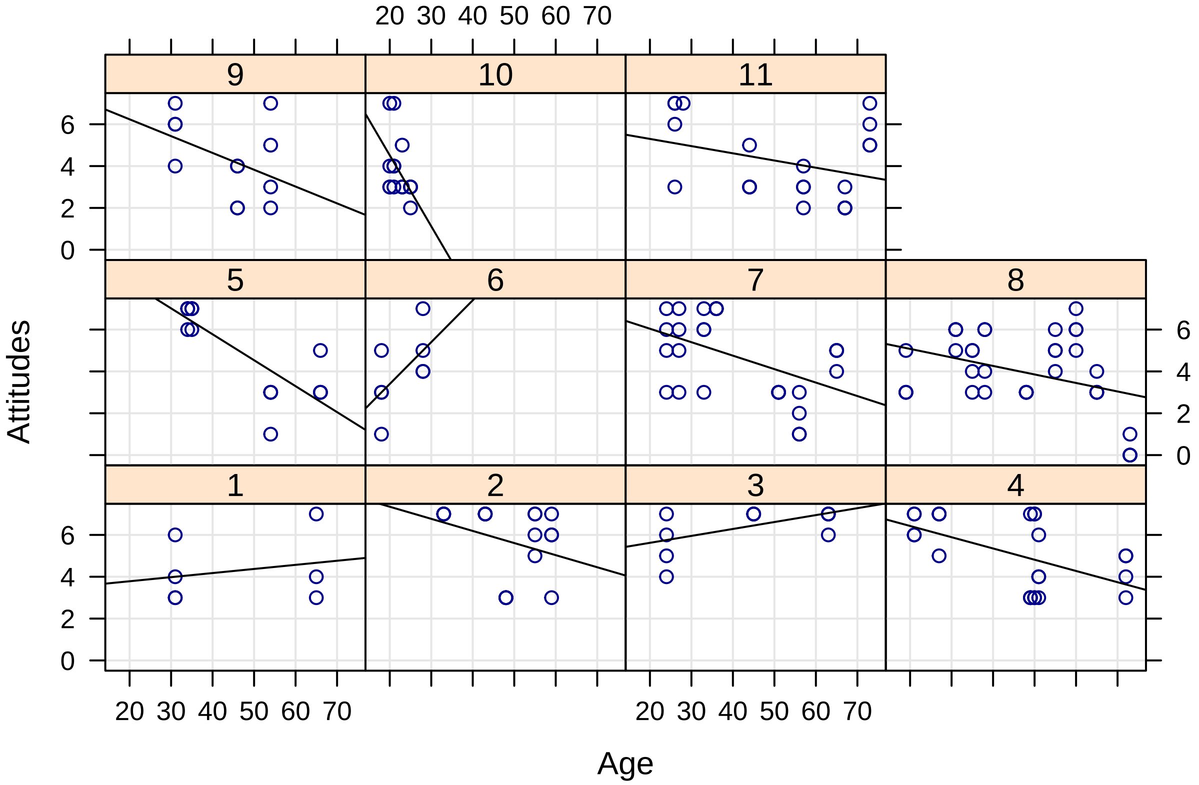
The next step is to model in the effect of age. Let's visualise it first.

```r
ggplot(dataAbortion, aes(x=Age,y=Attitudes,colour = factor(ID))) +
  geom_point(position=position_jitter(w=0.00, h=0.5)) + guides(colour=FALSE) +
  geom_smooth(method=lm, se=FALSE, colour="black", size=2)
```



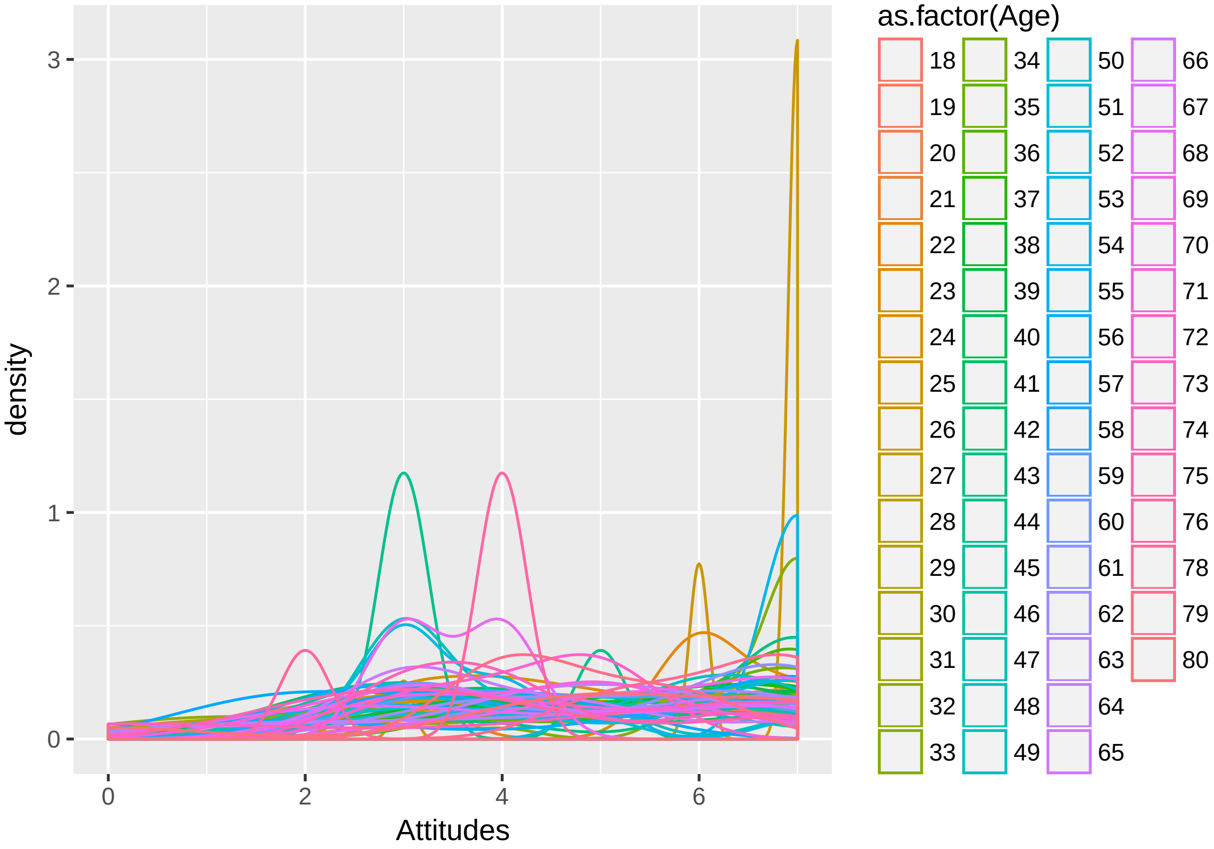
Both graphs, above and below, point out the effect of age. The one below though suggests that slopes of such effects may vary accross groups quite significantly.

```
xyplot(Attitudes~Age|as.factor(ID),data=dataAbortion[1:200,],
       type=c("p","g","r"),
       col="dark blue",col.line="black",
       xlab="Age", ylab="Attitudes")
```



Further examination of the density function adds evidence to the above assumption.

```
ggplot(dataAbortion, aes(x=Attitudes, colour=as.factor(Age))) + geom_density()
```



Let's add age as a predictor to the random intercept model. In the output, pay attention to the change in ICC1, the size and statistical significance of the effect.

```
AbortionIntercept2=lmer(Attitudes~Age+(1|ID),REML=TRUE,
                       data=dataAbortion)
summary(AbortionIntercept2)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: Attitudes ~ Age + (1 | ID)
##    Data: dataAbortion
## 
## REML criterion at convergence: 4223.8
## 
## Scaled residuals: 
##     Min      1Q  Median      3Q     Max 
## -3.1351 -0.8624  0.1434  0.8259  1.8745 
## 
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  ID       (Intercept) 0.4505   0.6712  
##  Residual             2.9725   1.7241  
## Number of obs: 1055, groups:  ID, 54
## 
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)  5.53519    0.19508  28.373
## Age         -0.01382    0.00374  -3.695
## 
```

```
## Correlation of Fixed Effects:
##     (Intr)
## Age -0.831
```

```
0.4505/(0.4505+2.9725)
```

```
## [1] 0.1316097
```

We can now try to fit the random coefficients model, in line with the syntax below (note: age was mean centred prior to the analysis)

```
dataAbortion$AgeMC=scale(dataAbortion$Age,center=TRUE,scale=TRUE)
AbortionMixed=lmer(Attitudes~AgeMC+(AgeMC|ID),REML=TRUE,
                        data=dataAbortion)
summary(AbortionMixed)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: Attitudes ~ AgeMC + (AgeMC | ID)
##    Data: dataAbortion
## 
## REML criterion at convergence: 4186.8
## 
## Scaled residuals:
##     Min      1Q  Median      3Q     Max 
## -2.8605 -0.7935  0.1253  0.8129  2.0350 
## 
## Random effects:
##  Groups   Name        Variance Std.Dev. Corr 
##  ID       (Intercept) 0.4341   0.6589        
##           AgeMC       0.2945   0.5427   -0.15
##  Residual             2.7561   1.6601        
## Number of obs: 1055, groups:  ID, 54
## 
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)   5.0121     0.1094   45.82
## AgeMC        -0.1862     0.1002   -1.86
## 
## Correlation of Fixed Effects:
##       (Intr)
## AgeMC -0.067
```

The size and statistical significance of the effect have changed quite dramatically!

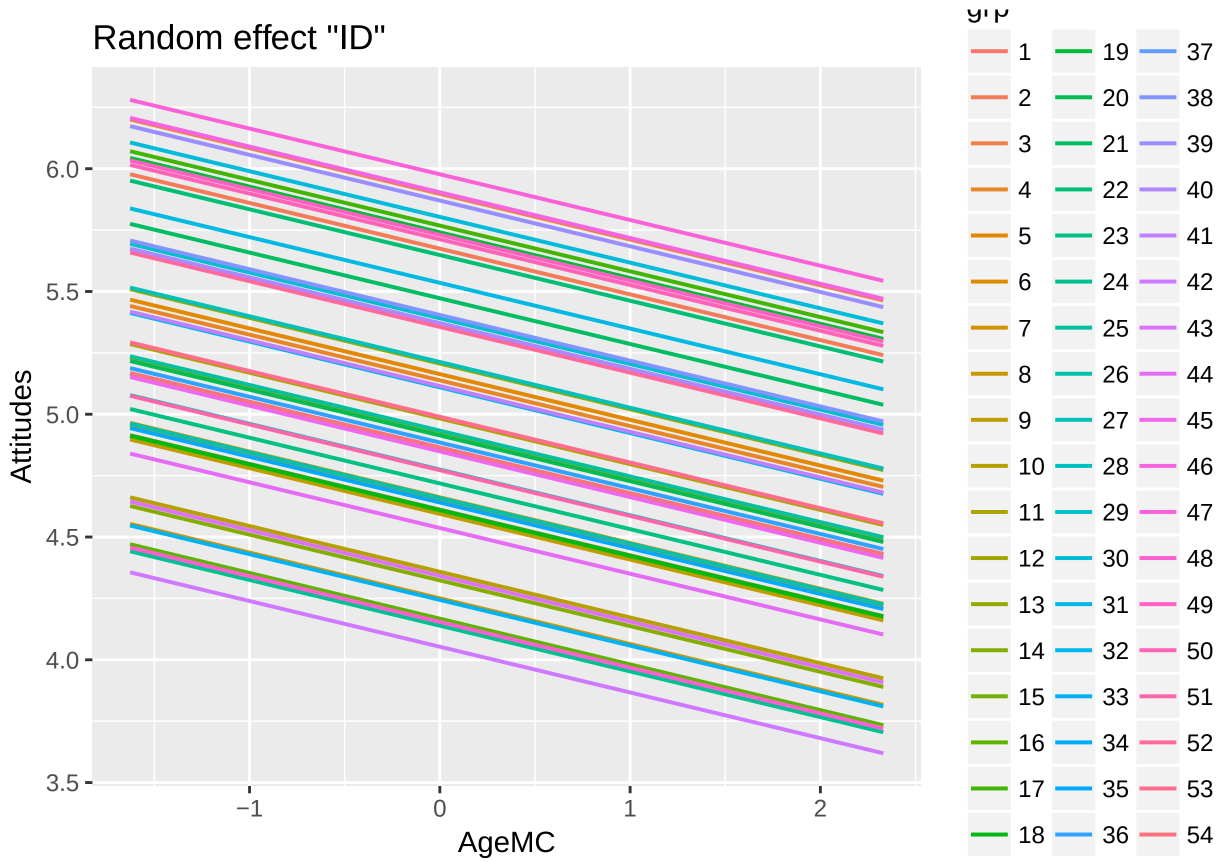
We can again ensure that random coefficients model is better than the random intercept one.

```
logMixed=logLik(AbortionMixed)*-2
LogInt2=logLik(AbortionIntercept2)*-2
test2=LogInt2-logMixed
pchisq(test2,2,lower.tail=FALSE)
```

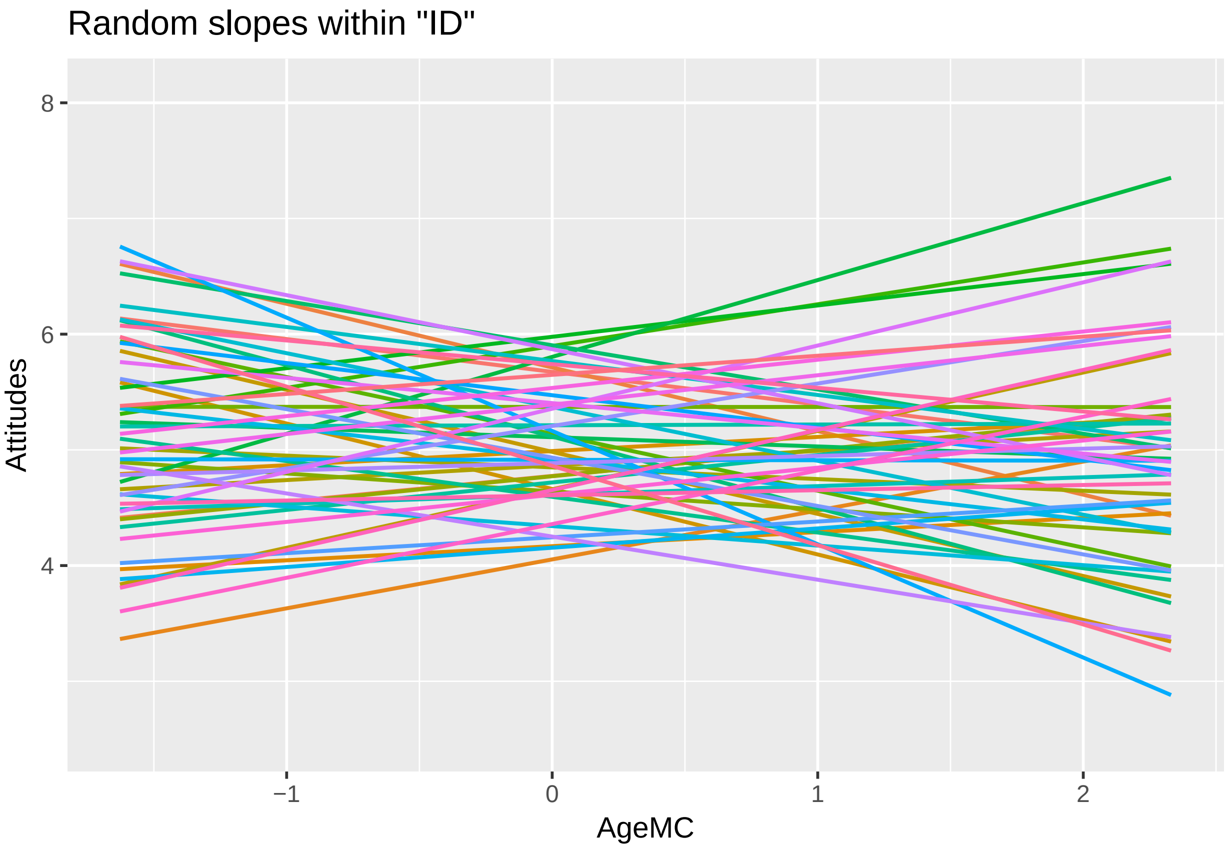
```
## 'log Lik.' 8.835483e-09 (df=4)
```

Finally, we can produce some diagnostic plots. The first two plots exhibit the difference between random intercept and random slopes models.

```
sjp.lmer(AbortionMixed,
         type = "ri.slope",
         facet.grid = FALSE)
```

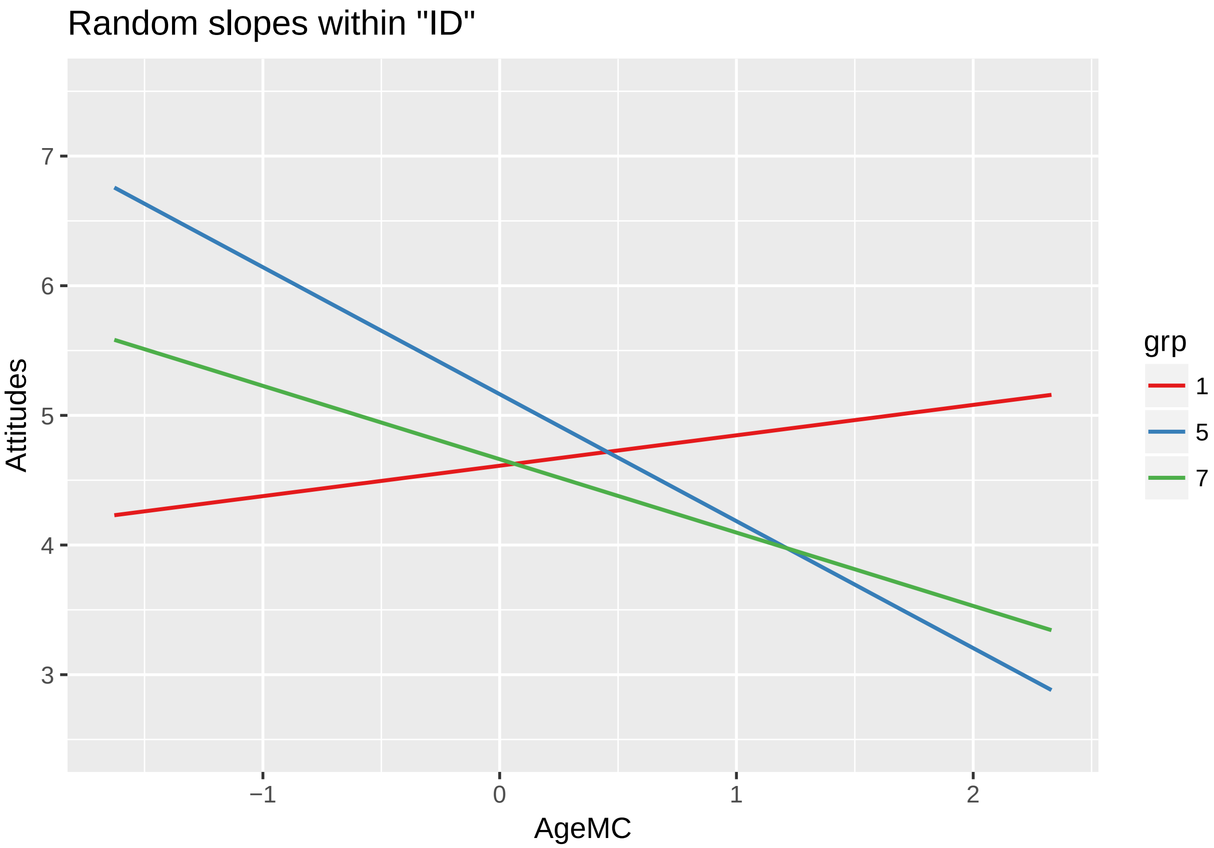


```
sjp.lmer(AbortionMixed, type = "rs.ri", sample.n = 50)
```



The random plot can also be fitted for a fixed number of groups, as shown below.

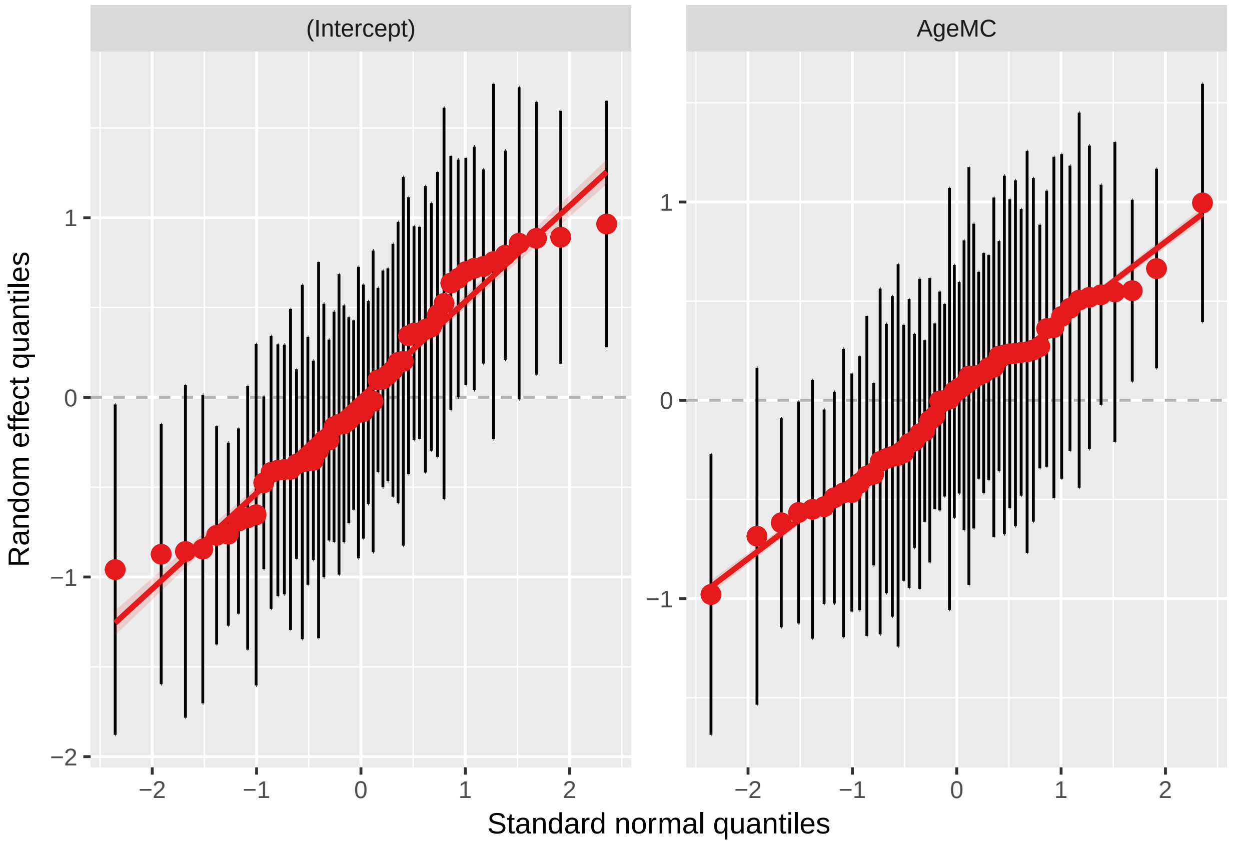
```
sjp.lmer(AbortionMixed, type = "rs.ri",
         sample.n = c(1, 5, 7),
         show.legend = TRUE)
```



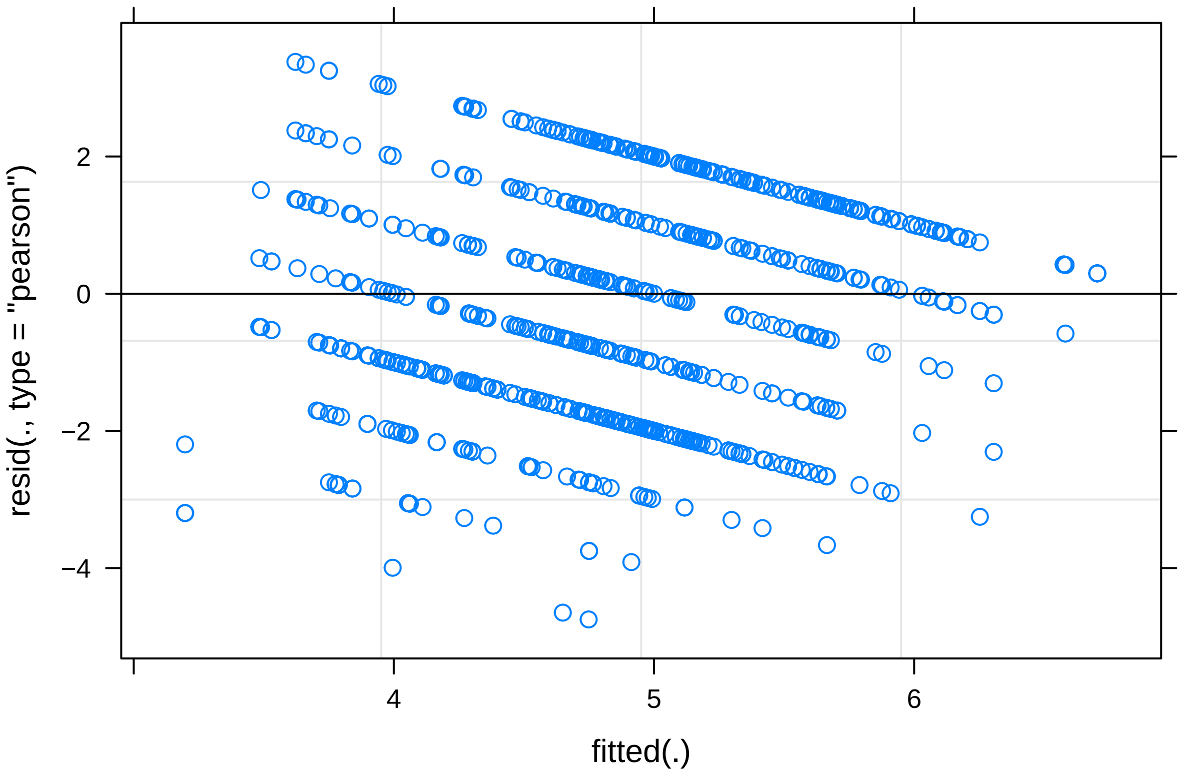
Normality of residuals and homoskedasticity can be cheked as follows.

```
sjp.lmer(AbortionMixed, type = "re.qq")
```

```
## Testing for normal distribution. Dots should be plotted along the line.
```



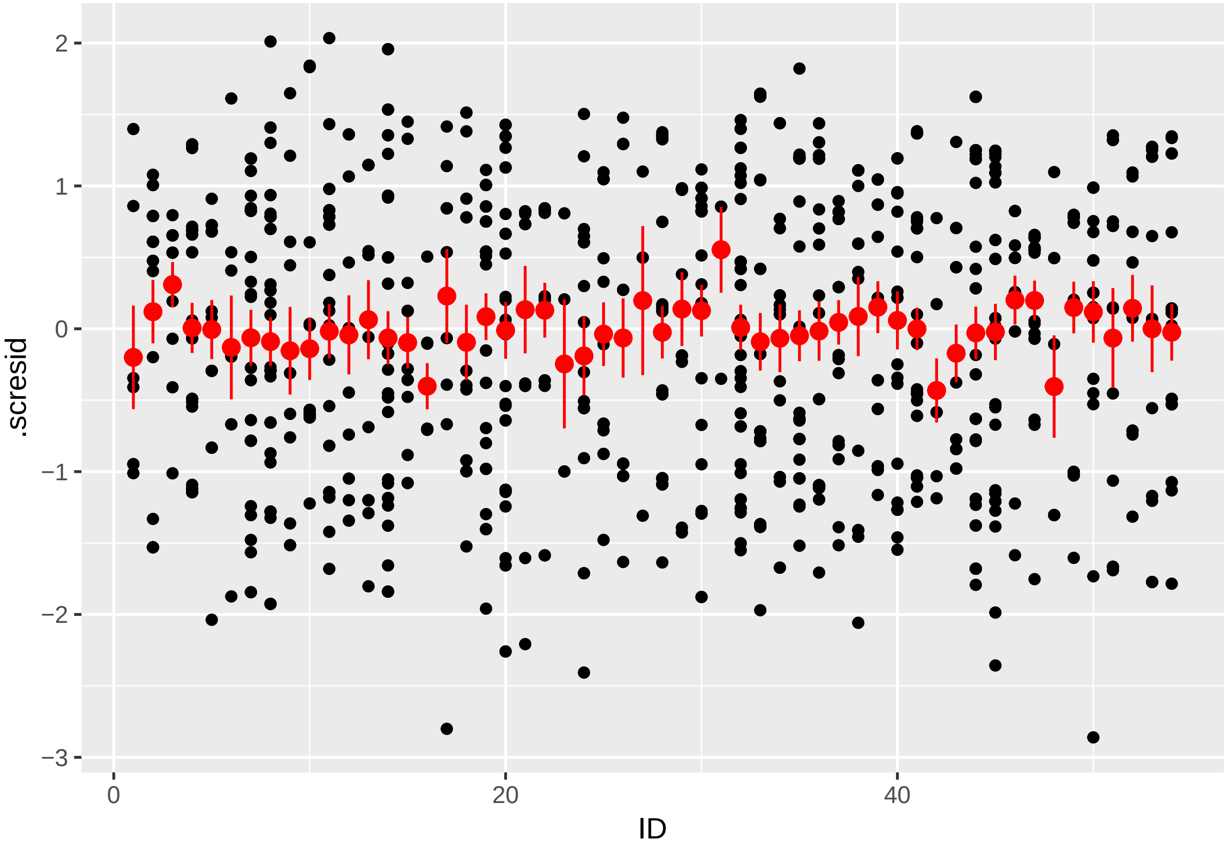
```r
plot(AbortionMixed)
```



Residulas in multilevel modeles are clustred. This can be inspected by the following commands.

```
diag2=fortify(AbortionMixed)
ggplot(diag2, aes(x=ID, y=.scresid)) + geom_point() +
  stat_summary(color="red")
```

```
## No summary function supplied, defaulting to `mean_se()
```


.scresid
ID