TMA4315: Compulsory exercise 3

Group 13: Magnus Liland, Jakob Gerhard Martinussen and Emma Skarstein

23.11.2018

The dataset

We will use a *simulated* dataset with clustered data. This data is generated from a fitted model to the `jsp` dataset in the `faraway` R-package.

The following variables are made available:

- `school`: 50 schools, with code 1-50.
- `gender`: A factor with levels boy, girl.
- `social`: Social class of the father, categorical. Original class 1-2 = S1, 3-4 = S2, 5-6 = S3 and 7-9 = S4 Note that these are not ordered and S1 is not necessarily higher or lower class than S2!
- `raven`: Test score (centered around 0).
- `math`: Math score (centered around 0).

We will use `math` as response, and group the data by school.

```
dataset <- read.table("https://www.math.ntnu.no/emner/TMA4315/2018h/jsp2.txt", header = TRUE)
```

The number of schools is 49, as we omit school number 43 due to the lack of measurements for our particular subset.

Analysis

a) Fitting a linear model

Visualizing the dataset

First we want to explore the dataset. We will group the data based on `gender` and only include covariates `social`, `raven`, and `math`.

```
library(GGally)
ggpairs(
  data = dataset,
  mapping = aes(col = gender, alpha = 0.7),
  columns = c("social", "raven", "math"),
  legend = 1,
)
```

Some observations can be made:

- A positive correlation can be observed between `raven` and `math` score for both genders, although the correlation is marginally stronger for boys. The Raven test measures abstract reasoning capabilities, so this makes intuitive sense.
- Girls perform, on average, better than boys on the math test. The weakest math students are mainly boys, and the strongest math students are mainly girls, as well.

Fitting a linear model

We will now fit a *linear model* with `math` as response, and `raven` and `gender` as covariates. The model for the kth student is therefore

$$Y_k = \mathbf{x}_k^{\mathrm{T}}\boldsymbol{\beta} + \varepsilon_k$$

where we assume that the ε_ks are independent (between students), and have mean 0 and variance σ^2 for all students. Some explanations for the terms in the model:

- Y_k is the `math` score **response** for student k. It is assumed to be normally distributed with mean $\mathbf{x}_k\boldsymbol{\beta}$ and variance σ^2.
- $\mathbf{x}_k$ is a 3×1 **covariate vector** for student k. For instance, if student k is a boy with `raven` score 5, then $\mathbf{x}_k = [1, 5, 0]^{\mathrm{T}}$. Observe that $\mathbf{x}_{k0}$ is always 1, since this covariate "represents" the intercept of the model.
- $\boldsymbol{\beta}$ is the **regression coefficients** of the fitted model. $\boldsymbol{\beta} = [\beta_{\text{intercept}}, \beta_{\text{raven}}, \beta_{\text{gender}}]^{\mathrm{T}}$. For example, you would expect a given student to improve their `math` score by β_{raven} if the student improves their `raven` score by one, everything else being equal.

- ε_k is the **error** in the modelled relationship for student k. The important part here is that ε_k is assumed to be independently and identically distributed with mean 0 and variance σ^2 for *all* students.

Now let's estimate the regression coefficients of this linear model and inspect its summary:

```
model <- lm(math ~ raven + gender, data=dataset)
summary(model)
```

```
##
## Call:
## lm(formula = math ~ raven + gender, data = dataset)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -20.6704  -1.8791   0.1166   2.1166  19.6134
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -1.3131     0.2024  -6.488 1.29e-10 ***
## raven         0.1965     0.0240   8.188 6.98e-16 ***
## gendergirl    2.5381     0.2807   9.041  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.76 on 1151 degrees of freedom
## Multiple R-squared:  0.1105, Adjusted R-squared:  0.109
## F-statistic:  71.5 on 2 and 1151 DF,  p-value: < 2.2e-16
```

The parameter estimates seem to be significant. We have $\beta \approx [-1.3131, 0.1965, 2.5831]^{\mathrm{T}}$. A girl is expected to achieve ~2.5 additional `math` points compared to a boy with the same number of `raven` points. Additionally, a student is expected to score ~0.2 additional `math` points for every `raven` point he or she achieves.

We can rewrite this model as two separate models, one for boys and one for girls, making these relationships clear. For boys

$$Y_k = 0.1965 \cdot x_{k,\text{raven}} - 1.3131,$$

and likewise for girls

$$Y_k = 0.1965 \cdot x_{k,\text{raven}} + 1.225.$$

Making the statistical advantage of female students apparent.

With this model we investigate how two different factors affect the mathematical capabilities of a student. The first of these factors is gender. Gender does indeed seem to have a significant effect, as girls seem to achieve better math scores than boys. Secondly, there is a positive correlation between the performing well on the "Raven" test and the mathematics test. As the Raven test measures abstract, cognitive capabilities, this comes at no large surprise.

---

b) Fitting a random intercept model

Explanation of the model

We will now fit a *random intercept model* with `school` as the random intercept. For school i we study the measurement model:

$$\mathbf{Y}_i = \mathbf{X}_i\beta + \mathbf{1}\gamma_{0i} + \varepsilon_i$$

We denote the number of students in school i as n_i, often called a **cluster**. Here follows some explanations of the different parts of the model:

- $\mathbf{Y}_i$ is the $n_i \times 1$ **response vector** for school i, containing the math scores of each student enrolled in that school.
- $\mathbf{X}_i$ is the $n_i \times 3$ **design matrix** of the model, containing "population covariates" for each school is student on each row.
- The 3×1 vector β contains the **fixed effects** of the model. These "population effects" are common amongst *all* the schools.
- The $n_i \times 1$ vector $\mathbf{1}\gamma_{0i}$ contains the **random effects** of the model. Specifically in this case, since we have fitted a *random intercept model*, we have a **random intercept** and *no* random slope. $\mathbf{1}$ is a $n_i \times 1$ vector containing solely ones, while $\gamma_{0i} \in \mathbb{R}$. With other words, $\mathbf{1}\gamma_{0i} = [\gamma_{0i}, ..., \gamma_{0i}]^{\mathrm{T}}$.
- The $n_i \times 1$ vector ε_i contains the **random errors** in the model prediction.

It is important to note that we have made the following distributional assumptions:

- $\gamma_{0i} \sim \mathrm{N}_1(0, \tau_0^2)$, where τ_0^2 is a scalar which must be estimated.
- $\varepsilon_i \sim \mathrm{N}_{n_i}(0, \sigma^2 I)$, where σ^2, the variance, also must be estimated.

We have also assumed that there is zero correlation between responses in *different* clusters, and we only have **intraclass correlation**. With other words, $\mathrm{Cov}(Y_{ij}, Y_{kl}) = 0$ for $i \neq k$.

Fitting the model

We will now fit this model for our dataset using the `lmer` function from the `lem4` R-package, and print a summary of this fitted model:

```
library(lme4)
fitRI1 <- lmer(
    math ~ raven + gender + (1 | school),
    data = dataset,
)
summary(fitRI1)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: math ~ raven + gender + (1 | school)
##    Data: dataset
##
## REML criterion at convergence: 6772.4
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -4.4607 -0.4305 -0.0127  0.4083  4.2761
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
```

```
##  school   (Intercept)  3.879   1.969
##  Residual             19.220   4.384
## Number of obs: 1154, groups:  school, 49
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept) -1.26915    0.34375  -3.692
## raven        0.21442    0.02331   9.197
## gendergirl   2.51119    0.26684   9.411
##
## Correlation of Fixed Effects:
##            (Intr) raven
## raven      -0.017
## gendergirl -0.404  0.034
```

Comparing the models

We can now compare this *random intercept model* with the *classical linear model* in section a). First observe that β_{raven} has changed from `0.1965` to `0.2144`. The random intercept model has therefore concluded that `raven` score has a even stronger effect upon the `math` score of the pupils. For every additional `raven` score, a student is predicted to score `0.2144` more on the `math` test, everything else being equal.

Also β_{gender} has changed from `2.5381` to `2.5312`. The random intercept model therefore concludes that `gender` has less of an impact on predicted `math` score compared to the classical linear model. The difference is relatively small though, so the models can be said to have similar conclusions regarding gender. A girl is predicted to score `2.5312` more than a boy with the otherwise identical covariate attributes.

Hypothesis testing

Observe that the output of `summary(fitRI1)` does *not* contain any p-values, as we have grown accustomed to with the `lm` and `glm` models. This is intentionally omitted by the `lme4` library authors, as there are a lot of "gotchas" related to the calculation of p-values for parameter estimations wrt. linear mixed models.

Quoting "Fitting Linear Mixed-Effects Models Using lme4" by Bates, Bolker, Mächler and Walker (2015) in Journal of Statistical Software (p. 35):

> While the null distributions (and the sampling distributions of non-null estimates) are asymptotically normal, these distributions are not t distributed for finite size samples – nor are the corresponding null distributions of differences in scaled deviances F distributed.

The T-distribution can therefore not be used for a hypothesis test for finite size samples without making a lot of assumptions which might render the conclusion invalid.

A rationalization can also be found the email thread "lmer, p-values and all that" written by Douglas Bates, the author of the `lme4` R-package.

We will now ignore all these wise words, arguing that our sample size is large enough and well-behaved, and still calculate a p-value for the following hypothesis test.

$$\mathbf{H}_0 : \beta_{\text{raven}} = 0 \quad \text{vs.} \quad \mathbf{H}_1 : \beta_{\text{raven}} \neq 0$$

With other words, we want to find out how probable it is that a student's `raven` score has absolutely no effect on the `math` score of the same student. We will perform this test using the *asymptotic* (normal) distribution of β_{raven}.

We construct the following test statistic:

$$z = \frac{\hat{\beta}_{\text{raven}} - 0}{\hat{se}(\beta_{\text{raven}})}$$

Under the null hypothesis, we assume this test statistic to have a asymptotic standard normal distribution:

$$Z \sim N_1(0, 1)$$

We now want to calculate a two-sided z-test:

$$p = 2 \cdot \Pr(Z \geq |z|)$$

We can calculate the test statistic manually:

```
coefficients <- summary(fitRI1)$coefficients
betaRavenHat <- coefficients[2, 1]
standardErrorRavenHat <- coefficients[2, 2]
ZStatistic <- (betaRavenHat - 0) / standardErrorRavenHat
pValue <- 2 * pnorm(q=abs(ZStatistic), mean=0, sd=0, lower.tail=FALSE)
pValue
```

```
## [1] 0
```

Or by retrieving the t-value directly:

```
tValue <- coefficients[2, 3]
pValue <- 2 * pnorm(q=abs(tValue), mean=0, sd=0, lower.tail=FALSE)
pValue
```

```
## [1] 0
```

Alternatively, we can use the `lmerTest` R-package which provides p-values in the `lmer` summary output:

```
library(lmerTest)
fitRI1Test <- lmer(
    math ~ raven + gender + (1 | school),
    data = dataset,
)
summary(fitRI1Test)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: math ~ raven + gender + (1 | school)
##    Data: dataset
##
## REML criterion at convergence: 6772.4
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -4.4607 -0.4305 -0.0127  0.4083  4.2761
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  school   (Intercept)  3.879   1.969
##  Residual             19.220   4.384
## Number of obs: 1154, groups:  school, 49
```

```
##
## Fixed effects:
##               Estimate Std. Error         df t value Pr(>|t|)
## (Intercept)   -1.26915    0.34375   66.69659  -3.692  0.00045 ***
## raven          0.21442    0.02331 1143.53881   9.197  < 2e-16 ***
## gendergirl     2.51119    0.26684 1130.21403   9.411  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##            (Intr) raven
## raven      -0.017
## gendergirl -0.404  0.034
```

All three methods yield the same result, we can relatively confidently assume `raven` scores to have an impact on `math` scores.

We can also construct a 95% confidence interval for the effect of the female gender on the `math` score, using the Wald approximation for fixed effects:

```
confint(fitRI1, method='Wald')[5,]
```

```
##    2.5 %   97.5 %
## 1.988188 3.034185
```

Since the entirety of the 95% confidence interval is contained by $\mathbb{R}^+$, we can confidently conclude that girls perform better on the `math` test compared to boys.

c) Random intercept model without `gender` covariate

We now fit a random intercept model, still clustered by school, but *without* `gender` as a fixed effect.

```
fitRI2 <- lmer(math ~ raven + (1 | school), data=dataset)
```

Covariance and correlation calculations

The covariance between response $\mathbf{Y}_{ij}$ and $\mathbf{Y}_{il}$ from the same school i has the following form:

$$\text{Cov}(Y_{ij}, Y_{il}) = \begin{cases} \tau_0^2 & \text{, if } j \neq l \\ \tau_0^2 + \sigma^2 & \text{, if } j = l \end{cases}$$

Recall that τ_0^2 is the variance of the normally distributed random intercept, γ_{0i}, and that σ^2 is the variance of the error term ϵ_{ij}, also normally distributed.

Also interest, notice the *compound symmetry* of this expression. The *intraclass correlation* can now be simply derived as:

$$\text{Corr}(Y_{ij}, Y_{il}) = \frac{\tau_0^2}{\tau_0^2 + \sigma^2} \quad , \ j \neq l.$$

We can retrieve these values from the summary of our fit:

```
summary(fitRI2)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: math ~ raven + (1 | school)
##    Data: dataset
##
## REML criterion at convergence: 6856.9
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -4.2705 -0.4725 -0.0045  0.4603  4.4890
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  school   (Intercept)  4.002   2.001
##  Residual             20.711   4.551
## Number of obs: 1154, groups:  school, 49
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)  0.03840    0.32071   0.120
## raven        0.20682    0.02418   8.554
##
## Correlation of Fixed Effects:
##       (Intr)
## raven -0.004
```

For our specific model fit we have $\hat{\tau_0^2} = 4.002$ and $\hat{\sigma^2} = 20.711$. Inserting this into the correlation formula yields $\text{Corr}(Y_{ij}, Y_{il}) \approx 0.162$. Such a correlation calculation will *always* yield a positive correlation, due to model assumptions.

In this case we observe a somewhat weak positive correlation, but not necessarily insignificant. There is good reason to believe that if one student performs well at a school, others will do as well, and vice versa. This can be explained by school resources, social climate, knowledge sharing, and a multitude of other factors.

Random intercept parameter prediction

The formula for γ_{0i} in the random intercept model is

$$\hat{\gamma}_{0i} = \frac{n_i \hat{\tau}_0^2}{\hat{\sigma}^2 + n_i \hat{\tau}_0^2} e_i,$$

e_i is here the **average residual** of the predictions rendered by the *systematic effects* of the model (for school i):

$$e_i := \frac{1}{n_i} \sum_{j=1}^{n_i} (Y_{ij} - \mathbf{x}_{ij}^T \hat{\boldsymbol{\beta}}).$$

We can interpret the formula for γ_{0i} as a *weighted sum* between the conditional expectation 0 and the average residual e_i, the weight being:

$$\frac{n_i \hat{\tau}_0^2}{\hat{\sigma}^2 + n_i \hat{\tau}_0^2}$$

Some notes regarding the weighting:

- $n_i\hat{\tau}_0^2$ is the predicted intraclass covariance weighted by the number of observations in that class. The more observations, the greater credence can be given to the predicted correlation.
- $\hat{\sigma}^2$ is the variance observed in the entire population. If this value is large, it is difficult to distinguish genuine large correlations from random effects arising from large variance.
- The weighting therefore represents the intraclass covariance as a fraction of the population variance.
- If $\hat{\sigma}^2$ grows large compared to $n_i\hat{\tau}_0^2$, population variance dominates intraclass correlation. There is therefore good reason to believe that the intercept should not be set too hight for the class, and the weight approaches 0.
- Oppositely, the weight approaches 1 if intraclass correlation dominates. The intercept is therefore set to the average of the residuals, since all of them can be explained by a normal distribution around the class' intercept.

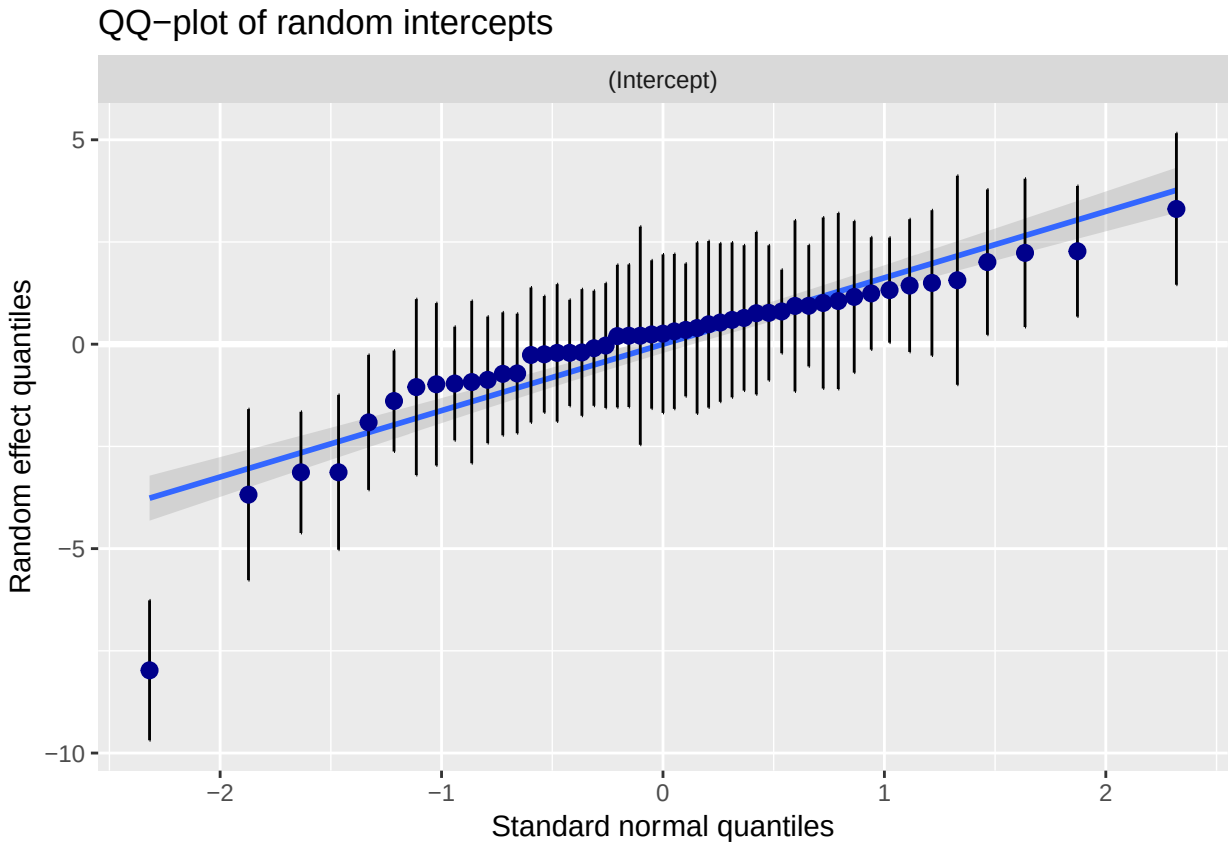
Visual analysis

Now we will use the `ggplot2` R-package in order to inspect model fit and if the model's assumptions are satisfied or not.

QQ-plot of random intercepts

We start of by using the `plot_model` function from the `sjPlot` package in order to check model assumptions.

```
library(ggplot2)
library(sjPlot)
library(ggpubr)
gg1 <- plot_model(fitRI2, type = "diag", prnt.plot = FALSE, geom.size = 1)
gg1[[2]]$school + ggtitle("QQ-plot of random intercepts")
```

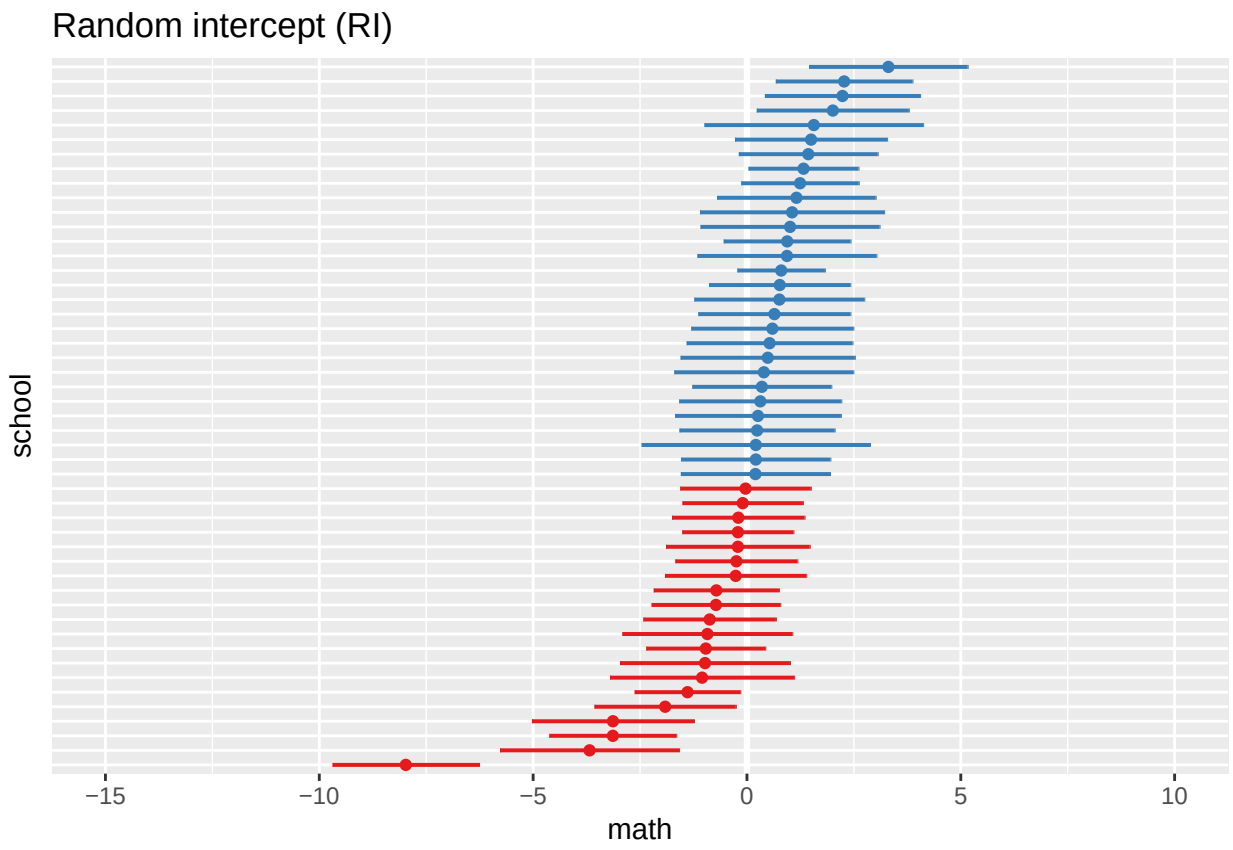


This plot QQ-plot can be used to check if the assumption of normal distribution of the random intercept is satisfied. We expect these data points to lie along the same line, and taking the confidence intervals into account, all of the intercepts besides the left outlier does. All over, this can be considered to satisfy our normal distribution assumption.

Sorted random intercepts plot

Now we plot the *random effects* of the model, i.e. the random intercepts of each school, sorted by intercept value.

```
gg2 <- plot_model(fitRI2, type = "re", sort.est = "(Intercept)", y.offset = 0.4, dot.size = 1.5) +
  theme(axis.text.y = element_blank(), axis.ticks.y = element_blank()) +
  labs(title = "Random intercept (RI)", x = "school", y = "math")
gg2
```



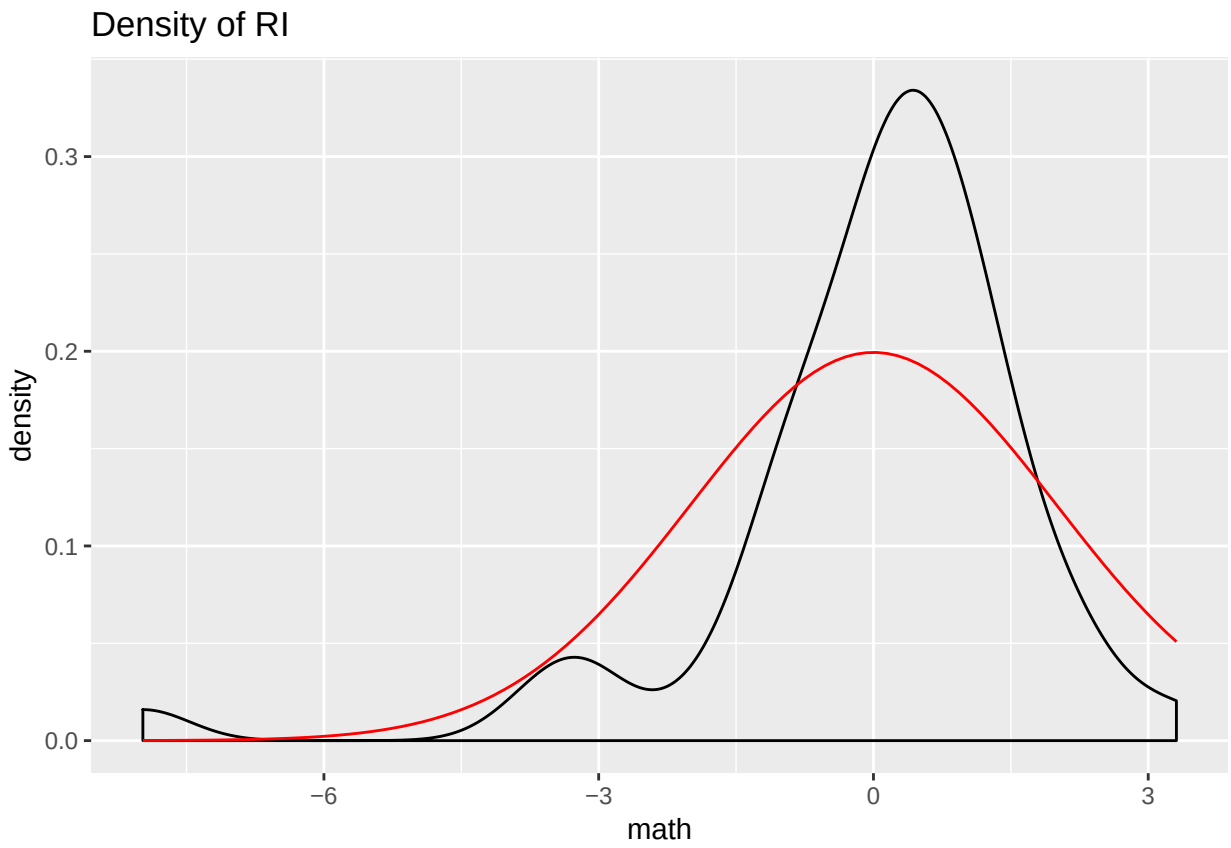
Here we expect a normal distribution with mean zero. With other words, most of the intercepts should be clustered around 0, and should be distributed symmetrically around 0. Again, this seems to be somewhat satisfied, with a close to 50/50 split of negative and positive intercepts.

Density plot of random intercept

Now we plot a density plot of the predicted values $\hat{\gamma}_{0i}$. We compare this observed density plot with the theoretical asymptotic density of γ_{0i} using the `dnorm` function. Here we use $\hat{\tau}_0$ for the standard deviation of the theoretical density.

```
gg3 <- ggplot(data = data.frame(x = ranef(fitRI2)$school[[1]]), aes(x = x)) +
  geom_density() +
  labs(x = "math", y = "density", title = "Density of RI") +
```

```
  stat_function(
      fun = dnorm,
      args = list(mean = 0, sd = attr(VarCorr(fitRI2)$school, "stddev")),
      col = "red",
  )
gg3
```



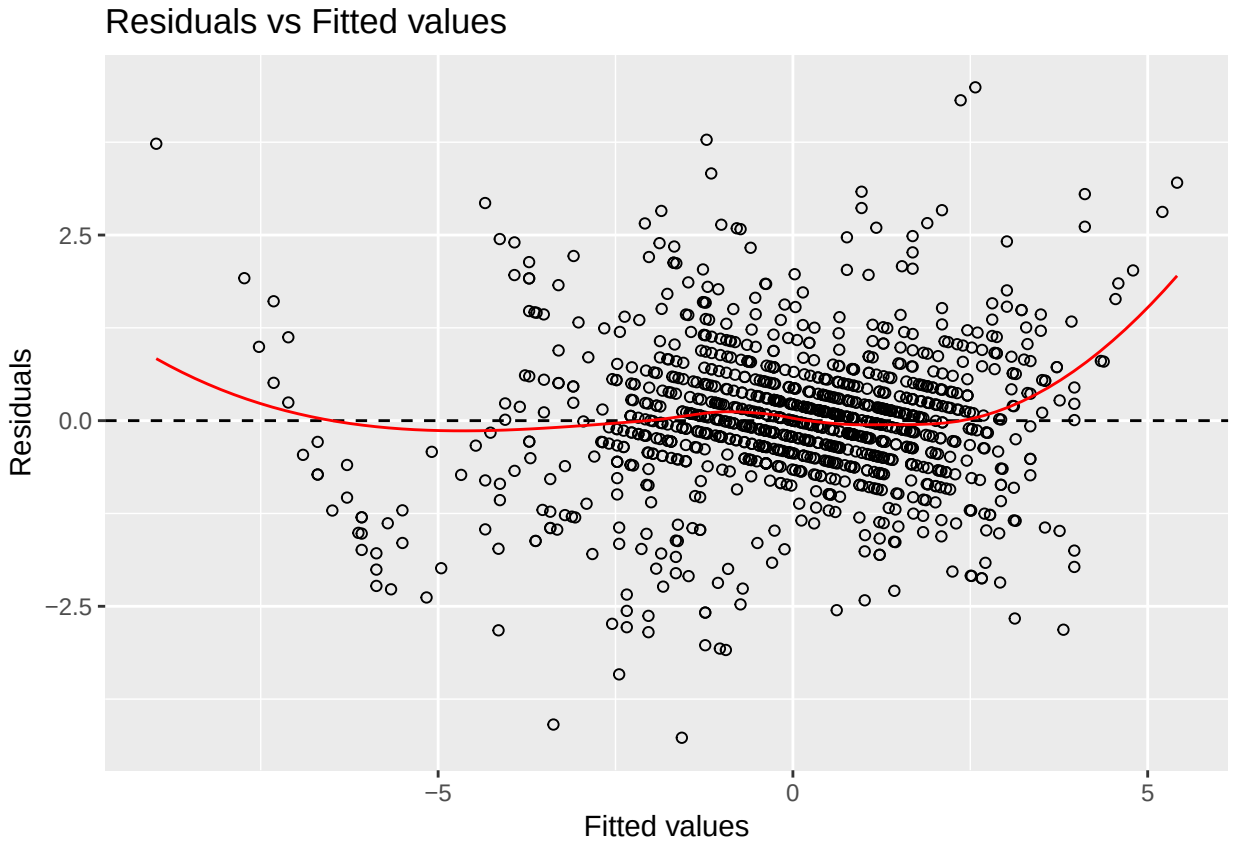
Although the observations seem to form a bell curve, as expected, we all observe minor negative skew in the predicted slopes. With other words, we observe a bit too many schools that perform quite badly.

Small note: research on high schools in Brazil have observed a long tail power law to be applicable for the distribution of high school performance.

Residuals plotted against fitted values

Now we plot the residual values against the fitted values:

```
df <- data.frame(fitted = fitted(fitRI2), resid = residuals(fitRI2, scaled = TRUE))
gg4 <- ggplot(df, aes(fitted,resid)) +
  geom_point(pch = 21) +
  geom_hline(yintercept = 0, linetype = "dashed") +
  geom_smooth(se = FALSE, col = "red", size = 0.5, method = "loess") +
  labs(x = "Fitted values", y = "Residuals", title = "Residuals vs Fitted values")
gg4
```



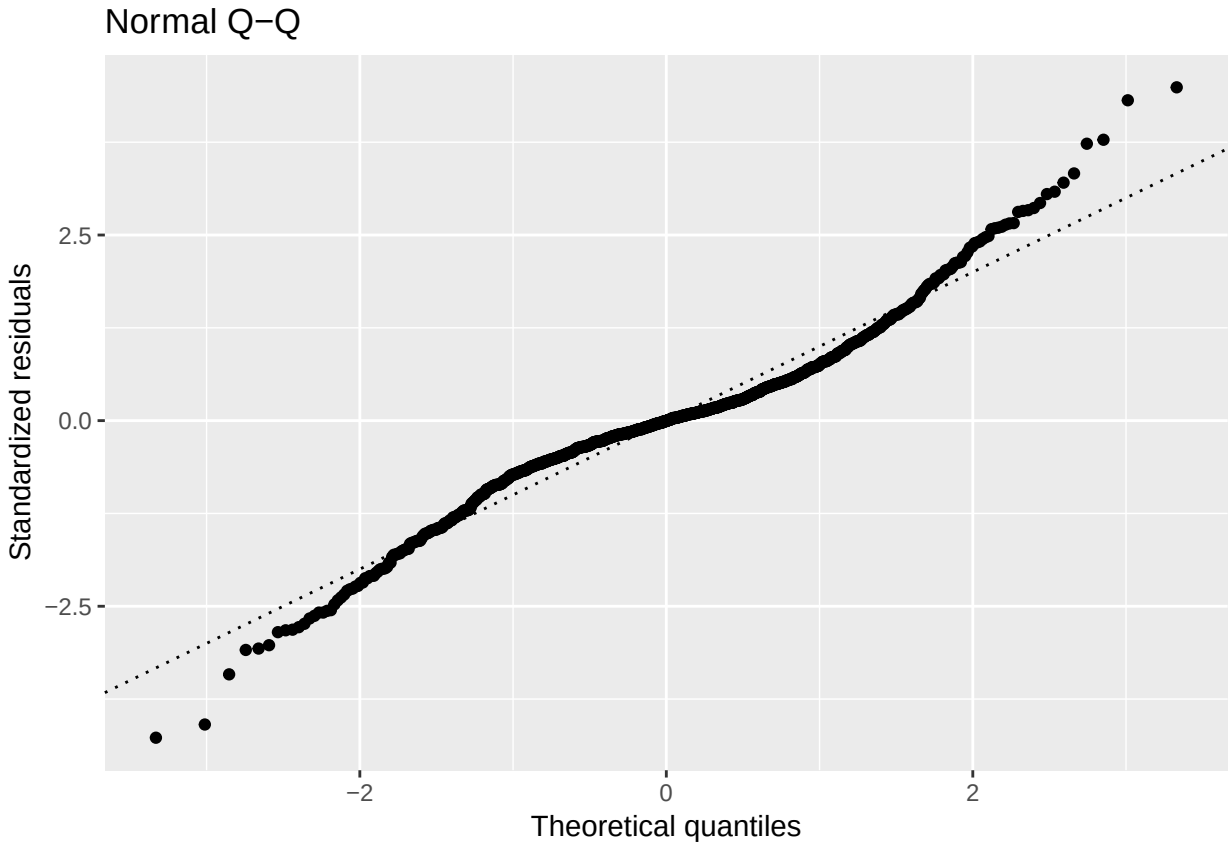
We have made the classical distributional assumption that the error terms ε_i are identically and independently distributed with mean 0 and common variance σ^2. We should therefore observe no discernible trend wrt. the fitted values. Here we use the (standardized) *residuals* as estimators for the *error* terms, and check this assumption.

Luckily, the residuals seem to be symmetrically distributed around 0 and no clear trend is visible.

QQ-plot of standardized residuals

In order to check the *normality* of the residuals, instead of only independence and mean 0 as in the previous plot, we now render a QQ-plot of the standardized residuals.

```
gg5 <- ggplot(df, aes(sample=resid)) +
  stat_qq(pch = 19) +
  geom_abline(intercept = 0, slope = 1, linetype = "dotted") +
  labs(x = "Theoretical quantiles", y = "Standardized residuals", title = "Normal Q-Q")
gg5
```

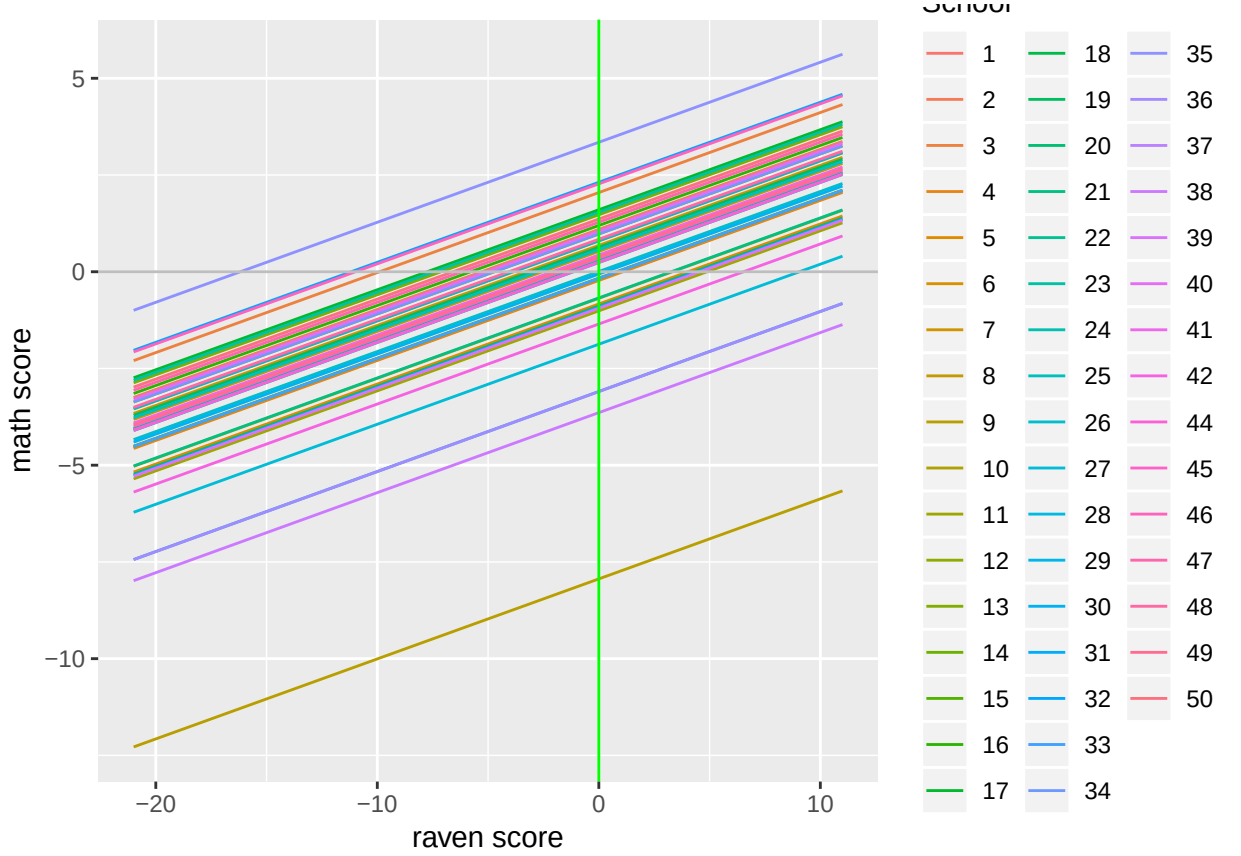


Arguably, the residuals portray normal (as in distribution) behaviour, and we are relatively confident in our assumption of $\varepsilon_i \sim N(0, \sigma^2)$.

Visualizing the predicted random intercepts

We now plot the regression curves for every single school.

```
df <- data.frame(
  x = rep(range(dataset$raven), each = 49),
  y = coef(fitRI2)$school[,1] + coef(fitRI2)$school[,2] * rep(range(dataset$raven), each = 49),
  School = factor(rep(c(1:42, 44:50), times = 2))
)
ggplot(df, aes(x = x, y = y, col = School)) +
  geom_line() +
  geom_hline(yintercept=0, linetype='solid', color='grey') +
  geom_vline(xintercept=0, linetype='solid', color='green') +
  labs(x = "raven score", y = "math score")
```



These lines represent the model predictions split by school, depending on the `raven` score of the student.

As you can see, all the schools have identical slopes but different intercepts, as expected for a random intercept model.

The green slice representing `raven` $= 0$ shows the intercept values for each school, and it is these values that have been discussed thoroughly in previous plots. You could therefore argue that this plot is somewhat redundant, but it offers a clear picture of all of the 49 "sub-models", one for each school.

Again, the intercepts seem to be relatively symmetric around 0, and the badly performing school clearly appears as an outlier in this plot.

d) Expanding the model

Adding social status of father as a fixed effect

We now include the `social` status of the father of the students as a *fixed effect*, and compare this expanded model with the previous "sub-model".

```
fitRI3 <- lmer(math ~ raven + social + (1 | school), data = dataset)
anova(fitRI2, fitRI3)
```

```
## refitting model(s) with ML (instead of REML)
```

```
## Data: dataset
## Models:
## fitRI2: math ~ raven + (1 | school)
## fitRI3: math ~ raven + social + (1 | school)
```

```
##        Df    AIC    BIC  logLik deviance  Chisq Chi Df Pr(>Chisq)
## fitRI2  4 6858.9 6879.1 -3425.4   6850.9
## fitRI3  7 6856.8 6892.1 -3421.4   6842.8 8.1175      3    0.04364 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Note: Observe that `anova` reports *"refitting model(s) with ML (instead of REML)"* when invoked with `lmer` fitted mixed effects models. The reason for this is that `lmer` uses the **restricted maximum likelihood** method to determine the model parameters instead of the classical **maximum likelihood** method. REML is used because it is *less* downwards biased than ML. The problem with REML is that two different covariate structures will result in different mean structures. This causes the log-likelihoods to be based on *different* observations, and the assumptions of ANOVA are therefore not applicable anymore. This is "solved" by refitting the models with the ML method instead, before comparing them with ANOVA analysis.

The addition of social status to the model seems not to have a great impact on the log-likelihood of the model fit. The BIC and AIC heuristics also seem to disagree on which model to choose in this case, with the model `fitRI2` yielding the smallest BIC and `fitRI3` yielding the smallest AIC. This is no big surprise, since the BIC penalizes more for higher complexity. Since the AIC and BIC don't lead to any obvious preference, we would opt for the simpler model in order to prevent overfitting. This is an opinion loosely held, though.

Adding a random slope to the model

We now add a *random slope* for the `raven` score at each school, in addition to the random intercept to the model,

Formulae for random intercept and slope

The cluster specific formula for the random intercept and slope model is as follows:

$$\mathbf{Y}_i = \mathbf{X}_i\beta + \mathbf{U_i}\gamma_i + \varepsilon_i$$

Written more explicitly for our model:

$$\mathbf{Y}_i = \mathbf{1}\beta_{\text{intercept}} + x_{\text{i, raven}}\beta_{\text{raven}} + \mathbf{1}\gamma_{i0} + x_{\text{i, raven}}\gamma_{i,\text{raven}} + \varepsilon_i$$

Fitting the new model

We can now fit this new model and print a summary.

```
fitRIS <- lmer(math ~ raven + (1 + raven | school), data = dataset)
summary(fitRIS)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: math ~ raven + (1 + raven | school)
##    Data: dataset
##
## REML criterion at convergence: 4537.6
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.87462 -0.66206 -0.03913  0.65818  3.09716
##
## Random effects:
##  Groups   Name        Variance Std.Dev. Corr
```

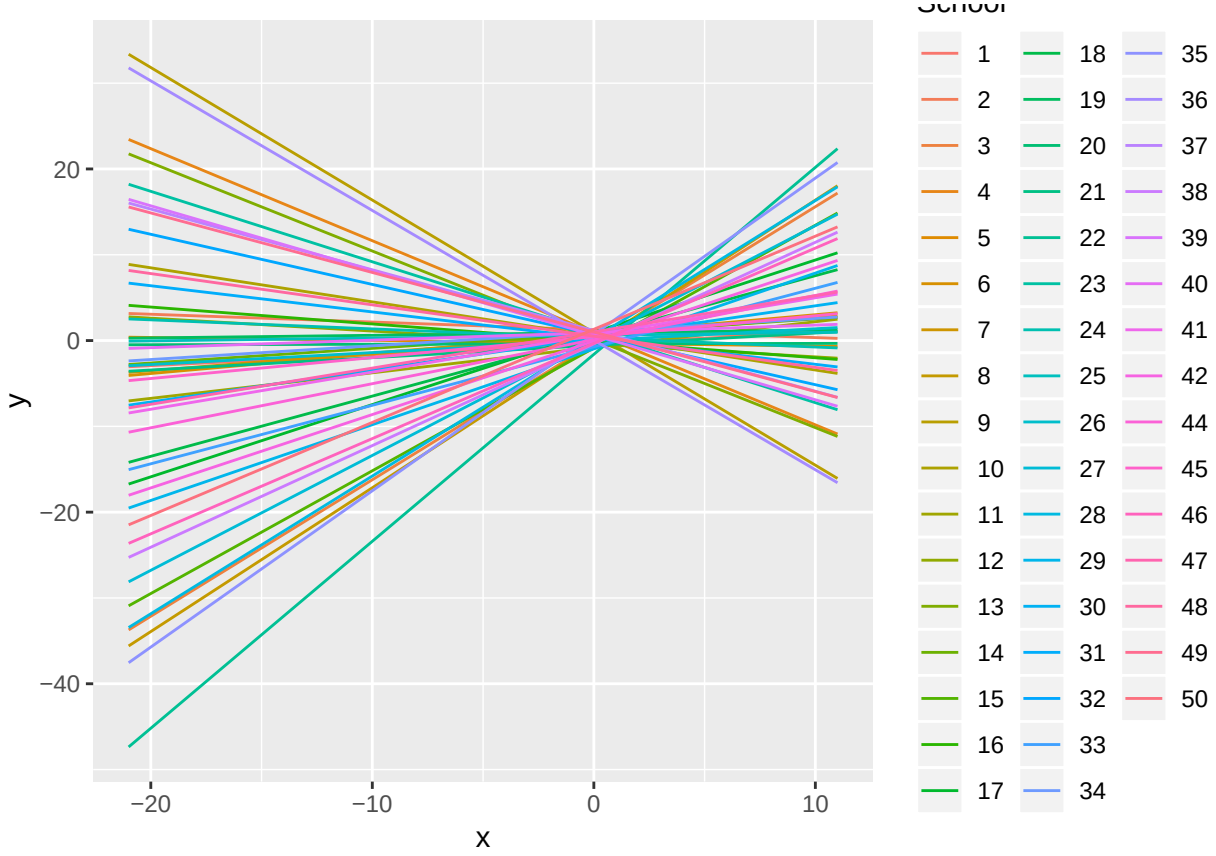
```
##  school   (Intercept) 0.5519   0.7429
##           raven       0.7293   0.8540   -0.40
##  Residual             2.2094   1.4864
## Number of obs: 1154, groups:  school, 49
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)   0.2603     0.1183   2.200
## raven         0.2498     0.1223   2.042
##
## Correlation of Fixed Effects:
##       (Intr)
## raven -0.356
```

The estimated variance of this random slope, 0.7293, is quite a lot more than we expected. Schools vary drastically when it comes to the ability to transfer students capabilities in the `raven` test over to the `math` test, according to our model.

Visualizing sub-models for each school

These different slopes and intercept for each school in this new model can be visualized as follows:

```
df <- data.frame(
  x = rep(range(dataset$raven), each = 49),
  y = coef(fitRIS)$school[,1] + coef(fitRIS)$school[,2] * rep(range(dataset$raven), each = 49),
  School = factor(rep(c(1:42, 44:50), times = 2))
)
gg1 <- ggplot(df, aes(x = x, y = y, col = School)) +
  geom_line()
gg1
```



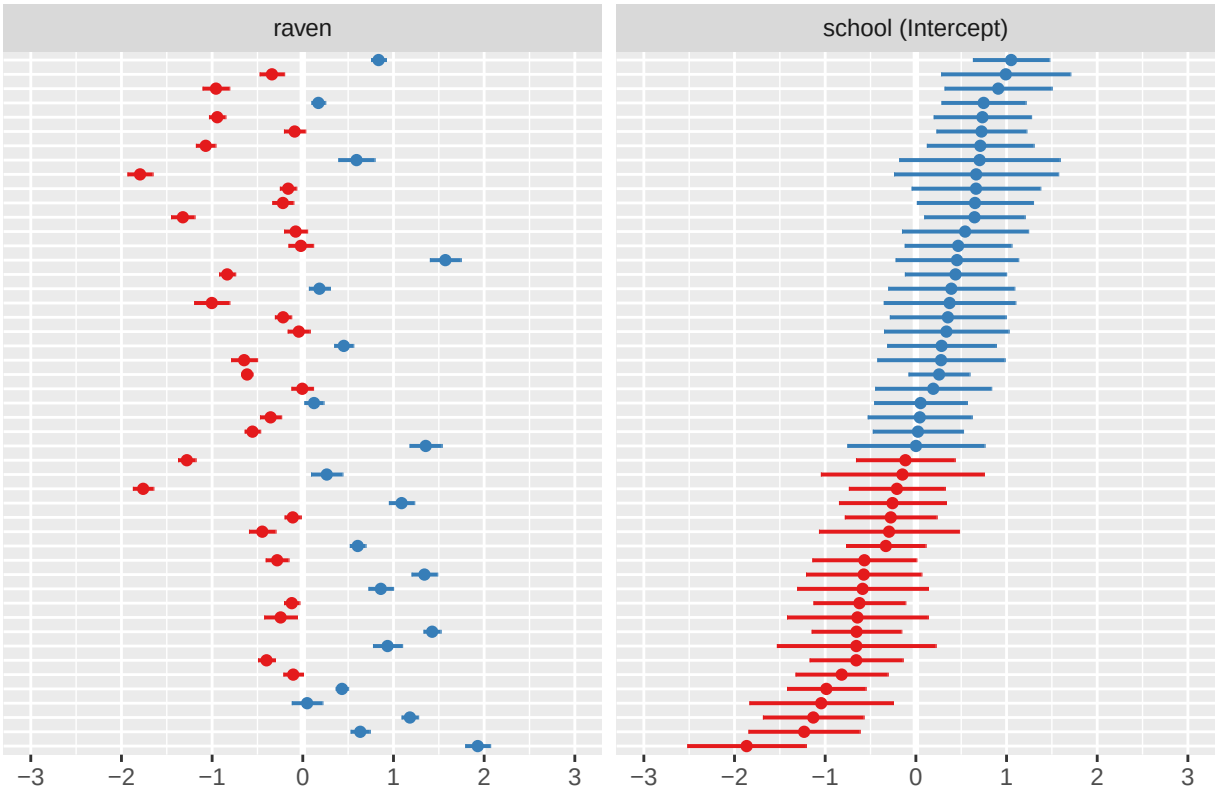
The variance of the intercepts have been greatly reduced, and much of the differences between schools are now attributed to different slopes instead.

Visualizing distribution of random intercepts and slopes

We can also visualize the values for the predicted intercepts and slopes of this mixed effects model:

```
gg2 <- plot_model(fitRIS, type = "re", sort.est = "(Intercept)", y.offset = 0.4, dot.size = 1.5) +
  theme(axis.text.y = element_blank(), axis.ticks.y = element_blank()) +
  labs(title = "Random intercept (RI)")
gg2
```

Random intercept (RI)



Again, we observe relative symmetric distribution around 0 for both prediction types, as assumed.

e) Modelling probability to fail in maths

Let's say that we now want to model the *probability* for a student to fail maths. A *linear* mixed effect model is not suited for this task, since the linear model predicts specific numerical scores on the real number interval $[-\infty, \infty]$.

We are now interested in predicting a *probability* ranging from 0 to 1 instead. A more fitting *random component* for such a model would therefore be

$$y_{ij} \sim \text{Bin}(1, \pi_{ij}),$$

i.e. a binomial distribution with probability $1 - \pi_{ij}$ to fail in maths. This requires us to fit a *generalized* linear mixed effect model, where we pre-process the math score such that it is dichotomized based on the failing grade cut-off:

$$y_{ij} = \begin{cases} 0, \text{ if failing score} \\ 1, \text{ else} \end{cases}$$

In order to introduce a random school intercept into the model, we choose a fitting link function $g(\pi_{ij})$, for example the logit link function, such that:

$$g(\pi_{ij}) = \mathbf{X}_{ij}\beta + \gamma_{0i}$$

With γ_{0i} having the same properties as explained before.

Such a model is not without its intrinsic difficulties, though. One of the main problems is the derivation of a *marginal* probability distribution for the model

$$f(y_{ij}) = \int_{\gamma_i} f(y_{ij}|\gamma_i) f(\gamma_i) \mathrm{d}\gamma_i.$$

The integral of an integrand consisting of the product of a binomial *and* a normal distribution is not known to us, and thus we face analytical difficulties. Other numerical techniques are required, and is out of scope for this course.