Multilevel Modeling Formative

Anonymous Marking Code: Z0195806

2024-03-01

Part 1: Introduction

1.1 Overview of Randomized and Cluster Randomized Trials

Randomized Controlled Trials (RCTs) are the gold standard for evaluating intervention effectiveness, minimizing bias by randomly assigning participants to intervention or control groups. Cluster Randomized Trials (CRTs), a subset of RCTs, randomize groups or 'clusters' (like schools or communities) instead of individuals, making them ideal for educational research where interventions are applied at group levels. This methodology addresses data's inherent groupings, ensuring a more accurate assessment of interventions.

1.2 Intro to the Dataset

1.2.1 Prepare for the Runtime and Dataset

gridExtra is a good library for combining multiple ggplots graphs in one graph, while par() can't make it.

```
# -------------------------------------------------------
## clear the environment var area
# rm(list = ls())
## clear all plots
# graphics.off()
## clear the console area
# cat("\014")
# -------------------------------------------------------
# install.packages("gridExtra")
# -------------------------------------------------------
require(lme4)
require(lmerTest)
require(ggplot2)
require(sjPlot)
require(gridExtra)
```

Download data set from GitHub and show the first lines.

```
CRT <-
  read.csv("https://andygolightly.github.io/teaching/MATH43515/CRT.csv",
           header = TRUE)
head(CRT)
```

```
##   Pupil School Posttest Intervention Pretest FSM class
## 1     1      1       16            1       1   0     1
## 2     2      1       13            1       4   1     1
## 3     3      1       18            1       5   1     1
## 4     4      1       14            1       4   1     1
## 5     5      1       25            1       5   1     1
## 6     6      1       13            1       2   0     1
```

```
# dim(CRT)
```

1.2.2 Dataset Overview

For our analysis, we have access to a dataset comprising data collected from a cluster randomized trial in an educational setting. The dataset includes the following variables:

- **Pupil**: Anonymized student ID, ensuring the confidentiality of participants' identities.
- **School**: Anonymized school ID, ranging from 1 to 20, representing the different clusters within the study.
- **Class**: Anonymized class ID within schools, categorized into two groups (1 or 2), indicating the specific class units.
- **Intervention**: An indicator of whether a pupil was in the treatment group (1) receiving the educational intervention or in the control group (0) not receiving the intervention.
- **FSM**: An indicator of pupil eligibility for free school meals (1 = eligible, 0 = not eligible), serving as a proxy for socio-economic status.
- **Pretest**: Scores from a pre-test administered to all pupils before the intervention, providing a baseline measure of their knowledge or skills.
- **Posttest**: Scores from a post-test administered after the intervention, serving as the response variable to assess the impact of the educational intervention on pupil outcomes.

1.3 Exploratory Data Analysis

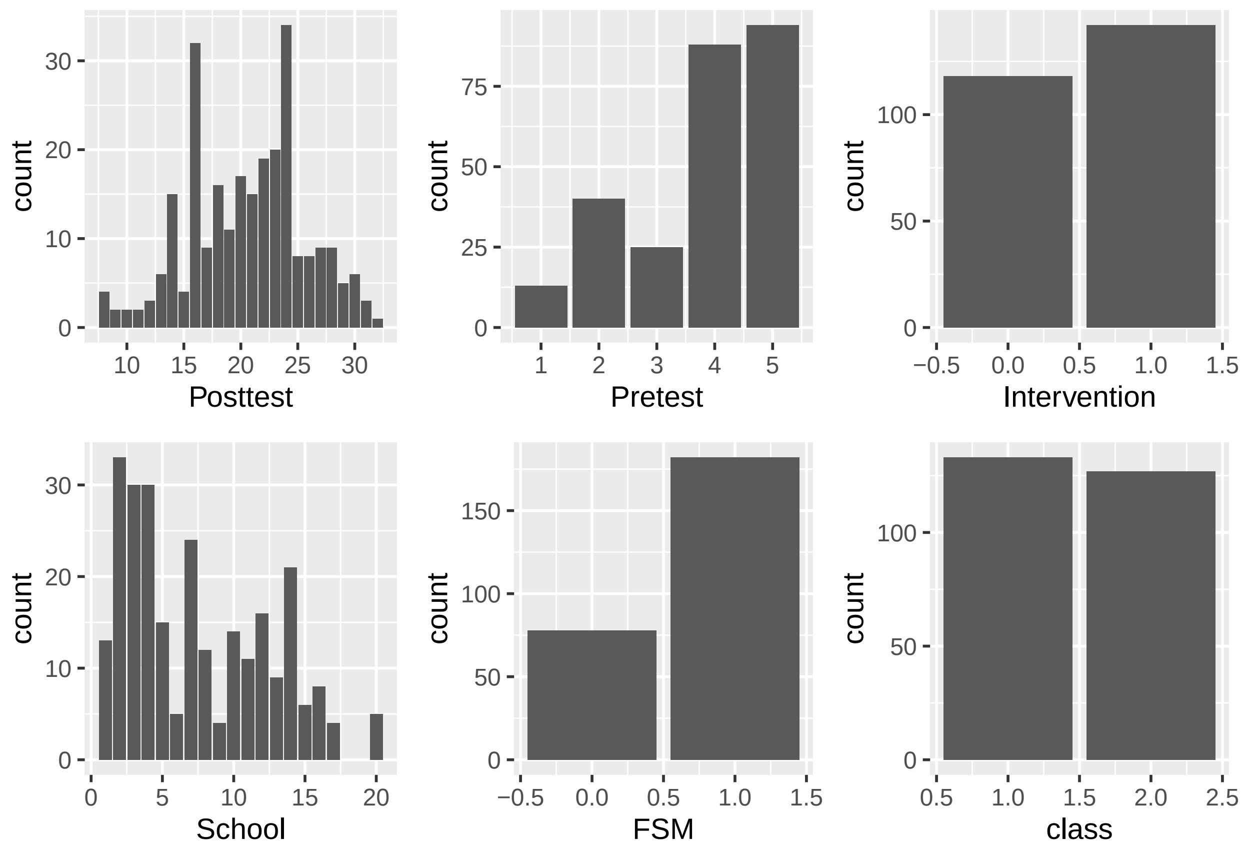
1.3.1 Single Variables Descriptive Statistics

```
summary(CRT)
```

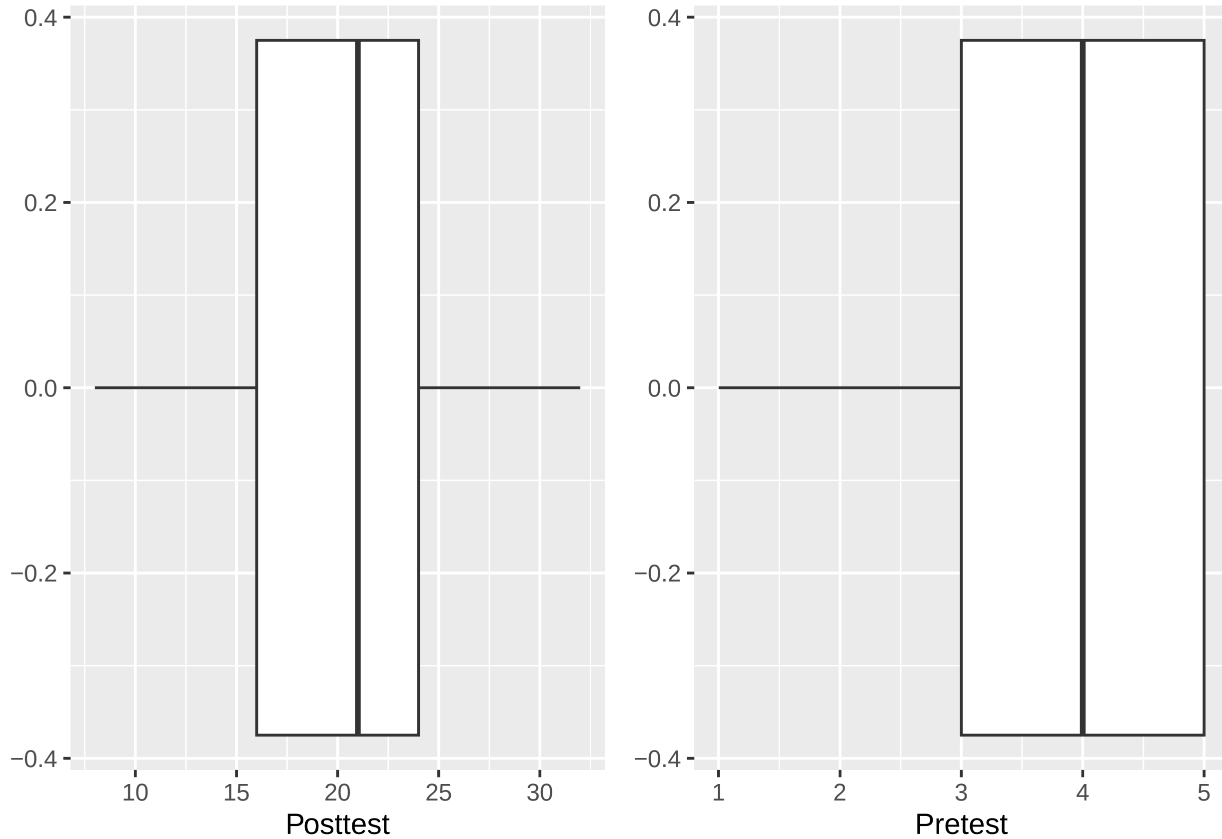
```
##      Pupil            School          Posttest      Intervention
##  Min.   :  1.00   Min.   : 1.000   Min.   : 8.00   Min.   :0.0000
##  1st Qu.: 65.75   1st Qu.: 3.000   1st Qu.:16.00   1st Qu.:0.0000
##  Median :130.50   Median : 7.000   Median :21.00   Median :1.0000
##  Mean   :130.56   Mean   : 7.477   Mean   :20.54   Mean   :0.5462
##  3rd Qu.:195.25   3rd Qu.:12.000   3rd Qu.:24.00   3rd Qu.:1.0000
##  Max.   :263.00   Max.   :20.000   Max.   :32.00   Max.   :1.0000
##     Pretest           FSM           class
##  Min.   :1.000   Min.   :0.0   Min.   :1.000
##  1st Qu.:3.000   1st Qu.:0.0   1st Qu.:1.000
##  Median :4.000   Median :1.0   Median :1.000
```

```
##  Mean   :3.808   Mean   :0.7   Mean   :1.488
##  3rd Qu.:5.000   3rd Qu.:1.0   3rd Qu.:2.000
##  Max.   :5.000   Max.   :1.0   Max.   :2.000
```

```r
# --------------------------------------------------------
bar.Posttest = ggplot(data = CRT, aes(Posttest)) +
  geom_bar()
bar.Pretest = ggplot(data = CRT, aes(Pretest)) +
  geom_bar()
bar.Intervention = ggplot(data = CRT, aes(Intervention)) +
  geom_bar()
bar.School = ggplot(data = CRT, aes(School)) +
  geom_bar()
bar.FSM = ggplot(data = CRT, aes(FSM)) +
  geom_bar()
bar.class = ggplot(data = CRT, aes(class)) +
  geom_bar()
# --------------------------------------------------------
# boxplot
boxplot.Posttest = ggplot(data = CRT, aes(Posttest)) +
  geom_boxplot(outlier.colour = "red", outlier.shape = 1)
boxplot.Pretest = ggplot(data = CRT, aes(Pretest)) +
  geom_boxplot(outlier.colour = "red", outlier.shape = 1)
# --------------------------------------------------------
# put these bar charts together
grid.arrange(bar.Posttest,
             bar.Pretest,
             bar.Intervention,
             bar.School,
             bar.FSM,
             bar.class,
             ncol = 3)
```



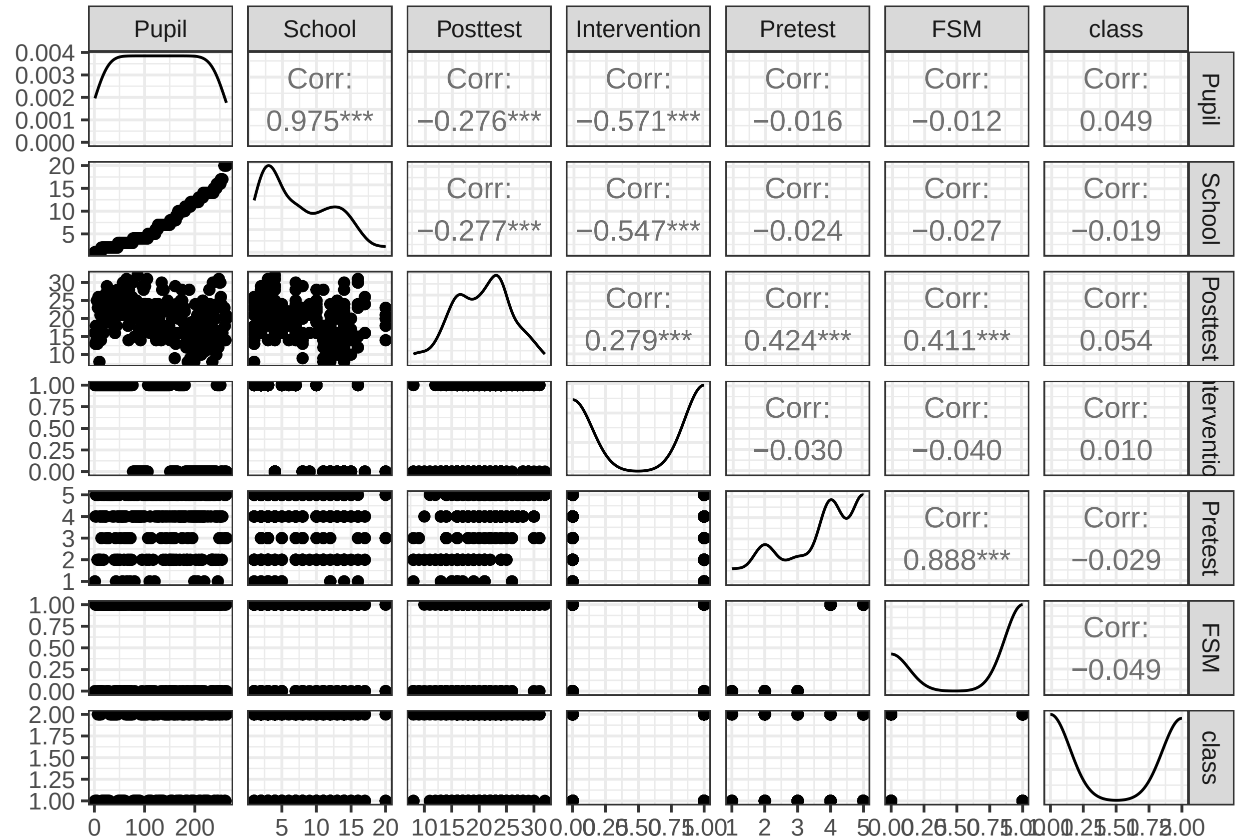

```
grid.arrange(boxplot.Posttest,
             boxplot.Pretest,
             ncol = 2)
```




```
# ---------------------------------------------------------
# remove the plot cache memory
remove(bar.Posttest)
remove(bar.Pretest)
remove(bar.Intervention)
remove(bar.School)
remove(bar.FSM)
remove(bar.class)
remove(boxplot.Posttest)
remove(boxplot.Pretest)
```

1.3.2 Correlation between variables

```
# Correlation between two varibales with GGpairs
library("GGally")
ggpairs(CRT)+theme_bw()
```



Part 2: Methods

2.1 Multilevel Models in RCTs Overview

Multilevel models, also known as hierarchical linear models or mixed-effects models, are essential for analyzing RCT data with nested structures, such as participants within classrooms or schools. These models address the independence assumption violation by accounting for data hierarchy, making them ideal for cluster RCTs where interventions target groups, leading to intra-cluster correlations.

2.2 Application to a 3-Level Cluster Randomized Trial

The educational intervention dataset under study is an example of a 3-level cluster randomized trial, where pupils (level 1) are nested within classes (level 2), which in turn are nested within schools (level 3). Multilevel modeling is particularly apt for this dataset as it allows us to account for the variability at each of these levels, providing more accurate estimates of the intervention effects and associated uncertainties.

2.3 Variance Decomposition and ICC

In multilevel modeling, variance decomposition and the Intra-Class Correlation Coefficient (ICC) are central concepts. Variance decomposition breaks down the total variance in the outcome into components associated with each hierarchical level. The ICC, calculated from this decomposition, indicates the variance proportion due to clustering at higher levels. High ICC values highlight the importance of multilevel modeling. Testing

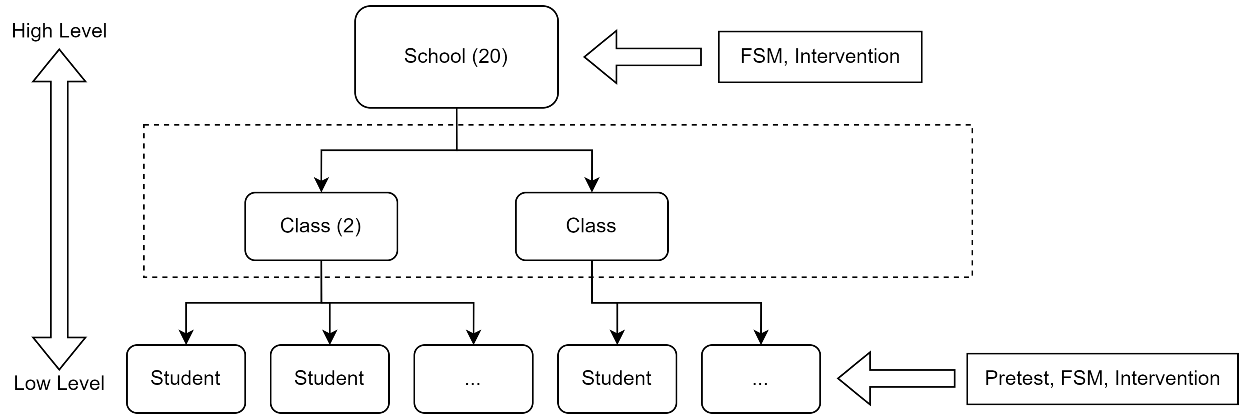


Figure 1: Hierarchical Data

for significant variance components involves likelihood ratio tests to compare models with and against random effects at each level, guiding the inclusion of relevant variance components.

$$VPC = \frac{\sigma^2_{between}}{\sigma^2_{between} + \sigma^2_{within}}$$

where $\sigma^2_{between}$ is the variance between clusters (e.g., schools), and σ^2_{within} is the variance within clusters (e.g., individual students).

$$ICC = \frac{\sigma^2_{between}}{\sigma^2_{between} + \sigma^2_{within}}$$

2.4 Methodological Approach

In multilevel modeling, assessing variance components' significance across levels is crucial, often achieved through likelihood ratio tests comparing models with varying random effects. Initially, a null model estimates ICC by partitioning variance, followed by models incorporating fixed effects (like intervention and pre-test scores) and random intercepts (for classes and schools). Model efficacy is then evaluated using likelihood ratio tests alongside AIC/BIC criteria, ensuring the inclusion of meaningful variance components and optimizing model fit.

Part 3: Analysis

In this section, we conduct a detailed analysis using multilevel modeling to understand the effects of the educational intervention. We fit several models to our data, compare them using appropriate diagnostic tools, and address key research questions.

3.1 Empty Model

3.1.1 Original Model Fitting (2 or 3 levels?)

We start by fitting a basic multilevel model without the intervention variable to establish a baseline. This model includes random intercepts for schools and classes to account for the hierarchical structure of our data.

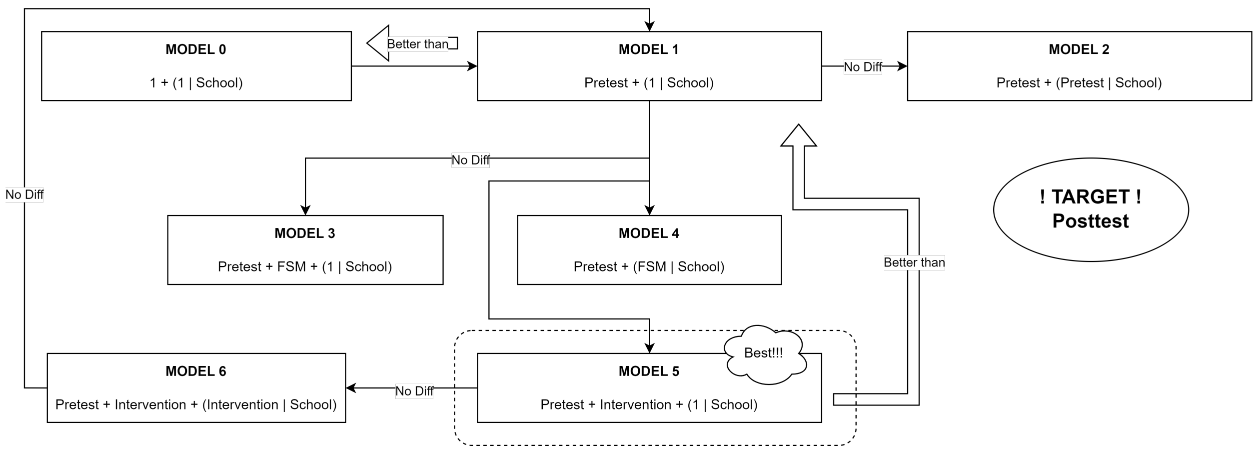


Figure 2: Create Models Bottom-up

```
# empty model / intercept-only mode
Model.0.2level = lmer(Posttest ~ 1 + (1 | School),
                      data = CRT)
Model.0.3level = lmer(Posttest ~ 1 + (1 | School)
                      + (1 | School:class),
                      data = CRT)
ranova(Model.0.2level)
```

```
## ANOVA-like table for random-effects: Single term deletions
##
## Model:
## Posttest ~ (1 | School)
##                npar  logLik    AIC   LRT Df Pr(>Chisq)
## <none>            3 -768.93 1543.9
## (1 | School)      2 -792.14 1588.3 46.43  1  9.494e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
ranova(Model.0.3level)
```

```
## ANOVA-like table for random-effects: Single term deletions
##
## Model:
## Posttest ~ (1 | School) + (1 | School:class)
##                      npar  logLik    AIC    LRT Df Pr(>Chisq)
## <none>                  4 -768.73 1545.5
## (1 | School)            3 -772.30 1550.6 7.1322  1   0.007571 **
## (1 | School:class)      3 -768.93 1543.9 0.3933  1   0.530550
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Model.0.2level is better
Model.0 = Model.0.2level
```

3.1.2 Intra-Class Correlations (ICCs) and Variance Partition Coefficients

We calculate the ICCs to quantify how much of the total variance in post-test scores is attributable to differences between schools and classes. The VPCs further help us understand the proportion of variance explained by each level of the hierarchy. (Finally we can say only School-Pupil is better)

```
# require(performance)
## https://easystats.github.io/performance/reference/icc.html
# icc(Model.0)
# icc(Model.1)
# icc(Model.5)
Model.0.REsummary <- as.data.frame(VarCorr(Model.0.3level))
Model.0.REsummary
```

```
##            grp        var1 var2       vcov     sdcor
## 1 School:class (Intercept) <NA>  0.6930698 0.8325081
## 2       School (Intercept) <NA>  5.9880288 2.4470449
## 3     Residual        <NA> <NA> 19.2260578 4.3847529
```

```
sig <- Model.0.REsummary$vcov[3]  #Residual variance
sigv <- Model.0.REsummary$vcov[2] #RE variance for school
sigu <- Model.0.REsummary$vcov[1] #RE variance for class
totalvar <- sum(Model.0.REsummary$vcov) #total variance
vpc.school <- sigv / totalvar
vpc.class <- sigu / totalvar
cat('The VPC of School is: ', vpc.school, '\n')
```

```
## The VPC of School is:  0.2311342
```

```
cat('The VPC of class is: ', vpc.class, '\n')
```

```
## The VPC of class is:  0.02675206
```

```
# -------------------------------------------------------
icc.school <- sigv / totalvar
icc.class <- (sigu + sigv) / totalvar
cat('The ICC of School is: ', icc.school, '\n')
```

```
## The ICC of School is:  0.2311342
```

```
cat('The ICC of class is: ', icc.class, '\n')
```

```
## The ICC of class is:  0.2578862
```

3.2 Effect of Pre-test Scores

To understand the impact of baseline knowledge, we add the pre-test scores to our model and observe how the intervention effect changes. (Add Pretest to the Empty Model can significantly reduce `AIC`, `BIC`. And the `Pr(>Chisq)` is quite low that can reject the Null Hypothesis.

```r
# summary(Model.0)
# --------------------------------------------------------
# REsummary <- as.data.frame(VarCorr(Model.0))
# REsummary
# summary(Model.0)$varcor
# --------------------------------------------------------
Model.1 = lmer(Posttest ~ Pretest + (1 | School),
               data = CRT)
# summary(Model.1)
# --------------------------------------------------------
ranova(Model.1)
```

```
## ANOVA-like table for random-effects: Single term deletions
##
## Model:
## Posttest ~ Pretest + (1 | School)
##                npar  logLik    AIC    LRT Df Pr(>Chisq)
## <none>            4 -733.93 1475.9
## (1 | School)      3 -767.00 1540.0 66.121  1  4.241e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
anova(Model.1, Model.0)
```

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

## Data: CRT
## Models:
## Model.0: Posttest ~ 1 + (1 | School)
## Model.1: Posttest ~ Pretest + (1 | School)
##         npar    AIC    BIC  logLik deviance Chisq Df Pr(>Chisq)
## Model.0    3 1544.9 1555.6 -769.44   1538.9
## Model.1    4 1475.6 1489.8 -733.78   1467.6 71.33  1  < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
# Model 1 is better than Model 0
```

But the random effects of Pretest in School level seems not good. We can't add it to the Random effects.

```r
# Model 2 is not better than model 1
Model.2 = lmer(Posttest ~ Pretest + (Pretest | School),
               data = CRT)
ranova(Model.2)
```

```
## ANOVA-like table for random-effects: Single term deletions
##
## Model:
## Posttest ~ Pretest + (Pretest | School)
##                                  npar  logLik    AIC     LRT Df Pr(>Chisq)
## <none>                              6 -733.77 1479.5
## Pretest in (Pretest | School)       4 -733.93 1475.9 0.33308  2     0.8466
```

```r
anova(Model.2, Model.1)
```

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

## Data: CRT
## Models:
## Model.1: Posttest ~ Pretest + (1 | School)
## Model.2: Posttest ~ Pretest + (Pretest | School)
##         npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
## Model.1    4 1475.6 1489.8 -733.78   1467.6
## Model.2    6 1479.4 1500.7 -733.68   1467.4 0.1946  2     0.9073
```

3.3 Role of FSM

Next, we explore the role of free school meal eligibility (FSM) by including it in our model. This helps us understand if socio-economic status influences the intervention effect.

```r
# Model 3 (add FSM to Fix Effects) is not better than Model 1
Model.3 = lmer(Posttest ~ Pretest + FSM + (1 | School),
               data = CRT)
ranova(Model.3)
```

```
## ANOVA-like table for random-effects: Single term deletions
##
## Model:
## Posttest ~ Pretest + FSM + (1 | School)
##              npar  logLik    AIC    LRT Df Pr(>Chisq)
## <none>          5 -732.03 1474.1
## (1 | School)    4 -764.90 1537.8 65.739  1  5.147e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
anova(Model.3, Model.1)
```

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

## Data: CRT
## Models:
## Model.1: Posttest ~ Pretest + (1 | School)
## Model.3: Posttest ~ Pretest + FSM + (1 | School)
##         npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
## Model.1    4 1475.6 1489.8 -733.78   1467.6
## Model.3    5 1475.9 1493.7 -732.95   1465.9 1.6475  1     0.1993
```

```r
# Model 4 (add FSM to Random Effects) is not better than Model 1
Model.4 = lmer(Posttest ~ Pretest + (FSM | School),
               data = CRT)
```

```
## boundary (singular) fit: see help('isSingular')
```

```r
ranova(Model.4)
```

```
## ANOVA-like table for random-effects: Single term deletions
##
## Model:
## Posttest ~ Pretest + (FSM | School)
##                       npar  logLik    AIC     LRT Df Pr(>Chisq)
## <none>                   6 -733.83 1479.7
## FSM in (FSM | School)    4 -733.93 1475.9 0.20514  2     0.9025
```

```r
anova(Model.4, Model.1)
```

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

```
## Data: CRT
## Models:
## Model.1: Posttest ~ Pretest + (1 | School)
## Model.4: Posttest ~ Pretest + (FSM | School)
##         npar    AIC    BIC  logLik deviance Chisq Df Pr(>Chisq)
## Model.1    4 1475.6 1489.8 -733.78   1467.6
## Model.4    6 1479.3 1500.7 -733.67   1467.3 0.211  2     0.8999
```

From the output of `Model.3` and `Model.4`, when we add FSM to the fixed / random effects, neither of them can improve the model performance. Both of their `Pr(>Chisq)` are bigger than 0.05.

3.4 Intervention Effect

Finally, we introduce the intervention variable to assess its effect. We fit a model including the intervention and inspect the coefficient and confidence interval for significance.

```r
Model.5 = lmer(Posttest ~ Pretest + Intervention + (1 | School),
               data = CRT)
ranova(Model.5)
```

```
## ANOVA-like table for random-effects: Single term deletions
##
## Model:
## Posttest ~ Pretest + Intervention + (1 | School)
##              npar  logLik    AIC    LRT Df Pr(>Chisq)
## <none>          5 -730.06 1470.1
## (1 | School)    4 -752.58 1513.2 45.041  1   1.93e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
anova(Model.5, Model.1)
```

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

```
## Data: CRT
## Models:
## Model.1: Posttest ~ Pretest + (1 | School)
## Model.5: Posttest ~ Pretest + Intervention + (1 | School)
##         npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
## Model.1    4 1475.6 1489.8 -733.78   1467.6
## Model.5    5 1471.6 1489.4 -730.81   1461.6 5.9407  1     0.0148 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Adding Intervention to random effects cannot reject the zero hypothesis, so it is not appropriate.

```
Model.6 = lmer(Posttest ~ Pretest + Intervention + (Intervention | School),
               data = CRT)
```

```
## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv, :
## unable to evaluate scaled gradient
```

```
## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv, :
## Model failed to converge: degenerate Hessian with 1 negative eigenvalues
```

```
## Warning: Model failed to converge with 1 negative eigenvalue: -4.2e-05
```

```
ranova(Model.6)
```

```
## ANOVA-like table for random-effects: Single term deletions
##
## Model:
## Posttest ~ Pretest + Intervention + (Intervention | School)
##                                         npar  logLik    AIC      LRT Df
## <none>                                     7 -730.03 1474.1
## Intervention in (Intervention | School)    5 -730.06 1470.1 0.062275  2
##                                         Pr(>Chisq)
## <none>
## Intervention in (Intervention | School)     0.9693
```

```
anova(Model.6, Model.5)
```

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

```
## Data: CRT
## Models:
## Model.5: Posttest ~ Pretest + Intervention + (1 | School)
## Model.6: Posttest ~ Pretest + Intervention + (Intervention | School)
##         npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
## Model.5    5 1471.6 1489.4 -730.81   1461.6
## Model.6    7 1475.6 1500.5 -730.79   1461.6 0.0429  2     0.9788
```

Part 4: Discussion of results

So far the best model is Model.5, which mathematical form is:

$$Y_{ijk} = \beta_0 + \beta_1 \text{Pretest}_{ijk} + \beta_2 \text{Intervention}_{ijk} + u_j + \varepsilon_{ijk}$$

Where:

- Y_{ijk} is the post-test score for the i-th pupil in the j-th school.
- β_0 is the intercept, representing the expected post-test score for a pupil with a pre-test score of 0 in the control group.
- β_1 is the coefficient for the pre-test scores, indicating the expected change in post-test score for each one-unit increase in the pre-test score, holding all other variables constant.
- β_2 is the coefficient for the intervention, representing the expected difference in post-test scores between the treatment and control groups, controlling for pre-test scores.
- u_j is the random effect for the j-th school, capturing the variation in post-test scores across schools that is not explained by the pre-test scores or intervention status.
- ε_{ijk} is the residual error term for the i-th pupil in the j-th school, capturing the individual-level variation in post-test scores not explained by the model.

In this model, the fixed effects are the intercept, pre-test scores, and intervention status, while the random effect is the variation at the school level, indicated by u_j. The model assumes normal distributions for both the random effects and the residual errors.

$$u_j \sim N(0, \sigma)$$

4.1 Model Summary

```r
summary(Model.5)
```

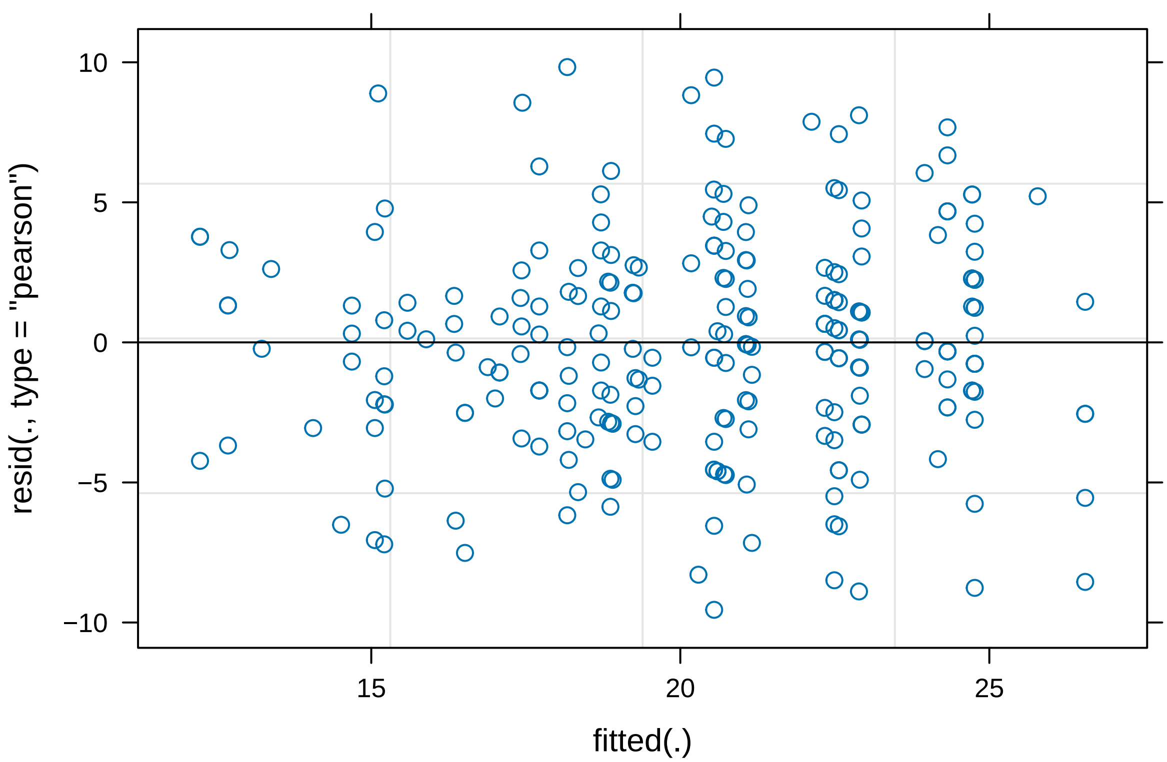
```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: Posttest ~ Pretest + Intervention + (1 | School)
##    Data: CRT
##
## REML criterion at convergence: 1460.1
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.49679 -0.62157  0.01219  0.59590  2.57030
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  School   (Intercept)  4.952   2.225
##  Residual             14.623   3.824
## Number of obs: 260, groups:  School, 18
##
## Fixed effects:
##              Estimate Std. Error       df t value Pr(>|t|)
## (Intercept)   11.4244     1.1186  56.2982  10.213 1.99e-14 ***
```

```
## Pretest         1.8299     0.2013 247.7522   9.091  < 2e-16 ***
## Intervention    3.0007     1.1927  15.7822   2.516   0.0231 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr) Pretst
## Pretest     -0.683
## Interventin -0.501  0.002
```

4.2 Model Diagnostics and Visualization

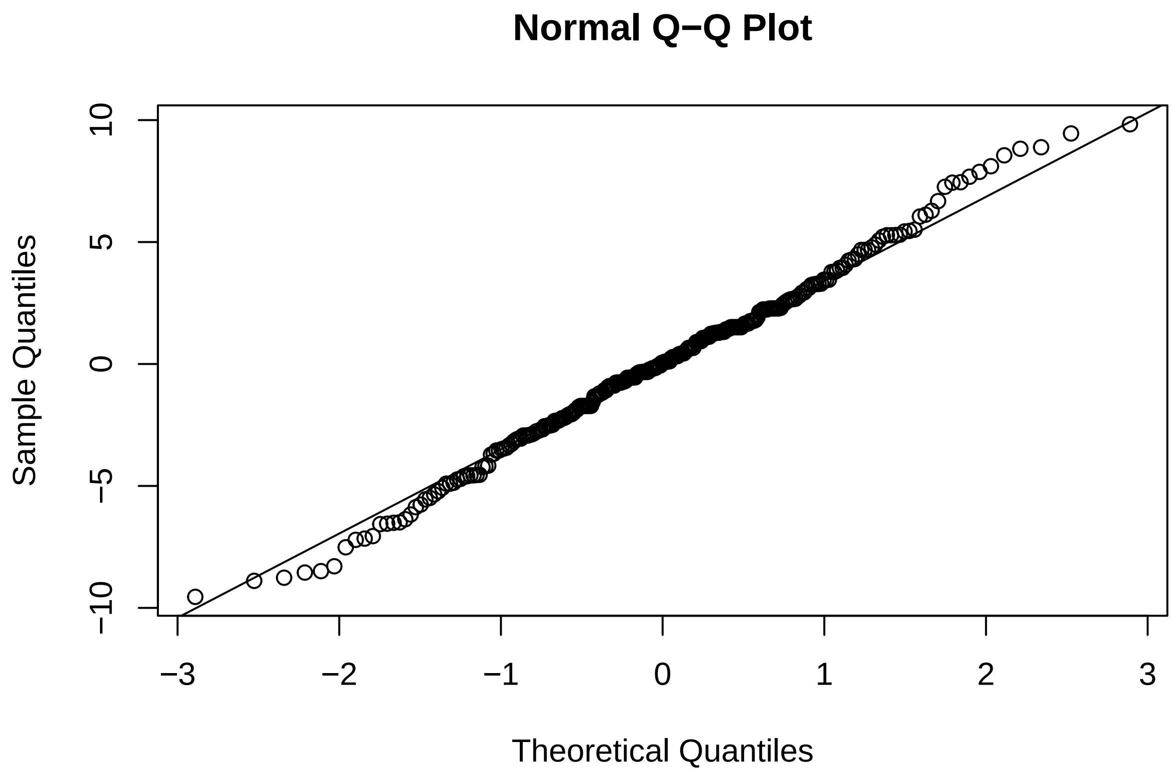
4.2.1 Plot the Residuals

```r
plot(Model.5)
```



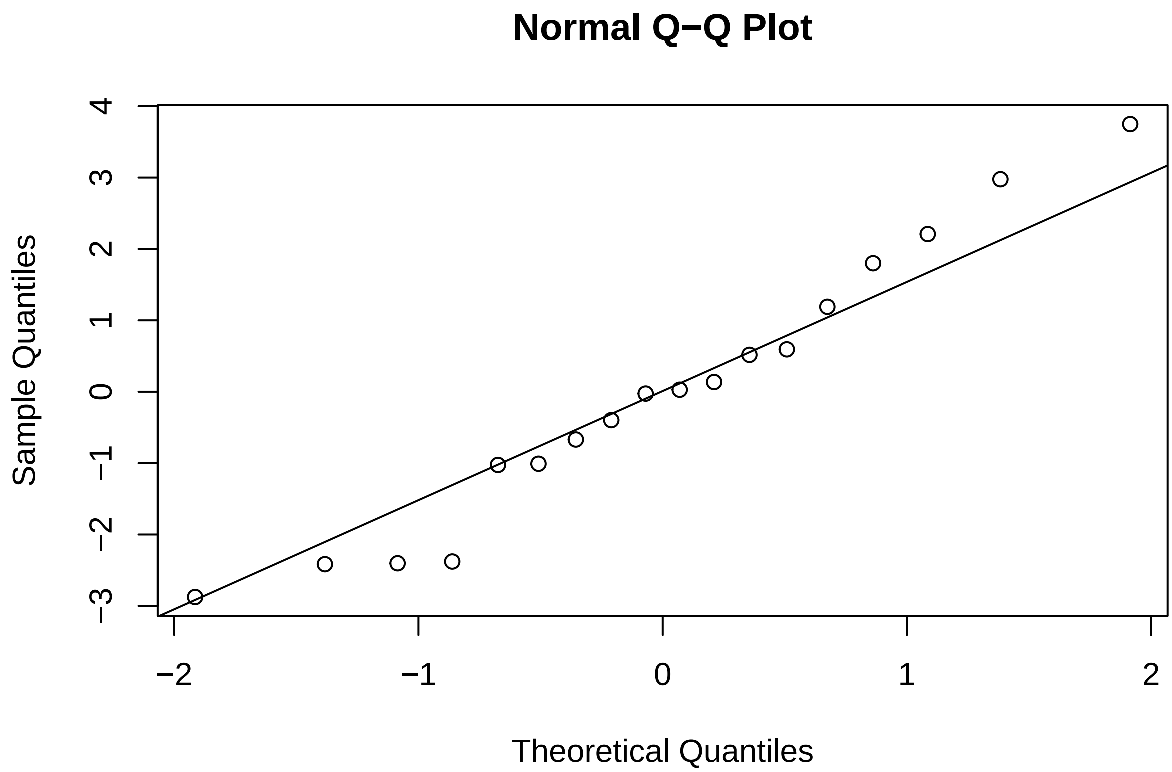
4.2.2 Q-Q plot of Residuals

```r
qqnorm(resid(Model.5))
qqline(resid(Model.5))
```



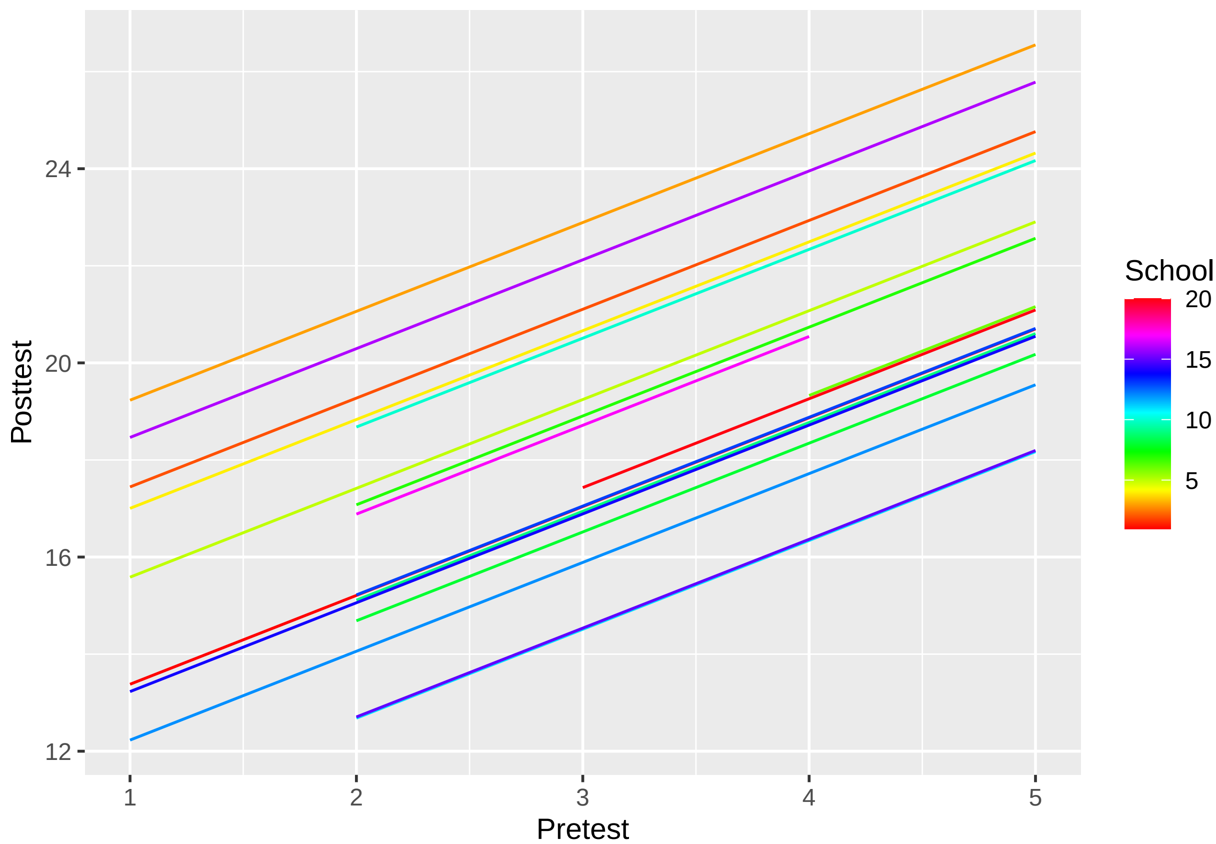
4.2.3 Q-Q plot of Random Effects

```
# We only have one random effect
# It is the intercept of School level
qqnorm(ranef(Model.5)[[1]][,1])
qqline(ranef(Model.5)[[1]][,1])
```



From `Part 4.2.3` and `Part 4.2.4` we can say the random effects obey the Normal Distribution assumption and the Model fits well.

```
CRT$Posttest.pred = predict(Model.5)
ggplot(CRT,
       aes(
         x = Pretest,
         y = Posttest,
         col = School,
         group = School
       )) +
  geom_line(aes(y = Posttest.pred)) +
  scale_color_gradientn(colours = rainbow(100))
```



References

Evaluating Intervention Programs with a Pretest-Posttest Design: A Structural Equation Modeling Approach

Word count

```
# install.packages("devtools")
# devtools::install_github("benmarwick/wordcountaddin",
#                          type = "source", dependencies = TRUE)
require(wordcountaddin)
word_count()
```

```
## [1] 1356
```

```
text_stats()
```

Method	koRpus	stringi
Word count	1356	1218
Character count	8584	8646
Sentence count	110	Not available
Reading time	6.8 minutes	6.1 minutes