# To Classical Music Or Not To Classical Music

A Statistical Analysis Using Regression Techniques In Mixed Effects Modeling To Analyze Factors Influencing Quantitative Classification Of Classical Music

Arsh Gupta

Department of Statistics and Data Science, Carnegie Mellon University

arshg@andrew.cmu.edu

## Contents



## 1 Abstract

The main goal of this study is to analyze the influence of Instrument, Harmonic Motion, and Voice Leading on individuals' classification of music as "classical" or "popular". This was accomplished by collecting data across 24 categories from participants at the University of Pittsburgh that were evaluated on a variety of

factors. The statistical modeling employed revolves around linear regression techniques in mixed level models by treating each participant as a separate group. Our final optimal model is successful in answering the research questions posed at the beginning of the study and either providing enough evidence to confirm the preliminary hypotheses or lack of to reject the same. An analysis of our results confirms that `Instrument` has the strongest effect on `Classical` ratings and that participants that rate themselves highly on the scale of their musician ability are likely to identify classical music differently than those that do not.

## 2 Introduction

There is a wide diversity in the way that individuals interpret and comprehend various genres of music. Interpretation or classification of classical music, to some degree, remains subjective school of thought even among the community of musical scholars. One such scholar, Ivan Jimenez, a composer and musicologist visiting the University of Pittsburgh, sought to gain some tangible insight into this exact topic and understand what influences a listeners' identification of music as "classical" or "popular". His experiment was primarily concerned with the effect of three main design factors: `Instrument, Harmonic Motion,` and `Voice Leading`.

In collecting the data and undertaking this experiment, Jimenez and his student, Vincent Rossi outlined two main research questions that they hoped to address:

1. What experimental factor, or combination of factors, has the strongest influence on ratings?
2. Are there differences in the way that musicians and non-musicians identify classical music?

Jimenez and Rossi also formulated some key hypotheses, which were:

1. Instrument shoud have the largest influence on rating.
2. One particular harmonic progression, I-V-VI, might be frequently rates as classical, because it is the beginning progression for Pachelbel's Canon in D, which many people have heard. On the other hand, it is also a very common chord progression in popular music of the past 20 years or so.
3. Based on previous research, contrary motion would also be frequently rated as classical.

# 3 Data

The data for this study was collected by Ivan Jimenez, a composer and musicologist visiting the University of Pittsburgh, and his student, Vincent Rossi through a designed experiment intended to measure the influence of instrument, harmonic motion, and voice leading on listeners' identification of music as "classical" or "popular". The researchers presented 36 musical stimuli to 70 listeners, recruited from the population of undergraduates at the University of Pittsburgh, and asked them to rate the music on two different scales:

- How classical does the music sound (1 to 10, 1 = Not at all, 10 = Very classical sounding)
- How popular does the music sound (1 to 10, 1 = Not at all, 10 = Very popular sounding)

Listeners were told that a piece could be rated as both classical and popular, neither classical nor popular, or mostly classical and not popular (or vice versa), so that the scales should have functioned more or less independently.

The data set contains the following variables:

- `Classical`: How classical does the stimulus sound
- `Popular`: How popular does the stimulus sound
- `Subject`: Unique subject ID
- `Harmony`: Harmonic Motion (4 levels)
- `Instrument`: Instrument (3 levels)
- `Voice`: Voice Leading (3 levels)
- `Selfdeclare`: Are you a musician? (1-6, 1 = Not at all)
- `OMSI`: Score on a test of musical knowledge
- `X16.minus.17`: Auxiliary measure of listener's ability to distinguish classical vs popular music
- `ConsInstr`: How much did you concentrate on the instrument while listening? (0-5, 0 = Not at all)
- `ConsNotes`: How much did you concentrate on the notes while listening? (0-5, 0 = Not at all)
- `Instr.minus.Notes`: Difference between previous two variables

- `PachListen`: How familiar are you with Pachelbel's Canon in D? (0-5, 0 = Not at all)
- `ClsListen`: How much do you listen to classical music? (0-5, 0 = Not at all)
- `KnowRob`: Have you heard Rob Paravonian's Pachbelbel Rant (0-5, 0 = Not at all)
- `KnowAxis`: Have you heard Axis of Evil's Comedy bit on the 4 Pachelbel chords in popular music? (0-5, 0 = Not at all)
- `X1990s2000s`: How much do you listen to pop and rock from the 90's and 2000's? (0-5, 0 = Not at all)
- `X1990s2000s.minus.1960s1970s`: Difference between previous variable and a similar variable referring to 60's and 70's pop and rock
- `CollegeMusic`: Have you taken music classes in college? (0 = No, 1 = Yes)
- `NoClass`: How many music classes have you taken?
- `APTheory`: Did you take AP Music Theory class in High School? (0 = No, 1 = Yes)
- `Composing`: Have you done any music composing? (0-5, 0 = Not at all)
- `PianoPlay`: Do you play piano? (0-5, 0 = Not at all)
- `GuitarPlay`: Do you play guitar? (0-5, 0 = Not at all)
- `X1stInstr`: How proficient are you at your first musical instrument? (0-5, 0 = Not at all)
- `X2ndInstr`: How proficient are you at your second musical instrument? (0-5, 0 = Not at all)

The 36 stimuli were chosen by completely crossing these factors:

- **Instrument:** String Quarter, Piano, Electric Guitar
- **Harmonic Motion:** I-V-vi, I-VI-V, I-V-IV, IV-I-V
- **Voice Leading:** Contrary Motion, Parallel 3rds, Parallel 5ths

The data set contains a total of 2520 observations and 27 features (parameters) with two independent response variables `Classical` and `Popular`, which are ratings on a scale of 1 to 10. We find that there are two observations that have values for `Classical` and `Popular` as 19 respecively, which is outside the range of acceptable values. This is likely due to human error where the participant might have been intending to enter 9 instead of 19.

We ignore the variables `X` and `first12` from the data set for the purposes of this analysis since they do not provide us with any meaningful information. Thus, we obtain a total of 22 possible predictors which will be considered in building the optimal model.

The primarily variables of interest include: `Subject, Harmony, Instrument, Voice, Selfdelare, OMSI, X16.minus.X17, ConsInstr, ConsNotes, Inst.minus.Notes, ClsListen, KnowRob, KnowAxis, X1990s2000s, X1990s200s.minus.1960s1970s, CollegeMusic, NoClass, APTheory, Composing, PianoPlay, GuitarPlay, X1stInstr, X2ndInstr`.

Most of the covariates above are factor variables with six levels so we do not apply any transformations to the data in order to preserve that. Additionally, there does not seem to be any strong skewness within the data set.

# 4 Methods

## 4.1 Initial steps

The statistical framework of the analysis in this study is primarily concerned with linear modeling around multivariate regression techniques and mixed effects models.

As a preliminary starting point, we create a basic ordinary least squares linear regression model to analyze the influence of the three main design variables `Instrument, Harmony,` and `Voice` on `Classical` ratings. At various stages along the analysis, we adopt Analysis of Variance (ANOVA) tests to evaluate model performance both by itself and in comparison to other models.

## 4.2 Incorporating group level effects

As the next step, we transform the OLS model into a multi-level/mixed effects model by treating every participant (aka `Subject`) as a separate group since we have 36 observations for each participant. We then introduce a different intercept for each group/participant (`Subject`). The performance of this mixed effects model is assessed on multiple factors, including conditional and marginal residual plots, normality of the standardized residuals and standardized random effects, and fixed and random effect variances $\hat{\tau}_j^2$ and $\hat{\sigma}^2$.

At various stages in the analysis, we compare multiple mixed effects model in order to evaluate the marginal effect of the one fixed or random effects covariate in model performance. We do this by comparing the Akaike Information Criterion (AIC), Bayesian Information Criterion (BIC), and Deviance Information Criteria (DIC). These have been briefly elaborated below:

### Akaike Information Criterion (AIC)

AIC is a relative measure of the quality of a model for a given set of data and helps in model selection among a finite set of models. It uses the maximized likelihood estimate and the number of parameters to estimate the information lost in the model. The AIC measure gives a trade-off between the model accuracy and model complexity thus preventing overfitting.

The formula to define AIC is given as follows:

$$AIC = -2 \cdot \ln(\hat{L}) + 2K$$

where

- $K$: Number of estimated parameters
- $\hat{L}$: Maximized value of the log-likelihood function of the model

### Bayesian Information Criterion (BIC)

Similar to AIC, BIC is also a criterion for model selection. It is closely related to AIC in using the likelihood function. The way it differs to AIC is by introducing a penalty term for the number of parameters in the model since it is possible to increase the likelihood by adding parameters which also results in overfitting.

The formula for BIC is defined as follows:

$$BIC = -2 \cdot \ln(\hat{L}) + \ln(K)$$

where

- $K$: Number of estimated parameters
- $\hat{L}$: Maximized value of the log-likelihood function of the model

**Deviance Information Criteria (DIC)**

DIC is a similar criterion to AIC used for model selection in Bayesian inference and multi-level modeling where the degrees of freedom (or the number of parameters) is not fixed. It considers the effective degrees of freedom.

The formula for DIC is defined as follows:

$$DIC = -2(\hat{L}) + 2k_{eff}$$

where

- $k_{eff}$: Effective degrees of freedom, estimated from the curvature of the likelihood, which is driven by the size of the $\tau^2$'s
- $\hat{L}$: Log-likelihood based on the marginal model $f\left(\underline{Y}|\underline{\beta}, \underline{\omega}, \sigma^2\right)$

Lower AIC and BIC values indicates less information lost hence a better model. Though these two measures are derived from a different perspective, they are closely related. Apparently, the only difference is, BIC considers the number of observations in the formula, which AIC does not. Though BIC is always higher than AIC, lower the value of this two measure, better the model.

All of AIC, BIC, and DIC are helpful statistical tools when performing variable selection. When evaluating the marginal effect of a fixed level covariate in model performance, we primarily look at AIC and BIC, and when evaluating the effect of a random effect covariate, we compare the DIC.

## 4.3 Assessing effect of other predictors

Once we've incorporated the three main design factors `Instrument, Harmony,` and `Voice` in our model as the appropriate fixed and random effects covariates, we evaluate if any of the remaining 19 predictors would be appropriate to include as fixed or random effects covariates. First, we check which of them would improve our model by adding them as fixed effects covariates. We do this by performing `regsubsets()` (an R function used for model/variable selection using the exhausitive search, forward selection, or backward selection methods) on the fixed effects part of our optimal multi-level `lmer()` model by treating it as a separate OLS `lm()` model. After performing `regsubsets()` and comparing AIC/BIC, we add the covariates that minimize AIC and BIC as fixed effects covariates in the mixed effects model.

With the fixed effects now fixed, we evaluate if there should be any changes made to the random effects part of our model. We do this by implementing an idea similar to forward selection while minimizing DIC. Taking a base model with no new random effects covariates added, we create new models each containing one new random effects covariate from the fixed level covariates added at the previous step and compare DIC with the base level model.

## 4.4 Answering the research questions

The first research question is concerned with analying which combination of factor(s) has the strongest influence on `Classical` rating scores, which has can be addressed by comparing the coefficient estimates, standard error values, and t-statistic values (also referred to as t-values) of our final mixed effects model generated using the methodology outlined above.

The second research question focuses on whether there are any differences in the way that musicians and non-musicians identify classical music. This has been done by dichotomizing the variable `Selfdeclare`, which is a factor variable wherein the participants ranked themselves on a scale of 1 to 6 whether they are a musician or not (1 being not at all and 6 being an adept musician). The process for this has been described in greater detail in the following sub-section titled "Dichotomization of `Selfdeclare`".

## 4.5 Dichotomization of `Selfdeclare`

Through some vanilla exploratory data anlaysis, we find that about 60% of the survey participants have selected a value of 1 or 2 for `Selfdeclare`, and the remaining 40% have selected $3, 4$, or 5. Thus, in order to address the second research question, we dichotomize that predictor by creating a new variable called `self_declared_musician` where:

- `self_declared_musician = yes`, if `Selfdeclare` $\in \{1, 2\}$
- `self_declared_musician = no`, if `Selfdeclare` $\in \{3, 4, 5\}$

We then replace the variable `Selfdeclare` with as the fixed level covariate in our optimal mixed effects model for the purpose of this exercise.

Then, in order to see which interactions with `self_declared_musician` on the fixed level are useful, we create new models where each of them contains an interaction with `self_declared_musician` and one of the other fixed level covariates, and compare the AIC and BIC with the base level model which has no interactions with `self_declared_musician`. If any model containing interaction with `self_declared_musician` has a lower AIC/BIC than the base-level model, then it would suggest that the dichotomized musician variable is sensitive to interaction with that other fixed level covariate.

# 5 Results

We arrive at the optimal model that best addresses our research questions and captures the fixed and random effects of all significant covariates in predicting the `Classical` ratings as follows:

## 5.1 Final Model

We get the final model as follows (in R format):

```
Classical ~ Instrument + Harmony + Voice + Harmony:Voice + Selfdeclare + KnowAxis +
X1990s2000s.minus.1960s1970s + X2ndInstr + (1 + Instrument + Harmony + KnowAxis | Subject)
```

We attempt add the variable `GuitarPlay` after our analyses with variable selection using `regsubsets()`, ANOVA tests, and comparing AIC/BIC/DIC, but we find that R automatically drops it due to the fixed-effect model matrix being rank deficient.

Figure 1 shows the fixed and random effects variances for the different covariates.

| Random Effects | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| **Groups** | **Name** | **Variance** | **Std. Dev.** | **Correlation** | | | | |
| Subject | **(Intercept)** | 0.209 | 0.457 | | | | | |
| | **Instrumentpiano** | 2.72 x 1e-3 | 5.21 x 1e-2 | 0.24 | | | | |
| | **Instrumentstring** | 2.214 | 1.488 | -1 | -0.23 | | | |
| | **HarmonyI-V-IV** | 0.164 | 0.405 | -0.61 | 0.62 | 0.61 | | |
| | **HarmonyI-V-VI** | 1.88 | 1.371 | 0.7 | -0.53 | -0.7 | -0.99 | |
| | **HarmonyIV-I-V** | 0.552 | 0.743 | -0.63 | 0.6 | 0.64 | 1 | -1 |
| | **KnowAxis5** | 6.99 x 1e-4 | 2.64 x 1e-2 | -1 | -0.28 | 1 | 0.57 | -0.67 |
| **Residual** | 2.46 | 1.57 | | | | | | |

Figure 1: Fixed and Random Effects Variances from `summary()` output in R

Figure 2 shows the AIC, BIC, and DIC for four key models across the entire process of arriving at our final optimal model.

| | AIC | BIC | DIC |
|---|---|---|---|
| Best Linear Model | 11226.94 | 11314.26 | 11226.94 |
| MLM (random intercept) | 10458.10 | 10551.24 | 10426.10 |
| MLM (random intercept and slopes) | 9937.27 | 10146.84 | 9865.27 |
| Best Multilevel Model | 1178.30 | 1360.04 | 1078.30 |

Figure 2: AIC, BIC, DIC for different models

## 5.2 Research Questions

We have that for our final model

Upon analyzing the effect of individual harmonic motion levels on the `Classical` ratings, we see that the covariate `HarmonicI-V-VI` has the largest coefficient estimate (2.47) and t-value (3.71) compared to the other two levels of Harmonic Motion (0.53 and 1.10 for `HarmonicI-V-IV`, and 0.58 and 1.10 for `HarmonicIV-I-V`).

Among the fixed level covariates for factor variable Voice Leading, we find that the level `Voicepar3rd` has a coefficient estimate $-0.155$ and t-value $-0.338$, and the level `Voicepar5th` has a coefficient estimate 0.478 and t-value 1.034.

We find that the models containing interaction terms `self_declared_musician:Instrument` and `self_declared_musician:Harmony` have a lower AIC (1176.912 and 1176.076 respectively) than the model not containing any interaction terms with `self_declared_musician` and the other fixed level covariates.

Out of these two, the model containing the interaction term `self_declared_musician:Instrument` has a lower BIC (1365.921).

# 6 Discussion

## 6.1 Technical Machinery and Methodology

As a preliminary step, we create an OLS linear model evaluating the effect on `Classical` scores using predictors `Instrument`, `Harmony`, and `Voice`. We evaluate evidence of any significant interactions up to the third order between these three covariates and keep `Harmony:Voice` in the model after running an ANOVA test. Since we have 36 ratings from each participant (`Subject`), we expand our previous linear model by including a random intercept for each participant (`Subject`), leaving us with what is called a repeated measures model that performs better than the preliminary OLS model.

In an attempt to account for participants' personal biases across the type of instrument, harmony, or voice, it might be helpful to include random effect covariates for `Instrument`, `Harmony`, and/or `Voice`. We test this by creating models with different combinations of these random effects and comparing their DICs, which suggests that random effect covariates for `Instrument` and `Harmony` are significant. Thus, we add them to our model.

So far, our model includes the fixed effects covariates `Instrument`, `Harmony`, `Voice`, and `Harmony:Voice`, and the random effects covariates `Instrument` and `Harmony` in addition to a random intercept for each group. We then check which of the remaining person covariates would improve our model by adding them as fixed effects covariates and find that predictors `Selfdeclare`, `KnowAxis`, `X1990s2000s.minus.1960s1970s`, and `X2ndInstr` improve our model by minimizing AIC/BIC. Thus, we add them as fixed effects covariates in our mixed effects model. With the fixed effects now fixed, we evaluate if there should be any changes made to the random effects part of our model and find that the covariate `KnowAxis` should be added to the random effects part.

## 6.2 Final Result

We have the final model as follows:

## Level 1

$$\text{Classical}_i = \alpha_{0j[i]} + \alpha_{1j[i]} \cdot \text{Instrument}_i + \alpha_{2j[i]} \cdot \text{Harmony}_i + \alpha_{3j[i]} \cdot \text{KnowAxis}_{j[i]} + \alpha_4 \cdot \text{Voice}_i +$$

$$\alpha_5 \cdot (\text{Harmony}_i \times \text{Voice}_i) + \epsilon_i$$

## Level 2

$$\alpha_{0j[i]} = \beta_{00} + \beta_{01} \cdot \text{SelfDeclare}_{j[i]} + \beta_{02} \cdot \text{X1990s2000s.minus.1960s1970s}_{j[i]} + \beta_{04} \cdot \text{X2ndInstr}_{j[i]} + \eta_{0j[i]}$$

$$\alpha_{1j[i]} = \beta_{10} + \eta_{1j[i]}$$

$$\alpha_{2j[i]} = \beta_{20} + \eta_{2j[i]}$$

$$\alpha_{3j[i]} = \beta_{30} + \eta_{3j[i]}$$

where

**Fixed Effects Covariates**

The following variables are treated by the same way across each participant without any bias, so there is a constant slope for each of them across every participant.

- `Instrument`: The type of instrument played can have a distinct effect in a listener's ability to classify a piece of music as classical or not.
- `Harmony`: The different levels of harmonic progression can influence the listener's ability to classify music as classical or not.
- `Voice`: The leading voice has an effect on the listener's ability to classify a piece of music as classical or not.
- `Harmony:Voice`: This interaction term suggests as the value of either of the variables `Harmony` or `Voice` changes, it leads to a subsequent change in the value of the other variable.
- `Selfdeclare`: The more likely a person is to consider themselves a musician, the less likely they are to classify a piece of music as classical keeping everything else constant.

- `KnowAxis`: If a listener is familiar with Axis of Evil's Comedy bit on the 4 Pachelbel chords in popular music, they are more likely to rate a piece of music as classical keeping other factors constant.

- `X1990s2000s.minus.1960s1970s`: Keeping every other factor constant, if a listener listens to more pop and rock music from the 1960s and 1970s than from the 1990s and 2000s, they are more likely to classify music as classical.

- `X2ndInstr`: Keeping other factors as constant, an increased proficiency in playing their second music instrument makes the listener less likely to rate a piece of music as classical.

### Random Effects Covariates

Different participants might interpret these variables differently, or there could be an inherent bias in the way they respond to these, so we kept a different slope for every participant for the following variables.

- `Instrument`
- `Harmony`
- `KnowAxis`

### Individual Level Covariates

These variables have different values for every observation across the same participant.

- `Instrument`
- `Harmony`
- `Voice`
- `Harmony:Voice`

### Group Level Covariates

These variables have the same values across a single participant (group).

- `Selfdeclare`
- `KnowAxis`
- `X1990s2000s.minus.1960s1970s`
- `X2ndInstr`

## 6.3 Research Question 1

The researchers hypothesize that `Instrument` exerts the strongest influence among the three design factors (`Instrument, Harmonic Motion, Voice Leading`), which is confirmed by the fixed effect coefficient estimate and t-values for `Instrument` being the largest relative to the other two design factors. Hence, it is confirmed that `Instrument` does exert the strongest influence among the three design factors.

Based off the high coefficient estimate and t-value for I-V-VI among the three different Harmonic Levels, we have evidence to support the fact that among the three different Harmonic Motion levels, I-V-VI has the strongest association with `Classical` ratings.

Similarly, upon comparing the coefficient estimates and t-values, we find that `Voicepar5th` has a larger coefficient and t-value, confirming the hypothesis that Parallel 5ths have the strongest association with `Classical` ratings.

## 6.4 Research Question 2

The second research question is focused around whether there are any differences in the way that musicians and non-musicians identify classical music. We have attempted to address this by dichotomizing the covariate `Selfdeclare` by re-introducing that as a new variable `self_declared_musician`, as described in the Methods section of this paper. As a refresher to the reader:

- `self_declared_musician = yes`, if `Selfdeclare` $\in \{1, 2\}$
- `self_declared_musician = no`, if `Selfdeclare` $\in \{3, 4, 5\}$

Based on the results of the above dichotimization, we find evidence suggesting that `Instrument` and `Harmony` are sensitive to the dichotomization of `self_declared_musician` with `Instrument` being the more sensitive covariate out of the two.

What this means is that the type of instrument that an individual plays could be influential towards whether they consider themselves as a musician or not. This points towards statistically significant evidence that there are differences in the way that musicians and non-musicians identify classical music. Primarily, the former's classification of classical music is influenced by the type of instrument present in a particular musical piece, whereas this is not really the case for the latter.

## 6.5 Key Takeaways

We have some key takeaways over here. Firstly, the presence of an interaction term `Harmony:Voice` as a fixed effects covariate suggests that as the value for one of those predictors (`Harmony` or `Voice`) changes, there is a subsequent change on the other predictor. Additionally, we see that there are various variables on the fixed level that influence an individual's rating of a stimuli's classical sound, such as the instrument used, harmonic progression, leading voice, how good of a musician the individual considers themselves to be, whether they have heard of Axis of Evil's Comedy bit on the 4 Pachelbel chords in popular music or not, difference in the amount of pop and rock music they listen to from the 1990s and 2000s than from the 1960s and 1970s, and proficiency in their second musical instrument played. These results are not very surprising it would be reasonable for them to vary across each participant.

As random effects covariates, we see that `Instrument`, `Harmony`, and `KnowAxis` influence an individual's rating of a stimuli's classical sound. This points towards a possible personal bias varying with the type of instrument and harmony, wherein people might vary in the degree of what they would call music played by a certain instrument or at a specific harmony classical or not, which is why we treat them on the group level. `KnowAxis` is also an indicator variable since is has only two levels, so the value for that would vary across each participant and serve as a group level covariate too. We see that the standard errors for all fixed level coefficients is less than 1, suggesting that the final model is robust and fits the data well.

Upon testing out the researchers' hypothesis if people who self-identify as musicians may be influenced by things that do not influence non-musicians, we find that there to be evidence in support of that. Specifically, we find that the type of instrument that an individual plays could be influential towards whether they consider themselves as a musician or not. We do this by dichotomizing the fixed effects covariate `Selfdeclare` and considering any significant interactions with other fixed level predictors by minimizing AIC/BIC. We find that the interaction of this new dichotomized variable is sensitive to the covariate `Instrument`.

## 6.6 Limitations and Future Research

One of the biggest limitations of this analysis is that it ignores the qualitative aspect of the musical phenonmenon which quantification cannot fully capture. Additionally, almost every variable that the data is collected on is a factor level variable, meaning that its individual levels are treated as discrete levels in the statistical analysis that compromises on interpretability to some degree. One way to address this could be to include more continuous predictors that measure data along a spectrum or range as opposed to discrete levels.

While the study does a good job of accounting for personal biases across the participants by including group level intercepts, and the way each person might be prone to interpreting a certain type of musical piece as being classical or not by introducing relevant random effects, there is still a possibility of hidden variables at play that might be causing some type of bias to lurk in the final results.

A final limitation of this study is that the final results and findings are volatile to the methodology and approach adopted by a researcher. There is no guarantee that another researcher conducting a similar study and evaluating the same research questions might arrive at the same optimal model or results that this paper does.

One way this could be addressed by future developments on this study is by considering more robust mechanisms that could generate standardized and reproducible results, possibly within the realm of non-linear predictive modelling.

# 7 References

1. "Deviance Information Criterion." Wikipedia, Wikimedia Foundation, 21 Dec. 2021, https://en.wikipedia.org/wiki/Deviance_information_criterion#.

2. Datalab, Analyttica. "Akaike Information Criterion(AIC)." Medium, Medium, 7 Jan. 2019, https://medium.com/@analyttica/akaike-information-criterion-aic-7a4b58bce206.

3. Datalab, Analyttica. "What Is Bayesian Information Criterion (BIC)?" Medium, Medium, 16 Jan. 2019, https://medium.com/@analyttica/what-is-bayesian-information-criterion-bic-b3396a894be6.

4. Junker, Brian. "Multilevel Models - The Basics." Applied Linear Models. Carnegie Mellon University. Pittsburgh. Lecture.

5. Junker, Brian. "Multilevel Models - The Basics II." Applied Linear Models. Carnegie Mellon University. Pittsburgh. Lecture.

6. Junker, Brian. "Random slopes, correlation & centering, sample size." Applied Linear Models. Carnegie Mellon University. Pittsburgh. Lecture.

7. Junker, Brian. "mlm residuals." Applied Linear Models. Carnegie Mellon University. Pittsburgh. Lecture.

8. Junker, Brian. "Lmer estimation and model selection." Applied Linear Models. Carnegie Mellon University. Pittsburgh. Lecture.

# Technical Appendix

## 1 Loading the libraries

```
library(ggplot2)
library(tidyverse)
library(dplyr)
library(leaps)
library(MASS)
library(foreign)
library(lme4)
library(knitr)
library(HLMdiag)
library(boot)
library(arm)
library(car)
```

```
ratings <- read.csv("ratings.csv")
ratings <- ratings[-c(1, 26)]
```

### 1.1 Exploring the data set

### 1.2 Harmony

- Four unique values

```
unique(ratings$Harmony)
```

```
## [1] "I-IV-V" "I-V-IV" "I-V-VI" "IV-I-V"
```

```
summary(ratings$Harmony)
```

```
##    Length     Class      Mode
##      2520 character character
```

### 1.3 Instrument

- Three unique values

```
unique(ratings$Instrument)
```

```
## [1] "guitar" "piano"  "string"
```

```r
summary(ratings$Instrument)
```

```
##    Length     Class      Mode 
##      2520 character character 
```

## 1.4 Voice

- Three unique values

```r
unique(ratings$Voice)
```

```
## [1] "contrary" "par3rd"   "par5th"
```

```r
summary(ratings$Voice)
```

```
##    Length     Class      Mode 
##      2520 character character 
```

## 1.5 Selfdeclare

- 6 unique values
- 2 is most common
- Mean value is 2.443

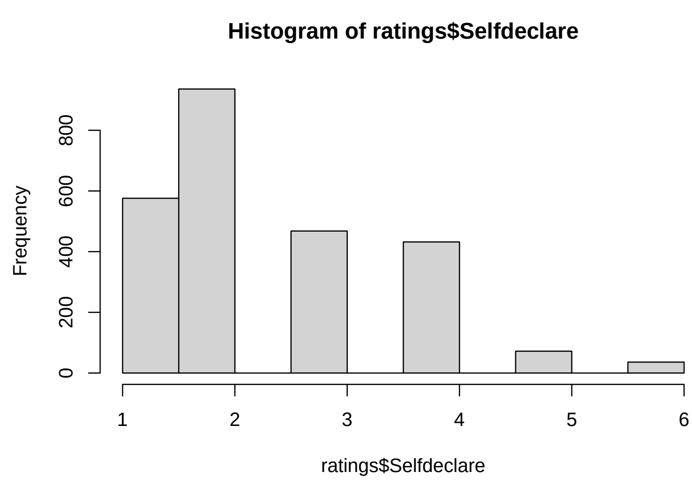
```r
unique(ratings$Selfdeclare)
```

```
## [1] 5 1 2 4 3 6
```

```r
summary(ratings$Selfdeclare)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##   1.000   2.000   2.000   2.443   3.000   6.000 
```

```r
hist(ratings$Selfdeclare)
```



### 1.6 OMSI

- 60 unique values
- Mean is 145.5
- Histogram is right skewed

```
length(sort(unique(ratings$OMSI)))
```

```
## [1] 60
```

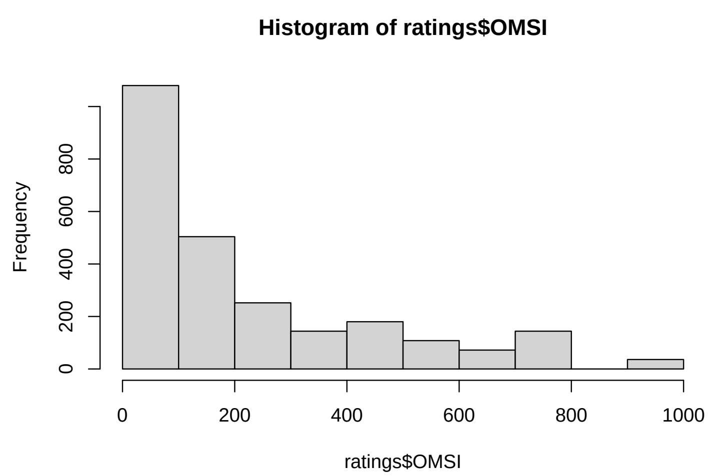
```
sort(unique(ratings$OMSI))
```

```
##  [1]  11  14  15  18  20  21  23  29  30  31  38  40  44  46  49  55  67  68  82
## [20]  88  94  96  97 122 127 142 145 146 147 150 164 179 180 194 199 201 204 233
## [39] 234 259 277 319 323 325 345 421 425 466 481 482 541 567 586 642 649 734 749
## [58] 759 784 970
```

```
summary(ratings$OMSI)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##    11.0    49.0   145.5   225.9   323.0   970.0
```

```
hist(ratings$OMSI)
```




### 1.7 X16.minus.17

- 13 unique values
- Mean is 1.721
- Histogram has slight right skewed

```r
length(sort(unique(ratings$X16.minus.17)))
```

```
## [1] 13
```

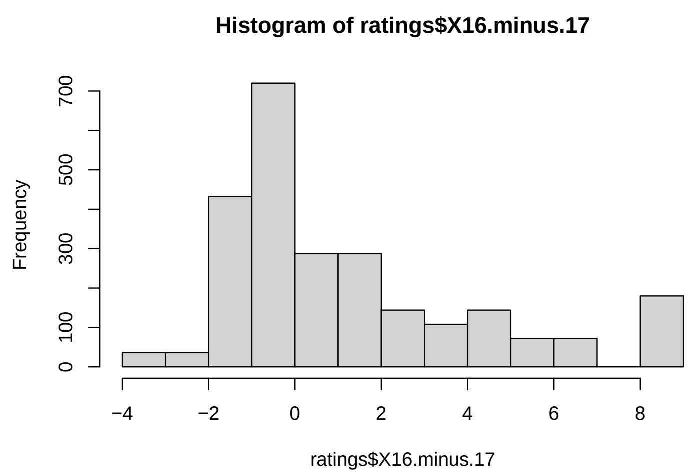
```r
sort(unique(ratings$X16.minus.17))
```

```
##  [1] -4.0 -2.0 -1.0 -0.5  0.0  1.0  2.0  3.0  4.0  5.0  6.0  7.0  9.0
```

```r
summary(ratings$X16.minus.17)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  -4.000   0.000   1.000   1.721   3.000   9.000
```

```r
hist(ratings$X16.minus.17)
```




## 1.8 ConsInstr

- 14 unique values
- Mean is 2.857
- Histogram is mostly uniformly distributed

```r
length(sort(unique(ratings$ConsInstr)))
```

```
## [1] 14
```

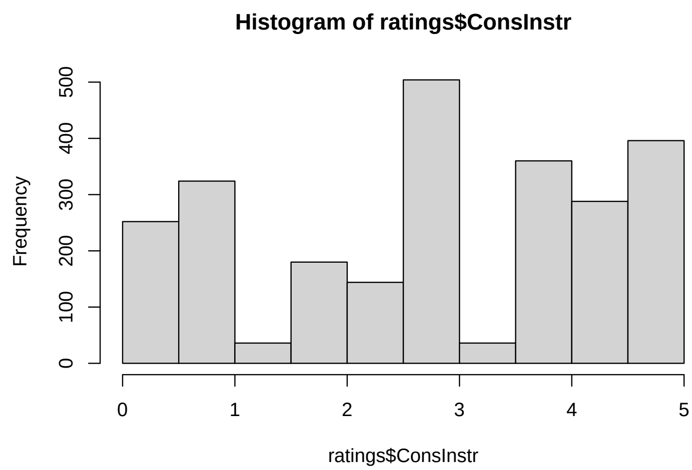
```r
sort(unique(ratings$ConsInstr))
```

```
##  [1] 0.00 0.67 1.00 1.33 1.67 2.00 2.33 2.67 3.00 3.33 3.67 4.00 4.33 5.00
```

```
summary(ratings$ConsInstr)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   1.670   3.000   2.857   4.330   5.000
```

```
hist(ratings$ConsInstr)
```




### 1.9 ConsNotes

- 5 factor levels with some NA values
- Mean is 2.533
- Histogram has three peaks and values $0, 3$, and 5 are most common

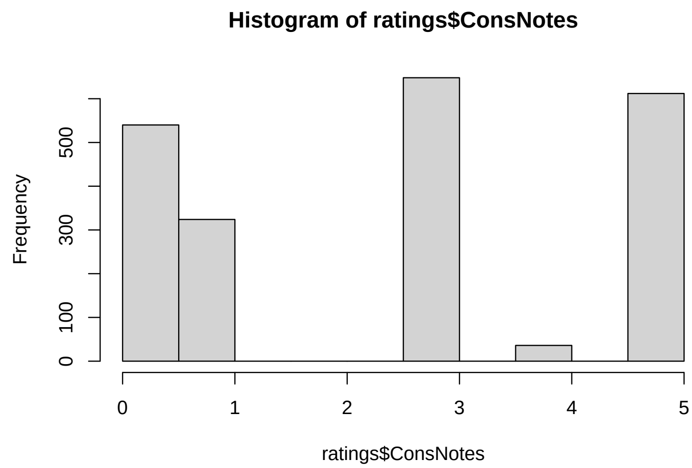
```
unique(ratings$ConsNotes)
```

```
## [1]  5 NA  0  3  1  4
```

```
summary(ratings$ConsNotes)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##   0.000   0.750   3.000   2.533   5.000   5.000     360
```

```
hist(ratings$ConsNotes)
```




### 1.10 `Instr.minus.Notes`

- 20 unique values
- Mean is 0.6857
- Distribution is normal with one peak

```
length(sort(unique(ratings$Instr.minus.Notes)))
```

```
## [1] 20
```

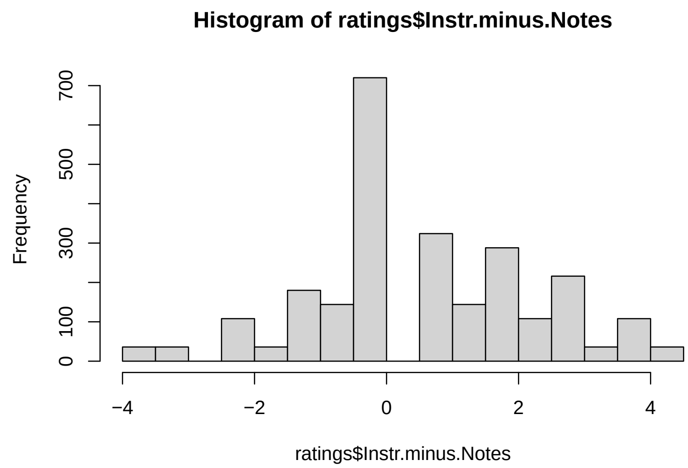
```
sort(unique(ratings$Instr.minus.Notes))
```

```
##  [1] -4.00 -3.00 -2.00 -1.67 -1.33 -1.00 -0.67  0.00  0.67  1.00  1.33  1.67
## [13]  2.00  2.33  2.67  3.00  3.33  3.67  4.00  4.33
```

```
summary(ratings$Instr.minus.Notes)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## -4.0000  0.0000  0.3350  0.6857  2.0000  4.3300
```

```
hist(ratings$Instr.minus.Notes)
```




## 1.11 PachListen

- 6 factor levels with some NA values
- Mean is 4.515
- 5 is most common
- Distribution is highly left skewed

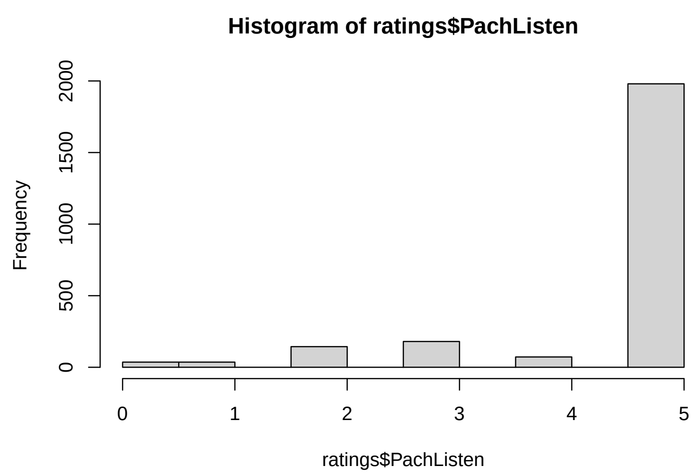
```
unique(ratings$PachListen)
```

```
## [1]  5  3 NA  2  1  4  0
```

```
summary(ratings$PachListen)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##   0.000   5.000   5.000   4.515   5.000   5.000      72
```

```
hist(ratings$PachListen)
```



### 1.12 ClsListen

- 6 factor levels with some NA values
- Mean is 2.159
- 1 and 3 are most common
- Histogram has three peaks

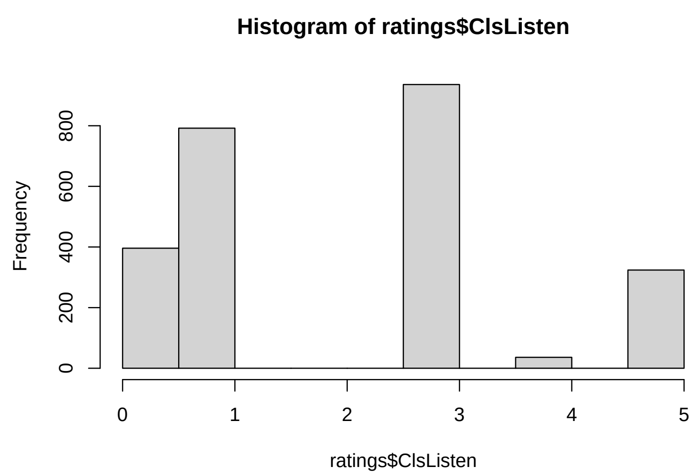
```
unique(ratings$ClsListen)
```

```
## [1]  4  0  1 NA  3  5
```

```
summary(ratings$ClsListen)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##   0.000   1.000   3.000   2.159   3.000   5.000      36
```

```
hist(ratings$ClsListen)
```




### 1.13 KnowRob

- 6 factor levels with some NA values
- Mean is 0.7692
- 0 is most common
- Histogram has one peak and is highly right skewed

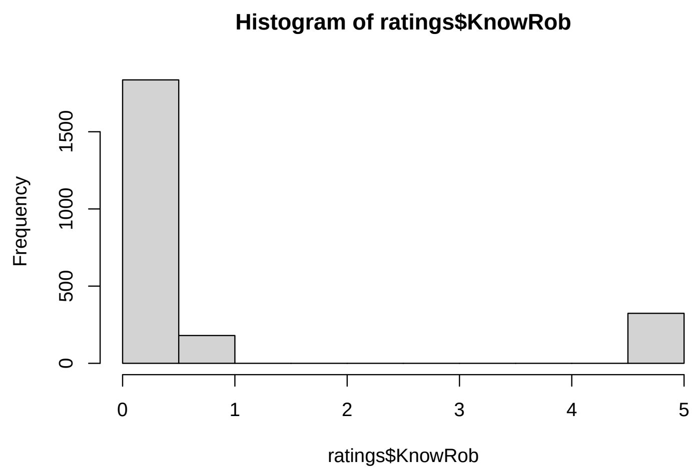
```
unique(ratings$KnowRob)
```

```
## [1]  0 NA  5  1
```

```r
summary(ratings$KnowRob)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's 
##  0.0000  0.0000  0.0000  0.7692  0.0000  5.0000     180
```

```r
hist(ratings$KnowRob)
```




## 1.14 KnowAxis

- 6 factor levels with some NA values
- Mean is 0.9032
- 0 is most common
- Histogram has one peak and is highly right skewed

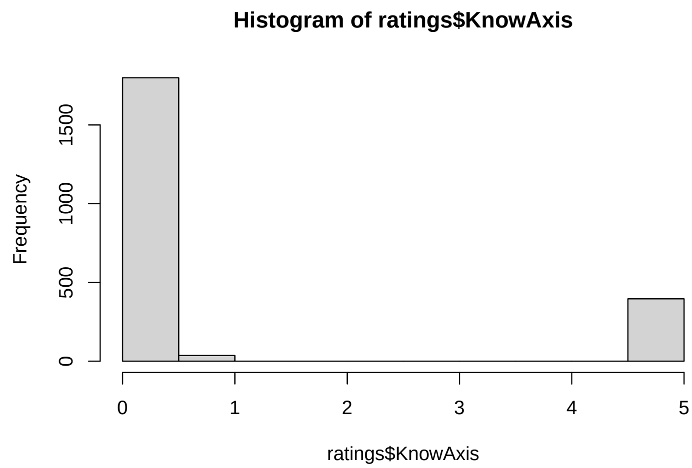
```r
unique(ratings$KnowAxis)
```

```
## [1]  0 NA  5  1
```

```r
summary(ratings$KnowAxis)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's 
##  0.0000  0.0000  0.0000  0.9032  0.0000  5.0000     288
```

```r
hist(ratings$KnowAxis)
```




## 1.15 X1990s2000s

- 6 factor levels with some NA values
- Mean is 4.061
- 5 is most common
- Histogram has one peak and is highly left skewed

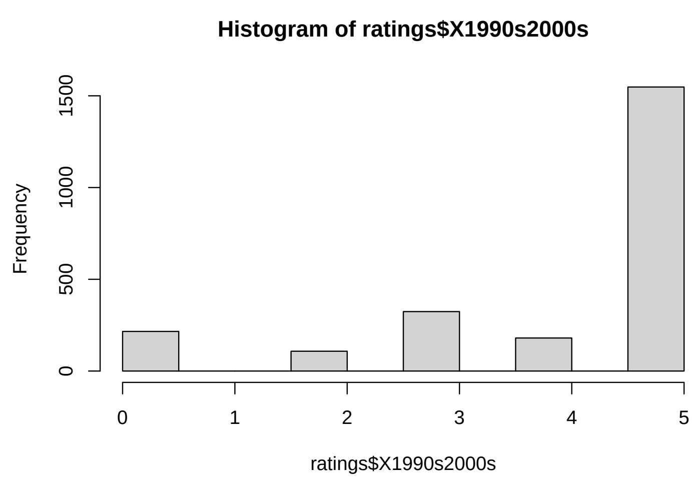
```
unique(ratings$X1990s2000s)
```

```
## [1]  5 NA  0  3  2  4
```

```
summary(ratings$X1990s2000s)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##   0.000   3.000   5.000   4.061   5.000   5.000     144
```

```
hist(ratings$X1990s2000s)
```



## 1.16 X1990s2000s.minus.1960s1970s

- 9 unique values with some NA values
- Mean is 2.015
- 0 and 3 are most common
- Histogram has mostly uniformly distributed

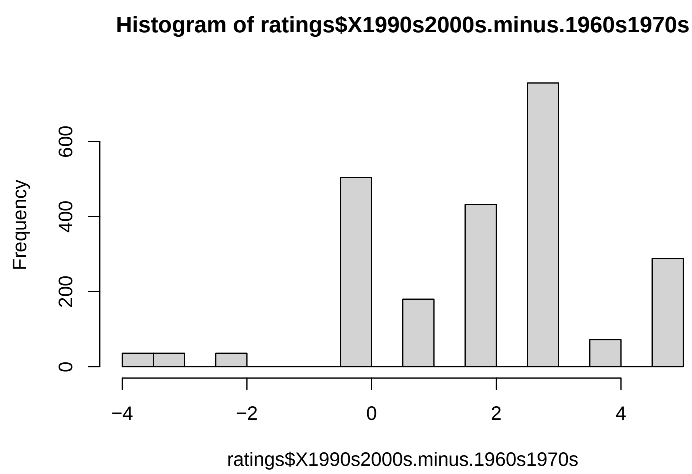
```
unique(ratings$X1990s2000s.minus.1960s1970s)
```

```
## [1]  2  3  5 NA  4  0  1 -2 -4 -3
```

```
summary(ratings$X1990s2000s.minus.1960s1970s)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##  -4.000   0.000   2.000   2.015   3.000   5.000     180
```

```
hist(ratings$X1990s2000s.minus.1960s1970s)
```




## 1.17 CollegeMusic

- 2 factor levels with some NA values
- Mean is 0.791 > 0.5 meaning 1 is more common than 0

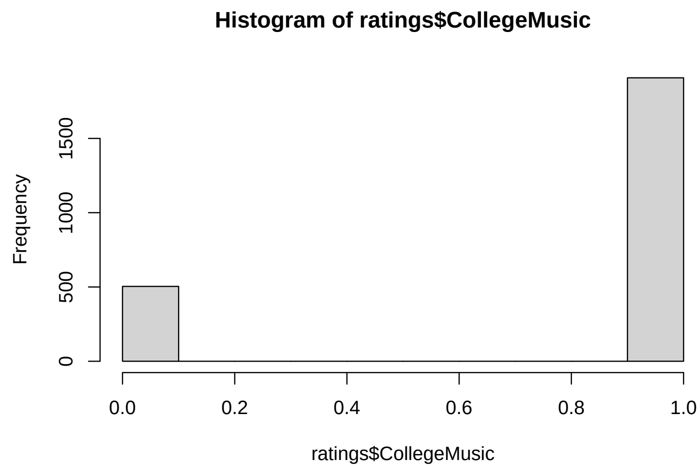
```
unique(ratings$CollegeMusic)
```

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

```
summary(ratings$CollegeMusic)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##   0.000   1.000   1.000   0.791   1.000   1.000     108
```

```
hist(ratings$CollegeMusic)
```




### 1.18 NoClass

- 6 factor levels with some NA values
- Mean is 0.9194
- Histogram is highly right skewed

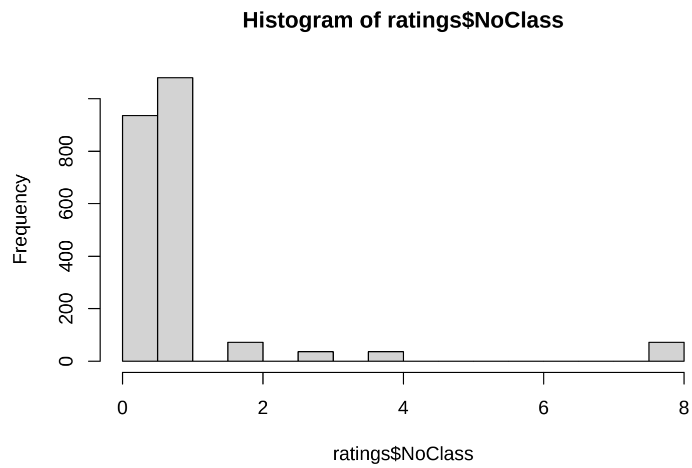
```
unique(ratings$NoClass)
```

```
## [1]  0  1 NA  4  3  8  2
```

```
summary(ratings$NoClass)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##  0.0000  0.0000  1.0000  0.9194  1.0000  8.0000     288
```

```
hist(ratings$NoClass)
```




### 1.19 APTheory

- 2 factor levels with some NA values
- Mean is 0.2344 $<$ 0.5 meaning 0 is more common than 1

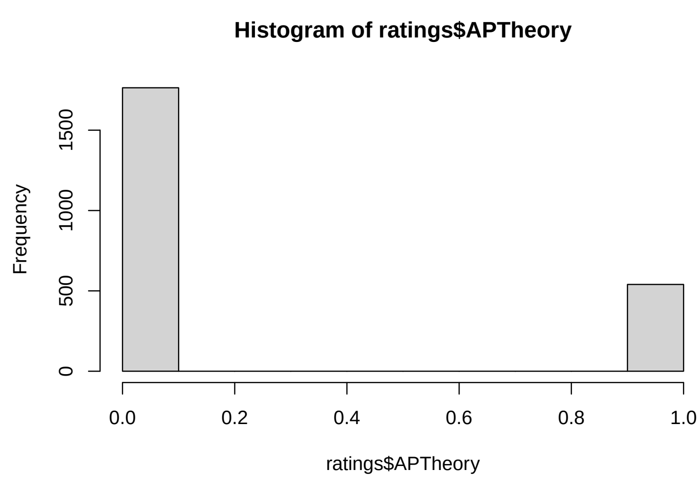
```
unique(ratings$APTheory)
```

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

```
summary(ratings$APTheory)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##  0.0000  0.0000  0.0000  0.2344  0.0000  1.0000     216
```

```
hist(ratings$APTheory)
```



## 1.20 Composing

- 6 factor levels with some NA values
- Mean is 1
- Histogram is highly right skewed

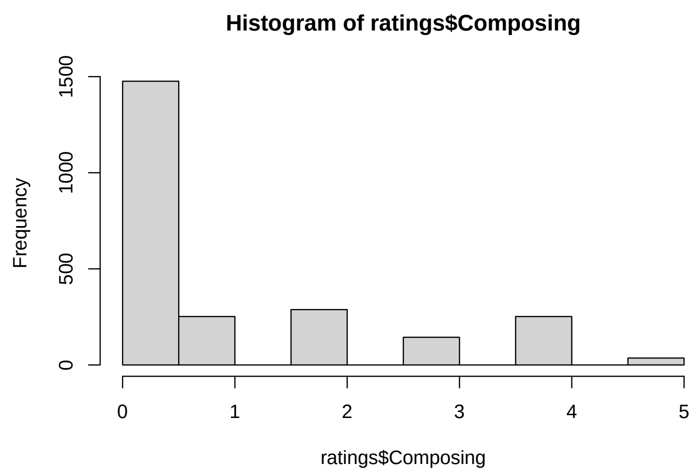
```
unique(ratings$Composing)
```

```
## [1]  4  0  1  2 NA  3  5
```

```
summary(ratings$Composing)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's 
##       0       0       0       1       2       5      72
```

```
hist(ratings$Composing)
```




## 1.21 PianoPlay

- 6 factor levels with no NA values
- Mean is 1.086
- Histogram is right skewed

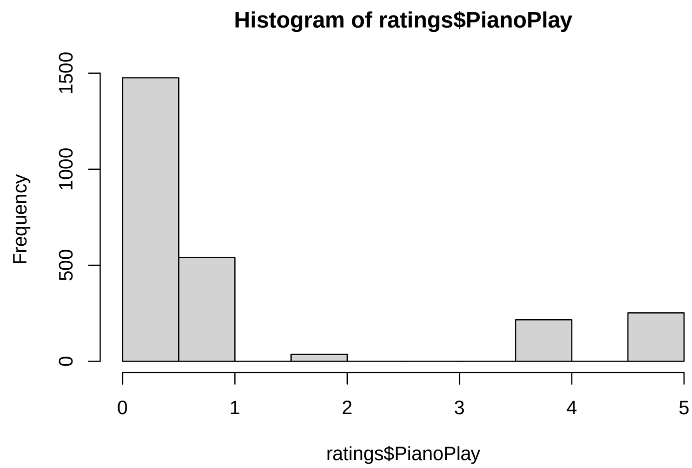
```
unique(ratings$PianoPlay)
```

```
## [1] 1 0 5 4 2
```

```
summary(ratings$PianoPlay)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##   0.000   0.000   0.000   1.086   1.000   5.000
```

```r
hist(ratings$PianoPlay)
```




## 1.22 GuitarPlay

- 6 factor levels with no NA values
- Mean is 0.6857
- Histogram is highly right skewed

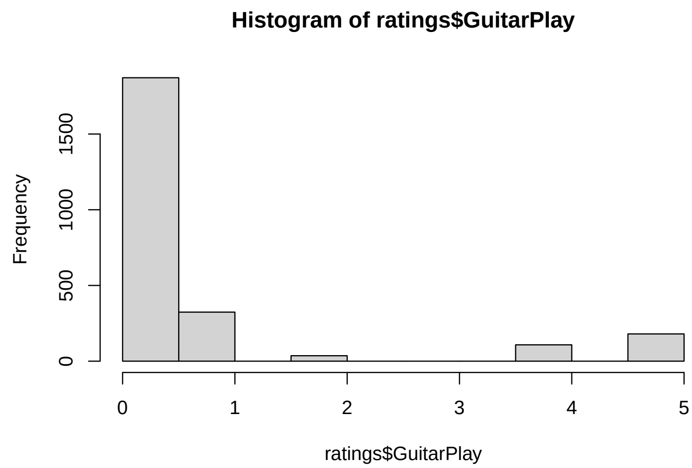
```r
unique(ratings$GuitarPlay)
```

```
## [1] 5 0 1 4 2
```

```r
summary(ratings$GuitarPlay)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.0000  0.0000  0.0000  0.6857  1.0000  5.0000
```

```r
hist(ratings$GuitarPlay)
```




## 1.23 X1stInstr

- 6 factor levels with some NA values
- Mean is 2.786
- 1 and 4 are most common values
- Histogram is has no skew and two peaks

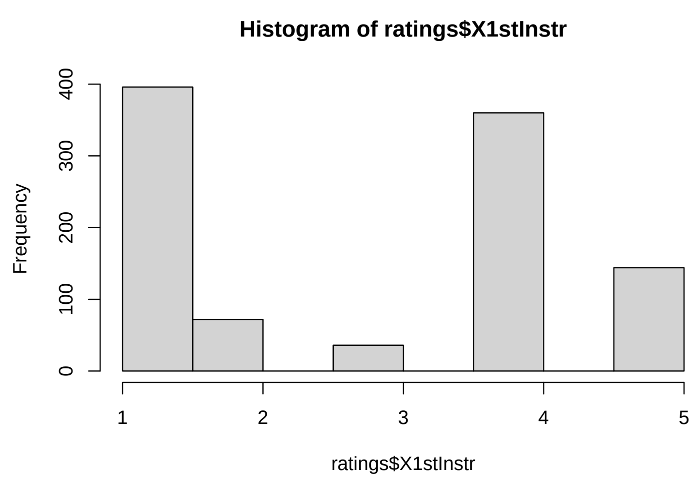
```
unique(ratings$X1stInstr)
```

```
## [1]  4  3 NA  1  5  2
```

```
summary(ratings$X1stInstr)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##   1.000   1.000   3.500   2.786   4.000   5.000    1512
```

```
hist(ratings$X1stInstr)
```



## 1.24 X1stInstr

- 6 factor levels with some NA values
- Mean is 1.556
- 1 is the most common value
- Histogram is right skewed with one peak

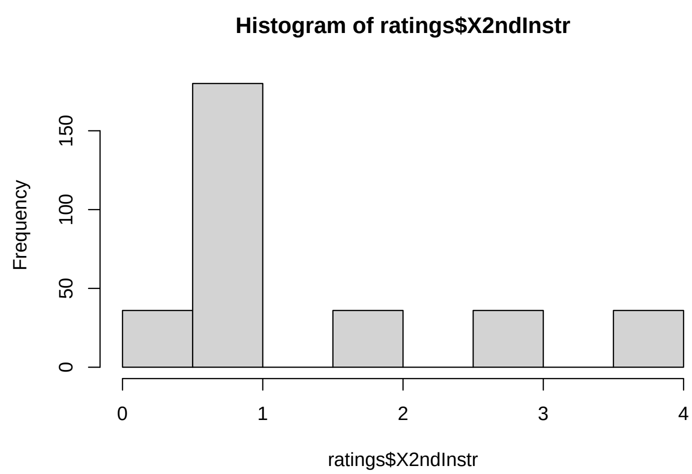
```
unique(ratings$X2ndInstr)
```

```
## [1] NA  1  0  4  2  3
```

```
summary(ratings$X2ndInstr)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##   0.000   1.000   1.000   1.556   2.000   4.000    2196
```

```
hist(ratings$X2ndInstr)
```




## 1.25 Classical

- 17 unique values with some NA values
- One of the unique values is 19, which is greater than the scale that the participants were presented with. We see that it occurs only once, so it can be reasonably inferred that this might be an error input. Hence, we will remove it from our data set
- Mean is 5.783
- Distribution is normal

```r
length(unique(ratings$Classical))
```

```
## [1] 17
```

```r
unique(ratings$Classical)
```

```
##  [1]  3.0  1.0  2.0  8.0 10.0  6.0  5.0  4.0  9.0  7.0   NA  0.0 19.0  9.5  4.6
## [16]  3.5  4.2
```

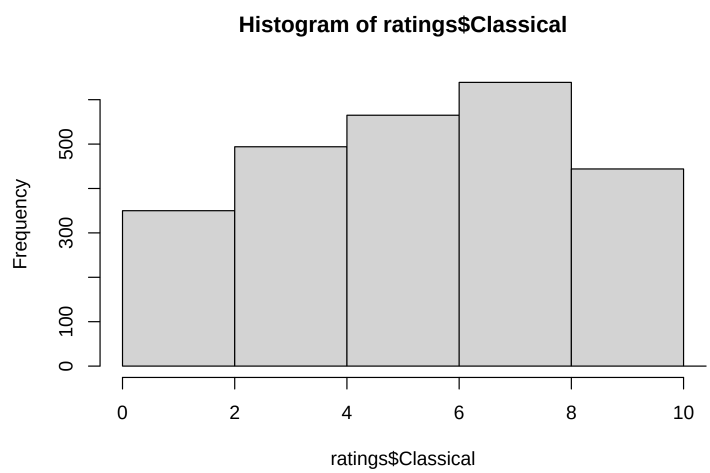
```r
summary(ratings$Classical)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##   0.000   4.000   6.000   5.783   8.000  19.000      27
```

```r
count(ratings[which(ratings$Classical > 10), ])
```

```
##   n
## 1 1
```

```r
hist(ratings$Classical, xlim = c(0, 10))
```




## 1.26 Popular

- 17 unique values with some NA values
- One of the unique values is 19, which is greater than the scale that the participants were presented with. We see that it occurs only once, so it can be reasonably inferred that this might be an error input. Hence, we will remove it from our data set
- Mean is 5.381
- Distribution is normal

```r
length(unique(ratings$Popular))
```

```
## [1] 17
```

```r
unique(ratings$Popular)
```

```
##  [1]  9.0  7.0  8.0  3.0  1.0  4.0  5.0  6.0  2.0 10.0  0.0   NA 19.0  3.5  4.6
## [16]  6.8  4.2
```

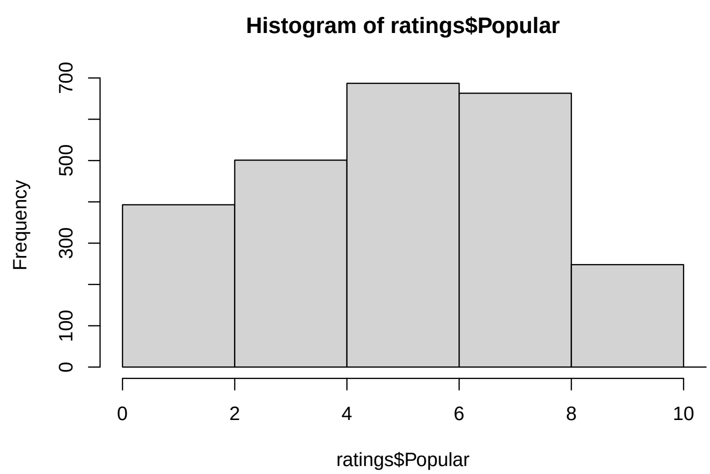
```r
summary(ratings$Popular)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's 
##   0.000   4.000   5.000   5.381   7.000  19.000      27
```

```r
count(ratings[which(ratings$Popular > 10), ])
```

```
##   n
## 1 1
```

```r
hist(ratings$Popular, xlim = c(0, 10))
```




## 1.27 Creating a preliminary OLS mode

We first create a linear model including three-way interactions between all the predictors `Instrument`, `Harmony`, and `Voice`. Upon running the `anova()` function, we see that the only interaction that is statistically significant at the 5% level of significance is `Harmony:Voice`, we keep only that. Then, we run the `anova()` function between the current and previous model to compare performance, which confirms that the second model is better.

```r
model_1 <- lm(Classical ~ Instrument * Harmony * Voice,
              data = ratings)

anova(model_1)
```

```
## Analysis of Variance Table
##
## Response: Classical
##                              Df  Sum Sq Mean Sq  F value    Pr(>F)
## Instrument                    2  4127.9 2063.96 391.7698 < 2.2e-16 ***
## Harmony                       3   273.6   91.20  17.3120  4.01e-11 ***
## Voice                         2    85.6   42.82   8.1278 0.0003032 ***
## Instrument:Harmony            6    10.4    1.74   0.3305 0.9211803
## Instrument:Voice              4     9.5    2.37   0.4504 0.7722123
## Harmony:Voice                 6    81.2   13.53   2.5691 0.0175040 *
## Instrument:Harmony:Voice     12    62.1    5.18   0.9829 0.4626550
## Residuals                  2457 12944.2    5.27
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
model_2 <- lm(Classical ~ Instrument + Harmony + Voice + Harmony:Voice,
              data = ratings)

anova(model_1, model_2)
```

```
## Analysis of Variance Table
##
## Model 1: Classical ~ Instrument * Harmony * Voice
## Model 2: Classical ~ Instrument + Harmony + Voice + Harmony:Voice
##   Res.Df   RSS  Df Sum of Sq      F Pr(>F)
## 1   2457 12944
## 2   2479 13026 -22   -82.005 0.7075 0.8362
```

## 1.28 Creating a multi-level model

We then create a multi-level model with only a random intercept for each individual.

```
multilevel_model_1 <- lmer(Classical ~
                             Instrument +
                             Harmony +
                             Voice +
                             Harmony:Voice +
                             (1 | Subject),
                           data = ratings,
                           REML = FALSE)
```

## 1.29 Examining the performance of MLM

We examine the influence of the three main experimental factors on the initial MLM using couple of approaches:

- **ANOVA:** We run the `anova()` function which confirms that the effect of these three experimental factors is significant.
- **Plotting residuals:** We plot the conditional and marginal residuals as a function of conditional and fitted values respectively. The smooth fitted line for both is almost a horizontal line centered at zero suggesting that the data fits our current model well.

- **Checking normality:** The standardized residuals and standardized random effects are normally distributed suggesting a good fit.
- **Fixed and random effect variances:** From the `summary()` output, we see that $\hat{\tau}_0^2 = 1.678$ and $\hat{\sigma}_0^2 = 3.537$. While these are not as low as we would like, suggesting possible scope of improvement within the model, it is a good starting point.
- **AIC/BIC:** Upon computing the AIC and BIC for the OLS `lm()` model and multi-level `lmer()` model, we see that the `lmer()` model has lower values for both AIC and BIC relative to the `lm()` model.

Based on the above findings, we decide that including the random effects part has improved the original model and thus we will keep it.

```
anova(multilevel_model_1)
```

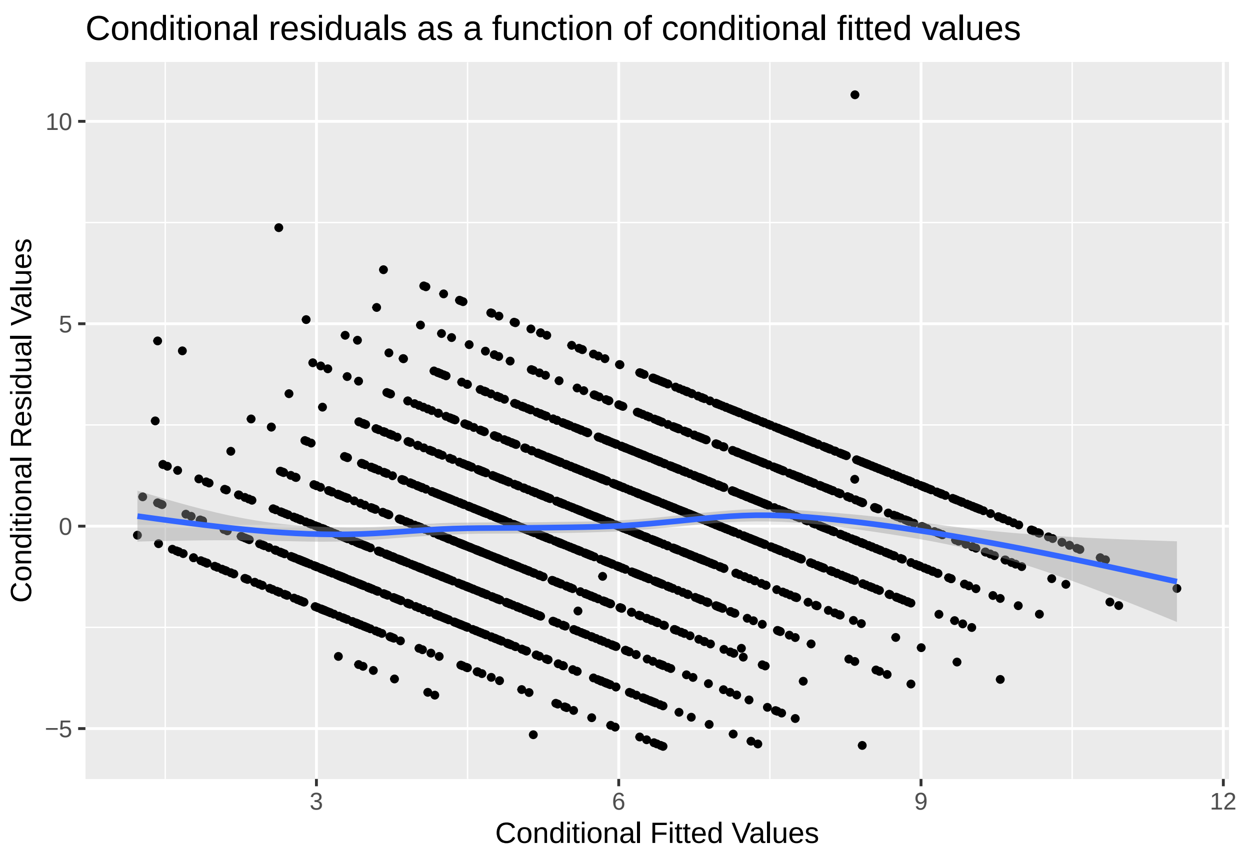
```
## Analysis of Variance Table
##                npar Sum Sq Mean Sq  F value
## Instrument        2 4119.1 2059.53 582.2537
## Harmony           3  275.4   91.79  25.9495
## Voice             2   87.0   43.49  12.2958
## Harmony:Voice     6   80.9   13.48   3.8108
```

```
summary(multilevel_model_1)
```

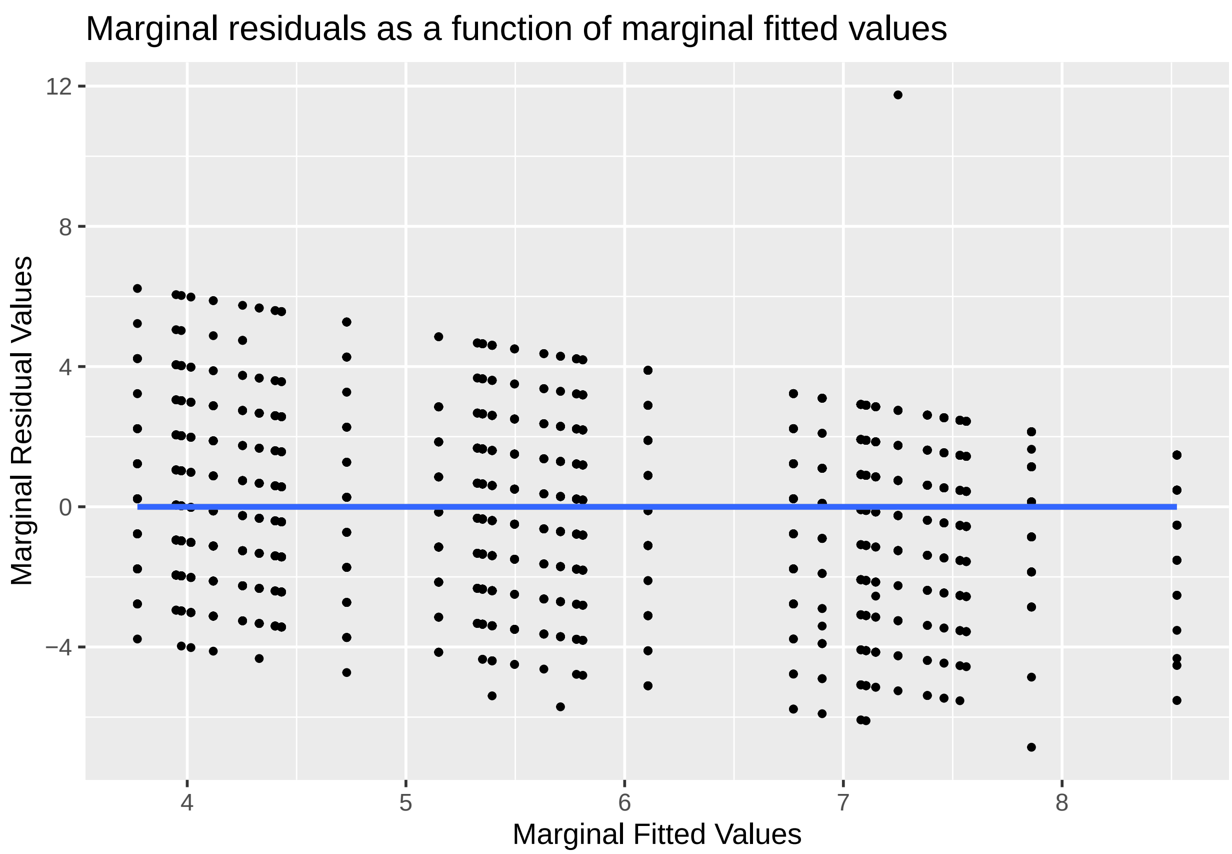
```
## Linear mixed model fit by maximum likelihood  ['lmerMod']
## Formula: Classical ~ Instrument + Harmony + Voice + Harmony:Voice + (1 |
##     Subject)
##    Data: ratings
##
##      AIC      BIC   logLik deviance df.resid
##  10458.1  10551.2  -5213.1  10426.1     2477
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.8924 -0.6212 -0.0165  0.6392  5.6657
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  Subject  (Intercept) 1.678    1.295
##  Residual             3.537    1.881
## Number of obs: 2493, groups:  Subject, 70
##
## Fixed effects:
##                           Estimate Std. Error t value
## (Intercept)                4.25306    0.20907  20.342
## Instrumentpiano            1.37746    0.09261  14.873
## Instrumentstring           3.13086    0.09200  34.030
## HarmonyI-V-IV              0.14892    0.18469   0.806
## HarmonyI-V-VI              1.14100    0.18445   6.186
## HarmonyIV-I-V             -0.13397    0.18398  -0.728
## Voicepar3rd               -0.28018    0.18400  -1.523
## Voicepar5th               -0.23618    0.18444  -1.281
## HarmonyI-V-IV:Voicepar3rd -0.34960    0.26072  -1.341
```

```
## HarmonyI-V-VI:Voicepar3rd -0.68277    0.26100  -2.616
## HarmonyIV-I-V:Voicepar3rd  0.49026    0.26068   1.881
## HarmonyI-V-IV:Voicepar5th -0.19316    0.26130  -0.739
## HarmonyI-V-VI:Voicepar5th -0.42874    0.26087  -1.644
## HarmonyIV-I-V:Voicepar5th  0.06604    0.26051   0.254
```

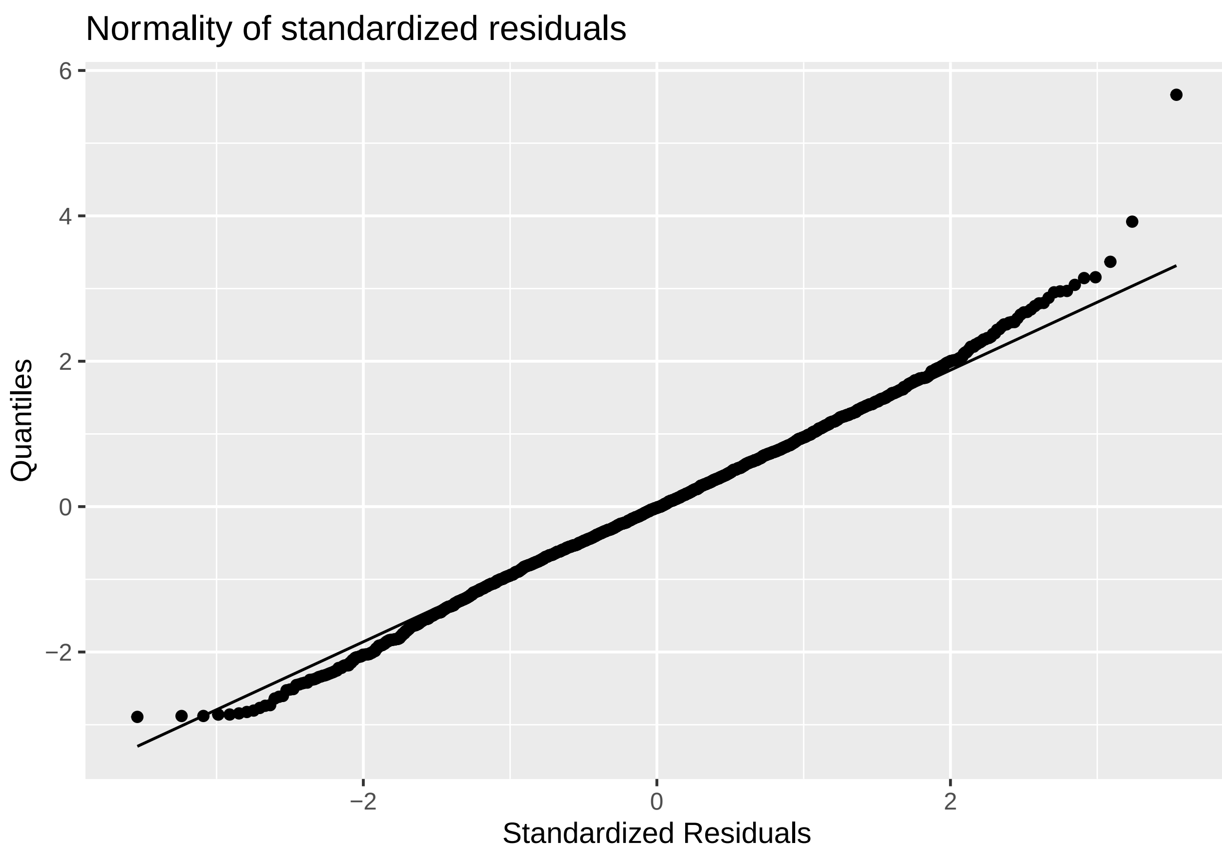
```
residuals_11 <- hlm_resid(multilevel_model_1,
                          level = 1,
                          include.ls = F)

residuals_11_std <- hlm_resid(multilevel_model_1,
                              level = 1,
                              include.ls = F,
                              standardize = T)

residuals_12 <- hlm_resid(multilevel_model_1,
                          level = "Subject",
                          include.ls = F)

residuals_12_std <- hlm_resid(multilevel_model_1,
                              level = "Subject",
                              include.ls = F,
                              standardize = T)

std_resid <- residuals_11_std$.std.resid
std_ranef_intercept <- residuals_12_std$.std.ranef.intercept

conditional_plot_1 <- ggplot(data = as.data.frame(residuals_11),
                             mapping = aes(y = residuals_11$.resid,
                                           residuals_11$.fitted)) +
  xlab("Conditional Fitted Values") +
  ylab("Conditional Residual Values") +
  ggtitle("Conditional residuals as a function of conditional fitted values") +
  geom_point(pch = 20) +
  geom_smooth()

conditional_plot_1
```



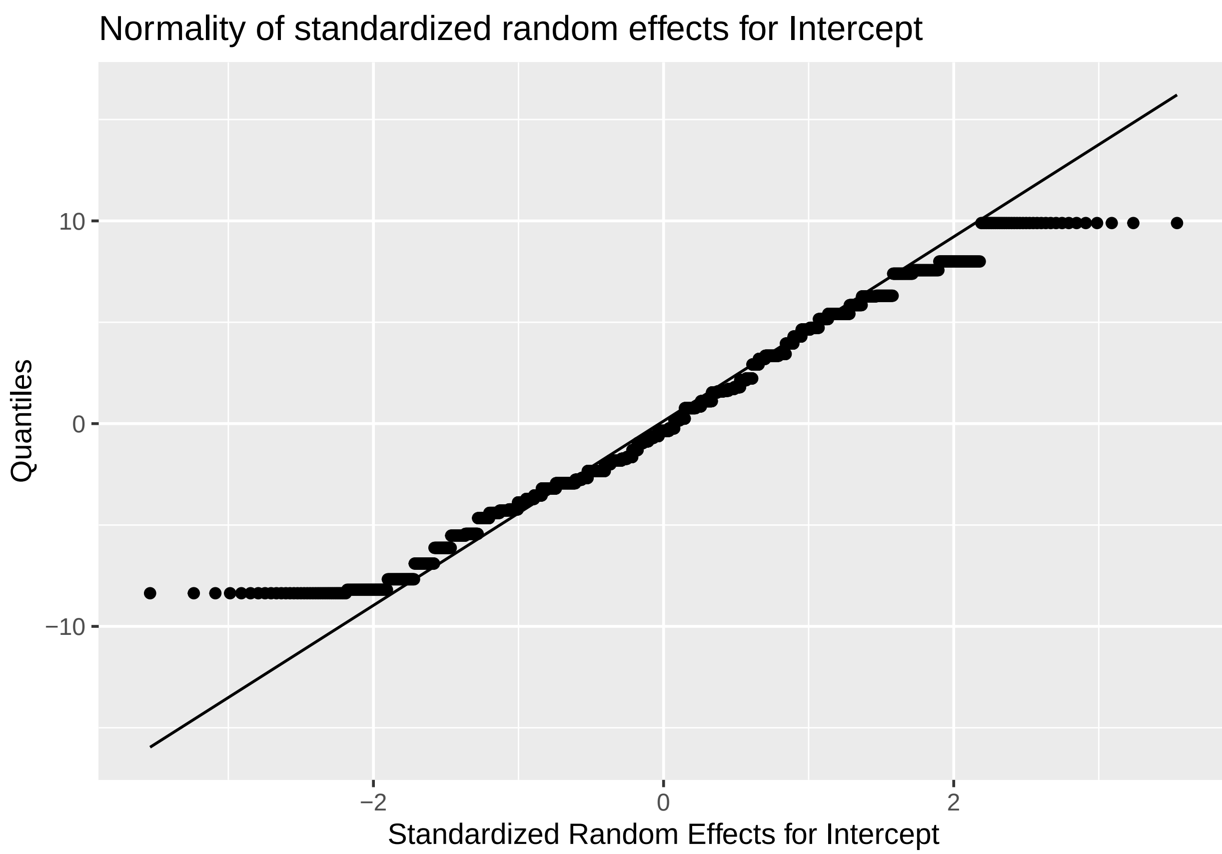

```
ggsave('conditional_plot_1.png')

marginal_plot_1 <- ggplot(data = as.data.frame(residuals_11),
                          mapping = aes(y = residuals_11$.mar.resid,
                                        x = residuals_11$.mar.fitted)) +
  xlab("Marginal Fitted Values") +
  ylab("Marginal Residual Values") +
  ggtitle("Marginal residuals as a function of marginal fitted values") +
  geom_point(pch = 20) +
  geom_smooth()

marginal_plot_1
```




```r
ggsave('marginal_plot_1.png')

params <- data.frame(cbind(std_resid, std_ranef_intercept))

plot_std_resid_1 <- params %>%
  ggplot(aes(sample = std_resid)) +
  stat_qq() +
  xlab("Standardized Residuals") +
  ylab("Quantiles") +
  ggtitle("Normality of standardized residuals") +
  stat_qq_line()

plot_std_resid_1
```




```
ggsave('plot_std_resid_1.png')

plot_std_ranef_intercept_1 <- params %>%
  ggplot(aes(sample = std_ranef_intercept)) +
  stat_qq() +
  xlab("Standardized Random Effects for Intercept") +
  ylab("Quantiles") +
  ggtitle("Normality of standardized random effects for Intercept") +
  stat_qq_line()

plot_std_ranef_intercept_1
```




```
ggsave('plot_std_ranef_intercept_1.png')
```

```
AIC_BIC_DIC <- cbind(AIC = sapply(list(model_2, multilevel_model_1),
                                  AIC),

                     BIC = sapply(list(model_2, multilevel_model_1),
                                  BIC),

                     DIC = sapply(list(multilevel_model_1, multilevel_model_1),
                                  extractDIC))

AIC_BIC_DIC <- t(as.tibble(AIC_BIC_DIC))
AIC_BIC_DIC[3, 1] = AIC_BIC_DIC[1, 1]
colnames(AIC_BIC_DIC) <- c("model_2", "multilevel_model_1")
kable(t(AIC_BIC_DIC), digits = 2)
```

| | AIC | BIC | DIC |
|---|---|---|---|
| model_2 | 11226.94 | 11314.26 | 11226.94 |
| multilevel_model_1 | 10458.10 | 10551.24 | 10426.10 |

### 1.30 Evaluating random effects for `Instrument`, `Harmony`, and `Voice`

As a next step, we test if our model could be improved by adding random slops for the three design variables. This is done by creating seven new `lmer()` models as follows in which we have included varying combinations of random effects for person/instrument, person/harmony, and person/voice. Upon computing the AIC, BIC, and DIC values for them all, we see that the multi-level model containing the random effect estimates for `(1 | Subject) + (0 + Instrument | Subject) + (0 + Harmony | Subject)` is the one yielding minimum values for all three of those. Thus, we will choose this as the best one so far.

```
multilevel_model_2 <- lmer(Classical ~
                             Instrument +
                             Harmony +
                             Voice +
                             Harmony:Voice +
                             (1 | Subject) +
                             (0 + Instrument | Subject) +
                             (0 + Harmony | Subject) +
                             (0 + Voice | Subject),
                           data = ratings,
                           REML = FALSE)

multilevel_model_3 <- lmer(Classical ~
                             Instrument +
                             Harmony +
                             Voice +
                             Harmony:Voice +
                             (1 | Subject) +
                             (0 + Instrument | Subject),
                           data = ratings,
                           REML = FALSE)

multilevel_model_4 <- lmer(Classical ~
                             Instrument +
                             Harmony +
                             Voice +
                             Harmony:Voice +
                             (1 | Subject) +
                             (0 + Harmony | Subject),
                           data = ratings,
                           REML = FALSE)

multilevel_model_5 <- lmer(Classical ~
                             Instrument +
                             Harmony +
                             Voice +
                             Harmony:Voice +
                             (1 | Subject) +
                             (0 + Voice | Subject),
                           data = ratings,
                           REML = FALSE)

multilevel_model_6 <- lmer(Classical ~
                             Instrument +
                             Harmony +
                             Voice +
                             Harmony:Voice +
                             (1 | Subject) +
                             (0 + Instrument | Subject) +
```

```r
                            (0 + Harmony | Subject),
                          data = ratings,
                          REML = FALSE)

multilevel_model_7 <- lmer(Classical ~
                            Instrument +
                            Harmony +
                            Voice +
                            Harmony:Voice +
                            (1 | Subject) +
                            (0 + Instrument | Subject) +
                            (0 + Voice | Subject),
                          data = ratings,
                          REML = FALSE)

multilevel_model_8 <- lmer(Classical ~
                            Instrument +
                            Harmony +
                            Voice +
                            Harmony:Voice +
                            (1 | Subject) +
                            (0 + Harmony | Subject) +
                            (0 + Voice | Subject),
                          data = ratings,
                          REML = FALSE)
```

```r
AIC_BIC_DIC <- cbind(AIC = sapply(list(multilevel_model_2,
                                       multilevel_model_3,
                                       multilevel_model_4,
                                       multilevel_model_5,
                                       multilevel_model_6,
                                       multilevel_model_7,
                                       multilevel_model_8),
                                  AIC),

                   BIC = sapply(list(multilevel_model_2,
                                     multilevel_model_3,
                                     multilevel_model_4,
                                     multilevel_model_5,
                                     multilevel_model_6,
                                     multilevel_model_7,
                                     multilevel_model_8),
                                BIC),

                   DIC = sapply(list(multilevel_model_2,
                                     multilevel_model_3,
                                     multilevel_model_4,
                                     multilevel_model_5,
                                     multilevel_model_6,
                                     multilevel_model_7,
                                     multilevel_model_8),
                                extractDIC))
```

```r
AIC_BIC_DIC <- t(as.tibble(AIC_BIC_DIC))
colnames(AIC_BIC_DIC) <- c("multilevel_model_2",
                           "multilevel_model_3",
                           "multilevel_model_4",
                           "multilevel_model_5",
                           "multilevel_model_6",
                           "multilevel_model_7",
                           "multilevel_model_8")
kable(t(AIC_BIC_DIC), digits = 2)
```

| | AIC | BIC | DIC |
|---|---|---|---|
| multilevel_model_2 | 9950.18 | 10171.39 | 9874.18 |
| multilevel_model_3 | 10086.96 | 10215.03 | 10042.96 |
| multilevel_model_4 | 10377.57 | 10528.92 | 10325.57 |
| multilevel_model_5 | 10470.10 | 10598.17 | 10426.10 |
| multilevel_model_6 | 9940.24 | 10126.52 | 9876.24 |
| multilevel_model_7 | 10098.45 | 10261.44 | 10042.45 |
| multilevel_model_8 | 10389.57 | 10575.85 | 10325.57 |

## 1.31 Examining the performance of MLM

We examine the influence of the three main experimental factors in our updated MLM using couple of approaches:

- **ANOVA:** we run the `anova()` function which confirms that the effect of these three experimental factors is significant.
- **Plotting residuals:** we plot the conditional and marginal residuals as a function of conditional and fitted values respectively. The smooth fitted line for both is almost a horizontal line centered at zero suggesting that the data fits our current model well.
- **Checking normality:** The standardized residuals show a normal fit with some outliers on either tails but they do not seem to pose a big issue.
- **Fixed and random effect variances:** From the `summary()` output, we see that $\hat{\sigma}_0^2 = 2.539209$, $\hat{\tau}_1^2 = 1.63176$, $\hat{\tau}_3^2 = 3.509338$, $\hat{\tau}_4^2 = 3.1353 \times 10^{-2}$, $\hat{\tau}_5^2 = 1.534336$, and $\hat{\tau}_6^2 = 4.644 \times 10^{-3}$.

These do seem to be an improvement from the previous multi-level model that we had.

```r
multilevel_model_6 <- lmer(Classical ~
                             Instrument +
                             Harmony +
                             Voice +
                             Harmony:Voice +
                             (1 + Instrument + Harmony | Subject),
                           data = ratings,
                           REML = FALSE)

anova(multilevel_model_6)
```

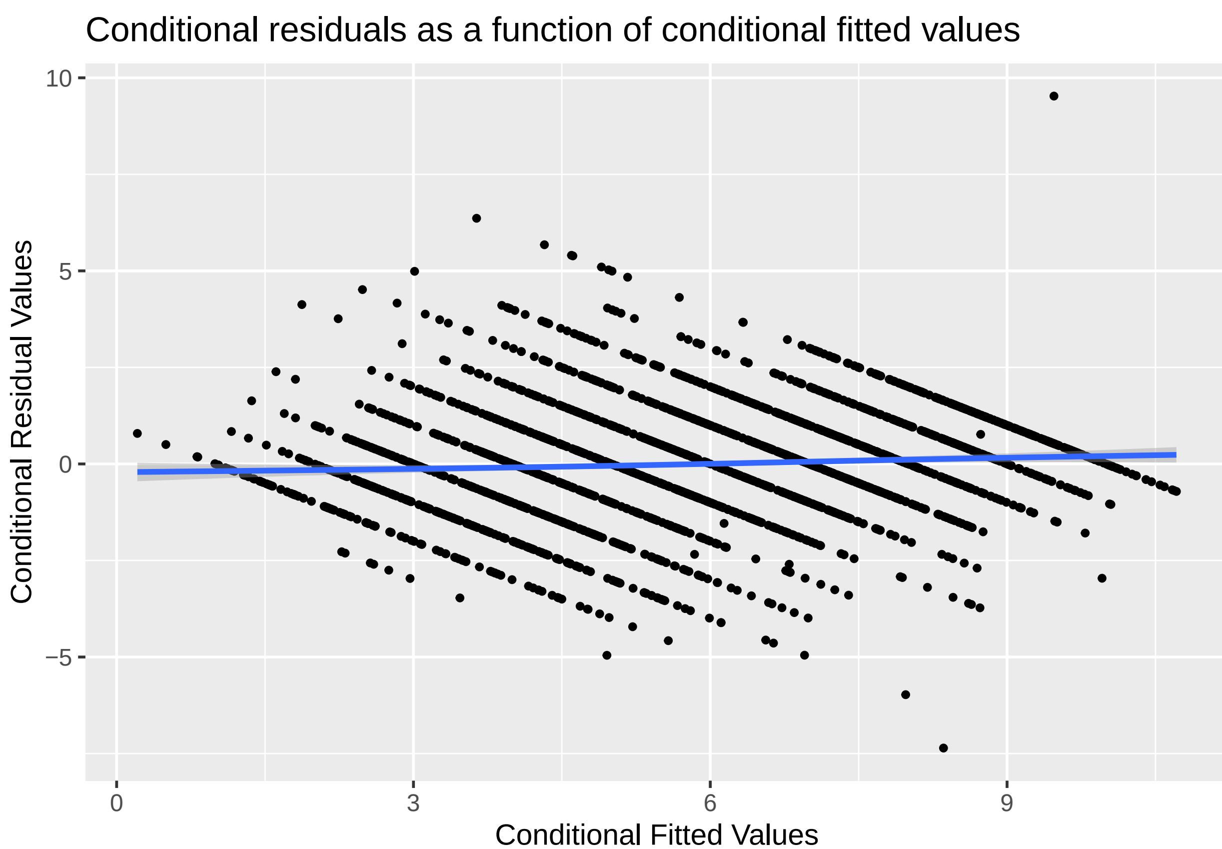
```
## Analysis of Variance Table
##                npar Sum Sq Mean Sq  F value
## Instrument        2 610.79 305.396 126.1343
## Harmony           3  53.92  17.973   7.4231
## Voice             2  84.96  42.482  17.5458
## Harmony:Voice     6  80.38  13.397   5.5331
```

```
summary(multilevel_model_6)
```

```
## Linear mixed model fit by maximum likelihood  ['lmerMod']
## Formula: Classical ~ Instrument + Harmony + Voice + Harmony:Voice + (1 +
##     Instrument + Harmony | Subject)
##    Data: ratings
##
##      AIC      BIC   logLik deviance df.resid
##   9937.3  10146.8  -4932.6   9865.3     2457
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -4.7284 -0.5798  0.0192  0.5701  6.1221
##
## Random effects:
##  Groups   Name             Variance Std.Dev. Corr
##  Subject  (Intercept)      2.539209 1.59349
##           Instrumentpiano  1.631760 1.27740  -0.39
##           Instrumentstring 3.509338 1.87332  -0.57  0.66
##           HarmonyI-V-IV    0.031353 0.17707   0.83 -0.77 -0.52
##           HarmonyI-V-VI    1.534336 1.23868  -0.03 -0.27 -0.43  0.00
##           HarmonyIV-I-V    0.004644 0.06815   0.27 -0.53  0.18  0.68 -0.12
##  Residual                  2.421195 1.55602
## Number of obs: 2493, groups:  Subject, 70
##
## Fixed effects:
##                           Estimate Std. Error t value
## (Intercept)                 4.2524     0.2232  19.055
## Instrumentpiano             1.3702     0.1710   8.012
## Instrumentstring            3.1274     0.2365  13.223
## HarmonyI-V-IV               0.1553     0.1543   1.007
## HarmonyI-V-VI               1.1387     0.2127   5.353
## HarmonyIV-I-V              -0.1335     0.1524  -0.876
## Voicepar3rd                -0.2707     0.1523  -1.778
## Voicepar5th                -0.2364     0.1526  -1.549
## HarmonyI-V-IV:Voicepar3rd  -0.3651     0.2158  -1.692
## HarmonyI-V-VI:Voicepar3rd  -0.6799     0.2160  -3.147
## HarmonyIV-I-V:Voicepar3rd   0.4854     0.2157   2.250
## HarmonyI-V-IV:Voicepar5th  -0.1891     0.2162  -0.874
## HarmonyI-V-VI:Voicepar5th  -0.4259     0.2160  -1.972
## HarmonyIV-I-V:Voicepar5th   0.0752     0.2156   0.349
## optimizer (nloptwrap) convergence code: 0 (OK)
## boundary (singular) fit: see help('isSingular')
```

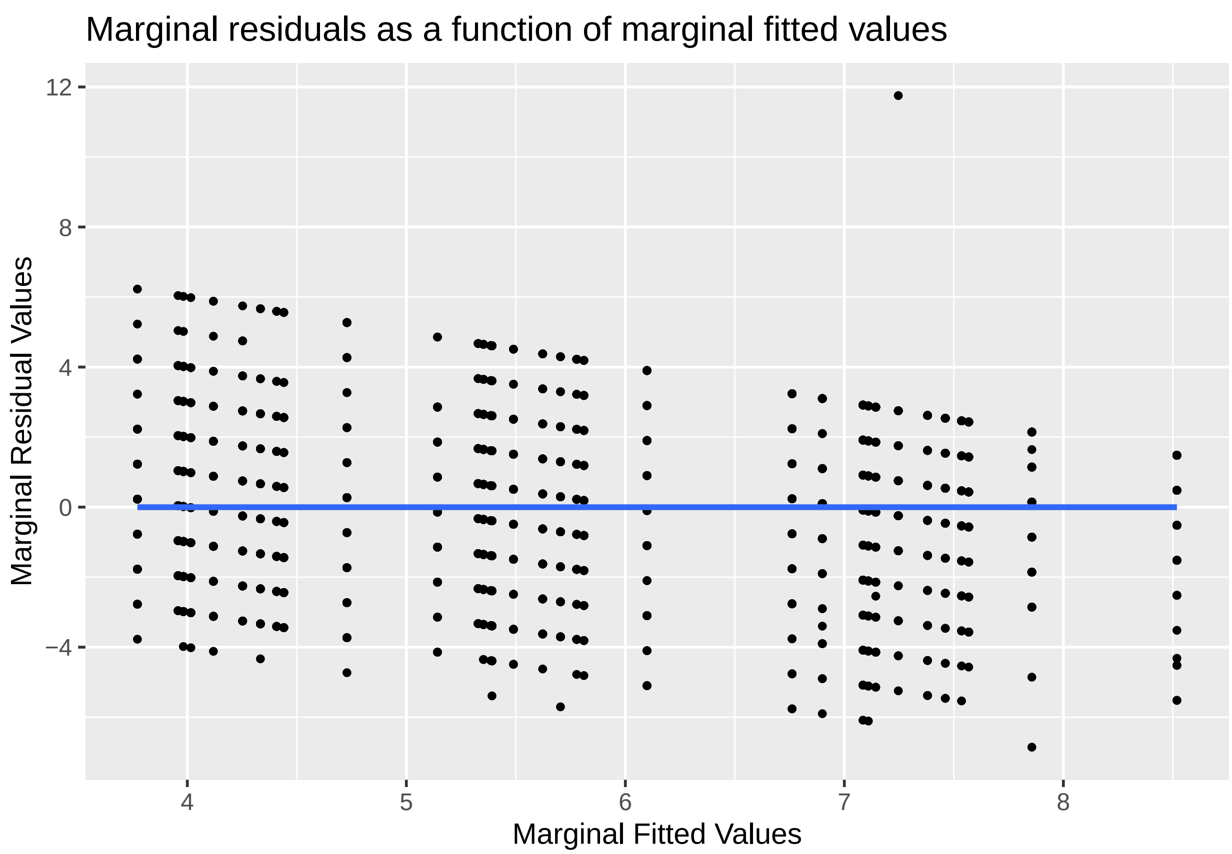
```
residuals_21 <- hlm_resid(multilevel_model_6,
                          level = 1,
```

```
                          include.ls = F)

residuals_21_std <- hlm_resid(multilevel_model_6,
                              level = 1,
                              include.ls = F,
                              standardize = T)

residuals_22 <- hlm_resid(multilevel_model_6,
                          level = "Subject",
                          include.ls = F)

conditional_plot_2 <- ggplot(data = as.data.frame(residuals_21),
                             mapping = aes(y = residuals_21$.resid,
                                           residuals_21$.fitted)) +
  xlab("Conditional Fitted Values") +
  ylab("Conditional Residual Values") +
  ggtitle("Conditional residuals as a function of conditional fitted values") +
  geom_point(pch = 20) +
  geom_smooth()

conditional_plot_2
```

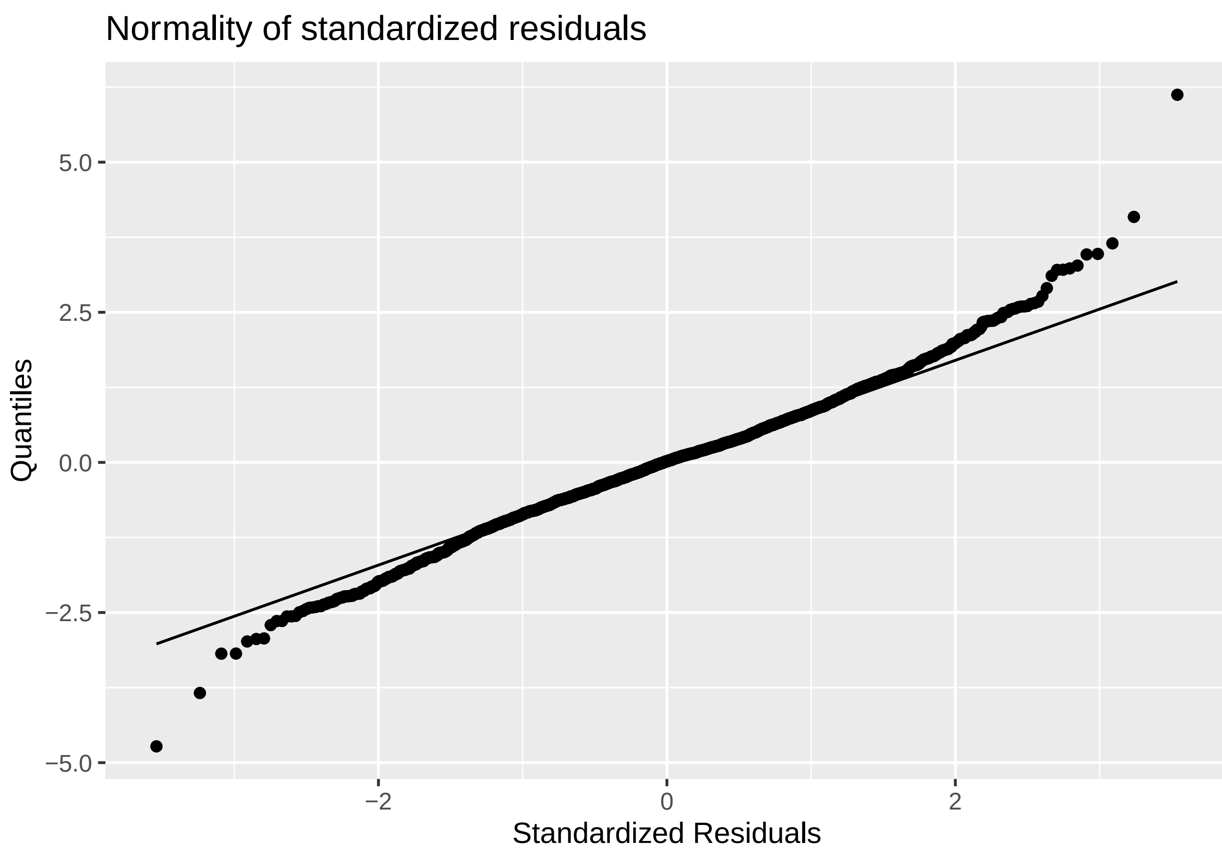



```
ggsave('conditional_plot_2.png')

marginal_plot_2 <- ggplot(data = as.data.frame(residuals_21),
```

```
                           mapping = aes(y = residuals_21$.mar.resid,
                                         x = residuals_21$.mar.fitted)) +
  xlab("Marginal Fitted Values") +
  ylab("Marginal Residual Values") +
  ggtitle("Marginal residuals as a function of marginal fitted values") +
  geom_point(pch = 20) +
  geom_smooth()

marginal_plot_2
```




```
ggsave('marginal_plot_2.png')

std_resid <- residuals_21_std$.std.resid
params <- data.frame(cbind(std_resid))

plot_std_resid_1 <- params %>%
  ggplot(aes(sample = std_resid)) +
  stat_qq() +
  xlab("Standardized Residuals") +
  ylab("Quantiles") +
  ggtitle("Normality of standardized residuals") +
  stat_qq_line()

plot_std_resid_1
```




```
ggsave('plot_std_resid_1.png')
```

## 1.32 Selecting covariates for fixed effects in model

In order to determine which person covariates should be added to the model as fixed effects, we will first only consider the fixed effects part of our multi-level model as described above and treat is as an OLS `lm()` function to perform analyses. we will perform `regsubsets()` on it to determine the optimal subset of person covariates that should be included in the model.

The model with the lowest BIC has the following covariates: `Harmony`, `Instrument`, `Voice`, `KnowAxis`.

The model with the lowest AIC has the following covariates: `Harmony`, `Instrument`, `Voice`, `SelfDeclare`, `X1990s2000s.minus.1960s1970s`, `GuitarPlay`, `X2ndInstr`

We will include all the covariates as selected by the lowest AIC and BIC models, and add them as fixed level predictors within my chosen multi-level model.

```
regsubsets <- regsubsets(Classical ~
                           Harmony +
                           Instrument +
                           Voice +
                           Selfdeclare +
                           OMSI +
                           X16.minus.17 +
                           ConsInstr +
                           ConsNotes +
```

```
                          Instr.minus.Notes +
                          PachListen +
                          ClsListen +
                          KnowRob +
                          KnowAxis +
                          X1990s2000s +
                          X1990s2000s.minus.1960s1970s +
                          CollegeMusic +
                          NoClass +
                          APTheory +
                          Composing +
                          PianoPlay +
                          GuitarPlay +
                          X1stInstr +
                          X2ndInstr,
                        data = ratings,
                        method = "exhaustive",
                        really.big = T,
                        nvmax = 99)

p <- dim(summary(regsubsets)$which)[2]
n <- dim(ratings)[1]

aic <- n * log(summary(regsubsets)$rss) + 2 * (p + 2)

results <- data.frame(summary(regsubsets)$which,
                      BIC = summary(regsubsets)$bic,
                      AIC = aic)

results[which(results$AIC == min(results$AIC) | results$BIC == min(results$BIC)), ]
```

```
##    X.Intercept. HarmonyI.V.IV HarmonyI.V.VI HarmonyIV.I.V Instrumentpiano
## 4          TRUE         FALSE          TRUE         FALSE            TRUE
## 11         TRUE          TRUE          TRUE          TRUE            TRUE
##    Instrumentstring Voicepar3rd Voicepar5th Selfdeclare  OMSI X16.minus.17
## 4              TRUE       FALSE       FALSE       FALSE FALSE        FALSE
## 11             TRUE        TRUE        TRUE        TRUE FALSE        FALSE
##    ConsInstr ConsNotes Instr.minus.Notes PachListen ClsListen KnowRob KnowAxis
## 4      FALSE     FALSE             FALSE      FALSE     FALSE   FALSE     TRUE
## 11     FALSE     FALSE             FALSE      FALSE     FALSE   FALSE    FALSE
##    X1990s2000s X1990s2000s.minus.1960s1970s CollegeMusic NoClass APTheory
## 4        FALSE                        FALSE        FALSE   FALSE    FALSE
## 11       FALSE                         TRUE        FALSE   FALSE    FALSE
##    Composing PianoPlay GuitarPlay X1stInstr X2ndInstr       BIC      AIC
## 4      FALSE     FALSE      FALSE     FALSE     FALSE -71.40665 16509.98
## 11     FALSE     FALSE       TRUE     FALSE      TRUE -41.25725 16423.16
```

## 1.33 Analyzing for changes in random effects on the model

In order to see if there should be any changes in the random effects, we implement an idea similar to **forward selection** but using DIC. The idea here is that we first add create six new models each containing one of the new predictors added under the fixed effects part of the model. we then compare the DIC for all these models and if the lowest DIC of these models containing the new predictor in the random effects part is lower than the DIC of the base model without any of the new predictors in its random effects part, then we add that predictor into the random effects. Taking that model then as the base, we repeat the process where we create five new models each containing one of the new predictors added and compare their DICs with that of the model selected in previous step.

Our findings are as follows:

**Step 1:** Model with random effects part `(1 + Instrument + Harmony + KnowAxis | Subject)` has lowest DIC among all models, lower than the previous `(1 + Instrument + Harmony | Subject)`, so we select that.

**Step 2:** Model with random effects part `(1 + Instrument + Harmony + KnowAxis + X2ndInstr | Subject)` has the lowest DIC among all models, but not lower than the best model selected at previous step with fixed effects part `(1 + Instrument + Harmony + KnowAxis | Subject)`. Thus, our optimal model does not change.

## 1.34 Step 1: Add `KnowAxis` as a random effect covariate

```
multilevel_model_7 <- lmer(Classical ~
                             Instrument +
                             Harmony +
                             Voice +
                             Harmony:Voice +
                             as.factor(Selfdeclare) +
                             as.factor(KnowAxis) +
                             X1990s2000s.minus.1960s1970s +
                             as.factor(GuitarPlay) +
                             as.factor(X2ndInstr) +
                             (1 + Instrument + Harmony | Subject),
                           data = ratings,
                           REML = FALSE)

multilevel_model_8 <- lmer(Classical ~
                             Instrument +
                             Harmony +
                             Voice +
                             Harmony:Voice +
                             as.factor(Selfdeclare) +
                             as.factor(KnowAxis) +
                             X1990s2000s.minus.1960s1970s +
                             as.factor(GuitarPlay) +
                             as.factor(X2ndInstr) +
                             (1 + Instrument + Harmony + as.factor(Selfdeclare) | Subject),
                           data = ratings,
                           REML = FALSE)

multilevel_model_9 <- lmer(Classical ~
                             Instrument +
                             Harmony +
                             Voice +
                             Harmony:Voice +
                             as.factor(Selfdeclare) +
                             as.factor(KnowAxis) +
                             X1990s2000s.minus.1960s1970s +
```

```
                             as.factor(GuitarPlay) +
                             as.factor(X2ndInstr) +
                             (1 + Instrument + Harmony + as.factor(KnowAxis) | Subject),
                           data = ratings,
                           REML = FALSE)

multilevel_model_10 <- lmer(Classical ~
                             Instrument +
                             Harmony +
                             Voice +
                             Harmony:Voice +
                             as.factor(Selfdeclare) +
                             as.factor(KnowAxis) +
                             X1990s2000s.minus.1960s1970s +
                             as.factor(GuitarPlay) +
                             as.factor(X2ndInstr) +
                             (1 + Instrument + Harmony + X1990s2000s.minus.1960s1970s | Subject),
                           data = ratings,
                           REML = FALSE)

multilevel_model_11 <- lmer(Classical ~
                             Instrument +
                             Harmony +
                             Voice +
                             Harmony:Voice +
                             as.factor(Selfdeclare) +
                             as.factor(KnowAxis) +
                             X1990s2000s.minus.1960s1970s +
                             as.factor(GuitarPlay) +
                             as.factor(X2ndInstr) +
                             (1 + Instrument + Harmony + as.factor(GuitarPlay) | Subject),
                           data = ratings,
                           REML = FALSE)

multilevel_model_12 <- lmer(Classical ~
                             Instrument +
                             Harmony +
                             Voice +
                             Harmony:Voice +
                             as.factor(Selfdeclare) +
                             as.factor(KnowAxis) +
                             X1990s2000s.minus.1960s1970s +
                             as.factor(GuitarPlay) +
                             as.factor(X2ndInstr) +
                             (1 + Instrument + Harmony + as.factor(X2ndInstr) | Subject),
                           data = ratings,
                           REML = FALSE)

DIC <- cbind(DIC = sapply(list(multilevel_model_7,
                               multilevel_model_8,
                               multilevel_model_9,
                               multilevel_model_10,
                               multilevel_model_11,
```

```
                              multilevel_model_12),
                         extractDIC))

DIC <- t(as.tibble(DIC))
colnames(DIC) <- c("multilevel_model_7",
                   "multilevel_model_8",
                   "multilevel_model_9",
                   "multilevel_model_10",
                   "multilevel_model_11",
                   "multilevel_model_12")
kable(t(DIC), digits = 3)
```

| | DIC |
|---|---|
| multilevel_model_7 | 1078.344 |
| multilevel_model_8 | 1078.736 |
| multilevel_model_9 | 1078.299 |
| multilevel_model_10 | 1078.305 |
| multilevel_model_11 | 1078.301 |
| multilevel_model_12 | 1078.541 |

## 1.35 Step 2: Add `X2ndInstr` as a random effect covariate

```
multilevel_model_13 <- lmer(Classical ~
                             Instrument +
                             Harmony +
                             Voice +
                             Harmony:Voice +
                             as.factor(Selfdeclare) +
                             as.factor(KnowAxis) +
                             X1990s2000s.minus.1960s1970s +
                             as.factor(GuitarPlay) +
                             as.factor(X2ndInstr) +
                             (1 + Instrument + Harmony + as.factor(KnowAxis) + as.factor(Selfdeclare) |
                           data = ratings,
                           REML = FALSE)

multilevel_model_14 <- lmer(Classical ~
                             Instrument +
                             Harmony +
                             Voice +
                             Harmony:Voice +
                             as.factor(Selfdeclare) +
                             as.factor(KnowAxis) +
                             X1990s2000s.minus.1960s1970s +
                             as.factor(GuitarPlay) +
                             as.factor(X2ndInstr) +
                             (1 + Instrument + Harmony + as.factor(KnowAxis) + X1990s2000s.minus.1960s1
                           data = ratings,
                           REML = FALSE)
```

```
multilevel_model_15 <- lmer(Classical ~
                              Instrument +
                              Harmony +
                              Voice +
                              Harmony:Voice +
                              as.factor(Selfdeclare) +
                              as.factor(KnowAxis) +
                              X1990s2000s.minus.1960s1970s +
                              as.factor(GuitarPlay) +
                              as.factor(X2ndInstr) +
                              (1 + Instrument + Harmony + as.factor(KnowAxis) + as.factor(GuitarPlay) | S
                            data = ratings,
                            REML = FALSE)

multilevel_model_16 <- lmer(Classical ~
                              Instrument +
                              Harmony +
                              Voice +
                              Harmony:Voice +
                              as.factor(Selfdeclare) +
                              as.factor(KnowAxis) +
                              X1990s2000s.minus.1960s1970s +
                              as.factor(GuitarPlay) +
                              as.factor(X2ndInstr) +
                              (1 + Instrument + Harmony + as.factor(KnowAxis) + as.factor(X2ndInstr) | Su
                            data = ratings,
                            REML = FALSE)

DIC <- cbind(DIC = sapply(list(multilevel_model_9,
                               multilevel_model_13,
                               multilevel_model_14,
                               multilevel_model_15,
                               multilevel_model_16),
                          extractDIC))

DIC <- t(as.tibble(DIC))
colnames(DIC) <- c("multilevel_model_9",
                   "multilevel_model_13",
                   "multilevel_model_14",
                   "multilevel_model_15",
                   "multilevel_model_16")
kable(t(DIC), digits = 3)
```

| | DIC |
|---|---|
| multilevel_model_9 | 1078.299 |
| multilevel_model_13 | 1078.739 |
| multilevel_model_14 | 1078.336 |
| multilevel_model_15 | 1079.325 |
| multilevel_model_16 | 1078.309 |

## 1.36 Testing second hypothesis

Upon running the function `count()` and doing some preliminary analysis on the data set we see that about 60% of the survey participants have selected a value of 1 or 2 for `Selfdeclare`, and the remaining 40% have selected $3, 4$, or 5. Thus, we dichotomize that variable by creating a new variable called `self_declared_musician` where:

- `self_declared_musician = yes`, if `Selfdeclare` $\in \{1, 2\}$
- `self_declared_musician = no`, if `Selfdeclare` $\in \{3, 4, 5\}$

We then replace the variable `Selfdeclare` with as the fixed level covariate in our model for the purpose of this exercise.

Then, in order to see which interactions with `self_declared_musician` on the fixed level are useful, we ucreate new models where each of them contains an interaction with `self_declared_musician` and one of the other fixed level covariates, and compare the AIC and BIC with the base level model which has no interactions with `self_declared_musician`. If any model containing interaction with `self_declared_musician` has a lower AIC/BIC than the base-level model, then it would suggest that the dichotomized musician variable is sensitive to interaction with that other fixed level covariate.

We find that the models containing interaction terms `self_declared_musician:Instrument` and `self_declared_musician:Harmony` have a lower AIC than the model not containing any interaction terms with `self_declared_musician` and the other fixed level covariates, suggesting that `Instrument` and `Harmony` are sensitive to the dichotomization of `self_declared_musiciant`. Out of these two, the model containing the interaction term `self_declared_musician:Instrument` has a lower BIC suggesting that `Instrument` is the most sensitive.

**Contextual interpretation:** What this means is that the type of instrument that an individual plays could be influential towards whether they consider themselves as a musician or not.

```
ratings$self_declared_musician <- ifelse(ratings$Selfdeclare <= 2, "no", "yes")

multilevel_model_17 <- lmer(Classical ~
                              Instrument +
                              Harmony +
                              Voice +
                              Harmony:Voice +
                              self_declared_musician +
                              as.factor(KnowAxis) +
                              X1990s2000s.minus.1960s1970s +
                              as.factor(GuitarPlay) +
                              as.factor(X2ndInstr) +
                              (1 + Instrument + Harmony + as.factor(KnowAxis) | Subject),
                            data = ratings,
                            REML = FALSE)

multilevel_model_18 <- lmer(Classical ~
                              Instrument +
                              Harmony +
                              Voice +
                              Harmony:Voice +
                              self_declared_musician +
                              as.factor(KnowAxis) +
                              X1990s2000s.minus.1960s1970s +
```

```
                          as.factor(GuitarPlay) +
                          as.factor(X2ndInstr) +
                          self_declared_musician:Instrument +
                          (1 + Instrument + Harmony + as.factor(KnowAxis) | Subject),
                        data = ratings,
                        REML = FALSE)

multilevel_model_19 <- lmer(Classical ~
                          Instrument +
                          Harmony +
                          Voice +
                          Harmony:Voice +
                          self_declared_musician +
                          as.factor(KnowAxis) +
                          X1990s2000s.minus.1960s1970s +
                          as.factor(GuitarPlay) +
                          as.factor(X2ndInstr) +
                          self_declared_musician:Harmony +
                          (1 + Instrument + Harmony + as.factor(KnowAxis) | Subject),
                        data = ratings,
                        REML = FALSE)

multilevel_model_20 <- lmer(Classical ~
                          Instrument +
                          Harmony +
                          Voice +
                          Harmony:Voice +
                          self_declared_musician +
                          as.factor(KnowAxis) +
                          X1990s2000s.minus.1960s1970s +
                          as.factor(GuitarPlay) +
                          as.factor(X2ndInstr) +
                          self_declared_musician:Voice +
                          (1 + Instrument + Harmony + as.factor(KnowAxis) | Subject),
                        data = ratings,
                        REML = FALSE)

multilevel_model_21 <- lmer(Classical ~
                          Instrument +
                          Harmony +
                          Voice +
                          Harmony:Voice +
                          self_declared_musician +
                          as.factor(KnowAxis) +
                          X1990s2000s.minus.1960s1970s +
                          as.factor(GuitarPlay) +
                          as.factor(X2ndInstr) +
                          self_declared_musician:as.factor(KnowAxis) +
                          (1 + Instrument + Harmony + as.factor(KnowAxis) | Subject),
                        data = ratings,
                        REML = FALSE)

multilevel_model_22 <- lmer(Classical ~
```

```
                             Instrument +
                             Harmony +
                             Voice +
                             Harmony:Voice +
                             self_declared_musician +
                             as.factor(KnowAxis) +
                             X1990s2000s.minus.1960s1970s +
                             as.factor(GuitarPlay) +
                             as.factor(X2ndInstr) +
                             self_declared_musician:X1990s2000s.minus.1960s1970s +
                             (1 + Instrument + Harmony + as.factor(KnowAxis) | Subject),
                           data = ratings,
                           REML = FALSE)

multilevel_model_23 <- lmer(Classical ~
                             Instrument +
                             Harmony +
                             Voice +
                             Harmony:Voice +
                             self_declared_musician +
                             as.factor(KnowAxis) +
                             X1990s2000s.minus.1960s1970s +
                             as.factor(GuitarPlay) +
                             as.factor(X2ndInstr) +
                             self_declared_musician:as.factor(GuitarPlay) +
                             (1 + Instrument + Harmony + as.factor(KnowAxis) | Subject),
                           data = ratings,
                           REML = FALSE)

multilevel_model_24 <- lmer(Classical ~
                             Instrument +
                             Harmony +
                             Voice +
                             Harmony:Voice +
                             self_declared_musician +
                             as.factor(KnowAxis) +
                             X1990s2000s.minus.1960s1970s +
                             as.factor(GuitarPlay) +
                             as.factor(X2ndInstr) +
                             self_declared_musician:as.factor(X2ndInstr) +
                             (1 + Instrument + Harmony + as.factor(KnowAxis) | Subject),
                           data = ratings,
                           REML = FALSE)

AIC_BIC <- cbind(AIC = sapply(list(multilevel_model_17,
                                   multilevel_model_18,
                                   multilevel_model_19,
                                   multilevel_model_20,
                                   multilevel_model_21,
                                   multilevel_model_22,
                                   multilevel_model_23,
                                   multilevel_model_24),
                              AIC),
```

```r
                 BIC = sapply(list(multilevel_model_17,
                                   multilevel_model_18,
                                   multilevel_model_19,
                                   multilevel_model_20,
                                   multilevel_model_21,
                                   multilevel_model_22,
                                   multilevel_model_23,
                                   multilevel_model_24),
                                  BIC))

AIC_BIC <- t(as.tibble(AIC_BIC))
colnames(AIC_BIC) <- c("multilevel_model_17",
                       "multilevel_model_18",
                       "multilevel_model_19",
                       "multilevel_model_20",
                       "multilevel_model_21",
                       "multilevel_model_22",
                       "multilevel_model_23",
                       "multilevel_model_24")
kable(t(AIC_BIC))
```

| | AIC | BIC |
|---|---|---|
| multilevel_model_17 | 1179.325 | 1361.065 |
| multilevel_model_18 | 1176.912 | 1365.921 |
| multilevel_model_19 | 1176.076 | 1368.720 |
| multilevel_model_20 | 1180.091 | 1369.101 |
| multilevel_model_21 | 1179.325 | 1361.065 |
| multilevel_model_22 | 1179.325 | 1361.065 |
| multilevel_model_23 | 1179.325 | 1361.065 |
| multilevel_model_24 | 1179.325 | 1361.065 |

### 1.37 Summary output of final model

```r
summary(multilevel_model_9)
```

```
## Linear mixed model fit by maximum likelihood  ['lmerMod']
## Formula:
## Classical ~ Instrument + Harmony + Voice + Harmony:Voice + as.factor(Selfdeclare) +
##     as.factor(KnowAxis) + X1990s2000s.minus.1960s1970s + as.factor(GuitarPlay) +
##     as.factor(X2ndInstr) + (1 + Instrument + Harmony + as.factor(KnowAxis) |
##     Subject)
##    Data: ratings
##
##      AIC      BIC   logLik deviance df.resid
##   1178.3   1360.0   -539.1   1078.3      230
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.0010 -0.6266 -0.0263  0.6008  5.6514
```

```
##
## Random effects:
##  Groups   Name                  Variance  Std.Dev. Corr
##  Subject  (Intercept)           0.2091199 0.45730
##           Instrumentpiano       0.0027190 0.05214   0.24
##           Instrumentstring      2.2144858 1.48811  -1.00 -0.23
##           HarmonyI-V-IV         0.1639360 0.40489  -0.61  0.62  0.61
##           HarmonyI-V-VI         1.8800964 1.37117   0.70 -0.53 -0.70 -0.99
##           HarmonyIV-I-V         0.5523051 0.74317  -0.63  0.60  0.64  1.00 -1.00
##           as.factor(KnowAxis)5 0.0006984 0.02643  -1.00 -0.28  1.00  0.57 -0.67
##  Residual                       2.4583357 1.56791
##
##
##
##
##
##
##
##   0.60
##
## Number of obs: 280, groups:  Subject, 8
##
## Fixed effects:
##                              Estimate Std. Error t value
## (Intercept)                   3.52670    0.57383   6.146
## Instrumentpiano               1.71037    0.23450   7.294
## Instrumentstring              3.38542    0.57274   5.911
## HarmonyI-V-IV                 0.53014    0.48404   1.095
## HarmonyI-V-VI                 2.47032    0.66680   3.705
## HarmonyIV-I-V                 0.58163    0.53190   1.093
## Voicepar3rd                  -0.15468    0.45783  -0.338
## Voicepar5th                   0.47826    0.46235   1.034
## as.factor(Selfdeclare)2      -1.00963    0.52279  -1.931
## as.factor(Selfdeclare)3       0.97850    0.40007   2.446
## as.factor(Selfdeclare)4      -0.64329    0.56881  -1.131
## as.factor(KnowAxis)5          0.99503    0.48702   2.043
## X1990s2000s.minus.1960s1970s  0.04429    0.05283   0.838
## as.factor(X2ndInstr)1        -0.06253    0.40211  -0.156
## as.factor(X2ndInstr)2        -2.30108    0.63015  -3.652
## HarmonyI-V-IV:Voicepar3rd    -0.62793    0.65067  -0.965
## HarmonyI-V-VI:Voicepar3rd    -1.00402    0.64820  -1.549
## HarmonyIV-I-V:Voicepar3rd    -0.01923    0.65067  -0.030
## HarmonyI-V-IV:Voicepar5th    -0.74642    0.65071  -1.147
## HarmonyI-V-VI:Voicepar5th    -0.98479    0.65094  -1.513
## HarmonyIV-I-V:Voicepar5th    -1.50624    0.65076  -2.315
## fit warnings:
## fixed-effect model matrix is rank deficient so dropping 3 columns / coefficients
## optimizer (nloptwrap) convergence code: 0 (OK)
## boundary (singular) fit: see help('isSingular')
```

```
AIC_BIC_DIC <- cbind(AIC = sapply(list(model_2,
                                       multilevel_model_1,
                                       multilevel_model_6,
                                       multilevel_model_9), AIC),
```

```
                     BIC = sapply(list(model_2,
                                       multilevel_model_1,
                                       multilevel_model_6,
                                       multilevel_model_9), BIC),

                     DIC = sapply(list(multilevel_model_1,
                                       multilevel_model_1,
                                       multilevel_model_6,
                                       multilevel_model_9),
                     extractDIC))

AIC_BIC_DIC <- t(as.tibble(AIC_BIC_DIC))
AIC_BIC_DIC[3, 1] = AIC_BIC_DIC[1, 1]

colnames(AIC_BIC_DIC) <- c("Best Linear Model",
                           "MLM (random intercept)",
                           "MLM (random intercept and slopes)",
                           "Best Multilevel Model")

kable(t(AIC_BIC_DIC), digits = 2)
```

| | AIC | BIC | DIC |
|---|---|---|---|
| Best Linear Model | 11226.94 | 11314.26 | 11226.94 |
| MLM (random intercept) | 10458.10 | 10551.24 | 10426.10 |
| MLM (random intercept and slopes) | 9937.27 | 10146.84 | 9865.27 |
| Best Multilevel Model | 1178.30 | 1360.04 | 1078.30 |