Factors Determining Classical or Popular Music

Zhenyin Wu
zhenyinw@andrew.cmu.edu

8 December 2019

Abstract

We aim to build a valid model to help music researchers confirm their hypothesis on the factors that will influence listeners' identification of music as classical or popular. We examine the data on a designed experiment taken with 70 undergraduates at the University of Pittsburgh in 2012. In consideration of the "personal biases" exist in the experiment result, we used multilevel model to fit the dataset. The final model, we believe, is most clearly indicated by the data and validates the hypothesis of researchers. In the future, cooperating with music scholars and inviting them to involve in the factor selection and interpretation parts of modeling may yield more insightful results.

1 Introduction

Music is everywhere in our lives – in phones, in movies, in television commercials, in schools, and even in our memories. However, how many of us could distinguish whether the music we are listening is popular or classical. The question is probably difficult for people who have learned music, who play instruments or who are professional musicians, because classical music and popular music share many aspects of musical language and yet have some prominent differences as well. Therefore, our goal is to find out what are the most important factors that influence listeners' identification of music as classical or popular.

We answered this question from a statistical perspective, using data analysis and modeling. The unit of sampling in the data we collected is individual listener and their identification of music's category is represented by their ratings on classical and popular. Specifically, we will address the following three questions in this article:

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

2 Methods

The data for this study is taken from Vincent Rossi's experiment in 2012[1]. The experiment measured the influence of instrument, harmonic motion, and voice leading on listeners' identification of music as "classical" or "popular". The dataset contains music related information of 70 listeners and their classical and popular ratings on 36 musical stimuli. The variables X1stInstr and X2ndInstr were dropped from the dataset because 60% data of X1stInstr and 87% data of X2ndInstr were missing. We found that the two dependent variables Classical and Popular both have 27 missing values out of 2520 total observations. Since the missing data was a fairly small proportion, we just removed the 27 rows containing null dependent variables. For all the other variables containing missing data, we interpolate them with mode, because mode works well with categorical features. Since the rating range is from 1 to 10, we also dropped the rows with invalid rating values that are not in the range and not integers. Finally, we had 22 predictor variables and 2 response variables. The definitions of these variables are given in Table 1:

Variable Name	**Description**
Classical	How classical does the stimulus sound? (*Repose Variable*)
Popular	How popular does the stimulus sound? (*Repose Variable*)
Subject	Unique subject ID
Harmony	Harmonic Motion (4 levels: I-V-vi, I-VI-V, I-V-IV, IV-I-V)
Instrument	Instrument (3 levels: String Quartet, Piano, Electric Guitar)
Voice	Voice Leading (3 levels: Contrary Motion, Parallel 3rds, Parallel 5ths)
Selfdeclare	Are you a musician? (1-6, 1=not at all)
Log(OMSI)	Log 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 prev. 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 Pachelbel Rant (0-5, 0=not at all)

[1] Original source: Sibelius Institute, University of the Arts, Helsinki Finland.

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 prev 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)

Table 1: Variables Description

First, we fit a linear regression to only the three main factor, Harmony, Instrument and Voice. We used the backward stepwise AIC to choose which interaction term of them should be included. Lastly, we used ANOVA test to choose between the original model and the model with interaction. The ANOVA test result and the selection process of the linear model can be found in Appendix part (b) Linear Model for Classical Rating on p.15.

Second, we made a repeated measures model by fitting a random intercept for each participant. The summary of this repeated measures model could be found in Appendix part (c) Random Effect for Classical Rating on p.19. We compared the AIC and BIC of this model with the previous linear model to determine whether the random intercept is needed. Next, we used "fitLMER.fnc()" [2]function to decide what other random effect we should add to current model by forward-fitting. The automatic selection details are shown in Appendix part (c) Random Effect for Classical Rating, and the summary of the result model is shown on p.22. Finally, we used the AIC and BIC to compare the multilevel model with random effect on intercept and new factors, with the multilevel model with random effect on intercept only.

Third, we applied backward stepwise AIC on the all the 22 predictor variables to select other fixed effect variables. The selecting output is shown in Appendix part (d) Multilevel Model for Classical Rating on p. 27. After selection, we used the multilevel model to fit all these selected fixed variables with the random effect we selected in step above together. This model is our final model for classical rating.

[2] R function; Back-fit fixed effects and forward-fit random effects of an LMER Model.

We used exactly the same three steps to get the final multilevel model for popular rating. The details are provided in Appendix part (f) Multilevel Model for Popular Rating on p. 37.

For the second question, according to the rule that variable "selfdeclare" is less than or equal to 2 as a non-musician and greater than 2 is a musician, we replace "selfdeclare" with a variable "is.musician". Then we put the interaction of is.musician with all the other variables into the lmer model, and automatically selected the variables to get the model (6). The modeling details could be found in Appendix part (e) Differences between Musicians and non-Musicians on p. 30.

3 Results

3.1 Factors Influencing Rating

To answer our first question, we need to find out the model that best fits our dataset. We first consider whether there are variables that need to be transformed. The histogram plots for all the continuous numeric variables is shown in Appendix part (a) on p. 13. From the histogram plot we could see that the distribution of OMSI is right skewed, so we applied log transformation to fix it. We also checked the correlation between continuous variables with Repose Variables, we didn't find highly related pairs of variables.

The final linear regression on the three design factors, Harmony, Instrument and Voice is:

$$Classical \sim Instrument + Harmony + Voice + Harmony{:}Voice \tag{1}$$

The summary and the diagnostic plots of the model 1 could be found in Appendix part (b)Linear Model for Classical Rating on p.18. The linear regression fits well, because the residual plot is almost a flat line, which means the residuals are normally distributed, in other words, it has constant variance. For the QQ-plot, the line is very straight and close to 45 degrees, so we still think the residuals are close to normal distribution. For the leverage plot, we found that all points are fall into the Cook's distance, so there is no bad leverage point. According the summary table, instrument exert the strongest influence among the three design factors, because the coefficients of the two instrument variables are the largest. Among the levels of Harmonic Motion, I-V-vi has the strongest association with classical rating, because only this level is significant; the other two levels are not significant.

Although either the diagnostic plots or the interpretability of variables of model 1 are great, we considered that everyone has a different understanding and definition of classic and popular, so that there are many biases in the linear model, thus we decided to try multi-level model to fit a better model.

The equation of multilevel model with random effect on intercept is:

$$Classical \sim Instrument + Harmony + Voice + (1 \mid Subject) \tag{2}$$

The equation of multilevel model with random effect on intercept, harmony and instrument is:

$$Classical \sim Harmony + Instrument + Voice + (1 \mid Subject) + (Harmony \mid Subject) + (Instrument \mid Subject) \quad (3)$$

The AIC and BIC result of model 1, 2 and 3:

	Model 1	**Model 2**	**Model 3**
AIC	10990.4	10220.34	9735.298
BIC	11077.48	10278.39	9886.23

Table 2: AIC and BIC

From the table above we could see that model 3 has the lowest AIC and BIC among the three. Therefore, we think that in our multilevel mode, including random effect on intercept only is not enough. In another words, in order to achieve the lowest AIC and BIC, considering personal biases as well as the part of personal biases which vary with the type of instrument and type of harmony is necessary. Thus, we determined to add these three random effects on intercept, harmony and instrument in our final models.

The equation of our final model for classical rating:

$$Classical \sim Voice + Instrument + Harmony + Selfdeclare + OMSI + X16.minus.17 + PachListen + ClsListen + X1990s2000s + X1990s2000s.minus.1960s1970s + Composing + PianoPlay + Harmony{:}Voice + (1 + Harmony + Instrument \mid Subject) \quad (4)$$

The fixed effect summary of our final multilevel model, model 4 is:

	Estimate	*Std. Error*	*t-value*
(Intercept)	3.55161	0.86401	4.111
Voicepar3rd	-0.21773	0.15117	-1.440
Voicepar5th	-0.17562	0.15206	-1.155
Instrumentpiano	1.37023	0.17460	7.848
Instrumentstring	3.07986	0.23596	13.053
HarmonyI-V-IV	0.16252	0.15489	1.049
HarmonyI-V-VI	1.19195	0.21768	5.476
HarmonyIV-I-V	-0.10109	0.15187	-0.666
Selfdeclare	-0.44400	0.17410	-2.550
OMSI	0.21006	0.15339	1.369
X16.minus.17	-0.07276	0.04633	-1.570
PachListen	0.10496	0.13108	0.801

ClsListen	0.31205	0.10885	2.867
X1990s2000s	-0.24789	0.10478	-2.366
X1990s2000s.minus.1960s1970s	0.21479	0.09398	2.286
Composing	0.15053	0.12005	1.254
PianoPlay	0.14714	0.09129	1.612
Voicepar3rd:HarmonyI-V-IV	-0.33513	0.21453	-1.562
Voicepar5th:HarmonyI-V-IV	-0.20535	0.21517	-0.954
Voicepar3rd:HarmonyI-V-VI	-0.71722	0.21460	-3.342
Voicepar5th:HarmonyI-V-VI	-0.48100	0.21509	-2.236
Voicepar3rd:HarmonyIV-I-V	0.49819	0.21449	2.323
Voicepar5th:HarmonyIV-I-V	0.01457	0.21434	0.068

Table 3: Final Model Summary of Fixed Effect

Random effects of Harmony

Name	*Variance*	*Std. Dev.*	*Corr*				
HarmonyI-IV-V	2.166	1.472					
HarmonyI-V-IV	2.773	1.665	1.00				
HarmonyI-V-VI	3.061	1.750	0.70	0.74			
HarmonyIV-I-V	2.242	1.497	1.00	1.00	0.69		
Instrumentpiano	1.678	1.295	-0.38	-0.43	-0.50	-0.41	
Instrumentstring	3.387	1.840	-0.57	-0.58	-0.77	-0.54	0.67

Full levels Fixed effects of Harmony

	Estimate	*Std. Error*	*t-value*
HarmonyI-IV-V	3.52074	0.86001	4.094
HarmonyI-V-IV	3.68279	0.86536	4.256
HarmonyI-V-VI	4.71223	0.86787	5.430
HarmonyIV-I-V	3.41964	0.86042	3.974

Table 4: Random Effect of Harmony

Random effects of Instrument

Name	*Variance*	*Std. Dev.*	*Corr*				
Instrumentguitar	2.15789	1.4690					
Instrumentpiano	2.37055	1.5397	0.63				
Instrumentstring	2.46136	1.5689	0.27	0.61			
HarmonyI-V-IV	0.06234	0.2497	0.75	0.14	0.07		
HarmonyI-V-VI	1.64864	1.2840	-0.20	-0.39	-0.64	0.28	
HarmonyIV-I-V	0.01234	0.1111	0.20	-0.02	0.64	0.53	-0.16

Full levels Fixed effects of Instrument

	Estimate	*Std. Error*	*t-value*
Instrumentguitar	3.55172	0.86403	4.111
Instrumentpiano	4.92195	0.86579	5.685
Instrumentstring	6.63158	0.86727	7.646

Table 5: Random Effect of Instrument

Full levels Fixed effects of Voice

	Estimate	*Std. Error*	*t-value*
Voicecontrary	3.55172	0.86403	4.111
Voicepar3rd	3.33399	0.86380	3.860
Voicepar5th	3.37610	0.86400	3.908

Table 6: Full levels Fixed effects of Voice

Instruments exert the strongest influence among the three design factors, because the variables Instrument piano and Instrument string have the largest coefficients and t-values. From Table 3, for fixed effect, we can see the coefficient of piano is about 1.37 and string is about 3.08, which means if the music is performed by piano, the classical rating would approximately increase by 1.37 points, and if the music is performed by string, the classical rating would almost increase by 3.08 points. There is random variability across different listeners in the degree to which they are inclined to call music played by the three instruments "classical". For example, according to Table 5, the standard deviation of guitar is 1.469 and the coefficient of guitar is 3.55 so within 2 standard deviations, the coefficient of guitar is positive. Thus, guitar has a positive association with the classical rating for majority of the participants. By the same logic, we found that for 95% of the participants, the instrument piano and string have positive association with classical ratings with consideration of the influence of random effect. In conclusion, instrument exert the strongest and positive influence among the three design factors.

Among the levels of Harmonic Motion, I-V-vi has the strongest association with classical rating, because only this level is significant. From the table3, we can see that the other two levels are not significant. Among the interaction terms between voice and harmony, we also found that the interactions with Harmony I-V-VI are also significant, because the t-value of Voicepar3rd:HarmonyI-V-VI is -3.342 and the t-value of Voicepar5th:HarmonyI-V-VI is -2.236. For the random effect of Harmony, according to Table 4, the estimates of Harmony I-V-VI is largest, which equals 4.71 and we could also see that Harmony I-V-VI has the largest variance, which equals 3.061. Random effect is the variation produced by each subject's different understanding of variables. In this case, the random effect of Harmony-V-VI means personal bias on the relationship between harmony-V-VI and classical rating. Thus, the larger variance means the rating will be more different among subjects. Therefore, by all these evidences, Harmony I-V- VI has the strongest association. Besides, this fact does not seem to matter whether the respondent is familiar with both Pachelbel rants and comedy bits, because these two variables, “KnowRob” and “KnowAxis” are both not significant and not selected into our model.

Among the levels of Voice Leading, contrary motion has the strongest association with classical ratings. From Table 6, the estimate of Voice contrary is 3.55172 which is slightly larger than Voicepar3rd, 3.33399 and Voicepar5th, 3.37610. Besides, the Voice contrary is more significant than Voicepar3rd and Voicepar5th, because its t-value is also larger than them. Therefore, among the levels of Voice Leading, contrary motion has the strongest association with classical ratings, although the difference is not very significant.

With the same steps, we made the multilevel model for popular rating. The equation of our final model for popular rating is:

$$\begin{aligned} Popular \sim Harmony &+ Instrument + Voice + Selfdeclare + X16.minus.17 \\ &+ ConsInstr + ConsNotes + Instr.minus.Notes + PachListen \\ &+ ClsListen + KnowRob + X1990s2000s \\ &+ X1990s2000s.minus.1960s1970s + APTheory + Composing \\ &+ GuitarPlay + (1 + Harmony + Instrument \mid Subject) \end{aligned} \tag{5}$$

Similarly, the fixed effect summary and the random effect tables of the popular rating model could be found in appendix part (f) Multilevel Model for Popular Rating on p. 46. Instrument piano and string still exert the strongest influence among the three design factors, but the association with popular rating becomes negative relationship. If the music is performed with piano, the expected popular rating will be lower by about 1 point and if the music is performed with string, the popular will decrease by about 2.6 points. After considering the influence of random effect, the instrument guitar, piano and string all have negative association with popular rating.

3.2 Musicians, vs. non-Musicians

The final model including variable is.musician is:

$$\begin{aligned} Classical \sim is.musician &+ Harmony + Instrument + ConsInstr + Instr.minus.Notes \\ &+ X1990s2000s + CollegeMusic + APTheory + PianoPlay + GuitarPlay \\ &+ (1 + Harmony + Instrument + Voice \mid Subject) \\ &+ is.musician{:}Harmony + is.musician{:}X16.minus.17 \\ &+ is.musician{:}CollegeMusic + is.musician{:}APTheory \\ &+ is.musician{:}PianoPlay + is.musician{:}GuitarPlay \end{aligned} \tag{6}$$

The summary table of model 6 is:

	Estimate	*Std. Error*	*t-value*
(Intercept)	4.346023	0.599988	7.244
is.musician1	1.685818	0.694993	2.426
HarmonyI-V-IV	-0.129642	0.120821	-1.073

HarmonyI-V-VI	0.044161	0.188946	0.234
HarmonyIV-I-V	0.049095	0.114551	0.429
Instrumentpiano	1.526482	0.168114	9.080
Instrumentstring	3.440247	0.217507	15.817
ConsInstr	-0.147146	0.084476	-1.742
Instr.minus.Notes	0.148438	0.082230	1.805
X1990s2000s	-0.230760	0.091350	-2.526
CollegeMusic1	0.967008	0.449087	2.153
APTheory1	1.697638	0.694235	2.445
PianoPlay	-0.258047	0.201194	-1.283
GuitarPlay	1.516994	0.516674	2.936
is.musician1:HarmonyI-V-IV	0.006901	0.188879	0.037
is.musician1:HarmonyI-V-VI	1.156735	0.293992	3.935
is.musician1:HarmonyIV-I-V	-0.020950	0.179803	-0.117
is.musician0:X16.minus.17	-0.085374	0.053098	-1.608
is.musician1:X16.minus.17	-0.299528	0.087109	-3.439
is.musician1:CollegeMusic1	-1.543498	0.663394	-2.327
is.musician1:APTheory1	-1.640124	0.849191	-1.931
is.musician1:PianoPlay	0.232135	0.227522	1.020
is.musician1:GuitarPlay	-1.370333	0.526869	-2.601

Table 7: Musicians, vs. non-Musicians Output

From table 7, we can see that variable is.musician is significant, which means the classical rating of musicians is approximately 1.69 points higher than that of non-musicians. We think the explanation is that musicians usually have a stronger classical music knowledge background, so they have a better ability to distinguish classical from popular, therefore they will give a music higher classical score when they think it is classical. Among the interaction variable, we found "is.musician1:HarmonyI-V-VI", "is.musician1:X16.minus.17", "is.musician1:CollegeMusic1" and "is.musician1:GuitarPlay" are significant. These factors show the differences in the way that musicians and non-musicians identify classical music:

- Musicians will give the music with Harmony I-V-VI about 1.16 points higher than non-Musicians
- The "X16.minus.17" measures of Musicians increase by one unit, the rating for classical point will decrease about 0.3 points.
- Musicians who took AP Music Theory class in High School give the classical rating about 1.64 points lower.
- Musicians who play guitar will give the music classical rating approximately 1.37 points lower.

It is not difficult to find that because of their musical knowledge, musicians can clearly distinguish the Harmony IV-VI variable, which has a positive effect on classical rating (that we have previously analyzed), higher classic rating. Besides, it is also interesting to notice that musicians who play guitar have a lower classical rating. The reason may be because the guitar is the mainstream instrument of popular music, so they could distinguish popular music better and will give higher score on popular. In conclusion, musicians and non-musicians identify classical music differently based on whether the music has Harmony I-V-VI, their “X16.minus.17” score and whether they learned AP Music Theory or play guitar.

3.3 Classical, vs. Popular

To answer the question “Are there differences in the things that drive classical, vs. popular, ratings”, we should precisely compare the summary of model 4 and model 5.

For the three design factors, Instruments have positive association with classical rating, while have negative association with popular rating; voices have negative association with classical rating, while have positive association with popular rating; HarmonyI-V-IV and HarmonyI-V-VI have positive association with classical rating, while have negative association with popular rating; HarmonyIV-I-V has negative association with both classical and popular ratings, but this variable is the least significant. These findings are in line with our expectations, because the factors driving classical rating increase should theoretically drive popular rating decrease.

For the other factors, selfdeclare, ClsListen and X1990s2000s.minus.1960s1970s are significant in classical rating but not significant in popular rating. Selfdeclare represents if the participant is a musician and ClsListen represents how much classical music does the participant listen. These two variables represent the musical knowledge of the participants and their familiarity with classical music. The people who have more music knowledge and familiarity with classical music are more likely to give a music which they believe classical a higher classical rating. The interaction factors of voice and harmony are kept in the model of classical but are dropped in the model of popular. KnowRob is selected into the model for popular but not into the model of classical. Finally, classical model chooses PianoPlay but the popular model chooses GuitarPlay. This choice is consistent with our common sense. Popular music will use more instruments such as guitars, while classical music will use more classical instruments such as pianos.

4 Discussion

We used a dataset from an experiment, which lets 70 listeners rate on 36 musical stimuli, to build multilevel models analyzing the influence of instrument, harmonic motion, and voice leading on listeners’ identification of music as “classical” or “popular”. Before the model building, we carefully cleaned the dataset, analyzed the correlation between variables and made transform on variables. We chose multilevel model to fit the dataset because we think by adding random intercept and random effect on specific variable can well account for the different understanding and definition of classic and popular of participants, and also personal biases vary with factors.

The most fulfilling part of this article is that we found two reliable and interpretable models for classical and popular ratings respectively. According to our model, we can find that the three main design factors, instrument, harmonic motion, and voice are all interpretable and showing correct associations with the rating. We found that for either classical or popular, instruments exert the strongest influence on participants' rating. The using of string and piano will increase the classical rating and decrease the popular rating. Among all the 4 levels of harmony motion, we could see that Harmony I-V-VI has the strongest association with classical rating. Although the difference is not particularly noticeable, contrary motion are the most significant and influential level in Voice Leading. All findings from our model are in consideration with both fixed effect and random effect. All results we get from the models are in consistence with researchers' suspect.

However, there are some weakness of our models. For both final models of classical and popular, there are some non-significant variables. Therefore, in the process of building our model, we may still have room for improvement. For example, in addition to the automatic method of selecting variables, we could have added some manual variable selecting processes, combining the actual meaning of each variable and the knowledge of music, to make our model fit better. Secondly, some variables in the model cannot be explained with common sense. For example, in the model 6, we found that the interaction term "is.musician1:CollegeMusic1" is significant, so we said "musicians who took AP Music Theory class in High School give the classical rating about 1.64 points lower". However, the APTheory1 itself has a positive association with classical rating. Thus, it is very hard to explain why musicians who took APtheory will decrease their classical rating.

The main value of this article is that it validates the hypothesis of music researchers through statistical models. For music lovers or professional musicians, when they want to distinguish between classic and popular music, they might as well first discern what instrument this music is played on. If they have music knowledge, they could try to discern what particular harmonic progression is in this music for the next step. And finally, they could think if there is contrary motion in the music to help them make their final identification. In the future, if we have the opportunity, we might invite people who have knowledge of music to join us in the process of selecting variables and use their music knowledge to further judge how well our model is performing.

Reference

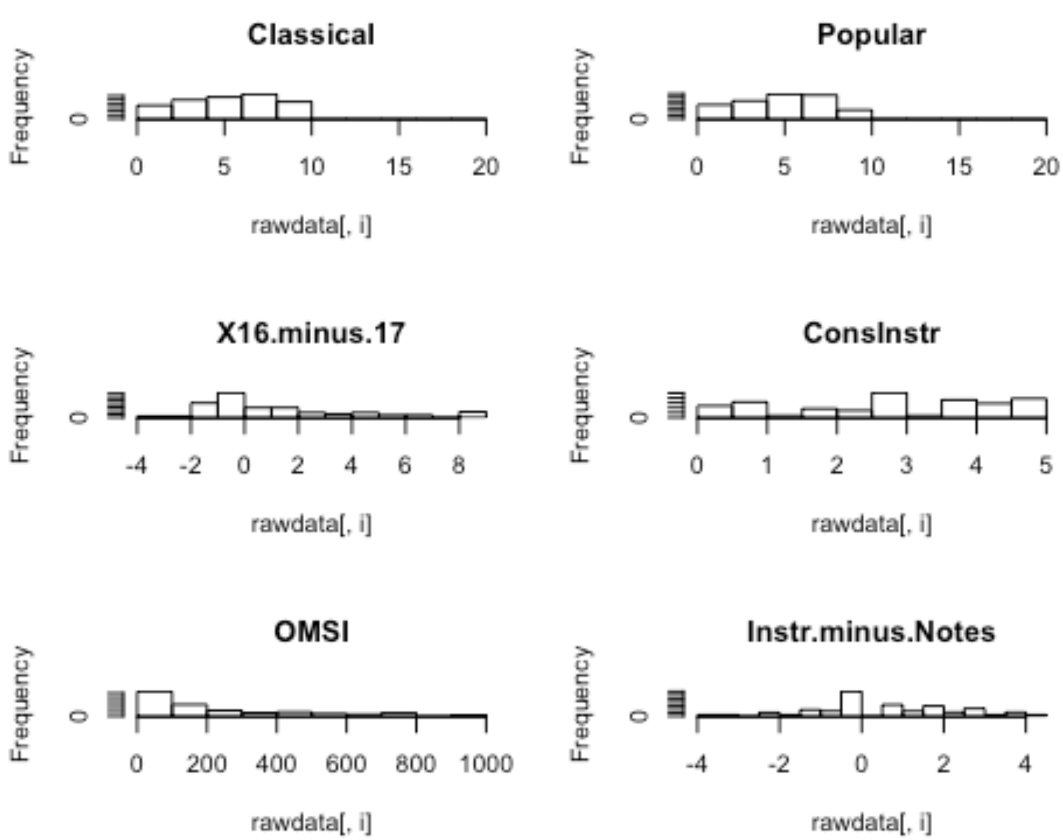
R Core Team (2017), *R: A language and environment for statistical computing*. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/.

Appendix

(a) Data Cleaning and Preparation

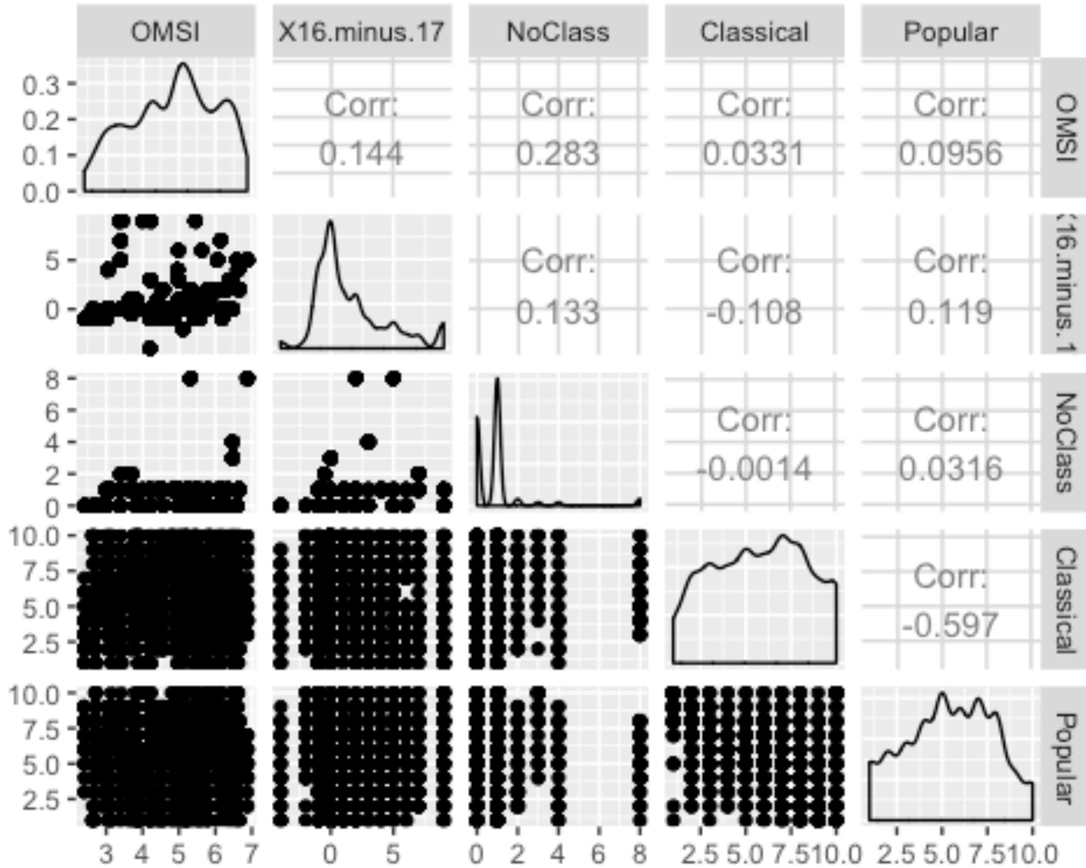
```
rawdata = read.csv('ratings.csv')
rawdata$X <- NULL
rawdata$first12 <- NULL

#drop two columns with too many na values
rawdata$X1stInstr <- NULL
rawdata$X2ndInstr <- NULL

#drop the 27 missting data
miss_popular <- which(is.na(rawdata$Popular))
miss_classical <- which(is.na(rawdata$Classical))
rawdata <- rawdata[-miss_popular, ]

#Fill other na with mode
mode <- function(x) {
  ux <- unique(x)
  ux[which.max(tabulate(match(x, ux)))]
}
rawdata$ConsNotes[is.na(rawdata$ConsNotes)] <- mode(rawdata$ConsNotes[!
is.na(rawdata$ConsNotes)])
rawdata$PachListen[is.na(rawdata$PachListen)] <- mode(rawdata$PachListe
n[!is.na(rawdata$PachListen)])
rawdata$ClsListen[is.na(rawdata$ClsListen)] <- mode(rawdata$ClsListen[!
is.na(rawdata$ClsListen)])
rawdata$KnowRob[is.na(rawdata$KnowRob)] <- mode(rawdata$KnowRob[!is.na
(rawdata$KnowRob)])
rawdata$KnowAxis[is.na(rawdata$KnowAxis)] <- mode(rawdata$KnowAxis[!is.
na(rawdata$KnowAxis)])
rawdata$X1990s2000s[is.na(rawdata$X1990s2000s)] <- mode(rawdata$X1990s2
000s[!is.na(rawdata$X1990s2000s)])
rawdata$X1990s2000s.minus.1960s1970s[is.na(rawdata$X1990s2000s.minus.19
60s1970s)] <- mode(rawdata$X1990s2000s.minus.1960s1970s[!is.na(rawdata
$X1990s2000s.minus.1960s1970s)])
rawdata$CollegeMusic[is.na(rawdata$CollegeMusic)] <- mode(rawdata$Colle
geMusic[!is.na(rawdata$CollegeMusic)])
```

```r
rawdata$NoClass[is.na(rawdata$NoClass)] <- mode(rawdata$NoClass[!is.na
(rawdata$NoClass)])
rawdata$APTheory[is.na(rawdata$APTheory)] <- mode(rawdata$APTheory[!is.
na(rawdata$APTheory)])
rawdata$X1stInstr[is.na(rawdata$X1stInstr)] <- mode(rawdata$X1stInstr[!
is.na(rawdata$X1stInstr)])
rawdata$X2ndInstr[is.na(rawdata$X2ndInstr)] <- mode(rawdata$X2ndInstr[!
is.na(rawdata$X2ndInstr)])

#plot histograms for raw data
par(mfrow=c(3,2))
for (i in c('Classical','Popular','X16.minus.17', 'ConsInstr', 'OMSI','
Instr.minus.Notes')) {
    hist(rawdata[,i],main=i)
}
```




```r
#chaneg data type
data <- rawdata

```

```r
data[,c("APTheory","CollegeMusic")] =
sapply(data[,c("APTheory","CollegeMusic")], as.factor)

# the range of rating is between 1 and 10, so we drop thoes invalid rati
ngs
data <- data[!(data$Classical < 1 | data$Classical > 10 |
                     data$Popular < 1 | data$Popular > 10), ]


#drop the decimal values of ratings
data <- data[data$Classical == as.integer(data$Classical) &
                    data$Popular == as.integer(data$Popular), ]


# transform variable OMSI with log()
data$OMSI = log(data$OMSI)

#eda for numeric veriables
numdata = data[,c("OMSI", "X16.minus.17", "NoClass", "Classical", "Popu
lar")]
ggpairs(numdata)
```




```
#check structure of final data
str(data)
```

```
## 'data.frame':    2453 obs. of  24 variables:
##  $ Subject                   : Factor w/ 70 levels "15","16","17",..:
 1 1 1 1 1 1 1 1 1 1 ...
##  $ Harmony                   : Factor w/ 4 levels "I-IV-V","I-V-IV
",..: 1 1 1 1 1 1 1 1 1 2 ...
##  $ Instrument                : Factor w/ 3 levels "guitar","piano
",..: 1 1 1 2 2 2 3 3 3 1 ...
##  $ Voice                     : Factor w/ 3 levels "contrary","par3rd
",..: 1 2 3 1 2 3 1 2 3 1 ...
##  $ Selfdeclare               : int  5 5 5 5 5 5 5 5 5 5 ...
##  $ OMSI                      : num  6.6 6.6 6.6 6.6 6.6 ...
##  $ X16.minus.17              : num  5 5 5 5 5 5 5 5 5 5 ...
##  $ ConsInstr                 : num  4.33 4.33 4.33 4.33 4.33 4.33 4.3
3 4.33 4.33 4.33 ...
##  $ ConsNotes                 : int  5 5 5 5 5 5 5 5 5 5 ...
##  $ Instr.minus.Notes          : num  -0.67 -0.67 -0.67 -0.67 -0.67 -
```

```
0.67 -0.67 -0.67 -0.67 -0.67 ...
##  $ PachListen                 : int  5 5 5 5 5 5 5 5 5 5 ...
##  $ ClsListen                  : int  4 4 4 4 4 4 4 4 4 4 ...
##  $ KnowRob                    : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ KnowAxis                   : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ X1990s2000s                : int  5 5 5 5 5 5 5 5 5 5 ...
##  $ X1990s2000s.minus.1960s1970s: int  2 2 2 2 2 2 2 2 2 2 ...
##  $ CollegeMusic               : chr  "0" "0" "0" "0" ...
##  $ NoClass                    : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ APTheory                   : chr  "0" "0" "0" "0" ...
##  $ Composing                  : int  4 4 4 4 4 4 4 4 4 4 ...
##  $ PianoPlay                  : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ GuitarPlay                 : int  5 5 5 5 5 5 5 5 5 5 ...
##  $ Classical                  : num  3 3 1 3 2 8 10 6 5 1 ...
##  $ Popular                    : num  9 7 8 7 8 3 1 4 5 8 ...
```

(b) Linear Model for Classical Rating

```r
lm.1 <- lm(Classical ~ Harmony + Instrument + Voice , data = data)
summary(lm.1)
```

```
##
## Call:
## lm(formula = Classical ~ Harmony + Instrument + Voice, data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.8281 -1.7550 -0.0173  1.7323  6.0736
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)       4.35551    0.12939  33.662  < 2e-16 ***
## HarmonyI-V-IV    -0.04417    0.12981  -0.340 0.733670
## HarmonyI-V-VI     0.78436    0.12976   6.045 1.72e-09 ***
## HarmonyIV-I-V     0.05159    0.12955   0.398 0.690484
## Instrumentpiano   1.34128    0.11222  11.952  < 2e-16 ***
## Instrumentstring  3.04673    0.11226  27.140  < 2e-16 ***
## Voicepar3rd      -0.38492    0.11239  -3.425 0.000626 ***
## Voicepar5th      -0.35847    0.11233  -3.191 0.001434 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 2.272 on 2445 degrees of freedom
## Multiple R-squared:  0.2485, Adjusted R-squared:  0.2464
## F-statistic: 115.5 on 7 and 2445 DF,  p-value: < 2.2e-16
```

```r
lm.2 <- step(lm(Classical ~ Instrument * Harmony * Voice, data = data),
             direction = "backward")
```

```
## Start:  AIC=4054.97
## Classical ~ Instrument * Harmony * Voice
##
##                            Df Sum of Sq   RSS  AIC
## - Instrument:Harmony:Voice 12    61.106 12502 4043
## <none>                                  12441 4055
##
## Step:  AIC=4042.98
## Classical ~ Instrument + Harmony + Voice + Instrument:Harmony +
##     Instrument:Voice + Harmony:Voice
##
##                      Df Sum of Sq   RSS    AIC
## - Instrument:Harmony  6    10.962 12513 4033.1
## - Instrument:Voice    4     9.953 12512 4036.9
## <none>                            12502 4043.0
## - Harmony:Voice       6    93.107 12596 4049.2
##
## Step:  AIC=4033.13
## Classical ~ Instrument + Harmony + Voice + Instrument:Voice +
##     Harmony:Voice
##
##                    Df Sum of Sq   RSS    AIC
## - Instrument:Voice  4     9.977 12523 4027.1
## <none>                          12513 4033.1
## - Harmony:Voice     6    93.147 12606 4039.3
##
## Step:  AIC=4027.09
## Classical ~ Instrument + Harmony + Voice + Harmony:Voice
##
##                 Df Sum of Sq   RSS    AIC
## <none>                       12523 4027.1
## - Harmony:Voice  6      93.2 12617 4033.3
## - Instrument     2    3817.7 16341 4675.8
```

```r
summary(lm.2)
```

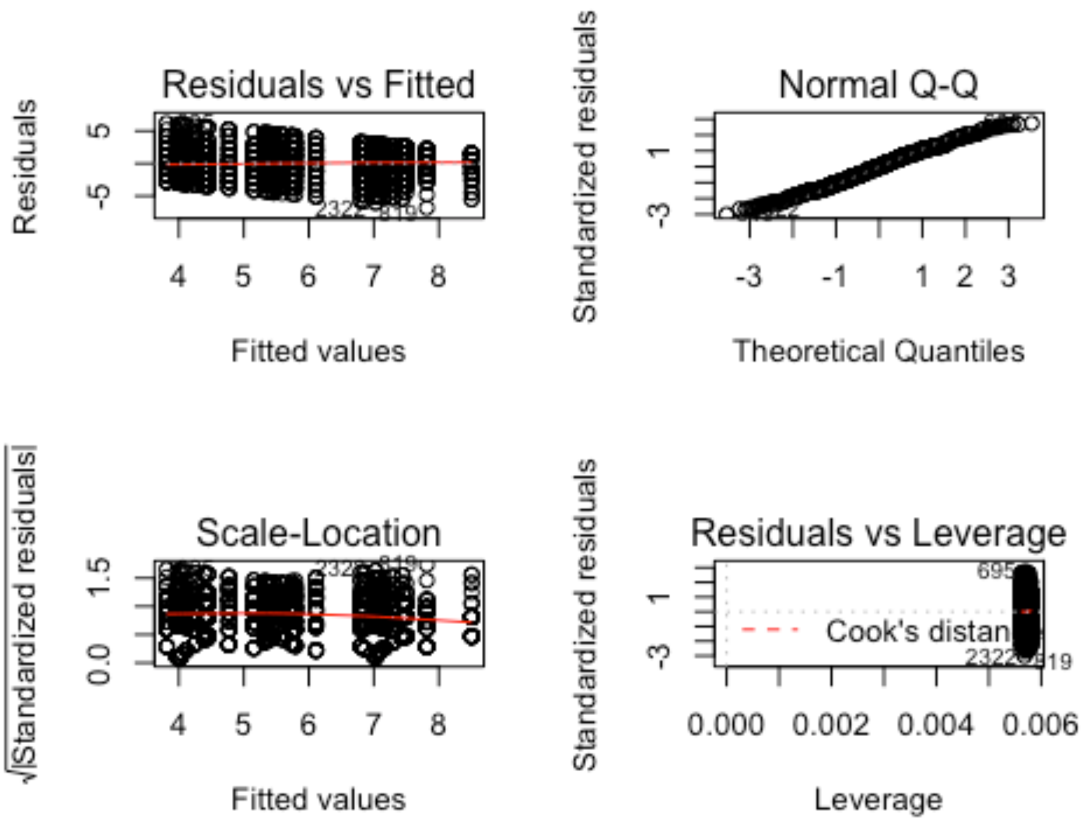
```
##
## Call:
## lm(formula = Classical ~ Instrument + Harmony + Voice + Harmony:Voic
e,
##     data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.8152 -1.7368 -0.0223  1.6753  6.1881
##
## Coefficients:
##                            Estimate Std. Error t value Pr(>|t|)
## (Intercept)                 4.24877    0.17052  24.917  < 2e-16 ***
## Instrumentpiano             1.34142    0.11195  11.982  < 2e-16 ***
## Instrumentstring            3.04699    0.11198  27.209  < 2e-16 ***
## HarmonyI-V-IV               0.14665    0.22410   0.654   0.5129
## HarmonyI-V-VI               1.21005    0.22438   5.393  7.6e-08 ***
## HarmonyIV-I-V              -0.13107    0.22327  -0.587   0.5572
## Voicepar3rd                -0.22871    0.22300  -1.026   0.3052
## Voicepar5th                -0.19369    0.22466  -0.862   0.3887
## HarmonyI-V-IV:Voicepar3rd -0.35478    0.31654  -1.121   0.2625
## HarmonyI-V-VI:Voicepar3rd -0.77804    0.31673  -2.456   0.0141 *
## HarmonyIV-I-V:Voicepar3rd  0.50344    0.31635   1.591   0.1116
## HarmonyI-V-IV:Voicepar5th -0.21846    0.31771  -0.688   0.4918
## HarmonyI-V-VI:Voicepar5th -0.49694    0.31752  -1.565   0.1177
## HarmonyIV-I-V:Voicepar5th  0.05127    0.31636   0.162   0.8713
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.266 on 2439 degrees of freedom
## Multiple R-squared:  0.2541, Adjusted R-squared:  0.2501
## F-statistic:  63.9 on 13 and 2439 DF,  p-value: < 2.2e-16
```

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

```
## Analysis of Variance Table
##
## Model 1: Classical ~ Harmony + Instrument + Voice
## Model 2: Classical ~ Instrument + Harmony + Voice + Harmony:Voice
##   Res.Df   RSS Df Sum of Sq      F   Pr(>F)
```

```
## 1   2445 12617
## 2   2439 12523  6     93.19 3.0249 0.005995 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
par(mfrow = c(2,2))
plot(lm.2)
```



(c) Random Effect for Classical Rating

```r
library(lme4)
library(RLRsim)
library(arm)
library(plyr)

m.0 <- lmer(Classical ~ Harmony + Instrument + Voice + (1|Subject),
         data = data, control = lmerControl(optimizer = "bobyqa"), REML=
F)
```

```r
display(m.0)
```

```
## lmer(formula = Classical ~ Harmony + Instrument + Voice + (1 |
##     Subject), data = data, REML = F, control = lmerControl(optimizer
= "bobyqa"))
##                   coef.est coef.se
## (Intercept)        4.34     0.19
## HarmonyI-V-IV     -0.04     0.11
## HarmonyI-V-VI      0.79     0.11
## HarmonyIV-I-V      0.06     0.11
## Instrumentpiano    1.35     0.09
## Instrumentstring   3.04     0.09
## Voicepar3rd       -0.39     0.09
## Voicepar5th       -0.36     0.09
##
## Error terms:
##  Groups   Name        Std.Dev.
##  Subject  (Intercept) 1.29
##  Residual             1.86
## ---
## number of obs: 2453, groups: Subject, 70
## AIC = 10220.3, DIC = 10200.3
## deviance = 10200.3
```

```r
m.1 <- fitLMER.fnc(m.0, method="AIC")
```

```
## =======================================================
## ===              backfitting fixed effects          ===
## =======================================================
## setting REML to FALSE
## processing model terms of interaction level 1
##   iteration 1
##     p-value for term "Voice" = 0 >= 0
##     not part of higher-order interaction
##     AIC simple = 10238; AIC complex = 10220; decrease = 18 >= 5
##     skipping term
## length = 3
##   iteration 2
##     p-value for term "Harmony" = 0 >= 0
##     not part of higher-order interaction
##     AIC simple = 10296; AIC complex = 10220; decrease = 76 >= 5
```

```
##     skipping term
## length = 2
##   iteration 3
##     p-value for term "Instrument" = 0 >= 0
##     not part of higher-order interaction
##     AIC simple = 11120; AIC complex = 10220; decrease = 899 >= 5
##     skipping term
## length = 1
## pruning random effects structure ...
##   nothing to prune
## ==========================================================
## ===            forwardfitting random effects       ===
## ==========================================================
##  ===      random slopes       ===
## ==========================================================
## ===            re-backfitting fixed effects        ===
## ==========================================================
## setting REML to FALSE
## processing model terms of interaction level 1
##   iteration 1
##     p-value for term "Voice" = 0 >= 0
##     not part of higher-order interaction
##     AIC simple = 10238; AIC complex = 10220; decrease = 18 >= 5
##     skipping term
## length = 3
##   iteration 2
##     p-value for term "Harmony" = 0 >= 0
##     not part of higher-order interaction
##     AIC simple = 10296; AIC complex = 10220; decrease = 76 >= 5
##     skipping term
## length = 2
##   iteration 3
##     p-value for term "Instrument" = 0 >= 0
##     not part of higher-order interaction
##     AIC simple = 11120; AIC complex = 10220; decrease = 899 >= 5
##     skipping term
## length = 1
## resetting REML to TRUE
## pruning random effects structure ...
##   nothing to prune
```

```
## log file is /var/folders/2z/10l5bnjj24n7r12mj3htn6_c0000gn/T//Rtmplt
8doo/fitLMER_log_Fri_Dec__6_12-56-28_2019.txt
```

```r
vars <- attr(terms(formula(m.1)),"term.labels")
vars <- vars[-length(vars)]

m.2 <- fitLMER.fnc(m.1,
                        ran.effects=
                          list(slopes=vars, by.vars="Subject",
                             corr=rep(1,length(vars))))
```

```
## ==========================================================
## ===             backfitting fixed effects        ===
## ==========================================================
## processing model terms of interaction level 1
##   all terms of interaction level 1 significant
## pruning random effects structure ...
##   nothing to prune
## ==========================================================
## ===           forwardfitting random effects      ===
## ==========================================================
##  ===       random slopes      ===
## evaluating addition of (Harmony|Subject) to model

## boundary (singular) fit: see ?isSingular

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

##  log-likelihood ratio test p-value = 3.232812e-18
##  adding (Harmony | Subject) to model
## evaluating addition of (Instrument|Subject) to model

## Warning in optwrap(optimizer, devfun, getStart(start, rho$lower, rho
$pp), :
## convergence code 1 from bobyqa: bobyqa -- maximum number of function
## evaluations exceeded

## boundary (singular) fit: see ?isSingular
## refitting model(s) with ML (instead of REML)

## Warning in optwrap(optimizer, devfun, x@theta, lower = x@lower, calc.
derivs
```

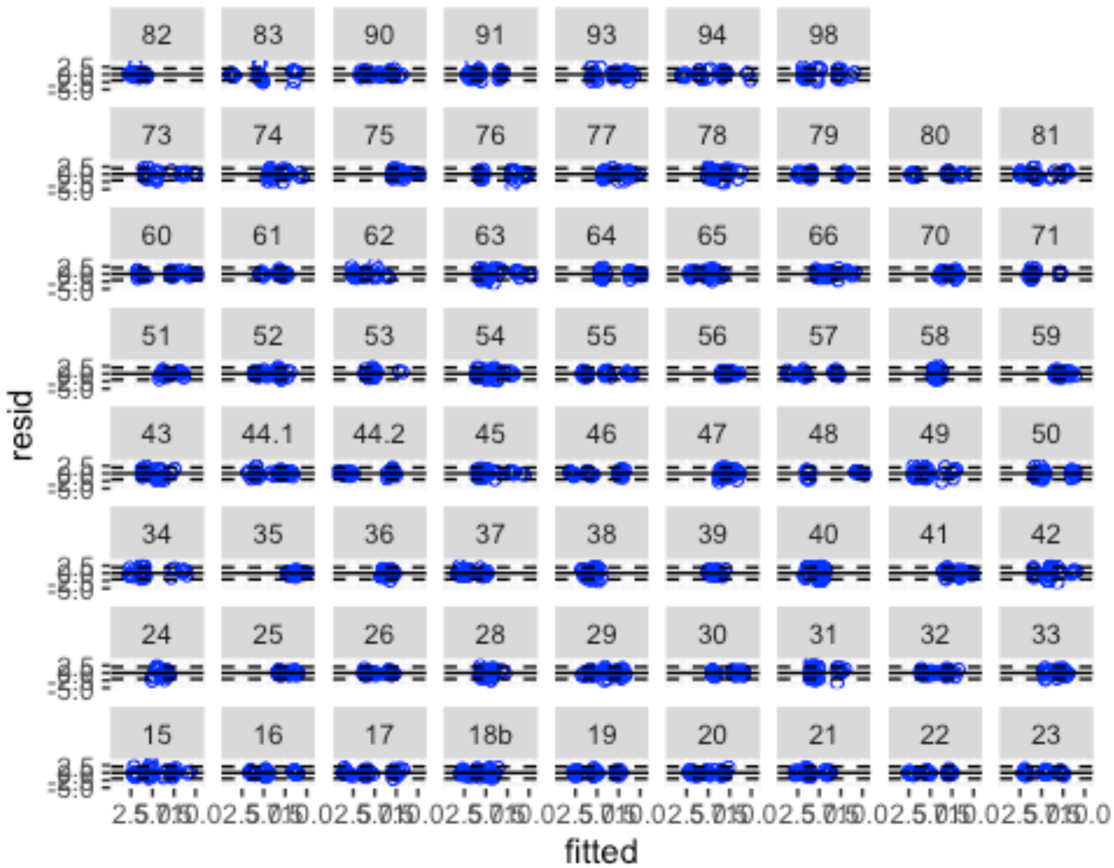
```
## = TRUE, : convergence code 1 from bobyqa: bobyqa -- maximum number of
## function evaluations exceeded

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

## Warning in optwrap(optimizer, devfun, x@theta, lower = x@lower, calc.
derivs
## = TRUE, : convergence code 1 from bobyqa: bobyqa -- maximum number of
## function evaluations exceeded

##  log-likelihood ratio test p-value = 5.658767e-90
##  adding (Instrument | Subject) to model
## evaluating addition of (Voice|Subject) to model

## boundary (singular) fit: see ?isSingular
## refitting model(s) with ML (instead of REML)

## Warning in optwrap(optimizer, devfun, x@theta, lower = x@lower, calc.
derivs
## = TRUE, : convergence code 1 from bobyqa: bobyqa -- maximum number of
## function evaluations exceeded

## Warning in optwrap(optimizer, devfun, x@theta, lower = x@lower, calc.
derivs
## = TRUE, : convergence code 1 from bobyqa: bobyqa -- maximum number of
## function evaluations exceeded

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

## Warning in optwrap(optimizer, devfun, x@theta, lower = x@lower, calc.
derivs
## = TRUE, : convergence code 1 from bobyqa: bobyqa -- maximum number of
## function evaluations exceeded


## Warning in optwrap(optimizer, devfun, x@theta, lower = x@lower, calc.
derivs
## = TRUE, : convergence code 1 from bobyqa: bobyqa -- maximum number of
## function evaluations exceeded

##  log-likelihood ratio test p-value = 0.9999867
##  not adding (Voice | Subject) to model
## ======================================================
## ===           re-backfitting fixed effects       ===
## ======================================================
```

```
## processing model terms of interaction level 1
##   all terms of interaction level 1 significant
## resetting REML to TRUE

## Warning in optwrap(optimizer, devfun, getStart(start, rho$lower, rho
$pp), :
## convergence code 1 from bobyqa: bobyqa -- maximum number of function
## evaluations exceeded

## boundary (singular) fit: see ?isSingular

## pruning random effects structure ...
##   nothing to prune
## log file is /var/folders/2z/10l5bnjj24n7r12mj3htn6_c0000gn/T//Rtmplt
8doo/fitLMER_log_Fri_Dec__6_12-56-28_2019.txt

anova(m.0, m.2)

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

## Warning in optwrap(optimizer, devfun, x@theta, lower = x@lower, calc.
derivs
## = TRUE, : convergence code 1 from bobyqa: bobyqa -- maximum number of
## function evaluations exceeded

## Data: data
## Models:
## m.0: Classical ~ Harmony + Instrument + Voice + (1 | Subject)
## m.2: Classical ~ Harmony + Instrument + Voice + (1 | Subject) + (Harm
ony |
## m.2:     Subject) + (Instrument | Subject)
##     Df     AIC   BIC  logLik deviance  Chisq Chi Df Pr(>Chisq)
## m.0 10 10220.3 10278 -5100.2  10200.3
## m.2 26  9715.1  9866 -4831.6   9663.1 537.24     16  < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

AIC(m.2)

## [1] 9735.298

BIC(m.2)

## [1] 9886.23
```

```r
summary(m.2)

## Linear mixed model fit by REML ['lmerMod']
## Formula:
## Classical ~ Harmony + Instrument + Voice + (1 | Subject) + (Harmony |

##     Subject) + (Instrument | Subject)
##    Data: data
## Control: lmerControl(optimizer = "bobyqa")
##
## REML criterion at convergence: 9683.3
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -4.7648 -0.5853  0.0300  0.5512  4.2692
##
## Random effects:
##  Groups    Name            Variance  Std.Dev. Corr
##  Subject   (Intercept)      4.640e-06 0.002154
##  Subject.1 (Intercept)      1.690e+00 1.300111
##            HarmonyI-V-IV    4.320e-02 0.207851  0.45
##            HarmonyI-V-VI    1.633e+00 1.277808 -0.41  0.34
##            HarmonyIV-I-V    4.236e-03 0.065084  0.46  0.80 -0.21
##  Subject.2 (Intercept)      1.275e+00 1.129192
##            Instrumentpiano  1.688e+00 1.299083 -0.63
##            Instrumentstring 3.380e+00 1.838382 -0.99  0.68
##  Residual                  2.369e+00 1.539098
## Number of obs: 2453, groups:  Subject, 70
##
## Fixed effects:
##                  Estimate Std. Error t value
## (Intercept)       4.34995    0.22387  19.430
## HarmonyI-V-IV    -0.04003    0.09146  -0.438
## HarmonyI-V-VI     0.77776    0.17639   4.409
## HarmonyIV-I-V     0.05792    0.08814   0.657
## Instrumentpiano   1.33728    0.17323   7.720
## Instrumentstring  3.05317    0.23294  13.107
## Voicepar3rd      -0.38437    0.07622  -5.043
## Voicepar5th      -0.35464    0.07618  -4.655
##
## Correlation of Fixed Effects:
```

```
##            (Intr) HI-V-I HI-V-V HIV-I- Instrmntp Instrmnts Vcpr3r
## HrmnyI-V-IV -0.104
## HrmnyI-V-VI -0.347  0.319
## HrmnyIV-I-V -0.167  0.498  0.233
## Instrumntpn -0.418  0.001  0.001  0.000
## Instrmntstr -0.620  0.001  0.001  0.000  0.647
## Voicepar3rd -0.170  0.000 -0.001  0.003  0.000    -0.002
## Voicepar5th -0.168 -0.004 -0.003 -0.005 -0.002     0.000     0.500
## convergence code: 1
## boundary (singular) fit: see ?isSingular
```

```
res <- r.cond(m.2)
robust.sd <- diff(quantile(res,c(.025,.975)))/(2*1.96)
res <- res/robust.sd
fit <- yhat.cond(m.2)

resdata <- data.frame(subject=data$Subject,resid=res,fitted=fit)

resparams <- data.frame(subject=unique(data$Subject),
                      int1=0,slo1=0,
                      int2=2,slo2=0,
                      int3=-2,slo3=0)

mlm_facets(resdata,"subject",x="fitted",y="resid",params=resparams,
          lty=c(1,2,2),size=c(0.5,0.5,0.5))
```



(d) Multilevel Model for Classical Rating

```
X.cont <- names(data)
X.cont <- X.cont[-grep("Classical",X.cont)]
X.cont <- X.cont[-grep("Popular",X.cont)]
X.cont <- X.cont[-grep("Subject",X.cont)]
X.disc <- c("Subject")

max.fla <- as.formula(paste("Classical ~",
                            paste(c(X.cont,"Voice:Harmony"),
                                  collapse="+")))

music.max <- lm(max.fla,data=data)

music.fix <- stepAIC(music.max, direction = "backward", trace = 0)
summary(music.fix)
```

```
##
## Call:
## lm(formula = Classical ~ Harmony + Instrument + Voice + Selfdeclare +

##     OMSI + X16.minus.17 + PachListen + ClsListen + X1990s2000s +
##     X1990s2000s.minus.1960s1970s + Composing + PianoPlay + Harmony:Vo
ice,
##     data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.1610 -1.6000  0.0409  1.5847  6.2968
##
## Coefficients:
##                              Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   3.60396    0.33115  10.883  < 2e-16 ***
## HarmonyI-V-IV                 0.16162    0.21844   0.740  0.45943
## HarmonyI-V-VI                 1.21043    0.21844   5.541 3.34e-08 ***
## HarmonyIV-I-V                -0.10405    0.21760  -0.478  0.63259
## Instrumentpiano               1.37856    0.10913  12.632  < 2e-16 ***
## Instrumentstring              3.06627    0.10906  28.115  < 2e-16 ***
## Voicepar3rd                  -0.21073    0.21733  -0.970  0.33233
## Voicepar5th                  -0.16851    0.21871  -0.770  0.44108
## Selfdeclare                  -0.59982    0.05993 -10.009  < 2e-16 ***
## OMSI                         0.24544    0.05253   4.672 3.15e-06 ***
## X16.minus.17                 -0.09710    0.01642  -5.912 3.88e-09 ***
## PachListen                    0.08518    0.04481   1.901  0.05745 .
## ClsListen                     0.27970    0.03733   7.493 9.48e-14 ***
## X1990s2000s                  -0.14080    0.03580  -3.933 8.62e-05 ***
## X1990s2000s.minus.1960s1970s  0.16302    0.03219   5.065 4.41e-07 ***
## Composing                     0.21242    0.04098   5.183 2.37e-07 ***
## PianoPlay                     0.09945    0.03113   3.195  0.00142 **
## HarmonyI-V-IV:Voicepar3rd    -0.34701    0.30833  -1.125  0.26052
## HarmonyI-V-VI:Voicepar3rd    -0.72744    0.30852  -2.358  0.01846 *
## HarmonyIV-I-V:Voicepar3rd     0.49184    0.30833   1.595  0.11080
## HarmonyI-V-IV:Voicepar5th    -0.21745    0.30949  -0.703  0.48237
## HarmonyI-V-VI:Voicepar5th    -0.49219    0.30911  -1.592  0.11146
## HarmonyIV-I-V:Voicepar5th     0.01469    0.30814   0.048  0.96198
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 2.176 on 2361 degrees of freedom
##   (69 observations deleted due to missingness)
## Multiple R-squared:  0.3181, Adjusted R-squared:  0.3117
## F-statistic: 50.06 on 22 and 2361 DF,  p-value: < 2.2e-16
```

```
music.fix$call
```

```
## lm(formula = Classical ~ Harmony + Instrument + Voice + Selfdeclare +

##     OMSI + X16.minus.17 + PachListen + ClsListen + X1990s2000s +
##     X1990s2000s.minus.1960s1970s + Composing + PianoPlay + Harmony:Vo
ice,
##     data = data)
```

```
music.mlm<-lmer(Classical ~  Voice+ Instrument + Harmony + Selfdeclare +

    OMSI + X16.minus.17 + PachListen + ClsListen + X1990s2000s +
    X1990s2000s.minus.1960s1970s + Composing + PianoPlay + Harmony:Voice
  + (1 + Harmony + Instrument | Subject),
    data = data, control = lmerControl(optimizer = "bobyqa"), REML=F)
```

```
## Warning in optwrap(optimizer, devfun, getStart(start, rho$lower, rho
$pp), :
## convergence code 1 from bobyqa: bobyqa -- maximum number of function
## evaluations exceeded
```

```
## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl =
## control$checkConv, : Model failed to converge with max|grad| = 0.0046
4679
## (tol = 0.002, component 1)
```

```
summary(music.mlm)
```

```
## Linear mixed model fit by maximum likelihood  ['lmerMod']
## Formula: Classical ~ Voice + Instrument + Harmony + Selfdeclare + OMS
I +
##     X16.minus.17 + PachListen + ClsListen + X1990s2000s + X1990s2000
s.minus.1960s1970s +
##     Composing + PianoPlay + Harmony:Voice + (1 + Harmony + Instrument
 |
##     Subject)
##    Data: data
## Control: lmerControl(optimizer = "bobyqa")
##
```

```
##      AIC      BIC   logLik deviance df.resid
##   9385.8   9645.7  -4647.9   9295.8     2339
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -4.8038 -0.5734  0.0167  0.5598  3.8209
##
## Random effects:
##  Groups   Name            Variance Std.Dev. Corr

##  Subject  (Intercept)      2.15792  1.4690

##           HarmonyI-V-IV    0.06234  0.2497    0.75

##           HarmonyI-V-VI    1.64882  1.2841   -0.20  0.28

##           HarmonyIV-I-V    0.01233  0.1110    0.20  0.53 -0.16

##           Instrumentpiano  1.67381  1.2938   -0.38 -0.69 -0.24 -0.25

##           Instrumentstring 3.38306  1.8393   -0.57 -0.54 -0.39  0.39
0.67
##  Residual                  2.28632  1.5121

## Number of obs: 2384, groups:  Subject, 68
##
## Fixed effects:
##                              Estimate Std. Error t value
## (Intercept)                    3.55161    0.86401   4.111
## Voicepar3rd                   -0.21773    0.15117  -1.440
## Voicepar5th                   -0.17562    0.15206  -1.155
## Instrumentpiano                1.37023    0.17460   7.848
## Instrumentstring               3.07986    0.23596  13.053
## HarmonyI-V-IV                  0.16252    0.15489   1.049
## HarmonyI-V-VI                  1.19195    0.21768   5.476
## HarmonyIV-I-V                 -0.10109    0.15187  -0.666
## Selfdeclare                   -0.44400    0.17410  -2.550
## OMSI                          0.21006    0.15339   1.369
## X16.minus.17                  -0.07276    0.04633  -1.570
## PachListen                     0.10496    0.13108   0.801
## ClsListen                      0.31205    0.10885   2.867
```

```
## X1990s2000s                  -0.24789    0.10478  -2.366
## X1990s2000s.minus.1960s1970s  0.21479    0.09398   2.286
## Composing                     0.15053    0.12005   1.254
## PianoPlay                     0.14714    0.09129   1.612
## Voicepar3rd:HarmonyI-V-IV    -0.33513    0.21453  -1.562
## Voicepar5th:HarmonyI-V-IV    -0.20535    0.21517  -0.954
## Voicepar3rd:HarmonyI-V-VI    -0.71722    0.21460  -3.342
## Voicepar5th:HarmonyI-V-VI    -0.48100    0.21509  -2.236
## Voicepar3rd:HarmonyIV-I-V     0.49819    0.21449   2.323
## Voicepar5th:HarmonyIV-I-V     0.01457    0.21434   0.068

##
## Correlation matrix not shown by default, as p = 23 > 12.
## Use print(x, correlation=TRUE)  or
##     vcov(x)        if you need it

## convergence code: 1
## Model failed to converge with max|grad| = 0.00464679 (tol = 0.002, co
mponent 1)
```

(e) Differences between Musicians and non-Musicians

```
is.musician <- as.numeric(data$Selfdeclare)
is.musician[which(is.musician <= 2)] <- 0
is.musician[which(is.musician > 2)] <- 1
is.musician  <- as.factor(is.musician)

musician.data <- data
musician.data$is.musician <- is.musician
musician.data <- musician.data[, -c(which(colnames(musician.data) == "S
elfdeclare"))]
colnames(musician.data)
```

```
##  [1] "Subject"                      "Harmony"
##  [3] "Instrument"                   "Voice"
##  [5] "OMSI"                         "X16.minus.17"
##  [7] "ConsInstr"                    "ConsNotes"
##  [9] "Instr.minus.Notes"            "PachListen"
## [11] "ClsListen"                    "KnowRob"
## [13] "KnowAxis"                     "X1990s2000s"
## [15] "X1990s2000s.minus.1960s1970s" "CollegeMusic"
## [17] "NoClass"                      "APTheory"
```

```
## [19] "Composing"                   "PianoPlay"
## [21] "GuitarPlay"                  "Classical"
## [23] "Popular"                     "is.musician"
```

```
# Step BIC on Fixed effect
mlm.final.fix <- step(lm(Classical ~ is.musician*(. - Popular - Subjec
t),
                        data = musician.data),
                 direction = "backward", trace = 0,
                 k = log(nrow(musician.data)))
summary(mlm.final.fix)
```

```
##
## Call:
## lm(formula = Classical ~ is.musician + Harmony + Instrument +
##     X16.minus.17 + ConsInstr + ConsNotes + Instr.minus.Notes +
##     KnowRob + KnowAxis + X1990s2000s + CollegeMusic + APTheory +
##     Composing + PianoPlay + GuitarPlay + is.musician:Harmony +
##     is.musician:X16.minus.17 + is.musician:ConsInstr + is.musician:Co
nsNotes +
##     is.musician:KnowRob + is.musician:CollegeMusic + is.musician:APTh
eory +
##     is.musician:PianoPlay + is.musician:GuitarPlay, data = musician.d
ata)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.0024 -1.4495 -0.0182  1.3998  7.1451
##
## Coefficients:
##                            Estimate Std. Error t value Pr(>|t|)
## (Intercept)                 4.15957    0.26050  15.967  < 2e-16 ***
## is.musician1                1.98018    0.39931   4.959 7.59e-07 ***
## HarmonyI-V-IV              -0.06285    0.15627  -0.402  0.68756
## HarmonyI-V-VI               0.31109    0.15628   1.991  0.04664 *
## HarmonyIV-I-V               0.06935    0.15594   0.445  0.65655
## Instrumentpiano             1.36704    0.10466  13.061  < 2e-16 ***
## Instrumentstring            3.05535    0.10457  29.219  < 2e-16 ***
## X16.minus.17                0.02394    0.02096   1.142  0.25363
## ConsInstr                  -0.42161    0.05589  -7.544 6.48e-14 ***
## ConsNotes                   0.23844    0.05523   4.317 1.64e-05 ***
## Instr.minus.Notes           0.41515    0.05361   7.743 1.43e-14 ***
```

```
## KnowRob                         0.28989    0.06400   4.529 6.21e-06 ***
## KnowAxis                        0.13670    0.02969   4.604 4.36e-06 ***
## X1990s2000s                    -0.24320    0.03420  -7.112 1.51e-12 ***
## CollegeMusic1                   1.04764    0.16950   6.181 7.50e-10 ***
## APTheory1                       1.99175    0.26222   7.596 4.39e-14 ***
## Composing                       0.21275    0.05101   4.171 3.14e-05 ***
## PianoPlay                      -0.54770    0.07678  -7.133 1.30e-12 ***
## GuitarPlay                      2.32596    0.20023  11.616  < 2e-16 ***
## is.musician1:HarmonyI-V-IV  0.09172    0.24667   0.372  0.71006
## is.musician1:HarmonyI-V-VI  1.19591    0.24652   4.851 1.31e-06 ***
## is.musician1:HarmonyIV-I-V -0.02202    0.24630  -0.089  0.92878
## is.musician1:X16.minus.17  -0.45019    0.04074 -11.051  < 2e-16 ***
## is.musician1:ConsInstr     -0.26906    0.08954  -3.005  0.00269 **
## is.musician1:ConsNotes      0.30255    0.07022   4.309 1.71e-05 ***
## is.musician1:KnowRob       -0.37936    0.07579  -5.005 6.00e-07 ***
## is.musician1:CollegeMusic1 -1.38807    0.24980  -5.557 3.06e-08 ***
## is.musician1:APTheory1     -2.20782    0.33578  -6.575 5.96e-11 ***
## is.musician1:PianoPlay      0.61829    0.08853   6.984 3.73e-12 ***
## is.musician1:GuitarPlay    -2.32704    0.19775 -11.767  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.086 on 2354 degrees of freedom
##   (69 observations deleted due to missingness)
## Multiple R-squared:  0.375,  Adjusted R-squared:  0.3673
## F-statistic:  48.7 on 29 and 2354 DF,  p-value: < 2.2e-16
```

```
# random intercept
mlm.semi <- lmer(as.formula(paste("Classical ~",
                                  paste(as.character(formula(mlm.final.fi
x))[3],
                                        "(1 | Subject)",
                                        sep = "+"))),
                 lmerControl(optimizer = "bobyqa"),  REML = F, data = mu
sician.data)

exactLRT(mlm.semi, mlm.final.fix)

## No restrictions on fixed effects. REML-based inference preferable.

## Warning in exactLRT(mlm.semi, mlm.final.fix): Null distribution has m
ass 0.9799 at zero.
```

```
##
##  simulated finite sample distribution of LRT. (p-value based on
##  10000 simulated values)
##
## data:
## LRT = 398.8, p-value < 2.2e-16
```

```r
summary(mlm.semi)
```

```
## Linear mixed model fit by maximum likelihood  ['lmerMod']
## Formula: Classical ~ is.musician + Harmony + Instrument + X16.minus.1
7 +
##     ConsInstr + ConsNotes + Instr.minus.Notes + KnowRob + KnowAxis +

##     X1990s2000s + CollegeMusic + APTheory + Composing + PianoPlay +
##     GuitarPlay + is.musician:Harmony + is.musician:X16.minus.17 +
##     is.musician:ConsInstr + is.musician:ConsNotes + is.musician:KnowR
ob +
##     is.musician:CollegeMusic + is.musician:APTheory + is.musician:Pia
noPlay +
##     is.musician:GuitarPlay + (1 | Subject)
##    Data: musician.data
## Control: lmerControl(optimizer = "bobyqa")
##
##      AIC      BIC   logLik deviance df.resid
##   9906.8  10091.6  -4921.4   9842.8     2352
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.2913 -0.6348 -0.0190  0.6249  3.6851
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  Subject  (Intercept) 0.8899   0.9434
##  Residual             3.4034   1.8448
## Number of obs: 2384, groups:  Subject, 68
##
## Fixed effects:
##                             Estimate Std. Error t value
## (Intercept)                 4.110810   0.670291   6.133
## is.musician1                1.940467   1.055252   1.839
## HarmonyI-V-IV              -0.065489   0.138203  -0.474
```

```
## HarmonyI-V-VI                0.325682   0.138255   2.356
## HarmonyIV-I-V                0.071712   0.137905   0.520
## Instrumentpiano              1.380351   0.092629  14.902
## Instrumentstring             3.061534   0.092591  33.065
## X16.minus.17                 0.003835   0.055876   0.069
## ConsInstr                   -0.433840   0.157741  -2.750
## ConsNotes                    0.242246   0.156448   1.548
## Instr.minus.Notes            0.399846   0.151506   2.639
## KnowRob                      0.284147   0.182370   1.558
## KnowAxis                     0.137682   0.084282   1.634
## X1990s2000s                 -0.227153   0.095846  -2.370
## CollegeMusic1                1.050253   0.482078   2.179
## APTheory1                    2.010390   0.744325   2.701
## Composing                    0.216837   0.142372   1.523
## PianoPlay                   -0.535140   0.217550  -2.460
## GuitarPlay                   2.310450   0.567740   4.070
## is.musician1:HarmonyI-V-IV   0.095521   0.218139   0.438
## is.musician1:HarmonyI-V-VI   1.182824   0.218031   5.425
## is.musician1:HarmonyIV-I-V  -0.022730   0.217810  -0.104
## is.musician1:X16.minus.17   -0.422023   0.112981  -3.735
## is.musician1:ConsInstr      -0.233179   0.249804  -0.933
## is.musician1:ConsNotes       0.282376   0.198436   1.423
## is.musician1:KnowRob        -0.368291   0.215942  -1.706
## is.musician1:CollegeMusic1  -1.409522   0.709845  -1.986
## is.musician1:APTheory1      -2.188848   0.947398  -2.310
## is.musician1:PianoPlay       0.598237   0.250179   2.391
## is.musician1:GuitarPlay     -2.322420   0.561548  -4.136

##
## Correlation matrix not shown by default, as p = 30 > 12.
## Use print(x, correlation=TRUE)  or
##     vcov(x)        if you need it

mlm.semi.2 <- lme4::lmer(Classical ~ is.musician + Harmony + Instrument
+
    ConsInstr+ Instr.minus.Notes  +
    X1990s2000s + CollegeMusic + APTheory + PianoPlay +
    GuitarPlay + is.musician:Harmony + is.musician:X16.minus.17   +
    is.musician:CollegeMusic + is.musician:APTheory + is.musician:PianoP
lay +
    is.musician:GuitarPlay + (1 | Subject),
                 lmerControl(optimizer = "bobyqa"),  REML = F, data = mu
```

```
sician.data)
exactRLRT(mlm.semi.2)

## Using restricted likelihood evaluated at ML estimators.

## Refit with method="REML" for exact results.

##
##  simulated finite sample distribution of RLRT.
##
##  (p-value based on 10000 simulated values)
##
## data:
## RLRT = 534.84, p-value < 2.2e-16

# Random slope
mlm.final <- ffRanefLMER.fnc(mlm.semi.2,
                             ran.effects = c("(Harmony + Instrument + Voic
e | Subject)"),
                             log.file = F)

## evaluating addition of (Harmony+Instrument+Voice|Subject) to model

## Warning in commonArgs(par, fn, control, environment()): maxfun < 10 *
## length(par)^2 is not recommended.

## Warning in optwrap(optimizer, devfun, getStart(start, rho$lower, rho
$pp), :
## convergence code 1 from bobyqa: bobyqa -- maximum number of function
## evaluations exceeded

## boundary (singular) fit: see ?isSingular

##  log-likelihood ratio test p-value = 1.236733e-90
##  adding (Harmony+Instrument+Voice|Subject) to model

summary(mlm.final)

## Linear mixed model fit by maximum likelihood  ['lmerMod']
## Formula: Classical ~ is.musician + Harmony + Instrument + ConsInstr +

##     Instr.minus.Notes + X1990s2000s + CollegeMusic + APTheory +
##     PianoPlay + GuitarPlay + (1 | Subject) + (Harmony + Instrument +
##     Voice | Subject) + is.musician:Harmony + is.musician:X16.minus.17
 +
```

```
##     is.musician:CollegeMusic + is.musician:APTheory + is.musician:Pia
noPlay +
##     is.musician:GuitarPlay
##    Data: musician.data
## Control: lmerControl(optimizer = "bobyqa")
##
##      AIC      BIC   logLik deviance df.resid
##   9734.7  10088.8  -4806.4   9612.7     2392
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -4.8447 -0.5765  0.0187  0.5786  4.3967
##
## Random effects:
##  Groups    Name             Variance Std.Dev. Corr
##  Subject   (Intercept)      0.18858  0.4343
##  Subject.1 (Intercept)      2.29301  1.5143
##            HarmonyI-V-IV    0.10950  0.3309    0.61
##            HarmonyI-V-VI    1.33434  1.1551    0.09  0.28
##            HarmonyIV-I-V    0.02011  0.1418   -0.02  0.73 -0.05
##            Instrumentpiano  1.70602  1.3061   -0.50 -0.48 -0.19 -0.16
##            Instrumentstring 3.47857  1.8651   -0.75 -0.41 -0.40  0.19
##            Voicepar3rd      0.23864  0.4885   -0.55 -0.72 -0.83 -0.22
##            Voicepar5th      0.15592  0.3949   -0.57 -0.94 -0.52 -0.60
##  Residual                   2.33105  1.5268
##
##
##
##
##
##
##
##   0.69
##   0.45  0.66
##   0.40  0.54  0.87
##
## Number of obs: 2453, groups:  Subject, 70
##
## Fixed effects:
##                              Estimate Std. Error t value
## (Intercept)                   4.346023   0.599988   7.244
```

```
## is.musician1                  1.685818   0.694993   2.426
## HarmonyI-V-IV                -0.129642   0.120821  -1.073
## HarmonyI-V-VI                 0.044161   0.188946   0.234
## HarmonyIV-I-V                 0.049095   0.114551   0.429
## Instrumentpiano               1.526482   0.168114   9.080
## Instrumentstring              3.440247   0.217507  15.817
## ConsInstr                    -0.147146   0.084476  -1.742
## Instr.minus.Notes             0.148438   0.082230   1.805
## X1990s2000s                  -0.230760   0.091350  -2.526
## CollegeMusic1                 0.967008   0.449087   2.153
## APTheory1                     1.697638   0.694235   2.445
## PianoPlay                    -0.258047   0.201194  -1.283
## GuitarPlay                    1.516994   0.516674   2.936
## is.musician1:HarmonyI-V-IV   0.006901   0.188879   0.037
## is.musician1:HarmonyI-V-VI   1.156735   0.293992   3.935
## is.musician1:HarmonyIV-I-V -0.020950   0.179803  -0.117
## is.musician0:X16.minus.17  -0.085374   0.053098  -1.608
## is.musician1:X16.minus.17  -0.299528   0.087109  -3.439
## is.musician1:CollegeMusic1 -1.543498   0.663394  -2.327
## is.musician1:APTheory1     -1.640124   0.849191  -1.931
## is.musician1:PianoPlay      0.232135   0.227522   1.020
## is.musician1:GuitarPlay    -1.370333   0.526869  -2.601

##
## Correlation matrix not shown by default, as p = 23 > 12.
## Use print(x, correlation=TRUE)  or
##     vcov(x)        if you need it

## convergence code: 1
## boundary (singular) fit: see ?isSingular
## maxfun < 10 * length(par)^2 is not recommended.
```

(f) Multilevel Model for Popular Rating

```r
lm.1.p <- lm(Popular ~ Harmony + Instrument + Voice , data = data)
lm.2.p <- step(lm(Popular ~ Instrument * Harmony * Voice, data = data),
            direction = "backward")
```

```
## Start:  AIC=3914.76
## Popular ~ Instrument * Harmony * Voice
##
##                           Df Sum of Sq    RSS    AIC
```

```
## - Instrument:Harmony:Voice 12    57.332 11808 3902.7
## <none>                                  11750 3914.8
##
## Step:  AIC=3902.7
## Popular ~ Instrument + Harmony + Voice + Instrument:Harmony +
##     Instrument:Voice + Harmony:Voice
##
##                      Df Sum of Sq   RSS    AIC
## - Instrument:Harmony  6    18.840 11826 3894.6
## - Instrument:Voice    4    16.667 11824 3898.2
## - Harmony:Voice       6    42.832 11850 3899.6
## <none>                            11808 3902.7
##
## Step:  AIC=3894.61
## Popular ~ Instrument + Harmony + Voice + Instrument:Voice + Harmony:V
oice
##
##                    Df Sum of Sq   RSS    AIC
## - Instrument:Voice  4    16.409 11843 3890.0
## - Harmony:Voice     6    42.786 11869 3891.5
## <none>                          11826 3894.6
##
## Step:  AIC=3890.02
## Popular ~ Instrument + Harmony + Voice + Harmony:Voice
##
##                 Df Sum of Sq   RSS    AIC
## - Harmony:Voice  6     42.75 11886 3886.9
## <none>                       11843 3890.0
## - Instrument     2   2655.13 14498 4382.2
##
## Step:  AIC=3886.86
## Popular ~ Instrument + Harmony + Voice
##
##              Df Sum of Sq   RSS    AIC
## - Voice       2     14.49 11900 3885.8
## <none>                    11886 3886.9
## - Harmony     3     39.40 11925 3889.0
## - Instrument  2   2655.44 14541 4377.5
##
## Step:  AIC=3885.84
## Popular ~ Instrument + Harmony
```

```
##
##              Df Sum of Sq   RSS    AIC
## <none>                     11900 3885.8
## - Harmony     3     39.22 11939 3887.9
## - Instrument  2   2655.02 14555 4375.9
```

```
anova(lm.1.p, lm.2.p)
```

```
## Analysis of Variance Table
##
## Model 1: Popular ~ Harmony + Instrument + Voice
## Model 2: Popular ~ Instrument + Harmony
##   Res.Df   RSS Df Sum of Sq      F Pr(>F)
## 1   2445 11886
## 2   2447 11900 -2   -14.495 1.4909 0.2254
```

```
#anova choose lm.2

m.0.p <- lmer(Popular ~ Harmony + Instrument + Voice + (1|Subject),
         data = data, control = lmerControl(optimizer = "bobyqa"), REML=
F)

anova(m.0.p, lm.2.p)
```

```
## Data: data
## Models:
## lm.2.p: Popular ~ Instrument + Harmony
## m.0.p: Popular ~ Harmony + Instrument + Voice + (1 | Subject)
##        Df   AIC   BIC  logLik deviance  Chisq Chi Df Pr(>Chisq)
## lm.2.p  7 10849 10890 -5417.6    10835
## m.0.p  10 10160 10218 -5069.8    10140 695.59      3  < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# anova choose m.o.p

m.1.p <- fitLMER.fnc(m.0.p, method="AIC")
```

```
## ==========================================================
## ===              backfitting fixed effects          ===
## ==========================================================
## setting REML to FALSE
## processing model terms of interaction level 1
```

```
##   iteration 1
##     p-value for term "Voice" = 0.1187 >= 0
##     not part of higher-order interaction
##     AIC simple = 10160; AIC complex = 10160; decrease = 0 < 5
##     removing term
##   iteration 2
##     p-value for term "Harmony" = 0.0071 >= 0
##     not part of higher-order interaction
##     AIC simple = 10166; AIC complex = 10160; decrease = 6 >= 5
##     skipping term
## length = 2
##   iteration 3
##     p-value for term "Instrument" = 0 >= 0
##     not part of higher-order interaction
##     AIC simple = 10845; AIC complex = 10160; decrease = 686 >= 5
##     skipping term
## length = 1
## pruning random effects structure ...
##   nothing to prune
## =======================================================
## ===           forwardfitting random effects      ===
## =======================================================
##  ===      random slopes      ===
## =======================================================
## ===           re-backfitting fixed effects       ===
## =======================================================
## setting REML to FALSE
## processing model terms of interaction level 1
##   iteration 1
##     p-value for term "Harmony" = 0.0071 >= 0
##     not part of higher-order interaction
##     AIC simple = 10166; AIC complex = 10160; decrease = 6 >= 5
##     skipping term
## length = 2
##   iteration 2
##     p-value for term "Instrument" = 0 >= 0
##     not part of higher-order interaction
##     AIC simple = 10845; AIC complex = 10160; decrease = 686 >= 5
##     skipping term
## length = 1
## resetting REML to TRUE
```

```
## pruning random effects structure ...
##   nothing to prune
## log file is /var/folders/2z/10l5bnjj24n7r12mj3htn6_c0000gn/T//Rtmpv1
uEgj/fitLMER_log_Sat_Dec__7_14-16-13_2019.txt
```

```
vars <- attr(terms(formula(m.1.p)),"term.labels")
vars <- vars[-length(vars)]

m.2.p <- fitLMER.fnc(m.1.p,
                     ran.effects=
                       list(slopes=vars, by.vars="Subject",
                            corr=rep(1,length(vars))))
```

```
## ========================================================
## ===             backfitting fixed effects        ===
## ========================================================
## processing model terms of interaction level 1
##   all terms of interaction level 1 significant
## pruning random effects structure ...
##   nothing to prune
## ========================================================
## ===           forwardfitting random effects       ===
## ========================================================
##  ===       random slopes       ===
## evaluating addition of (Harmony|Subject) to model
```

```
## boundary (singular) fit: see ?isSingular
```

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

```
##  log-likelihood ratio test p-value = 4.892754e-10
##  adding (Harmony | Subject) to model
## evaluating addition of (Instrument|Subject) to model
```

```
## Warning in optwrap(optimizer, devfun, getStart(start, rho$lower, rho
$pp), :
## convergence code 1 from bobyqa: bobyqa -- maximum number of function
## evaluations exceeded
```

```
## boundary (singular) fit: see ?isSingular
## refitting model(s) with ML (instead of REML)
```

```
## Warning in optwrap(optimizer, devfun, x@theta, lower = x@lower, calc.
derivs
## = TRUE, : convergence code 1 from bobyqa: bobyqa -- maximum number of
## function evaluations exceeded

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

## Warning in optwrap(optimizer, devfun, x@theta, lower = x@lower, calc.
derivs
## = TRUE, : convergence code 1 from bobyqa: bobyqa -- maximum number of
## function evaluations exceeded

##  log-likelihood ratio test p-value = 3.264409e-82
##  adding (Instrument | Subject) to model
## =======================================================
## ===           re-backfitting fixed effects        ===
## =======================================================
## processing model terms of interaction level 1
##   all terms of interaction level 1 significant
## resetting REML to TRUE

## Warning in optwrap(optimizer, devfun, getStart(start, rho$lower, rho
$pp), :
## convergence code 1 from bobyqa: bobyqa -- maximum number of function
## evaluations exceeded

## boundary (singular) fit: see ?isSingular

## pruning random effects structure ...
##   nothing to prune
## log file is /var/folders/2z/10l5bnjj24n7r12mj3htn6_c0000gn/T//Rtmpv1
uEgj/fitLMER_log_Sat_Dec__7_14-16-13_2019.txt

anova(m.0.p, m.2.p)

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

## Warning in optwrap(optimizer, devfun, x@theta, lower = x@lower, calc.
derivs
## = TRUE, : convergence code 1 from bobyqa: bobyqa -- maximum number of
## function evaluations exceeded

## Data: data
## Models:
## m.0.p: Popular ~ Harmony + Instrument + Voice + (1 | Subject)
```

```
## m.2.p: Popular ~ Harmony + Instrument + (1 | Subject) + (Harmony | Su
bject) +
## m.2.p:     (Instrument | Subject)
##       Df     AIC     BIC  logLik deviance  Chisq Chi Df Pr(>Chisq)
## m.0.p 10 10159.6 10217.6 -5069.8  10139.6
## m.2.p 24  9732.2  9871.5 -4842.1   9684.2 455.34     14  < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
#ANOVA choose m.2.p

music.max.p <- lm(Popular~. - Subject - Classical, data=data)

music.fix.p <- stepAIC(music.max.p, direction = "backward", trace = 0)
music.fix.p$call
```

```
## lm(formula = Popular ~ Harmony + Instrument + Selfdeclare + X16.minu
s.17 +
##     ConsInstr + ConsNotes + Instr.minus.Notes + PachListen +
##     ClsListen + KnowRob + X1990s2000s + X1990s2000s.minus.1960s1970s
+
##     APTheory + Composing + GuitarPlay, data = data)
```

```
music.mlm.p<-lmer(Popular ~ Harmony  + Instrument + Voice + Selfdeclare
+
    X16.minus.17 + ConsInstr + ConsNotes + Instr.minus.Notes +
    PachListen + ClsListen + KnowRob + X1990s2000s +
    X1990s2000s.minus.1960s1970s + APTheory + Composing +
    GuitarPlay + (1 + Harmony + Instrument | Subject),
    data = data, control = lmerControl(optimizer = "bobyqa"), REML=F)
```

```
## Warning in optwrap(optimizer, devfun, getStart(start, rho$lower, rho
$pp), :
## convergence code 1 from bobyqa: bobyqa -- maximum number of function
## evaluations exceeded
```

```
summary(music.mlm.p)
```

```
## Linear mixed model fit by maximum likelihood  ['lmerMod']
## Formula:
## Popular ~ Harmony + Instrument + Voice + Selfdeclare + X16.minus.17 +

##     ConsInstr + ConsNotes + Instr.minus.Notes + PachListen +
```

```
##     ClsListen + KnowRob + X1990s2000s + X1990s2000s.minus.1960s1970s
+
##     APTheory + Composing + GuitarPlay + (1 + Harmony + Instrument |
##     Subject)
##    Data: data
## Control: lmerControl(optimizer = "bobyqa")
##
##      AIC      BIC   logLik deviance df.resid
##   9438.5   9686.9  -4676.2   9352.5     2341
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -4.0354 -0.5939  0.0238  0.5902  3.3241
##
## Random effects:
##  Groups   Name            Variance Std.Dev. Corr

##  Subject  (Intercept)      1.34190  1.1584

##           HarmonyI-V-IV    0.09992  0.3161    0.69

##           HarmonyI-V-VI    0.84455  0.9190   -0.03 -0.28

##           HarmonyIV-I-V    0.21962  0.4686   -0.26 -0.49 -0.41

##           Instrumentpiano  1.39311  1.1803   -0.19 -0.31 -0.21 -0.17

##           Instrumentstring 3.23361  1.7982   -0.49 -0.45 -0.24 -0.04
0.72
##  Residual                  2.35839  1.5357

## Number of obs: 2384, groups:  Subject, 68
##
## Fixed effects:
##                              Estimate Std. Error t value
## (Intercept)                   6.33174    0.74667   8.480
## HarmonyI-V-IV                -0.05061    0.09699  -0.522
## HarmonyI-V-VI                -0.31914    0.14275  -2.236
## HarmonyIV-I-V                -0.24581    0.10556  -2.329
## Instrumentpiano              -0.98551    0.16286  -6.051
## Instrumentstring             -2.61554    0.23166 -11.291
```

```
## Voicepar3rd                    0.14547    0.07715   1.886
## Voicepar5th                    0.17367    0.07711   2.252
## Selfdeclare                    0.13048    0.15908   0.820
## X16.minus.17                   0.11331    0.04572   2.478
## ConsInstr                      0.24210    0.13818   1.752
## ConsNotes                     -0.25581    0.14008  -1.826
## Instr.minus.Notes             -0.19465    0.13706  -1.420
## PachListen                    -0.20947    0.12259  -1.709
## ClsListen                     -0.05101    0.09993  -0.510
## KnowRob                        0.19057    0.08566   2.225
## X1990s2000s                    0.23684    0.10363   2.285
## X1990s2000s.minus.1960s1970s -0.09490    0.09689  -0.979
## APTheory1                      0.12193    0.33677   0.362
## Composing                      0.25345    0.12956   1.956
## GuitarPlay                    -0.30139    0.13621  -2.213

##
## Correlation matrix not shown by default, as p = 21 > 12.
## Use print(x, correlation=TRUE)  or
##     vcov(x)        if you need it

## convergence code: 1
```

```
Random effects:
 Groups   Name              Variance Std.Dev. Corr
 Subject  Instrumentguitar 1.34189  1.1584
          Instrumentpiano  2.20803  1.4859    0.63
          Instrumentstring 2.52152  1.5879    0.17  0.67
          HarmonyI-V-IV    0.09992  0.3161    0.69  0.29 -0.01
          HarmonyI-V-VI    0.84455  0.9190   -0.03 -0.20 -0.30 -0.28
          HarmonyIV-I-V    0.21962  0.4686   -0.26 -0.33 -0.23 -0.49 -0.41
 Residual                  2.35839  1.5357
Number of obs: 2384, groups:  Subject, 68

Fixed effects:
                            Estimate Std. Error t value
Instrumentguitar             6.33172    0.74667   8.480
Instrumentpiano              5.34620    0.75540   7.077
Instrumentstring             3.71617    0.75921   4.895
```

```
Random effects:
 Groups   Name             Variance Std.Dev. Corr
 Subject  HarmonyI-IV-V    1.342    1.158
          HarmonyI-V-IV    1.944    1.394     0.99
          HarmonyI-V-VI    2.116    1.455     0.78  0.73
          HarmonyIV-I-V    1.283    1.133     0.92  0.87  0.60
          Instrumentpiano  1.393    1.180    -0.19 -0.23 -0.29 -0.27
          Instrumentstring 3.234    1.798    -0.49 -0.51 -0.54 -0.52  0.72
 Residual                  2.358    1.536
Number of obs: 2384, groups:  Subject, 68

Fixed effects:
                            Estimate Std. Error t value
HarmonyI-IV-V                6.33172    0.74667   8.480
HarmonyI-V-IV                6.28110    0.75267   8.345
HarmonyI-V-VI                6.01258    0.75449   7.969
HarmonyIV-I-V                6.08591    0.74604   8.158
```

```
Fixed effects:
                    Estimate Std. Error t value
Voicecontrary        6.33174    0.74667   8.480
Voicepar3rd          6.47721    0.74667   8.675
Voicepar5th          6.50540    0.74682   8.711
```