# Code Index

*Mason Paccione*

*12/6/2019*

## Contents

# EDA and Filtering

## Initialization

```
ratings_df <- read_csv("ratings.csv") %>%
  dplyr::select(-X1, -first12) %>%
  mutate(Subject = as.factor(Subject),
         Harmony = as.factor(Harmony),
         Instrument = as.factor(Instrument),
         Voice = as.factor(Voice))

ratings_df %>% group_by(Subject) %>%
  summarise(length(Subject))
```

The following variables are made into factors, as there is not implicit ordering of their values. Some of these are binary, so this is not strictly necessary, but is still done to distinguish these variables from the truly numeric variables:

- Subject
- Harmony
- Instrument
- Voice
- CollegeMusic
- APTheory

## Missingness

```
library(skimr)
summary(ratings_df)
```

There are a large number of missing values.

Some of these are appropriate and may have meaning, while others are not. We examine all subject-constant variables, to ensure they are constant across trials within a subject:

```
check_same <- ratings_df %>%
  group_by(Subject) %>%
  summarise(same_APTheory = length(unique(APTheory)) == 1,
            same_CollegeMusic = length(unique(CollegeMusic)) == 1,
            same_ClsListen = length(unique(ClsListen)) == 1,
            same_Composing = length(unique(Composing)) == 1,
            same_ConsNotes = length(unique(ConsNotes)) == 1,
            same_KnowAxis = length(unique(KnowAxis)) == 1,
            same_KnowRob = length(unique(KnowRob)) == 1,
            same_NoClass = length(unique(NoClass)) == 1,
            same_PachListen = length(unique(PachListen)) == 1,
            same_X1990s2000s = length(unique(X1990s2000s)) == 1,
            same_X1990s2000s.minus.1960s1970s = length(unique(X1990s2000s.minus.1960s1970s)) == 1,
            same_Selfdeclare = length(unique(Selfdeclare)) == 1,
            same_PianoPlay = length(unique(PianoPlay)) == 1,
            same_GuitarPlay = length(unique(GuitarPlay)) == 1,
            same_ConsInstr = length(unique(ConsInstr)) == 1,
            same_OMSI = length(unique(OMSI)) == 1,
            same_X2ndInstr = length(unique(X2ndInstr)) == 1,
            same_X1stInstr = length(unique(X1stInstr)) == 1,
            Valid_Diff = all((ConsInstr - ConsNotes) == Instr.minus.Notes))

apply(check_same[,-1], 2, all)
```

All of the subjects have consistent values of those covariates that should not differ by song rated. However, the values for `Instr.minus.Notes` do not appear to be the differences between `ConsInstr` and `ConsNotes` as they should be. Upon inspection, this was discovered to primarily be due to missing values in `ConsNotes`. These missing values can then be defined as `ConsInstr` minus `Instr.minus.Notes`, if both of these are defined. Both of these being defined implies that earlier researchers were comfortable interpreting missing values in `ConsNotes` to be zero, or failed to pass these zeroes along to us. Thus we defer to their decision here. We also filter those rows of the data missing either `Popular` or `Classical`, as there are very few, and these are the primary variables of analysis. Interpolating them could very well effect our primary results.

```
ratings_df <- ratings_df %>%
  mutate(ConsNotes = ifelse(is.na(ConsNotes),
                            ConsInstr - Instr.minus.Notes,
                            ConsNotes)) %>%
  filter(!is.na(Classical) & !is.na(Popular))
```

## Repairs

Many variables have missing values, and potentially incorrect values. The following can be checked using other variables or be used to check other varaibles.

- `CollegeMusic`
- `NoClass`

If `NoClass` is greater than zero, then `CollegeMusic` should be true. Likewise, if `CollegeMusic` is true, then `NoClass` should be 1 or greater.

- `PianoPlay`
- `GuitarPlay`
- `X2ndInstr`
- `X1stInstr`

If one plays guitar or piano to some competence level, then that individual's `X1stInstr` measure should be, at least that competence level, if we consider these to be on the same scale. This is a reasonable assumption, as both are competency measure defined on the same range. If both guitar and piano are played to some level, then `X2ndInstr` should be at least as great as the lesser of these measures, following the same assumption of common scaling.

```
ratings_df <- ratings_df %>%
  mutate(X1stInstr = ifelse(X1stInstr < pmax(PianoPlay, GuitarPlay)  |
                              (is.na(X1stInstr) &
                                 pmax(PianoPlay, GuitarPlay) == 5),
                            pmax(PianoPlay, GuitarPlay),
                            X1stInstr),
         X2ndInstr = ifelse(PianoPlay > 0 & GuitarPlay > 0 &
                              (is.na(X2ndInstr) |
                                 X2ndInstr < pmin(PianoPlay, GuitarPlay)),
                            pmin(PianoPlay, GuitarPlay), X2ndInstr),
         X2ndInstr = ifelse(is.na(X2ndInstr) & X1stInstr == 0, 0, X2ndInstr),
         NoClass = ifelse(is.na(NoClass) & CollegeMusic == 0, 0, NoClass),
         CollegeMusic = ifelse(!is.na(NoClass) & (NoClass > 0), 1,
                                       CollegeMusic),
         CollegeMusic = ifelse(!is.na(NoClass) & (NoClass == 0), 0,
                                       CollegeMusic)) %>%
  dplyr::select(-X1stInstr, -X2ndInstr)
```

After these repairs, we still find many missing values in `X1stInstr`, `X2ndInstr`, and so choose to drop these covariates.

We then choose to develop a new variable from `APTheory` and `CollegeMusic`, which we call `Theory`, and which measures whether a subject took either of these groups. We then drop these two variables, in favor of the composite and largely eqivalent in meaning `Theory`, which also has fewer missing values.

```
ratings_df <- ratings_df %>%
  mutate(Theory = ifelse(is.na(CollegeMusic),
                         ifelse(is.na(APTheory), NA, APTheory),
                         ifelse(is.na(APTheory), CollegeMusic,
                                as.numeric(CollegeMusic == 1 |
                                             APTheory == 1)))) %>%
  dplyr::select(-APTheory, -CollegeMusic)
```

We then choose only those variables with a concise meaning, dropping the highly composite variables of `OMSI`, a score on a test of musical comprehension, `X16.minus.17`, an auxilliary measure of a listeners ability to distinguish classical and popular music. Conclusions in our analysis which include these variables would be tautological or overly vague, and thus uninformative. For example to say "a listeners music comprehension effects their Classical Rating in this way" is extremely vague, while "A listeners ability to distinguish classical and poular music effects their ability to distinguish classical music in this way" is tautological.

We also drop `X1990s.minus.1960s1970s` as previous researchers have chosen not to include the feature `X1960s.1970s`. The exact meaning of this composite is somehat difficult to determine, as the scale of the variables it is derived from is arbitarary. Also, having two variables measuring how often pop is listened two is somewhat incongruent, given that we only have one variable measuring how often classical music is listened to.

Finally, we choose to drop NoClass as it has a large number of missing values despite our efforts, and the vast majority of existing values are `0` or `1` and thus this variable likely does not add much more than our derived `Theory` feature.

```
plot1<- ggplot(ratings_df, aes(x = as.factor(NoClass))) +
  geom_bar(aes(y=..count../sum(..count..))) +
  scale_y_continuous(labels=percent_format()) +
  labs(x = "NoClass", y = "Percentage")

ggsave("NoClass.png", plot1, device = png())

ratings_df <- ratings_df %>%
  dplyr::select(-OMSI, -X16.minus.17, -X1990s2000s.minus.1960s1970s, -NoClass)
```

Thus we are left with a handful of missing values in `PachListen`, `ClsListen`, `KnowRob`, `KnowAxis`, `X1990s2000s`, `Composing`, and `Theory`. We then choose to develop two final datasets, given the primary research questions. In the first, primary dataset, we drop `PachListen`, `KnowRob`, and `KnowAxis`, and filter to only complete cases on the remaining variables. In the secondary dataset, we filter to complete cases including these.

```
summary(ratings_df)
ratings_1 <- ratings_df %>%
  dplyr::select(-PachListen, -KnowRob, -KnowAxis) %>%
  filter(complete.cases(.))

ratings_2 <- ratings_df %>%
  filter(complete.cases(.))
```

# Modelling Classical Scores

## Basic Linear

```
##  Basic lm() model selection
classical_full <- lm(Classical ~ Instrument*Harmony*Voice, data = ratings_1)

X_mat <- model.matrix(Classical ~ Instrument*Harmony*Voice,
                      data = ratings_1)
Y <- filter(ratings_1, !is.na(Classical))$Classical
nn <- length(Y)

#  Lasso selection
find_min_lambda <- glmnet::cv.glmnet(X_mat, Y, alpha = 1)
classical_lasso <- glmnet::glmnet(X_mat, Y, lambda = find_min_lambda$lambda.1se)
broom::tidy(classical_lasso$beta) %>%
  dplyr::select(-column)

#  Stepwise BIC selection
library(MASS)
classical_stepAIC <- stepAIC(classical_full, direction = "both", k = log(nn),
                         trace = F)
summary(classical_stepAIC)

#  All subsets selection
library(leaps)
classical_subsets <- regsubsets(Classical ~ Instrument*Harmony*Voice,
                           data = ratings_1,
                           method = "seqrep",
                           nvmax = 36)
coef(classical_subsets, 5)

#  Final lm() model
class_lm_1 <- lm(Classical ~ Instrument*Harmony, data = ratings_1)
class_lm_2 <- lm(Classical ~ Instrument+Harmony+Voice, data = ratings_1)
class_lm_3 <- lm(Classical ~ Instrument + Harmony, data = ratings_1)
BIC(class_lm_1, class_lm_2, class_lm_3)
AIC(class_lm_1, class_lm_2, class_lm_3)
summary(class_lm_2)
```

The best model, including all of the first order `Instrument`, `Harmony`, and `Voice` without interactions, was selected after reviewing the automated selections from all subsets, stepwise (both directions) BIC, and lasso with a cross validation selected $\lambda_{1se}$. The models selected by these were then compared by their BIC and AIC. The final model had both the best AIC and BIC, with the form:

$$\begin{aligned}\text{Classical}_i = \beta_0 &+ \beta_1\mathbb{I}(\text{Instrument} = \text{piano})_i + \beta_2\mathbb{I}(\text{Instrument} = \text{string})_i \\ &+ \beta_3\mathbb{I}(\text{Harmony} = \text{I-V-IV})_i + \beta_4\mathbb{I}(\text{Harmony} = \text{I-V-VI})_i \\ &+ \beta_5\mathbb{I}(\text{Harmony} = \text{IV-I-V})_i + \beta_6\mathbb{I}(\text{Voice} = \text{Parallel 3rds})_i \\ &+ \beta_7\mathbb{I}(\text{Voice} = \text{Parallel 5ths})_i\epsilon_i\end{aligned}$$

Where our estimates of these coefficients are those reported above.

Those pieces featuring an electric guitar are less likely to be interpreted as classical by subjects, scoring only 4.29 points in expectation on the classical scale when paired with a I-VI-V harmonic progression and contrary motion, all else equal. Pieces featuring the piano were in expectation scored 1.46 points higher than guitars on the classical scale by the subjects, all else equal. Pieces featuring string instruments scored 3.25 points higher than the guitar baseline, in expectation with all else equal.

Those pieces featuring an I-V-IV harmonic progression scored in expectation .005 points lower than those with a I-VI-V harmonic, all else equal, though this effect is not statistically significant. Those pieces featuring an IV-I-V harmonic progression were likewise in expectation scored .08 points higher compared to the same baseline, though this effect is not statistically significant wither. Finally, the harmonic progression I-V-VI score in expectation 0.76 more highly. This final effect was significant, and was the progression for Pachelbel's Canon in D, a canonical classical piece which many people have heard.

Finally, those pieces featuring a parallel 3rds voice part scored 0.39 points lower than those featuring a contrary motion voice segment, all else equal. Those pieces featuring a parallel 5ths voice part likewise scored .35 points lower.

## Multilevel Modelling

### Random Intercepts

$$y_i = \alpha_{0j[i]} + \sum_{k=1}^{K} \alpha_k x_{ki} + \epsilon_i, \epsilon_i \overset{iid}{\sim} N(0, \sigma^2)$$

$$\alpha_{0j} = \beta_0 + \eta_j, \eta_j \overset{iid}{\sim} N(0, \tau^2)$$

Where $j$ varies across subjects.

The model specified above is a random intercept model with $K$ individual level predictors, as the homework did not specify that the model be intercept only or the number of individual level predictors to include.

```
# Fitting random intercept only model
library(lme4)
class_int <- lmer(Classical ~ (1 | Subject),
                  data = ratings_1, REML = F)
display(class_int)
```

The value for $\beta_0$ here is 5.84, with random effects of standard deviation $\tau = 1.28$.

```
# Basic intercept only model for comparison
flat_model <- lm(Classical ~ 1, data = ratings_1)

# Compare single intercept to random intercepts
anova(class_int, flat_model)
```

We find that the random effect is needed. The AIC and BIC of the model including random intercepts are lower than those of the model including only a single intercept. An anova chi-squared test find that the random model explains significantly more than the solely-fixed model.

### Fixed Effects

```
#  Manual stepwise forward selection of fixed effects based on BIC
class_int_i <- update(class_int, . ~ . + Instrument)
class_int_v <- update(class_int, . ~ . + Voice)
class_int_h <- update(class_int, . ~ . + Harmony)
anova(class_int_i, class_int_v, class_int_h, class_int)

class_int_ih <- update(class_int_i, . ~ . + Harmony)
class_int_iv <- update(class_int_i, . ~ . + Voice)
anova(class_int_ih, class_int_iv, class_int_i)

class_int_ihv <- update(class_int_ih, . ~ . + Voice)
class_int_i_h <- update(class_int_ih, . ~ . + Instrument:Harmony)
anova(class_int_ihv,  class_int_i_h, class_int_ih)

class_int_i_hv <- update(class_int_ihv, . ~ . + Instrument:Harmony)
class_int_i_vh <- update(class_int_ihv, . ~ . + Instrument:Voice)
class_int_ih_v <- update(class_int_ihv, . ~ . + Harmony:Voice)
anova(class_int_i_hv, class_int_i_vh, class_int_ih_v, class_int_ihv)

display(class_int_ihv) # Selected based on BIC
```

```
library(LMERConvenienceFunctions)
big_lmer <- lmer(Classical ~ Instrument * Voice * Harmony + (1 | Subject),
                 data = ratings_1,
```

```r
                 REML = F)
best_int <- fitLMER.fnc(big_lmer, method = "BIC")
```

```r
best_int <- lmer(Classical ~ Instrument + Harmony + Voice + (1 | Subject),
                 data = ratings_1, REML = FALSE)
display(best_int)
rand_ints <- ranef(best_int)$Subject
rand_ints %>%
  rownames_to_column("Subject") %>%
  arrange(`(Intercept)`)
fixef(best_int) - class_lm_2$coefficients
```

We have performed manual forward selection of terms based on BIC, and automatic forward and backward selection based also on BIC. These disagree due to the automatic fitter BIC decrease threshold for including terms. We simply choose the final model with lower BIC. We find a multilevel model including `Instrument`, `Harmony`, and `Voice` in addition to a random intercept explains the data best, based on these selection methods and BIC.

We thus find that, with the addition of the random intercepts, the effects of the fixed effects are largely the same as our base linear model. This suggests that accounting for subjects implicit biases, at least in terms of intercept, does not significantly change the fixed effects of either `Instrument` or `Harmony`.

We find that subject 37 is much less generous with their classical rating than subject 35, for example, who rates things as significantly more classical in expectation than the average subject.

## Random Effects

### (i)

```
class_slo_i <- update(best_int, . ~ . + (0 + Instrument | Subject))
class_slo_v <- update(best_int, . ~ . + (0 + Voice | Subject))
class_slo_h <- update(best_int, . ~ . + (0 + Harmony | Subject))
anova(class_slo_i, class_slo_v, class_slo_h, best_int)

class_slo_ih <- update(best_int, . ~ . + (0 + Instrument + Harmony | Subject))
class_slo_iv <- update(best_int, . ~ . + (0 + Instrument + Voice | Subject))
anova(class_slo_ih, class_slo_iv, class_slo_i)

class_slo_ihv <- update(best_int, . ~ . +
                          (0 + Instrument + Harmony + Voice | Subject))
class_slo_i_h <- lmer(Classical ~ Instrument + Harmony + Voice +
                        (1 + Instrument * Harmony | Subject),
                      data = ratings_1, REML = FALSE)
anova(class_slo_ihv, class_slo_i_h, class_slo_ih)
```

We do stepwise forward selection based on BIC. We assume independence from the intercept term. The final model is as follows:

```
best_slo <- lmer(Classical ~ Instrument + Harmony + Voice +
                   (1 + Instrument + Harmony | Subject),
                 data = ratings_1, REML = FALSE)
display(best_slo)
anova(best_slo, best_int, class_lm_2)
fixef(best_slo) - class_lm_2$coefficients
fixef(best_slo) - fixef(best_int)
```

We fit the fixed effects of `Instrument` and `Harmony` with a random intercept, and random effects from `Instrument`, `Harmony`, and `Voice` again. These additional random effect terms do not change the fixed coefficient estimates very much.

### (ii)

We fit the fixed effects of `Instrument` and `Harmony` with a random intercept, and random effects from `Instrument` and `Harmony` again. These additional random effect terms do not change the fixed coefficient estimates very much.

**(iii)**

$$
\begin{aligned}
\text{Classical}_i = &\alpha_{0j[i]} + \alpha_{1j[i]}(\text{Instrument} = \text{piano})_i + \alpha_{2j[i]}\mathbb{I}(\text{Instrument} = \text{string})_i + \\
&+ \alpha_{3j[i]}\mathbb{I}(\text{Harmony} = \text{I-V-IV}) + \alpha_{4j[i]}\mathbb{I}(\text{Harmony} = \text{I-V-VI})_i + \alpha_{5j[i]}\mathbb{I}(\text{Harmony} = \text{IV-I-V})_i \\
&+ \alpha_6\mathbb{I}(\text{Voice} = \text{Parallel 3rds})_i + \alpha_7\mathbb{I}(\text{Voice} = \text{Parallel 5ths})_i \\
&+ \epsilon_i, \epsilon_i \overset{iid}{\sim} N(0, \sigma^2) \\
\alpha_{0j} &= \beta_{00} + \eta_{0j}, \eta_{0j} \overset{iid}{\sim} N(0, \tau_0^2) \\
\alpha_{1j} &= \beta_{10} + \eta_{1j}, \eta_{1j} \overset{iid}{\sim} N(0, \tau_1^2) \\
\alpha_{2j} &= \beta_{20} + \eta_{2j}, \eta_{2j} \overset{iid}{\sim} N(0, \tau_2^2) \\
\alpha_{3j} &= \beta_{30} + \eta_{3j}, \eta_{3j} \overset{iid}{\sim} N(0, \tau_3^2) \\
\alpha_{4j} &= \beta_{40} + \eta_{4j}, \eta_{4j} \overset{iid}{\sim} N(0, \tau_4^2) \\
\alpha_{5j} &= \beta_{50} + \eta_{5j}, \eta_{5j} \overset{iid}{\sim} N(0, \tau_5^2)
\end{aligned}
$$

Where the $\beta$ and $\eta$ are, in order:

```
fixef(best_slo)
ranef(best_slo)
```

## Individual-inherent effects

```
best_model <- lmer(Classical ~ Instrument + Harmony  + Voice +
                     (1 + Instrument + Harmony | Subject),
                   data = ratings_1, REML = FALSE)

# full_model <- lmer(Classical ~ Instrument + Harmony + Voice +
#                      Selfdeclare + OMSI +  X16.minus.17 + ConsInstr + ConsNotes +
#                      Instr.minus.Notes + PachListen + ClsListen + KnowRob +
#                      KnowAxis + CollegeMusic + NoClass + APTheory + Composing +
#                      PianoPlay + GuitarPlay + (1 + Instrument + Harmony | Subject),
#                    data = ratings_df,
#                    REML = FALSE)
#
# #  The below model selection crashes due to the random term
# new_model <- fitLMER.fnc(full_model, method = "BIC")

model_sd <- update(best_model, . ~ . + Selfdeclare)
model_ci <- update(best_model, . ~ . + ConsInstr)
model_cn <- update(best_model, . ~ . + ConsNotes)
model_in <- update(best_model, . ~ . + Instr.minus.Notes)
model_cl <- update(best_model, . ~ . + ClsListen)
model_th <- update(best_model, . ~ . + Theory)
model_co <- update(best_model, . ~ . + Composing)
model_pp <- update(best_model, . ~ . + PianoPlay)
model_gp <- update(best_model, . ~ . + GuitarPlay)


#  First step selection finds nothing with lower BIC, switching to AIC
anova(model_sd, model_ci, model_cn, model_in,
      model_cl, model_th, model_co, model_pp, model_gp, best_model)

best_model_1 <- update(best_model, . ~ . + ClsListen)
model_cs_1 <-  update(best_model_1, . ~ . + ConsInstr)
model_in_1 <-  update(best_model_1, . ~ . + Instr.minus.Notes)
model_co_1 <-  update(best_model_1, . ~ . + Composing)
model_pp_1 <-  update(best_model_1, . ~ . + PianoPlay)

anova(model_cs_1, model_in_1, model_co_1, model_pp_1, best_model_1)

best_model_2 <- update(best_model_1, . ~ . + X16.minus.17)
anova(best_model, best_model_2)
display(best_model_2)
```

I have selected to add *NO* additional fixed effects, based on BIC comparison. If I were to decide based on AIC, the final model would then be:

$$
\begin{aligned}
\text{Classical}_i &= \alpha_{0j[i]} + \alpha_{1j[i]}\mathbb{I}(\text{Instrument} = \text{piano})_i + \alpha_{2j[i]}\mathbb{I}(\text{Instrument} = \text{string})_i \\
&+ \alpha_{3j[i]}\mathbb{I}(\text{Harmony} = \text{I-V-IV}) + \alpha_{4j[i]}\mathbb{I}(\text{Harmony} = \text{I-V-VI})_i + \alpha_{5j[i]}\mathbb{I}(\text{Harmony} = \text{IV-I-V})_i \\
&+ \alpha_6\mathbb{I}(\text{Voice} = \text{Parallel 3rds})_i + \alpha_7\mathbb{I}(\text{Voice} = \text{Parallel 5ths})_i \\
&+ \alpha_8\mathbb{I}(\text{ClsListen}) \\
&+ \epsilon_i, \epsilon_i \overset{iid}{\sim} N(0, \sigma^2) \\
\alpha_{0j} &= \beta_{00} + \eta_{0j}, \eta_{0j} \overset{iid}{\sim} N(0, \tau_0^2) \\
\alpha_{1j} &= \beta_{10} + \eta_{1j}, \eta_{1j} \overset{iid}{\sim} N(0, \tau_1^2) \\
\alpha_{2j} &= \beta_{20} + \eta_{2j}, \eta_{2j} \overset{iid}{\sim} N(0, \tau_2^2) \\
\alpha_{3j} &= \beta_{30} + \eta_{3j}, \eta_{3j} \overset{iid}{\sim} N(0, \tau_3^2) \\
\alpha_{4j} &= \beta_{40} + \eta_{4j}, \eta_{4j} \overset{iid}{\sim} N(0, \tau_3^2) \\
\alpha_{5j} &= \beta_{50} + \eta_{5j}, \eta_{5j} \overset{iid}{\sim} N(0, \tau_3^2)
\end{aligned}
$$

As I have selected to add no additional fixed effects, due to the differences in AIC and BIC, and selecting to select models based on BIC, there is no need to reselect random effects. The model selection is the same as in $2(c)i$.

If I were to select based on AIC, then:

```
base_fixed <- lmer(Classical ~ Instrument + Harmony + Voice +
                     ClsListen + (1 | Subject),
                   data = ratings_1, REML = FALSE)

class_slo_i <- update(base_fixed, . ~ . + (0 + Instrument | Subject))
class_slo_v <- update(base_fixed, . ~ . + (0 + Voice | Subject))
class_slo_h <- update(base_fixed, . ~ . + (0 + Harmony | Subject))
anova(class_slo_i, class_slo_v, class_slo_h, base_fixed)

class_slo_ih <- update(base_fixed, . ~ . + (0 + Instrument + Harmony | Subject))
class_slo_iv <- update(base_fixed, . ~ . + (0 + Instrument + Voice | Subject))
anova(class_slo_ih, class_slo_iv, class_slo_i)

class_slo_ihv <- update(base_fixed, . ~ . +
                          (0 + Instrument + Harmony + Voice | Subject))
class_slo_i_h <- lmer(Classical ~ Instrument + Harmony + Voice +
                        ClsListen +
                        (1 + Instrument * Harmony | Subject),
                      data = ratings_1, REML = FALSE)
anova(class_slo_ihv, class_slo_i_h, class_slo_ih)
```

A stepwise variable selection based on BIC does not suggest different random effects than our earlier model excluding all non-design covariates.

Those pieces featuring an electric guitar are less likely to be interpreted as classical by subjects, scoring only 3.26 points in expectation on the classical scale when paired with a I-VI-V harmonic progression, not being a listener of classical music, and . Pieces featuring the piano were in expectation scored 1.54 points higher than guitars on the classical scale by the subjects. Pieces featuring string instruments scored 3.46 points higher than the guitar baseline, in expectation with all else equal.

Those pieces featuring an I-VI-V and guitar scored only 3.74 points in expectation on the classical scale, all else equal. This baseline is modified bases on instrument used. Those pieces featuring an I-V-IV harmonic progression were in expectation score .0007 points higher than those with a I-VI-V harmonic, all else equal, though this effect is not statistically significant. Those pieces featuring an IV-I-V harmonic progression were in expectation score .042 points higher than those with a I-VI-V harmonic, all else equal, though this effect is not statistically significant. Finally, and most significantly for harmonics, the harmonic progression I-V-VI score in expectation 0.87 more highly than those with a I-VI-V harmonic, all else equal. This effect was significant, and was progression for Pachelbel's Canon in D, which many people have heard.

Listeners who described themselves as classical music listeners scored pieces 0.31 higher on the classical music scale in expectation. Listeners with a 1 point difference in auxilliary discernment scores had a .11 difference in their classical scoring of a piece, in expectation, with the listener with a higher score scoring a piece lower on the scale in expectation.

## Musician vs. Non Musician

```
ratings_df_musc <- ratings_1 %>%
  mutate(Artist1 = Selfdeclare > 1,
         Artist2 = Selfdeclare > 2,
         Artist3 = Selfdeclare > 3,
         Artist4 = Selfdeclare > 4,
         Artist5 = Selfdeclare > 5)

summary(ratings_df_musc$Selfdeclare)

model_a1 <- lmer(Classical ~ Artist1 * (Instrument + Harmony + Voice) +
                   (1 + Instrument + Harmony | Subject),
                 data = ratings_df_musc, REML = FALSE)
model_a2 <- lmer(Classical ~ Artist2 * (Instrument + Harmony + Voice) +
                   (1 + Instrument + Harmony | Subject),
                 data = ratings_df_musc, REML = FALSE)
model_a3 <- lmer(Classical ~ Artist3 * (Instrument + Harmony + Voice) +
                   (1 + Instrument + Harmony | Subject),
                 data = ratings_df_musc, REML = FALSE)
model_a4 <- lmer(Classical ~ Artist4 * (Instrument + Harmony + Voice) +
                   (1 + Instrument + Harmony | Subject),
                 data = ratings_df_musc, REML = FALSE)
model_a5 <- lmer(Classical ~ Artist5 * (Instrument + Harmony + Voice) +
                   (1 + Instrument + Harmony | Subject),
                 data = ratings_df_musc, REML = FALSE)
model_base <- lmer(Classical ~ Instrument + Harmony + Voice +
                     (1 + Instrument + Harmony | Subject),
                   data = ratings_df_musc, REML = FALSE)

anova(model_a1, model_a2, model_a3, model_a4, model_a5, model_base)
display(model_a1)
display(model_a2)
display(model_a3)
display(model_a4)
display(model_a5)
```

The variable `Selfdeclare` is best bisected by splitting on `Selfdeclare = 3`. All splits are attempted, and their $AIC, BIC$, and $p$ values are reported above, in addition to the base model.

There is extreme variation in interaction based on where we place the split. Considering the "best" split, on 3, we find that "artists" score pieces .01 less classical on average in general, all else equal. They are also less reliant on the instrument than non-artists, scoring pieces with pianos .34 less, and strings 0.83 less in expectation compared to non-artists. They also are more likely to recognize the I-V-VI harmonic progression, and score pieces with it 1.49 points higher on the classical scale in expectation all else equal, than non-artists. They do not change their ratings based on voice all that much, once the other effects are accounted for. The ordinality of these effects is largely preserved for splits on other levels, excepting 1, which likely introduces too many non-artists to really call those in the group "musicians".

## Popular Analysis

```
##  Basic lm() model selection
pop_full <- lm(Popular ~ Instrument*Harmony*Voice, data = ratings_1)

X_mat <- model.matrix(Popular ~ Instrument*Harmony*Voice,
                      data = ratings_1)
Y <- filter(ratings_1, !is.na(Popular))$Popular
nn <- length(Y)

#  Lasso selection
find_min_lambda <- glmnet::cv.glmnet(X_mat, Y, alpha = 1)
pop_lasso <- glmnet::glmnet(X_mat, Y, lambda = find_min_lambda$lambda.1se)
broom::tidy(pop_lasso$beta) %>%
  dplyr::select(-column)

#  Stepwise BIC selection
pop_stepAIC <- stepAIC(pop_full, direction = "both", k = log(nn),
                         trace = F)
summary(pop_stepAIC)

#  All subsets selection
pop_subsets <- regsubsets(Popular ~ Instrument*Harmony*Voice,
                           data = ratings_1,
                           method = "seqrep",
                           nvmax = 36)
coef(pop_subsets, 5)

#  Final lm() model
pop_lm_1 <- lm(Popular ~ Instrument, data = ratings_1)
pop_lm_2 <- lm(Popular ~ Instrument + Harmony, data = ratings_1)
pop_lm_3 <- lm(Popular ~ Harmony*Voice*Instrument, data = ratings_1)
anova(pop_lm_1, pop_lm_2, pop_lm_3)
BIC(pop_lm_1, pop_lm_2, pop_lm_3)
summary(pop_lm_1)
```

```
pop_int <- lmer(Popular ~ (1 | Subject),
                  data = ratings_1, REML = F)
# Basic intercept only model for comparison
flat_model <- lm(Popular ~ 1, data = ratings_1)
display(pop_int)
# Compare single intercept to random intercepts
anova(class_int, flat_model)

big_lmer <- lmer(Popular ~ Instrument * Voice * Harmony + (1 | Subject),
                 data = ratings_1,
                 REML = F)
best_int <- fitLMER.fnc(big_lmer, method = "BIC")
display(best_int)
best_int <- lmer(Popular ~ Instrument + (1 | Subject),
                 data = ratings_1, REML = FALSE)


best_slo <- lmer(Popular ~ Instrument + (1 + Instrument | Subject),
```

```
                 data = ratings_1, REML = FALSE)

anova(best_int, best_slo)
display(best_slo)
```

Interestingly, the model selected by stepwise BIC includes only `Instrument` as a fixed and random effect. Having an electric guiter makes a song score 6.9 points higher on the popular scale in expectation, all else equal. In contrast having a piano makes it score 1.12 points less, and strings makes it score 3.04 points less.

```
best_model <- lmer(Popular ~ Instrument + (1 + Instrument | Subject),
                 data = ratings_1, REML = FALSE)

model_sd <- update(best_model, . ~ . + Selfdeclare)
model_ci <- update(best_model, . ~ . + ConsInstr)
model_cn <- update(best_model, . ~ . + ConsNotes)
model_in <- update(best_model, . ~ . + Instr.minus.Notes)
model_cl <- update(best_model, . ~ . + ClsListen)
model_th <- update(best_model, . ~ . + Theory)
model_co <- update(best_model, . ~ . + Composing)
model_pp <- update(best_model, . ~ . + PianoPlay)
model_gp <- update(best_model, . ~ . + GuitarPlay)


#  First step selection finds nothing with lower BIC, switching to AIC
anova(model_sd, model_ci, model_cn, model_in,
      model_cl, model_th, model_co, model_pp, model_gp, best_model)

best_model_1 <- update(best_model, . ~ . + Selfdeclare)
model_cn_1 <-  update(best_model_1, . ~ . + ConsNotes)

anova(model_cn_1, best_model_1)
```

No additional covariates should be added according to both BIC. According to AIC we could add `ConsNotes`.

```
model_a1 <- lmer(Popular ~ Artist1 * (Instrument) +
                   (1 + Instrument | Subject),
                 data = ratings_df_musc, REML = FALSE)
model_a2 <- lmer(Popular ~ Artist2 * (Instrument) +
                   (1 + Instrument  | Subject),
                 data = ratings_df_musc, REML = FALSE)
model_a3 <- lmer(Popular ~ Artist3 * (Instrument) +
                   (1 + Instrument  | Subject),
                 data = ratings_df_musc, REML = FALSE)
model_a4 <- lmer(Popular ~ Artist4 * (Instrument ) +
                   (1 + Instrument  | Subject),
                 data = ratings_df_musc, REML = FALSE)
model_a5 <- lmer(Popular ~ Artist5 * (Instrument ) +
                   (1 + Instrument  | Subject),
                 data = ratings_df_musc, REML = FALSE)
model_base <- lmer(Popular ~ Instrument   +
                     (1 + Instrument  | Subject),
                   data = ratings_df_musc, REML = FALSE)

anova(model_a1, model_a2, model_a3, model_a4, model_a5, model_base)
display(model_a1)
```

```
display(model_a2)
display(model_a3)
display(model_a4)
display(model_a5)
```

The variable `Selfdeclare` is best bisected by splitting on `Selfdeclare = 1`. All splits are attempted, and their $AIC, BIC$, and $p$ values are reported above, in addition to the base model.

There is extreme variation in interaction based on where we place the split. Considering the "best" split, on 1, we find that "artists" score pieces 1.42 points more popular on average in general, all else equal. They are also more penalizing to instruments than non-artists, scoring pieces with pianos .57 less, and strings .78 less in expectation compared to non-artists.