Mixed Effects Regression Analysis on Freshman Statistics Course Experiment

Lee, Woo Chan
woochanl@andrew.cmu.edu

Department of Statistics and Data Science, Carnegie Mellon University

November 2021

Abstract

We explore a recent rating experiment with 91 sample project papers from the Freshman Statistics course, and explore the relationship between the ratings and the various factors in the experiment. The data explored includes factors such as rubrics, raters, rating scale, as well as other relevant variables used in the experiment. We use exploratory data analyses, multi-level models, and variable selection techniques to investigate if the various factors in the experiment interact in any interesting ways. We find that the ratings are not entirely indistinguishable among different raters and rubrics, and that the final multi-level model consists of fixed effects, random effects and some interaction terms between the variables *Rater, Semester,* and *Rubric.* Some missing observations for certain variables were either imputed appropriately or deleted to prevent modeling limitations, and it would be worthwhile to take note on the Fall, Spring semester data imbalance when repeating this experiment in the future.

Introduction

[1] Dietrich College at Carnegie Mellon University is in the process of implementing a new "General Education" program for undergraduates. This program specifies a set of courses and experiences that all undergraduates must take, and in order to determine whether the new program is successful, the college hopes to rate student work performed in each of the "Gen Ed" courses each year. Recently the college has been experimenting with rating work in Freshman Statistics, using raters from across the college. In a recent experiment, 91 project papers — referred to as "artifacts" — were randomly sampled from a Fall and Spring section of Freshman Statistics. Three raters from three different departments were asked to rate these artifacts on seven distinct rubrics.

In particular, we will:

- Identify if there are rubrics that tend to receive especially high or low ratings, and whether certain raters tend to give especially high or low ratings
- Investigate whether the 3 raters generally agree on their scores

- Explore various fixed, random effects, and interactions of variables in this experiment (Rater, Semester, Sex, Repeated, Rubric) and examine how they affect the ratings
- Discover additional insight on the data set

Data

The name of the 7 rubrics and their descriptions are shown in Table 1.

Short Name	Full Name	Description
RsrchQ	Research Question	Given a scenario, the student generates, critiques or evaluates a relevant empirical research question.
CritDes	Critique Design	Given an empirical research question, the student critiques or evaluates to what extent a study design convincingly answer that question.
InitEDA	Initial EDA	Given a data set, the student appropriately describes the data and provides initial Exploratory Data Analysis.
SelMeth	Select Method(s)	Given a data set and a research question, the student selects appropriate method(s) to analyze the data.
InterpRes	Interpret Results	The student appropriately interprets the results of the selected method(s).
VisOrg	Visual Organization	The student communicates in an organized, coherent and effective fashion with visual elements (charts, graphs, tables, etc.).
TxtOrg	Text Organization	The student communicates in an organized, coherent and effective fashion with text elements (words, sentences, paragraphs, section and subsection titles, etc.).

Table 1: Rubrics for rating Freshman Statistics projects

The common rating scale for all rubrics is shown in Table 2.

Rating	Meaning
1	Student does not generate any relevant evidence.
2	Student generates evidence with significant flaws.
3	Student generates competent evidence; no flaws, or only minor ones.
4	Student generates outstanding evidence; comprehensive and sophisticated.

Table 2: Rating scale used for all rubrics

The full variables available for analysis are defined in Table 3. Along with the dataset organized exactly as in Table 3, we also used an identical dataset with a slight variation where each row of the data contained just one rating, and its respective rubric labelled in the column with the name ”Rubric”.

Variable Name	Values	Description
(X)	1, 2, 3, . . .	Row number in the data set
Rater	1, 2 or 3	Which of the three raters gave a rating
(Sample)	1, 2, 3, . . .	Sample number
(Overlap)	1, 2, . . . , 13	Unique identifier for artifact seen by all 3 raters
Semester	Fall or Spring	Which semester the artifact came from
Sex	M or F	Sex or gender of student who created the artifact
RsrchQ	1, 2, 3 or 4	Rating on Research Question
CritDes	1, 2, 3 or 4	Rating on Critique Design
InitEDA	1, 2, 3 or 4	Rating on Initial EDA
SelMeth	1, 2, 3 or 4	Rating on Select Method(s)
InterpRes	1, 2, 3 or 4	Rating on Interpret Results
VisOrg	1, 2, 3 or 4	Rating on Visual Organization
TxtOrg	1, 2, 3 or 4	Rating on Text Organization
Artifact	(text labels)	Unique identifier for each artifact
Repeated	0 or 1	1 = this is one of the 13 artifacts seen by all 3 raters

Table 3: Full dataset with relevant variables

Methods

Below we will outline the methods used for each research questions defined in the introduction section.

1. Identify if certain rubrics or raters tend to give especially high or low ratings

Introductory exploratory data analysis (EDA) was performed to investigate numerical summaries and distribution plots for the dataset.

In order to investigate if there are certain rubrics that received especially high or low ratings, we produced a facet plot consisting of individual bar plots showing the rating distributions for each rubrics. We also produced bar plots showing the rating distributions for each raters, to determine if there were particular raters giving especially high or low ratings.

One important fact was that the artifacts that each of the raters graded were not identical. Out of the 91 graded artifacts, there were 13 that were commonly graded by all three raters, while the rest were graded individually. Therefore, to really compare the rating characteristics of the raters, it was necessary to separately explore numerical summaries and distributions just using the subset of 13 common artifacts.

2. Investigate whether the 3 raters generally agree on their scores

Similarly, in order to identify if the 3 raters generally agreed on their scores, it made sense to focus on just the 13 artifacts seen by the 3 raters. The main measure of agreement used was the interclass correlation (ICC), which is the common correlation among the raters' ratings for each artifact. We treated each artifact as a cluster of 3 ratings, and proceeded to fit 7 random-intercept models, one for each rubric, thereafter calculating the 7 ICC values.

Generally, high ICC corresponds to high correlation among raters, and low ICC demonstrates low correlation among the raters. But ICC's cannot tell us which specific raters might be contributing to disagreement. In order to explore into this further, we made a 2-way table of counts for the ratings for each pair of raters, on each rubric. For each table, the percentage of observations on the main diagonal was used to calculate the percent exact agreement between the two raters. The percent exact agreement was used as a measure used to help determine who is agreeing with whom on each rubric.

Next, we re-computed the ICC values using the full dataset in order to observe whether the 7 ICC's for the full dataset agreed with the 7 ICC's for the subset corresponding to the 13 artifacts that all 3 raters saw. This enabled us to determine whether the subset of 13 artifacts was reasonably representative of the full dataset in terms of general agreement of ratings between raters.

3. Explore various fixed, random effects, and interactions of variables

The 3rd research question was further divided into 3 sections.

Fixed effects on the 7 rubric-specific models using the data subset containing 13 common artifacts

We first looked at producing a multi-level model on each of the 7 rubrics using the reduced dataset composed of the 13 common artifacts examined by all the raters. For each of the 7 models, we originally fit a "big" model consisting of all possible fixed effects from the variables *Rater, Semester, and Sex*. Then, backward elmination, a form of variable selection, was performed to reduce the "big" models into models with an optimal subset of fixed effect variables. The likelihood ratio chi-squared test was also used to compare each of the 7 resulting models with its intercept-only models to determine the validity of the variable selection.

Fixed, random effects and interaction terms on the 7 rubric-specific models using the full dataset

When fitting models using the full dataset, it was important to identify and deal with observations that were not defined ("NA"). The 2 NA values for the variable *Rating* was imputed using the mode rating across the specific rubric, because *Rating* is a categorical variable that is ordinal, and there were certain ratings that occur more frequently in each Rubric. However, imputing the variable *Sex* was a harder task as it was not a good idea to simply guess a student's gender. Thus, observations with null value for the *Gender* variable was excluded from consideration.

Similar to the methods used for the 13-artifact subset data, a common "big" model was initially generated for each of the 7 rubrics. Then, backward elimination was performed for each of the models to select the optimal subset of fixed effect.

For those models with a more complicated combination of fixed effects, we investigated whether adding random effects or including interaction terms would result in a better fit. We mainly used t-values and ANOVA (Analysis of Variance) to check for interaction terms, while using AIC and BIC values to compare various random effects.

Fixed, random effects and interaction terms on the "generalized" model

Unlike the two previous sections, this time we used a single "general" model that could similarly explain the 7 rubrics using random effects, without having to fit 7 separate models.

Similar to before, we fit a "big" model with all potential fixed effects included, and performed variable selection through backward elimination. Then, once the optimal fixed effects were chosen, we were able to investigate interaction terms for the model. AIC, BIC and likelihood ratio tests were used to compare models with interactions terms since the models were nested.

Next, random effects were considered from the subset of the chosen fixed effects. We needed to add random effects without random intercepts in order to preserve the structure of the model. After having candidate models with different random effects to choose from, we inspected the AIC and BIC values to select the best model.

Finally, the summary of the best model was generated to interpret the coefficients in the context of the problem statement.

4. Discover additional insight on the data set

Additional EDA was performed using variables not explored in the previous sections.

We also explored for better ways to illustrate or interpret the models we have fitted through residual diagnostics.

Finally, we attempted to justify, based on the data and on where the data came from, imputing the missing Sex value as "M" or "F".

Results

1. Identify if certain rubrics or raters tend to give especially high or low ratings

First, we looked at the rating distribution per rubric in the full dataset. Figure 1 shows a facet plot consisting of multiple bar plots each corresponding to the rating distribution of each rubric.

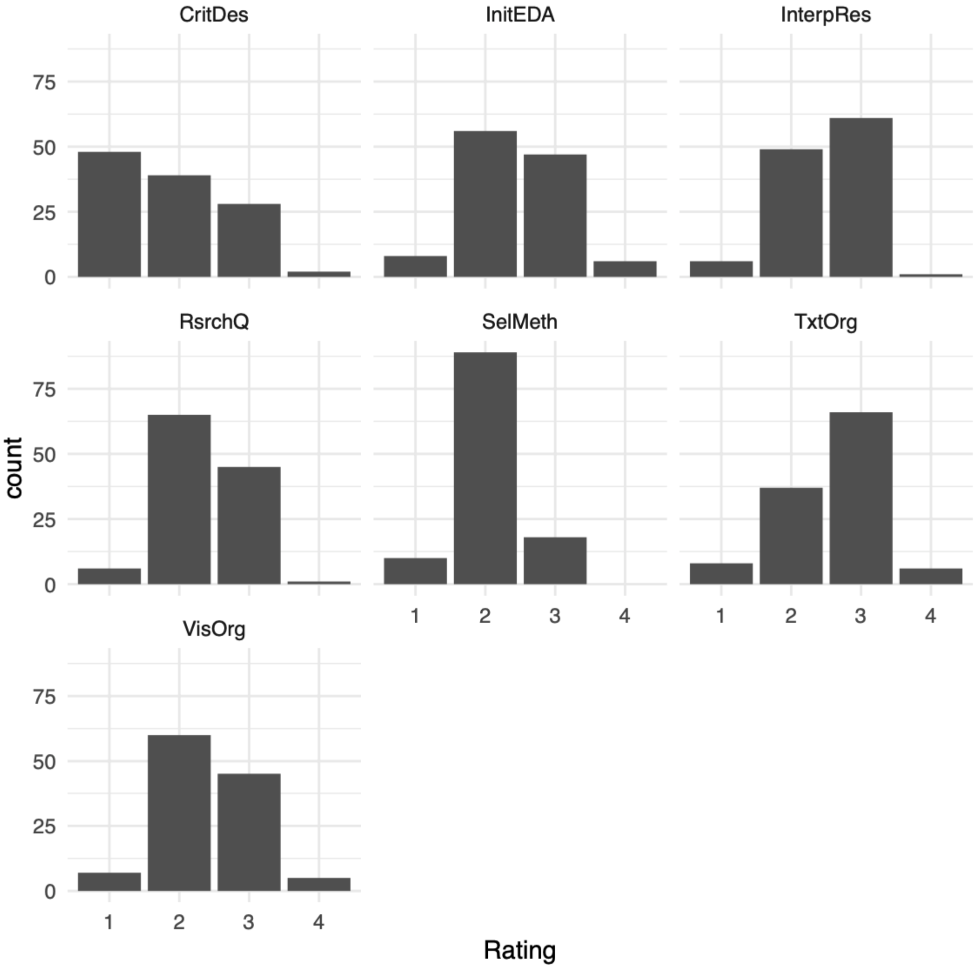


Figure 1: Rating distribution per rubric on full data

Figure 2 shows the rating distribution bar plot per rater (*additional comment to be added).

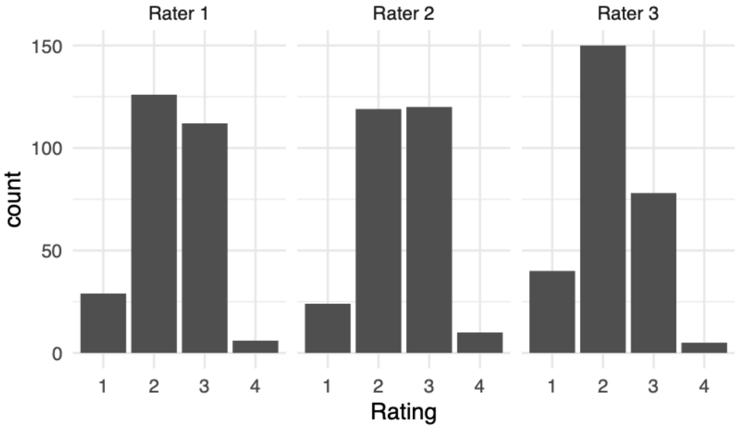


Figure 2: Rating distribution per rater on full data

Next we looked at the rating distribution per rubric for the subset of 13 common artifacts that all 3 raters assessed. Figure 3 shows the facet plot consisting of the 7 bar plots for each rubrics.

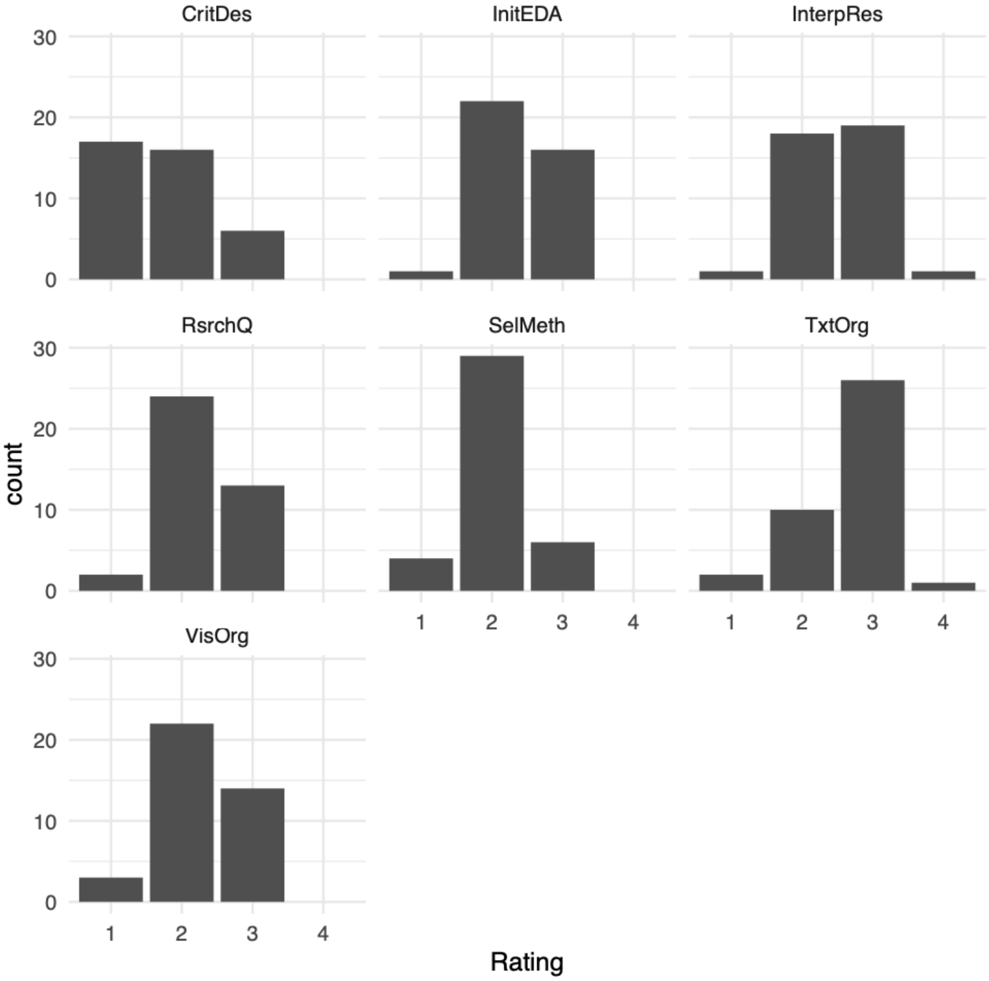


Figure 3: Rating distribution per rater on 13-subset data

Figure 4 shows the rating distribution per rater for the 13-artifact subset (*additional comment to be added).

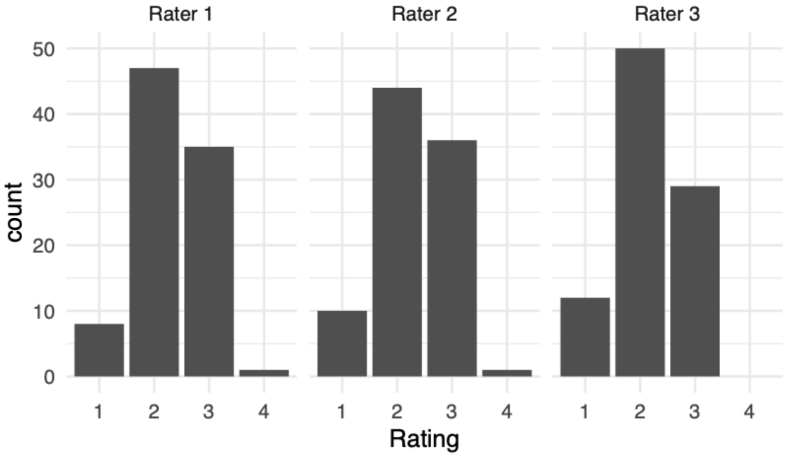


Figure 4: Rating distribution per rater on full data

(Talk about summary table, mean, median) (Talk about table, if certain rubrics have more of certain rating than others)

2. Investigate whether the 3 raters generally agree on their scores

Table 4 below shows the ICC's for each of the 7 models corresponding to each rubric, and the percent exact agreement between 2 raters for each rubric.

Rubric Names	ICC (13-artifacts)	ICC (Full)	a12	a13	a23
RsrchQ	0.19	0.21	0.38	0.77	0.54
CritDes	0.57	0.67	0.54	0.62	0.69
InitEDA	0.49	0.69	0.69	0.54	0.85
SelMeth	0.52	0.47	0.92	0.62	0.69
InterpRes	0.23	0.22	0.62	0.54	0.62
VisOrg	0.59	0.66	0.54	0.77	0.77
TxtOrg	0.14	0.19	0.69	0.62	0.54

Table 4: ICC and percent exact agreement summary table

We can notice that in general, the ICC value for *RsrchQ, InterpRes*, and *TxtOrg* is low, meaning that any two raters did not give a high proportion of similar ratings for the same artifact. On the other hand, for the rest of the rubrics *CritDes, InitEDA, SelMeth*, and *VisOrg*, we can see that the ICC's are a lot higher, suggesting high correlation among the raters.

(Additonal comment on percent exact agreement does not really agree with trend of ICC)

(Additional comment about ICC for 13 artifacts vs full data - are they similar?)

3. Explore various fixed, random effects, and interactions of variables

The results section for the 3rd research question is further divided into 3 sections.

Fixed effects on the 7 rubric-specific models using the data subset containing 13 common artifacts

Page 17 of the technical appendix shows the initial "big" model consisting of all possible fixed effects using the variables *Rater, Semester,* and *Sex*. Note that the variable *Repeat* was not used because the data used was a subset consisting of the 13 common artifacts that all raters graded.

Using backward elimination variable selection in pages 18 and 19 of the technical appendix, we were able to find out that for all 7 models, none of the fixed effect variables were retained, and there was no need to check for any interaction terms or additional random effects.

Fixed, random effects and interaction terms on the 7 rubric-specific models using the full dataset

Similarly, we repeated the variable selection on the 7 rubric-specific models for the full dataset. In pages 19 and 20 of the technical appendix, we were able to find that for 3 rubric-specific models *InitEDA, RsrchQ,* and *TxtOrg*, none of the fixed effect variables were retained. But for the rest of the 4 rubric-specific models, the fixed effect subset turned out to be a bit more complex.

Page 20 of the technical appendix shows the specific formula for each of the 4 non-null models. Next, we were able to use ANOVA t-tests to find out whether we would need to include additional interaction terms and random effects or not. Pages 20 to 27 of the technical appendix details the process of choosing interaction terms and random effects. Consequently, we found out that the variable *Rater* turned out to be statistically significant and important to all of the 4 models. Interaction terms turned out to be insignificant and random effects were not needed because there were more random effects than there were observations in the dataset. This meant that the *lmer()* function could not properly fit a model.

We also computed the ICC's for each of the 7 fitted multi-level models in page 27 of the technical appendix. The general trend and pattern of the ICC across each rubric was similar to the 7 models fitted previously in section 2, although the magnitude of the values were slightly different.

Fixed, random effects and interaction terms on the "generalized" model

Next, we used a single general model that could similarly explain the 7 rubric using random effects, without having to fit 7 separate models. The initial intercept-only model is shown in page 28 of the technical appendix. In the *random effects* section of the model summary, it was evident that a lot of the random effects were highly correlated to each other. This was understandable because we would expect that if a student receives a high score on one or two of the rubrics, he or she would be likely to score high on the other rubrics as well.

On pages 28 and 29 of the technical appendix, we fit a "full" model with all potential fixed effects, and then performed variable selection through backwards elimination. Page 30 shows the resulting model with the selected fixed effect variables being Rater, Semester, and *Rubric*.

Next, we attempted to include all possible interaction terms using the selected fixed effect variables, then carried out backward elimination again to select the best subset of interaction terms. Page 33 of the technical appendix presents the resulting model with the chosen interaction term being *Rater * Rubric*.

We were then able to use the AIC, BIC values as well as the likelihood ratio tests to compare models that included all the interaction terms, the subset of interaction terms after backward elimination, and no interaction terms at all. Pages 34 and 35 of the technical appendix showed that the AIC value and the likelihood ratio test agreed that the second model with selected subset of interaction terms provided the best fit.

In page 35, we were able to inspect the coefficient summary of the chosen model with selected interaction terms and noticed that most of the interaction terms had statistically significant coefficients, suggesting that the raters do not all use the rubrics in the same manner. There are some rubrics such as InitEDA or RsrchQ where the 3 raters seem to have little difference in grading using those rubrics. But for the others:

- *CritDes*: Rater 1 tends to give the lower score compared to Raters 2 and 3
- *InterpRes*: Rater 3 tends to give the lower score compared to Raters 1 and 2 (-0.75 coefficient for interactions + 0.21 coefficient for Rater 3 = -0.54)
- *SelMeth*: Rater 3 tends to give the lower score compared to Raters 1 and 2 (-0.41 coefficient for interactions + 0.21 coefficient for Rater 3 = -0.20)
- *TxtOrg*: Rater 1 tends to give overall higher score compared to Raters 2 and 3
- *VisOrg*: Rater 2 tends to give overall higher score compared to Raters 1 and 3

We further verified this by observing the facet plot in Figure 5, showing the ratings given by the 3 raters across the 7 different rubrics. This did not mean that a certain rater was simply more harsh than the others, but it told us that all the raters have different interpretations of grading across the different rubrics. This justified that the best model was in fact the selected model with the interaction term *Rater * Rubric* included.

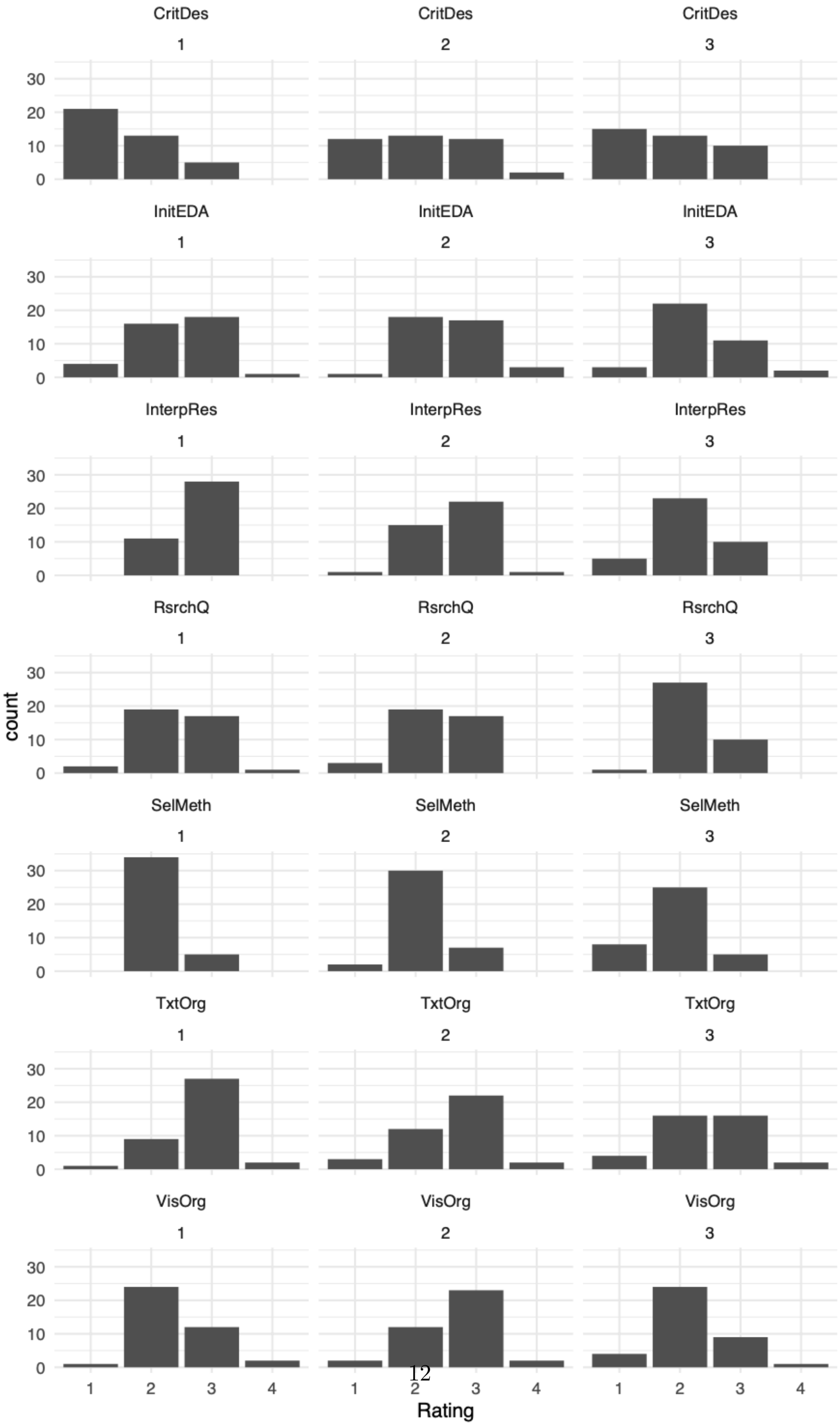


Figure 5: Rating distribution from raters across rubrics

Finally, we attempted to add random effects to this model. We primarily used ANOVA tests to inspect the AIC and BIC values in order to compare different candidate models. Pages 38 and 39 of the technical appendix shows the experimental process of choosing random effects. Ultimately, we chose to add a random effect for the variable *Rater* since it resulted in a smaller AIC and BIC value for the model. The final model and its coefficient summary is presented in page 39 and 40 of the technical appendix.

We can interpret the final model as below (more info to be included in final report):

- **(0 + Rater) | Artifact) + Rater**: There is a *Rater x Artifact* interaction. (elaborate)
- **Rubric + Rater + Rater * Rubric**: There is a *Rater x Rubric* interaction. (elaborate)
- **(0 + Rubric | Artifact) + Rubric**: There is a kind of *Rubric x Artifact* interaction. (elaborate)

4. Discover additional insight on the data set

Most of the models seem to agree that *RsrchQ, InterpRes*, and *TxtOrg* give the lowest ICC value. This means that the raters tend to disagree in their ratings for these 3 rubrics the most. Having more data to work on could solidify this statement and give more reliable results in the degree of agreement between different raters. But the final model that we constructed in research question 3 seems to explain most of the important information, where we left *Rubric* and *Rater* as random effects across *Artifacts*, and having interaction terms better explain the *Rating* variation in the data.

In addition to fitting a multi-level model, we also tried to compare summary statistics according to student gender. Page 41 of the technical appendix shows that the mean and median ratings are almost identical for male and female students, suggesting that there are no apparent rating differences between the two genders of which the artifact was written by.

Discussion

1. Identify if certain rubrics or raters tend to give especially high or low ratings

By examining the distribution plots for rating, we were able to notice that there were in fact certain rubrics that tended to give higher ratings than others.

Looking at summary tables for each rater, we were also able to spot slight differences in ratings given by each rater (median, mean values)

2. Investigate whether the 3 raters generally agree on their scores

(Talk about trends seen in ICC table)

It was found that percent exact agreement did not accurately follow the trends set by the ICC's calculated for the 7 models, because in the two way table, only the observations on the main diagonal was considered for the exact agreement percentages, even though raters may have graded a rubric similarly (one rating off difference).

It was possible to do ICC calculations on the full data set but not percent exact agreement calculations. More specifically, we were not able to replicate the two-way table and the percentage of exact agreement because the raters all graded different artifacts. It was however possible when using the 13 common artifacts that were graded by all 3 raters, since the observations for each raters all intersected with each other.

3. Explore various fixed, random effects, and interactions of variables

In the third research question, we were able to fit multi-level models for both the individual 7-rubric models, and the more ”generalized” model. For the individual models, it turned out that interaction terms and random effects did not serve to be significantly important. This is reasonable because we are in fact fitting individual models for each of the rubrics that may affect the ratings in a different manner. There could be no need to add any more complex interactions or random effects to explain the relationship further.

The ”generalized” model, however, provided a more interesting selection of variables. *Rater* and *Rubric* both turned out to be significant in determining the ratings of an artifact, and several combinations of interaction terms and random effects were useful. In all of this, the fact that Rubric scores depend on Artifact, from the Rubric x Artifact ineraction, suggests that the artifacts aren't all of equal quality on each rubric, and so we should expect the average scores on each Rubric to vary from one Artifact to the next. The Rater x Rubric interaction suggests that the Raters are not all interpreting the Rubrics in the same way. The Rater x Artifact interaction suggests that the Raters are not interpreting the evidence in the artifacts in the same way. These interactions suggest that perhaps the raters should be trained more or given more guidance before the grading process, to make the raters' ratings more similar to each other.

One important procedure that had to be undergone was dealing with the missing ”NA” values. Although not a big concern for the 13-artifact commonly rated dataset, the full dataset was identified to contain 2 NA values for Rating and several observations without a defined gender. For the rating, it made sense to impute them with the mode value across that specific Rubric, because *Rating* was categorical, and there were certain Ratings that occur much more frequently in each Rubric. Using the mode here would be highly unlikely to impact the model trends as well.

However, imputing the Sex of the student who didn't report this to either M or F was a much more difficult task, as it is almost impossible and unreasonable to guess a student's gender. Thus, we had to make decision to eliminate these observations from the dataset.

4. Discover additional insight on the data set

For research question 4, additional analyses and EDA were performed on the dataset in order to gain further insight. One interesting question that was not raised in this project was the effect of gender on ratings. However, it turned out that there were no apprent rating differences across student genders of which the artifacts were written by.

It was also noted that there was an apparent imbalance in the Semesters of the observations, where there were much more data on the Fall semester than the Spring. In the future, it might help to bring in more data on Spring observations and re-assess this experiment in order to evaluate any discrepancies.

References

[1] Dietrich College, Carnegie Mellon University . ”Experiment on General Education program for undergraduates” . Project 2 instruction sheet

Project 2 Technical Appendix

Lee, Woo Chan

11/14/2021

Research Question 1

EDA, Distribution of ratings for each rubrics & Distribution of ratings from each rater

NA values for rubric were replaced with median values of each rubric score, since some of the columns turned out to be slightly skewed. There was also an NA value for the *Sex* variable, but this was kept as a third category of “NA” since there was no way for us to know or estimate the true value.

```r
# Dealing with NA values (Replace with mode)
Mode <- function(x) {
  ux <- unique(x)
  ux[which.max(tabulate(match(x, ux)))]
}

CritDes_m <- Mode(ratings$CritDes)
VisOrg_m <- Mode(ratings$VisOrg)

ratings$CritDes[is.na(ratings$CritDes)] <- CritDes_m
ratings$VisOrg[is.na(ratings$VisOrg)] <- VisOrg_m
```

```r
# Dealing with NA values for tall dataset
# Ratings were replaced with the mode for that rubric
tall_ratings$Rating[tall_ratings$Rubric == 'CritDes'][44] <- CritDes_m
tall_ratings$Rating[tall_ratings$Rubric == 'VisOrg'][99] <- VisOrg_m

# The "ratings" data frames has 1 row where the missing "Sex" value is denoted
#as "--", while in the "tall_ratings" data frame it is denoted as ""
#(string of length 0).
# We will make the "tall_ratings" be consistent by changing it to "--"
tall_ratings$Sex[is.na(tall_ratings$Sex)] <- "--"
```

```r
# Make sure that all ratings run from 1 to 4
tall_ratings$Rating <- factor(tall_ratings$Rating,levels=1:4)
```

Below are EDA tables for categorical variables *Rater*, *Semester*, *Sex* and *Repeated*.

```
# Summary of categorical variables
ratings_cat <- ratings %>% as_tibble() %>%
  # First change categorical to factor
  mutate(
    Rater = as.factor(Rater),
    Semester = as.factor(Semester),
    Sex = as.factor(Sex),
    Repeated = as.factor(Repeated)
  ) %>%
  dplyr::select(Rater, Semester, Sex, Repeated)
```

```
# Summary of Rater
table(ratings$Rater)
```

```
##
##  1  2  3
## 39 39 39
```

```
# Summary of Semester
table(ratings$Semester)
```

```
##
##   Fall Spring
##     83     34
```

```
# Summary of Sex
table(ratings$Sex)
```

```
##
## --   F  M
##  1 64 52
```

```
# Summary of Repeated
table(ratings$Repeated)
```

```
##
##  0  1
## 78 39
```

Below are the summary statistics for the ratings (continuous variables). We can notice that the mean score for *CritDes* is 1.87, which is considerably lower than the other rubrics. *TxtOrg* seems to be another rubric that tends to have a mean score of 2.6, which is significantly higher than the other rubrics.

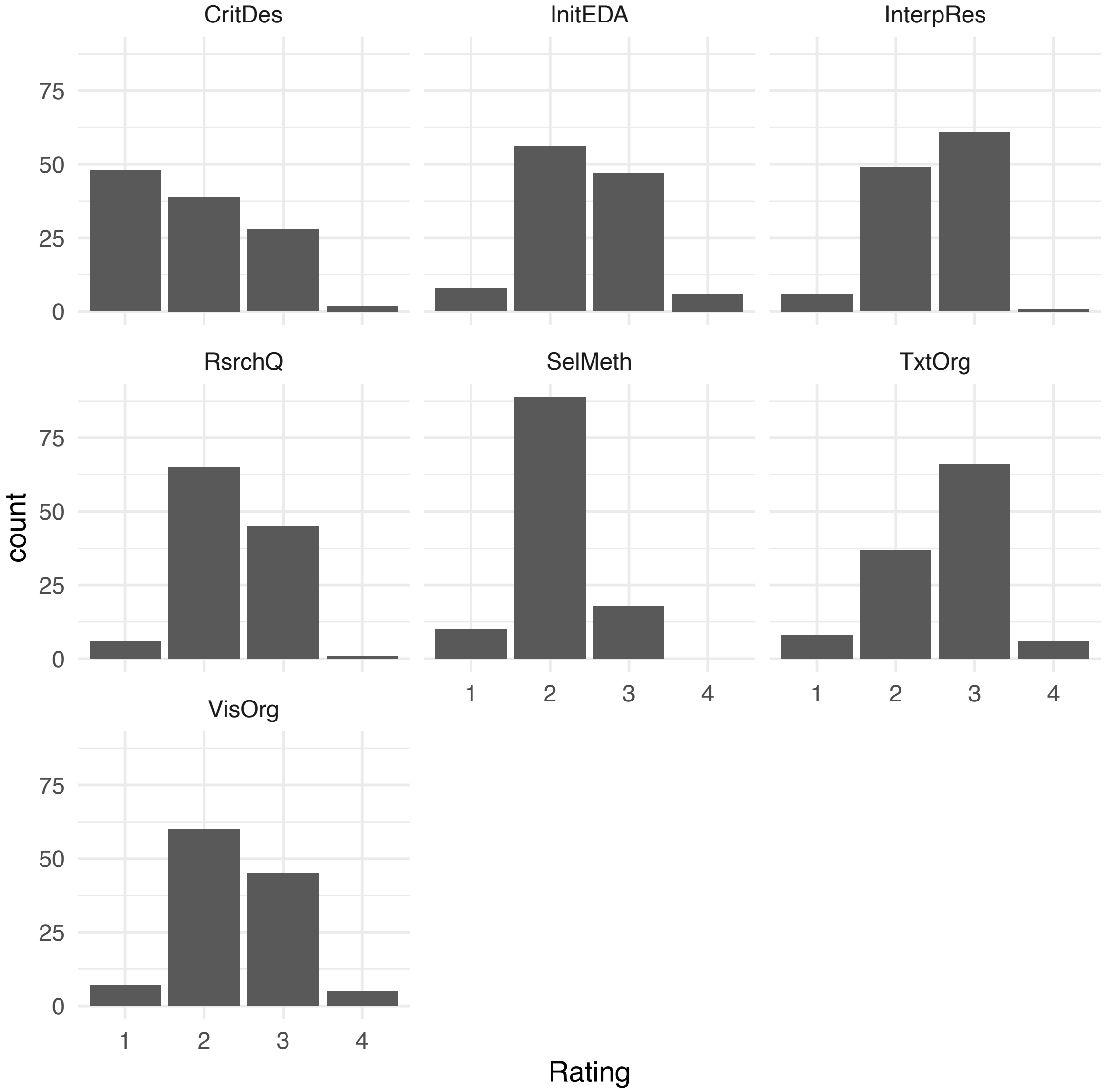
Table 1: Full Dataset

	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.	SD
RsrchQ	1	2	2	2.35	3	4	0.59
CritDes	1	1	2	1.86	3	4	0.84
InitEDA	1	2	2	2.44	3	4	0.70
SelMeth	1	2	2	2.07	2	3	0.49
InterpRes	1	2	3	2.49	3	4	0.61
VisOrg	1	2	2	2.41	3	4	0.67
TxtOrg	1	2	3	2.60	3	4	0.70

```r
# Summary statistics of continuous variables
ratings_con <- ratings %>% as_tibble() %>%
  dplyr::select(RsrchQ, CritDes, InitEDA, SelMeth, InterpRes, VisOrg, TxtOrg)

apply(ratings_con, 2, function(x) c(summary(x),SD=sd(x))) %>%
  as.data.frame %>% t() %>%
  round(digits=2) %>%
  kbl(booktabs=T,caption="Full Dataset") %>%
  kable_classic()
```

Below are the histograms for each of the rubric scores (continuous variables in the dataset). Apart from *CritDes* being skewed to the right, most of the variables seem to be relatively symmetric.

```r
g <- ggplot(tall_ratings, aes(x=Rating)) +
  facet_wrap( ~ Rubric) +
  geom_bar() + theme_minimal()

g
```



Below is the table of counts for each raters giving a certain rating. Rater 3 seems to have given a lot more ratings of 2, and gave the least number of the higest rating 4.

```
tmp0 <- lapply(split(tall_ratings$Rating,tall_ratings$Rater),summary)
tmp <- data.frame(matrix(0,nrow=5,ncol=3))  ## three raters...
names(tmp) <- names(tmp0)
row.names(tmp) <- c(paste("Rating",1:4),"<NA>")
for (i in names(tmp0)) {
  tmp[,i] <- tmp[,i] + c(tmp0[[i]],0)[1:5]
}
names(tmp) <- paste("Rater",1:3)
tmp
```

```
##          Rater 1 Rater 2 Rater 3
## Rating 1      29      24      40
## Rating 2     126     119     150
## Rating 3     112     120      78
```

Table 2: Rater 1

	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.	SD
RsrchQ	1	2.0	2	2.44	3	4	0.64
CritDes	1	1.0	1	1.59	2	3	0.72
InitEDA	1	2.0	2	2.41	3	4	0.72
SelMeth	2	2.0	2	2.13	2	3	0.34
InterpRes	2	2.0	3	2.72	3	3	0.46
VisOrg	1	2.0	2	2.38	3	4	0.63
TxtOrg	1	2.5	3	2.77	3	4	0.58

```
## Rating 4       6      10       5
## <NA>           0       0       0
```

We can also look at the summary table for each raters in order to see if there are raters that give especially high or low ratings. In the three tables below, we can see that Rater 1 gave significantly lower scores for *CritDes* and higher scores for *TxtOrg*. Rater 2 and 3 tended to give relatively balanced scores for all rubrics.

```
rater_1 <- ratings %>% as_tibble() %>% filter(Rater == 1) %>%
  dplyr::select(RsrchQ, CritDes, InitEDA, SelMeth, InterpRes, VisOrg, TxtOrg)
rater_2 <- ratings %>% as_tibble() %>% filter(Rater == 2) %>%
  dplyr::select(RsrchQ, CritDes, InitEDA, SelMeth, InterpRes, VisOrg, TxtOrg)
rater_3 <- ratings %>% as_tibble() %>% filter(Rater == 3) %>%
  dplyr::select(RsrchQ, CritDes, InitEDA, SelMeth, InterpRes, VisOrg, TxtOrg)
```

```
# Summary table for Rater 1
apply(rater_1, 2, function(x) c(summary(x),SD=sd(x))) %>%
  as.data.frame %>% t() %>%
  round(digits=2) %>%
  kbl(booktabs=T,caption="Rater 1") %>%
  kable_classic()
```

```
# Summary table for Rater 2
apply(rater_2, 2, function(x) c(summary(x),SD=sd(x))) %>%
  as.data.frame %>% t() %>%
  round(digits=2) %>%
  kbl(booktabs=T,caption="Rater 2") %>%
  kable_classic()
```

```
# Summary Table for Rater 3
apply(rater_3, 2, function(x) c(summary(x),SD=sd(x))) %>%
  as.data.frame %>% t() %>%
```

Table 3: Rater 2

	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.	SD
RsrchQ	1	2	2	2.36	3	3	0.63
CritDes	1	1	2	2.10	3	4	0.91
InitEDA	1	2	3	2.56	3	4	0.68
SelMeth	1	2	2	2.13	2	3	0.47
InterpRes	1	2	3	2.59	3	4	0.59
VisOrg	1	2	3	2.64	3	4	0.67
TxtOrg	1	2	3	2.59	3	4	0.72

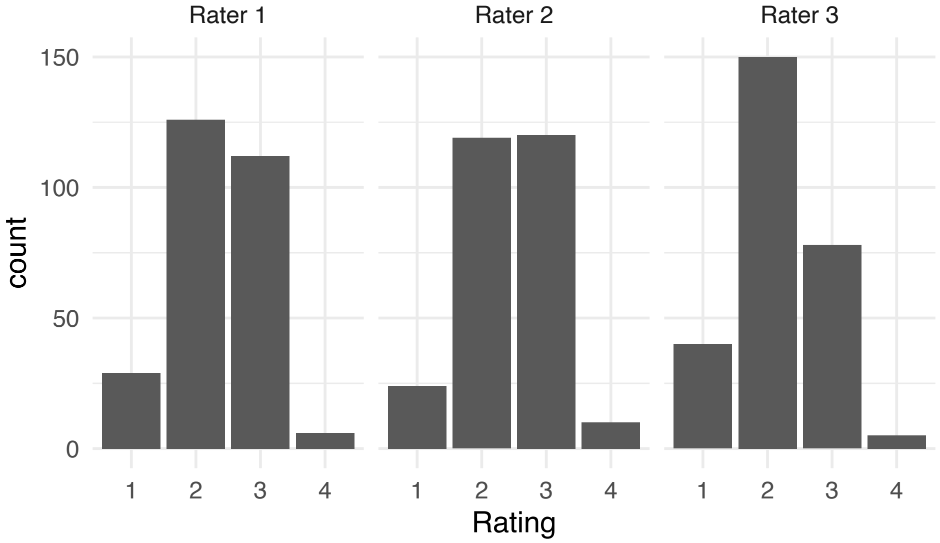
Table 4: Rater 3

	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.	SD
RsrchQ	1	2	2	2.26	3	3	0.50
CritDes	1	1	2	1.90	3	3	0.82
InitEDA	1	2	2	2.33	3	4	0.70
SelMeth	1	2	2	1.95	2	3	0.60
InterpRes	1	2	2	2.15	3	3	0.63
VisOrg	1	2	2	2.21	3	4	0.66
TxtOrg	1	2	2	2.44	3	4	0.75

```r
  round(digits=2) %>%
  kbl(booktabs=T,caption="Rater 3") %>%
  kable_classic()
```

Below are the barplots for each Rater on the Ratings they gave.

```r
## Barplots for full data
rater.name <- function(x) { paste("Rater",x) }

g <- ggplot(tall_ratings,aes(x = Rating)) +
  facet_wrap( ~ Rater, labeller=labeller(Rater=rater.name)) +
  geom_bar() + theme_minimal()

g
```


Rater 1
Rater 2
Rater 3
150
100
50
0
count
1
2
3
4
Rating

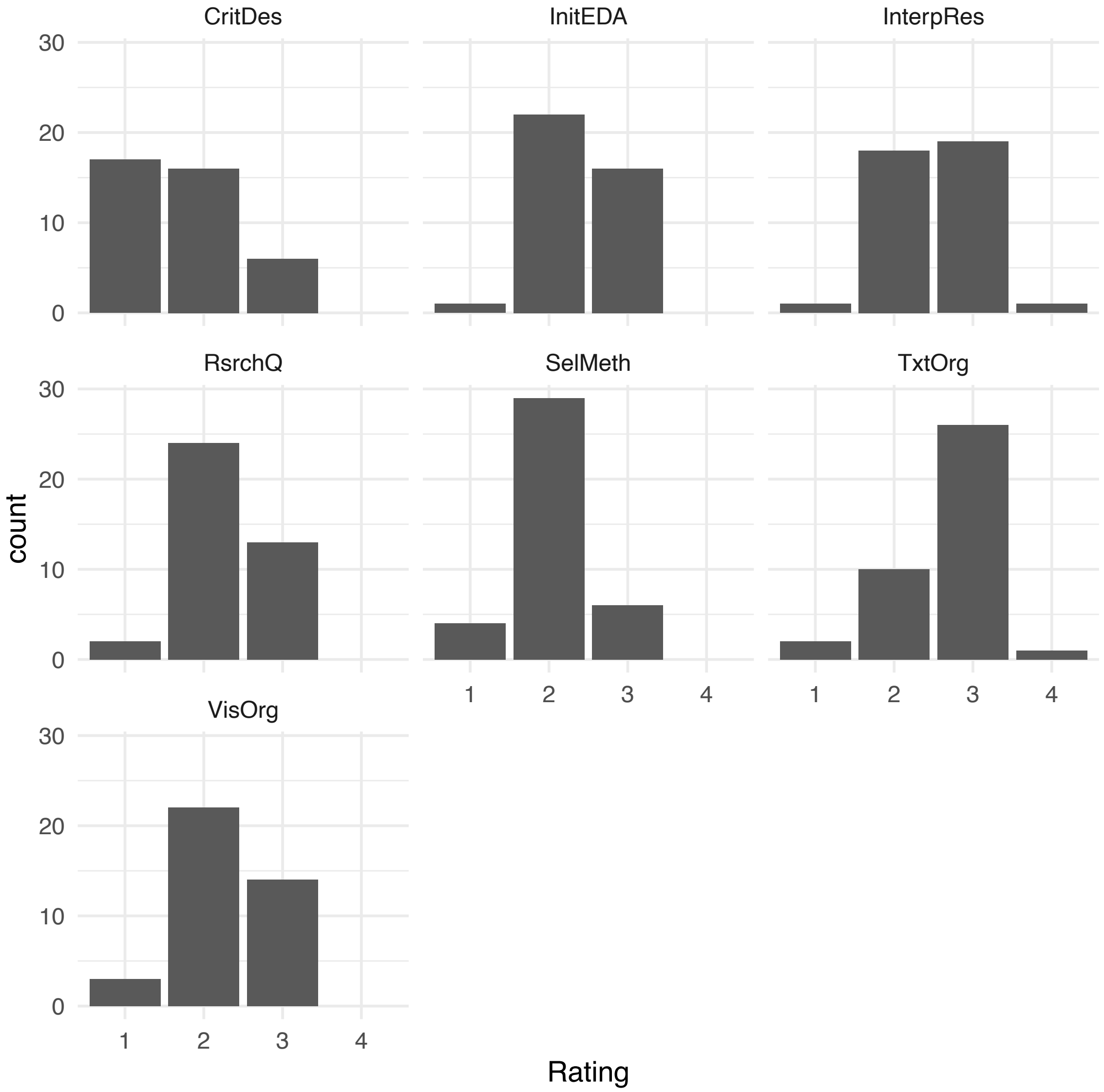
EDA for reduced 13 common artifacts

Next we will look at the distribution for the 13 artifacts seen by all three raters. This will be done by subsetting out the 39 rows that are "repeated". The summary statistics and the histogram will be explored below. We can observe that there doesn't seem to be a big difference in mean or median value, and the shape of the histograms are also very similar. The 13 aritifacts are relatively representative of the 91 artifacts.

```r
# 13 artifcats seen by all three raters
ratings_repeat <- ratings[grep("0",ratings$Artifact),]
tall_repeat <- tall_ratings[grep("0",tall_ratings$Artifact),]
```

First, we will look at the barplots of the reduced 13 artifact dataset. (Ratings vs count)

```r
# Barplots for reduced 13 artifact dataset
g <- ggplot(tall_repeat,aes(x = Rating)) +
  facet_wrap( ~ Rubric) +
  geom_bar() + theme_minimal()
g
```



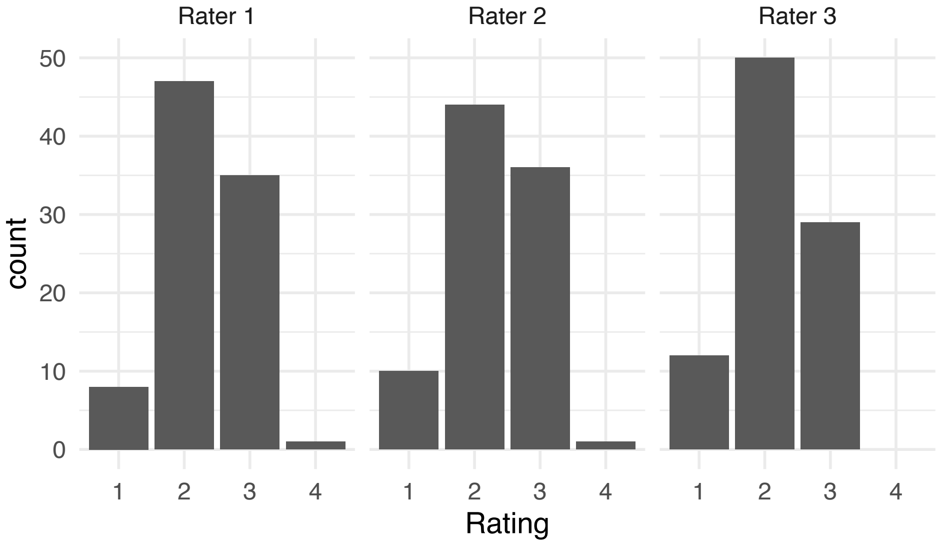
Below is a table of counts for the reduced 13 artifacts dataset. (Ratings vs Rubric)

```
# Table of counts for reduced 13 artifacts
tmp <- data.frame(lapply(split(tall_repeat$Rating,tall_repeat$Rubric),summary))
row.names(tmp) <- paste("Rating",1:4)

tmp
```

```
##          CritDes InitEDA InterpRes RsrchQ SelMeth TxtOrg VisOrg
## Rating 1      17       1         1      2       4      2      3
## Rating 2      16      22        18     24      29     10     22
## Rating 3       6      16        19     13       6     26     14
## Rating 4       0       0         1      0       0      1      0
```

Below are the graphs to compare distributions across Raters. (13 artifacts)

```
## Barplots for reduced 13 artifacts data
g <- ggplot(tall_repeat,aes(x = Rating)) +
  facet_wrap( ~ Rater, labeller=labeller(Rater=rater.name)) +
  geom_bar() + theme_minimal()

g
```



Re-investigating the NA values. We notice from our previous EDA plots and the table below that the NA values do not occur at the reduced 13 common artifacts dataset. And since the main analysis will be related to these 13-artifact dataset, we do not have to worry about any missing data. A key evidence for this is the *Repeated* column in the tables below, which shows that value 0.

```
# Table showing and identifying NA observations
tall_ratings[apply(tall_ratings,1,function(x){any(is.na(x))}),]
```

```
## # A tibble: 0 x 8
## # ... with 8 variables: ...1 <dbl>, Rater <dbl>, Artifact <chr>,
## #   Repeated <dbl>, Semester <chr>, Sex <chr>, Rubric <chr>, Rating <fct>
```

```
ratings[ratings$Sex=="--",]
```

```
## # A tibble: 1 x 15
##    ...1 Rater Sample Overlap Semester Sex   RsrchQ CritDes InitEDA SelMeth
##   <dbl> <dbl>  <dbl>   <dbl> <chr>    <chr>  <dbl>   <dbl>   <dbl>   <dbl>
## 1     5     3      5      NA Fall     --         3       3       3       3
## # ... with 5 more variables: InterpRes <dbl>, VisOrg <dbl>, TxtOrg <dbl>,
## #   Artifact <chr>, Repeated <dbl>
```

Research question 2

ICC values for each rubric

```
Rubric.names <- sort(unique(tall_ratings$Rubric))
Rubric.names
```

```
## [1] "CritDes"   "InitEDA"   "InterpRes" "RsrchQ"    "SelMeth"   "TxtOrg"   
## [7] "VisOrg"
```

```
common <- tall_ratings[grep("O", tall_ratings$Artifact),]
head(common)
```

```
## # A tibble: 6 x 8
##    ...1 Rater Artifact Repeated Semester Sex   Rubric Rating
##   <dbl> <dbl> <chr>       <dbl> <chr>    <chr> <chr>  <fct> 
## 1     1     3 O5              1 F19      M     RsrchQ 3     
## 2     2     3 O7              1 F19      F     RsrchQ 3     
## 3     3     3 O9              1 S19      F     RsrchQ 2     
## 4     4     3 O8              1 S19      M     RsrchQ 2     
## 5    10     3 O10             1 F19      F     RsrchQ 2     
## 6    11     3 O13             1 F19      M     RsrchQ 2
```

```
dim(common)
```

```
## [1] 273   8
```

Get the data subset for RsrchQ rubric for the 13 common artifacts (13 * 3 raters = 39 total)

```
RsrchQ_ratings <- common[common$Rubric == "RsrchQ", ]
CritDes_ratings <- common[common$Rubric == "CritDes", ]
InitEDA_ratings <- common[common$Rubric == "InitEDA", ]
SelMeth_ratings <- common[common$Rubric == "SelMeth", ]
InterpRes_ratings <- common[common$Rubric == "InterpRes", ]
VisOrg_ratings <- common[common$Rubric == "VisOrg", ]
TxtOrg_ratings <- common[common$Rubric == "TxtOrg", ]
```

Next we fit an lmer model (random intercept model) for each rubrics.

```
RsrchQ_lmer <- lmer(as.numeric(Rating) ~ 1 + (1|Artifact), data=RsrchQ_ratings)
CritDes_lmer <- lmer(as.numeric(Rating) ~ 1 + (1|Artifact), data=CritDes_ratings)
InitEDA_lmer <- lmer(as.numeric(Rating) ~ 1 + (1|Artifact), data=InitEDA_ratings)
SelMeth_lmer <- lmer(as.numeric(Rating) ~ 1 + (1|Artifact), data=SelMeth_ratings)
InterpRes_lmer <- lmer(as.numeric(Rating) ~ 1 + (1|Artifact), data=InterpRes_ratings)
VisOrg_lmer <- lmer(as.numeric(Rating) ~ 1 + (1|Artifact), data=VisOrg_ratings)
TxtOrg_lmer <- lmer(as.numeric(Rating) ~ 1 + (1|Artifact), data=TxtOrg_ratings)
```

Below are the ICC values for each rubric, showing the correlation between any two raters on the same artifact. We can notice that the ICC value for *RsrchQ*, *InterpRes*, and *TxtOrg* is low, meaning that any two raters did not give a high proportion of similar ratings for the same artifact.

```
icc_values <- c(icc(RsrchQ_lmer)$ICC_adjusted, icc(CritDes_lmer)$ICC_adjusted,
                icc(InitEDA_lmer)$ICC_adjusted, icc(SelMeth_lmer)$ICC_adjusted,
                icc(InterpRes_lmer)$ICC_adjusted, icc(VisOrg_lmer)$ICC_adjusted,
                icc(TxtOrg_lmer)$ICC_adjusted)

rubric_names <- c("RsrchQ", "CritDes", "InitEDA", "SelMeth", "InterpRes",
               "VisOrg", "TxtOrg")

icc_df <- data.frame(rubric_names, round(icc_values, 2))
icc_df
```

```
##   rubric_names round.icc_values..2.
## 1       RsrchQ                 0.19
## 2      CritDes                 0.57
## 3      InitEDA                 0.49
## 4      SelMeth                 0.52
## 5    InterpRes                 0.23
## 6       VisOrg                 0.59
## 7       TxtOrg                 0.14
```

Two-way table for ratings given by each raters.

The first is for *RsrchQ*. The percent exact agreement for each pair of raters is shown below each tables.

```
rater123_RsrchQ <- data.frame(r1=ratings_repeat$RsrchQ[ratings_repeat$Rater==1],
                              r2=ratings_repeat$RsrchQ[ratings_repeat$Rater==2],
                              r3=ratings_repeat$RsrchQ[ratings_repeat$Rater==3],
                              a1=ratings_repeat$Artifact[ratings_repeat$Rater==1],
                              a2=ratings_repeat$Artifact[ratings_repeat$Rater==2],
                              a3=ratings_repeat$Artifact[ratings_repeat$Rater==3]
)
r1 <- factor(rater123_RsrchQ$r1,levels=1:4)
r2 <- factor(rater123_RsrchQ$r2,levels=1:4)
r3 <- factor(rater123_RsrchQ$r3,levels=1:4)

t12 <- table(r1,r2)
RsrchQ_r12 <- sum(diag(t12))/sum(t12)
t13 <- table(r1,r3)
RsrchQ_r13 <- sum(diag(t13))/sum(t13)
t23 <- table(r2,r3)
RsrchQ_r23 <- sum(diag(t23))/sum(t23)
```

Then for *CritDes*. The percent exact agreement for each pair of raters is shown below each tables.

```
rater123_CritDes <- data.frame(r1=ratings_repeat$CritDes[ratings_repeat$Rater==1],
                               r2=ratings_repeat$CritDes[ratings_repeat$Rater==2],
                               r3=ratings_repeat$CritDes[ratings_repeat$Rater==3],
                               a1=ratings_repeat$Artifact[ratings_repeat$Rater==1],
                               a2=ratings_repeat$Artifact[ratings_repeat$Rater==2],
                               a3=ratings_repeat$Artifact[ratings_repeat$Rater==3]
)
r1 <- factor(rater123_CritDes$r1,levels=1:4)
r2 <- factor(rater123_CritDes$r2,levels=1:4)
r3 <- factor(rater123_CritDes$r3,levels=1:4)

t12 <- table(r1,r2)
CritDes_r12 <- sum(diag(t12))/sum(t12)
t13 <- table(r1,r3)
CritDes_r13 <- sum(diag(t13))/sum(t13)
t23 <- table(r2,r3)
CritDes_r23 <- sum(diag(t23))/sum(t23)
```

Then for *InitEDA*. The percent exact agreement for each pair of raters is shown below each tables.

```
rater123_InitEDA <- data.frame(r1=ratings_repeat$InitEDA[ratings_repeat$Rater==1],
                               r2=ratings_repeat$InitEDA[ratings_repeat$Rater==2],
                               r3=ratings_repeat$InitEDA[ratings_repeat$Rater==3],
                               a1=ratings_repeat$Artifact[ratings_repeat$Rater==1],
                               a2=ratings_repeat$Artifact[ratings_repeat$Rater==2],
                               a3=ratings_repeat$Artifact[ratings_repeat$Rater==3]
)
r1 <- factor(rater123_InitEDA$r1,levels=1:4)
r2 <- factor(rater123_InitEDA$r2,levels=1:4)
r3 <- factor(rater123_InitEDA$r3,levels=1:4)

t12 <- table(r1,r2)
InitEDA_r12 <- sum(diag(t12))/sum(t12)
t13 <- table(r1,r3)
InitEDA_r13 <- sum(diag(t13))/sum(t13)
t23 <- table(r2,r3)
InitEDA_r23 <- sum(diag(t23))/sum(t23)
```

Then for *SelMeth*. The percent exact agreement for each pair of raters is shown below each tables.

```
rater123_SelMeth <- data.frame(r1=ratings_repeat$SelMeth[ratings_repeat$Rater==1],
                               r2=ratings_repeat$SelMeth[ratings_repeat$Rater==2],
                               r3=ratings_repeat$SelMeth[ratings_repeat$Rater==3],
                               a1=ratings_repeat$Artifact[ratings_repeat$Rater==1],
                               a2=ratings_repeat$Artifact[ratings_repeat$Rater==2],
                               a3=ratings_repeat$Artifact[ratings_repeat$Rater==3]
)
r1 <- factor(rater123_SelMeth$r1,levels=1:4)
r2 <- factor(rater123_SelMeth$r2,levels=1:4)
r3 <- factor(rater123_SelMeth$r3,levels=1:4)

t12 <- table(r1,r2)
SelMeth_r12 <- sum(diag(t12))/sum(t12)
```

```
t13 <- table(r1,r3)
SelMeth_r13 <- sum(diag(t13))/sum(t13)
t23 <- table(r2,r3)
SelMeth_r23 <- sum(diag(t23))/sum(t23)
```

Then for *InterpRes*. The percent exact agreement for each pair of raters is shown below each tables.

```
rater123_InterpRes <- data.frame(r1=ratings_repeat$InterpRes[ratings_repeat$Rater==1],
                                   r2=ratings_repeat$InterpRes[ratings_repeat$Rater==2],
                                   r3=ratings_repeat$InterpRes[ratings_repeat$Rater==3],
                                   a1=ratings_repeat$Artifact[ratings_repeat$Rater==1],
                                   a2=ratings_repeat$Artifact[ratings_repeat$Rater==2],
                                   a3=ratings_repeat$Artifact[ratings_repeat$Rater==3]
)
r1 <- factor(rater123_InterpRes$r1,levels=1:4)
r2 <- factor(rater123_InterpRes$r2,levels=1:4)
r3 <- factor(rater123_InterpRes$r3,levels=1:4)

t12 <- table(r1,r2)
InterpRes_r12 <- sum(diag(t12))/sum(t12)
t13 <- table(r1,r3)
InterpRes_r13 <- sum(diag(t13))/sum(t13)
t23 <- table(r2,r3)
InterpRes_r23 <- sum(diag(t23))/sum(t23)
```

Then for *VisOrg*. The percent exact agreement for each pair of raters is shown below each tables.

```
rater123_VisOrg <- data.frame(r1=ratings_repeat$VisOrg[ratings_repeat$Rater==1],
                                   r2=ratings_repeat$VisOrg[ratings_repeat$Rater==2],
                                   r3=ratings_repeat$VisOrg[ratings_repeat$Rater==3],
                                   a1=ratings_repeat$Artifact[ratings_repeat$Rater==1],
                                   a2=ratings_repeat$Artifact[ratings_repeat$Rater==2],
                                   a3=ratings_repeat$Artifact[ratings_repeat$Rater==3]
)
r1 <- factor(rater123_VisOrg$r1,levels=1:4)
r2 <- factor(rater123_VisOrg$r2,levels=1:4)
r3 <- factor(rater123_VisOrg$r3,levels=1:4)

t12 <- table(r1,r2)
VisOrg_r12 <- sum(diag(t12))/sum(t12)
t13 <- table(r1,r3)
VisOrg_r13 <- sum(diag(t13))/sum(t13)
t23 <- table(r2,r3)
VisOrg_r23 <- sum(diag(t23))/sum(t23)
```

Then for *TxtOrg*.

```
rater123_TxtOrg <- data.frame(r1=ratings_repeat$TxtOrg[ratings_repeat$Rater==1],
                              r2=ratings_repeat$TxtOrg[ratings_repeat$Rater==2],
                              r3=ratings_repeat$TxtOrg[ratings_repeat$Rater==3],
                              a1=ratings_repeat$Artifact[ratings_repeat$Rater==1],
                              a2=ratings_repeat$Artifact[ratings_repeat$Rater==2],
                              a3=ratings_repeat$Artifact[ratings_repeat$Rater==3]
)
r1 <- factor(rater123_TxtOrg$r1,levels=1:4)
r2 <- factor(rater123_TxtOrg$r2,levels=1:4)
r3 <- factor(rater123_TxtOrg$r3,levels=1:4)

t12 <- table(r1,r2)
TxtOrg_r12 <- sum(diag(t12))/sum(t12)
t13 <- table(r1,r3)
TxtOrg_r13 <- sum(diag(t13))/sum(t13)
t23 <- table(r2,r3)
TxtOrg_r23 <- sum(diag(t23))/sum(t23)
```

Calculating ICC for full dataset & comparing with 13 common artifacts

```
RsrchQ_ratings <- tall_ratings[tall_ratings$Rubric == "RsrchQ", ]
CritDes_ratings <- tall_ratings[tall_ratings$Rubric == "CritDes", ]
InitEDA_ratings <- tall_ratings[tall_ratings$Rubric == "InitEDA", ]
SelMeth_ratings <- tall_ratings[tall_ratings$Rubric == "SelMeth", ]
InterpRes_ratings <- tall_ratings[tall_ratings$Rubric == "InterpRes", ]
VisOrg_ratings <- tall_ratings[tall_ratings$Rubric == "VisOrg", ]
TxtOrg_ratings <- tall_ratings[tall_ratings$Rubric == "TxtOrg", ]
```

Next we fit an lmer model (random intercept model) for each rubrics.

```
RsrchQ_lmer <- lmer(as.numeric(Rating) ~ 1 + (1|Artifact), data=RsrchQ_ratings)
CritDes_lmer <- lmer(as.numeric(Rating) ~ 1 + (1|Artifact), data=CritDes_ratings)
InitEDA_lmer <- lmer(as.numeric(Rating) ~ 1 + (1|Artifact), data=InitEDA_ratings)
SelMeth_lmer <- lmer(as.numeric(Rating) ~ 1 + (1|Artifact), data=SelMeth_ratings)
InterpRes_lmer <- lmer(as.numeric(Rating) ~ 1 + (1|Artifact), data=InterpRes_ratings)
VisOrg_lmer <- lmer(as.numeric(Rating) ~ 1 + (1|Artifact), data=VisOrg_ratings)
TxtOrg_lmer <- lmer(as.numeric(Rating) ~ 1 + (1|Artifact), data=TxtOrg_ratings)
```

Below are the ICC values for each rubric, showing the correlation between any two raters on the same artifact. It seems like the ICC values are quite similar in pattern (higher and smaller ICC for certain similar rubrics) although slightly different in values. We won't be able to replicate the two-way table and the percentage of exact agreement because the raters graded different artifacts. This was only possible when comparing the 13 common artifcats that were graded by all three raters.

```
icc_values_t <- c(icc(RsrchQ_lmer)$ICC_adjusted, icc(CritDes_lmer)$ICC_adjusted,
                icc(InitEDA_lmer)$ICC_adjusted, icc(SelMeth_lmer)$ICC_adjusted,
                icc(InterpRes_lmer)$ICC_adjusted, icc(VisOrg_lmer)$ICC_adjusted,
                icc(TxtOrg_lmer)$ICC_adjusted)
```

Table 5: ICC and rater agreement table

Rubric Names	ICC (13-artifacts)	ICC (Full)	a12	a13	a23
RsrchQ	0.19	0.21	0.38	0.77	0.54
CritDes	0.57	0.67	0.54	0.62	0.69
InitEDA	0.49	0.69	0.69	0.54	0.85
SelMeth	0.52	0.47	0.92	0.62	0.69
InterpRes	0.23	0.22	0.62	0.54	0.62
VisOrg	0.59	0.66	0.54	0.77	0.77
TxtOrg	0.14	0.19	0.69	0.62	0.54

```
tmp <- data.frame(rubric_names, icc_values_t, icc_values)
tmp
```

```
##   rubric_names icc_values_t icc_values
## 1       RsrchQ    0.2096214  0.1891892
## 2      CritDes    0.6730224  0.5725594
## 3      InitEDA    0.6867210  0.4929577
## 4      SelMeth    0.4719014  0.5212766
## 5    InterpRes    0.2200285  0.2295720
## 6       VisOrg    0.6586320  0.5924529
## 7       TxtOrg    0.1879927  0.1428571
```

Combining all the ICC values to one table

```
# Combine all the exact agreement values into separate vectors (combined)
r12 <- c(RsrchQ_r12, CritDes_r12, InitEDA_r12, SelMeth_r12, InterpRes_r12,
         VisOrg_r12, TxtOrg_r12)
r13 <- c(RsrchQ_r13, CritDes_r13, InitEDA_r13, SelMeth_r13, InterpRes_r13,
         VisOrg_r13, TxtOrg_r13)
r23 <- c(RsrchQ_r23, CritDes_r23, InitEDA_r23, SelMeth_r23, InterpRes_r23,
         VisOrg_r23, TxtOrg_r23)

# Combine all the data frames together
full_icc_table <- data.frame(rubric_names, round(icc_values,2), round(icc_values_t,2),
                             round(r12,2), round(r13,2), round(r23,2))

colnames(full_icc_table) <- c('Rubric Names', 'ICC (13-artifacts)', 'ICC (Full)', 'a12', 'a13', 'a23')
```

```
full_icc_table %>%
  kbl(caption = "ICC and rater agreement table") %>%
  kable_classic(full_width = F, html_font = "Cambria") %>%
  kableExtra::row_spec(2, hline_after = TRUE)
```

Research Question 3

Adding fixed effects to the seven rubric-specific models using just the data from the 13 common artifacts that all three raters saw

I will first explore the seven rubric-specific models and their fixed effects. Note that the results will be impacted in cases where we use the reduced 13 common artifacts dataset, and when using the full dataset. Since we will first be exploring the reduced dataset, there won't be the variable *Repeated* involved, because the reduced dataset consists of all the observations that are "repeated".

```
# Fitting a default model for RsrchQ
# Intercept was removed to prevent an intercept-only model, and rater to be always in the model
big13_RsrchQ <- lmer(as.numeric(Rating) ~ -1 + as.factor(Rater) +
              Semester + Sex + (1 | Artifact),
            data=tall_repeat[tall_repeat$Rubric=="RsrchQ",],REML=FALSE)
```

```
red13_RsrchQ <- fitLMER.fnc(big13_RsrchQ,set.REML.FALSE = TRUE,log.file.name = FALSE)
```

```
## Warning in fitLMER.fnc(big13_RsrchQ, set.REML.FALSE = TRUE, log.file.name = FALSE): Argument "ran.ef
## TRUE
```

```
## ======================================================
## ===              backfitting fixed effects         ===
## ======================================================
## processing model terms of interaction level 1
##   iteration 1
##     p-value for term "Semester" = 0.7355 >= 0.05
##     not part of higher-order interaction
##     removing term
##   iteration 2
##     p-value for term "Sex" = 0.279 >= 0.05
##     not part of higher-order interaction
##     removing term
## pruning random effects structure ...
##   nothing to prune
## ======================================================
## ===            forwardfitting random effects       ===
## ======================================================
##  ===       random slopes       ===
## ======================================================
## ===            re-backfitting fixed effects        ===
## ======================================================
## processing model terms of interaction level 1
##   all terms of interaction level 1 significant
## resetting REML to TRUE
## pruning random effects structure ...
##   nothing to prune
```

We can see that backward elimination resulted in a model with only *Rater* included for RsrchQ.

```
formula(red13_RsrchQ)
```

```
## as.numeric(Rating) ~ as.factor(Rater) + (1 | Artifact) - 1
```

Looking at the fixed effects coefficient for each rater, they don't seem to be too different from each other, ranging from 2.15 ~ 2.3. We will use ANOVA likelihood ratio chi squared test to test if they are different or not. We will be comparing our reduced model with the intercept-only model. Below, we see that the p-value is a lot larger than 0.05, meaning that the intercept model is better. There is also no need to check for interactions because all fixed effect variables were not retained.

```
red13_RsrchQ_int <- update(red13_RsrchQ, . ~ . + 1 - as.factor(Rater))
anova(red13_RsrchQ_int,red13_RsrchQ)$"Pr(>Chisq)"[2]
```

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

```
## [1] 0.4869707
```

Below, I will run the same process for all the 7 rubrics. The code is reused from the homework solutions provided by Dr. Brian Junker.

```
Rubric.names <- sort(unique(tall_ratings$Rubric))

model.formula.13 <- as.list(rep(NA,7))
names(model.formula.13) <- Rubric.names

for (i in Rubric.names) {

  ## fit each base model
  rubric.data <- tall_repeat[tall_repeat$Rubric==i,]
  tmp <- lmer(as.numeric(Rating) ~ -1 + as.factor(Rater) +
              Semester + Sex + (1|Artifact),
            data=rubric.data,REML=FALSE)

  ## do backwards elimination
  tmp.back_elim <- fitLMER.fnc(tmp,set.REML.FALSE = TRUE,log.file.name = FALSE)

  ## check to see if the raters are significantly different from one another
  tmp.single_intercept <- update(tmp.back_elim, . ~ . + 1 - as.factor(Rater))
  pval <- anova(tmp.single_intercept,tmp.back_elim)$"Pr(>Chisq)"[2]

  ## choose the best model
  if (pval<=0.05) {
    tmp_final <- tmp.back_elim
  } else {
    tmp_final <- tmp.single_intercept
  }

  ## and add to list...
  model.formula.13[[i]] <- formula(tmp_final)
}
```

Below we can see the resulting formula / models for the 7 rubrics using the reduced 13-artifact dataset. We can see that none of the fixed effects were retained, and there is no need to check for any interaction terms or additional random effects.

```
model.formula.13
```

```
## $CritDes
## as.numeric(Rating) ~ (1 | Artifact)
```

```
##
## $InitEDA
## as.numeric(Rating) ~ (1 | Artifact)
##
## $InterpRes
## as.numeric(Rating) ~ (1 | Artifact)
##
## $RsrchQ
## as.numeric(Rating) ~ (1 | Artifact)
##
## $SelMeth
## as.numeric(Rating) ~ (1 | Artifact)
##
## $TxtOrg
## as.numeric(Rating) ~ (1 | Artifact)
##
## $VisOrg
## as.numeric(Rating) ~ (1 | Artifact)
```

Adding fixed effects to the seven rubric-specific models using the full dataset

First, we note that for the full dataset, we identified 2 *NA* values for *Rating* and imputed them with the mode value across that specific *Rubric*. Mode makes the most sense to use here, because the variable *Rating* is categorical, and there are certain Ratings that occur much more frequently in each Rubric. And using the mode would be highly unlikely to impact the model trends. However, imputing the *Sex* of the student whoe didn't report this to either M or F is a much more difficult task, as it is almost impossible and unreasonable to guess a student's gender. Thus, I will be elminating this observation from this dataset.

```r
# Eliminate missing "Sex" observation (7 rows)
new_tall_ratings <- tall_ratings[tall_ratings$Sex != "--",]
```

Next, I will refer to the code snipped from HW10 solutions to perform backwards elimination for each of the 7 rubric models, and generate the optimal subset of fixed effects.

```r
model.formula.alldata <- as.list(rep(NA,7))
names(model.formula.alldata) <- Rubric.names

for (i in Rubric.names) {

  ## fit each base model
  rubric.data <- new_tall_ratings[new_tall_ratings$Rubric==i,]
  tmp <- lmer(as.numeric(Rating) ~ -1 + as.factor(Rater) +
                Semester + Sex + (1|Artifact),
              data=rubric.data,REML=FALSE)

  ## do backwards elimination
  tmp.back_elim <- fitLMER.fnc(tmp,set.REML.FALSE = TRUE,log.file.name = FALSE)

  ## check to see if the raters are significantly different from one another
  tmp.single_intercept <- update(tmp.back_elim, . ~ . + 1 - as.factor(Rater))
  pval <- anova(tmp.single_intercept,tmp.back_elim)$"Pr(>Chisq)"[2]

  ## choose the best model
  if (pval<=0.05) {
    tmp_final <- tmp.back_elim
  } else {
    tmp_final <- tmp.single_intercept
  }

  ## and add to list...
  model.formula.alldata[[i]] <- formula(tmp_final)

}
```

Below we can see the "final models" that were generated from variable selection. We can see below that for the Rubrics **InitEDA**, **RsrchQ**, **TxtOrg**, the models are the simple random-intercept models. The other 4 **CritDes**, **InterpRes**, **SelMeth**, and **VisOrg**, the models are more complex, with additional fixed effect variables added. For these models, the variable *Rater* seems to be a common important fixed effect to have.

```
model.formula.alldata
```

```
## $CritDes
## as.numeric(Rating) ~ as.factor(Rater) + (1 | Artifact) - 1
##
## $InitEDA
## as.numeric(Rating) ~ (1 | Artifact)
##
## $InterpRes
## as.numeric(Rating) ~ as.factor(Rater) + (1 | Artifact) - 1
##
## $RsrchQ
## as.numeric(Rating) ~ (1 | Artifact)
##
## $SelMeth
## as.numeric(Rating) ~ as.factor(Rater) + Semester + (1 | Artifact) -
##     1
##
## $TxtOrg
## as.numeric(Rating) ~ (1 | Artifact)
##
## $VisOrg
## as.numeric(Rating) ~ as.factor(Rater) + (1 | Artifact) - 1
```

```
# For the 3 simple models, there is no need to explore further interactions or random effects
fla <- formula(model.formula.alldata[["InitEDA"]])
InitEDA_lmer_final <- lmer(fla,data=new_tall_ratings[new_tall_ratings$Rubric=="InitEDA",])
fla <- formula(model.formula.alldata[["RsrchQ"]])
RsrchQ_lmer_final <- lmer(fla,data=new_tall_ratings[new_tall_ratings$Rubric=="RsrchQ",])
fla <- formula(model.formula.alldata[["TxtOrg"]])
TxtOrg_lmer_final <- lmer(fla,data=new_tall_ratings[new_tall_ratings$Rubric=="TxtOrg",])
```

Now we will look at the 4 models with more fixed effects, to find out whether we would need to include any interaction terms or random effects. For *CritDes*, the t-values below show that the fixed effect *Rater* is statistically significant. The ANOVA results below that shows that the model with *Rater* is better than the intercept-only model without it.

```
# CritDes
fla <- formula(model.formula.alldata[["CritDes"]])
tmp <- lmer(fla,data=new_tall_ratings[new_tall_ratings$Rubric=="CritDes",])
round(summary(tmp)$coef,2)  ## fixed effects and their t-values
```

```
##                    Estimate Std. Error t value
## as.factor(Rater)1     1.68       0.12   13.91
## as.factor(Rater)2     2.09       0.12   17.27
## as.factor(Rater)3     1.88       0.12   15.43
```

```
# CritDes
tmp.single_intercept <- update(tmp, . ~ . + 1 - as.factor(Rater))
anova(tmp.single_intercept,tmp)
```

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

## Data: new_tall_ratings[new_tall_ratings$Rubric == "CritDes", ]
## Models:
## tmp.single_intercept: as.numeric(Rating) ~ (1 | Artifact)
## tmp: as.numeric(Rating) ~ as.factor(Rater) + (1 | Artifact) - 1
##                      npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
## tmp.single_intercept    3 280.29 288.55 -137.14   274.29
## tmp                     5 276.86 290.63 -133.43   266.86 7.4231  2    0.02444 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Next we check for random effects. We will specifically look into adding (Rater | Artifact). Since, there are more random effect parameters than there are observations in the dataset, the model is not even possible (as shown in the error below). Thus, we will stick with the previous model.

```
# CritDes - Random effect: check for (Rater | Artifact)
m0 <- tmp
mA <- update(m0, . ~ . + (as.factor(Rater)|Artifact))
```

```
## Error: number of observations (=116) <= number of random effects (=270) for term (as.factor(Rater) |
```

Below is the final model summary for *CritDes*.

```
CritDes_lmer_final <- tmp
summary(CritDes_lmer_final)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: as.numeric(Rating) ~ as.factor(Rater) + (1 | Artifact) - 1
##    Data: new_tall_ratings[new_tall_ratings$Rubric == "CritDes", ]
##
```

```
## REML criterion at convergence: 274.2
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -1.56945 -0.49096 -0.06388  0.65647  1.64161
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  Artifact (Intercept) 0.4426   0.6653
##  Residual             0.2461   0.4961
## Number of obs: 116, groups:  Artifact, 90
##
## Fixed effects:
##                   Estimate Std. Error t value
## as.factor(Rater)1   1.6816     0.1209   13.91
## as.factor(Rater)2   2.0887     0.1209   17.27
## as.factor(Rater)3   1.8849     0.1221   15.43
##
## Correlation of Fixed Effects:
##             a.(R)1 a.(R)2
## as.fctr(R)2 0.245
## as.fctr(R)3 0.247  0.247
```

Next we look at **InterpRes**. The t-values and ANOVA show that *Rater* is an important fixed effect, and the model with *Rater* is better than the intercept-only model.

```
# InterpRes
fla <- formula(model.formula.alldata[["InterpRes"]])
tmp <- lmer(fla,data=new_tall_ratings[new_tall_ratings$Rubric=="InterpRes",])
round(summary(tmp)$coef,2)  ## fixed effects and their t-values
```

```
##                   Estimate Std. Error t value
## as.factor(Rater)1     2.70       0.09   30.34
## as.factor(Rater)2     2.59       0.09   29.01
## as.factor(Rater)3     2.14       0.09   23.70
```

```
# InterpRes
tmp.single_intercept <- update(tmp, . ~ . + 1 - as.factor(Rater))
anova(tmp.single_intercept,tmp)
```

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

## Data: new_tall_ratings[new_tall_ratings$Rubric == "InterpRes", ]
## Models:
## tmp.single_intercept: as.numeric(Rating) ~ (1 | Artifact)
## tmp: as.numeric(Rating) ~ as.factor(Rater) + (1 | Artifact) - 1
##                      npar    AIC    BIC   logLik deviance  Chisq Df Pr(>Chisq)
## tmp.single_intercept    3 218.53 226.79 -106.263   212.53
## tmp                     5 200.66 214.43  -95.331   190.66 21.864  2  1.787e-05
##
## tmp.single_intercept
## tmp                  ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Next we check for random effects. We will specifically look into adding (Rater | Artifact). Since, there are more random effect parameters than there are observations in the dataset, the model is not even possible (as shown in the error below). Thus, we will stick with the previous model.

```
# InterpRes - Random effect: check for (Rater | Artifact)
m0 <- tmp
mA <- update(m0, . ~ . + (as.factor(Rater)|Artifact))
```

```
## Error: number of observations (=116) <= number of random effects (=270) for term (as.factor(Rater) |
```

Below is the final model summary for *InterpRes*.

```
InterpRes_lmer_final <- tmp
summary(InterpRes_lmer_final)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: as.numeric(Rating) ~ as.factor(Rater) + (1 | Artifact) - 1
##    Data: new_tall_ratings[new_tall_ratings$Rubric == "InterpRes", ]
##
## REML criterion at convergence: 199.7
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.5317 -0.7627  0.2635  0.6614  2.6535
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  Artifact (Intercept) 0.06224  0.2495
##  Residual             0.25250  0.5025
## Number of obs: 116, groups:  Artifact, 90
##
## Fixed effects:
##                   Estimate Std. Error t value
## as.factor(Rater)1  2.70421    0.08912   30.34
## as.factor(Rater)2  2.58574    0.08912   29.01
## as.factor(Rater)3  2.13918    0.09027   23.70
##
## Correlation of Fixed Effects:
##             a.(R)1 a.(R)2
## as.fctr(R)2 0.061
## as.fctr(R)3 0.062  0.062
```

Next we look at **SelMeth**. Looking at the t-values we can see that all variables matter, and the ANOVA test shows that the model with the variable *Rater* is better than the intercept-only model.

```
# SelMeth
fla <- formula(model.formula.alldata[["SelMeth"]])
```

```
tmp <- lmer(fla,data=new_tall_ratings[new_tall_ratings$Rubric=="SelMeth",])
round(summary(tmp)$coef,2)  ## fixed effects and their t-values
```

```
##                    Estimate Std. Error t value
## as.factor(Rater)1      2.25       0.08   29.99
## as.factor(Rater)2      2.23       0.07   29.99
## as.factor(Rater)3      2.03       0.08   27.03
## SemesterS19           -0.36       0.10   -3.66
```

```
# SelMeth
tmp.single_intercept <- update(tmp, . ~ . + 1 - as.factor(Rater))
anova(tmp.single_intercept,tmp)
```

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

```
## Data: new_tall_ratings[new_tall_ratings$Rubric == "SelMeth", ]
## Models:
## tmp.single_intercept: as.numeric(Rating) ~ Semester + (1 | Artifact)
## tmp: as.numeric(Rating) ~ as.factor(Rater) + Semester + (1 | Artifact) - 1
##                      npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
## tmp.single_intercept    4 145.07 156.08 -68.534   137.07
## tmp                     6 142.05 158.58 -65.027   130.05 7.0146  2    0.02998 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

SelMeth also has an additional fixed effect *Semester* along with *Rater*. We will now check for fixed-effect interactions between these two variables. Below we also check the ANOVA test to see if adding the interaction term is better than the previous tmp model.

```
# Adding interaction between Rater and Semester
tmp.fixed_interactions <- update(tmp, . ~ . + as.factor(Rater)*Semester - Semester)
```

Below we can see that the p value is not small enough, suggesting that fixed-effect interactions are not needed.

```
anova(tmp, tmp.fixed_interactions)
```

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

```
## Data: new_tall_ratings[new_tall_ratings$Rubric == "SelMeth", ]
## Models:
## tmp: as.numeric(Rating) ~ as.factor(Rater) + Semester + (1 | Artifact) - 1
## tmp.fixed_interactions: as.numeric(Rating) ~ as.factor(Rater) + (1 | Artifact) + as.factor(Rater):Se
##                        npar    AIC    BIC  logLik deviance Chisq Df Pr(>Chisq)
## tmp                       6 142.05 158.58 -65.027   130.05
## tmp.fixed_interactions    8 143.46 165.49 -63.731   127.46 2.592  2     0.2736
```

Lastly, we will check for random effects. Since our model for *SelMeth* has two fixed effects, we will check the random effects for the two variables *Rater* and *Semester*.

```
# First check for (Semester | Artifact)
m0 <- tmp
mA <- update(m0, . ~ . + (Semester | Artifact))
```

```
## Error: number of observations (=116) <= number of random effects (=180) for term (Semester | Artifac
```

```
# Next, check for (Rater | Artifact)
m0 <- tmp
mA <- update(m0, . ~ . + (as.factor(Rater)|Artifact))
```

```
## Error: number of observations (=116) <= number of random effects (=270) for term (as.factor(Rater) |
```

We can see that the above tests are not possible because there are more random effects than there are observations in the dataset. This means that lmer() cannot fit a model. Since no testing is needed for these random effects, we will not be adding any. Thus, the final model for *SelMeth* is produced below.

```
SelMeth_lmer_final <- tmp
summary(SelMeth_lmer_final)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: as.numeric(Rating) ~ as.factor(Rater) + Semester + (1 | Artifact) -
##     1
##    Data: new_tall_ratings[new_tall_ratings$Rubric == "SelMeth", ]
##
## REML criterion at convergence: 143.6
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.0480 -0.3923 -0.0551  0.2674  2.5827
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  Artifact (Intercept) 0.08973  0.2996
##  Residual             0.10842  0.3293
## Number of obs: 116, groups:  Artifact, 90
##
## Fixed effects:
##                    Estimate Std. Error t value
## as.factor(Rater)1  2.25037    0.07503  29.992
## as.factor(Rater)2  2.22653    0.07424  29.991
## as.factor(Rater)3  2.03316    0.07521  27.033
## SemesterS19       -0.35860    0.09796  -3.661
##
## Correlation of Fixed Effects:
##             a.(R)1 a.(R)2 a.(R)3
## as.fctr(R)2  0.285
## as.fctr(R)3  0.287  0.280
## SemesterS19 -0.413 -0.391 -0.394
```

Finally, we look at **VisOrg**.

```
# VisOrg
fla <- formula(model.formula.alldata[["VisOrg"]])
tmp <- lmer(fla,data=new_tall_ratings[new_tall_ratings$Rubric=="VisOrg",])
round(summary(tmp)$coef,2)  ## fixed effects and their t-values
```

```
##                   Estimate Std. Error t value
## as.factor(Rater)1     2.37        0.1   24.87
## as.factor(Rater)2     2.65        0.1   27.78
## as.factor(Rater)3     2.28        0.1   23.70
```

```
# VisOrg
tmp.single_intercept <- update(tmp, . ~ . + 1 - as.factor(Rater))
anova(tmp.single_intercept,tmp)
```

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

```
## Data: new_tall_ratings[new_tall_ratings$Rubric == "VisOrg", ]
## Models:
## tmp.single_intercept: as.numeric(Rating) ~ (1 | Artifact)
## tmp: as.numeric(Rating) ~ as.factor(Rater) + (1 | Artifact) - 1
##                      npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
## tmp.single_intercept    3 228.69 236.95 -111.34   222.69
## tmp                     5 222.13 235.90 -106.06   212.13 10.558  2   0.005097
##
## tmp.single_intercept
## tmp                  **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Next we check for random effects. We will specifically look into adding (Rater | Artifact). Since, there are more random effect parameters than there are observations in the dataset, the model is not even possible (as shown in the error below). Thus, we will stick with the previous model.

```
# VisOrg - Random effect: check for (Rater | Artifact)
m0 <- tmp
mA <- update(m0, . ~ . + (as.factor(Rater)|Artifact))
```

```
## Error: number of observations (=116) <= number of random effects (=270) for term (as.factor(Rater) |
```

Below is the final model summary for *VisOrg*.

```
VisOrg_lmer_final <- tmp
summary(VisOrg_lmer_final)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: as.numeric(Rating) ~ as.factor(Rater) + (1 | Artifact) - 1
##    Data: new_tall_ratings[new_tall_ratings$Rubric == "VisOrg", ]
##
## REML criterion at convergence: 220.9
```

```
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -1.4911 -0.3307 -0.2475  0.3837  1.8693
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  Artifact (Intercept) 0.2883   0.5369
##  Residual             0.1463   0.3824
## Number of obs: 116, groups:  Artifact, 90
##
## Fixed effects:
##                   Estimate Std. Error t value
## as.factor(Rater)1  2.37016    0.09530   24.87
## as.factor(Rater)2  2.64690    0.09530   27.78
## as.factor(Rater)3  2.28122    0.09624   23.70
##
## Correlation of Fixed Effects:
##             a.(R)1 a.(R)2
## as.fctr(R)2 0.260
## as.fctr(R)3 0.261  0.261
```

Next I will look at the ICC for the above models. Although the magnitude changed a little bit, the general trend of ICC for each model did not change a lot.

```
icc_values <- c(icc(RsrchQ_lmer_final)$ICC_adjusted, icc(CritDes_lmer_final)$ICC_adjusted,
                icc(InitEDA_lmer_final)$ICC_adjusted, icc(SelMeth_lmer_final)$ICC_adjusted,
                icc(InterpRes_lmer_final)$ICC_adjusted, icc(VisOrg_lmer_final)$ICC_adjusted,
                icc(TxtOrg_lmer_final)$ICC_adjusted)

rubric_names <- c("RsrchQ", "CritDes", "InitEDA", "SelMeth", "InterpRes",
               "VisOrg", "TxtOrg")

icc_df <- data.frame(rubric_names, icc_values)
icc_df
```

```
##   rubric_names icc_values
## 1       RsrchQ  0.2072956
## 2      CritDes  0.6426639
## 3      InitEDA  0.6880645
## 4      SelMeth  0.4528468
## 5    InterpRes  0.1977433
## 6       VisOrg  0.6634323
## 7       TxtOrg  0.1914696
```

3.c - Fixed, random effects and interactions for "Combined" model

Now, instead of dividing the models into 7 different rubrics, I will use a single general model that can similarly explain the 7 rubric using random effects, without having to fit 7 separate models.

Below is the “combined” intercept-only model. We can see in the “Random effects” section that a lot of the random effects are highly correlated with each other. This is not surprising because we would expect that if a student is good at one or two of these rubrics, he or she is likely to be good at the other rubrics as well.

```
comb_lmer0 <- lmer(as.numeric(Rating) ~ 1 + (0 + Rubric | Artifact), data = new_tall_ratings, REML=F)
summary(comb_lmer0)
```

```
## Linear mixed model fit by maximum likelihood  ['lmerMod']
## Formula: as.numeric(Rating) ~ 1 + (0 + Rubric | Artifact)
##    Data: new_tall_ratings
## 
##      AIC      BIC   logLik deviance df.resid 
##   1530.6   1671.6   -735.3   1470.6      782 
## 
## Scaled residuals: 
##     Min      1Q  Median      3Q     Max 
## -3.0169 -0.5005 -0.0842  0.5281  3.7869 
## 
## Random effects:
##  Groups   Name             Variance Std.Dev. Corr                         
##  Artifact RubricCritDes    0.64643  0.8040                                
##           RubricInitEDA    0.38228  0.6183   0.26                         
##           RubricInterpRes  0.25505  0.5050   0.00 0.79                    
##           RubricRsrchQ     0.17316  0.4161   0.38 0.50 0.74               
##           RubricSelMeth    0.09518  0.3085   0.56 0.36 0.40 0.25          
##           RubricTxtOrg     0.40307  0.6349   0.01 0.68 0.80 0.64 0.23     
##           RubricVisOrg     0.31571  0.5619   0.17 0.78 0.76 0.60 0.27 0.80
##  Residual                  0.19442  0.4409                                
## Number of obs: 812, groups:  Artifact, 90
## 
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)   2.2314     0.0399   55.93
```

Now that we’ve explored the intercept-only model, we will fit a “full” model with all potential fixed effects, and then perform variable selection.

```
comb_lmer_full <- update(comb_lmer0, . ~ . + as.factor(Rater) + Semester +
                      Sex + Repeated + Rubric)
summary(comb_lmer_full)
```

```
## Linear mixed model fit by maximum likelihood  ['lmerMod']
## Formula: as.numeric(Rating) ~ (0 + Rubric | Artifact) + as.factor(Rater) +  
##     Semester + Sex + Repeated + Rubric
##    Data: new_tall_ratings
## 
##      AIC      BIC   logLik deviance df.resid 
##   1470.6   1663.3   -694.3   1388.6      771 
## 
## Scaled residuals: 
##     Min      1Q  Median      3Q     Max 
## -3.1218 -0.5213 -0.0265  0.5394  3.7747 
## 
```

```
## Random effects:
##  Groups   Name            Variance Std.Dev. Corr
##  Artifact RubricCritDes   0.53918  0.7343
##           RubricInitEDA   0.34143  0.5843   0.46
##           RubricInterpRes 0.16482  0.4060   0.23 0.76
##           RubricRsrchQ    0.16022  0.4003   0.59 0.43 0.71
##           RubricSelMeth   0.06128  0.2475   0.38 0.60 0.74 0.39
##           RubricTxtOrg    0.24936  0.4994   0.33 0.61 0.74 0.54 0.66
##           RubricVisOrg    0.24657  0.4966   0.34 0.74 0.67 0.52 0.38 0.78
##  Residual                 0.18951  0.4353
## Number of obs: 812, groups:  Artifact, 90
##
## Fixed effects:
##                    Estimate Std. Error t value
## (Intercept)        2.007112   0.107328  18.701
## as.factor(Rater)2  0.002207   0.054538   0.040
## as.factor(Rater)3 -0.176510   0.054702  -3.227
## SemesterS19       -0.176114   0.085525  -2.059
## SexM               0.009843   0.079099   0.124
## Repeated          -0.072668   0.095408  -0.762
## RubricInitEDA      0.555104   0.094704   5.861
## RubricInterpRes    0.593814   0.099485   5.969
## RubricRsrchQ       0.469026   0.086463   5.425
## RubricSelMeth      0.172454   0.093520   1.844
## RubricTxtOrg       0.701007   0.098477   7.118
## RubricVisOrg       0.534552   0.097936   5.458
##
## Correlation of Fixed Effects:
##             (Intr) a.(R)2 a.(R)3 SmsS19 SexM   Repetd RbIEDA RbrcIR RbrcRQ
## as.fctr(R)2 -0.247
## as.fctr(R)3 -0.240  0.499
## SemesterS19 -0.358  0.008  0.000
## SexM        -0.393 -0.027 -0.036  0.301
## Repeated    -0.152  0.001 -0.003  0.079  0.009
## RubrcIntEDA -0.553  0.000  0.000  0.000  0.000  0.009
## RbrcIntrpRs -0.663  0.000  0.000  0.000  0.000 -0.010  0.732
## RubrcRsrchQ -0.629  0.000  0.000  0.000  0.000 -0.040  0.580  0.754
## RubricSlMth -0.692  0.000  0.000  0.000  0.000 -0.091  0.655  0.772  0.687
## RubrcTxtOrg -0.615  0.000  0.000  0.000  0.000  0.005  0.670  0.759  0.675
## RubricVsOrg -0.611  0.000  0.000  0.000  0.000 -0.022  0.717  0.742  0.669
##             RbrcSM RbrcTO
## as.fctr(R)2
## as.fctr(R)3
## SemesterS19
## SexM
## Repeated
## RubrcIntEDA
## RbrcIntrpRs
## RubrcRsrchQ
## RubricSlMth
## RubrcTxtOrg  0.723
## RubricVsOrg  0.677  0.757
## optimizer (nloptwrap) convergence code: 0 (OK)
## unable to evaluate scaled gradient
## Model failed to converge: degenerate  Hessian with 1 negative eigenvalues
```

Below we will attempt variable selection on this full model by using “fitLMER.fnc” backward elimination methodology.

```
# Backward elimination on full model
comb_lmer1 <- fitLMER.fnc(comb_lmer_full, log.file.name = FALSE)
```

```
summary(comb_lmer1)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: as.numeric(Rating) ~ (0 + Rubric | Artifact) + as.factor(Rater) +
##     Semester + Rubric
##    Data: new_tall_ratings
##
## REML criterion at convergence: 1427.1
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.1192 -0.5090 -0.0206  0.5289  3.7748
##
## Random effects:
##  Groups   Name            Variance Std.Dev. Corr
##  Artifact RubricCritDes   0.55580  0.7455
##           RubricInitEDA   0.35086  0.5923   0.47
##           RubricInterpRes 0.16859  0.4106   0.24 0.75
##           RubricRsrchQ    0.16819  0.4101   0.59 0.44 0.71
##           RubricSelMeth   0.06505  0.2550   0.40 0.60 0.74 0.40
##           RubricTxtOrg    0.25563  0.5056   0.33 0.61 0.69 0.55 0.65
##           RubricVisOrg    0.25814  0.5081   0.35 0.74 0.68 0.52 0.40 0.76
##  Residual                 0.18938  0.4352
## Number of obs: 812, groups:  Artifact, 90
##
## Fixed effects:
##                     Estimate Std. Error t value
## (Intercept)        2.0017334  0.0985037  20.321
## as.factor(Rater)2  0.0001476  0.0547445   0.003
## as.factor(Rater)3 -0.1770527  0.0548906  -3.226
## SemesterS19       -0.1744404  0.0826698  -2.110
## RubricInitEDA      0.5546286  0.0951161   5.831
## RubricInterpRes    0.5936415  0.1002235   5.923
## RubricRsrchQ       0.4654696  0.0868573   5.359
## RubricSelMeth      0.1658900  0.0934817   1.775
## RubricTxtOrg       0.7002529  0.0995799   7.032
## RubricVisOrg       0.5333013  0.0985461   5.412
##
## Correlation of Fixed Effects:
##             (Intr) a.(R)2 a.(R)3 SmsS19 RbIEDA RbrcIR RbrcRQ RbrcSM RbrcTO
## as.fctr(R)2 -0.282
## as.fctr(R)3 -0.278  0.499
## SemesterS19 -0.266  0.016  0.011
## RubrcIntEDA -0.607  0.000  0.000 -0.001
## RbrcIntrpRs -0.733  0.000  0.000  0.001  0.731
## RubrcRsrchQ -0.699  0.000  0.000  0.003  0.580  0.753
## RubricSlMth -0.780  0.000  0.000  0.007  0.659  0.777  0.685
## RubrcTxtOrg -0.679  0.000  0.000  0.000  0.672  0.750  0.681  0.727
## RubricVsOrg -0.675  0.000  0.000  0.002  0.716  0.743  0.667  0.679  0.753
## optimizer (nloptwrap) convergence code: 0 (OK)
## Model failed to converge with max|grad| = 0.00203796 (tol = 0.002, component 1)
```

Now we can proceed to try interactions between the 3 variables *Rater*, *Semester*, and *Rubric* that was chosen as fixed effects. Since using the normal update function doesn't make the model converge, we will try switching optimizers and increasing the number of iterations allowed (code snipped from HW10 solutions used).

```
comb_inter_temp <- update(comb_lmer1, . ~ . + as.factor(Rater)*Semester*Rubric)
```

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

```
ss <- getME(comb_inter_temp,c("theta","fixef"))
comb_inter1<- update(comb_inter_temp,start=ss,
             control=lmerControl(optimizer="bobyqa",
                                 optCtrl=list(maxfun=2e5)))
```

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

```
summary(comb_inter1)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: as.numeric(Rating) ~ (0 + Rubric | Artifact) + as.factor(Rater) +
##     Semester + Rubric + as.factor(Rater):Semester + as.factor(Rater):Rubric +
##     Semester:Rubric + as.factor(Rater):Semester:Rubric
##    Data: new_tall_ratings
## Control: lmerControl(optimizer = "bobyqa", optCtrl = list(maxfun = 2e+05))
##
## REML criterion at convergence: 1428.5
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.9360 -0.5099 -0.0703  0.5170  3.6286
##
## Random effects:
##  Groups   Name            Variance Std.Dev. Corr
##  Artifact RubricCritDes   0.49810  0.7058
##           RubricInitEDA   0.35155  0.5929   0.43
##           RubricInterpRes 0.14375  0.3791   0.35 0.81
##           RubricRsrchQ    0.16394  0.4049   0.66 0.43 0.73
##           RubricSelMeth   0.06212  0.2492   0.44 0.64 0.80 0.49
##           RubricTxtOrg    0.25041  0.5004   0.42 0.64 0.72 0.58 0.61
##           RubricVisOrg    0.25119  0.5012   0.36 0.73 0.69 0.57 0.34 0.77
##  Residual                 0.18957  0.4354
## Number of obs: 812, groups:  Artifact, 90
##
## Fixed effects:
##                                                 Estimate Std. Error t value
## (Intercept)                                    1.739926   0.137668  12.639
## as.factor(Rater)2                              0.302892   0.155879   1.943
## as.factor(Rater)3                              0.236844   0.156632   1.512
## SemesterS19                                   -0.143622   0.252082  -0.570
## RubricInitEDA                                  0.765635   0.165240   4.633
## RubricInterpRes                                0.978833   0.162097   6.039
## RubricRsrchQ                                   0.711332   0.147817   4.812
## RubricSelMeth                                  0.462419   0.156517   2.954
## RubricTxtOrg                                   1.007965   0.162372   6.208
## RubricVisOrg                                   0.647624   0.165535   3.912
## as.factor(Rater)2:SemesterS19                  0.199601   0.302291   0.660
## as.factor(Rater)3:SemesterS19                 -0.071622   0.302681  -0.237
## as.factor(Rater)2:RubricInitEDA               -0.324601   0.204356  -1.588
## as.factor(Rater)3:RubricInitEDA               -0.373980   0.205609  -1.819
## as.factor(Rater)2:RubricInterpRes             -0.471093   0.201075  -2.343
## as.factor(Rater)3:RubricInterpRes             -0.711446   0.202345  -3.516
## as.factor(Rater)2:RubricRsrchQ                -0.446800   0.189772  -2.354
## as.factor(Rater)3:RubricRsrchQ                -0.475141   0.191132  -2.486
## as.factor(Rater)2:RubricSelMeth               -0.301167   0.194700  -1.547
## as.factor(Rater)3:RubricSelMeth               -0.364665   0.195992  -1.861
## as.factor(Rater)2:RubricTxtOrg                -0.444667   0.202147  -2.200
## as.factor(Rater)3:RubricTxtOrg                -0.402624   0.203424  -1.979
## as.factor(Rater)2:RubricVisOrg                 0.008272   0.204545   0.040
## as.factor(Rater)3:RubricVisOrg                -0.288100   0.205800  -1.400
## SemesterS19:RubricInitEDA                     -0.036027   0.301273  -0.120
## SemesterS19:RubricInterpRes                    0.141600   0.295342   0.479
## SemesterS19:RubricRsrchQ                       0.145584   0.268388   0.542
## SemesterS19:RubricSelMeth                     -0.075913   0.284939  -0.266
## SemesterS19:RubricTxtOrg                       0.177067   0.295774   0.599
## SemesterS19:RubricVisOrg                       0.160964   0.301766   0.533
## as.factor(Rater)2:SemesterS19:RubricInitEDA    0.091093   0.389655   0.234
## as.factor(Rater)3:SemesterS19:RubricInitEDA    0.249582   0.390314   0.639
## as.factor(Rater)2:SemesterS19:RubricInterpRes -0.193826   0.382630  -0.507
## as.factor(Rater)3:SemesterS19:RubricInterpRes -0.152705   0.383300  -0.398
## as.factor(Rater)2:SemesterS19:RubricRsrchQ    -0.141984   0.357498  -0.397
## as.factor(Rater)3:SemesterS19:RubricRsrchQ     0.352169   0.358222   0.983
## as.factor(Rater)2:SemesterS19:RubricSelMeth   -0.329281   0.369327  -0.892
## as.factor(Rater)3:SemesterS19:RubricSelMeth   -0.195077   0.370010  -0.527
## as.factor(Rater)2:SemesterS19:RubricTxtOrg    -0.466039   0.384571  -1.212
## as.factor(Rater)3:SemesterS19:RubricTxtOrg    -0.318173   0.385244  -0.826
## as.factor(Rater)2:SemesterS19:RubricVisOrg    -0.532018   0.389980  -1.364
## as.factor(Rater)3:SemesterS19:RubricVisOrg    -0.188809   0.390640  -0.483
```

```
##
## Correlation matrix not shown by default, as p = 42 > 12.
## Use print(x, correlation=TRUE)  or
##     vcov(x)        if you need it
```

```
## optimizer (bobyqa) convergence code: 0 (OK)
## boundary (singular) fit: see ?isSingular
```

Next, we can attempt variable selection using “fitLMER.fnc”.

```
comb_inter1_red <- fitLMER.fnc(comb_inter1, log.file.name=FALSE)
```

```
summary(comb_inter1_red)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: as.numeric(Rating) ~ (0 + Rubric | Artifact) + as.factor(Rater) +
##     Semester + Rubric + as.factor(Rater):Rubric
##    Data: new_tall_ratings
## Control: lmerControl(optimizer = "bobyqa", optCtrl = list(maxfun = 2e+05))
##
## REML criterion at convergence: 1423.2
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.9860 -0.5155 -0.0448  0.4914  3.5503
##
## Random effects:
##  Groups   Name            Variance Std.Dev. Corr
##  Artifact RubricCritDes   0.50683  0.7119
##           RubricInitEDA   0.35334  0.5944   0.45
##           RubricInterpRes 0.14995  0.3872   0.37 0.82
##           RubricRsrchQ    0.17847  0.4225   0.64 0.44 0.73
##           RubricSelMeth   0.06639  0.2577   0.42 0.61 0.74 0.37
##           RubricTxtOrg    0.25658  0.5065   0.41 0.63 0.72 0.53 0.63
##           RubricVisOrg    0.25106  0.5011   0.35 0.72 0.69 0.52 0.38 0.79
##  Residual                 0.18674  0.4321
## Number of obs: 812, groups:  Artifact, 90
##
## Fixed effects:
##                                  Estimate Std. Error t value
## (Intercept)                       1.75474    0.11818  14.848
## as.factor(Rater)2                 0.35238    0.13297   2.650
## as.factor(Rater)3                 0.21484    0.13344   1.610
## SemesterS19                      -0.17923    0.08228  -2.178
## RubricInitEDA                     0.75155    0.13671   5.498
## RubricInterpRes                   1.01924    0.13455   7.575
## RubricRsrchQ                      0.75454    0.12436   6.068
## RubricSelMeth                     0.43231    0.13082   3.304
## RubricTxtOrg                      1.05188    0.13605   7.731
## RubricVisOrg                      0.68818    0.13854   4.967
## as.factor(Rater)2:RubricInitEDA  -0.29389    0.17233  -1.705
## as.factor(Rater)3:RubricInitEDA  -0.29638    0.17312  -1.712
## as.factor(Rater)2:RubricInterpRes -0.52429    0.16977  -3.088
## as.factor(Rater)3:RubricInterpRes -0.75373    0.17056  -4.419
## as.factor(Rater)2:RubricRsrchQ   -0.48636    0.16134  -3.014
## as.factor(Rater)3:RubricRsrchQ   -0.37279    0.16218  -2.299
## as.factor(Rater)2:RubricSelMeth  -0.38268    0.16485  -2.321
## as.factor(Rater)3:RubricSelMeth  -0.41482    0.16565  -2.504
## as.factor(Rater)2:RubricTxtOrg   -0.56561    0.17164  -3.295
## as.factor(Rater)3:RubricTxtOrg   -0.48397    0.17244  -2.807
## as.factor(Rater)2:RubricVisOrg   -0.13143    0.17367  -0.757
## as.factor(Rater)3:RubricVisOrg    -0.33590    0.17446  -1.925
```

```
##
## Correlation matrix not shown by default, as p = 22 > 12.
## Use print(x, correlation=TRUE)  or
##     vcov(x)        if you need it
```

```
## optimizer (bobyqa) convergence code: 0 (OK)
## boundary (singular) fit: see ?isSingular
```

Now that we have several models to choose from, let us compare the models.

```
# Model with ALL interactions
formula(comb_inter1)
```

```
## as.numeric(Rating) ~ (0 + Rubric | Artifact) + as.factor(Rater) +
##     Semester + Rubric + as.factor(Rater):Semester + as.factor(Rater):Rubric +
##     Semester:Rubric + as.factor(Rater):Semester:Rubric
```

```
# Model with REDUCED interactions
formula(comb_inter1_red)
```

```
## as.numeric(Rating) ~ (0 + Rubric | Artifact) + as.factor(Rater) +
##     Semester + Rubric + as.factor(Rater):Rubric
```

```
# Model with NO interactions
formula(comb_lmer1)
```

```
## as.numeric(Rating) ~ (0 + Rubric | Artifact) + as.factor(Rater) +
##     Semester + Rubric
```

Let us use ANOVA to compare the three models. Since the models are nested, we can use AIC, BIC or likelihood ratio tests to perform the comparison. We can see that BIC prefers the simpler model without ANY interaction terms. The likelihood ratio test and the AIC agree that the model with REDUCED interactions is the best.

```
# Model comparison using ANOVA
anova(comb_inter1, comb_inter1_red, comb_lmer1)
```

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

```
## Data: new_tall_ratings
## Models:
## comb_lmer1: as.numeric(Rating) ~ (0 + Rubric | Artifact) + as.factor(Rater) + Semester + Rubric
## comb_inter1_red: as.numeric(Rating) ~ (0 + Rubric | Artifact) + as.factor(Rater) + Semester + Rubric
## comb_inter1: as.numeric(Rating) ~ (0 + Rubric | Artifact) + as.factor(Rater) + Semester + Rubric + a
##                 npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
## comb_lmer1        39 1466.9 1650.2 -694.47   1388.9
## comb_inter1_red   51 1458.1 1697.8 -678.04   1356.1 32.852 12   0.001021 **
## comb_inter1       71 1475.4 1809.1 -666.72   1333.4 22.635 20   0.307053
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
## the models are nested so we can use AIC, BIC or likelihod ratio (deviance)
## tests... AIC and the LRT agree on comb.inter_elim; BIC likes the simpler
## comb.back_elim.

## Interestingly, comb.inter_elim adds a rater x rubric interaction to
## the main-effects model comb.back_elim.  This suggests that the raters
## do not all use the rubrics in the same way.
```

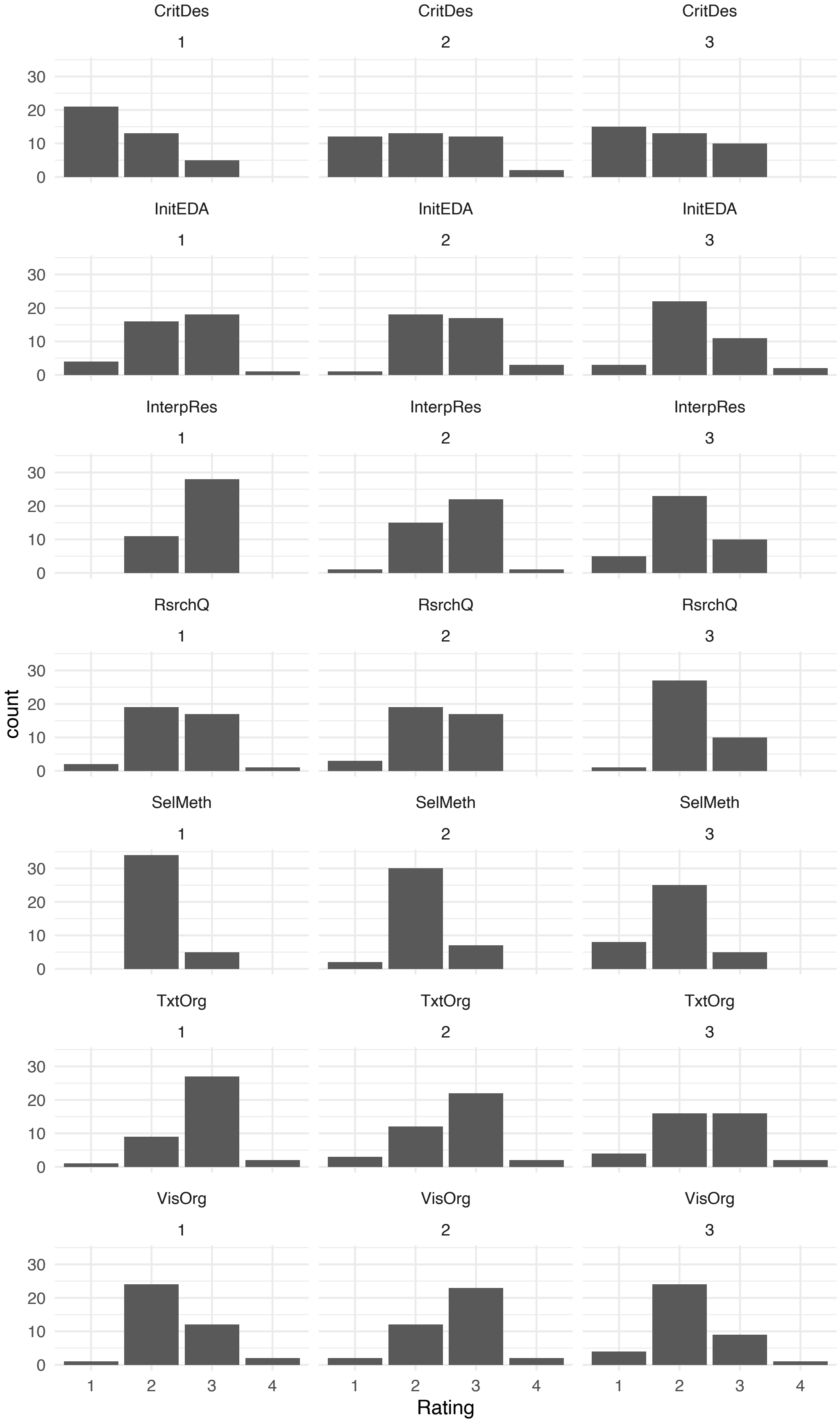
We will now look more specifically into the model that ANOVA chose, *comb_inter1_red*. First, let us take a quick look at the coefficients. Below, we notice that most of the interaction terms between *Rater* and *Rubric* have statistically significant coefficients with high absolute t values. This suggests that the raters do not all use the rubrics in the same manner. The coefficients tell us that there are some rubrics such as *InitEDA* or *RsrchQ* where the 3 raters seem to have little difference in grading using those rubrics. But for the others: * *CritDes*: Rater 1 tends to give the lower score compared to Raters 2 and 3 * *InterpRes*: Rater 3 tends to give the lower score compared to Raters 1 and 2 (-0.75 coefficient for interactions + 0.21 coefficient for Rater 3 = -0.54) * *SelMeth*: Rater 3 tends to give the lower score compared to Raters 1 and 2 (-0.41 coefficient for interactions + 0.21 coefficient for Rater 3 = -0.20) * *TxtOrg*: Rater 1 tends to give overall higher score compared to Raters 2 and 3 * *VisOrg*: Rater 2 tends to give overall higher score compared to Raters 1 and 3

```
summary(comb_inter1_red)$coef
```

```
##                                    Estimate Std. Error    t value
## (Intercept)                       1.7547407 0.11818295 14.8476635
## as.factor(Rater)2                 0.3523762 0.13296879  2.6500671
## as.factor(Rater)3                 0.2148360 0.13344457  1.6099269
## SemesterS19                      -0.1792281 0.08227611 -2.1783735
## RubricInitEDA                     0.7515513 0.13670569  5.4975858
## RubricInterpRes                   1.0192386 0.13455042  7.5751423
## RubricRsrchQ                      0.7545387 0.12435657  6.0675421
## RubricSelMeth                     0.4323094 0.13082471  3.3044937
## RubricTxtOrg                      1.0518766 0.13605171  7.7314471
## RubricVisOrg                      0.6881751 0.13854313  4.9672265
## as.factor(Rater)2:RubricInitEDA  -0.2938894 0.17232911 -1.7053960
## as.factor(Rater)3:RubricInitEDA  -0.2963821 0.17311802 -1.7120234
## as.factor(Rater)2:RubricInterpRes -0.5242899 0.16976541 -3.0883198
## as.factor(Rater)3:RubricInterpRes -0.7537294 0.17055832 -4.4191884
## as.factor(Rater)2:RubricRsrchQ   -0.4863568 0.16134235 -3.0144400
## as.factor(Rater)3:RubricRsrchQ   -0.3727938 0.16217826 -2.2986666
## as.factor(Rater)2:RubricSelMeth  -0.3826839 0.16485270 -2.3213685
## as.factor(Rater)3:RubricSelMeth  -0.4148157 0.16564805 -2.5041992
## as.factor(Rater)2:RubricTxtOrg   -0.5656066 0.17164454 -3.2952203
## as.factor(Rater)3:RubricTxtOrg   -0.4839733 0.17243702 -2.8066669
## as.factor(Rater)2:RubricVisOrg   -0.1314269 0.17367408 -0.7567444
## as.factor(Rater)3:RubricVisOrg   -0.3359030 0.17445641 -1.9254265
```

We can verify this by observing the facets plot for the Ratings given by the 3 raters throughout the different rubrics below. This does not mean that a certain rater is simply more harsh than the others, but it tells us that all the raters have different interpretations of grading across the different rubrics. This justifies that the best model to use here would be the reduced interactions model **comb_inter1_red**.

```
g <- ggplot(new_tall_ratings, aes(x=Rating)) +
  geom_bar() +
  facet_wrap( ~ Rubric + Rater, nrow=7) +
  theme_minimal()

g
```


CritDes
1
CritDes
2
CritDes
3
InitEDA
1
InitEDA
2
InitEDA
3
InterpRes
1
InterpRes
2
InterpRes
3
RsrchQ
1
RsrchQ
2
RsrchQ
3
SelMeth
1
SelMeth
2
SelMeth
3
TxtOrg
1
TxtOrg
2
TxtOrg
3
VisOrg
1
VisOrg
2
VisOrg
3
count
0
10
20
30
1
2
3
4
Rating

Lastly, we will consider adding random effects to our previous best model **comb_inter1_red**. Note that we want to add the random effects without having a random intercept, meaning that we would have to add a 0 in front of the random intercept term (to preserve the structure of the model). We will mainly be using ANOVA tests to inspect the AIC and BIC values for different models. Below are the random effects we can experiment with: * *as.factor(Rater)* * *Semester* * *as.factor(Rater):Rubric*

```
# We first try as.factor(Rater)
m0 <- comb_inter1_red
mA <- lmer(as.numeric(Rating) ~ (0 + Rubric | Artifact) +
             (0 + as.factor(Rater) | Artifact) + as.factor(Rater) +
             Semester + Rubric + as.factor(Rater):Rubric, data=new_tall_ratings)
```

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

Below in the anova results we can see that the AIC and BIC values for the alternative hypothesis (with Rater as random effect) are smaller and preferred.

```
# We first try as.factor(Rater)
anova(m0, mA)
```

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

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

```
## Data: new_tall_ratings
## Models:
## m0: as.numeric(Rating) ~ (0 + Rubric | Artifact) + as.factor(Rater) + Semester + Rubric + as.factor(
## mA: as.numeric(Rating) ~ (0 + Rubric | Artifact) + (0 + as.factor(Rater) | Artifact) + as.factor(Rate
##    npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
## m0   51 1458.1 1697.8 -678.04   1356.1
## mA   57 1419.1 1687.0 -652.55   1305.1 50.974  6  2.997e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Next we try *Semester*

```
# Next we try Semester
m0 <- comb_inter1_red
mA <- lmer(as.numeric(Rating) ~ (0 + Rubric | Artifact) +
             (0 + Semester | Artifact) + as.factor(Rater) +
             Semester + Rubric + as.factor(Rater):Rubric, data=new_tall_ratings)
```

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

It turns out that the AIC and BIC values do not like having *Semester* as a random effect.

```
# Next we try Semester
anova(m0, mA)
```

```
## refitting model(s) with ML (instead of REML)
## Data: new_tall_ratings
## Models:
## m0: as.numeric(Rating) ~ (0 + Rubric | Artifact) + as.factor(Rater) + Semester + Rubric + as.factor(
## mA: as.numeric(Rating) ~ (0 + Rubric | Artifact) + (0 + Semester | Artifact) + as.factor(Rater) + Se
##    npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
## m0   51 1458.1 1697.8 -678.04   1356.1
## mA   54 1461.5 1715.3 -676.75   1353.5 2.5802  3      0.461
```

Finally, we try the interaction term *as.factor(Rater):Rubric*. But there is an error that says there are not enough observations compared to the random effects in mA. Therefore, we will not move forward with this random effect.

```
# Finally, we try Rater:Rurbic
m0 <- comb_inter1_red
mA <- lmer(as.numeric(Rating) ~ (0 + Rubric | Artifact) +
             (0 + as.factor(Rater):Rubric | Artifact) + as.factor(Rater) +
             Semester + Rubric + as.factor(Rater):Rubric, data=new_tall_ratings)
```

```
## Error: number of observations (=812) <= number of random effects (=1890) for term (0 + as.factor(Rat
```

Final Model

The final model turned out to be **comb_inter1_red** with an added random effect for Raters.

```
comb_final <- lmer(as.numeric(Rating) ~ (0 + Rubric | Artifact) +
                     (0 + as.factor(Rater) | Artifact) + as.factor(Rater) +
                     Semester + Rubric + as.factor(Rater):Rubric, data=new_tall_ratings)
```

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

```
formula(comb_final)
```

```
## as.numeric(Rating) ~ (0 + Rubric | Artifact) + (0 + as.factor(Rater) |
##     Artifact) + as.factor(Rater) + Semester + Rubric + as.factor(Rater):Rubric
```

Below is the summary of our final model:

```
summary(comb_final)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula:
## as.numeric(Rating) ~ (0 + Rubric | Artifact) + (0 + as.factor(Rater) |
##     Artifact) + as.factor(Rater) + Semester + Rubric + as.factor(Rater):Rubric
##    Data: new_tall_ratings
##
## REML criterion at convergence: 1373.9
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -3.06640 -0.47423 -0.02938  0.45849  2.74420
##
```

```
## Random effects:
##  Groups     Name              Variance Std.Dev. Corr
##  Artifact   RubricCritDes     0.50049  0.7075
##             RubricInitEDA     0.31798  0.5639    0.33
##             RubricInterpRes   0.10199  0.3194    0.16  0.67
##             RubricRsrchQ      0.17917  0.4233    0.50  0.19  0.54
##             RubricSelMeth     0.03825  0.1956    0.14  0.23  0.38 -0.24
##             RubricTxtOrg      0.25023  0.5002    0.25  0.44  0.37  0.31  0.21
##             RubricVisOrg      0.22935  0.4789    0.19  0.51  0.45  0.28 -0.16
##  Artifact.1 as.factor(Rater)1 0.01274  0.1129
##             as.factor(Rater)2 0.11180  0.3344   -0.49
##             as.factor(Rater)3 0.09415  0.3068    0.33  0.66
##  Residual                     0.13470  0.3670
##
##
##
##
##
##
##
##   0.54
##
##
##
##
## Number of obs: 812, groups:  Artifact, 90
##
## Fixed effects:
##                                   Estimate Std. Error t value
## (Intercept)                        1.75277    0.11421  15.347
## as.factor(Rater)2                  0.35347    0.13899   2.543
## as.factor(Rater)3                  0.19542    0.12980   1.506
## SemesterS19                       -0.15969    0.07646  -2.089
## RubricInitEDA                      0.74448    0.12968   5.741
## RubricInterpRes                    0.99766    0.12738   7.832
## RubricRsrchQ                       0.73033    0.11787   6.196
## RubricSelMeth                      0.41536    0.12492   3.325
## RubricTxtOrg                       1.01986    0.13082   7.796
## RubricVisOrg                       0.66026    0.13241   4.986
## as.factor(Rater)2:RubricInitEDA   -0.28692    0.15561  -1.844
## as.factor(Rater)3:RubricInitEDA   -0.29496    0.15621  -1.888
## as.factor(Rater)2:RubricInterpRes -0.50217    0.15303  -3.282
## as.factor(Rater)3:RubricInterpRes -0.71542    0.15345  -4.662
## as.factor(Rater)2:RubricRsrchQ    -0.47296    0.14676  -3.223
## as.factor(Rater)3:RubricRsrchQ    -0.32238    0.14725  -2.189
## as.factor(Rater)2:RubricSelMeth   -0.37345    0.15015  -2.487
## as.factor(Rater)3:RubricSelMeth   -0.38687    0.14977  -2.583
## as.factor(Rater)2:RubricTxtOrg    -0.53720    0.15677  -3.427
## as.factor(Rater)3:RubricTxtOrg    -0.44324    0.15735  -2.817
## as.factor(Rater)2:RubricVisOrg    -0.09389    0.15765  -0.596
## as.factor(Rater)3:RubricVisOrg    -0.27676    0.15816  -1.750

##
## Correlation matrix not shown by default, as p = 22 > 12.
## Use print(x, correlation=TRUE)  or
##     vcov(x)        if you need it

## optimizer (nloptwrap) convergence code: 0 (OK)
## boundary (singular) fit: see ?isSingular
```

Research Question 4

Most of the models seem to agree that *RsrchQ*, *InterpRes* and *TxtOrg* give the lowest ICC value. This means that the raters tend to disagree in their ratings for these 3 rubrics the most. Having more data to work on could solidify this statement and give more reliable results in the degree of agreement between different raters. But the final model that we constructed in research question 3 seems to explain most of the important information, where we left *Rubric* and *Rater* as random effects across *Artifacts*, and having interaction terms better explain the *Rating* variation in the data.

One interesting question that was not raised was the effect of the variable *Sex* on *Rating*. Below we can filter out the Artifacts from male and female students separately to compare the summary statistics. But it turns out that the mean and median Ratings are almost identical for the two filtered datasets. This suggests that there are no apparent Rating differences between the gender of which the artifact was written by.

```
male_ratings_tall <- tall_ratings %>%
  filter(Sex == 'M')

female_ratings_tall <- tall_ratings %>%
  filter(Sex == 'F')

summary(male_ratings_tall$Rating)
```

```
##   1   2   3   4
##  42 177 135  10
```

```
summary(female_ratings_tall$Rating)
```

```
##   1   2   3   4
##  51 218 168  11
```

Further analysis could be made in the future to explore the effect of the *Semester* on the ratings and how they vary across different Rubrics. Or even even explore whether *Raters* grade artifacts differently across different semesters.