Evaluating the Success of the New Dietrich College General Education Program

Megan Christy
Department of Statistics and Data Science, Carnegie Mellon University
mechrist@andrew.cmu.edu
10 December 2021

Abstract

We aim to understand the factors of the experiment that are related to ratings on student artifacts in order to evaluate the success of the rating system for the new Dietrich College General Education program. We examine data on 91 artifacts produced by students that were rated by three raters from three different departments. We use plots, ICCs, and percent exact agreements to examine differences in ratings between different rubrics and raters, and we use mixed effect models to determine what factors from the experiment are related to ratings. We find that rubric, rater, semester, and the interaction of rater and rubric are related to ratings. Dietrich College should aim to understand how these factors impact ratings before using them to evaluate the success of the new General Education program to ensure that ratings are fair and effective.

1 Introduction

General Education programs give students the opportunity to be introduced to a variety of disciplines and learn ideas and skills they may otherwise never be exposed to. Dietrich College at Carnegie Mellon University is in the process of implementing a new General Education program and aims to determine whether the program is successful so that they may provide the best education possible for their students. The college has been experimenting with rating work in the Freshmen Statistics course to evaluate the success of the new program. We aim to address the following four research questions regarding the recent experiment:

- Is the distribution of ratings for each rubric pretty much indistinguishable from the other rubrics, or are there rubrics that tend to get especially high or low ratings? Is the distribution of ratings given by each rater pretty much indistinguishable from the other raters, or are there raters that tend to give especially high or low ratings?
- For each rubric, do the raters generally agree on their scores? If not, is there one rater who disagrees with the others? Or do they all disagree?

- More generally, how are the various factors in this experiement (Rater, Semester, Sex, Repeated, Rubric) related to the ratings? Do the factors interact in any interesting ways?
- Is there anything else interesting to say about this data?

2 Data

The data comes from a Dietrich College experiment in which raters from across the college rated project papers, which will now be referred to as "artifacts", from the Freshmen Statistics course. 91 artifacts were randomly sampled from a Fall and Spring section of the course, and three raters from three different departments rated the artifacts on seven rubrics that are summarized below 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 answers 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 appro-priate 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: Table of rubric names and descriptions for rating Freshmen Statistics artifacts

The rating scale for all seven rubrics is shown below 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: Table of rating scale and meanings used for rating Freshmen Statistics artifacts

The raters were blind to which class and which students completed the artifacts they rated. In addition, thirteen of the artifacts were rated by all three raters while the remaining 78 were only rated by one rater. The variables available from the experiment are captured in Table 3 below.

Variable Name	Values	Description
X	1, 2, 3...	Row number
Rater	1, 2, or 3	Which rater 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 the student who produced 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: Table of variable names, values, and descriptions available from the experiment

We obtained the data by downloading the files ratings.csv and tall.csv. The files contain the same information in different formats. The ratings.csv file is formatted in the same way that the information in Table 3 is displayed. The tall.csv file is structured so each row contains a single rating in a column labelled Rating and the rubric for the rating in a column labelled Rubric. See Technical Appendix, Page 2 to view the first few rows of the data in both the ratings.csv and tall.csv file.

To get an exploratory look at the data, we examine some tables of counts of each of the variables that are important to our analyses. Each rater rated 39 artifacts, and slightly more artifacts that were rated came from the Fall semester. In addition, slightly more of the artifacts were created by females than males, and 39 of the observations in the data represent one of the thirteen artifacts seen by all three raters. See Technical Appendix, Pages 3-4 for the tables of counts for these variables. Table 4 below contains the counts of the ratings for each of the seven rubrics.

	1	2	3	4	NA's
CritDes	47	39	28	2	1
InitEDA	8	56	47	6	0
InterpRes	6	49	61	1	0
RsrchQ	6	65	45	1	0
SelMeth	10	89	18	0	0
TxtOrg	8	37	66	6	0
VisOrg	7	59	45	5	1

Table 4: Table of counts of ratings for each rubric

Looking at the tables of counts for the ratings for each rubric, we see that for almost all rubrics, the majority of the ratings are 2 or 3. The exception to this is the CritDes rubric, which has a majority rating of 1. We also notice that the SelMeth rubric is the only rubric with no ratings of 4, and the CritDes and TxtOrg rubrics both feature a missing observation. The distribution of ratings given rubric and how the missing data is handled will be further addressed in the Methods and Results sections.

3 Methods

Before describing the methods used to address each research question, we will explain how we handled the missing data. There is a missing value of Sex, a missing value of CritDes, and a missing value of VisOrg. When we model with Rating as the response variable, the two observations with missing data are automatically dropped. Since we must use the exact same data in order to appropriately compare models, we decided to delete the observations with missing data when we began modelling. It is worth noting that when we are using the reduced dataset that includes only artifacts that were rated by all three raters, there are no missing values, so we do not have to worry about any of these issues when using this data.

3.1.1 Examining and Comparing the Distribution of Ratings for Each Rubric

To address the first question in the first bullet point in the Introduction, we constructed bar plots and produced counts of the ratings for each of the seven rubrics using the full dataset. Then, we made the same bar plots and counts for the data set with just the artifacts that all three raters saw. See Technical Appendix, Pages 4-7.

3.1.2 Examining and Comparing the Distribution of Ratings for Each Rater

To address the second question in the first bullet point in the Introduction, we constructed bar plots and produced counts of the ratings for each of the three raters using the full dataset. Then, we made the same bar plots and counts for the data set with just the artifacts that all three raters saw. See Technical Appendix, Pages 7-9.

3.2 Examining Whether Raters Generally Agree on Their Scores

To answer the questions in the second bullet point in the Introduction, we first fitted seven random-intercept models, one for each rubric, and calculated the ICCs for each using the subset of the data for just the 13 artifacts seen by all three raters. In these models, Rating is the response variable, and the random intercept is grouped by Artifact. Next, we computed the percent exact agreement between each pair of raters for each rubric to determine who is agreeing/disagreeing with whom on each rubric. The percent exact agreements were calculated by making a 2-way table of counts for the ratings of each pair of raters, on each rubric, and then calculating the percentage of observations on the main diagonal of the table. Finally, we redid the seven ICC calculations using the full data set to see if they agree with the ICCs we calculated for the repeated only data set. See Technical Appendix, Pages 10-17.

3.3 Looking into how the Various Factors are Related to Ratings and if the Factors Interact

First, we tried adding fixed effects to the seven rubric-specific random intercept models using just the data from the artifacts that all three raters saw. We did this by finding a model using backwards elimination and comparing it to the intercept-only model using a likelihood ratio test for each rubric. After finding fixed effects to add to the model, we planned to try adding interactions and random effects. See Technical Appendix, Pages 18-24.

Next, we followed the same process for the seven rubric-specific random intercept models using the full data set. Before modelling, we deleted any observations with missing data to ensure that all models were fit and compared using the exact same data. Again, we first tried finding a model using backwards elimination and comparing it to the intercept-only model using a likelihood ratio test for each rubric. For the rubrics where we found that adding fixed effects improves the fit of the model, we checked the t-statistics of the fixed effects to make sure they made sense, then tried adding interactions and new random effects. We tried interactions when there were two or more fixed effects added, and we tried random effects for any fixed effect we added. See Technical Appendix, Pages 24-48.

Finally, we tried modelling in a way that would allow us to explore interactions with Rubric directly. We tried adding fixed effects, interactions, and new random effects to a "combined" model with Rubric as a random effect grouped by Artifact. We started with the intercept-only model, then tried adding fixed effects using backward elimination. Next, we tried adding interactions based on the fixed effects we found, and we tried using different optimizers when the model failed to converge. We decided on a model by comparing AIC, BIC , and p-values from likelihood ratio tests. Based on the model we chose, we tried adding random effects, and again used AIC, BIC, and p-values from likelihood ratio tests to compare models. See Technical Appendix, Pages 48-56.

3.4 Finding Other Interesting Things About This Data

To take a deeper look into the differences in the models when fitted with just the data with the 13 repeated artifacts and the models when fitted with all data, we constructed bar plots of Rating faceted by Rater for each of the two fits for the rubrics that included fixed effects when fit using the full data set. See Technical Appendix, Pages 64-74.

We also constructed density plots of Ratings filled by Semester, Sex, and Repeated to see if we can visually support the decisions to include or not include these factors in the models. See Technical Appendix, Pages 56-72.

4 Results

4.1.1 Examining and Comparing the Distribution of Ratings for Each Rubric

We use bar plots (see Figure 1 below) and counts (see Technical Appendix, Page 5) to compare the distribution of ratings for each rubric using the full data set.

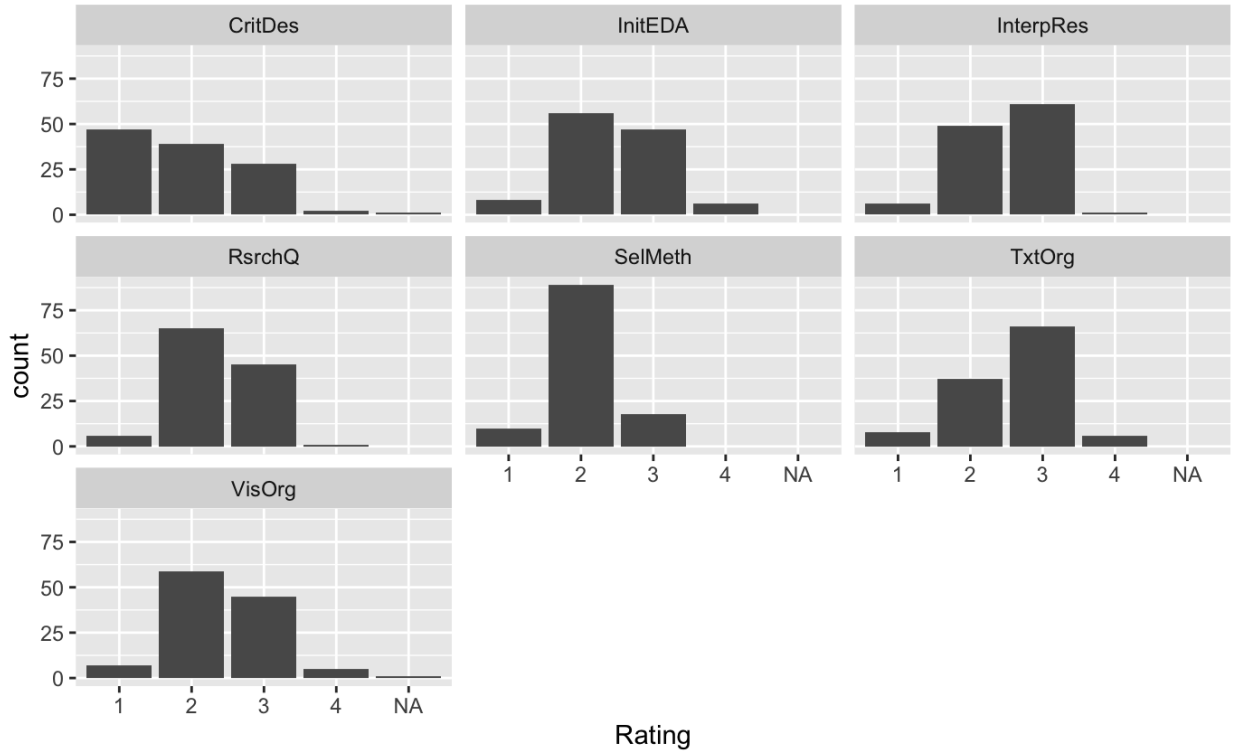


Figure 1: Bar plots of the distribution of Rating for each rubric

Looking at the bar plots and counts, we see that the distributions of ratings for the InitEDA, InterpRes, RsrchQ, TxtOrg, and VisOrg rubrics are generally similar, with most ratings being either 2 or 3. Out of these five rubrics, we see that most of the observations for the InterpRes and TxtOrg rubrics have a rating of 3 while the rest of the five have 2 as the most common rating, suggesting that these two rubrics tend to get slightly higher ratings than the rest of the five. The distribution of the CritDes rubric is different in that most of the ratings are 1, and the number of observations with each rating falls as rating increases. This suggests that this rubric tends to get especially low ratings. The SelMeth rubric is different in that the large majority of ratings are 2, and it is the only rubric with no ratings of 4. This suggests that this rubric may tend to get slightly lower ratings.

We make the same bar plots and table of counts (see Technical Appendix, Pages 6-7) for the dataset with just the artifacts that all three raters saw to see whether these artifacts are representative of the whole data set. The bar plots and table of counts show that the distributions of ratings for each rubric are largely similar to the distributions we saw when looking at the full dataset. The CritDes and SelMeth rubrics stand out for the same reasons as before, and InterpRes and TxtOrg still have more ratings of 3 than ratings of 2, which is different from the other three rubrics with similar distributions (InitEDA, RsrchQ, and VisOrg). This suggests that these thirteen artifacts are generally representative of the full dataset. One interesting difference to note, however, is that there are only two ratings of 4 in the entire reduced dataset, whereas all but

one of the rubrics in the full dataset had at least one rating of 4. This suggests that in some cases the ratings on these artifacts tended to be a bit lower when we just look at the data with repeated artifacts.

4.1.2 Examining and Comparing the Distribution of Ratings for Each Rater

To compare the distributions of ratings across raters, we construct bar plots (see Figure 2 below) and produce a table of counts (see Technical Appendix, Page 8) for each rater first using the full dataset.

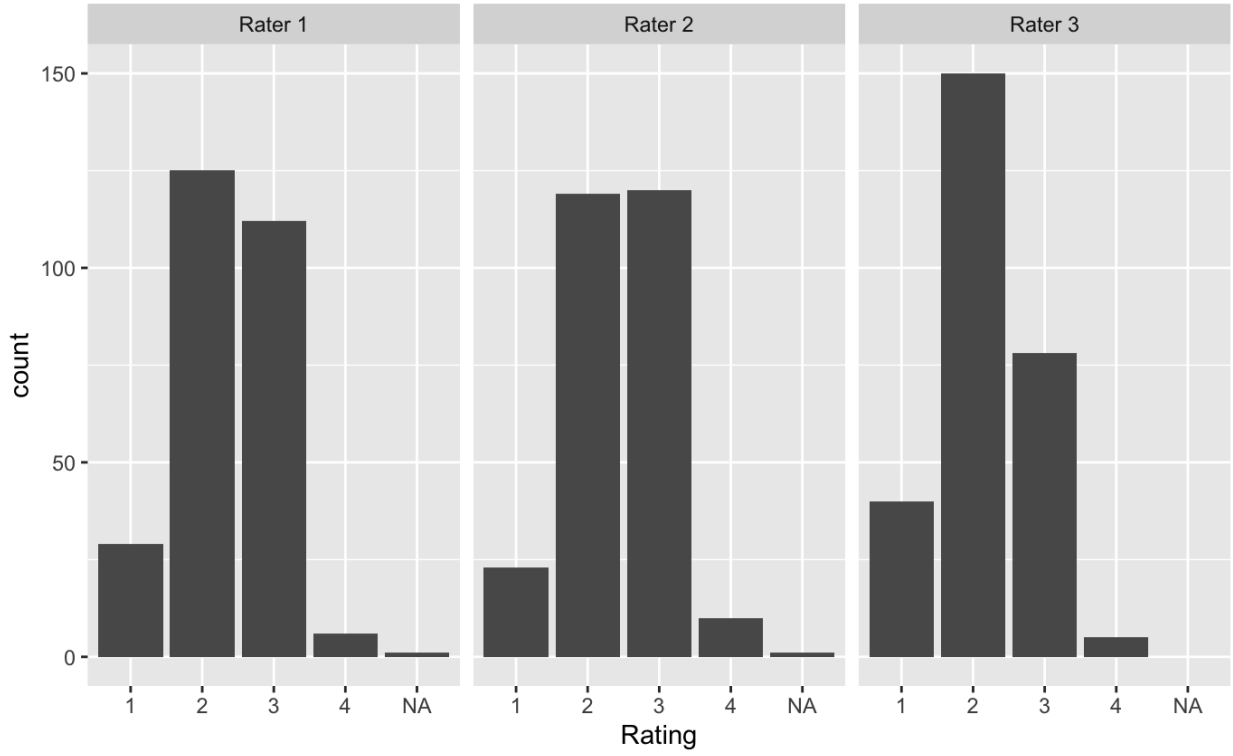


Figure 2: Bar plots of the distribution of Rating for each rater

The distribution of ratings for Raters 1 and 2 appear to be very similar, with most ratings being either 2 or 3. The majority of ratings for Rater 3 are 2, and Rater 3 gave more ratings of 1 than the other two. Thus, it seems that Rater 3 tends to give somewhat lower ratings.

We make the same bar plots and table of counts (see Technical Appendix, Page 9) for the dataset with just the artifacts that all three raters saw to see whether these artifacts are representative of the whole data set. The bar plots and table of counts show that the distributions of ratings for each rater are largely similar to the distributions we saw when looking at the full dataset. The distribution of ratings for Raters 1 and 2 appear to be similar, with most ratings being either 2 or 3. The same is true for Rater 3, but Rater 3 gives fewer ratings of 3 and no

ratings of 4. Thus, it seems that Rater 3 tends to give somewhat lower ratings, and this is similar to what we found when looking at the full dataset.

4.2 Examining Whether Raters Generally Agree on Their Scores

Table 5 below contains the ICCs and percent exact agreements for the data with only the repeated artifacts and the ICCs for the full data set.

	ICC.full	ICC.repeat	a12	a23	a13
CritDes	0.67	0.57	0.54	0.69	0.62
InitEDA	0.69	0.49	0.69	0.85	0.54
InterpRes	0.22	0.23	0.62	0.62	0.54
RsrchQ	0.21	0.19	0.38	0.54	0.77
SelMeth	0.47	0.52	0.92	0.69	0.62
TxtOrg	0.19	0.14	0.69	0.54	0.62
VisOrg	0.66	0.59	0.54	0.77	0.77

Table 5: Table of ICCs and percent exact agreements

Looking at the ICCs for the repeated only data, we see that raters generally do not agree much on Research Question, Interpret Results, and Text Organization. They agree around half the time on Initial EDA and Select Methods, and a bit more than half the time on Critique Design and Visual Organization. Generally, there is not a rubric with an overly high degree of agreement between the raters according to these ICCs.

Looking at the percent agreements, it does not seem like there is one rater who consistently disagrees with the others. The lowest percent agreement value is different for each of the three rubrics with the lowest ICCs (i.e., the rubrics with the lowest general agreement). For Research Question, Raters 1 and 2 have the lowest agreement. For Interpret Results, Raters 1 and 3 have the lowest agreement. For Text Organization, Raters 2 and 3 have the lowest agreement. Therefore, it seems that who disagrees with whom depends on rubric, and there is not one rater who constantly disagrees with the others.

The ICCs calculated using the full data set are generally pretty similar to the ICCs we found on the data with only repeated artifacts, though the ICCs of Critique Design and Initial EDA on the full dataset are a good bit higher. When looking at the full dataset, there are some

rubrics where the raters generally agree on their scores (Critique Design, Initial EDA, and Visual Organization) and some where they have low agreement (Interpret Results, Research Question, and Text Organization).

4.3 Looking into how the Various Factors are Related to Ratings and if the Factors Interact

For the seven rubric-specific random intercept models using just the data from the artifacts that all three raters saw, we find that adding fixed effects does not improve the fit of any of the models (see Technical Appendix, Pages 18-19). Since we did not find that any fixed effects are significant, we are not going to try interactions or new random effects. Thus, these models fit a different intercept for each artifact. See Technical Appendix, pages 19-24 for model summaries. We can interpret the intercept of these models by saying that the overall mean rating for CritDes is 1.72 (and similarly for the other six rubrics). We can interpret the random effect coefficients by saying that the mean rating for CritDes, Artifact O5 is 1.72 + 1.00 = 2.72 (and similarly for the other random effect coefficients). This random effect coefficient was higher relative to the other coefficients for the CritDes rubric, which indicates that the mean rating for this artifact differs quite a bit from the overall mean rating for the rubric. Similar interpretations could be made for any random effect coefficient for any rubric.

The model summaries also include two estimates of variance pertaining to the random effects (see Technical Appendix, Pages 19-24). The first (the "Artifact" group in the summary) represents how much the prediction of the rating for the artifact varies from the fixed effect. The fact that these values are highest for the CritDes and VisOrg rubrics indicates that there is more variability across artifacts for these rubrics compared to the others (above and beyond what is captured by the overall mean fixed effect). The second (the "Residual" group in the summary) represents the general variance in the ratings (not capturing differences in individual artifacts). The fact that this value is highest for the TxtOrg rubric means that there is generally more variability in the ratings for this rubric compared to the others.

When looking at the seven models fit using the full data set and deleting the observations with missing data, we find that the results are different. For InitEDA, RsrchQ, and TxtOrg, adding fixed effects does not improve the fit of the model. However, adding Rater and removing the intercept improves the fit for CritDes, InterpRes, and VisOrg, and adding Rater, adding Semester, and removing the intercept improves the fit for SelMeth. Thus, for some rubrics, it seems that Rater is related to Ratings, and for one rubric, Semester is related to Ratings (see Technical Appendix, Pages 24-26).

Based on the t-values and the significant likelihood ratio test (see Technical Appendix, Pages 26-30), it seems that including Rater in the models for CritDes, InterpRes, and VisOrg

really does matter. There are no fixed effect interactions to try since Rater is the only fixed effect included in these models. Since there are more random effects than there are observations in the data set, the models with the random intercept of Rater grouped by Artifact cannot be fit, so we also are not including any new random intercepts. Therefore, the final model for CritDes, InterpRes, and VisOrg includes Rater as a fixed effect, but no additional fixed interactions or random effects.

Similar to what we found for the previous rubrics, we see that based on the t-statistics and likelihood ratio test p-value, including Rater in the model for SelMeth matters (see Technical Appendix, Pages 30-32). We also see that including Semester in the model matters according to the t-statistic. We tried the interaction of Rater and Semester, but it was not significant. Since there are more random effects than there are observations in the data set, the model with the random intercept of Rater grouped by Artifact and the model with the random intercept of Semester grouped by Artifact cannot be fit, so we also are not including any new random intercepts. Therefore, the final model for SelMeth includes Rater and Semester as fixed effects, but no additional fixed interactions or random effects. See Technical Appendix, pages 32-48 for final model summaries. Below, we summarize the coefficients of the fixed effects and the estimates of the random effect variance for the Artifact group for the final seven models in Table 6.

Fixed Effect	RsrchQ	CritDes	InitEDA	SelMeth	InterpRes	VisOrg	TxtOrg
Intercept	2.35	-	2.44	-	-	-	2.59
Rater1	-	1.69	-	2.25	2.70	2.38	-
Rater2	-	2.11	-	2.23	2.59	2.65	-
Rater3	-	1.89	-	2.03	2.14	2.28	-
Semester S19	-	-	-	-0.36	-	-	-
Artifact Variance	0.07	0.43	0.37	0.09	0.06	0.29	0.09

Table 6: Fixed effects coefficients and estimates of the random effect variance for the Artifact group for the final seven models

Some examples of how to interpret these fixed effects would be that 2.35 is the overall mean rating for the RsrchQ rubric, and compared to the fall semester, the ratings on the SelMeth rubric are 0.36 units lower on average.

The estimated random effect coefficients are also included in the model summaries in the Technical Appendix (Pages 32-48) and can be interpreted in the same way we described with the previous seven models. There are several estimates that stand out from the others as being substantially higher or lower than the others, and this indicates that the mean rating for these artifacts differs quite a bit from the overall mean rating for the rubric. The two variance terms captured in the model summaries can also be interpreted similarly to the way we described with the previous seven models. The CritDes and InitEDA rubrics feature more variability across artifacts compared to the other rubrics (above and beyond what is captured by the overall mean fixed effect) (see Table 6). The TxtOrg rubric again features generally more variability in the ratings compared to the other rubrics.

Although these models allow us to look at the relationship between different factors of the experiment with Ratings, they do not allow us directly examine interactions with Rubric. Therefore, we will try modelling the data in a single model, starting with a model that includes Rubric as a random effect grouped by Artifact.

The best final combined model that we find includes Rater, Semester, Rubric, and the interaction of Rater and Rubric as fixed effects and Rubric and Rater as random effects (grouped by Artifact) (see Technical Appendix, pages 48-56) according to AIC and the likelihood ratio test. Below in Table 7 we summarize the coefficients of the fixed effects and the estimates of the random effect variance for the Artifact group .

Variable Name	Coefficient
Intercept	1.76
Rater2	0.37
Rater3	0.20
SemesterS19	-0.16
RubricInitEDA	0.74
RubricInterpRes	0.99
RubricRsrchQ	0.73
RubricSelMeth	0.41
RubricTxtOrg	1.02
RubricVisOrg	0.65

Rater2:RubricInitEDA	-0.3
Rater3:RubricInitEDA	-0.29
Rater2:RubricInterpRes	-0.51
Rater3:RubricInterpRes	-0.71
Rater2:RubricRsrchQ	-0.49
Rater3:RubricRsrchQ	-0.32
Rater2:RubricSelMeth	-0.39
Rater3:RubricSelMeth	-0.39
Rater2:RubricTxtOrg	-0.55
Rater3:RubricTxtOrg	-0.44
Rater2:RubricVisOrg	-.10
Rater3:RubricVisOrg	-0.28
RubricCritDes (Variance)	0.5
RubricInitEDA (Variance)	0.32
RubricInterpRes (Variance)	0.1
RubricRsrchQ (Variance)	0.18
RubricSelMeth (Variance)	0.04
RubricTxtOrg (Variance)	0.25
RubricVisOrg (Variance)	0.23

Table 7: Fixed effects coefficients and estimates of the random effect variance for the Artifact group for the final model

We can interpret the relationship between the fixed effects and Rating. Compared to Rater 1, we would expect the ratings from Rater 2 to be 0.37 units higher and the ratings from Rater 3 to be 0.2 units higher on average, holding all other predictors constant. Compared to the Fall semester, we would expect the ratings from the Spring semester to be 0.16 units lower on average, holding all other predictors constant. Compared to the CritDes rubric, we would expect the ratings on the InitEDA rubric to be 0.74 units higher, the ratings on the InterpRes rubric to be 0.99 units higher, the ratings on the RsrchQ rubric to be 0.73 units higher, the ratings on the

SelMeth rubric to be 0.41 units higher, the ratings on the TxtOrg rubric to be 1.02 units higher, and the ratings on the VisOrg rubric to be 0.65 units higher on average, holding all other predictors constant.

The interaction terms are a bit more complex to interpret. Compared to a rating from Rater 1 on the CritDes rubric, we would expect ratings from Rater 2 on the InitEDA rubric to be 0.3 units lower and ratings from Rater 3 to be 0.29 units lower on average. Compared to a rating from Rater 1 on the CritDes rubric, we would expect ratings from Rater 2 on the InterpRes rubric to be 0.51 units lower and ratings from Rater 3 to be 0.71 units lower on average. Compared to a rating from Rater 1 on the CritDes rubric, we would expect ratings from Rater 2 on the RsrchQ rubric to be 0.49 units lower and ratings from Rater 3 to be 0.32 units lower on average. Compared to a rating from Rater 1 on the CritDes rubric, we would expect ratings from Rater 2 on the SelMeth rubric to be 0.39 units lower and ratings from Rater 3 to be 0.39 units lower on average. Compared to a rating from Rater 1 on the CritDes rubric, we would expect ratings from Rater 2 on the TxtOrg rubric to be 0.55 units lower and ratings from Rater 3 to be 0.44 units lower on average. Compared to a rating from Rater 1 on the CritDes rubric, we would expect ratings from Rater 2 on the VisOrg rubric to be 0.1 units lower and ratings from Rater 3 to be 0.28 units lower on average.

The random effect estimated coefficients can be interpreted in a similar way to how they were interpreted previously, and they are included in the Technical Appendix (pages 51-56). There are several estimates that stand out from the others as being substantially higher or lower than the others, and this indicates that the mean rating for these artifacts differs quite a bit from the overall mean rating for the rubric or rater in question. The variance terms pertaining to the random effects can also be interpreted similarly to how they were interpreted previously (see Technical Appendix, Page 50 and Table 7). We see that the ratings for the CritDes rubric feature more variability across artifacts compared to the other rubrics.

4.4 Finding Other Interesting Things About This Data

The Technical Appendix (pages 56-72) includes all bar plots and density curves that are being compared in this section. Comparing the bar plots for CritDes, it makes sense why Rater would be included in the model using the full data set and not just the repeated data set. The distributions of ratings look roughly similar for the repeated data, whereas the distribution of ratings for Rater 1 looks quite different from the other two raters with a majority of ratings of 1.

Similar to the CritDes bar plots, we see that when we look at the full dataset, the distribution of ratings between raters seems to differ for InterpRes. Rater 1 gives mostly ratings of 3, Rater 2 gives similar numbers of 2 and 3 ratings, and Rater 3 gives mostly 2 ratings. In

contrast, the three raters have similar distributions of ratings when looking at the reduced dataset, with all three raters giving roughly similar numbers of 2 and 3 ratings.

The distribution of ratings between the different raters also seems to differ for the VisOrg rubric when looking at the full dataset. Raters 1 and 3 give out mostly 2s while Rater 2 gives out more 3s. Given the small sample size, the distributions for the reduced data seem roughly similar.

There are slight differences in the distributions of ratings given the full dataset for SelMeth. Rater 1 gives almost all 2s while Raters 2 and 3 give some 1s and 3s in addition to mostly 2s. There are also differences in the distributions of ratings given semester for the full dataset. Practically all ratings in the spring are 2 whereas there were a decent number of 3s in the fall.

Comparing these bar plots allows us to see how the models fitted to the data from the 13 common items, vs fitting to all the data are different since there are clearer differences in the distributions of ratings when looking at the full dataset compared with the reduced set.

Next, we produce some plots of the Semester, Sex, and Repeated variables to see if we can visually support our choices to include or not includes these factors in modelling ratings.

Looking at the density curves for Semester faceted by rubric, we see that it makes sense that Semester is included as a fixed effect in the random-intercept model for SelMeth as the distribution seems to be generally shifted to the left for the spring compared to the fall. It also makes sense that Semester was not included as a fixed effect in the other random-intercept models since the distributions of fall and spring ratings look similar for the other six rubrics. Looking at the density curves for Semester as a whole (no facets), we see that the distributions between Fall and Spring ratings when looking at all the data do not appear to be that different. However, since the combined model we fit includes interactions with Rubric, it makes sense that Semester would still be added as a fixed effect since for at least one rubric the distributions of ratings between the two semesters appear to be different.

The distributions of ratings given Sex for each rubric and for the data all together appear to be very similar- each are generally bimodal and the male and female curves mostly overlap with each other. This suggests that there is not a difference in the distribution of ratings for artifacts created by males versus females. Thus, it makes sense that Sex was not included in any modelling.

Similar to Sex, the distributions of ratings given whether or not the artifact was seen by all three raters also appear to be very similar for each rubric and for the data all together- each are generally bimodal and the curves mostly overlap with each other. This suggests that there is

not a difference in the distribution of ratings for artifacts for artifacts rated by all three raters vs just one. Thus, it makes sense that Repeated was not included in any modelling.

5 Discussion

In this paper, we aim to evaluate the effectiveness and fairness of the rating system that Dietrich College is using to evaluate the success of the new General Education program. We discover that the ratings on student artifacts are related to the rater, rubric, and semester, which suggests that there are potential flaws in the rating system. Perhaps rater trainings are ineffective or some rubrics are just more challenging than others. Regardless of the reasons for the differences in the ratings, it is important for the Dean's office to know that ratings are not always consistent across different raters, rubrics, and semesters.

We find that while most rubrics tend to have similar distributions of ratings, the CritDes and SelMeth rubrics seem to get lower ratings. This may be because these rubric items are just more difficult than the others, or it could be that these items are not being taught as effectively as the others. We also find that the distribution of ratings for Raters 1 and 2 are similar, but it seems like Rater 3 gives lower ratings. Perhaps this rater is just more critical than the others, or maybe the raters were not trained well enough to make the ratings consistent.

We determine that raters do not generally agree on their scores, particularly for the Research Question, Interpret Results, and Text Organization rubrics. Based on percent agreements, we find that who disagrees with whom depends on rubric, and there is not one rater who constantly disagrees with the others. This would again suggest that perhaps rater training is ineffective since ratings do not seem to be very consistent.

When looking at the seven random-intercept models, we see that Rater and Semester are related to ratings for some rubrics. When looking at the single combined model, we see that Rater, Semester, and Rubric are related to ratings. Specifically, the model that includes Rater, Semester, Rubric, and the interaction of Rater and Rubric as fixed effects and Rubric and Rater as random effects (grouped by Artifact) is the best fit. The fact that fixed effects are added to these models suggests that there may be something unfair about the ratings. As we have mentioned previously, this may be due to ineffective rater training. It could also be because raters are from different departments and thus may have different criteria that they focus on, or maybe some rubrics are just easier to score well on than others. The random effects estimates that stand out suggest that the mean rating for these artifacts differs quite a bit from the overall mean rating for the rubric or rater in question. These represent artifacts that seem to be rated differently from the others, and it would be worth looking into other features of these observations that could potentially explain these differences. The rubrics with high random effect variance estimates (for the Artifact group) feature more variability across artifacts compared to the other rubrics. It is

possible that these rubrics are just more subjective than the others, or maybe the skills pertaining to these rubrics are just less straightforward to teach to students.

In examining the differences in the distribution of ratings given rater for each rubric, we find that it makes sense that there are differences in distributions of ratings when looking at the full dataset compared with the reduced set for the rubrics that ultimately include Rater as a fixed effect in the random-intercept models fit with the full data set. For these rubrics, we are able to visually see differences in the distributions of ratings, and this helps us understand why fixed effects are added to the model. By producing density curves of Ratings filled by Semester, Sex, and Repeated, we are able to visually justify including Semester and not including Sex or Repeated in modelling.

A weakness of this study is that it is difficult to actually measure student success when using a simple four point rating scale, and it is hard to know how well the ratings are actually capturing the success of the new General Education program. In addition, the sample size of the data using only the artifacts that were rated by all three raters is relatively small, and thus may not be representative of the actual results we would find. A future study could include more artifacts that are repeated across all raters. A future study could also include more information about the students, like major and what year of college they are in, to help determine if personal factors that are not being captured in this analysis contribute to success in the general education courses.

In summary, we find that ratings are influenced by rater, rubric, and semester, so when evaluating the success of the general education program, it is important to keep these factors in mind since they may suggest that something unfair is going on. In order for the college to provide the best education for students, they should be sure to understand the factors that contribute to the ratings before making decisions based on the results of their experiments.

6 References

Junker, B. W. (2021). *Project 02 assignment sheet and data for 36-617: Applied Regression Analysis.* Department of Statistics and Data Science, Carnegie Mellon University, Pittsburgh PA. Accessed Nov 17, 2021 from https://canvas.cmu.edu/courses/25337/files/folder/Project02

Sheather, S.J. (2009). *A Modern Approach to Regression with R*. New York: Springer Science + Business Media LLC.

36-617 Project 2 Technical Appendix

Megan Christy

12/10/2021

Contents

Loading Packages and Reading in the Data

To begin, we load in the necessary packages and read in the data sets.

```
# Loading the necessary packages
library(arm)
```

```
## Loading required package: MASS
```

```
## Loading required package: Matrix
```

```
## Loading required package: lme4
```

```
##
## arm (Version 1.12-2, built: 2021-10-15)
```

```
## Working directory is /Users/meganchristy/Downloads
```

```
library(plyr)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:plyr':
##
##     arrange, count, desc, failwith, id, mutate, rename, summarise,
##     summarize
```

```
## The following object is masked from 'package:MASS':
##
##     select
```

```
## The following objects are masked from 'package:stats':
##
```

```
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(LMERConvenienceFunctions,
        warn.conflicts = F, quietly = T)
library(lme4, warn.conflicts = F, quietly = T)
library(RLRsim)
library(ggplot2)
library(kableExtra)
```

```
##
## Attaching package: 'kableExtra'
```

```
## The following object is masked from 'package:dplyr':
##
##     group_rows
```

```
library(knitr)
```

```
# Reading in the data (and a bit of cleaning)

# wide format
ratings.data = read.csv("ratings.csv", header = T)

# tall format
tall.ratings = read.csv("tall.csv", header = T)

# Make sure ratings run from 1 to 4, and code them as a factor
tall.ratings$Rating <- factor(tall.ratings$Rating,levels=1:4)
for (i in unique(tall.ratings$Rubric)) {
  ratings.data[,i] <- factor(ratings.data[,i],levels=1:4)
}

# Make sure missing Sex value is consistent across both datasets
tall.ratings$Sex[nchar(tall.ratings$Sex)==0] <- "--"

# Extract the reduced dataset with just the artifacts that all three raters saw
ratings.repeated = subset(ratings.data, ratings.data$Repeated == 1)
tall.repeated = subset(tall.ratings, tall.ratings$Repeated == 1)
```

Here we print the first few rows of the data so we can become familiar with it before diving into exploration.

```
# Heads of the data
head(ratings.data)
```

```
##   X Rater Sample Overlap Semester Sex RsrchQ CritDes InitEDA SelMeth InterpRes
## 1 1     3      1       5     Fall   M      3       3       2       2         2
## 2 2     3      2       7     Fall   F      3       3       3       3         3
## 3 3     3      3       9   Spring   F      2       1       3       2         3
## 4 4     3      4       8   Spring   M      2       2       2       1         1
## 5 5     3      5      NA     Fall  --      3       3       3       3         3
## 6 6     3      6      NA     Fall   M      2       1       2       2         2
##   VisOrg TxtOrg Artifact Repeated
## 1      2      3       05        1
```

```
## 2      3      3       07        1
## 3      3      3       09        1
## 4      1      1       08        1
## 5      3      3        5        0
## 6      2      2        6        0
```

```
head(tall.ratings)
```

```
##   X Rater Artifact Repeated Semester Sex Rubric Rating
## 1 1     3       05        1      F19   M RsrchQ      3
## 2 2     3       07        1      F19   F RsrchQ      3
## 3 3     3       09        1      S19   F RsrchQ      2
## 4 4     3       08        1      S19   M RsrchQ      2
## 5 5     3        5        0      F19  -- RsrchQ      3
## 6 6     3        6        0      F19   M RsrchQ      2
```

Exploratory Data Analysis

Before diving into our research questions, we will perform some EDA. Specifically, we will create a table of counts for each of the variables in the data set.

```
# tables of counts for the variables
kable(table(ratings.data$Rater), col.names = c("Rater", "Freq"))
```

Rater	Freq
1	39
2	39
3	39

```
kable(table(ratings.data$Semester), col.names = c("Semester", "Freq"))
```

Semester	Freq
Fall	83
Spring	34

```
kable(table(ratings.data$Sex), col.names = c("Sex", "Freq"))
```

Sex	Freq
–	1
F	64
M	52

```
kable(table(ratings.data$Repeated), col.names = c("Repeated", "Freq"))
```

Repeated	Freq
0	78
1	39

```
# counts of ratings by rubric
counts.tab = rbind(summary(ratings.data$CritDes),
                   summary(ratings.data$InitEDA),
                   summary(ratings.data$InterpRes),
                   summary(ratings.data$RsrchQ),
                   summary(ratings.data$SelMeth),
                   summary(ratings.data$TxtOrg), summary(ratings.data$VisOrg))
```

```
## Warning in rbind(summary(ratings.data$CritDes), summary(ratings.data$InitEDA), :
## number of columns of result is not a multiple of vector length (arg 2)
```

```
rownames(counts.tab) = c("CritDes", "InitEDA", "InterpRes", "RsrchQ", "SelMeth",
                         "TxtOrg", "VisOrg")
# manually fixing NA column
counts.tab[c(2:6), 5] = 0
kable(counts.tab)
```

	1	2	3	4	NA's
CritDes	47	39	28	2	1
InitEDA	8	56	47	6	0
InterpRes	6	49	61	1	0
RsrchQ	6	65	45	1	0
SelMeth	10	89	18	0	0
TxtOrg	8	37	66	6	0
VisOrg	7	59	45	5	1

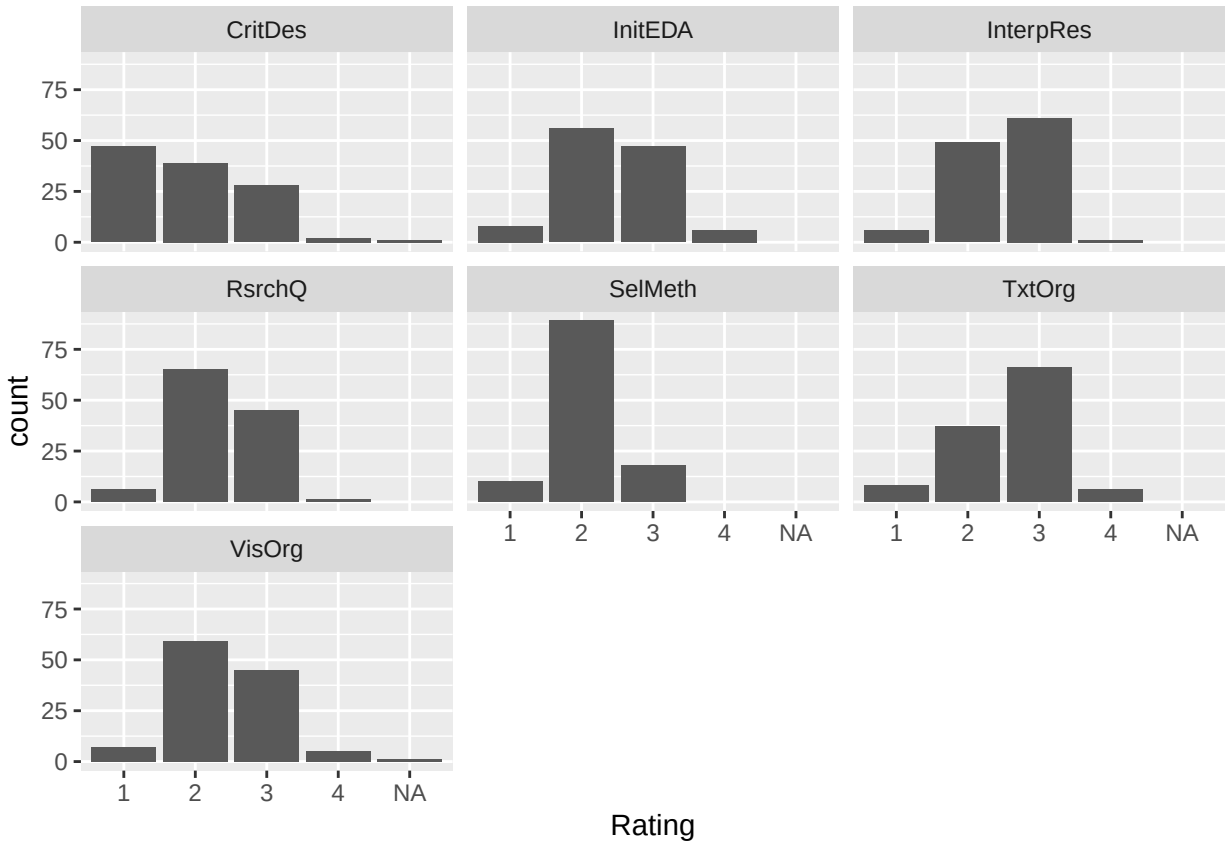
We see that each rater rated 39 artifacts and that slightly more artifacts that were rated came from the fall. We also see that slightly more of the artifacts were created by females than males, and 39 of the observations in the data represent one of the thirteen artifacts seen by all three raters. Looking at the tables of counts for the ratings for each rubric, we see that for almost all rubrics, the majority of the ratings are 2 or 3. The exception to this is the CritDes rubric, which has a majority rating of 1. We also notice that the SelMeth rubric is the only rubric with no ratings of 4, and the CritDes and TxtOrg rubrics both feature a missing observation.

Now, we will dive into the research questions, where we will first take a deeper dive into the distributions of ratings by rubric that we just saw in our EDA.

Examining and Comparing the Distributions of Ratings for Each Rubric

To compare the distributions of ratings across rubrics, we construct bar plots and produce a table of counts for each rubric first using the full dataset.

```
# bar plots of ratings by rubric
ggplot(tall.ratings, aes(x = Rating)) +
facet_wrap( ~ Rubric) + geom_bar()
```




```
# counts of ratings by rubric
counts.tab = rbind(summary(ratings.data$CritDes),
                   summary(ratings.data$InitEDA),
                   summary(ratings.data$InterpRes),
                   summary(ratings.data$RsrchQ),
                   summary(ratings.data$SelMeth),
                   summary(ratings.data$TxtOrg),
                   summary(ratings.data$VisOrg))
```

```
## Warning in rbind(summary(ratings.data$CritDes), summary(ratings.data$InitEDA), :
## number of columns of result is not a multiple of vector length (arg 2)
```

```
rownames(counts.tab) = c("CritDes", "InitEDA", "InterpRes", "RsrchQ", "SelMeth",
                         "TxtOrg", "VisOrg")
# manually fixing NA column
counts.tab[c(2:6), 5] = 0
kable(counts.tab)
```

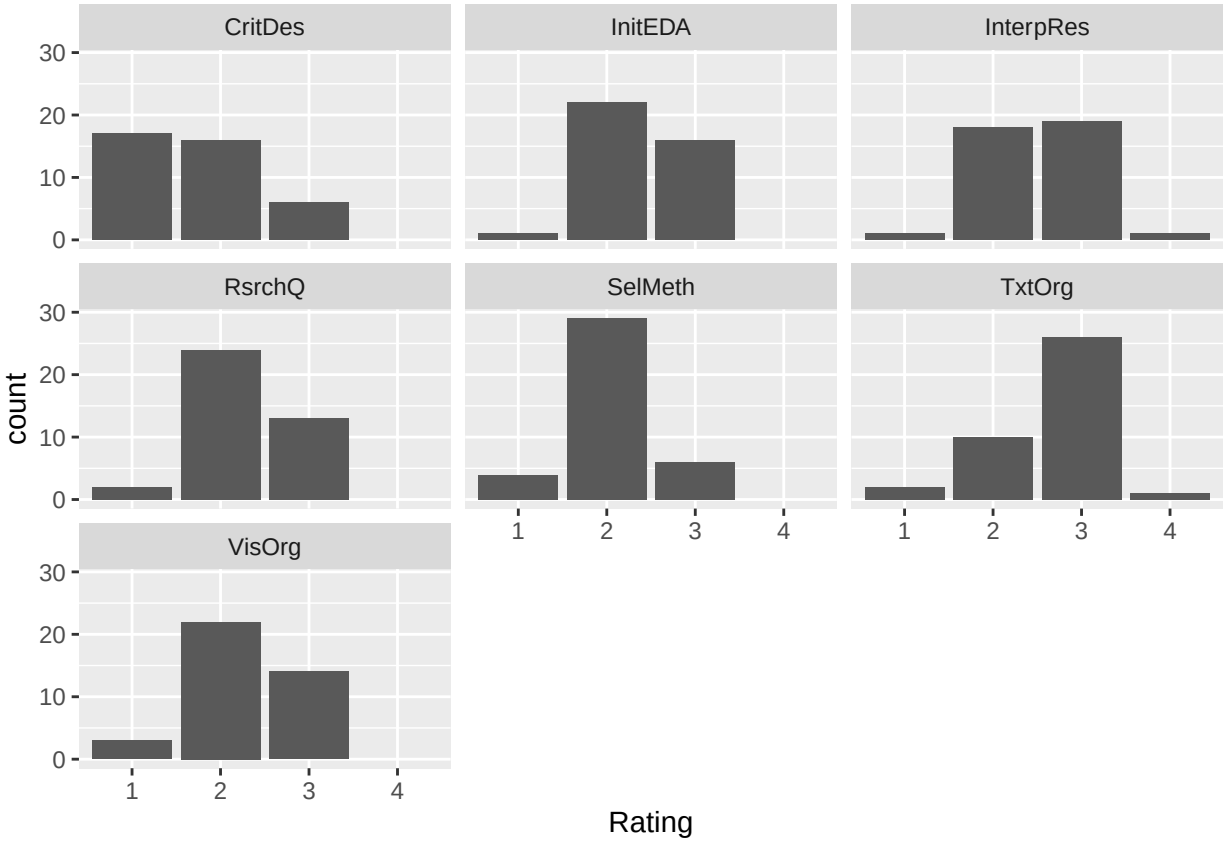
	1	2	3	4	NA's
CritDes	47	39	28	2	1
InitEDA	8	56	47	6	0
InterpRes	6	49	61	1	0
RsrchQ	6	65	45	1	0
SelMeth	10	89	18	0	0
TxtOrg	8	37	66	6	0
VisOrg	7	59	45	5	1

Looking at the bar plots and counts, we see that the distributions of ratings for the InitEDA, InterpRes, RsrchQ, TxtOrg, and VisOrg rubrics are generally similar, with most ratings being either 2 or 3. Out of

these five rubrics, we see that most of the observations for the InterpRes and TxtOrg rubrics have a rating of 3 while the rest of the five have 2 as the most common rating, suggesting that these two rubrics tend to get slightly higher ratings than the rest of the five. The distribution of the CritDes rubric is different in that most of the ratings are 1, and the number of observations with each rating falls as rating increases. This suggests that this rubric tends to get especially low ratings. The SelMeth rubric is different in that the large majority of ratings are 2, and it is the only rubric with no ratings of 4. This suggests that this rubric may tend to get slightly lower ratings.

Next, we will make the same bar plots and table of counts for the dataset with just the artifacts that all three raters saw to see whether these artifacts are representative of the whole data set.

```
# bar plots of ratings by rubric
ggplot(tall.repeated, aes(x = Rating)) +
facet_wrap( ~ Rubric) + geom_bar()
```




```
# counts of ratings by rubric
counts.tab = rbind(summary(ratings.repeated$CritDes),
                   summary(ratings.repeated$InitEDA),
                   summary(ratings.repeated$InterpRes),
                   summary(ratings.repeated$RsrchQ),
                   summary(ratings.repeated$SelMeth),
                   summary(ratings.repeated$TxtOrg),
                   summary(ratings.repeated$VisOrg))
rownames(counts.tab) = c("CritDes", "InitEDA", "InterpRes", "RsrchQ", "SelMeth",
                         "TxtOrg", "VisOrg")

kable(counts.tab)
```

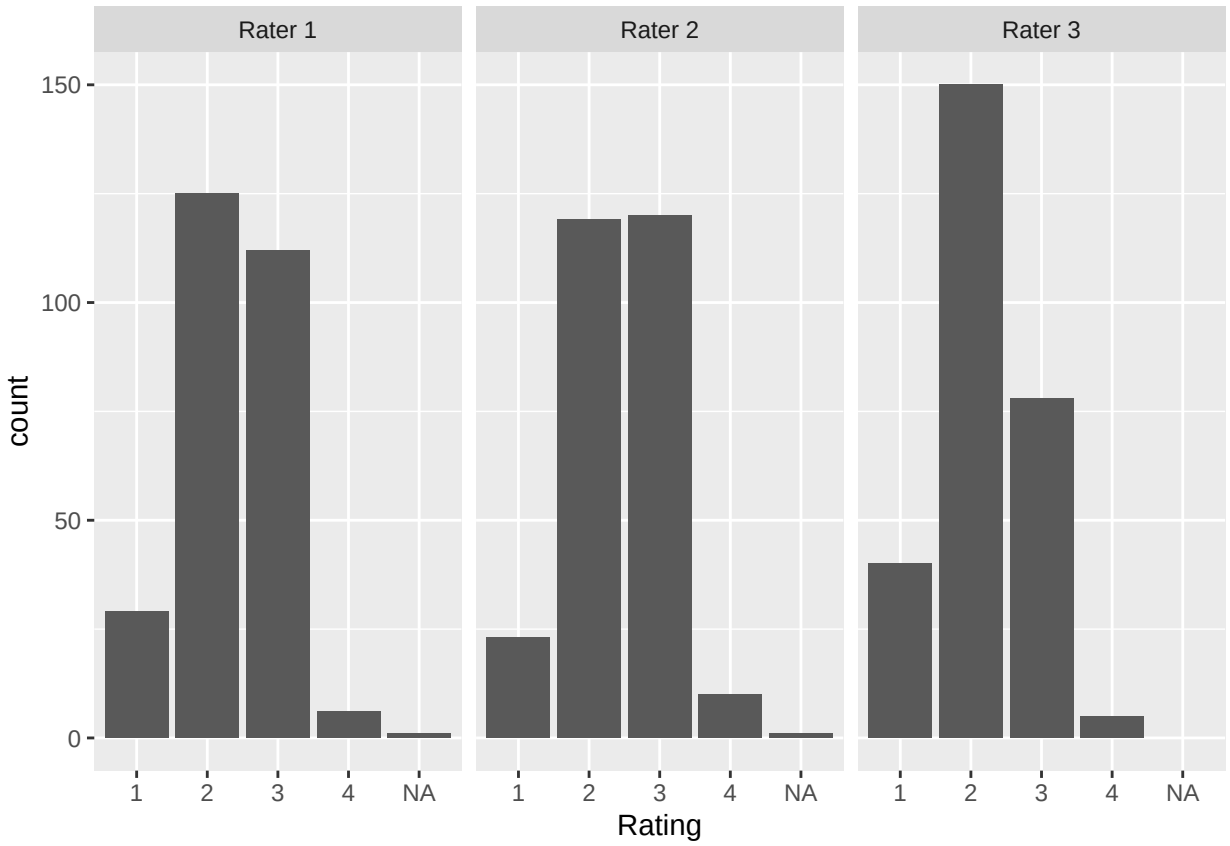
	1	2	3	4
CritDes	17	16	6	0
InitEDA	1	22	16	0
InterpRes	1	18	19	1
RsrchQ	2	24	13	0
SelMeth	4	29	6	0
TxtOrg	2	10	26	1
VisOrg	3	22	14	0

Looking at the barplots and table of counts, we see that the distributions of ratings for each rubric are largely similar to the distributions we saw when looking at the full dataset. The CritDes and SelMeth rubrics stand out for the same reasons as before, and InterpRes and TxtOrg still have more ratings of 3 than ratings of 2, which is different from the rubrics with similar distributions. This suggests that these thirteen artifacts are generally representative of the full dataset. One interesting difference to note, however, is that there are only two ratings of 4 in the entire reduced dataset, whereas all but one of the rubrics in the full dataset had at least one rating of 4. This suggests that in some cases the ratings on these artifacts tended to be a bit lower when we just look at the reduced dataset.

Examining and Comparing the Distributions of Ratings for Each Rater

To compare the distributions of ratings across raters, we construct bar plots and produce a table of counts for each rater first using the full dataset.

```
# bar plots of ratings by rater
rater.name <- function(x) { paste("Rater",x) }
ggplot(tall.ratings,aes(x = Rating)) +
facet_wrap( ~ Rater, labeller=labeller(Rater=rater.name)) +
  geom_bar()
```




```
# counts of ratings by rater
rat1 = subset(tall.ratings, tall.ratings$Rater == 1)
rat2 = subset(tall.ratings, tall.ratings$Rater == 2)
rat3 = subset(tall.ratings, tall.ratings$Rater == 3)
counts.tab = rbind(summary(rat1$Rating), summary(rat2$Rating),
                   summary(rat3$Rating))
```

```
## Warning in rbind(summary(rat1$Rating), summary(rat2$Rating),
## summary(rat3$Rating)): number of columns of result is not a multiple of vector
## length (arg 3)
```

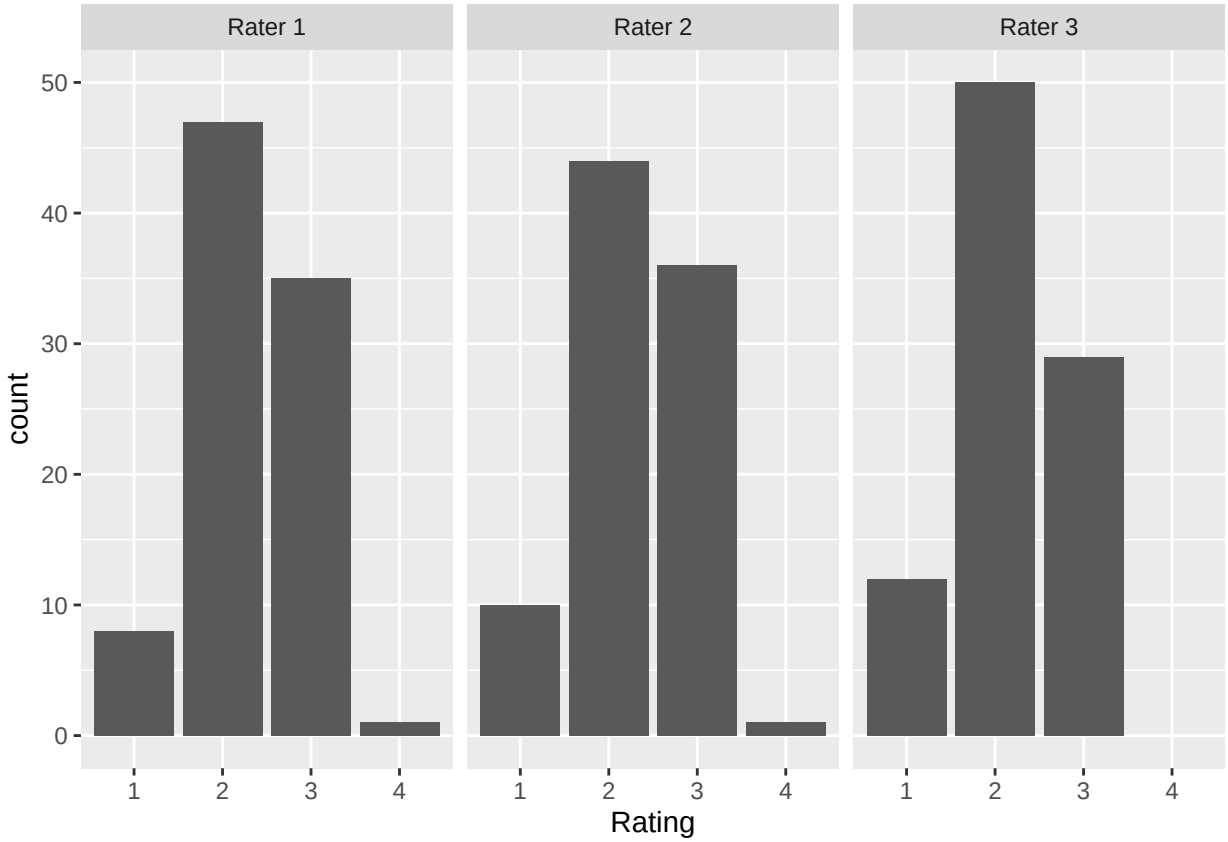
```
rownames(counts.tab) = c("Rater1", "Rater2", "Rater3")

# manually fix NA column
counts.tab[3,5] = 0

kable(counts.tab)
```

	1	2	3	4	NA's
Rater1	29	125	112	6	1
Rater2	23	119	120	10	1
Rater3	40	150	78	5	0

The distribution of ratings for Raters 1 and 2 appear to be similar, with most ratings being either 2 or 3. The majority of ratings for Rater 3 are 2, and Rater 3 gave more ratings of 1 than the other two. Thus, it seems that Rater 3 tends to give somewhat lower ratings.

Next, we will make the same bar plots and table of counts for the dataset with just the artifacts that all three raters saw to see whether these artifacts are representative of the whole data set.

```
# bar plots of ratings by rater
ggplot(tall.repeated,aes(x = Rating)) +
facet_wrap( ~ Rater, labeller=labeller(Rater=rater.name)) +
  geom_bar()
```




```
# counts of ratings by rater
rat1 = subset(tall.repeated, tall.repeated$Rater == 1)
rat2 = subset(tall.repeated, tall.repeated$Rater == 2)
rat3 = subset(tall.repeated, tall.repeated$Rater == 3)
counts.tab = rbind(summary(rat1$Rating), summary(rat2$Rating),
                   summary(rat3$Rating))
rownames(counts.tab) = c("Rater1", "Rater2", "Rater3")

kable(counts.tab)
```

	1	2	3	4
Rater1	8	47	35	1
Rater2	10	44	36	1
Rater3	12	50	29	0

Looking at the barplots and table of counts, we see that the distributions of ratings for each rater are largely similar to the distributions we saw when looking at the full dataset. The distribution of ratings for Raters 1 and 2 appear to be similar, with most ratings being either 2 or 3. The same is true for Rater 3, but Rater 3 gives fewer ratings of 3 and no ratings of 4. Thus, it seems that Rater 3 tends to give somewhat lower ratings, and this is similar to what we found when looking at the full dataset.

Note on the Missing Data

In looking at the distributions of ratings, we have found that there is missing data. We will now take a look at these missing values and comment on how they potentially impact our analysis. We know that there is a missing value of sex, a missing value of CritDes, and a missing value of VisOrg.

```
# printing the rows with missing data
tall.ratings[which(is.na(tall.ratings$Rating == TRUE)),]
```

```
##       X Rater Artifact Repeated Semester Sex  Rubric Rating
## 161 161     2       45        0      S19   F CritDes   <NA>
## 684 684     1      100        0      F19   F  VisOrg   <NA>
```

```
ratings.data[which(ratings.data$Sex == "--"),]
```

```
##   X Rater Sample Overlap Semester Sex RsrchQ CritDes InitEDA SelMeth InterpRes
## 5 5     3      5      NA     Fall  --      3       3       3       3         3
##   VisOrg TxtOrg Artifact Repeated
## 5      3      3        5        0
```

It is important to note that when we are modelling with Rating as the response variable, the two observations with missing data will be dropped. This is important to keep in mind when we are comparing models since we may not be using the exact same data for models that involve different rubrics.

We chose to keep the missing Sex value coded the way it is so that the observation is not dropped when Sex is included in modelling.

Finally, it is worth noting that when we are using the reduced dataset that includes only artifacts that were rated by all three raters, there are no missing values, so we do not have to worry about any of these issues when using this data.

Examining Whether Raters Generally Agree on Their Scores

We will now look at rater agreement by rubric. Specifically, do the raters generally agree on their scores? If not, is there one rater who disagrees with the others? Or do they all disagree?

First, we fit seven random-intercept models, one for each rubric, and calculate the ICCs for each using the subset of the data for just the 13 artifacts seen by all three raters. Then, we will compute the percent exact agreement between each pair of raters for each rubric to determine who is agreeing/disagreeing with whom on each rubric.

```
# ICCs

# RsrchQ
RsrchQ.ratings <- tall.repeated[tall.repeated$Rubric=="RsrchQ",]
RsrchQ.mod = lmer(as.numeric(Rating) ~ 1 + (1|Artifact), data=RsrchQ.ratings)
#summary(RsrchQ.mod)
RsrchQ.vcov = as.data.frame(VarCorr(RsrchQ.mod))$vcov
RsrchQ.icc = RsrchQ.vcov[1]/(RsrchQ.vcov[1] + RsrchQ.vcov[2])

# CritDes
CritDes.ratings <- tall.repeated[tall.repeated$Rubric=="CritDes",]
CritDes.mod = lmer(as.numeric(Rating) ~ 1 + (1|Artifact), data=CritDes.ratings)
#summary(CritDes.mod)
CritDes.vcov = as.data.frame(VarCorr(CritDes.mod))$vcov
CritDes.icc = CritDes.vcov[1]/(CritDes.vcov[1] + CritDes.vcov[2])
```

```
# InitEDA
InitEDA.ratings <- tall.repeated[tall.repeated$Rubric=="InitEDA",]
InitEDA.mod = lmer(as.numeric(Rating) ~ 1 + (1|Artifact), data=InitEDA.ratings)
#summary(InitEDA.mod)
InitEDA.vcov = as.data.frame(VarCorr(InitEDA.mod))$vcov
InitEDA.icc = InitEDA.vcov[1]/(InitEDA.vcov[1] + InitEDA.vcov[2])

# SelMeth
SelMeth.ratings <- tall.repeated[tall.repeated$Rubric=="SelMeth",]
SelMeth.mod = lmer(as.numeric(Rating) ~ 1 + (1|Artifact), data=SelMeth.ratings)
#summary(SelMeth.mod)
SelMeth.vcov = as.data.frame(VarCorr(SelMeth.mod))$vcov
SelMeth.icc = SelMeth.vcov[1]/(SelMeth.vcov[1] + SelMeth.vcov[2])

# InterpRes
InterpRes.ratings <- tall.repeated[tall.repeated$Rubric=="InterpRes",]
InterpRes.mod = lmer(as.numeric(Rating) ~ 1 + (1|Artifact), data=InterpRes.ratings)
#summary(InterpRes.mod)
InterpRes.vcov = as.data.frame(VarCorr(InterpRes.mod))$vcov
InterpRes.icc = InterpRes.vcov[1]/(InterpRes.vcov[1] + InterpRes.vcov[2])

# VisOrg
VisOrg.ratings <- tall.repeated[tall.repeated$Rubric=="VisOrg",]
VisOrg.mod = lmer(as.numeric(Rating) ~ 1 + (1|Artifact), data=VisOrg.ratings)
#summary(VisOrg.mod)
VisOrg.vcov = as.data.frame(VarCorr(VisOrg.mod))$vcov
VisOrg.icc = VisOrg.vcov[1]/(VisOrg.vcov[1] + VisOrg.vcov[2])

# TxtOrg
TxtOrg.ratings <- tall.repeated[tall.repeated$Rubric=="TxtOrg",]
TxtOrg.mod = lmer(as.numeric(Rating) ~ 1 + (1|Artifact), data=TxtOrg.ratings)
#summary(TxtOrg.mod)
TxtOrg.vcov = as.data.frame(VarCorr(TxtOrg.mod))$vcov
TxtOrg.icc = TxtOrg.vcov[1]/(TxtOrg.vcov[1] + TxtOrg.vcov[2])
```

```
# Percent exact agreements

# RsrchQ
raters_1_and_2_on_RsrchQ <- data.frame(r1=ratings.repeated$RsrchQ[ratings.repeated$Rater==1],
                                       r2=ratings.repeated$RsrchQ[ratings.repeated$Rater==2],
                                       a1=ratings.repeated$Artifact[ratings.repeated$Rater==1],
                                       a2=ratings.repeated$Artifact[ratings.repeated$Rater==2])

r1 <- factor(raters_1_and_2_on_RsrchQ$r1,levels=1:4)
r2 <- factor(raters_1_and_2_on_RsrchQ$r2,levels=1:4)
t12 <- table(r1,r2)
RsrchQ_12 = sum(diag(t12))/13

raters_2_and_3_on_RsrchQ <- data.frame(r2=ratings.repeated$RsrchQ[ratings.repeated$Rater==2],
                                       r3=ratings.repeated$RsrchQ[ratings.repeated$Rater==3],
                                       a2=ratings.repeated$Artifact[ratings.repeated$Rater==2],
                                       a3=ratings.repeated$Artifact[ratings.repeated$Rater==3])

r2 <- factor(raters_2_and_3_on_RsrchQ$r2,levels=1:4)
```

```
r3 <- factor(raters_2_and_3_on_RsrchQ$r3,levels=1:4)
t23 <- table(r2,r3)
RsrchQ_23 = sum(diag(t23))/13

raters_1_and_3_on_RsrchQ <- data.frame(r1=ratings.repeated$RsrchQ[ratings.repeated$Rater==1],
                                       r3=ratings.repeated$RsrchQ[ratings.repeated$Rater==3],
                                       a1=ratings.repeated$Artifact[ratings.repeated$Rater==1],
                                       a3=ratings.repeated$Artifact[ratings.repeated$Rater==3])

r1 <- factor(raters_1_and_3_on_RsrchQ$r1,levels=1:4)
r3 <- factor(raters_1_and_3_on_RsrchQ$r3,levels=1:4)
t13 <- table(r1,r3)
RsrchQ_13 = sum(diag(t13))/13

# CritDes
raters_1_and_2_on_CritDes <- data.frame(r1=ratings.repeated$CritDes[ratings.repeated$Rater==1],
                                       r2=ratings.repeated$CritDes[ratings.repeated$Rater==2],
                                       a1=ratings.repeated$Artifact[ratings.repeated$Rater==1],
                                       a2=ratings.repeated$Artifact[ratings.repeated$Rater==2])

r1 <- factor(raters_1_and_2_on_CritDes$r1,levels=1:4)
r2 <- factor(raters_1_and_2_on_CritDes$r2,levels=1:4)
t12 <- table(r1,r2)
CritDes_12 = sum(diag(t12))/13

raters_2_and_3_on_CritDes <- data.frame(r2=ratings.repeated$CritDes[ratings.repeated$Rater==2],
                                       r3=ratings.repeated$CritDes[ratings.repeated$Rater==3],
                                       a2=ratings.repeated$Artifact[ratings.repeated$Rater==2],
                                       a3=ratings.repeated$Artifact[ratings.repeated$Rater==3])

r2 <- factor(raters_2_and_3_on_CritDes$r2,levels=1:4)
r3 <- factor(raters_2_and_3_on_CritDes$r3,levels=1:4)
t23 <- table(r2,r3)
CritDes_23 = sum(diag(t23))/13

raters_1_and_3_on_CritDes <- data.frame(r1=ratings.repeated$CritDes[ratings.repeated$Rater==1],
                                       r3=ratings.repeated$CritDes[ratings.repeated$Rater==3],
                                       a1=ratings.repeated$Artifact[ratings.repeated$Rater==1],
                                       a3=ratings.repeated$Artifact[ratings.repeated$Rater==3])

r1 <- factor(raters_1_and_3_on_CritDes$r1,levels=1:4)
r3 <- factor(raters_1_and_3_on_CritDes$r3,levels=1:4)
t13 <- table(r1,r3)
CritDes_13 = sum(diag(t13))/13

# InitEDA
raters_1_and_2_on_InitEDA <- data.frame(r1=ratings.repeated$InitEDA[ratings.repeated$Rater==1],
                                       r2=ratings.repeated$InitEDA[ratings.repeated$Rater==2],
                                       a1=ratings.repeated$Artifact[ratings.repeated$Rater==1],
                                       a2=ratings.repeated$Artifact[ratings.repeated$Rater==2])

r1 <- factor(raters_1_and_2_on_InitEDA$r1,levels=1:4)
r2 <- factor(raters_1_and_2_on_InitEDA$r2,levels=1:4)
```

```
t12 <- table(r1,r2)
InitEDA_12 = sum(diag(t12))/13

raters_2_and_3_on_InitEDA <- data.frame(r2=ratings.repeated$InitEDA[ratings.repeated$Rater==2],
                                       r3=ratings.repeated$InitEDA[ratings.repeated$Rater==3],
                                       a2=ratings.repeated$Artifact[ratings.repeated$Rater==2],
                                       a3=ratings.repeated$Artifact[ratings.repeated$Rater==3])

r2 <- factor(raters_2_and_3_on_InitEDA$r2,levels=1:4)
r3 <- factor(raters_2_and_3_on_InitEDA$r3,levels=1:4)
t23 <- table(r2,r3)
InitEDA_23 = sum(diag(t23))/13

raters_1_and_3_on_InitEDA <- data.frame(r1=ratings.repeated$InitEDA[ratings.repeated$Rater==1],
                                       r3=ratings.repeated$InitEDA[ratings.repeated$Rater==3],
                                       a1=ratings.repeated$Artifact[ratings.repeated$Rater==1],
                                       a3=ratings.repeated$Artifact[ratings.repeated$Rater==3])

r1 <- factor(raters_1_and_3_on_InitEDA$r1,levels=1:4)
r3 <- factor(raters_1_and_3_on_InitEDA$r3,levels=1:4)
t13 <- table(r1,r3)
InitEDA_13 = sum(diag(t13))/13

# SelMeth
raters_1_and_2_on_SelMeth <- data.frame(r1=ratings.repeated$SelMeth[ratings.repeated$Rater==1],
                                       r2=ratings.repeated$SelMeth[ratings.repeated$Rater==2],
                                       a1=ratings.repeated$Artifact[ratings.repeated$Rater==1],
                                       a2=ratings.repeated$Artifact[ratings.repeated$Rater==2])

r1 <- factor(raters_1_and_2_on_SelMeth$r1,levels=1:4)
r2 <- factor(raters_1_and_2_on_SelMeth$r2,levels=1:4)
t12 <- table(r1,r2)
SelMeth_12 = sum(diag(t12))/13

raters_2_and_3_on_SelMeth <- data.frame(r2=ratings.repeated$SelMeth[ratings.repeated$Rater==2],
                                       r3=ratings.repeated$SelMeth[ratings.repeated$Rater==3],
                                       a2=ratings.repeated$Artifact[ratings.repeated$Rater==2],
                                       a3=ratings.repeated$Artifact[ratings.repeated$Rater==3])

r2 <- factor(raters_2_and_3_on_SelMeth$r2,levels=1:4)
r3 <- factor(raters_2_and_3_on_SelMeth$r3,levels=1:4)
t23 <- table(r2,r3)
SelMeth_23 = sum(diag(t23))/13

raters_1_and_3_on_SelMeth <- data.frame(r1=ratings.repeated$SelMeth[ratings.repeated$Rater==1],
                                       r3=ratings.repeated$SelMeth[ratings.repeated$Rater==3],
                                       a1=ratings.repeated$Artifact[ratings.repeated$Rater==1],
                                       a3=ratings.repeated$Artifact[ratings.repeated$Rater==3])

r1 <- factor(raters_1_and_3_on_SelMeth$r1,levels=1:4)
r3 <- factor(raters_1_and_3_on_SelMeth$r3,levels=1:4)
t13 <- table(r1,r3)
SelMeth_13 = sum(diag(t13))/13
```

```
# InterpRes
raters_1_and_2_on_InterpRes <-
  data.frame(r1=ratings.repeated$InterpRes[ratings.repeated$Rater==1],
                                        r2=ratings.repeated$InterpRes[ratings.repeated$Rater==2],
                                        a1=ratings.repeated$Artifact[ratings.repeated$Rater==1],
                                        a2=ratings.repeated$Artifact[ratings.repeated$Rater==2])

r1 <- factor(raters_1_and_2_on_InterpRes$r1,levels=1:4)
r2 <- factor(raters_1_and_2_on_InterpRes$r2,levels=1:4)
t12 <- table(r1,r2)
InterpRes_12 = sum(diag(t12))/13

raters_2_and_3_on_InterpRes <- data.frame(r2=ratings.repeated$InterpRes[ratings.repeated$Rater==2],
                                        r3=ratings.repeated$InterpRes[ratings.repeated$Rater==3],
                                        a2=ratings.repeated$Artifact[ratings.repeated$Rater==2],
                                        a3=ratings.repeated$Artifact[ratings.repeated$Rater==3])

r2 <- factor(raters_2_and_3_on_InterpRes$r2,levels=1:4)
r3 <- factor(raters_2_and_3_on_InterpRes$r3,levels=1:4)
t23 <- table(r2,r3)
InterpRes_23 = sum(diag(t23))/13

raters_1_and_3_on_InterpRes <- data.frame(r1=ratings.repeated$InterpRes[ratings.repeated$Rater==1],
                                        r3=ratings.repeated$InterpRes[ratings.repeated$Rater==3],
                                        a1=ratings.repeated$Artifact[ratings.repeated$Rater==1],
                                        a3=ratings.repeated$Artifact[ratings.repeated$Rater==3])

r1 <- factor(raters_1_and_3_on_InterpRes$r1,levels=1:4)
r3 <- factor(raters_1_and_3_on_InterpRes$r3,levels=1:4)
t13 <- table(r1,r3)
InterpRes_13 = sum(diag(t13))/13

# VisOrg
raters_1_and_2_on_VisOrg <- data.frame(r1=ratings.repeated$VisOrg[ratings.repeated$Rater==1],
                                        r2=ratings.repeated$VisOrg[ratings.repeated$Rater==2],
                                        a1=ratings.repeated$Artifact[ratings.repeated$Rater==1],
                                        a2=ratings.repeated$Artifact[ratings.repeated$Rater==2])

r1 <- factor(raters_1_and_2_on_VisOrg$r1,levels=1:4)
r2 <- factor(raters_1_and_2_on_VisOrg$r2,levels=1:4)
t12 <- table(r1,r2)
VisOrg_12 = sum(diag(t12))/13

raters_2_and_3_on_VisOrg <- data.frame(r2=ratings.repeated$VisOrg[ratings.repeated$Rater==2],
                                        r3=ratings.repeated$VisOrg[ratings.repeated$Rater==3],
                                        a2=ratings.repeated$Artifact[ratings.repeated$Rater==2],
                                        a3=ratings.repeated$Artifact[ratings.repeated$Rater==3])

r2 <- factor(raters_2_and_3_on_VisOrg$r2,levels=1:4)
r3 <- factor(raters_2_and_3_on_VisOrg$r3,levels=1:4)
t23 <- table(r2,r3)
VisOrg_23 = sum(diag(t23))/13
```

```
raters_1_and_3_on_VisOrg <- data.frame(r1=ratings.repeated$VisOrg[ratings.repeated$Rater==1],
                                       r3=ratings.repeated$VisOrg[ratings.repeated$Rater==3],
                                       a1=ratings.repeated$Artifact[ratings.repeated$Rater==1],
                                       a3=ratings.repeated$Artifact[ratings.repeated$Rater==3])

r1 <- factor(raters_1_and_3_on_VisOrg$r1,levels=1:4)
r3 <- factor(raters_1_and_3_on_VisOrg$r3,levels=1:4)
t13 <- table(r1,r3)
VisOrg_13 = sum(diag(t13))/13

# TxtOrg
raters_1_and_2_on_TxtOrg <- data.frame(r1=ratings.repeated$TxtOrg[ratings.repeated$Rater==1],
                                       r2=ratings.repeated$TxtOrg[ratings.repeated$Rater==2],
                                       a1=ratings.repeated$Artifact[ratings.repeated$Rater==1],
                                       a2=ratings.repeated$Artifact[ratings.repeated$Rater==2])

r1 <- factor(raters_1_and_2_on_TxtOrg$r1,levels=1:4)
r2 <- factor(raters_1_and_2_on_TxtOrg$r2,levels=1:4)
t12 <- table(r1,r2)
TxtOrg_12 = sum(diag(t12))/13

raters_2_and_3_on_TxtOrg <- data.frame(r2=ratings.repeated$TxtOrg[ratings.repeated$Rater==2],
                                       r3=ratings.repeated$TxtOrg[ratings.repeated$Rater==3],
                                       a2=ratings.repeated$Artifact[ratings.repeated$Rater==2],
                                       a3=ratings.repeated$Artifact[ratings.repeated$Rater==3])

r2 <- factor(raters_2_and_3_on_TxtOrg$r2,levels=1:4)
r3 <- factor(raters_2_and_3_on_TxtOrg$r3,levels=1:4)
t23 <- table(r2,r3)
TxtOrg_23 = sum(diag(t23))/13

raters_1_and_3_on_TxtOrg <- data.frame(r1=ratings.repeated$TxtOrg[ratings.repeated$Rater==1],
                                       r3=ratings.repeated$TxtOrg[ratings.repeated$Rater==3],
                                       a1=ratings.repeated$Artifact[ratings.repeated$Rater==1],
                                       a3=ratings.repeated$Artifact[ratings.repeated$Rater==3])

r1 <- factor(raters_1_and_3_on_TxtOrg$r1,levels=1:4)
r3 <- factor(raters_1_and_3_on_TxtOrg$r3,levels=1:4)
t13 <- table(r1,r3)
TxtOrg_13 = sum(diag(t13))/13
```

```
# Table of iccs and agreements
icc.df = data.frame("ICC.repeat" = c(CritDes.icc, InitEDA.icc, InterpRes.icc,
                                     RsrchQ.icc, SelMeth.icc, TxtOrg.icc, VisOrg.icc),
                    "a12" = c(CritDes_12, InitEDA_12, InterpRes_12, RsrchQ_12,
                              SelMeth_12, TxtOrg_12, VisOrg_12),
                    "a23" = c(CritDes_23, InitEDA_23, InterpRes_23, RsrchQ_23,
                              SelMeth_23, TxtOrg_23, VisOrg_23),
                    "a13" = c(CritDes_13, InitEDA_13, InterpRes_13, RsrchQ_13,
                              SelMeth_13, TxtOrg_13, VisOrg_13))
rownames(icc.df) = c("CritDes", "InitEDA", "InterpRes", "RsrchQ", "SelMeth", "TxtOrg",
                     "VisOrg")
kable(round(icc.df,2))
```

	ICC.repeat	a12	a23	a13
CritDes	0.57	0.54	0.69	0.62
InitEDA	0.49	0.69	0.85	0.54
InterpRes	0.23	0.62	0.62	0.54
RsrchQ	0.19	0.38	0.54	0.77
SelMeth	0.52	0.92	0.69	0.62
TxtOrg	0.14	0.69	0.54	0.62
VisOrg	0.59	0.54	0.77	0.77

Looking at the ICCs, we see that raters generally do not agree much on Research Question, Interpret Results, and Text Organization. They agree around half the time on Initial EDA and Select Methods, and a bit more than half the time on Critique Design and Visual Organization. Generally, there is not a rubric with an overly high degree of agreement between the raters according to these ICCs.

Looking at the percent agreements, it does not seem like there is one rater who disagrees with the others. The lowest percent agreement value is different for each the three rubrics with the lowest ICCs (i.e., the rubrics with the lowest general agreement). For Research Question, Raters 1 and 2 have the lowest agreement. For Interpret Results, Raters 1 and 3 have the lowest agreement. For Text Organization, Raters 2 and 3 have the lowest agreement. Therefore, it seems that who disagrees with whom depends on rubric, and there is not one rater who consistently disagrees with the others.

Now we will redo ICC calculations on the full dataset and see if they agree with the ICCs we calculated for the reduced data set.

```
# ICCs for full dataset

# RsrchQ
RsrchQ.ratings.full <- tall.ratings[tall.ratings$Rubric=="RsrchQ",]
RsrchQ.mod.full = lmer(as.numeric(Rating) ~ 1 + (1|Artifact), data=RsrchQ.ratings.full)
#summary(RsrchQ.mod.full)
RsrchQ.vcov.full = as.data.frame(VarCorr(RsrchQ.mod.full))$vcov
RsrchQ.icc.full = RsrchQ.vcov.full[1]/(RsrchQ.vcov.full[1] + RsrchQ.vcov.full[2])

# CritDes
CritDes.ratings.full <- tall.ratings[tall.ratings$Rubric=="CritDes",]
CritDes.mod.full = lmer(as.numeric(Rating) ~ 1 + (1|Artifact), data=CritDes.ratings.full)
#summary(CritDes.mod.full)
CritDes.vcov.full = as.data.frame(VarCorr(CritDes.mod.full))$vcov
CritDes.icc.full = CritDes.vcov.full[1]/(CritDes.vcov.full[1] + CritDes.vcov.full[2])

# InitEDA
InitEDA.ratings.full <- tall.ratings[tall.ratings$Rubric=="InitEDA",]
InitEDA.mod.full = lmer(as.numeric(Rating) ~ 1 + (1|Artifact), data=InitEDA.ratings.full)
#summary(InitEDA.mod.full)
InitEDA.vcov.full = as.data.frame(VarCorr(InitEDA.mod.full))$vcov
InitEDA.icc.full = InitEDA.vcov.full[1]/(InitEDA.vcov.full[1] + InitEDA.vcov.full[2])

# SelMeth
SelMeth.ratings.full <- tall.ratings[tall.ratings$Rubric=="SelMeth",]
SelMeth.mod.full = lmer(as.numeric(Rating) ~ 1 + (1|Artifact), data=SelMeth.ratings.full)
#summary(SelMeth.mod.full)
SelMeth.vcov.full = as.data.frame(VarCorr(SelMeth.mod.full))$vcov
SelMeth.icc.full = SelMeth.vcov.full[1]/(SelMeth.vcov.full[1] + SelMeth.vcov.full[2])

# InterpRes
InterpRes.ratings.full <- tall.ratings[tall.ratings$Rubric=="InterpRes",]
```

```
InterpRes.mod.full = lmer(as.numeric(Rating) ~ 1 + (1|Artifact), data=InterpRes.ratings.full)
#summary(InterpRes.mod.full)
InterpRes.vcov.full = as.data.frame(VarCorr(InterpRes.mod.full))$vcov
InterpRes.icc.full = InterpRes.vcov.full[1]/(InterpRes.vcov.full[1] + InterpRes.vcov.full[2])

# VisOrg
VisOrg.ratings.full <- tall.ratings[tall.ratings$Rubric=="VisOrg",]
VisOrg.mod.full = lmer(as.numeric(Rating) ~ 1 + (1|Artifact), data=VisOrg.ratings.full)
#summary(VisOrg.mod.full)
VisOrg.vcov.full = as.data.frame(VarCorr(VisOrg.mod.full))$vcov
VisOrg.icc.full = VisOrg.vcov.full[1]/(VisOrg.vcov.full[1] + VisOrg.vcov.full[2])

# TxtOrg
TxtOrg.ratings.full <- tall.ratings[tall.ratings$Rubric=="TxtOrg",]
TxtOrg.mod.full = lmer(as.numeric(Rating) ~ 1 + (1|Artifact), data=TxtOrg.ratings.full)
#summary(TxtOrg.mod.full)
TxtOrg.vcov.full = as.data.frame(VarCorr(TxtOrg.mod.full))$vcov
TxtOrg.icc.full = TxtOrg.vcov.full[1]/(TxtOrg.vcov.full[1] + TxtOrg.vcov.full[2])
```

```
# adding full dataset ICCs to the table
icc.df.full = data.frame("ICC.full" = c(CritDes.icc.full, InitEDA.icc.full, InterpRes.icc.full,
                                        RsrchQ.icc.full, SelMeth.icc.full, TxtOrg.icc.full,
                                        VisOrg.icc.full),
                         "ICC.repeat" = c(CritDes.icc, InitEDA.icc, InterpRes.icc,
                                          RsrchQ.icc, SelMeth.icc, TxtOrg.icc, VisOrg.icc),
                         "a12" = c(CritDes_12, InitEDA_12, InterpRes_12, RsrchQ_12,
                                   SelMeth_12, TxtOrg_12, VisOrg_12),
                         "a23" = c(CritDes_23, InitEDA_23, InterpRes_23, RsrchQ_23,
                                   SelMeth_23, TxtOrg_23, VisOrg_23),
                         "a13" = c(CritDes_13, InitEDA_13, InterpRes_13, RsrchQ_13,
                                   SelMeth_13, TxtOrg_13, VisOrg_13))
rownames(icc.df.full) = c("CritDes", "InitEDA", "InterpRes", "RsrchQ", "SelMeth", "TxtOrg",
                          "VisOrg")
kable(round(icc.df.full,2))
```

	ICC.full	ICC.repeat	a12	a23	a13
CritDes	0.67	0.57	0.54	0.69	0.62
InitEDA	0.69	0.49	0.69	0.85	0.54
InterpRes	0.22	0.23	0.62	0.62	0.54
RsrchQ	0.21	0.19	0.38	0.54	0.77
SelMeth	0.47	0.52	0.92	0.69	0.62
TxtOrg	0.19	0.14	0.69	0.54	0.62
VisOrg	0.66	0.59	0.54	0.77	0.77

These are generally pretty similar to the ICCs we found on the data with only repeated artifacts, though the ICCs of Critique Design and Initial EDA on the full dataset are a good bit higher. When looking at the full dataset, there are some rubrics where the raters generally agree on their scores (Critique Design, Initial EDA, and Visual Organization) and some where they have low agreement (Interpret Results, Research Question, and Text Organization).

Looking into how the Various Factors are Related to Ratings and if the Factors Interact

Now we will look into how the various factors of the experiment are related to the ratings. First, we will try adding fixed effects to the seven rubric-specific random intercept models using just the data from the artifacts that all three raters saw. We do this by finding a model using backwards elimination and comparing it to the intercept-only model using a likelihood ratio test for each rubric.

```
# loop to fit model using backward elimination and compare it to intercept-only
# model for each rubric
Rubric.names <- sort(unique(tall.repeated$Rubric))
model.formula.repeated <- as.list(rep(NA,7))
names(model.formula.repeated) <- Rubric.names
## There will be a lot of output from fitLMER.fnc() here... Sorry!
for (i in Rubric.names) {
  ## fit each base model
rubric.data <- tall.repeated[tall.repeated$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.repeated[[i]] <- formula(tmp_final) }
```

```
# print the final models
model.formula.repeated
```

```
## $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)
```

When looking at just the data with the artifacts repeated across the three raters, we see that adding fixed effects does not improve the fit of the model. Since we did not find that any fixed effects are significant, we are not going to try interactions or new random effects. Below are summaries of the seven models.

```
# model summaries
summary(lmer(as.numeric(Rating) ~ (1|Artifact),
             data= tall.repeated[tall.repeated$Rubric== "CritDes",],
             REML=FALSE))
```

```
## Linear mixed model fit by maximum likelihood  ['lmerMod']
## Formula: as.numeric(Rating) ~ (1 | Artifact)
##    Data: tall.repeated[tall.repeated$Rubric == "CritDes", ]
##
##      AIC      BIC   logLik deviance df.resid
##     79.4     84.4    -36.7     73.4       36
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -1.9549 -0.4174 -0.3226  0.5761  2.2084
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  Artifact (Intercept) 0.2794   0.5286
##  Residual             0.2308   0.4804
## Number of obs: 39, groups:  Artifact, 13
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)   1.7179     0.1656   10.38
```

```
ranef(lmer(as.numeric(Rating) ~ (1|Artifact),
           data= tall.repeated[tall.repeated$Rubric== "CritDes",],
           REML=FALSE))
```

```
## $Artifact
##     (Intercept)
## O1  -0.30158956
## O10 -0.30158956
## O11 -0.56296717
## O12 -0.04021194
## O13  0.22116567
## O2  -0.30158956
## O3   0.48254329
## O4   0.22116567
## O5   1.00529852
## O6  -0.56296717
## O7   0.48254329
## O8   0.22116567
## O9  -0.56296717
##
## with conditional variances for "Artifact"
```

```
summary(lmer(as.numeric(Rating) ~ (1|Artifact),
             data= tall.repeated[tall.repeated$Rubric== "InitEDA",],
             REML=FALSE))
```

```
## Linear mixed model fit by maximum likelihood  ['lmerMod']
## Formula: as.numeric(Rating) ~ (1 | Artifact)
##    Data: tall.repeated[tall.repeated$Rubric == "InitEDA", ]
##
##      AIC      BIC   logLik deviance df.resid
##     60.4     65.4    -27.2     54.4       36
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.2060 -0.2712 -0.2712  0.4340  1.6635
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  Artifact (Intercept) 0.1341   0.3662
##  Residual             0.1538   0.3922
## Number of obs: 39, groups:  Artifact, 13
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)   2.3846     0.1194   19.97
```

```
ranef(lmer(as.numeric(Rating) ~ (1|Artifact),
           data= tall.repeated[tall.repeated$Rubric== "InitEDA",],
           REML=FALSE))
```

```
## $Artifact
##     (Intercept)
## O1   0.44517185
## O10 -0.03709765
## O11 -0.27823241
## O12 -0.27823241
## O13  0.20403710
## O2  -0.03709765
## O3  -0.03709765
## O4   0.20403710
## O5  -0.27823241
## O6  -0.51936716
## O7   0.44517185
## O8  -0.27823241
## O9   0.44517185
##
## with conditional variances for "Artifact"
```

```
summary(lmer(as.numeric(Rating) ~ (1|Artifact),
             data= tall.repeated[tall.repeated$Rubric== "InterpRes",],
             REML=FALSE))
```

```
## Linear mixed model fit by maximum likelihood  ['lmerMod']
## Formula: as.numeric(Rating) ~ (1 | Artifact)
##    Data: tall.repeated[tall.repeated$Rubric == "InterpRes", ]
##
##      AIC      BIC   logLik deviance df.resid
##     74.6     79.6    -34.3     68.6       36
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
```

```
## -2.1666 -0.8210  0.5247  0.7933  2.6763
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  Artifact (Intercept) 0.07035  0.2652
##  Residual             0.28205  0.5311
## Number of obs: 39, groups:  Artifact, 13
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)   2.5128     0.1124   22.35
```

```
ranef(lmer(as.numeric(Rating) ~ (1|Artifact),
           data= tall.repeated[tall.repeated$Rubric== "InterpRes",],
           REML=FALSE))
```

```
## $Artifact
##     (Intercept)
## O1   0.06584615
## O10  0.06584615
## O11  0.20851282
## O12 -0.07682051
## O13 -0.21948718
## O2  -0.07682051
## O3   0.06584615
## O4   0.20851282
## O5   0.06584615
## O6  -0.21948718
## O7   0.06584615
## O8  -0.36215385
## O9   0.20851282
##
## with conditional variances for "Artifact"
```

```
summary(lmer(as.numeric(Rating) ~ (1|Artifact),
             data= tall.repeated[tall.repeated$Rubric== "RsrchQ",],
             REML=FALSE))
```

```
## Linear mixed model fit by maximum likelihood  ['lmerMod']
## Formula: as.numeric(Rating) ~ (1 | Artifact)
##    Data: tall.repeated[tall.repeated$Rubric == "RsrchQ", ]
##
##      AIC      BIC   logLik deviance df.resid
##     69.5     74.4    -31.7     63.5       36
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.3298 -0.5937 -0.3550  1.0229  1.6199
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  Artifact (Intercept) 0.04865  0.2206
##  Residual             0.25641  0.5064
## Number of obs: 39, groups:  Artifact, 13
##
## Fixed effects:
```

```
##             Estimate Std. Error t value
## (Intercept)   2.2821     0.1016   22.47
```

```
ranef(lmer(as.numeric(Rating) ~ (1|Artifact),
           data= tall.repeated[tall.repeated$Rubric== "RsrchQ",],
           REML=FALSE))
```

```
## $Artifact
##     (Intercept)
## O1   0.01860231
## O10  0.01860231
## O11  0.26043238
## O12 -0.10231272
## O13 -0.10231272
## O2  -0.10231272
## O3   0.13951734
## O4   0.01860231
## O5   0.13951734
## O6  -0.10231272
## O7   0.13951734
## O8  -0.22322775
## O9  -0.10231272
##
## with conditional variances for "Artifact"
```

```
summary(lmer(as.numeric(Rating) ~ (1|Artifact),
             data= tall.repeated[tall.repeated$Rubric== "SelMeth",],
             REML=FALSE))
```

```
## Linear mixed model fit by maximum likelihood  ['lmerMod']
## Formula: as.numeric(Rating) ~ (1 | Artifact)
##    Data: tall.repeated[tall.repeated$Rubric == "SelMeth", ]
##
##      AIC      BIC   logLik deviance df.resid
##     54.4     59.4    -24.2     48.4       36
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.13464 -0.03637 -0.03637  0.65821  2.06191
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  Artifact (Intercept) 0.1256   0.3544
##  Residual             0.1282   0.3581
## Number of obs: 39, groups:  Artifact, 13
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)   2.0513     0.1138   18.03
```

```
ranef(lmer(as.numeric(Rating) ~ (1|Artifact),
           data= tall.repeated[tall.repeated$Rubric== "SelMeth",],
           REML=FALSE))
```

```
## $Artifact
##     (Intercept)
```

```
## O1  -0.28695914
## O10 -0.03826122
## O11 -0.03826122
## O12 -0.03826122
## O13 -0.03826122
## O2  -0.53565705
## O3  -0.03826122
## O4   0.70783253
## O5   0.45913462
## O6  -0.03826122
## O7   0.21043670
## O8  -0.28695914
## O9  -0.03826122
##
## with conditional variances for "Artifact"
```

```
summary(lmer(as.numeric(Rating) ~ (1|Artifact),
             data= tall.repeated[tall.repeated$Rubric== "TxtOrg",],
             REML=FALSE))
```

```
## Linear mixed model fit by maximum likelihood  ['lmerMod']
## Formula: as.numeric(Rating) ~ (1 | Artifact)
##    Data: tall.repeated[tall.repeated$Rubric == "TxtOrg", ]
##
##      AIC      BIC   logLik deviance df.resid
##     78.1     83.1    -36.0     72.1       36
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.7264 -0.8339  0.4170  0.4170  2.4698
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  Artifact (Intercept) 0.04274  0.2067
##  Residual             0.33333  0.5774
## Number of obs: 39, groups:  Artifact, 13
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)   2.6667     0.1088   24.51
```

```
ranef(lmer(as.numeric(Rating) ~ (1|Artifact),
           data= tall.repeated[tall.repeated$Rubric== "TxtOrg",],
           REML=FALSE))
```

```
## $Artifact
##       (Intercept)
## O1  -9.259259e-02
## O10 -2.702641e-16
## O11  9.259259e-02
## O12 -9.259259e-02
## O13  9.259259e-02
## O2  -9.259259e-02
## O3   9.259259e-02
## O4   9.259259e-02
## O5  -2.702641e-16
```

```
## O6  -2.702641e-16
## O7   9.259259e-02
## O8  -2.777778e-01
## O9   9.259259e-02
##
## with conditional variances for "Artifact"
```

```
summary(lmer(as.numeric(Rating) ~ (1|Artifact),
             data= tall.repeated[tall.repeated$Rubric== "VisOrg",],
             REML=FALSE))
```

```
## Linear mixed model fit by maximum likelihood  ['lmerMod']
## Formula: as.numeric(Rating) ~ (1 | Artifact)
##    Data: tall.repeated[tall.repeated$Rubric == "VisOrg", ]
##
##      AIC      BIC   logLik deviance df.resid
##     64.5     69.5    -29.2     58.5       36
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -1.5015 -0.7420 -0.1453  0.3699  1.7261
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  Artifact (Intercept) 0.2025   0.4500
##  Residual             0.1538   0.3922
## Number of obs: 39, groups:  Artifact, 13
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)   2.2821     0.1397   16.33
```

```
ranef(lmer(as.numeric(Rating) ~ (1|Artifact),
           data= tall.repeated[tall.repeated$Rubric== "VisOrg",],
           REML=FALSE))
```

```
## $Artifact
##     (Intercept)
## O1  -0.22505646
## O10  0.04091936
## O11 -0.22505646
## O12 -0.22505646
## O13  0.30689518
## O2   0.04091936
## O3   0.30689518
## O4   0.04091936
## O5  -0.22505646
## O6   0.04091936
## O7   0.57287100
## O8  -1.02298393
## O9   0.57287100
##
## with conditional variances for "Artifact"
```

Next, we will try the same thing using the full dataset. Before we begin modelling, we must address the missing data that we discussed previously. We decided to delete these observations for the modelling to

ensure that all models are fit and compared using the exact same data.

```
# deleting missing observations
tall.ratings[which(is.na(tall.ratings$Rating == TRUE)),]
```

```
##       X Rater Artifact Repeated Semester Sex  Rubric Rating
## 161 161     2       45        0      S19   F CritDes   <NA>
## 684 684     1      100        0      F19   F  VisOrg   <NA>
```

```
tall.ratings[which(tall.ratings$Sex == "--"),]
```

```
##       X Rater Artifact Repeated Semester Sex    Rubric Rating
## 5     5     3        5        0      F19  --    RsrchQ      3
## 122 122     3        5        0      F19  --   CritDes      3
## 239 239     3        5        0      F19  --   InitEDA      3
## 356 356     3        5        0      F19  --   SelMeth      3
## 473 473     3        5        0      F19  -- InterpRes      3
## 590 590     3        5        0      F19  --    VisOrg      3
## 707 707     3        5        0      F19  --    TxtOrg      3
```

```
tall.nonmissing = tall.ratings[-c(5, 122, 239, 356, 473, 590, 707, 161, 684), ]
```

Just as before, we will try adding fixed effects by finding a model using backwards elimination and comparing it to the intercept-only model using a likelihood ratio test for each rubric.

```
# loop to fit model using backward elimination and compare it to intercept-only
# model for each rubric
model.formula.alldata <- as.list(rep(NA,7))
names(model.formula.alldata) <- Rubric.names
## There will be a lot of output from fitLMER.fnc() here... Sorry!
for (i in Rubric.names) {
  ## fit each base model
rubric.data <- tall.nonmissing[tall.nonmissing$Rubric==i,]
tmp <- lmer(as.numeric(Rating) ~ -1 + as.factor(Rater) +
Semester + Sex + (1|Artifact), data=rubric.data,REML=FALSE)
  ## do backwards elimination
tmp.back_elim <- fitLMER.fnc(tmp,set.REML.FALSE = TRUE,log.file.name = FALSE)
  ## check to see if the raters are significantly different from one another
tmp.single_intercept <- update(tmp.back_elim, . ~ . + 1 - as.factor(Rater))
pval <- anova(tmp.single_intercept,tmp.back_elim)$"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) }
```

```
# print the final models
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
```

We see that these results differ from what we found when looking just at the data with repeated artifacts. For InitEDA, RsrchQ, and TxtOrg, adding fixed effects does not improve the fit of the model. However, adding Rater and removing the intercept improves the fit for CritDes, InterpRes, and VisOrg, and adding Rater, adding Semester, and removing the intercept improves the fit for SelMeth. Thus, for some rubrics, it seems that Rater is related to Ratings, and for one rubric, Semester is related to Ratings.

Next, for the rubrics where we found that adding fixed effects improves the fit of the model, we will first check the t-statistics of the fixed effects to make sure they make sense, then try adding interactions and new random effects. We will start by doing this for the model for CritDes.

```
#CritDes

# checking that fixed effects really matter
CritDes.mod = lmer(formula(model.formula.alldata[["CritDes"]]),
                   data = tall.nonmissing[tall.nonmissing$Rubric=="CritDes",])
round(summary(CritDes.mod)$coef,2)
```

```
##                    Estimate Std. Error t value
## as.factor(Rater)1      1.69       0.12   13.98
## as.factor(Rater)2      2.11       0.12   17.34
## as.factor(Rater)3      1.89       0.12   15.51
```

```
# checking that rater really improves model (anova)
critdes_sing = lmer(as.numeric(Rating) ~ (1|Artifact),
                    data = tall.nonmissing[tall.nonmissing$Rubric=="CritDes",])
anova(critdes_sing, CritDes.mod)
```

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

```
## Data: tall.nonmissing[tall.nonmissing$Rubric == "CritDes", ]
## Models:
## critdes_sing: as.numeric(Rating) ~ (1 | Artifact)
## CritDes.mod: as.numeric(Rating) ~ as.factor(Rater) + (1 | Artifact) - 1
##              npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
## critdes_sing    3 277.68 285.91 -135.84   271.68
## CritDes.mod     5 273.62 287.35 -131.81   263.62 8.0535  2    0.01783 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# no fixed effect interactions to try since only Rater is involved

# trying new random effects
# critdes_random = lmer(as.numeric(Rating) ~ as.factor(Rater) + (1 | Artifact) +
#                         (as.factor(Rater)| Artifact) - 1,
#                       data = tall.nonmissing[tall.nonmissing$Rubric=="CritDes",])
# this model is not possible because there are
# more random effects than there are observations in the data set

# Final model summary
summary(CritDes.mod)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: as.numeric(Rating) ~ as.factor(Rater) + (1 | Artifact) - 1
##    Data: tall.nonmissing[tall.nonmissing$Rubric == "CritDes", ]
##
## REML criterion at convergence: 271
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -1.55495 -0.50027 -0.08228  0.64663  1.60935
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  Artifact (Intercept) 0.4349   0.6595
##  Residual             0.2473   0.4972
## Number of obs: 115, groups:  Artifact, 89
##
## Fixed effects:
##                   Estimate Std. Error t value
## as.factor(Rater)1   1.6863     0.1207   13.98
## as.factor(Rater)2   2.1129     0.1219   17.34
## as.factor(Rater)3   1.8908     0.1219   15.51
##
## Correlation of Fixed Effects:
##             a.(R)1 a.(R)2
## as.fctr(R)2 0.244
## as.fctr(R)3 0.244  0.246
```

Based on the t-values and the significant likelihood ratio test, it seems that including Rater in the model for CritDes really does matter. There are no fixed effect interactions to try since Rater is the only fixed effect included. Since there are more random effects than there are observations in the data set, the model with the random intercept of (as.factor(Rater)|Artifact) cannot be fit, so we also are not including any new random intercepts. Therefore, the final model for CritDes includes Rater as a fixed effect, but no additional fixed interactions or random effects.

Now we will follow the same process for the InterpRes rubric.

```
#InterpRes

# checking that fixed effects really matter
InterpRes.mod = lmer(formula(model.formula.alldata[["InterpRes"]]),
                     data = tall.nonmissing[tall.nonmissing$Rubric=="InterpRes",])
round(summary(InterpRes.mod)$coef,2)
```

```
##                   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
```

```
# checking that rater really improves model (anova)
interpres_sing = lmer(as.numeric(Rating) ~ (1|Artifact),
                      data = tall.nonmissing[tall.nonmissing$Rubric=="InterpRes",])
anova(interpres_sing, InterpRes.mod)
```

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

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

```
# no fixed effect interactions to try since only Rater is involved

# trying new random effects
# interpres_random = lmer(as.numeric(Rating) ~ as.factor(Rater) + (1 | Artifact) +
#                         (as.factor(Rater)| Artifact) - 1,
#                       data = tall.nonmissing[tall.nonmissing$Rubric=="InterpRes",])
# this model is not possible because there are
# more random effects than there are observations in the data set

# Final model summary
summary(InterpRes.mod)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: as.numeric(Rating) ~ as.factor(Rater) + (1 | Artifact) - 1
##    Data: tall.nonmissing[tall.nonmissing$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
```

Similar to what we found for the CritDes rubric, we see that based on the t-statistics and likelihood ratio test p-value, including Rater in the model for InterpRes matters. There are no fixed effect interactions to try since Rater is the only fixed effect included. Since there are more random effects than there are observations in the data set, the model with the random intercept of (as.factor(Rater)|Artifact) cannot be fit, so we also are not including any new random intercepts. Therefore, the final model for InterpRes includes Rater as a fixed effect, but no additional fixed interactions or random effects.

Now we will follow the same process for the VisOrg rubric.

```
# VisOrg

# checking that fixed effects really matter
VisOrg.mod = lmer(formula(model.formula.alldata[["VisOrg"]]),
                  data = tall.nonmissing[tall.nonmissing$Rubric=="VisOrg",])
round(summary(VisOrg.mod)$coef,2)
```

```
##                    Estimate Std. Error t value
## as.factor(Rater)1     2.38        0.1   24.62
## as.factor(Rater)2     2.65        0.1   27.70
## as.factor(Rater)3     2.28        0.1   23.64
```

```
# checking that rater really improves model (anova)
visorg_sing = lmer(as.numeric(Rating) ~ (1|Artifact),
                  data = tall.nonmissing[tall.nonmissing$Rubric=="VisOrg",])
anova(visorg_sing, VisOrg.mod)
```

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

```
## Data: tall.nonmissing[tall.nonmissing$Rubric == "VisOrg", ]
## Models:
## visorg_sing: as.numeric(Rating) ~ (1 | Artifact)
## VisOrg.mod: as.numeric(Rating) ~ as.factor(Rater) + (1 | Artifact) - 1
##             npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
## visorg_sing    3 227.21 235.44 -110.60   221.21
## VisOrg.mod     5 220.82 234.54 -105.41   210.82 10.392  2   0.005539 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# no fixed effect interactions to try since only Rater is involved

# trying new random effects
# visorg_random = lmer(as.numeric(Rating) ~ as.factor(Rater) + (1 | Artifact) +
#                        (as.factor(Rater)| Artifact) - 1,
#                      data = tall.nonmissing[tall.nonmissing$Rubric=="VisOrg",])
# this model is not possible because there are
# more random effects than there are observations in the data set

# Final model summary
summary(VisOrg.mod)
```

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

```
## REML criterion at convergence: 219.6
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -1.5004 -0.3365 -0.2483  0.3841  1.8552
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  Artifact (Intercept) 0.2907   0.5392
##  Residual             0.1467   0.3830
## Number of obs: 115, groups:  Artifact, 89
##
## Fixed effects:
##                   Estimate Std. Error t value
## as.factor(Rater)1  2.37794    0.09658   24.62
## as.factor(Rater)2  2.64891    0.09564   27.70
## as.factor(Rater)3  2.28355    0.09658   23.64
##
## Correlation of Fixed Effects:
##             a.(R)1 a.(R)2
## as.fctr(R)2 0.263
## as.fctr(R)3 0.265  0.263
```

Similar to what we found for the CritDes and InterpRes rubrics, we see that based on the t-statistics and likelihood ratio test p-value, including Rater in the model for VisOrg matters. There are no fixed effect interactions to try since Rater is the only fixed effect included. Since there are more random effects than there are observations in the data set, the model with the random intercept of (as.factor(Rater)|Artifact) cannot be fit, so we also are not including any new random intercepts. Therefore, the final model for VisOrg includes Rater as a fixed effect, but no additional fixed interactions or random effects.

Now we will follow the same process for the SelMeth rubric.

```
# SelMeth

# checking that fixed effects really matter
SelMeth.mod = lmer(formula(model.formula.alldata[["SelMeth"]]),
                   data = tall.nonmissing[tall.nonmissing$Rubric=="SelMeth",])
round(summary(SelMeth.mod)$coef,2)
```

```
##                   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
```

```
# checking that rater really improves model (anova)
selmeth_sing = lmer(as.numeric(Rating) ~ (1|Artifact) + Semester,
                    data = tall.nonmissing[tall.nonmissing$Rubric=="SelMeth",])
anova(selmeth_sing, SelMeth.mod)
```

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

```
## Data: tall.nonmissing[tall.nonmissing$Rubric == "SelMeth", ]
## Models:
## selmeth_sing: as.numeric(Rating) ~ (1 | Artifact) + Semester
## SelMeth.mod: as.numeric(Rating) ~ as.factor(Rater) + Semester + (1 | Artifact) - 1
##              npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
```

```
## selmeth_sing    4 145.07 156.08 -68.534   137.07
## SelMeth.mod     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
```

```
# trying interactions
selmeth_interact = lmer(as.numeric(Rating) ~ as.factor(Rater) + (1 | Artifact) +
                          as.factor(Rater):Semester,
                        data = tall.nonmissing[tall.nonmissing$Rubric=="SelMeth",])
anova(SelMeth.mod, selmeth_interact)
```

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

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

```
# trying new random effects
# selmeth_random = lmer(as.numeric(Rating) ~ as.factor(Rater) + Semester + (1 | Artifact) +
#                         (as.factor(Rater)| Artifact) - 1,
#                       data = tall.nonmissing[tall.nonmissing$Rubric=="SelMeth",])
# selmeth_random2 = lmer(as.numeric(Rating) ~ as.factor(Rater) + (1 | Artifact) +
#                         (Semester| Artifact) - 1,
#                       data = tall.nonmissing[tall.nonmissing$Rubric=="SelMeth",])
# these models are not possible because there are
# more random effects than there are observations in the data set

# Final model summary
summary(SelMeth.mod)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: as.numeric(Rating) ~ as.factor(Rater) + Semester + (1 | Artifact) -
##     1
##    Data: tall.nonmissing[tall.nonmissing$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
```

Similar to what we found for the previous rubrics, we see that based on the t-statistics and likelihood ratio test p-value, including Rater in the model for SelMeth matters. We also see that including Semester in the model matters according to the t-statistic. We tried the interaction of Rater and Semester, but it was not significant. Since there are more random effects than there are observations in the data set, the model with the random intercept of (as.factor(Rater)|Artifact) and the model with the random intercept of (Semester|Artifact) cannot be fit, so we also are not including any new random intercepts. Therefore, the final model for SelMeth includes Rater and Semester as fixed effects, but no additional fixed interactions or random effects.

Below, we print the summaries and random effect coefficients of the final seven models.

```
RsrchQ.mod = lmer(formula(model.formula.alldata[["RsrchQ"]]),
                  data = tall.nonmissing[tall.nonmissing$Rubric=="RsrchQ",])
InitEDA.mod = lmer(formula(model.formula.alldata[["InitEDA"]]),
                  data = tall.nonmissing[tall.nonmissing$Rubric=="InitEDA",])
TxtOrg.mod = lmer(formula(model.formula.alldata[["TxtOrg"]]),
                  data = tall.nonmissing[tall.nonmissing$Rubric=="TxtOrg",])
summary(RsrchQ.mod)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: as.numeric(Rating) ~ (1 | Artifact)
##    Data: tall.nonmissing[tall.nonmissing$Rubric == "RsrchQ", ]
##
## REML criterion at convergence: 209.1
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.2694 -0.5285 -0.3736  0.9743  2.4770
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  Artifact (Intercept) 0.07276  0.2697
##  Residual             0.27825  0.5275
## Number of obs: 116, groups:  Artifact, 90
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)  2.35169    0.05794   40.59
```

```
ranef(RsrchQ.mod)
```

```
## $Artifact
##      (Intercept)
## 100 -0.072903664
## 101 -0.280199221
## 102 -0.280199221
## 103 -0.072903664
## 104 -0.072903664
## 105 -0.072903664
```

```
## 106  0.134391893
## 107  0.134391893
## 111 -0.072903664
## 112  0.134391893
## 113 -0.072903664
## 114 -0.072903664
## 115  0.134391893
## 116 -0.072903664
## 117 -0.072903664
## 118 -0.072903664
## 13  -0.072903664
## 15  -0.072903664
## 16  -0.072903664
## 17   0.134391893
## 21   0.134391893
## 22   0.134391893
## 23  -0.072903664
## 24  -0.072903664
## 25  -0.072903664
## 26  -0.072903664
## 27  -0.072903664
## 28  -0.280199221
## 32   0.134391893
## 33  -0.072903664
## 34  -0.072903664
## 35  -0.072903664
## 36  -0.072903664
## 37  -0.072903664
## 38  -0.072903664
## 39   0.134391893
## 40  -0.072903664
## 45  -0.072903664
## 46  -0.072903664
## 47   0.134391893
## 48   0.134391893
## 49   0.134391893
## 53   0.134391893
## 54  -0.280199221
## 55   0.134391893
## 56  -0.072903664
## 57  -0.072903664
## 6   -0.072903664
## 61  -0.072903664
## 62   0.134391893
## 63   0.134391893
## 64  -0.072903664
## 65   0.134391893
## 66   0.134391893
## 67   0.134391893
## 68   0.134391893
## 7   -0.072903664
## 72  -0.072903664
## 73  -0.072903664
## 74  -0.072903664
```

```
## 75  -0.072903664
## 76  -0.072903664
## 77  -0.072903664
## 78   0.134391893
## 79   0.134391893
## 8   -0.072903664
## 84   0.134391893
## 85   0.341687450
## 86   0.134391893
## 87   0.134391893
## 88   0.134391893
## 9    0.134391893
## 92   0.134391893
## 93   0.134391893
## 94   0.134391893
## 95  -0.072903664
## 96   0.134391893
## O1  -0.008069777
## O10 -0.008069777
## O11  0.285012168
## O12 -0.154610749
## O13 -0.154610749
## O2  -0.154610749
## O3   0.138471195
## O4  -0.008069777
## O5   0.138471195
## O6  -0.154610749
## O7   0.138471195
## O8  -0.301151721
## O9  -0.154610749
##
## with conditional variances for "Artifact"
```

```
summary(CritDes.mod)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: as.numeric(Rating) ~ as.factor(Rater) + (1 | Artifact) - 1
##    Data: tall.nonmissing[tall.nonmissing$Rubric == "CritDes", ]
##
## REML criterion at convergence: 271
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -1.55495 -0.50027 -0.08228  0.64663  1.60935
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  Artifact (Intercept) 0.4349   0.6595
##  Residual             0.2473   0.4972
## Number of obs: 115, groups:  Artifact, 89
##
## Fixed effects:
##                   Estimate Std. Error t value
## as.factor(Rater)1   1.6863     0.1207   13.98
## as.factor(Rater)2   2.1129     0.1219   17.34
```

```
## as.factor(Rater)3   1.8908     0.1219   15.51
##
## Correlation of Fixed Effects:
##             a.(R)1 a.(R)2
## as.fctr(R)2 0.244
## as.fctr(R)3 0.244  0.246
```

```
ranef(CritDes.mod)
```

```
## $Artifact
##     (Intercept)
## 100  0.83753019
## 101 -0.43756481
## 102 -0.43756481
## 103  0.19998269
## 104 -0.43756481
## 105 -0.43756481
## 106  0.19998269
## 107 -0.43756481
## 111 -0.43756481
## 112 -0.43756481
## 113 -0.43756481
## 114 -0.43756481
## 115 -0.43756481
## 116 -0.43756481
## 117 -0.43756481
## 118 -0.43756481
## 13   0.06962462
## 15   0.70717212
## 16   0.70717212
## 17   0.06962462
## 21   0.70717212
## 22   0.70717212
## 23  -0.56792288
## 24   0.06962462
## 25   0.70717212
## 26  -0.56792288
## 27   0.06962462
## 28  -0.56792288
## 32   0.70717212
## 33   0.06962462
## 34   0.70717212
## 35  -0.56792288
## 36   0.06962462
## 37   0.70717212
## 38   0.06962462
## 39  -0.56792288
## 40   0.06962462
## 46  -0.07196897
## 47   0.56557853
## 48   0.56557853
## 49  -0.70951647
## 53   1.20312602
## 54  -0.70951647
## 55  -0.07196897
```

```
## 56   0.56557853
## 57  -0.70951647
## 6   -0.56792288
## 61  -0.07196897
## 62   1.20312602
## 63   0.56557853
## 64   0.56557853
## 65   0.56557853
## 66   0.56557853
## 67  -0.70951647
## 68   0.56557853
## 7   -0.56792288
## 72  -0.07196897
## 73  -0.70951647
## 74  -0.70951647
## 75  -0.07196897
## 76  -0.07196897
## 77  -0.07196897
## 78   0.56557853
## 79  -0.07196897
## 8   -0.56792288
## 84   0.19998269
## 85   0.83753019
## 86   0.19998269
## 87   0.19998269
## 88   0.83753019
## 9   -0.56792288
## 92  -0.43756481
## 93  -0.43756481
## 94   0.83753019
## 95   0.19998269
## 96   0.19998269
## O1  -0.47358754
## O10 -0.47358754
## O11 -0.75381650
## O12 -0.19335859
## O13  0.08687037
## O2  -0.47358754
## O3   0.36709933
## O4   0.08687037
## O5   0.92755724
## O6  -0.75381650
## O7   0.36709933
## O8   0.08687037
## O9  -0.75381650
##
## with conditional variances for "Artifact"
```

```
summary(InitEDA.mod)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: as.numeric(Rating) ~ (1 | Artifact)
##    Data: tall.nonmissing[tall.nonmissing$Rubric == "InitEDA", ]
##
## REML criterion at convergence: 239
```

```
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -1.8889 -0.3391 -0.1427  0.4276  1.6035
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  Artifact (Intercept) 0.3651   0.6042
##  Residual             0.1655   0.4068
## Number of obs: 116, groups:  Artifact, 90
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)  2.44226    0.07537    32.4
```

```
ranef(InitEDA.mod)
```

```
## $Artifact
##     (Intercept)
## 100 -0.30430226
## 101  0.38376223
## 102 -0.30430226
## 103  0.38376223
## 104  0.38376223
## 105 -0.30430226
## 106 -0.99236674
## 107 -0.30430226
## 111 -0.30430226
## 112  0.38376223
## 113 -0.99236674
## 114  0.38376223
## 115  0.38376223
## 116 -0.30430226
## 117 -0.30430226
## 118 -0.30430226
## 13  -0.99236674
## 15   0.38376223
## 16   1.07182672
## 17  -0.30430226
## 21   1.07182672
## 22   0.38376223
## 23  -0.99236674
## 24  -0.30430226
## 25  -0.30430226
## 26  -0.30430226
## 27   0.38376223
## 28  -0.99236674
## 32   0.38376223
## 33   0.38376223
## 34  -0.30430226
## 35  -0.30430226
## 36  -0.30430226
## 37  -0.30430226
## 38  -0.30430226
## 39   0.38376223
```

```
## 40   0.38376223
## 45  -0.30430226
## 46   0.38376223
## 47  -0.30430226
## 48   1.07182672
## 49   0.38376223
## 53   0.38376223
## 54   0.38376223
## 55  -0.30430226
## 56  -0.30430226
## 57  -0.30430226
## 6   -0.30430226
## 61   0.38376223
## 62   1.07182672
## 63   0.38376223
## 64  -0.30430226
## 65  -0.30430226
## 66   1.07182672
## 67   0.38376223
## 68  -0.30430226
## 7    0.38376223
## 72   0.38376223
## 73  -0.99236674
## 74   0.38376223
## 75   0.38376223
## 76  -0.30430226
## 77  -0.30430226
## 78   0.38376223
## 79   0.38376223
## 8   -0.30430226
## 84  -0.30430226
## 85   0.38376223
## 86  -0.30430226
## 87  -0.99236674
## 88   0.38376223
## 9   -0.30430226
## 92  -0.30430226
## 93  -0.30430226
## 94   1.07182672
## 95   0.38376223
## 96   0.38376223
## O1   0.48452197
## O10 -0.09462545
## O11 -0.38419916
## O12 -0.38419916
## O13  0.19494826
## O2  -0.09462545
## O3  -0.09462545
## O4   0.19494826
## O5  -0.38419916
## O6  -0.67377288
## O7   0.48452197
## O8  -0.38419916
## O9   0.48452197
```

```
##
## with conditional variances for "Artifact"
```

```
summary(SelMeth.mod)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: as.numeric(Rating) ~ as.factor(Rater) + Semester + (1 | Artifact) -
##     1
##    Data: tall.nonmissing[tall.nonmissing$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
```

```
ranef(SelMeth.mod)
```

```
## $Artifact
##     (Intercept)
## 100  0.33946601
## 101  0.04901108
## 102 -0.11338077
## 103  0.33946601
## 104 -0.11338077
## 105 -0.11338077
## 106 -0.11338077
## 107 -0.11338077
## 111  0.04901108
## 112 -0.11338077
## 113  0.04901108
## 114  0.04901108
## 115  0.04901108
## 116 -0.11338077
## 117 -0.11338077
## 118 -0.11338077
```

```
## 13  -0.46786343
## 15  -0.01501665
## 16  -0.01501665
## 17  -0.30547158
## 21   0.14737520
## 22  -0.01501665
## 23  -0.30547158
## 24  -0.01501665
## 25  -0.30547158
## 26   0.43783013
## 27  -0.01501665
## 28  -0.30547158
## 32  -0.01501665
## 33   0.43783013
## 34   0.43783013
## 35  -0.01501665
## 36  -0.01501665
## 37  -0.01501665
## 38  -0.01501665
## 39   0.14737520
## 40  -0.01501665
## 45   0.05980677
## 46   0.05980677
## 47  -0.39304001
## 48   0.35026170
## 49  -0.10258508
## 53   0.35026170
## 54  -0.10258508
## 55  -0.10258508
## 56  -0.10258508
## 57  -0.10258508
## 6   -0.01501665
## 61  -0.10258508
## 62   0.05980677
## 63   0.05980677
## 64  -0.10258508
## 65  -0.10258508
## 66   0.05980677
## 67  -0.10258508
## 68   0.05980677
## 7   -0.01501665
## 72   0.05980677
## 73  -0.10258508
## 74   0.35026170
## 75  -0.10258508
## 76  -0.10258508
## 77  -0.10258508
## 78   0.35026170
## 79   0.35026170
## 8    0.14737520
## 84   0.04901108
## 85  -0.11338077
## 86   0.04901108
## 87  -0.11338077
```

```
## 88   0.04901108
## 9    0.14737520
## 92  -0.11338077
## 93   0.04901108
## 94  -0.11338077
## 95   0.33946601
## 96  -0.11338077
## O1  -0.35883486
## O10 -0.12120653
## O11  0.13443559
## O12 -0.12120653
## O13 -0.12120653
## O2  -0.59646319
## O3  -0.12120653
## O4   0.59167847
## O5   0.35405014
## O6  -0.12120653
## O7   0.11642180
## O8  -0.10319274
## O9   0.13443559
##
## with conditional variances for "Artifact"
```

```
summary(InterpRes.mod)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: as.numeric(Rating) ~ as.factor(Rater) + (1 | Artifact) - 1
##    Data: tall.nonmissing[tall.nonmissing$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
```

```
ranef(InterpRes.mod)
```

```
## $Artifact
```

```
##     (Intercept)
## 100  0.05848973
## 101 -0.13925357
## 102  0.05848973
## 103  0.05848973
## 104  0.05848973
## 105 -0.13925357
## 106 -0.13925357
## 107  0.05848973
## 111 -0.13925357
## 112  0.05848973
## 113  0.05848973
## 114  0.05848973
## 115  0.05848973
## 116 -0.13925357
## 117  0.05848973
## 118  0.05848973
## 13  -0.22526567
## 15  -0.02752238
## 16   0.17022092
## 17  -0.02752238
## 21   0.17022092
## 22   0.17022092
## 23  -0.22526567
## 24  -0.02752238
## 25  -0.22526567
## 26  -0.02752238
## 27  -0.02752238
## 28  -0.22526567
## 32   0.17022092
## 33  -0.02752238
## 34   0.17022092
## 35  -0.02752238
## 36  -0.02752238
## 37  -0.02752238
## 38  -0.02752238
## 39  -0.02752238
## 40  -0.02752238
## 45  -0.11582665
## 46  -0.11582665
## 47  -0.11582665
## 48   0.08191665
## 49   0.08191665
## 53   0.08191665
## 54  -0.11582665
## 55   0.08191665
## 56  -0.11582665
## 57  -0.11582665
## 6   -0.02752238
## 61  -0.11582665
## 62   0.08191665
## 63   0.08191665
## 64   0.08191665
## 65  -0.31356994
```

```
## 66   0.08191665
## 67   0.08191665
## 68   0.08191665
## 7   -0.02752238
## 72   0.08191665
## 73  -0.11582665
## 74   0.08191665
## 75   0.08191665
## 76  -0.11582665
## 77   0.08191665
## 78   0.08191665
## 79   0.08191665
## 8   -0.02752238
## 84   0.05848973
## 85   0.05848973
## 86   0.05848973
## 87  -0.13925357
## 88   0.05848973
## 9   -0.02752238
## 92   0.05848973
## 93   0.05848973
## 94   0.05848973
## 95   0.05848973
## 96   0.05848973
## O1   0.08089221
## O10  0.08089221
## O11  0.22259425
## O12 -0.06080983
## O13 -0.20251187
## O2  -0.06080983
## O3   0.08089221
## O4   0.22259425
## O5   0.08089221
## O6  -0.20251187
## O7   0.08089221
## O8  -0.34421391
## O9   0.22259425
##
## with conditional variances for "Artifact"
```

```
summary(VisOrg.mod)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: as.numeric(Rating) ~ as.factor(Rater) + (1 | Artifact) - 1
##    Data: tall.nonmissing[tall.nonmissing$Rubric == "VisOrg", ]
##
## REML criterion at convergence: 219.6
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -1.5004 -0.3365 -0.2483  0.3841  1.8552
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  Artifact (Intercept) 0.2907   0.5392
```

```
##  Residual             0.1467   0.3830
## Number of obs: 115, groups:  Artifact, 89
##
## Fixed effects:
##                    Estimate Std. Error t value
## as.factor(Rater)1  2.37794    0.09658   24.62
## as.factor(Rater)2  2.64891    0.09564   27.70
## as.factor(Rater)3  2.28355    0.09658   23.64
##
## Correlation of Fixed Effects:
##             a.(R)1 a.(R)2
## as.fctr(R)2 0.263
## as.fctr(R)3 0.265  0.263
```

```
ranef(VisOrg.mod)
```

```
## $Artifact
##     (Intercept)
## 101  0.41341681
## 102 -0.25117695
## 103 -0.25117695
## 104 -0.25117695
## 105 -0.25117695
## 106 -0.25117695
## 107 -0.25117695
## 111 -0.25117695
## 112  0.41341681
## 113 -0.25117695
## 114 -0.25117695
## 115  0.41341681
## 116  0.41341681
## 117  1.07801056
## 118  0.41341681
## 13  -0.85303625
## 15   1.14074503
## 16   0.47615127
## 17  -0.18844249
## 21   0.47615127
## 22  -0.18844249
## 23  -0.85303625
## 24  -0.18844249
## 25  -0.18844249
## 26  -0.18844249
## 27  -0.18844249
## 28  -0.85303625
## 32   0.47615127
## 33  -0.18844249
## 34  -0.18844249
## 35   0.47615127
## 36  -0.18844249
## 37  -0.18844249
## 38   0.47615127
## 39  -0.18844249
## 40  -0.18844249
## 45  -0.43126354
```

```
## 46  -0.43126354
## 47  -1.09585730
## 48  -0.43126354
## 49   0.89792397
## 53   0.23333021
## 54  -0.43126354
## 55   0.23333021
## 56   0.23333021
## 57   0.23333021
## 6   -0.18844249
## 61   0.23333021
## 62   0.89792397
## 63   0.23333021
## 64   0.23333021
## 65   0.23333021
## 66   0.23333021
## 67   0.23333021
## 68   0.23333021
## 7   -0.18844249
## 72   0.23333021
## 73  -0.43126354
## 74   0.23333021
## 75  -0.43126354
## 76  -0.43126354
## 77   0.23333021
## 78   0.23333021
## 79   0.23333021
## 8   -0.18844249
## 84   0.41341681
## 85   0.41341681
## 86  -0.25117695
## 87  -0.25117695
## 88   1.07801056
## 9   -0.18844249
## 92  -0.25117695
## 93  -0.25117695
## 94   0.41341681
## 95  -0.25117695
## 96   0.41341681
## O1  -0.37389990
## O10 -0.08856703
## O11 -0.37389990
## O12 -0.37389990
## O13  0.19676584
## O2  -0.08856703
## O3   0.19676584
## O4  -0.08856703
## O5  -0.37389990
## O6  -0.08856703
## O7   0.48209871
## O8  -1.22989851
## O9   0.48209871
##
## with conditional variances for "Artifact"
```

```
summary(TxtOrg.mod)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: as.numeric(Rating) ~ (1 | Artifact)
##    Data: tall.nonmissing[tall.nonmissing$Rubric == "TxtOrg", ]
##
## REML criterion at convergence: 247.5
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.3557 -0.7550  0.3834  0.5302  2.4132
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  Artifact (Intercept) 0.09371  0.3061
##  Residual             0.39573  0.6291
## Number of obs: 116, groups:  Artifact, 90
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)  2.58745    0.06821   37.93
```

```
ranef(TxtOrg.mod)
```

```
## $Artifact
##     (Intercept)
## 100 -0.11247941
## 101  0.07899020
## 102 -0.30394902
## 103  0.07899020
## 104  0.07899020
## 105 -0.11247941
## 106  0.07899020
## 107 -0.11247941
## 111 -0.11247941
## 112  0.07899020
## 113 -0.11247941
## 114  0.07899020
## 115  0.07899020
## 116  0.07899020
## 117  0.07899020
## 118  0.07899020
## 13  -0.30394902
## 15   0.07899020
## 16   0.27045981
## 17  -0.11247941
## 21   0.27045981
## 22   0.07899020
## 23  -0.30394902
## 24   0.07899020
## 25  -0.11247941
## 26  -0.11247941
## 27   0.07899020
## 28  -0.30394902
```

```
## 32   0.07899020
## 33  -0.11247941
## 34  -0.11247941
## 35  -0.11247941
## 36   0.07899020
## 37   0.07899020
## 38  -0.11247941
## 39  -0.11247941
## 40  -0.11247941
## 45   0.07899020
## 46  -0.11247941
## 47  -0.30394902
## 48   0.27045981
## 49   0.07899020
## 53   0.07899020
## 54   0.07899020
## 55  -0.11247941
## 56  -0.11247941
## 57   0.07899020
## 6   -0.11247941
## 61   0.27045981
## 62   0.07899020
## 63   0.07899020
## 64   0.07899020
## 65   0.07899020
## 66  -0.11247941
## 67  -0.30394902
## 68   0.07899020
## 7   -0.11247941
## 72  -0.11247941
## 73   0.07899020
## 74  -0.11247941
## 75  -0.11247941
## 76  -0.11247941
## 77  -0.11247941
## 78   0.07899020
## 79   0.07899020
## 8   -0.11247941
## 84   0.07899020
## 85   0.07899020
## 86   0.07899020
## 87   0.07899020
## 88   0.07899020
## 9   -0.11247941
## 92   0.07899020
## 93   0.27045981
## 94   0.07899020
## 95   0.07899020
## 96   0.07899020
## O1  -0.10554956
## O10  0.03290165
## O11  0.17135286
## O12 -0.10554956
## O13  0.17135286
```

```
## O2  -0.10554956
## O3   0.17135286
## O4   0.17135286
## O5   0.03290165
## O6   0.03290165
## O7   0.17135286
## O8  -0.38245198
## O9   0.17135286
##
## with conditional variances for "Artifact"
```

Generally, we see that there are differences between the models for the different rubrics. Some rubrics include Rater as a fixed effects while others do not include any fixed effects. Therefore, we will now try modelling in a way that will allow us to explore interactions with Rubric directly. We will try adding fixed effects, interactions, and new random effects to a "combined" model of Rating ~ 1 + (0 + Rubric|Artifact), using all the data.

```
# Start with intercept-only model
comb.mod1 =  lmer(as.numeric(Rating) ~ 1 + (0 + Rubric|Artifact), data = tall.nonmissing)

# Adding fixed effects
comb.mod2 <- update(comb.mod1, . ~ . + as.factor(Rater) + Semester + Sex + Repeated + Rubric)

# backwards elimination
comb.mod3 <- fitLMER.fnc(comb.mod2, log.file.name = FALSE)
summary(comb.mod3)

# try adding interactions based on fixed effects from model 3
comb.mod4 <- update(comb.mod3, . ~ . + as.factor(Rater)*Semester*Rubric)

# model 4 fails to converge, so trying different optimizer for adding interactions
ss <- getME(comb.mod4, c("theta","fixef"))
comb.mod5 <- update(comb.mod4,start=ss,
control=lmerControl(optimizer="bobyqa", optCtrl=list(maxfun=2e5)))

# elimination for interactions
comb.mod6 = fitLMER.fnc(comb.mod5, log.file.name = FALSE)

# compare the models
anova(comb.mod3, comb.mod5, comb.mod6)
```

Based on AIC and the likelihood ratio test, the best model we have so far is the one with Rater, Semester, Rubric, and the interaction of Rater and Rubric as fixed effects. Now, based on this model, we will consider adding random effects. We will try adding random effects for Rater, Semester, and the interaction of Rater and Rubric.

```
# considering random effects

# trying (0 + Rater | Artifact)
comb.mod7 <- lmer(as.numeric(Rating) ~ (0 + Rubric | Artifact) +
(0 + as.factor(Rater) | Artifact) + as.factor(Rater) +
Semester + Rubric + as.factor(Rater):Rubric, data=tall.nonmissing)
```

```
## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv, :
## Model failed to converge with max|grad| = 0.00347545 (tol = 0.002, component 1)
```

```
anova(comb.mod6, comb.mod7)
```

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

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

```
## Data: tall.nonmissing
## Models:
## comb.mod6: as.numeric(Rating) ~ (0 + Rubric | Artifact) + as.factor(Rater) + Semester + Rubric + as.
## comb.mod7: as.numeric(Rating) ~ (0 + Rubric | Artifact) + (0 + as.factor(Rater) | Artifact) + as.fac
##           npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
## comb.mod6   51 1454.5 1694.1 -676.26   1352.5
## comb.mod7   57 1415.9 1683.6 -650.94   1301.9 50.647  6  3.487e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# trying (0 + Semester | Artifact)
comb.mod8 <- lmer(as.numeric(Rating) ~ (0 + Rubric | Artifact) +
(0 + Semester | Artifact) + as.factor(Rater) +
Semester + Rubric + as.factor(Rater):Rubric, data=tall.nonmissing)
```

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

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

```
anova(comb.mod6, comb.mod8)
```

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

```
## Data: tall.nonmissing
## Models:
## comb.mod6: as.numeric(Rating) ~ (0 + Rubric | Artifact) + as.factor(Rater) + Semester + Rubric + as.
## comb.mod8: as.numeric(Rating) ~ (0 + Rubric | Artifact) + (0 + Semester | Artifact) + as.factor(Rate
##           npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
## comb.mod6   51 1454.5 1694.1 -676.26   1352.5
## comb.mod8   54 1458.4 1712.0 -675.18   1350.4 2.1534  3     0.5412
```

```
# trying (0 + Rater:Rubric | Artifact)
# comb.mod9 <- lmer(as.numeric(Rating) ~ (0 + Rubric | Artifact) +
# (0 + as.factor(Rater) | Artifact) +
# (0 + as.factor(Rater):Rubric | Artifact) + as.factor(Rater) + Semester + Rubric +
 # as.factor(Rater):Rubric, data=tall.nonmissing)
# anova(comb.mod6, comb.mod9)
# does not run since more random effects than there are observations in the data set
```

Based on AIC, BIC, and the likelihood ratio test, the best model we had previously is improved when we add Rater as a random effect, but not when we add Semester as a random effect. We are not able to try adding the interaction of Rater and Rubric as a random effect because there are more random effects than there are observations in the data set. Therefore, the final model includes Rater, Semester, Rubric, and the interaction of Rater and Rubric as fixed effects and Rubric and Rater as random effects (grouped by Artifact). Below is the summary output of the final model:

```
# summary of the final model
comb.mod.final = comb.mod7
summary(comb.mod.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: tall.nonmissing
##
## REML criterion at convergence: 1370.6
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -3.06399 -0.46903 -0.02989  0.45345  2.73974
##
## Random effects:
##  Groups     Name               Variance Std.Dev. Corr
##  Artifact   RubricCritDes      0.49641  0.7046
##             RubricInitEDA      0.31792  0.5638    0.32
##             RubricInterpRes    0.10210  0.3195    0.14  0.67
##             RubricRsrchQ       0.17900  0.4231    0.50  0.19  0.54
##             RubricSelMeth      0.03828  0.1956    0.14  0.23  0.38 -0.24
##             RubricTxtOrg       0.25029  0.5003    0.27  0.44  0.36  0.31  0.21
##             RubricVisOrg       0.23234  0.4820    0.18  0.50  0.45  0.28 -0.16
##  Artifact.1 as.factor(Rater)1 0.01279  0.1131
##             as.factor(Rater)2 0.11170  0.3342   -0.49
##             as.factor(Rater)3 0.09407  0.3067    0.33  0.66
##  Residual                     0.13469  0.3670
##
##
##
##
##
##
##
##
##   0.54
##
##
##
##
## Number of obs: 810, groups:  Artifact, 90
##
## Fixed effects:
##                                   Estimate Std. Error t value
## (Intercept)                        1.75757    0.11404  15.412
## as.factor(Rater)2                  0.36605    0.13918   2.630
## as.factor(Rater)3                  0.19587    0.12968   1.510
## SemesterS19                       -0.15919    0.07647  -2.082
## RubricInitEDA                      0.73948    0.12996   5.690
## RubricInterpRes                    0.99152    0.12771   7.764
## RubricRsrchQ                       0.72619    0.11793   6.158
## RubricSelMeth                      0.41067    0.12470   3.293
## RubricTxtOrg                       1.01579    0.13000   7.814
## RubricVisOrg                       0.65425    0.13353   4.900
## as.factor(Rater)2:RubricInitEDA   -0.29980    0.15609  -1.921
## as.factor(Rater)3:RubricInitEDA   -0.29470    0.15635  -1.885
## as.factor(Rater)2:RubricInterpRes -0.51324    0.15349  -3.344
```

```
## as.factor(Rater)3:RubricInterpRes -0.71485    0.15365  -4.653
## as.factor(Rater)2:RubricRsrchQ    -0.48741    0.14722  -3.311
## as.factor(Rater)3:RubricRsrchQ    -0.32238    0.14727  -2.189
## as.factor(Rater)2:RubricSelMeth   -0.38637    0.15031  -2.570
## as.factor(Rater)3:RubricSelMeth   -0.38713    0.14962  -2.587
## as.factor(Rater)2:RubricTxtOrg    -0.55106    0.15646  -3.522
## as.factor(Rater)3:RubricTxtOrg    -0.44489    0.15673  -2.839
## as.factor(Rater)2:RubricVisOrg    -0.10491    0.15861  -0.661
## as.factor(Rater)3:RubricVisOrg    -0.27522    0.15885  -1.733
##
## 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)
## Model failed to converge with max|grad| = 0.00347545 (tol = 0.002, component 1)
ranef(comb.mod.final)
## $Artifact
##     RubricCritDes RubricInitEDA RubricInterpRes RubricRsrchQ RubricSelMeth
## 100   0.799238661  -0.260980308    -0.121640193 -0.164960384   0.232398535
## 101  -0.496536911   0.434140081    -0.166284437 -0.741420752   0.032172440
## 102  -0.770815217  -0.332926374    -0.232449594 -0.744376576   0.048596844
## 103   0.139716219   0.330257722     0.098938081 -0.281458083   0.271633948
## 104  -0.576684888   0.308458605     0.091202414 -0.260863609   0.073380751
## 105  -0.590148853  -0.482875264    -0.340116252 -0.404325504  -0.097627365
## 106   0.204297154  -1.028930906    -0.399264305  0.233287747  -0.136342946
## 107  -0.559967131  -0.401637123     0.057229088  0.183054673  -0.102350397
## 111  -0.461631857  -0.351169582    -0.241493186 -0.311730856  -0.067047556
## 112  -0.499246269   0.322659642     0.271667424  0.197919675  -0.103787736
## 113  -0.586387954  -0.804686323    -0.145360840 -0.131220736   0.004180362
## 114  -0.448167893   0.440164287     0.189825479 -0.168268961   0.103960560
## 115  -0.370729273   0.454365323     0.370290489  0.290514323  -0.073207927
## 116  -0.588739895  -0.354346280    -0.287661321 -0.351775663  -0.123771089
## 117  -0.672371385  -0.136881305     0.011689854 -0.194721996  -0.104347939
## 118  -0.654184088  -0.212095240    -0.029545569 -0.208950383  -0.027836786
## 13    0.387951896  -0.746778388    -0.434945183  0.050320030  -0.158946379
## 15    0.688577054   0.476238943    -0.047060609 -0.110293099  -0.052072684
## 16    0.682624643   1.153822204     0.314165832 -0.008152496   0.186499027
## 17    0.335665412  -0.087985271     0.066956179  0.549399543  -0.199177686
## 21    0.776129319   1.090635435     0.451767594  0.434874078   0.085336957
## 22    0.730652373   0.381960810     0.249027336  0.430992100   0.090401490
## 23   -0.272190064  -0.723849539    -0.327172589 -0.010250439  -0.176573784
## 24    0.049029060  -0.202416510    -0.150336661 -0.130271841   0.068973824
## 25    1.160204524   0.007254384    -0.288843255  0.162781115  -0.130528433
## 26   -0.863181373  -0.483969935    -0.159845792 -0.518355505   0.112902974
## 27    0.101640120   0.386484170     0.006504563 -0.172784543   0.092085785
## 28   -0.299061018  -0.592375772    -0.413640087 -0.405268374  -0.059556639
## 32    0.660732999   0.404159123     0.250563915  0.407948268   0.001581016
## 33   -0.083794630   0.150288893    -0.059642899 -0.452289099   0.169497415
## 34    0.465735609  -0.311661380    -0.060428848 -0.201017691   0.261718302
## 35   -0.743817534  -0.254864980    -0.085389943 -0.283324653  -0.098118034
## 36    0.049029060  -0.202416510    -0.150336661 -0.130271841   0.068973824
## 37    0.775804743  -0.157058362    -0.206974992 -0.021692734   0.102456303
```

```
## 38   -0.017041851  -0.209506832    -0.142028273 -0.174745546  -0.064635555
## 39   -0.527782424   0.248650618     0.207516464  0.140233052  -0.087347669
## 40    0.105488583   0.357195535     0.013276371 -0.194214416   0.047296880
## 45    0.014712774  -0.241842117    -0.172320085 -0.089847827   0.049766845
## 46    0.149541781   0.324180066    -0.013480172 -0.141180702   0.032003236
## 47    1.130787240  -0.177831970    -0.049170158  0.676980376  -0.148729520
## 48    0.537149963   0.660785447     0.224956947  0.134955984   0.242120383
## 49   -0.903770236   0.215428881     0.273326432  0.159476915  -0.189743665
## 53    1.141848123   0.106672794     0.017180058  0.239127642   0.118073571
## 54   -0.662474466   0.383710427    -0.091920305 -0.662629575   0.147353843
## 55   -0.148697979  -0.372283652     0.070561138  0.317326804  -0.133642709
## 56    0.687671804  -0.264919790    -0.276159520 -0.060767795  -0.062178422
## 57   -0.769720347  -0.326339583    -0.169648775 -0.256490605  -0.084352648
## 6    -0.673898160  -0.277063293    -0.086926522 -0.260280821  -0.009297560
## 61    0.001550849   0.332834923    -0.079492221 -0.172068699   0.016015372
## 62    1.385225502   0.997852503     0.303270907  0.483090252  -0.052845706
## 63    0.682905608   0.348720177     0.207016358  0.445217331  -0.018918976
## 64    0.550774570  -0.162662056    -0.076694867  0.054779232   0.062458703
## 65    0.831967452  -0.452141804    -0.411736816  0.253078698  -0.217056813
## 66    0.741070094   0.910079487     0.371937767  0.382503158  -0.040190234
## 67   -0.816111696   0.144944702     0.293039817  0.146896291  -0.188111513
## 68    0.630976869  -0.239481254     0.050698742  0.488221635  -0.041868584
## 7    -0.621287099   0.311837388     0.069914702 -0.302793523   0.013814401
## 72   -0.056306755   0.416395621     0.192082477 -0.072182660   0.022345611
## 73   -0.746462038  -0.931340953    -0.323460593 -0.186647186  -0.017228372
## 74   -1.048520644   0.086086325     0.117360716 -0.493705028   0.092943066
## 75   -0.041452098   0.337766358     0.148289609 -0.088812166   0.098063782
## 76    0.037280650  -0.325850686    -0.216096453 -0.141645020  -0.005302084
## 77   -0.168567886  -0.233635132    -0.010533804 -0.072646977  -0.014959709
## 78    0.416269921   0.062541837     0.074737327  0.131411302   0.084876025
## 79   -0.309308281   0.018410881     0.132294596  0.023694963   0.051678479
## 8    -0.607264438  -0.208776295    -0.035792259 -0.212272183   0.006557514
## 84    0.308494563  -0.083992469     0.160693341  0.445251222  -0.061633601
## 85    1.073451127   0.376466721     0.315669525  0.876578691  -0.131883649
## 86    0.326681859  -0.159206404     0.119457918  0.431022836   0.014877551
## 87    0.204297154  -1.028930906    -0.399264305  0.233287747  -0.136342946
## 88    1.088154910   0.644399236     0.316298426  0.538846828  -0.077157657
## 9    -0.580393484  -0.340250062     0.050675240  0.182745753  -0.110459631
## 92   -0.540370876  -0.348322074     0.068448595  0.221376128  -0.051982969
## 93   -0.392257626  -0.163301345     0.178291168  0.352292231   0.028964268
## 94    1.037137115   1.033247400     0.338423344  0.394338954  -0.006519929
## 95    0.139716219   0.330257722     0.098938081 -0.281458083   0.271633948
## 96    0.239289471   0.380069631     0.224053681  0.314971734  -0.067507025
## O1   -0.502091672   0.422168913     0.120487403 -0.008846794  -0.092672273
## O10  -0.441645802  -0.001277150     0.155485840  0.049612795   0.036698292
## O11  -0.766988121  -0.329857665     0.328631370  0.512691603   0.013923832
## O12  -0.354476640  -0.488418204    -0.316237282 -0.338446046  -0.015550340
## O13   0.030608752   0.083112040    -0.316943057 -0.367440564  -0.071176221
## O2   -0.116290896   0.329529348     0.171021008  0.139967618  -0.072217813
## O3    0.283458088  -0.135225013    -0.013069655  0.231071786  -0.039710623
## O4   -0.111395786   0.044958354     0.203424066 -0.216631696   0.365274648
## O5    0.878936358  -0.426138635    -0.025974271  0.189838134   0.250130031
## O6   -0.897811284  -0.773176898    -0.419320817 -0.412826901  -0.112197441
## O7    0.138211493   0.207025064    -0.005457626 -0.013440429  -0.042576390
```

```
## 08    0.358677516  -0.209742861    -0.396540355 -0.311360802   0.028817529
## 09   -0.799320437   0.585202438     0.349413725 -0.190551386   0.074853457
##      RubricTxtOrg RubricVisOrg as.factor(Rater)1 as.factor(Rater)2
## 100 -0.4896045706  -0.44091416       0.034872652      -0.050337771
## 101  0.2892265038   0.46518308      -0.030972391       0.044707846
## 102 -1.1195337988  -0.43446865      -0.071578730       0.103322045
## 103  0.1091986696  -0.23973216       0.041671903      -0.060152315
## 104  0.0888650812  -0.11667180      -0.025585377       0.036931830
## 105 -0.5322768661  -0.35615872      -0.059169443       0.085409562
## 106  0.0171410394  -0.33472025      -0.022606823       0.032632364
## 107 -0.5132072496  -0.30989807      -0.011070911       0.015980574
## 111 -0.4111243958  -0.25283501      -0.035905610       0.051828820
## 112  0.2084315853   0.41594898       0.019709421      -0.028450039
## 113 -0.4729523020  -0.30654589      -0.016173245       0.023345661
## 114  0.2100175515  -0.01334809      -0.002321544       0.003351089
## 115  0.3295840556   0.51927269       0.042973254      -0.062030781
## 116  0.1297523813   0.27754576      -0.030908165       0.044615137
## 117  0.2258126314   0.84076927       0.010790939      -0.015576442
## 118  0.1275340626   0.31597745      -0.008656746       0.012495789
## 13  -0.7211078953  -0.62638635      -0.065264863      -0.387261548
## 15   0.4109696880   0.90080156       0.013853977       0.082205224
## 16   0.8983842311   0.58526446       0.020413169       0.121125446
## 17  -0.1153832196   0.03060174      -0.017871293      -0.106042733
## 21   0.9176724819   0.59138800       0.043200340       0.256337481
## 22   0.2104040279  -0.12220340       0.018124187       0.107543329
## 23  -0.6711952643  -0.56019958      -0.056521892      -0.335383458
## 24   0.2457523591  -0.13974007      -0.007413357      -0.043988573
## 25  -0.0009367225   0.07458976      -0.038137408      -0.226295605
## 26  -0.4700254003  -0.47147070       0.015601543       0.092574742
## 27   0.2990450875  -0.05298478      -0.006244532      -0.037053125
## 28  -0.6276681655  -0.51275167      -0.068514790      -0.406545613
## 32   0.2599149688   0.36099742       0.027147790       0.161086608
## 33  -0.4038299531  -0.39733071       0.018821670       0.111681981
## 34  -0.5022366423  -0.50585671       0.030028666       0.178180832
## 35  -0.2592888538   0.26601222      -0.004530507      -0.026882629
## 36   0.2457523591  -0.13974007      -0.007413357      -0.043988573
## 37   0.2586550778  -0.15235538      -0.005362055      -0.031816782
## 38  -0.2463861351   0.25339692      -0.002479205      -0.014710838
## 39  -0.2362188155  -0.12430976       0.010401885       0.061721574
## 40  -0.2426043475  -0.14304861      -0.010333984      -0.061318670
## 45   0.2993111199  -0.21568392       0.014015451      -0.084798506
## 46  -0.1861943002  -0.21905637       0.017964343      -0.108690723
## 47  -0.6838706146  -0.67106227       0.060891858      -0.368417589
## 48   0.6090561240  -0.35368729      -0.058092755       0.351482011
## 49   0.2598205680   0.72501480      -0.028198314       0.170609914
## 53   0.0733266480  -0.06265082      -0.066587654       0.402879195
## 54   0.3346826420  -0.11276265       0.048267263      -0.292034259
## 55  -0.3649991584   0.05811973      -0.010336157       0.062537456
## 56  -0.2393999340   0.11161924       0.019261504      -0.116539011
## 57   0.2764513010   0.22691985       0.019280711      -0.116655221
## 6   -0.3087997947  -0.21718860      -0.013554111      -0.080425908
## 61   0.8802736195   0.38769307       0.008605433      -0.052065954
## 62   0.4046667618   0.81899279      -0.054657302       0.330696285
## 63   0.2956982703   0.26739833      -0.036775938       0.222507616
```

```
## 64   0.2363006113   0.18576276      -0.001789389       0.010826443
## 65   0.3135704666   0.16068823       0.010852476      -0.065661376
## 66  -0.1910511745   0.26546053      -0.032510507       0.196700228
## 67  -0.8635224619   0.06975998      -0.003074901       0.018604253
## 68   0.2430487628   0.18119160      -0.035068926       0.212179584
## 7   -0.2555070663  -0.13043330      -0.012385285      -0.073490461
## 72  -0.2053475819   0.24590839      -0.010292635       0.062274133
## 73   0.1792566683  -0.33825265       0.034166251      -0.206718075
## 74  -0.4512669215  -0.05698468      -0.034112701       0.206394081
## 75  -0.3067678174  -0.28156277       0.018650395      -0.112841582
## 76  -0.2957189179  -0.35376849       0.035435857      -0.214399648
## 77  -0.3148721997   0.11119627       0.007178879      -0.043434792
## 78   0.0615527894  -0.04907277      -0.063591829       0.384753380
## 79   0.0497789307  -0.03549473      -0.060596004       0.366627565
## 8   -0.2459844450  -0.16361714      -0.002759838      -0.016376028
## 84   0.2939633472   0.42396954       0.044875168      -0.064776145
## 85   0.2776972787   0.41745677       0.054398471      -0.078522786
## 86   0.1956847784  -0.10082228       0.025427483      -0.036703915
## 87   0.0171410394  -0.33472025      -0.022606823       0.032632364
## 88   0.4758403825   1.03774335       0.071262876      -0.102866119
## 9   -0.2895115439  -0.21106505       0.009233060       0.054786126
## 92   0.0505434291  -0.20098541      -0.002257318       0.003258380
## 93   0.7354465780   0.01125096       0.029820108      -0.043044555
## 94   0.3160189308   0.50177038       0.031070412      -0.044849336
## 95   0.1091986696  -0.23973216       0.041671903      -0.060152315
## 96   0.2324204643   0.41278840       0.024130389      -0.034831593
## O1  -0.2368773428  -0.25866965      -0.212159887       0.272115540
## O10  0.1214002525   0.01135211       0.108572421      -0.367891320
## O11  0.3058120548  -0.26197077       0.052578932       0.052255739
## O12 -0.3629030984  -0.45362830       0.058272653      -0.016862648
## O13  0.2746057615   0.14795510      -0.008168590       0.048631213
## O2   0.1737283681   0.34196971       0.172582771      -1.028287299
## O3   0.2430896893   0.13198046      -0.038571475       0.159659405
## O4   0.1614644726  -0.27808530      -0.089684939       0.418336734
## O5  -0.0423025873  -0.48854095       0.030560028       0.059691835
## O6  -0.0546947423  -0.18324192      -0.050097044       0.116111749
## O7   0.1232519876   0.22443979       0.147434876       0.138761070
## O8  -0.5688044556  -0.90901065      -0.039170504      -0.248413107
## O9   0.3977601448   0.55927030       0.048063682       0.035195534
##     as.factor(Rater)3
## 100       0.031207830
## 101      -0.027717454
## 102      -0.064056408
## 103       0.037292537
## 104      -0.022896569
## 105      -0.052951233
## 106      -0.020231036
## 107      -0.009907452
## 111      -0.032132233
## 112       0.017638127
## 113      -0.014473573
## 114      -0.002077569
## 115       0.038457127
## 116      -0.027659977
```

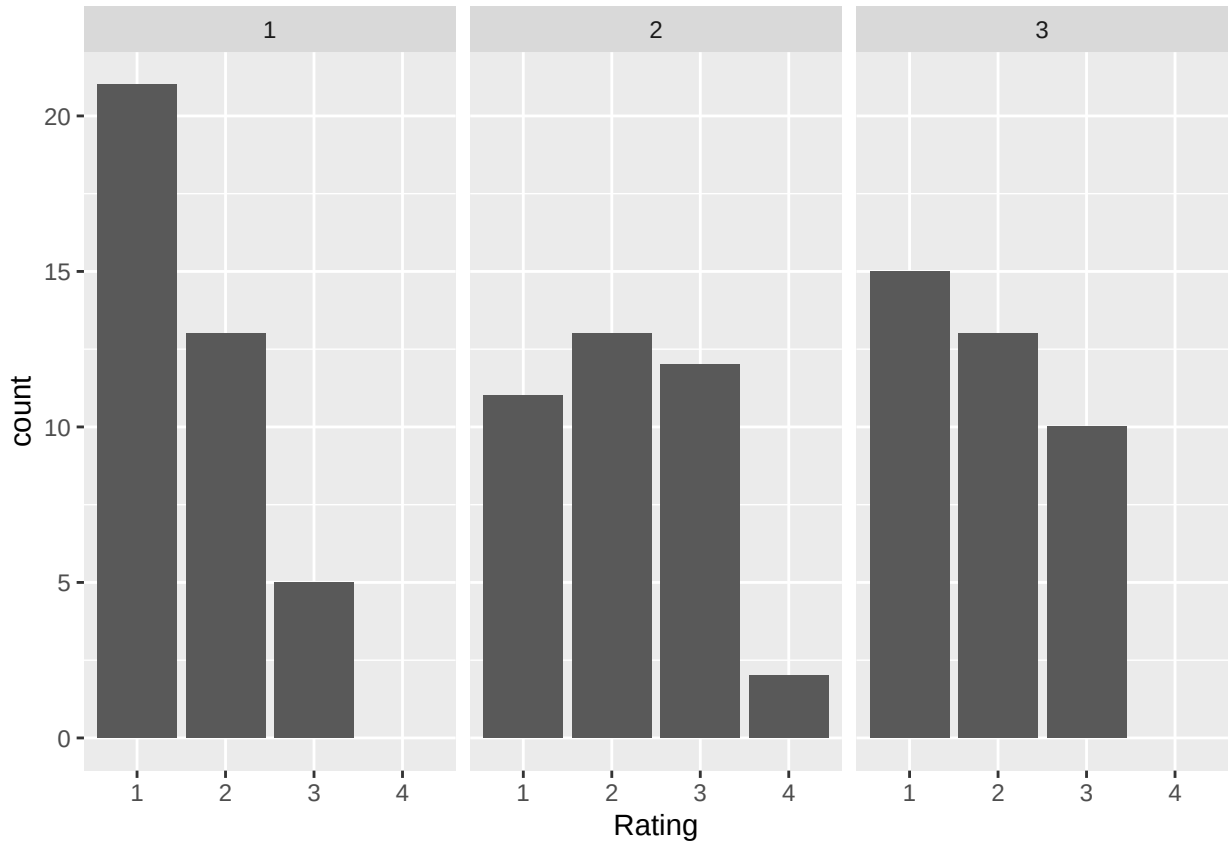
```
## 117       0.009656902
## 118      -0.007746995
## 13       -0.536399916
## 15        0.113863294
## 16        0.167772090
## 17       -0.146880871
## 21        0.355055657
## 22        0.148959360
## 23       -0.464543046
## 24       -0.060929020
## 25       -0.313444348
## 26        0.128226219
## 27       -0.051322661
## 28       -0.563110471
## 32        0.223122702
## 33        0.154691850
## 34        0.246800086
## 35       -0.037235404
## 36       -0.060929020
## 37       -0.044069749
## 38       -0.020376132
## 39        0.085491181
## 40       -0.084933114
## 45       -0.051558659
## 46       -0.066085455
## 47       -0.224002964
## 48        0.213705899
## 49        0.103733175
## 53        0.244956094
## 54       -0.177560848
## 55        0.038023634
## 56       -0.070857322
## 57       -0.070927980
## 6        -0.111398745
## 61       -0.031656816
## 62        0.201067892
## 63        0.135287692
## 64        0.006582626
## 65       -0.039923020
## 66        0.119596445
## 67        0.011311642
## 68        0.129008107
## 7        -0.101792386
## 72        0.037863530
## 73       -0.125687434
## 74        0.125490441
## 75       -0.068609234
## 76       -0.130357936
## 77       -0.026408951
## 78        0.233935349
## 79        0.222914604
## 8        -0.022682604
## 84        0.040159166
## 85        0.048681650
```

```
## 86        0.022755269
## 87       -0.020231036
## 88        0.063773748
## 9         0.075884822
## 92       -0.002020093
## 93        0.026686266
## 94        0.027805173
## 95        0.037292537
## 96        0.021594489
## O1       -0.223750379
## O10      -0.112482029
## O11       0.174280496
## O12       0.118916989
## O13       0.029264380
## O2       -0.619095860
## O3        0.068713591
## O4        0.206531634
## O5        0.130403059
## O6       -0.001350449
## O7        0.480983581
## O8       -0.337806735
## O9        0.146832247
##
## with conditional variances for "Artifact"
```

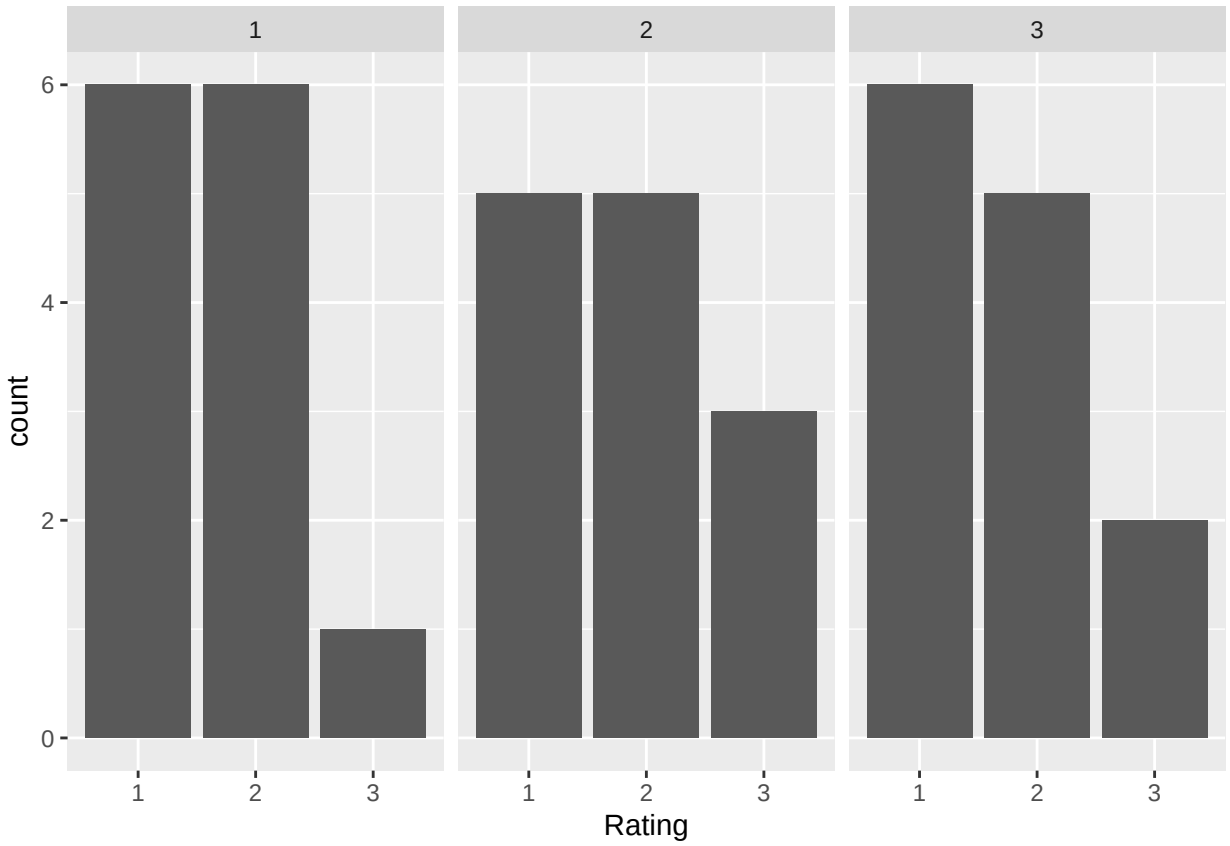
Finding Other Interesting Things About This Data

Finally, we will show some other interesting things we found while working with this data. First, we will use EDA to take a deeper look into the differences in the models when fitted with just the data with the 13 repeated artifacts and the models when fitted with all data. We found that the models are different for the CritDes, InterpRes, VisOrg, and SelMeth rubrics. We will construct bar plots of rating facetted by rater for each of the two models and compare them.

```
# CritDes
ggplot(tall.nonmissing[tall.nonmissing$Rubric=="CritDes",], aes(x = Rating)) +
  geom_bar() +
  facet_grid(~Rater)
```

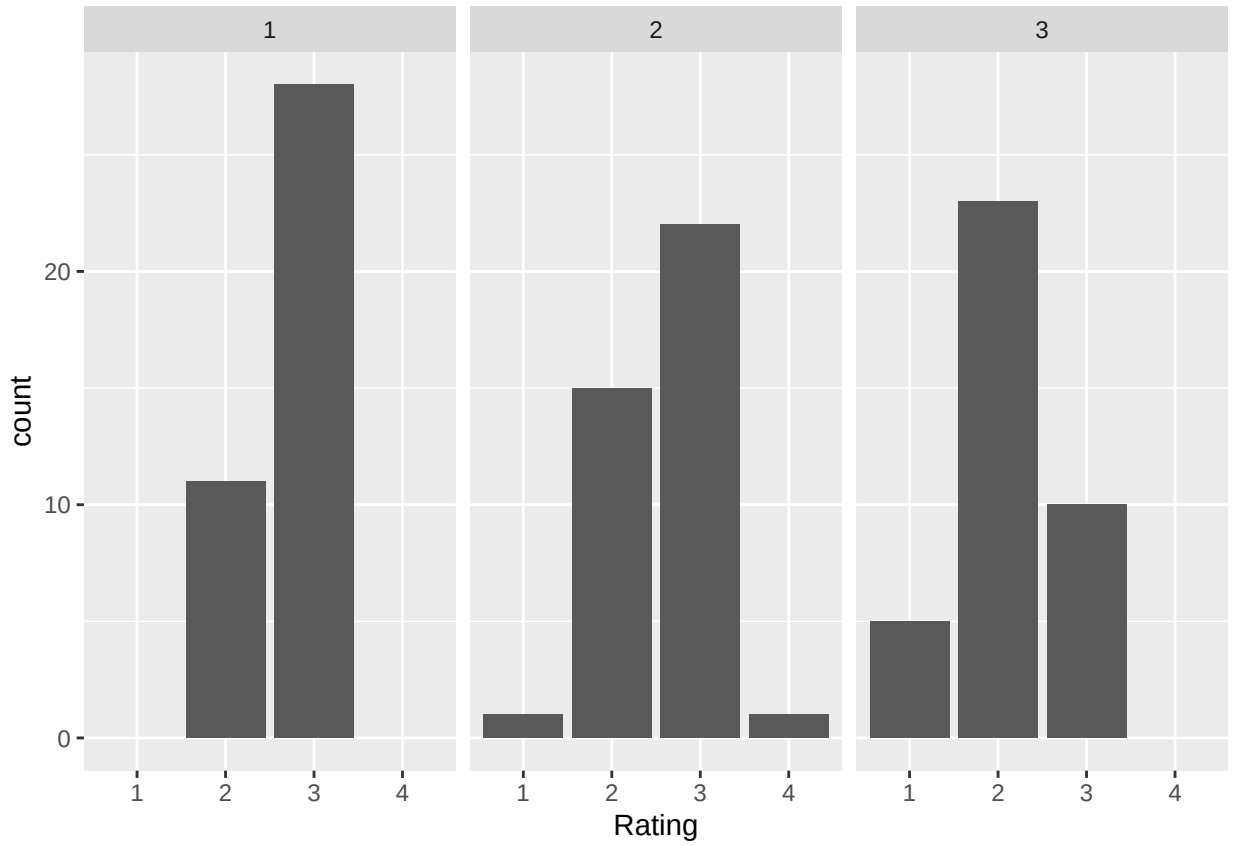



```
ggplot(tall.repeated[tall.repeated$Rubric=="CritDes",], aes(x = Rating)) +
  geom_bar() +
  facet_grid(~Rater)
```

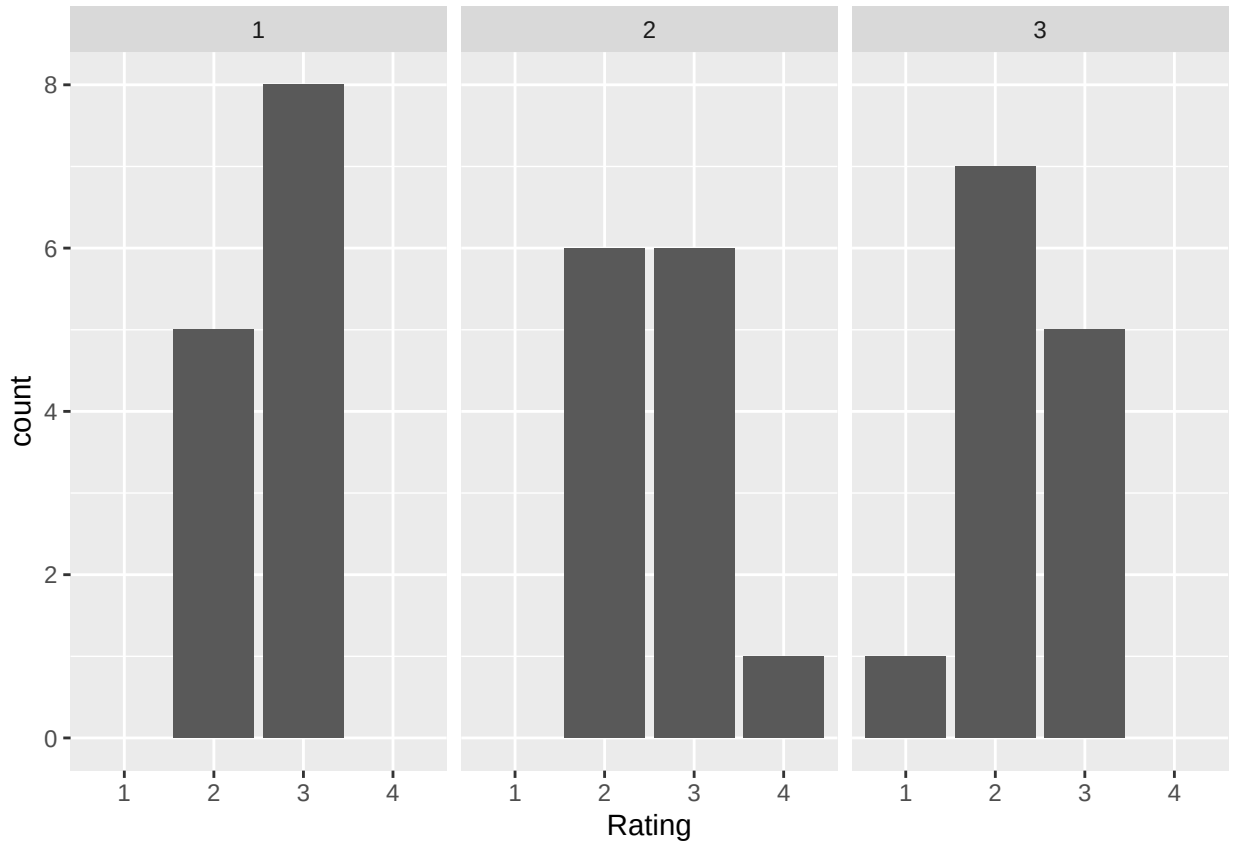


Comparing these bar plots, it makes sense why Rater would be included in the model using the full data set and not just the repeated data set. The distributions of ratings look roughly similar for the repeated data, whereas the distribution of ratings for Rater 1 looks quite different from the other two raters with a majority of ratings of 1.

```
# InterpRes
ggplot(tall.nonmissing[tall.nonmissing$Rubric=="InterpRes",], aes(x = Rating)) +
  geom_bar() +
  facet_grid(~Rater)
```

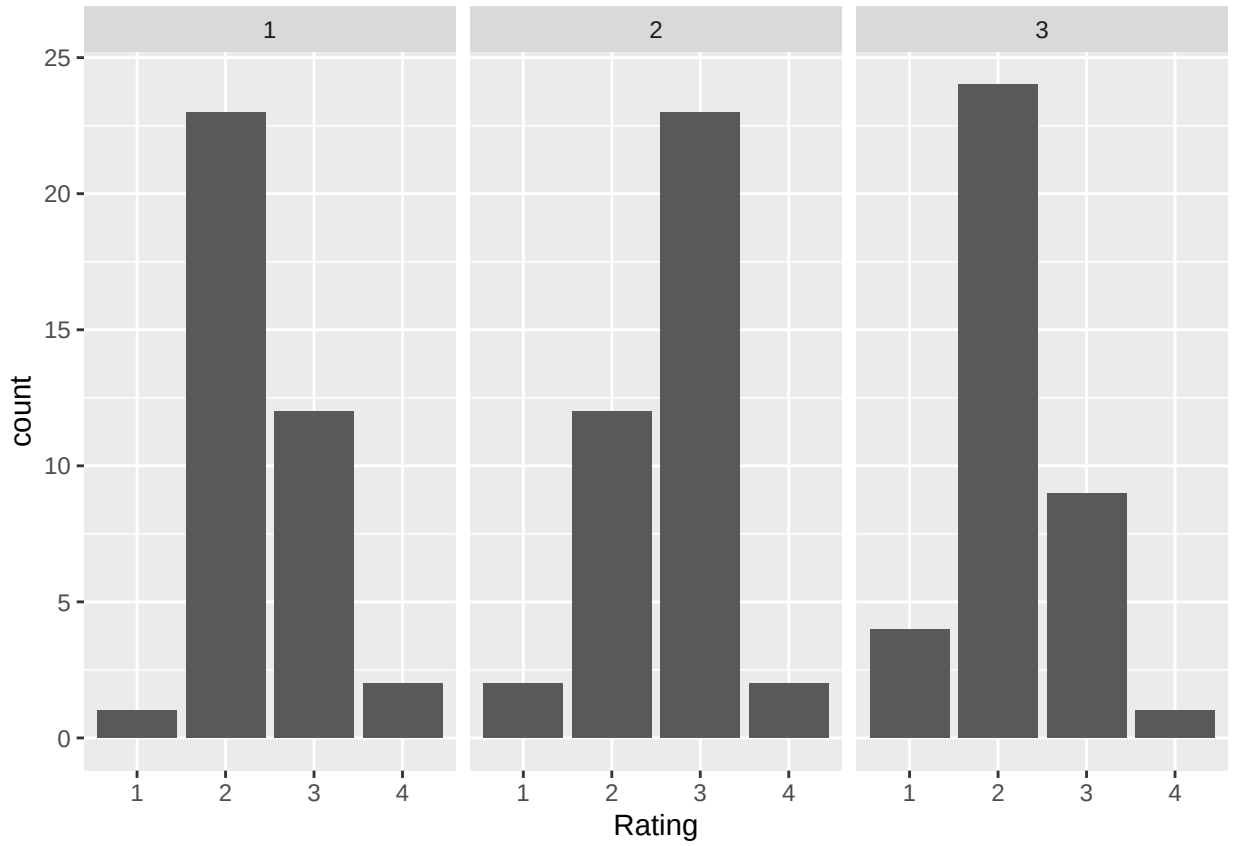



```
ggplot(tall.repeated[tall.repeated$Rubric=="InterpRes",], aes(x = Rating)) +
  geom_bar() +
  facet_grid(~Rater)
```

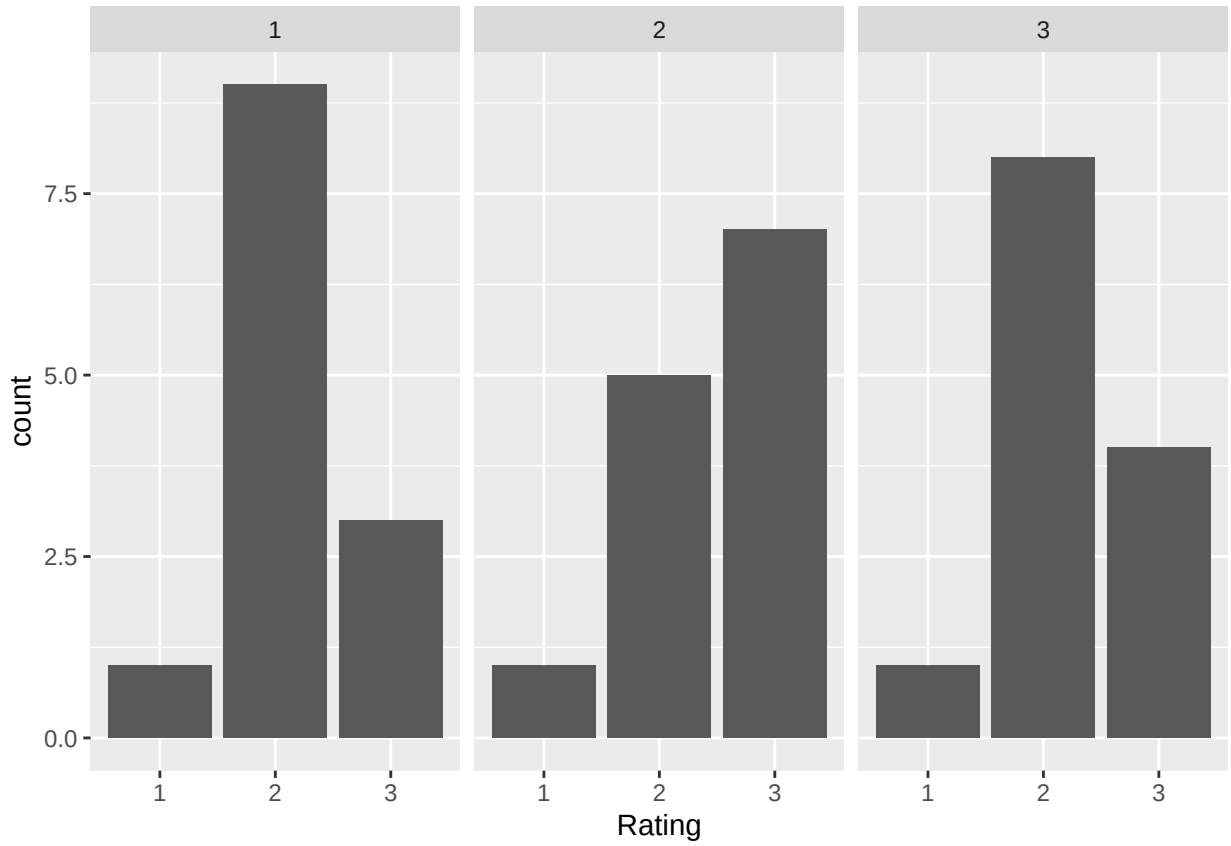


Similar to the CritDes bar plots, we see that when we look at the full dataset, the distribution of ratings between raters seems to differ. Rater 1 gives mostly ratings of 3, Rater 2 gives similar numbers of 2 and 3 ratings, and Rater 3 gives mostly 2 ratings. In contrast, the three raters have similar distributions of ratings when look at the reduced dataset, with all three raters giving roughly similar numbers of 2 and 3 ratings.

```
# VisOrg
ggplot(tall.nonmissing[tall.nonmissing$Rubric=="VisOrg",], aes(x = Rating)) +
  geom_bar() +
  facet_grid(~Rater)
```

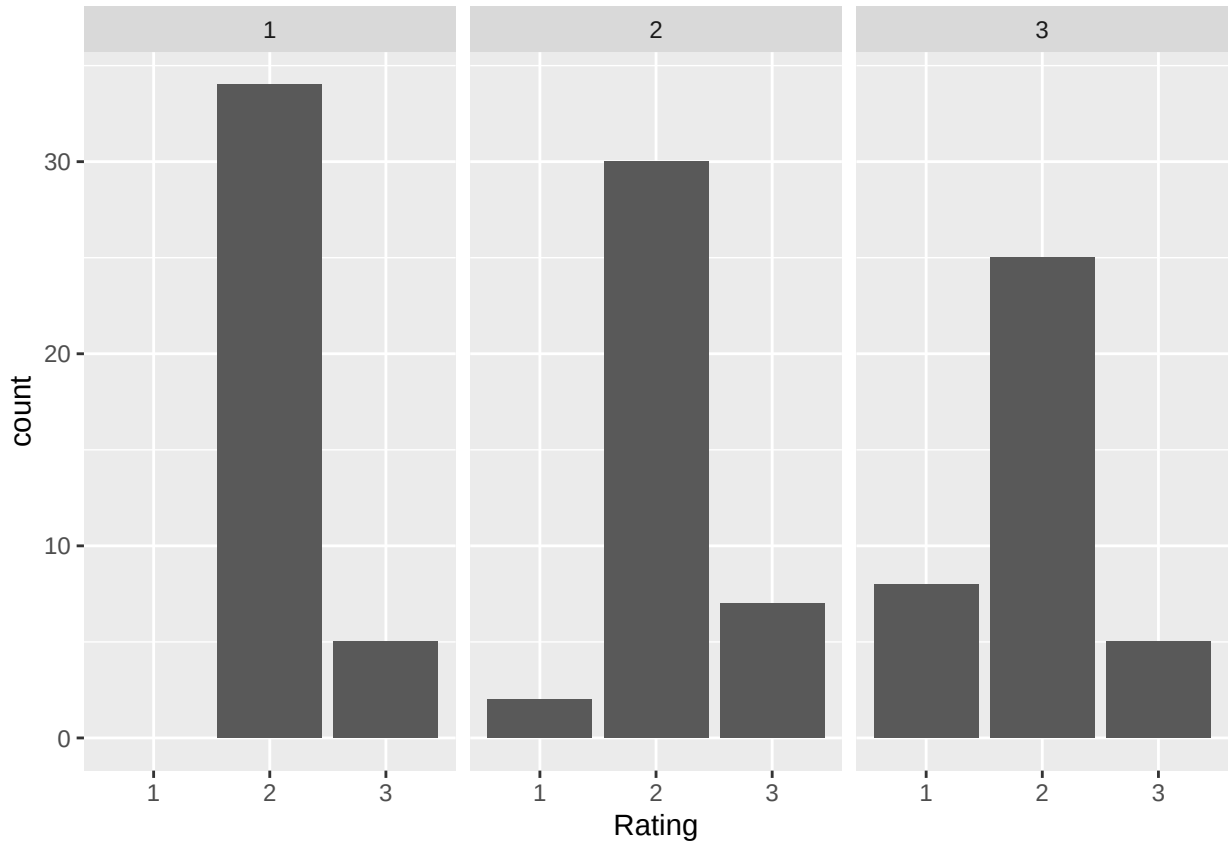



```
ggplot(tall.repeated[tall.repeated$Rubric=="VisOrg",], aes(x = Rating)) +
  geom_bar() +
  facet_grid(~Rater)
```

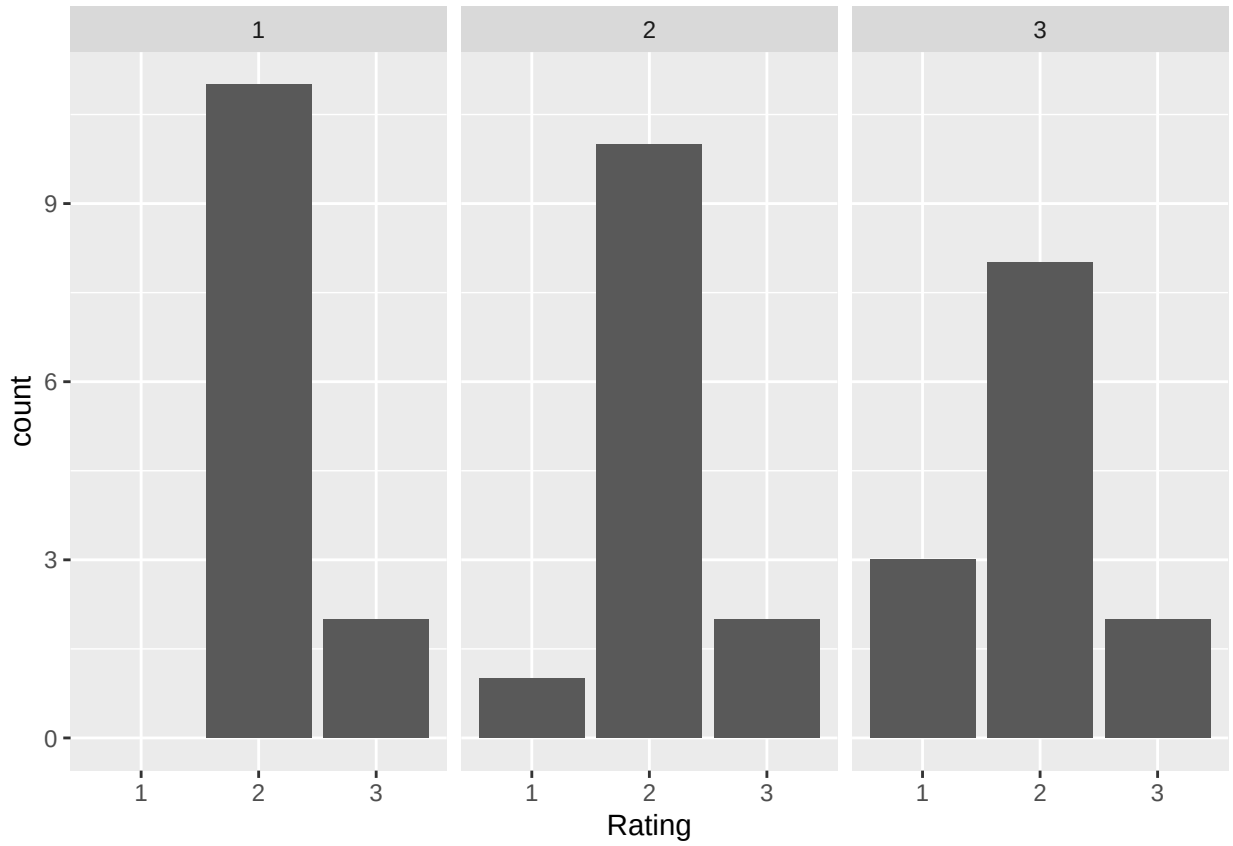


The distribution of ratings between the different raters also seems to differ for the VisOrg rubric when looking at the full dataset. Raters 1 and 3 give out mostly 2s while Rater 2 gives out more 3s. Given the small sample size, the distributions for the reduced data seem roughly similar.

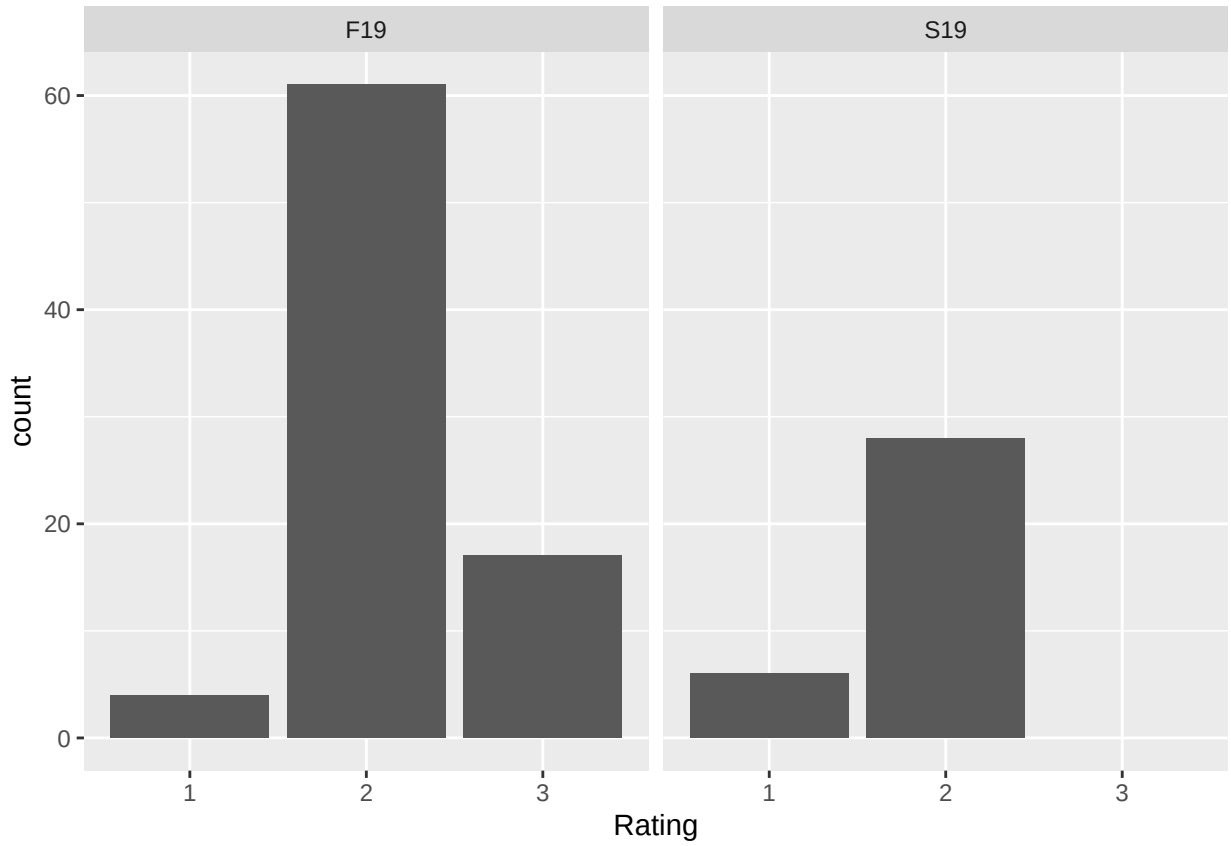
```
# SelMeth
ggplot(tall.nonmissing[tall.nonmissing$Rubric=="SelMeth",], aes(x = Rating)) +
  geom_bar() +
  facet_grid(~Rater)
```



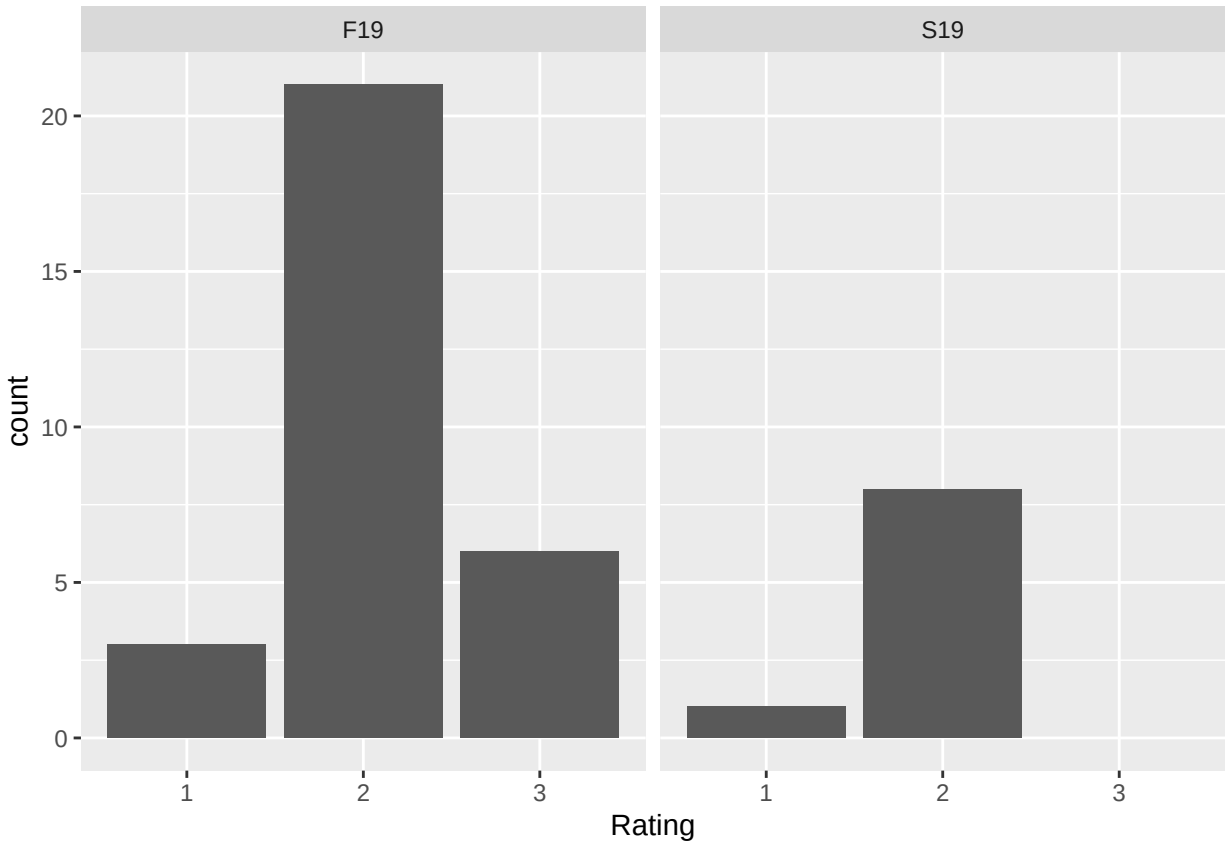

```
ggplot(tall.repeated[tall.repeated$Rubric=="SelMeth",], aes(x = Rating)) +
  geom_bar() +
  facet_grid(~Rater)
```




```
ggplot(tall.nonmissing[tall.nonmissing$Rubric=="SelMeth",], aes(x = Rating)) +
  geom_bar() +
  facet_grid(~Semester)
```




```
ggplot(tall.repeated[tall.repeated$Rubric=="SelMeth",], aes(x = Rating)) +
  geom_bar() +
  facet_grid(~Semester)
```



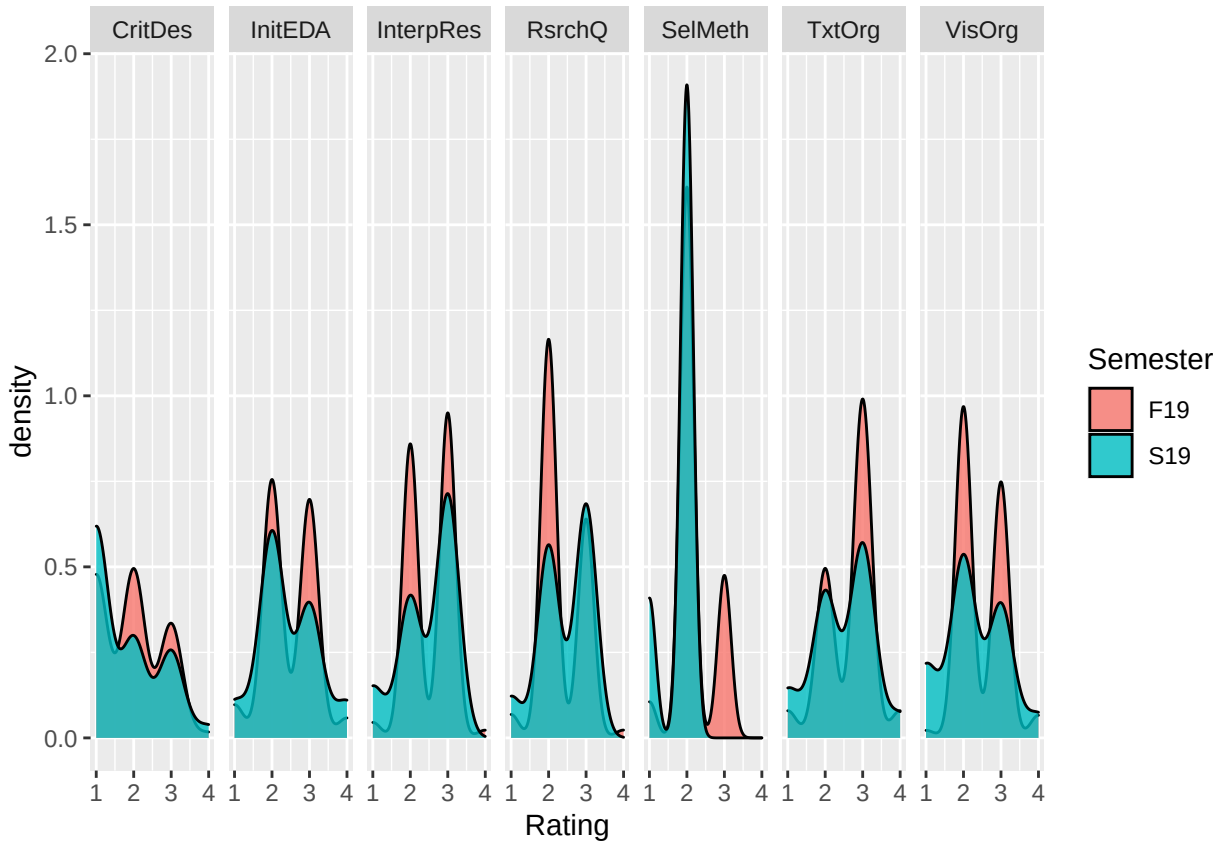
There are slight differences in the distributions of ratings given the full dataset for SelMeth. Rater 1 gives almost all 2s while Raters 2 and 3 give some 1s and 3s in addition to mostly 2s. There are also differences in the distributions of ratings given semester for the full dataset. Practically all ratings in the spring are 2 whereas there were a decent number of 3s in the fall.

Comparing these bar plots allows us to see how the models fitted to the data from the 13 common items, vs fitting to all the data are different since there are clearer differences in the distributions of ratings when looking at the full dataset compared with the reduced set.

Next, we will do some EDA to examine how the Semester variable is related to the ratings since it was added as a fixed effect in some of the previous models. We create density curves of rating filled by semester, with one plot facetted by Rubric and the other not.

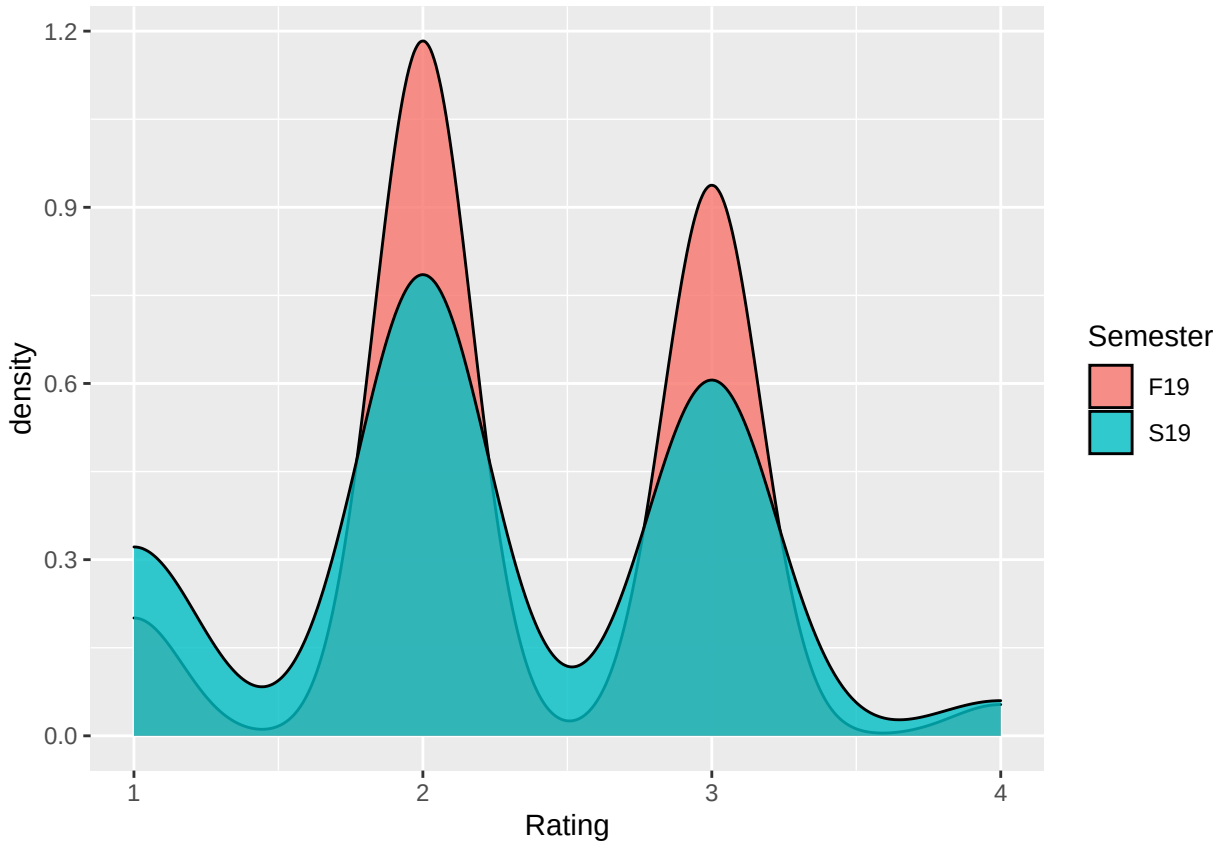
```
# density curves for Rating by Semester
ggplot(tall.ratings, aes(x = as.numeric(Rating), fill = Semester)) +
  geom_density(alpha = 0.8) +
  facet_grid(~Rubric) +
  labs(x = "Rating")
```

```
## Warning: Removed 2 rows containing non-finite values (stat_density).
```




```
ggplot(tall.ratings, aes(x = as.numeric(Rating), fill = Semester)) +
  geom_density(alpha = 0.8) +
  labs(x = "Rating")
```

```
## Warning: Removed 2 rows containing non-finite values (stat_density).
```



Looking at the first plot, we see that it makes sense that Semester is included as a fixed effect in the random-intercept model for SelMeth as the distribution seems to be generally shifted to the left for the spring compared to the fall. It also makes sense that Semester was not included as a fixed effects in the other random-intercept models since the distributions of fall and spring ratings look similar for the other six rubrics.

Looking at the second plot, we see that the distributions between fall and springs ratings when looking at all the data do not appear to be that different. However, since the combined model we fit includes interactions with Rubric, it makes sense that Semester would still be added as a fixed effect since for at least one rubric the distributions of ratings between the two semesters appear to be different.

We construct similar plots for the Sex and Repeated variables to see if there is visual evidence that they should not be included in modelling.

```
# density curves for Rating by Sex
ggplot(tall.ratings, aes(x = as.numeric(Rating), fill = Sex)) +
  geom_density(alpha = 0.8) +
  facet_grid(~Rubric) +
  labs(x = "Rating")
```

```
## Warning: Removed 2 rows containing non-finite values (stat_density).
```

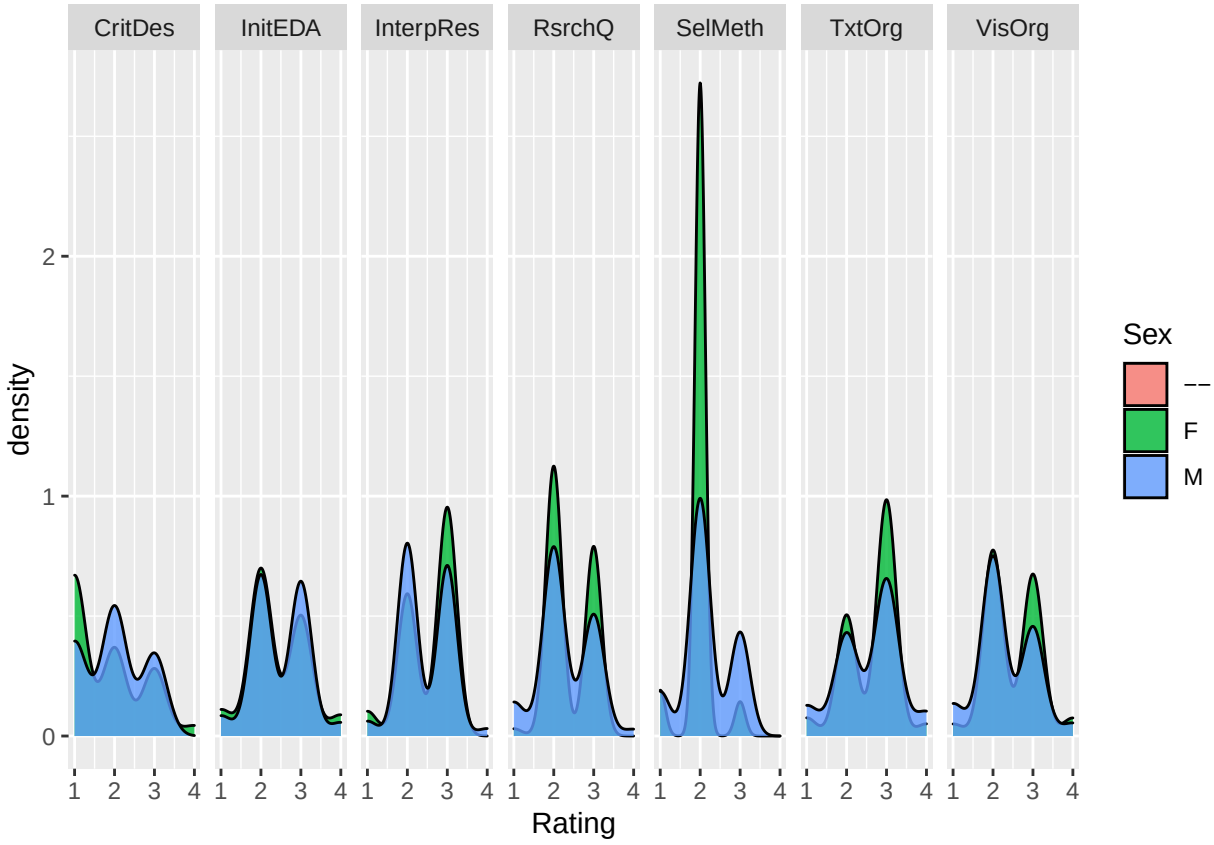
```
## Warning: Groups with fewer than two data points have been dropped.
```

```
## Warning: Groups with fewer than two data points have been dropped.
```

```
## Warning: Groups with fewer than two data points have been dropped.
```

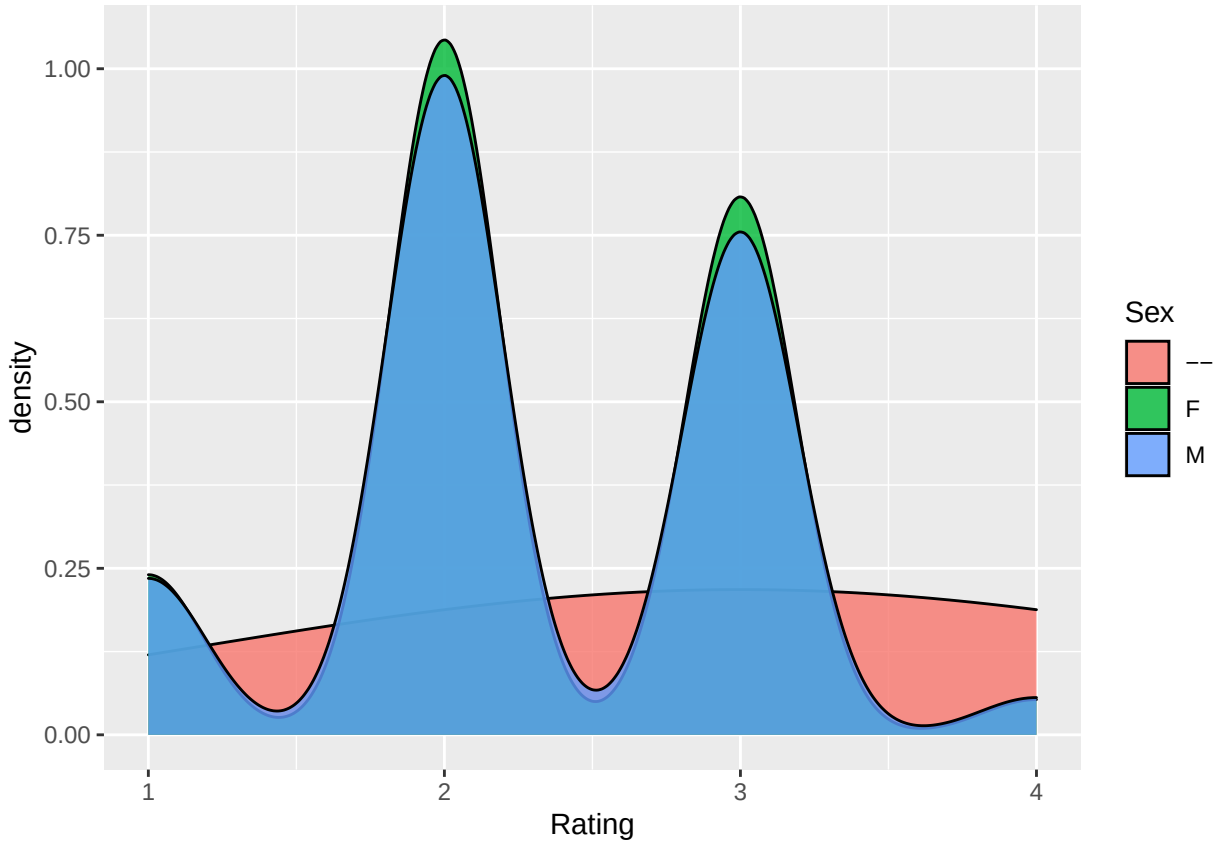
```
## Warning: Groups with fewer than two data points have been dropped.
```

```
## Warning: Groups with fewer than two data points have been dropped.

## Warning: Groups with fewer than two data points have been dropped.

## Warning: Groups with fewer than two data points have been dropped.
## Warning in max(ids, na.rm = TRUE): no non-missing arguments to max; returning
## -Inf

## Warning in max(ids, na.rm = TRUE): no non-missing arguments to max; returning
## -Inf

## Warning in max(ids, na.rm = TRUE): no non-missing arguments to max; returning
## -Inf

## Warning in max(ids, na.rm = TRUE): no non-missing arguments to max; returning
## -Inf

## Warning in max(ids, na.rm = TRUE): no non-missing arguments to max; returning
## -Inf

## Warning in max(ids, na.rm = TRUE): no non-missing arguments to max; returning
## -Inf

## Warning in max(ids, na.rm = TRUE): no non-missing arguments to max; returning
## -Inf
```



```
ggplot(tall.ratings, aes(x = as.numeric(Rating), fill = Sex)) +
  geom_density(alpha = 0.8) +
  labs(x = "Rating")
```

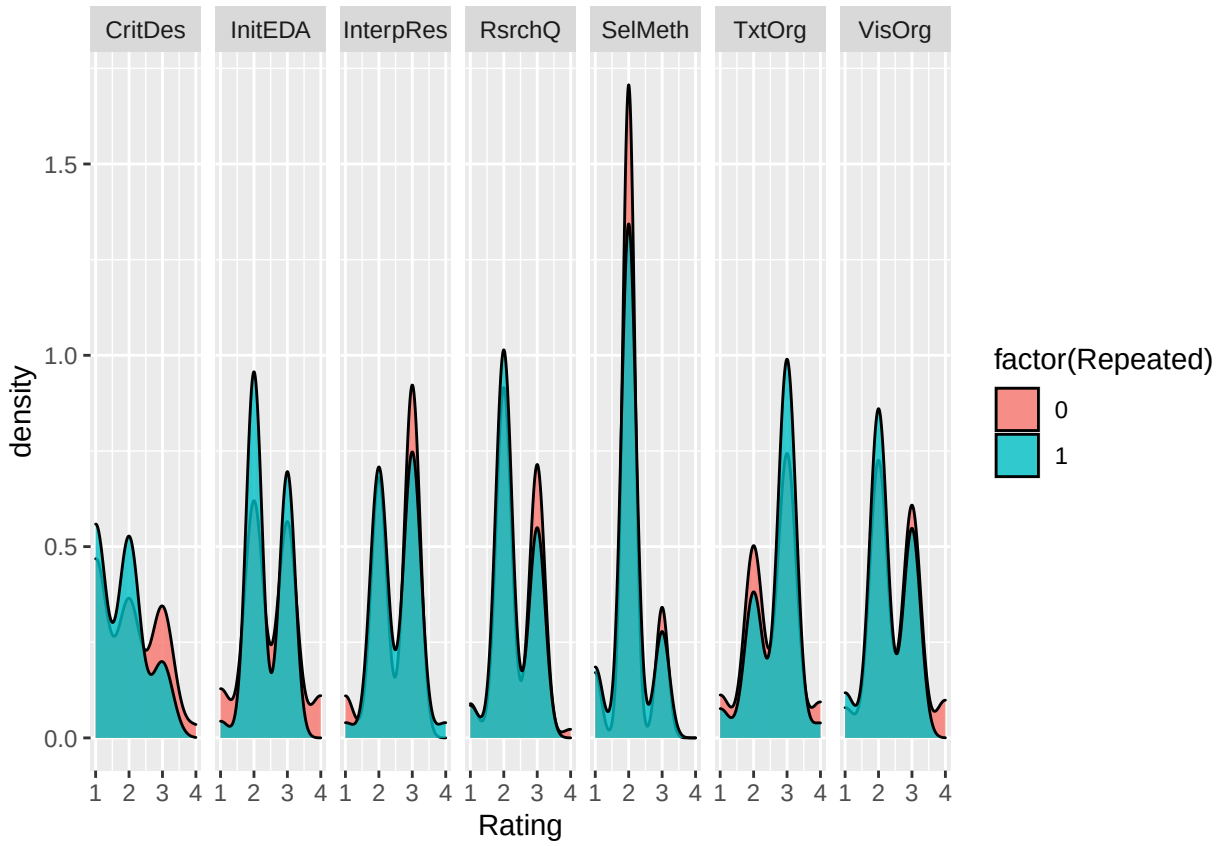
```
## Warning: Removed 2 rows containing non-finite values (stat_density).
```



The distributions of ratings given Sex for each rubric and for the data all together appear to be very similar- each are generally bimodal and the male and female curves mostly overlap with each other. This suggests that there is not a difference in the distribution of ratings for artifacts created by males versus females. Thus, it makes sense that Sex was not included in any modelling.

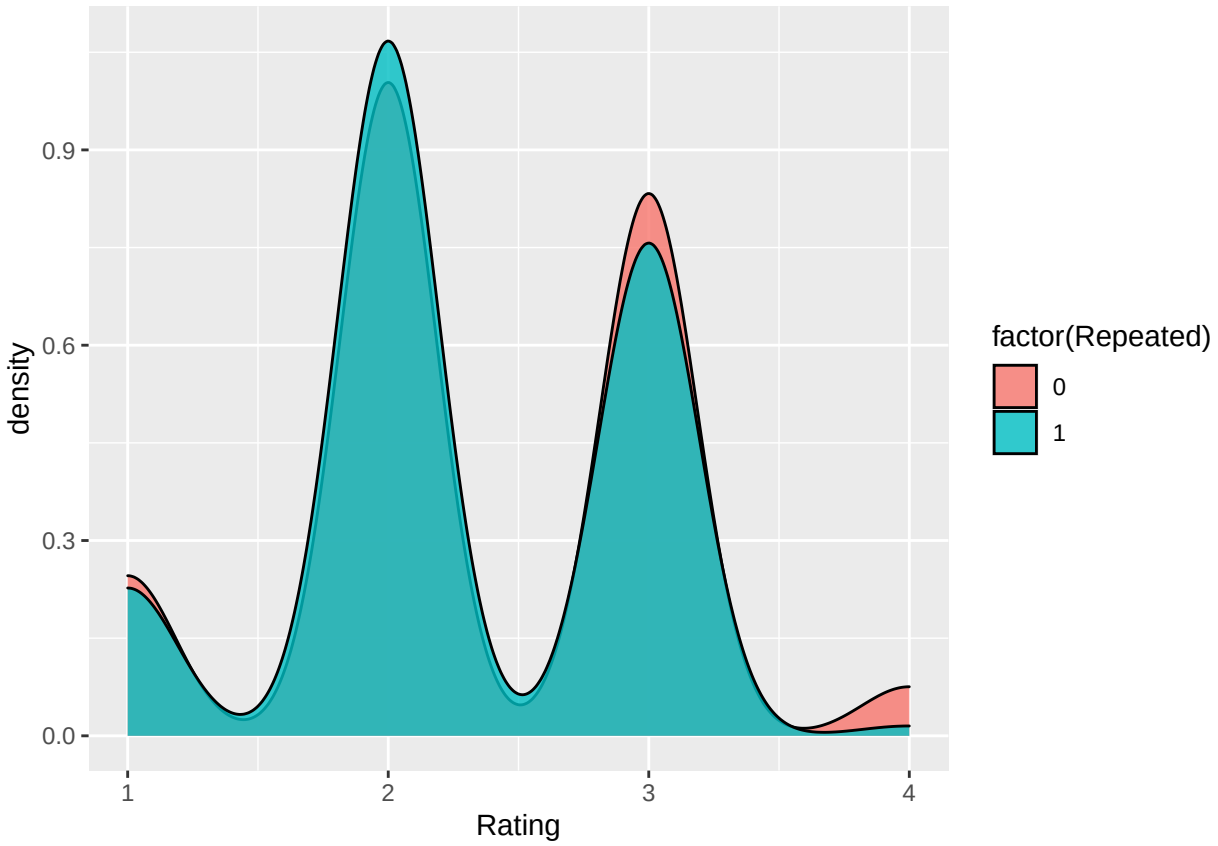
```
# density curves for Rating by Sex
ggplot(tall.ratings, aes(x = as.numeric(Rating), fill = factor(Repeated))) +
  geom_density(alpha = 0.8) +
  facet_grid(~Rubric) +
  labs(x = "Rating")
```

```
## Warning: Removed 2 rows containing non-finite values (stat_density).
```




```
ggplot(tall.ratings, aes(x = as.numeric(Rating), fill = factor(Repeated))) +
  geom_density(alpha = 0.8) +
  labs(x = "Rating")
```

```
## Warning: Removed 2 rows containing non-finite values (stat_density).
```



Similar to Sex, the distributions of ratings given whether or not the artifact was seen by all three raters also appear to be very similar for each rubric and for the data all together- each are generally bimodal and the curves mostly overlap with each other. This suggests that there is not a difference in the distribution of ratings for artifacts for artifacts rated by all three raters vs just one. Thus, it makes sense that Repeatted was not included in any modelling.