Investigating Students' Performance in General-Education Courses Using Mixed-Effect Models

Yueni Wang | Master of Statistical Practice
Carnegie Mellon University| yueniw@andrew.cmu.edu

Abstract

Dietrich college implemented a general education program, and the dean office is interested in students' success under this new program. We can evaluate students' performance through looking at the associations between different factors and rating scores on students' artifacts of departmental experiment. The data we used comes from 91 artifacts ratings of three raters from three different department grading the general education courses on seven rubrics. We looked at the indistinguishable patterns of distribution plots on ratings across different raters and rubrics, also percent of exact agreement to obtain descriptive knowledge on factors. We also investigated ICC values and mix-effect models of multiple factors to see which factor is most closely related to the ratings. We found out that the raters all have at least one rubric that they would give different ratings than others. The model suggested that rater, semester, rubric and the interaction term between rubric and rater have the biggest impact on ratings, thus very important in terms of students' performance.

1 Introduction

The project is inspired by a new policy that the Dietrich College at Carnegie Mellon University recently conducted. The college is in the process of implementing a new "General Education" program for undergraduates which specifies a set of courses and experiences that all undergraduates must take. To determine whether the new program is successful, the college hopes to rate student work performed in each of the "Gen Ed" courses each year. With the resulting ratings for artifacts on seven rubrics from three raters in three different departments, we are specifically interested in looking into the following questions to provide the dean from Dietrich College as an insight of the efficiency of whole 'Gen Ed' experiment:

1. 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?
2. 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?
3. More generally, how are the various factors in this experiment (Rater, Semester, Sex, Repeated, Rubric) related to the ratings? Do the factors interact in any interesting ways?
4. Why some factors from rating data are having very skewed distributions? How would the pattern affect our result and final model selection?

2 Data:

The data we are using comes from a designed experiment on General Education feedbacks. 91 project papers—referred to as "artifacts"—were randomly sampled from a Fall and Spring section of Fresh-man Statistics. Three raters from three different departments were asked to rate these artifacts on seven rubrics, as shown in Table1. The rating scale for all rubrics is shown in Table 2.

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

Table 1: Rubrics for rating Freshman Statistics projects. (Only approved in this experiment)

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

Table 2: Rating scale used for all rubrics. (Only approved in this experiment)

The raters did not know which class or which students produced the artifacts that they rated. Thirteen of the 91 artifacts were rated by all three raters; each of the remaining 78 artifacts were rated by only rater. We are using two csv files in this study, the first csv file with ratings has the following variables shown in Table3:

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

Table 3: Variables in the rating.csv file. (Only approved in this experiment)

The second csv file(tall.csv) contains the same data but organized so that each row has one rating listed under the rating column. We print out the first six rows as an example of data in ratings.csv:

X	Rater	Sample	Overlap	Semester	Sex	RsrchQ	CritDes	InitEDA	SelMeth
1	3	1	5	Fall	M	3	3	2	2
2	3	2	7	Fall	F	3	3	3	3
3	3	3	9	Spring	F	2	1	3	2
4	3	4	8	Spring	M	2	2	2	1
5	3	5	NA	Fall	–	3	3	3	3
6	3	6	NA	Fall	M	2	1	2	2

Table 4: First 6 rows of rating.csv file. (Only approved in this experiment)

We look at the descriptive summary statistics of ratings. To see if there are indistinguishable patterns of ratings distributions across raters and rubrics, we are making subsets according to rubrics and raters and see the distribution and summary statistics on ratings in method section to answer the first research question.

	X	Rater	Sample	Overlap	RsrchQ	CritDes
Min	1	1	1	1	1	1
Median	59	2	60	7	2	1
Mean	59	2	59.89	7	2	2
Max	117	3	118	13	2.35	1.871
1st Quantile	30	1	31	4	3	3
3rd Quantile	88	3	89	10	4	4
NAs						1

	InterpRes	VisOrg	TxtOrg	Repeated	InitEDA
Min	1	1	1	0	1
Median	3	2	3	0	2
Mean	2.487	2.414	2.598	0.3333	2.436
Max	4	4	4		4

1st Quantile	2	2	2	0	2
3rd Quantile	3	3	3	1	3
NAs		1			

	Semester	Sex	Artifact
Length	117	117	117
Class	character	character	character
Mode	character	character	character
NAs		1	

Table4: The summary statistics for ratings.csv

It is very important to notice that there are missing values in each of Sex, CritDes and VisOrg. To deal with the missing data, when we are fitting our models using Ratings as the response variable, missing values of CritDes and VisOrg will be dropped. We are not dropping the missing Sex value coded the way it is so that the observation is not dropped when Sex is included in modelling. When using the reduced dataset that includes only artifacts that were rated by all three raters, we do not have to do anything about the missing values.

3 Methods:

3.1.1 Comparing Distributions of Ratings in Each Rubric:

The method for research question 3.1.1 is just the exploratory data analysis on the csv file of ratings. By subsetting the data based on different rubrics using the full dataset, we look at the bar plots for ratings to see if there's difference in distribution for each subset.

3.1.2 Comparing Distributions of Ratings for Each Rater:

The method for research question 3.1.2 is just the exploratory data analysis on the csv file of ratings. By subsetting the data based on three different raters using the full dataset, we look at the bar plots for ratings to see if there's difference in distribution for each subset.

3.2 Studying Rater's Agreements Based on The Scores:

For the second research question, we fit random-intercept models of all seven rubrics. compare the ICC values across all three raters on every artifact after fitting our models. ICC is the common correlation among the raters' ratings for each artifact. We treat each artifact as a cluster of three ratings and fit the random-intercept model and fit seven random-intercept models, one for each rubric, then calculate the ICCs. Because ICC values could not tell us which raters might be contributing to the disagreement, we are also using the 2-way table to count for the ratings of each pair of raters. The percentage will tell us the exact agreement or disagreement on each rubric. This is also known as computing percent exact agreement between each two pairs of raters to see who is agreeing/disagreeing with the other one within each pair.

3.3 Relationships between Factors and Ratings and How They Interact & Affect Each Other:

To address the third research question, we first try adding fixed effects to seven random intercept models we fitted in 3.2 using data from the artifacts seen by three raters. Then we use backwards elimination method to look for a comparatively best model while comparing our models to intercept-only model using likelihood ratio test for each rubric.

Then we are considering the intersections and random effects. Before adding any of those, we first delete observations with missing values to ensure all models were fit and compared using the same set of data. Using backwards elimination method again, we find our best model and decide the factors of which we want to add fixed effects into the model. By comparing the t-stats of each fixed effects, we check if those coordinates are significant. We are interested in investigating interactions with Rubric factor. Based on what we did in the previous steps, we follow the sequence of adding fixed effects, interactions and random effects to a new model having Rubric as a random effect grouped by Artifact. Next, we tried adding interactions based on the fixed effects we found. To compare our models and decide their efficiency, we look at multiple statistics like AIC, BIC. In addition, interaction terms will be tested using ANOVA and lmer function to see if they are significant.

Interaction terms here will only be evaluated if the model contains a fixed effect and LRT will be excluded from the evaluation when the random effects are assessed. Finally, if any of the seven models contain a fixed effect, the equivalent random effect will also be tested to see if it should be included in the model. Looking at the summary statistics of the final model, we evaluate the performance of selected model accordingly.

3.4 The Skewness of Distributions on Some Factors:

For the fourth question, we are doing additional exploratory data analysis on rating score == 1 and making the marginal percentage table for each factor to see if the extremely skewed or unevenly distributed factors has obvious relationships with other factors. Statistical Summaries and necessary tables will also be included.

4 Results

4.1.1 Comparing Distributions of Ratings in Each Rubric:

After subsetting the full dataset using each rubric, our bar plots on different rubrics are as follows:

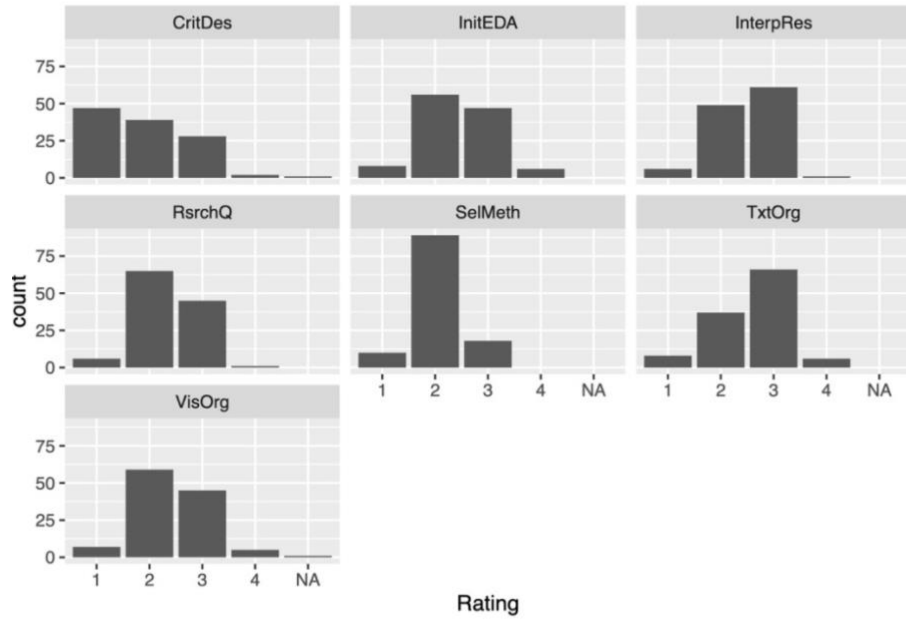


Figure 4.1.1: Barplots showing Distribution of Ratings by different Rubrics

We can see from the barplot that distributions of ratings in InitEDA, InterpRes, RsrchQ, TxtOrg, and VisOrg are very similar to each other, with a roughly normal bell shape and most ratings gathering at 2 or 3. However, CritDes and SelMeth have different distributions than the other 5 rubrics. CritDes as a descending curve while rating scores are increasing. SelMeth has sa extremely high frequency on receiving score 2 and no score 4. Similar distribution is discovered when we used full dataset after properly handled the NA values to generate our bar plots again. CritDes and SelMeth also stand out with similar distributions. InitEDA, RsrchQ, InterpRes, VisOrg, TetOrg, SelMeth have high values of rate 3 and rate 4, and very few rate 1 and rate 2. CritDes only has roughly all numbers of rate 1. After dividing our data into subsets, we notice that there are big differences across different rubrics and raters.

Therefore, we can achieve a primary conclusion that CritDes and SelMeth have different distributions than other factors.

4.1.2 Comparing Distributions of Ratings for Each Rater:

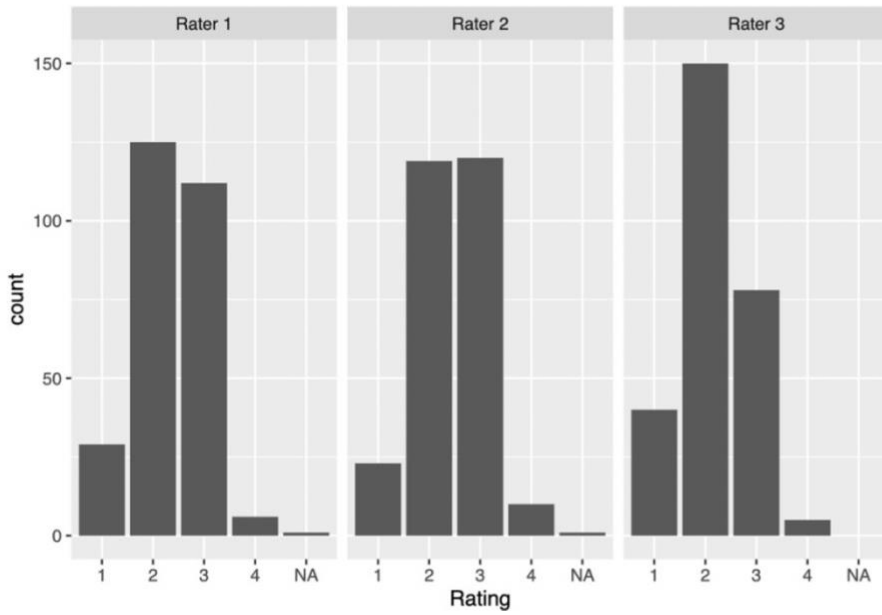


Figure 4.1.2:Barplots showing Distribution of Ratings by different Raters

Based on this bar plot showing distribution of ratings by different raters, we see that the distributions for raters 1 and 2 are very similar to each other. Among the rating scores, the highest values are 2 and 3, while rater 1 tends to have more 2 scores than 3 scores. However, rater3 has different distribution than raters 1 and 2, having extremely high value of 2 scores which looks like a right-skewed distribution. Similar to above approach, after we made the same bar plots using full dataset where NA values are properly handled, we still see the same distributions for the three raters. Therefore, we can get a primary conclusion that rater 3 tends to give lower scores comparing to rater 1 and 2.

4.2 Studying Rater's Agreements Based on The Scores:

	ICC.alldata	ICC.common	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 4.2 ICCs and percent exact agreements for the data with artifacts and the full data set.

Given the ICC values, we can't tell which rater generally disagree most with others on Research Question, Interpret Results, and Text Organization. Also, we can't see any single rubric that stands out from other rubrics that has the lowest agreement among three raters. The percent agreements illustrated by 2-way tables can also be confusing on interpretation. There is no one single rater who would disagree with the other more than half of the time: On 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. When looking at the full dataset, it is not quite the same as artifacts only data. Some rubrics where the raters generally agree on their scores (Critique Design, Initial EDA, Visual Organization) and others having a low agreement.

Because ICC values could not tell us which raters might be contributing to the disagreement, we are also using the 2-way table to count for the ratings of each pair of raters. The percentage will tell us the exact agreement or disagreement on each rubric. A sample interpretation for 2-way table (See Tech Appendix 9-10) is, we see that the rater 1 and rater 2 for the rubric RsrchQ have the same rate in 5 out of 13 of the cases. Even for some artifacts they had different rates, most of them are different by 2 and 3. Only 1 out of 13 of therates are very different. Therefore, we know that for RsrchQ, raters 1 and 2 has not much differences between each other. We see that the rater 2 and rater 3 for the rubric RsrchQ have the same rate in 7 out of 13 of the cases. Even for some artifacts they had different rates, most of them are different by 2 and 3. 0 out of 13 of therates are very different. Therefore, we know that for RsrchQ, raters 3 and 2 has some differences between each other. Interpreting 2-way tables like this, we can see that there are no clear pattern on whether which rater is more likely to disagree with other raters.

4.3 Relationships between Factors and Ratings and How They Interact & Affect Each Other:

The starting models were the seven random intercept models. The final model after adding factors to the single model and using backward elimination to find the model with lowest AIC value. ANOVA table is also used to select best model. We find out that adding fixed effects to the seven rubric-specific models using just the data from the 13 common artifacts that all three raters saw did not improve the performance of our model. Therefore, we are just interpreting the random effect coefficients of our models and we can find that for some of the factors, random effect coefficient is higher comparing to other coefficients under the same rubric.

When trying interactions and new random effects for the seven rubric specific models using all data, the results could be different. To factors InitEDA, RsrchQ, and TxtOrg, adding fixed effects does not improve the performance. However, adding Rater while getting rid of the intercept improves the performance for CritDes, InterpRes, and VisOrg. Similarly, Adding Rater, adding Semester, and removing the intercept improves the performance for SelMeth. For those model factors that got improved, the added factors could be seen as significant.

We also noticed that there are some differences among the models: For InitEDA, RsrchQ and SelMeth, the models are just the simple random-intercept models. We examine each of these 4 models to see if the fixed effects make sense to us and if there are any interactions or additional random effects to consider. After refitting the model and check on the t-statistics, we see the difference across the coefficients for these four factors and decided to keep Rater as an important factor. Adding random effect and perform the model selection again would give us the chosen final model. Since Rater is the only fixed effect, 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 is selected, but no additional fixed interactions or random effects. We also see that including Semester in the model matters according to the t-stats and p-values of the model coefficients. Because the number of random effects actually exceeded the number of observations, we can't fit the random intercept model of Rater or Semester grouped by artifact. The summarized model statistics are in the technical appendix (see page 25-27).

Using Rubric as a random effect grouped by Artifact, we got the final model according to the backward eliminations, ANOVA tables and comparison across AIC and BIC values. The chosen final model that we find includes Semester, Rubric, Raterand the interaction of Rater and Rubric as fixed effects and Rubric and Rater as random effects grouped by Artifact.

	Estimate	Std. Error	t value
(Intercept)	1.7575545	0.11404151	15.4115336
as.factor(Rater)2	0.3660542	0.13918252	2.6300297
as.factor(Rater)3	0.1959088	0.12966636	1.5108686
SemesterS19	-0.1591805	0.07647529	-2.0814634
RubricInitEDA	0.7394940	0.12996076	5.6901329
RubricInterpRes	0.9915148	0.12770767	7.7639406
RubricRsrchQ	0.7261869	0.11793023	6.1577676
RubricSelMeth	0.4106797	0.12470498	3.2932102
RubricTxtOrg	1.0157815	0.12999540	7.8139797
RubricVisOrg	0.6542506	0.13353098	4.8996162
as.factor(Rater)2:RubricInitEDA	-0.2998076	0.15609075	-1.9207264
as.factor(Rater)3:RubricInitEDA	-0.2947319	0.15635201	-1.8850532
as.factor(Rater)2:RubricInterpRes	-0.5132297	0.15348482	-3.3438467
as.factor(Rater)3:RubricInterpRes	-0.7148433	0.15363960	-4.6527283
as.factor(Rater)2:RubricRsrchQ	-0.4874137	0.14722146	-3.3107521
as.factor(Rater)3:RubricRsrchQ	-0.3223799	0.14726517	-2.1891116
as.factor(Rater)2:RubricSelMeth	-0.3863739	0.15030941	-2.5705236
as.factor(Rater)3:RubricSelMeth	-0.3871581	0.14961457	-2.5877033
as.factor(Rater)2:RubricTxtOrg	-0.5510439	0.15646043	-3.5219379
as.factor(Rater)3:RubricTxtOrg	-0.4448937	0.15673122	-2.8385772
as.factor(Rater)2:RubricVisOrg	-0.1048994	0.15861081	-0.6613632
as.factor(Rater)3:RubricVisOrg	-0.2752130	0.15884865	-1.7325485

Figure 4: Final model based on multiple selection methods

In all cases, there is more than one random effect to test (3 for raters,2 for semesters, 7 for rubrics, and 21 for the interaction). We inspect AIC and BIC from anova() tables. If we accept above model as our final model, we can interpret the pieces as follows:

- (0 + as.factor(Rater) | Artifact) + as.factor(Rater)

There is a kind of Rater x Artifact interaction: each Rater's rating on each Artifact differs from what we would expect (from the fixed effects alone) by a small random effect that depends on the Artifact

- Rubric + as.factor(Rater) + as.factor(Rater):Rubric

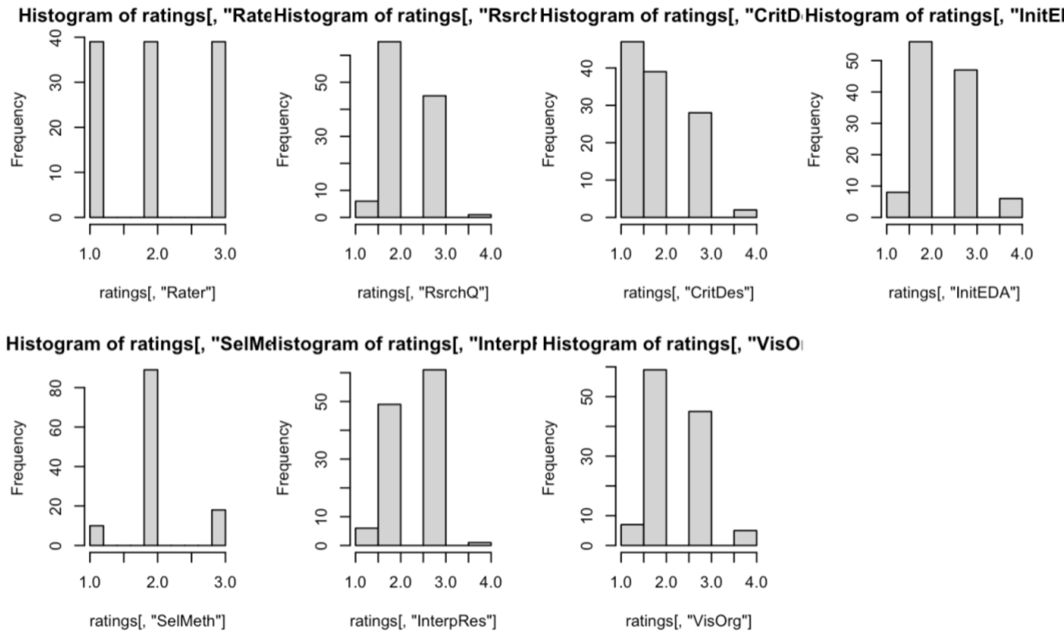
There is a Rater x Rubric interaction: each Rater uses each Rubric in a way that is not like, or even parallel to, other rater's Rubric usage. (we saw that in the facets plot above also).

- (0 + Rubric | Artifact) + Rubric

There is a kind of Rubric x Artifact interaction: There are different average scores on each rubric, but the rubric averages also vary a bit from one Artifact to the next, by a small random effect that depends on Artifact. In all of this, the fact that Rubric scores depend on Artifact (that is, there is a kind of Rubric x Artifact interaction) is what we might expect: the artifacts aren't all of equal quality on each rubric, and so we should expect the average scores on each Rubric to vary from one Artifact to the next.

There are more trouble interpreting the Rater x Rubric interaction and the "kind of" Rater x Artifact interaction. The Rater x Rubric interaction suggest that the Raters are not all interpreting the Rubrics in the same way. The "kind of" Rater x Artifact interaction suggests that the Raters are not interpreting the evidence in the artifacts in the same way. These interactions suggest that perhaps the raters should be trained more, to make the raters' ratings more like each other. For more detailed example, talking about a rating from Rater 1 on the CritDes rubric, we can expect ratings from Rater 2 on InitEDA rubric to be 0.3 units lower and ratings from Rater 3 to be 0.29 units lower on average.

4.4 The Skewness of Distributions on Some Factors:



Looking at the barplots of the whole dataset, it is very necessary to consider raters as a factor in our final model because the frequency is even for three raters.

After generating the percentage table for rating score ==1, we see that there are differences in rating counts is different from the full dataset. Some factors become even more skewed. If we do a further EDA on other rating scores, the result would probably be the same due to the fact that the rating counts for different groups are very different.

According to the percentage marginal tables of all the factors, we can see the difference of factors in each season. This might have something to do with the skewness in distributions, but more about whether to include that part in our model or not. There is no need to include Sex in our model because there are no much difference across difference genders. Similarly, the repeated factor: 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. Therefore, we should not include Sex and Repeated in our model.

5 Discussion

To let the department office know about the performance of students in new general education program, we need our model to tell us what are the factors that have strongest association with ratings. According to our explanatory data analysis on research question 1, our bar plots show very similar and approximately normal distributions of ratings across different rubrics. Two thing that are noticeable: There are two rubrics having the lowest ratings compared to others: CritDes, SelMeth; Rater 3 gives lower ratings than rater 1 and rater 2.

CritDes stands for Critique Design (Given an empirical research question, the student critiques or evaluates to what extent a study design convincingly answers that question), while SelMeth is short of Select Methods (Given a data set and a research question, the student selects appropriate methods to analyze the data). It could be reasonable that these two rubrics receive lowest rating scores because they test students' creativity and the ability to flexibly utilize what they learned from class rather than simply memorizing knowledge from books. We have noticed a relatively large fluctuation in the distribution of ratings given by rater 3. This can be caused by unstable standards that rater 3 used to give ratings to students.

According to the results on research question 2, we see that mostly raters do not agree with each other. For example, they have relatively diverse opinions on Interpret Results and Text Organization rubrics. The two-way table which represents percent agreements gives us information on the extent of disagreement on the other rater per rubric, and it is kind of hard to tell a pattern. Raters do not follow a regular pattern of agreement and disagreement, which makes it hard to interpret the results based on the two-way table. In the future, the department could consider establishing a consistent standard for raters to follow.

Looking at the results from research question 3, we chose the model that includes Rater, Semester, and Rubric are all significant to response variable ratings. Our final model selected by multiple methods was the one including Semester, Rubric, Rater, and interaction term between Rater and Rubric as fixed effect and Rubric with Rater as random effects grouped by Artifact.The reason why we got this final model is that rater 3 has a very significant difference than rater 1 and 2 in terms of ratings. The fixed effect between rater and rubric might suggest that there are some hard rubrics where students need to improve their current learning methods, with distinct distributions than the other rubrics or the inconsistent grading approach and individual difference in raters caused this fixed effect to impact ratings.

According to our research question 4, after constructing histograms for all the factors and computing the margin tables of percentage, we can see that it is correct to not including sex in our model because the percentage is varying too much across different rubrics. Also, some of the skewness could be caused by the different opinions across raters as we previously discovered.

One advantage of our study is that we did a throughout, complete statistical analysis using multilevel models. With the help of our EDA plots and ICC values on intercept-only models, we see that many improvements could be made on students' schedules, educational structures, and overall quality of raters. These methods also provided more sources to interpret our results and give recommendations on current program. By using mixed-effect models, we considered fixed effects and random effects at the same time, along with the interaction term which could not be evaluated in single linear regression models. However, there are also some weaknesses in our study. One weakness is that our data size is really limited by one experiment carried out in a temporary period, and the ratings are only given four numerical values which could not represent the opinions of raters very well. There could be more factors that matter not included in our dataset. If we are about to expand the study in the future, one improvement could be gathering more data from different time periods while using actual scores from 0-100 to represent detailed rubrics from raters when evaluating students' performance from general education courses.

As a recommendation to the department, when trying to improve the efficiency of ratings while reducing the flaws from previous rating approaches, it is important to consider raters, rubrics, and their interactions in general education program. Not only setting consistent grading standards for raters, but also reconsider the rubrics to avoid unfairness or improve the quality of education to achieve an overall improvement on students' grades.

6 References

Kutner, M.H., Nachsheim, C.J., Neter, J. & Li, W. (2005) *Applied Linear Statistical Models, fifth Edition.* NY: McGraw-Hill/Irwin.

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

Technical Appendix

Yueni Wang

12/10/2021

Research Question 1:

> 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?
>
> Load the required libraries and prepare the data:

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.5     v dplyr   1.0.7
## v tidyr   1.1.4     v stringr 1.4.0
## v readr   2.0.1     v forcats 0.5.1

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(arm)
```

```
## Loading required package: MASS

##
## Attaching package: 'MASS'

## The following object is masked from 'package:dplyr':
##
##     select

## Loading required package: Matrix

##
## Attaching package: 'Matrix'

## The following objects are masked from 'package:tidyr':
##
##     expand, pack, unpack

## Loading required package: lme4

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

## Working directory is /Users/yueniwang/Desktop
```

```
library(lme4)
library(latex2exp)
library(MASS)
library(plyr)
```

```
## ------------------------------------------------------------------------------

## You have loaded plyr after dplyr - this is likely to cause problems.
## If you need functions from both plyr and dplyr, please load plyr first, then dplyr:
## library(plyr); library(dplyr)

## ------------------------------------------------------------------------------

##
## Attaching package: 'plyr'

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

## The following object is masked from 'package:purrr':
##
##     compact
```

```
library(ggplot2)
library(performance)
```

```
##
## Attaching package: 'performance'

## The following object is masked from 'package:arm':
##
##     display
```

```
library(kableExtra)
```

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

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

```
library(grid)
library(gridExtra)
```

```
##
## Attaching package: 'gridExtra'

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

```
theme_set(theme_bw())
```

Reading the data file and obtain the summary descriptive statistics (EDA) on this ratings dataset.

```
ratings <- read.csv("ratings.csv",header=T)
tall <-  read.csv("tall.csv",header=T)
summary(ratings)
```

Table 1:

X	Rater	Sample	Overlap	Semester	Sex	RsrchQ	CritDes	InitEDA	SelMeth
1	3	1	5	Fall	M	3	3	2	2
2	3	2	7	Fall	F	3	3	3	3
3	3	3	9	Spring	F	2	1	3	2
4	3	4	8	Spring	M	2	2	2	1
5	3	5	NA	Fall	–	3	3	3	3
6	3	6	NA	Fall	M	2	1	2	2

```
##        X            Rater       Sample          Overlap     Semester
##  Min.   :  1    Min.   :1   Min.   :  1.00   Min.   : 1   Length:117
##  1st Qu.: 30    1st Qu.:1   1st Qu.: 31.00   1st Qu.: 4   Class :character
##  Median : 59    Median :2   Median : 60.00   Median : 7   Mode  :character
##  Mean   : 59    Mean   :2   Mean   : 59.89   Mean   : 7
##  3rd Qu.: 88    3rd Qu.:3   3rd Qu.: 89.00   3rd Qu.:10
##  Max.   :117    Max.   :3   Max.   :118.00   Max.   :13
##                                              NA's   :78
##      Sex                RsrchQ        CritDes         InitEDA
##  Length:117         Min.   :1.00   Min.   :1.000   Min.   :1.000
##  Class :character   1st Qu.:2.00   1st Qu.:1.000   1st Qu.:2.000
##  Mode  :character   Median :2.00   Median :2.000   Median :2.000
##                     Mean   :2.35   Mean   :1.871   Mean   :2.436
##                     3rd Qu.:3.00   3rd Qu.:3.000   3rd Qu.:3.000
##                     Max.   :4.00   Max.   :4.000   Max.   :4.000
##                                    NA's   :1
##     SelMeth        InterpRes        VisOrg          TxtOrg
##  Min.   :1.000   Min.   :1.000   Min.   :1.000   Min.   :1.000
##  1st Qu.:2.000   1st Qu.:2.000   1st Qu.:2.000   1st Qu.:2.000
##  Median :2.000   Median :3.000   Median :2.000   Median :3.000
##  Mean   :2.068   Mean   :2.487   Mean   :2.414   Mean   :2.598
##  3rd Qu.:2.000   3rd Qu.:3.000   3rd Qu.:3.000   3rd Qu.:3.000
##  Max.   :3.000   Max.   :4.000   Max.   :4.000   Max.   :4.000
##                                  NA's   :1
##    Artifact            Repeated
##  Length:117         Min.   :0.0000
##  Class :character   1st Qu.:0.0000
##  Mode  :character   Median :0.0000
##                     Mean   :0.3333
##                     3rd Qu.:1.0000
##                     Max.   :1.0000
##
```

```
#Print first several rows of ratings data:
head(ratings[,1:10]) %>% kbl(booktabs=T,caption=" ") %>% kable_classic()
```

Check for NA values and their occured patterns in the dataset:

```
colSums(is.na(ratings))
```

```
##         X     Rater    Sample   Overlap  Semester       Sex    RsrchQ   CritDes
##         0         0         0        78         0         0         0         1
##   InitEDA   SelMeth InterpRes    VisOrg    TxtOrg  Artifact  Repeated
```

```
##         0          0          0          1          0          0          0
```

```
colSums(is.na(tall))
```

```
##        X     Rater Artifact Repeated Semester      Sex   Rubric   Rating
##        0         0        0        0        0        0        0        2
```

```
ratings %>% filter(is.na(CritDes))
```

```
##    X Rater Sample Overlap Semester Sex RsrchQ CritDes InitEDA SelMeth InterpRes
## 1 44     2     45      NA   Spring   F      2      NA       2       2         2
##   VisOrg TxtOrg Artifact Repeated
## 1      2      3       45        0
```

```
ratings %>% filter(is.na(VisOrg))
```

```
##    X Rater Sample Overlap Semester Sex RsrchQ CritDes InitEDA SelMeth InterpRes
## 1 99     1    100      NA     Fall   F      2       3       2       3         3
##   VisOrg TxtOrg Artifact Repeated
## 1     NA      2      100        0
```

There is not many missing values, both NA values only occured one time. Therefore, we are simply removing the NA values from ratings dataset

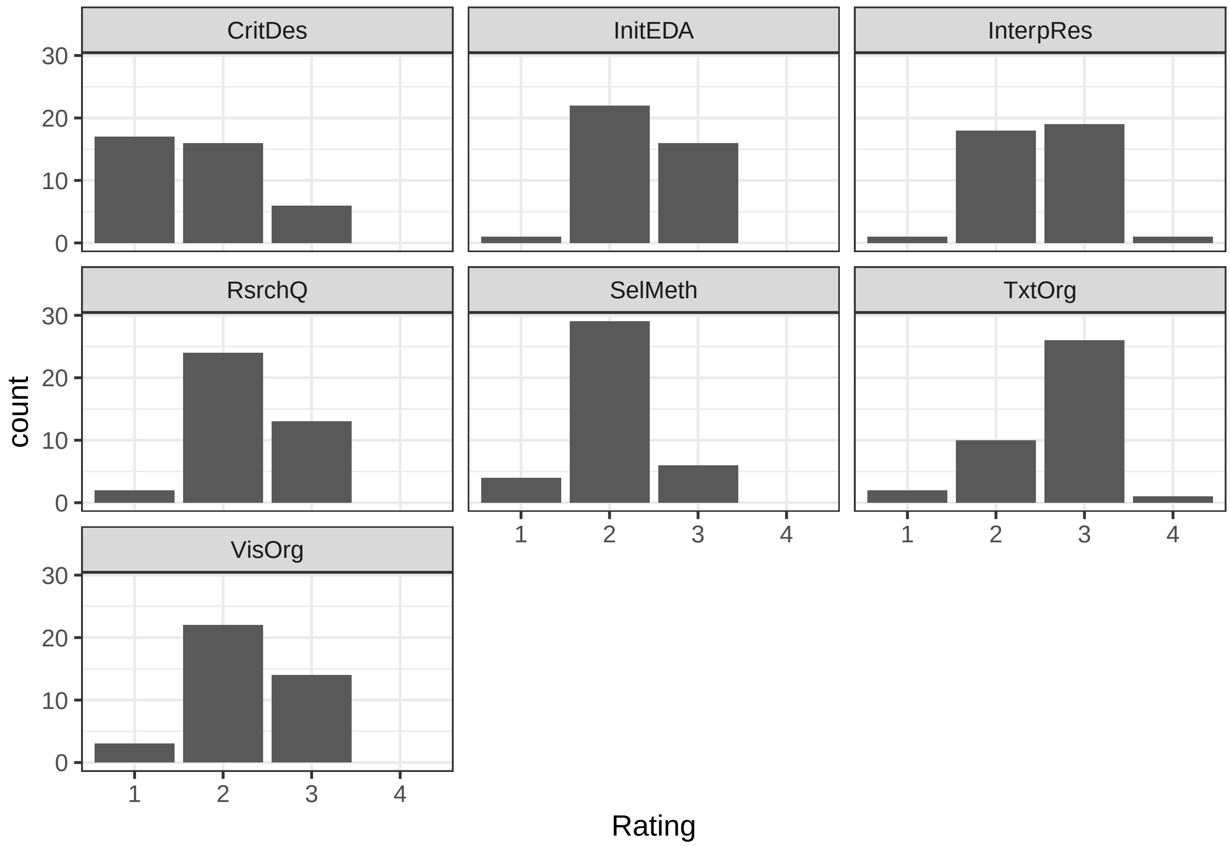
```
ratings <- ratings[!is.na(ratings$CritDes),]
ratings <- ratings[!is.na(ratings$VisOrg),]
```

Make tall and ratings datasets consistent.
Setting the "–" Sex factor to be 0:

```
tall$Rating <- factor(tall$Rating,levels=1:4)
for (i in unique(tall$Rubric)) {
  ratings[,i] <- factor(ratings[,i],levels=1:4)
}
tall$Sex[nchar(tall$Sex)==0] <- "--"

#Extract the reduced data set with the 13 artifacts that all 3 raters saw.
ratings.13 <- ratings[grep("0",ratings$Artifact),]
tall.13 <-  tall[grep("0",tall$Artifact),]
```

Bar plots based on rubrics along with the summary table:

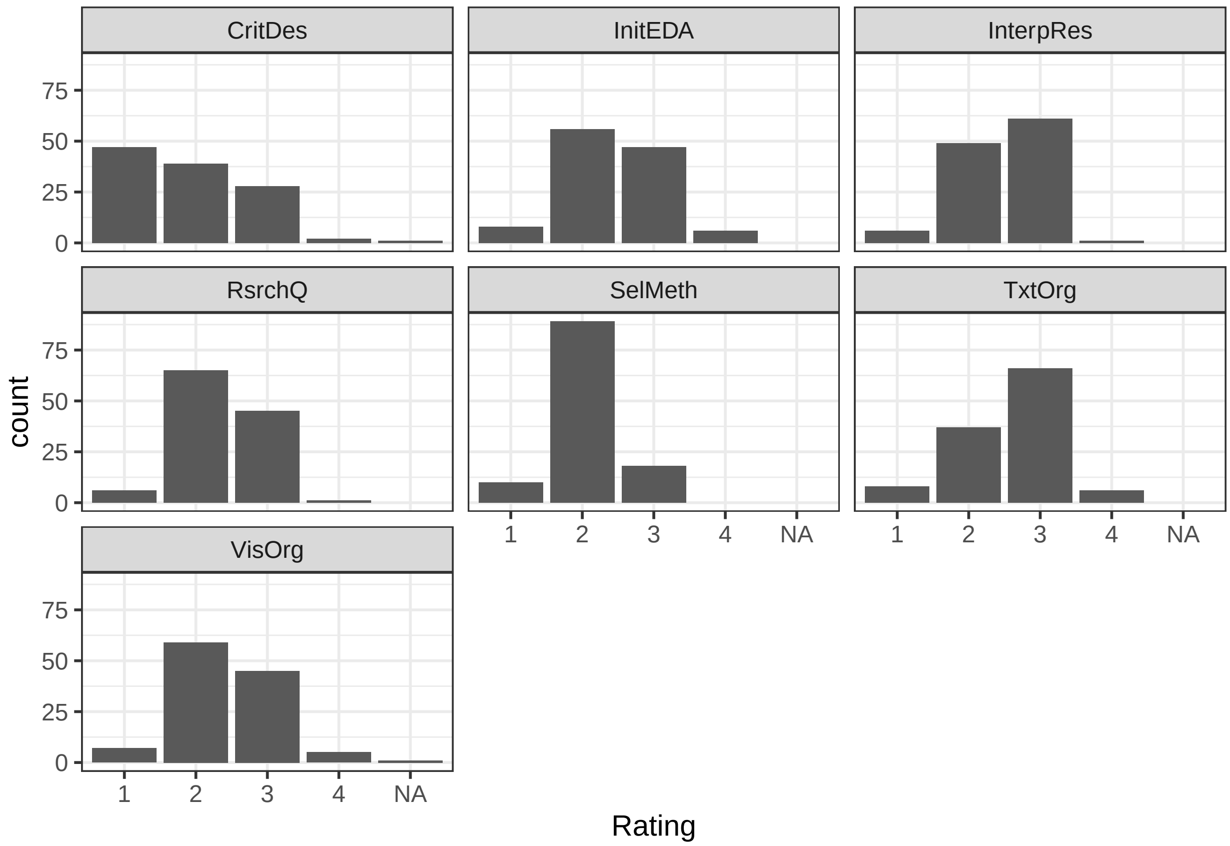
```
## Bar plots for the reduced data set:
g <- ggplot(tall.13,aes(x = Rating)) +
  facet_wrap( ~ Rubric) +
  geom_bar()

g
```




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

tmp
```

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

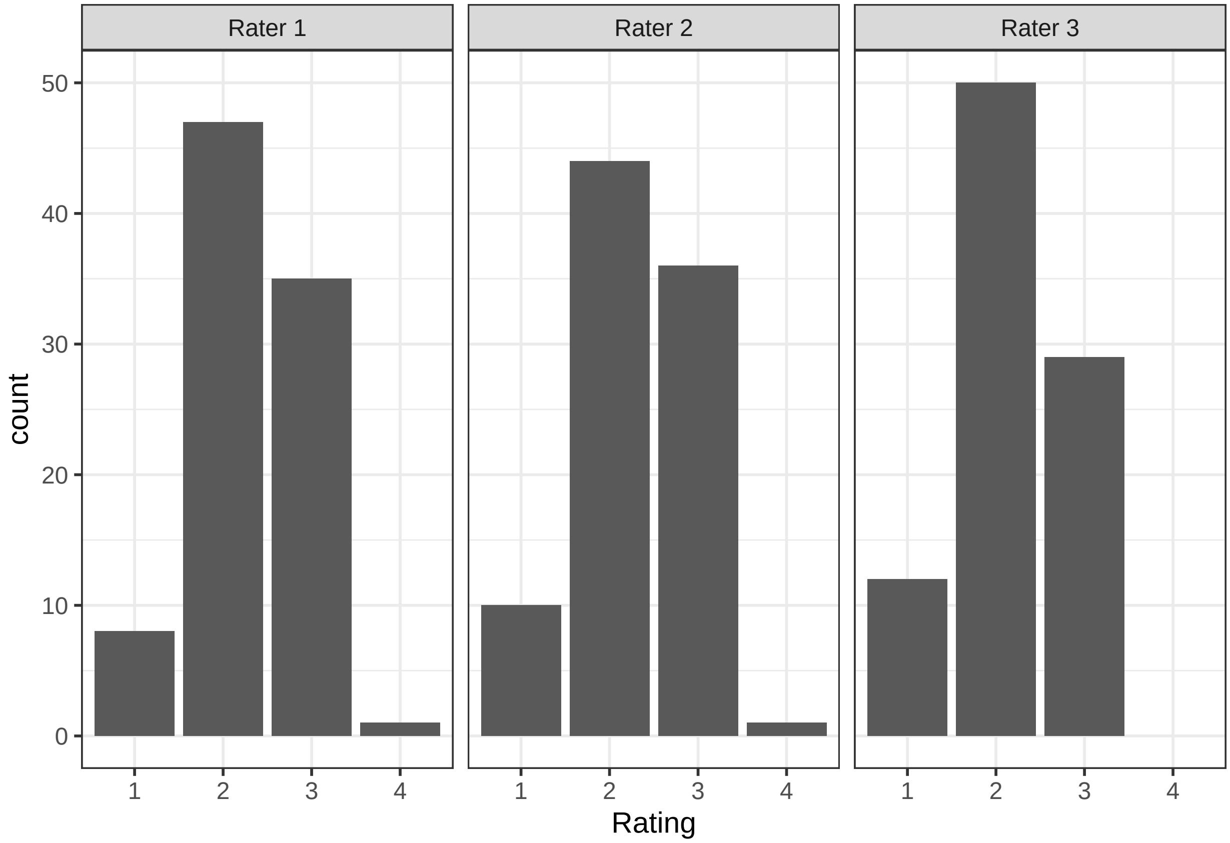
```
## Barplots for full data set
g <- ggplot(tall,aes(x = Rating)) +
  facet_wrap( ~ Rubric) +
  geom_bar()

g
```




```r
## Table of counts again.  A bit pesky since there are NA's...
tmp0 <- lapply(split(tall$Rating,tall$Rubric),summary)
tmp <- data.frame(matrix(0,nrow=5,ncol=7))  ## seven rubrics...
names(tmp) <- names(tmp0)
row.names(tmp) <- c(paste("Rating",1:4),"<NA>")
for (i in names(tmp0)) {
  tmp[,i] <- tmp[,i] + c(tmp0[[i]],0)[1:5]
}

tmp
```

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

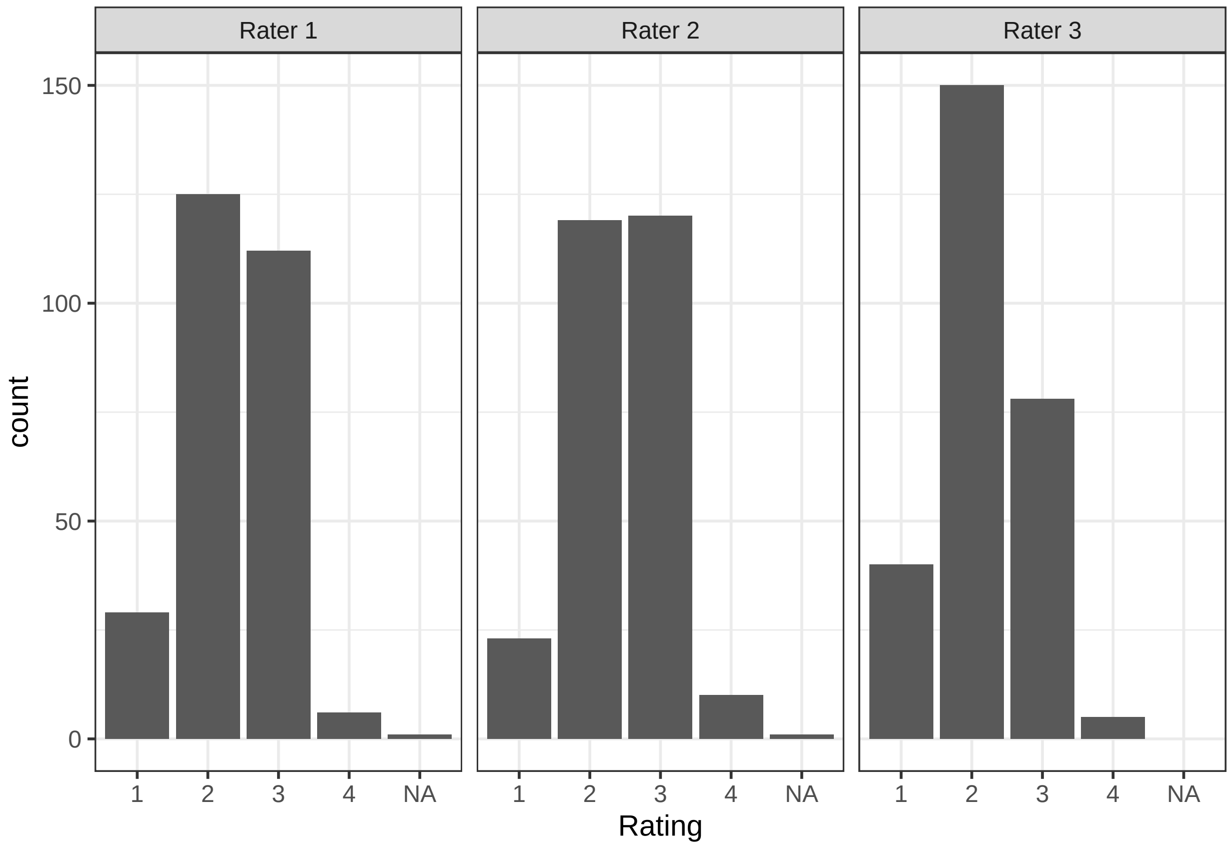
```r
## Needed to make the title of each facet more human-readable...
rater.name <- function(x) { paste("Rater",x) }

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

g
```




```
##
## Corresponding table of counts...
tmp <- data.frame(lapply(split(tall.13$Rating,tall.13$Rater),summary))
row.names(tmp) <- paste("Rating",1:4)
names(tmp) <- paste("Rater",1:3)

tmp
```

```
##          Rater 1 Rater 2 Rater 3
## Rating 1       8      10      12
## Rating 2      47      44      50
## Rating 3      35      36      29
## Rating 4       1       1       0
```

```
##
## Barplots for full data...
g <- ggplot(tall,aes(x = Rating)) +
  facet_wrap( ~ Rater, labeller=labeller(Rater=rater.name)) +
  geom_bar()

g
```




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

```
##          Rater 1 Rater 2 Rater 3
## Rating 1      29      23      40
## Rating 2     125     119     150
## Rating 3     112     120      78
## Rating 4       6      10       5
## <NA>           1       1       0
```

Dealing with the NA values in our data:

```
tall[apply(tall,1,function(x){any(is.na(x))}),]
```

```
##       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[ratings$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
```

To deal with the missing data, when we are fitting our models using Ratings as the response variable, missing values of CritDes and VisOrg will be dropped. We are not dropping the missing Sex value coded the way it is so that the observation is not dropped when Sex is included in modelling. When using the reduced dataset that includes only artifacts that were rated by all three raters, we do not have to do anything about the missing values.

Research Question 2: Studying Rater's Agreements Based on The Scores

For the second research question, we fit random-intercept models of all seven rubrics. compare the ICC values across all three raters on every artifact after fitting our models. ICC is the common correlation among the raters' ratings for each artifact. We treat each artifact as a cluster of three ratings and fit the random-intercept model and fit seven random-intercept models, one for each rubric, then calculate the ICCs.

```
# Measuring the correlations to see if raters agree with each other
common <- tall[grep("O",tall$Artifact),]
head(common)
```

```
##     X Rater Artifact Repeated Semester Sex Rubric Rating
## 1   1     3       O5        1      F19   M RsrchQ      3
## 2   2     3       O7        1      F19   F RsrchQ      3
## 3   3     3       O9        1      S19   F RsrchQ      2
## 4   4     3       O8        1      S19   M RsrchQ      2
## 10 10     3      O10        1      F19   F RsrchQ      2
## 11 11     3      O13        1      F19   M RsrchQ      2
```

```
CritDes.ratings <- common[common$Rubric=="CritDes",]
InitEDA.ratings <- common[common$Rubric=="InitEDA",]
SelMeth.ratings <- common[common$Rubric=="SelMeth",]
InterpRes.ratings <- common[common$Rubric=="InterpRes",]
VisOrg.ratings <- common[common$Rubric=="VisOrg",]
TxtOrg.ratings <- common[common$Rubric=="TxtOrg",]
```

```
CritDes_m=lmer(as.numeric(Rating) ~ 1 + (1|Artifact), data=CritDes.ratings)
InitEDA_m=lmer(as.numeric(Rating) ~ 1 + (1|Artifact), data=InitEDA.ratings)
SelMeth_m=lmer(as.numeric(Rating) ~ 1 + (1|Artifact), data=SelMeth.ratings)
InterpRes_m=lmer(as.numeric(Rating) ~ 1 + (1|Artifact), data=InterpRes.ratings)
VisOrg_m=lmer(as.numeric(Rating) ~ 1 + (1|Artifact), data=VisOrg.ratings)
TxtOrg_m=lmer(as.numeric(Rating) ~ 1 + (1|Artifact), data=TxtOrg.ratings)
```

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

```
##    r2
## r1  1 2 3 4
##   1 0 0 0 0
##   2 1 4 3 0
##   3 1 3 1 0
```

```
##   4 0 0 0 0
```

Because ICC values could not tell us which raters might be contributing to the disagreement, we are also using the 2-way table to count for the ratings of each pair of raters.

We see that the rater 1 and rater 2 for the rubric RsrchQ have the same rate in 5 out of 13 of the cases. Even for some artifacts they had different rates, most of them are different by 2 and 3. Only 1 out of 13 of therates are very different. Therefore, we know that for RsrchQ, raters 1 and 2 has not much differences between each other.

```
raters_2_and_3_on_RsrchQ <- data.frame(r2=repeated$RsrchQ[repeated$Rater==2],r3=repeated$RsrchQ[repeate
a1=repeated$Artifact[repeated$Rater==2], a2=repeated$Artifact[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))
```

```
##    r3
## r2  1 2 3 4
##   1 0 2 0 0
##   2 0 5 2 0
##   3 0 2 2 0
##   4 0 0 0 0
```

We see that the rater 2 and rater 3 for the rubric RsrchQ have the same rate in 7 out of 13 of the cases. Even for some artifacts they had different rates, most of them are different by 2 and 3. 0 out of 13 of therates are very different. Therefore, we know that for RsrchQ, raters 3 and 2 has not much differences between each other.

```
RsrchQ.ratings <- common[common$Rubric=="RsrchQ",]
RsrchQ_m=lmer(as.numeric(Rating) ~ 1 + (1|Artifact), data=RsrchQ.ratings)
summary(RsrchQ_m)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: as.numeric(Rating) ~ 1 + (1 | Artifact)
##    Data: RsrchQ.ratings
##
## REML criterion at convergence: 66.2
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.3025 -0.5987 -0.3276  0.9696  1.6472
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  Artifact (Intercept) 0.05983  0.2446
##  Residual             0.25641  0.5064
## Number of obs: 39, groups:  Artifact, 13
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)   2.2821     0.1057   21.59
```

```
#Calculating ICC value
performance::icc(model=RsrchQ_m)
```

```
## # Intraclass Correlation Coefficient
##
```

```
##      Adjusted ICC: 0.189
##   Conditional ICC: 0.189
```

```
repeated <- ratings[ratings$Repeated==1,]
raters_1_and_2_on_CritDes <- data.frame(r1=repeated$CritDes[repeated$Rater==1],r2=repeated$CritDes[repea
a2=repeated$Artifact[repeated$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))
```

```
##    r2
## r1  1 2 3 4
##   1 3 2 1 0
##   2 2 3 1 0
##   3 0 0 1 0
##   4 0 0 0 0
```

We see that the rater 1 and rater 2 for the rubric CritDes have the same rate in 7 out of 13 of the cases. Even for some artifacts they had different rates, most of them are different by 2 and 3. Only 1 out of 13 of therates are very different. Therefore, we know that for CritDes, raters 1 and 2 has not much differences between each other.

```
raters_2_and_3_on_CritDes <- data.frame(r2=repeated$CritDes[repeated$Rater==2],r3=repeated$CritDes[repea
a1=repeated$Artifact[repeated$Rater==2], a2=repeated$Artifact[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))
```

```
##    r3
## r2  1 2 3 4
##   1 5 0 0 0
##   2 1 3 1 0
##   3 0 2 1 0
##   4 0 0 0 0
```

We see that the rater 3 and rater 2 for the rubric CritDes have the same rate in 9 out of 13 of the cases. Even for some artifacts they had different rates, most of them are different by 2 and 3. Only 1 out of 13 of therates are very different. Therefore, we know that for CritDes, raters 3 and 2 has not much differences between each other.

```
CritDes.ratings <- common[common$Rubric=="CritDes",]
CritDes_m=lmer(as.numeric(Rating) ~ 1 + (1|Artifact), data=CritDes.ratings)
summary(CritDes_m)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: as.numeric(Rating) ~ 1 + (1 | Artifact)
##    Data: CritDes.ratings
##
## REML criterion at convergence: 75.1
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -1.9647 -0.4386 -0.2978  0.5318  2.1987
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
```

```
##  Artifact (Intercept) 0.3091   0.5560
##  Residual             0.2308   0.4804
## Number of obs: 39, groups:  Artifact, 13
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)   1.7179     0.1723   9.969
```

```
#Icc value:
performance::icc(model=CritDes_m)
```

```
## # Intraclass Correlation Coefficient
##
##      Adjusted ICC: 0.573
##   Conditional ICC: 0.573
```

Lower ICC means less agreement accross groups, so here we have higher agreements.

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

```
##    r2
## r1  1 2 3 4
##   1 0 1 0 0
##   2 0 4 0 0
##   3 0 3 5 0
##   4 0 0 0 0
```

For the artifacts which are rated by all three raters, we find that the rater 1 and rater 2 for the rubric InitEDA have the same rate in 9/13 of the cases, even for some artifacts they had different rates, most of them are only r1 = 2 and r2 = 3 or r2 = 2 and r1 = 3 (i.e. the rates are not that different).

```
raters_2_and_3_on_InitEDA <- data.frame(r2=repeated$InitEDA[repeated$Rater==2], r3=repeated$InitEDA[repe
)
r2 <- factor(raters_2_and_3_on_InitEDA$r2,levels=1:4)
r3 <- factor(raters_2_and_3_on_InitEDA$r3,levels=1:4)
(t23 <- table(r2,r3))
```

```
##    r3
## r2  1 2 3 4
##   1 0 0 0 0
##   2 0 8 0 0
##   3 0 2 3 0
##   4 0 0 0 0
```

We find that the rater 2 and rater 3 for the rubric InitEDA have the same rate in 9/13 of the cases, even for some artifacts they had different rates, most of them are only r1 = 2 and r2 = 3 or r2 = 2 and r1 = 3 (i.e. the rates are not that different).So for the rubric InitEDA, the rater 2 and 3, they do not usually disagree with each other.

```
InitEDA.ratings <- common[common$Rubric=="InitEDA",]
InitEDA_m=lmer(as.numeric(Rating) ~ 1 + (1|Artifact), data=InitEDA.ratings)
summary(InitEDA_m)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: as.numeric(Rating) ~ 1 + (1 | Artifact)
##    Data: InitEDA.ratings
##
## REML criterion at convergence: 56.8
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.1670 -0.2504 -0.2504  0.4006  1.6663
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  Artifact (Intercept) 0.1496   0.3867
##  Residual             0.1538   0.3922
## Number of obs: 39, groups:  Artifact, 13
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)   2.3846     0.1243   19.18
```

```
performance::icc(InitEDA_m)
```

```
## # Intraclass Correlation Coefficient
##
##      Adjusted ICC: 0.493
##   Conditional ICC: 0.493
```

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

```
##    r2
## r1   1  2  3  4
##   1  0  0  0  0
##   2  1 10  0  0
##   3  0  0  2  0
##   4  0  0  0  0
```

We find that the rater 1 and rater 2 for the rubric SelMeth have the same rate in 12/13 of the cases, even for the rest of the artifact it had different rates, it is only r2 = 1 and r1 = 2.So for the rubric SelMeth, the rater 1 and 2, they do not usually disagree with each other.

```
raters_2_and_3_on_SelMeth <- data.frame(r2=repeated$SelMeth[repeated$Rater==2], r3=repeated$SelMeth[repe
a1=repeated$Artifact[repeated$Rater==2], a2=repeated$Artifact[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))
```

```
##    r3
## r2  1 2 3 4
##   1 1 0 0 0
##   2 2 7 1 0
##   3 0 1 1 0
##   4 0 0 0 0
```

We find that the rater 2 and rater 3 for the rubric SelMeth have the same rate in 9/13 of the cases, even for some artifacts they had different rates, most of them are only r1 = 2 and r2 = 3 or r2 = 2 and r1 = 3 So for the rubric SelMeth, the rater 2 and 3, they do not usually disagree with each other.

```
SelMeth <- common[common$Rubric=="SelMeth",]
SelMeth_m=lmer(as.numeric(Rating) ~ 1 + (1|Artifact), data=SelMeth.ratings)
summary(SelMeth_m)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: as.numeric(Rating) ~ 1 + (1 | Artifact)
##    Data: SelMeth.ratings
##
## REML criterion at convergence: 50.9
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.11366 -0.03357 -0.03357  0.62101  2.04652
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  Artifact (Intercept) 0.1396   0.3736
##  Residual             0.1282   0.3581
## Number of obs: 39, groups:  Artifact, 13
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)   2.0513     0.1184   17.32
```

```
performance::icc(SelMeth_m)
```

```
## # Intraclass Correlation Coefficient
##
##      Adjusted ICC: 0.521
##   Conditional ICC: 0.521
```

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

```
##    r2
## r1  1 2 3 4
##   1 0 0 0 0
```

```
##   2 0 3 1 1
##   3 0 3 5 0
##   4 0 0 0 0
```

We find that the rater 1 and rater 2 for the rubric InterpRes have the same rate in 8/13 of the cases, even for the rest of the artifact it had different rates, it is only r2 = 1 and r1 = 2. Only one of the artifact had |r1-r2|=2.

```
raters_2_and_3_on_InterpRes <- data.frame(r2=repeated$InterpRes[repeated$Rater==2], r3=repeated$InterpRe
a1=repeated$Artifact[repeated$Rater==2], a2=repeated$Artifact[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))
```

```
##    r3
## r2  1 2 3 4
##   1 0 0 0 0
##   2 1 4 1 0
##   3 0 2 4 0
##   4 0 1 0 0
```

We find that the rater 2 and rater 3 for the rubric InterpRes have the same rate in 8/13 of the cases, even for some artifacts they had different rates, most of them are 2 and 3. Only one of the artifact had |r1-r2|=2.So for the rubric InterpRes, the rater 2 and 3, they do not usually disagree with each other.

```
InterpRes <- common[common$Rubric=="InterpRes",]
InterpRes_m=lmer(as.numeric(Rating) ~ 1 + (1|Artifact), data=InterpRes.ratings)
summary(InterpRes_m)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: as.numeric(Rating) ~ 1 + (1 | Artifact)
##    Data: InterpRes.ratings
##
## REML criterion at convergence: 71.1
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.0965 -0.8061  0.4844  0.7806  2.6635
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  Artifact (Intercept) 0.08405  0.2899
##  Residual             0.28205  0.5311
## Number of obs: 39, groups:  Artifact, 13
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)    2.513      0.117   21.47
```

```
performance::icc(InterpRes_m)
```

```
## # Intraclass Correlation Coefficient
##
##      Adjusted ICC: 0.230
##   Conditional ICC: 0.230
```

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

```
##    r2
## r1  1 2 3 4
##   1 1 0 0 0
##   2 0 4 5 0
##   3 0 1 2 0
##   4 0 0 0 0
```

```r
raters_2_and_3_on_VisOrg <- data.frame(r2=repeated$VisOrg[repeated$Rater==2], r3=repeated$VisOrg[repeat
a1=repeated$Artifact[repeated$Rater==2], a2=repeated$Artifact[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))
```

```
##    r3
## r2  1 2 3 4
##   1 1 0 0 0
##   2 0 5 0 0
##   3 0 3 4 0
##   4 0 0 0 0
```

We find that the rater 1 and rater 2 for the rubric VisOrg have the same rate in 7/13 of the cases, even for the rest of the artifact it had different rates, it is only $r2 = 1$ and $r1 = 2$. Only one of the artifact had $|r1\text{-}r2|$=2.We find that the rater 2 and rater 3 for the rubric VisOrg have the same rate in 8/13 of the cases, even for some artifacts they had different rates, most of them are 2 adn 3. So for the rubric VisOrg, the rater 1 and 2, they do not usually disagree with each other.

```r
VisOrg <- common[common$Rubric=="VisOrg",]
VisOrg_m=lmer(as.numeric(Rating) ~ 1 + (1|Artifact), data=VisOrg.ratings)
summary(VisOrg_m)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: as.numeric(Rating) ~ 1 + (1 | Artifact)
##    Data: VisOrg.ratings
##
## REML criterion at convergence: 60.5
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -1.5168 -0.7176 -0.1341  0.3414  1.7241
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  Artifact (Intercept) 0.2236   0.4729
##  Residual             0.1538   0.3922
## Number of obs: 39, groups:  Artifact, 13
##
```

```
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)   2.2821     0.1454   15.69
```

```
performance::icc(VisOrg_m)
```

```
## # Intraclass Correlation Coefficient
##
##      Adjusted ICC: 0.592
##   Conditional ICC: 0.592
```

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

```
##    r2
## r1  1 2 3 4
##   1 0 0 0 0
##   2 0 2 2 0
##   3 0 1 7 0
##   4 1 0 0 0
```

We find that the rater 1 and rater 2 for the rubric TxtOrg have the same rate in 9/13 of the cases, even for the rest of the artifact it had different rates, it is only r2 = 1 and r1 = 2. Only one of the artifact had |r1-r2|=2.

```
raters_2_and_3_on_TxtOrg <- data.frame(r2=repeated$TxtOrg[repeated$Rater==2], r3=repeated$TxtOrg[repeate
a1=repeated$Artifact[repeated$Rater==2], a2=repeated$Artifact[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))
```

```
##    r3
## r2  1 2 3 4
##   1 0 1 0 0
##   2 1 0 2 0
##   3 0 2 7 0
##   4 0 0 0 0
```

We find that the rater 2 and rater 3 for the rubric TxtOrg have the same rate in 7/13 of the cases, even for some artifacts they had different rates, most of them are 2 and 3. Only onehad |r1-r2|=2.So for the rubric TxtOrg, the rater 2 and 3, they do not usually disagree with each other.

```
TxtOrg <- common[common$Rubric=="TxtOrg",]
TxtOrg_m=lmer(as.numeric(Rating) ~ 1 + (1|Artifact), data=TxtOrg.ratings)
summary(TxtOrg_m)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: as.numeric(Rating) ~ 1 + (1 | Artifact)
##    Data: TxtOrg.ratings
##
```

```
## REML criterion at convergence: 74.6
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.6943 -0.7698  0.3849  0.3849  2.5019
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  Artifact (Intercept) 0.05556  0.2357
##  Residual             0.33333  0.5774
## Number of obs: 39, groups:  Artifact, 13
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)   2.6667     0.1132   23.55
```

```
performance::icc(TxtOrg_m)
```

```
## # Intraclass Correlation Coefficient
##
##      Adjusted ICC: 0.143
##   Conditional ICC: 0.143
```

We found that fitting the lmer model to (1|Raters) does not work here due to the singularity reason. It was our first approach but since we can't get a good ICC value, I prefer models grouped by artifacts.

Research Question 3:Relationships between Factors and Ratings and How They Interact & Affect Each Other:

The general approaches:

- 2(c)(i): Adding fixed effects to the seven rubric-specific models using just the data from the 13 common artifacts that al three raters saw
- 2(c)(ii): Adding fixed effects to the seven rubric-specific models using all the data
- 2(c)(iii): Trying interactions and new random effects for the seven rubric specific models using all the data
- 2(c)(iv): Trying to add fixed effects, interactions, and new random effects to the "combined" model Rating ~ 1 + (0 + Rubric|Artifact), using all the data.

```
library(LMERConvenienceFunctions)
library(RLRsim)

## started by fitting a single model and trying fitLMER.fnc() on it.

tmp <- lmer(as.numeric(Rating) ~ -1 + as.factor(Rater) +
              Semester + Sex + (1|Artifact),
            data=tall.13[tall.13$Rubric=="RsrchQ",],REML=FALSE)

## Since backwards-elimination always involves nested models, use t-tests, F-tests or likelihood ratio 


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

```
## ======================================================
```

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

```
formula(tmp.back_elim)
```

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

```
## The estimates for raters don't look that different from each other, so we can test to see if they ar

tmp.int_only <- update(tmp.back_elim, . ~ . + 1 - as.factor(Rater))

anova(tmp.int_only,tmp.back_elim)
```

```
## Data: tall.13[tall.13$Rubric == "RsrchQ", ]
## Models:
## tmp.int_only: as.numeric(Rating) ~ (1 | Artifact)
## tmp.back_elim: as.numeric(Rating) ~ as.factor(Rater) + (1 | Artifact) - 1
##               npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
## tmp.int_only     3 69.457 74.447 -31.728   63.457
## tmp.back_elim    5 72.018 80.335 -31.009   62.018 1.4391  2      0.487
```

```
anova(tmp.int_only,tmp.back_elim)$"Pr(>Chisq)"[2]
```

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

```
#it looks like the intercept-only model is adequate here (the p-value is much greater than 0.05 or any

Rubric.names <- sort(unique(tall$Rubric))

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

for (i in Rubric.names) {
```

```r
  ## fit each base model
  rubric.data <- tall.13[tall.13$Rubric==i,]
  tmp <- lmer(as.numeric(Rating) ~ -1 + as.factor(Rater) +
              Semester + Sex + (1|Artifact),
          data=rubric.data,REML=FALSE)

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

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

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

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

}
```

```
## ========================================================
## ===              backfitting fixed effects           ===
## ========================================================
## processing model terms of interaction level 1
##   iteration 1
##     p-value for term "Sex" = 0.2229 >= 0.05
##     not part of higher-order interaction
##     removing term
##   iteration 2
##     p-value for term "Semester" = 0.1826 >= 0.05
##     not part of higher-order interaction
##     removing term
## pruning random effects structure ...
##   nothing to prune
## ========================================================
## ===            forwardfitting random effects         ===
## ========================================================
##  ===       random slopes       ===
## ========================================================
## ===            re-backfitting fixed effects          ===
## ========================================================
## processing model terms of interaction level 1
##   all terms of interaction level 1 significant
## resetting REML to TRUE
## pruning random effects structure ...
##   nothing to prune
## ========================================================
## ===              backfitting fixed effects           ===
## ========================================================
```

```
## processing model terms of interaction level 1
##   iteration 1
##     p-value for term "Semester" = 0.8137 >= 0.05
##     not part of higher-order interaction
##     removing term
##   iteration 2
##     p-value for term "Sex" = 0.6429 >= 0.05
##     not part of higher-order interaction
##     removing term
## pruning random effects structure ...
##   nothing to prune
## ======================================================
## ===            forwardfitting random effects       ===
## ======================================================
##  ===       random slopes       ===
## ======================================================
## ===            re-backfitting fixed effects        ===
## ======================================================
## processing model terms of interaction level 1
##   all terms of interaction level 1 significant
## resetting REML to TRUE
## pruning random effects structure ...
##   nothing to prune
## ======================================================
## ===              backfitting fixed effects         ===
## ======================================================
## processing model terms of interaction level 1
##   iteration 1
##     p-value for term "Semester" = 0.8294 >= 0.05
##     not part of higher-order interaction
##     removing term
##   iteration 2
##     p-value for term "Sex" = 0.2947 >= 0.05
##     not part of higher-order interaction
##     removing term
## pruning random effects structure ...
##   nothing to prune
## ======================================================
## ===            forwardfitting random effects       ===
## ======================================================
##  ===       random slopes       ===
## ======================================================
## ===            re-backfitting fixed effects        ===
## ======================================================
## processing model terms of interaction level 1
##   all terms of interaction level 1 significant
## resetting REML to TRUE
## pruning random effects structure ...
##   nothing to prune
## ======================================================
## ===              backfitting fixed effects         ===
## ======================================================
## processing model terms of interaction level 1
##   iteration 1
```

```
##     p-value for term "Semester" = 0.7355 >= 0.05
##     not part of higher-order interaction
##     removing term
##   iteration 2
##     p-value for term "Sex" = 0.279 >= 0.05
##     not part of higher-order interaction
##     removing term
## pruning random effects structure ...
##   nothing to prune
## =========================================================
## ===            forwardfitting random effects       ===
## =========================================================
##  ===       random slopes       ===
## =========================================================
## ===            re-backfitting fixed effects        ===
## =========================================================
## processing model terms of interaction level 1
##   all terms of interaction level 1 significant
## resetting REML to TRUE
## pruning random effects structure ...
##   nothing to prune
## =========================================================
## ===              backfitting fixed effects         ===
## =========================================================
## processing model terms of interaction level 1
##   iteration 1
##     p-value for term "Sex" = 0.9383 >= 0.05
##     not part of higher-order interaction
##     removing term
##   iteration 2
##     p-value for term "Semester" = 0.4287 >= 0.05
##     not part of higher-order interaction
##     removing term
## pruning random effects structure ...
##   nothing to prune
## =========================================================
## ===            forwardfitting random effects       ===
## =========================================================
##  ===       random slopes       ===
## =========================================================
## ===            re-backfitting fixed effects        ===
## =========================================================
## processing model terms of interaction level 1
##   all terms of interaction level 1 significant
## resetting REML to TRUE
## pruning random effects structure ...
##   nothing to prune
## =========================================================
## ===              backfitting fixed effects         ===
## =========================================================
## processing model terms of interaction level 1
##   iteration 1
##     p-value for term "Semester" = 0.5358 >= 0.05
##     not part of higher-order interaction
```

```
##     removing term
##   iteration 2
##     p-value for term "Sex" = 0.1319 >= 0.05
##     not part of higher-order interaction
##     removing term
## pruning random effects structure ...
##   nothing to prune
## =======================================================
## ===            forwardfitting random effects       ===
## =======================================================
##  ===       random slopes       ===
## =======================================================
## ===            re-backfitting fixed effects        ===
## =======================================================
## processing model terms of interaction level 1
##   all terms of interaction level 1 significant
## resetting REML to TRUE
## pruning random effects structure ...
##   nothing to prune
## =======================================================
## ===              backfitting fixed effects         ===
## =======================================================
## processing model terms of interaction level 1
##   iteration 1
##     p-value for term "Semester" = 0.1922 >= 0.05
##     not part of higher-order interaction
##     removing term
##   iteration 2
##     p-value for term "Sex" = 0.1078 >= 0.05
##     not part of higher-order interaction
##     removing term
## pruning random effects structure ...
##   nothing to prune
## =======================================================
## ===            forwardfitting random effects       ===
## =======================================================
##  ===       random slopes       ===
## =======================================================
## ===            re-backfitting fixed effects        ===
## =======================================================
## processing model terms of interaction level 1
##   all terms of interaction level 1 significant
## resetting REML to TRUE
## pruning random effects structure ...
##   nothing to prune
```

```
## see what "final models" we got...
model.formula.13
```

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

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

2(c)(ii): Adding fixed effects to the seven rubric-specific models using all the data

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

#We want to use the same data set for every model fit and model comparison. I am going to eliminate by

tall[c(161,684),] ## just to check that these are the rows with missing ratings...
```

```
##       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.nonmissing <- tall[-c(161,684),]  ## now delete them...

tall.nonmissing[tall.nonmissing$Sex=="--",]  ## check which rows will be eliminated
```

```
##       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.nonmissing[tall.nonmissing$Sex!="--",] ## eliminate them

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

for (i in Rubric.names) {

  ## fit each base model
  rubric.data <- 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)
```

```r
  ## 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)

}
```

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

## ========================================================
## ===              backfitting fixed effects          ===
## ========================================================
## processing model terms of interaction level 1
##   iteration 1
##     p-value for term "Semester" = 0.7154 >= 0.05
##     not part of higher-order interaction
##     removing term
##   iteration 2
##     p-value for term "Sex" = 0.5297 >= 0.05
##     not part of higher-order interaction
##     removing term
## pruning random effects structure ...
##   nothing to prune
## ========================================================
## ===            forwardfitting random effects        ===
## ========================================================
##  ===       random slopes       ===
## ========================================================
## ===            re-backfitting fixed effects         ===
## ========================================================
## processing model terms of interaction level 1
##   all terms of interaction level 1 significant
## resetting REML to TRUE
## pruning random effects structure ...
##   nothing to prune

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

## Warning in fitLMER.fnc(tmp, set.REML.FALSE = TRUE, log.file.name = FALSE): Argument "ran.effects" is
## TRUE

## ========================================================
## ===              backfitting fixed effects          ===
## ========================================================
## processing model terms of interaction level 1
```

```
##   iteration 1
##     p-value for term "Semester" = 0.8802 >= 0.05
##     not part of higher-order interaction
##     removing term
##   iteration 2
##     p-value for term "Sex" = 0.7402 >= 0.05
##     not part of higher-order interaction
##     removing term
## pruning random effects structure ...
##   nothing to prune
## ========================================================
## ===            forwardfitting random effects       ===
## ========================================================
##  ===       random slopes       ===
## ========================================================
## ===            re-backfitting fixed effects        ===
## ========================================================
## processing model terms of interaction level 1
##   all terms of interaction level 1 significant
## resetting REML to TRUE
## pruning random effects structure ...
##   nothing to prune

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

## Warning in fitLMER.fnc(tmp, set.REML.FALSE = TRUE, log.file.name = FALSE): Argument "ran.effects" is
## TRUE

## ========================================================
## ===              backfitting fixed effects         ===
## ========================================================
## processing model terms of interaction level 1
##   iteration 1
##     p-value for term "Sex" = 0.608 >= 0.05
##     not part of higher-order interaction
##     removing term
##   iteration 2
##     p-value for term "Semester" = 0.5312 >= 0.05
##     not part of higher-order interaction
##     removing term
## pruning random effects structure ...
##   nothing to prune
## ========================================================
## ===            forwardfitting random effects       ===
## ========================================================
##  ===       random slopes       ===
## ========================================================
## ===            re-backfitting fixed effects        ===
## ========================================================
## processing model terms of interaction level 1
##   all terms of interaction level 1 significant
## resetting REML to TRUE
## pruning random effects structure ...
##   nothing to prune

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

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

```
## ======================================================
## ===              backfitting fixed effects          ===
## ======================================================
## processing model terms of interaction level 1
##   iteration 1
##     p-value for term "Sex" = 0.6166 >= 0.05
##     not part of higher-order interaction
##     removing term
##   iteration 2
##     p-value for term "Semester" = 0.3987 >= 0.05
##     not part of higher-order interaction
##     removing term
## pruning random effects structure ...
##   nothing to prune
## ======================================================
## ===            forwardfitting random effects        ===
## ======================================================
##  ===       random slopes       ===
## ======================================================
## ===            re-backfitting fixed effects         ===
## ======================================================
## processing model terms of interaction level 1
##   all terms of interaction level 1 significant
## resetting REML to TRUE
## pruning random effects structure ...
##   nothing to prune
```

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

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

```
## ======================================================
## ===              backfitting fixed effects          ===
## ======================================================
## processing model terms of interaction level 1
##   iteration 1
##     p-value for term "Sex" = 0.1935 >= 0.05
##     not part of higher-order interaction
##     removing term
## pruning random effects structure ...
##   nothing to prune
## ======================================================
## ===            forwardfitting random effects        ===
## ======================================================
##  ===       random slopes       ===
## ======================================================
## ===            re-backfitting fixed effects         ===
## ======================================================
## processing model terms of interaction level 1
##   all terms of interaction level 1 significant
## resetting REML to TRUE
## pruning random effects structure ...
```

```
##   nothing to prune
## refitting model(s) with ML (instead of REML)
## Warning in fitLMER.fnc(tmp, set.REML.FALSE = TRUE, log.file.name = FALSE): Argument "ran.effects" is
## TRUE
## ======================================================
## ===              backfitting fixed effects         ===
## ======================================================
## processing model terms of interaction level 1
##   iteration 1
##     p-value for term "Sex" = 0.5041 >= 0.05
##     not part of higher-order interaction
##     removing term
##   iteration 2
##     p-value for term "Semester" = 0.205 >= 0.05
##     not part of higher-order interaction
##     removing term
## pruning random effects structure ...
##   nothing to prune
## ======================================================
## ===            forwardfitting random effects       ===
## ======================================================
##  ===       random slopes       ===
## ======================================================
## ===            re-backfitting fixed effects        ===
## ======================================================
## processing model terms of interaction level 1
##   all terms of interaction level 1 significant
## resetting REML to TRUE
## pruning random effects structure ...
##   nothing to prune
## refitting model(s) with ML (instead of REML)
## Warning in fitLMER.fnc(tmp, set.REML.FALSE = TRUE, log.file.name = FALSE): Argument "ran.effects" is
## TRUE
## ======================================================
## ===              backfitting fixed effects         ===
## ======================================================
## processing model terms of interaction level 1
##   iteration 1
##     p-value for term "Semester" = 0.2158 >= 0.05
##     not part of higher-order interaction
##     removing term
##   iteration 2
##     p-value for term "Sex" = 0.3523 >= 0.05
##     not part of higher-order interaction
##     removing term
## pruning random effects structure ...
##   nothing to prune
## ======================================================
## ===            forwardfitting random effects       ===
## ======================================================
##  ===       random slopes       ===
```

```
## ======================================================
## ===            re-backfitting fixed effects        ===
## ======================================================
## processing model terms of interaction level 1
##   all terms of interaction level 1 significant
## resetting REML to TRUE
## pruning random effects structure ...
##   nothing to prune

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

```
## see what "final models" we got...
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
```

2(c)(iii): Trying interactions and new random effects for the seven rubric specific models using all the data

```
## refit the model and check on the t-statistics -- do all the variables matter?yes
fla <- formula(model.formula.alldata[["SelMeth"]])
tmp <- lmer(fla,data=tall.nonmissing[tall.nonmissing$Rubric=="SelMeth",])
round(summary(tmp)$coef,2)  ## fixed effects and their t-values
```

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

```
## now check to make sure we really need "Rater" as a factor...
tmp.single_intercept <- update(tmp, . ~ . + 1 - as.factor(Rater))
anova(tmp.single_intercept,tmp)
```

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

## Data: tall.nonmissing[tall.nonmissing$Rubric == "SelMeth", ]
```

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

```
## now let's check for fixed-effect interactions... Since only Rater and Semester are involved, we only

tmp.fixed_interactions <- update(tmp, . ~ . + as.factor(Rater)*Semester - Semester)

anova(tmp,tmp.fixed_interactions)
```

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

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

```
## Looks like the fixed-effect interactions are not needed; again we keep "tmp" as our best model so fa

## Finally we check for random effects.  We should only add random effects that are also present as fix

## Testing (Semester|Artifact)...

#m0 <- tmp                                       ## Null hypothesis
#mA <- update(m0, . ~ . + (Semester|Artifact))   ## Alternative hypotheses
#m  <- update(mA, . ~ . - (1|Artifact))          ## Model with only the new R.E.

#exactRLRT(m0=m0,mA=mA,m=m)

##lmer() cannot fit a model.  Thus, the model as.numeric(Rating) ~ -1 + as.factor(Rater) + Semester +
##                            (1 | Artifact) + (Semseter | Artifact) isn't even possible, so no testing



## Testng (as.factor(Rater)|Artifact)

#m0 <- tmp                                       ## Null hypothesis
#mA <- update(m0, . ~ . + (as.factor(Rater)|Artifact))   ## Alternative hypotheses
#m  <- update(mA, . ~ . - (1|Artifact))          ## Model with only the new R.E.

#exactRLRT(m0=m0,mA=mA,m=m)

## Same thing happened!  Again, the model as.numeric(Rating) ~ -1 + as.factor(Rater) + Semester +
##                            (1 | Artifact) + (as.factor(Rater) | Artifact) isn't even possible, so no 


## Thus, we weren't able to add or take away anything from the model "tmp",
## so this is our final model for SelMeth:
```

```r
summary(tmp)
```

```
## 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
```

2(c)(iv): Trying to add fixed effects, interactions, and new random effects to the "combined" model Rating ~ 1 + (0 + Rubric|Artifact), using all the data.

```r
## Start with the "combined" intercept-only model...

comb.0 <- lmer(as.numeric(Rating) ~ 1 + (0 + Rubric | Artifact),
               data=tall.nonmissing)
```

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

```r
summary(comb.0)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: as.numeric(Rating) ~ 1 + (0 + Rubric | Artifact)
##    Data: tall.nonmissing
##
## REML criterion at convergence: 1471.7
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.0218 -0.4940 -0.0753  0.5271  3.7759
##
```

```
## Random effects:
##  Groups   Name            Variance Std.Dev. Corr
##  Artifact RubricCritDes   0.64070  0.8004
##           RubricInitEDA   0.38288  0.6188   0.26
##           RubricInterpRes 0.25658  0.5065   0.00 0.79
##           RubricRsrchQ    0.17398  0.4171   0.38 0.50 0.74
##           RubricSelMeth   0.09619  0.3102   0.56 0.37 0.41 0.26
##           RubricTxtOrg    0.40425  0.6358   0.03 0.69 0.80 0.64 0.24
##           RubricVisOrg    0.31878  0.5646   0.17 0.78 0.76 0.60 0.29 0.79
##  Residual                 0.19477  0.4413
## Number of obs: 810, groups:  Artifact, 90
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)  2.23210    0.04013   55.63
## optimizer (nloptwrap) convergence code: 0 (OK)
## boundary (singular) fit: see ?isSingular
```

```
## R complains that we have a "boundary (singular) fit", i.e. the variance-covariance matrix for the ra
##
## Some of the random effects are highly correlated with one another.  We can see this in the "Random e
##
## * The random effects for VisOrg and TxtOrg seem highly correlated with
##   each other and with everything except for the rand. effect for SelMeth
##
## * The random effects for InterpRes and InitEDA are highly correlated
##
## * The random effects for RsrchQ and InterpRes are highly correlated
##

## Try adding fixed effects with no interactions...

comb.full <- update(comb.0, . ~ . + as.factor(Rater) + Semester +
                      Sex + Repeated + Rubric)

summary(comb.full)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: as.numeric(Rating) ~ (0 + Rubric | Artifact) + as.factor(Rater) +
##     Semester + Sex + Repeated + Rubric
##    Data: tall.nonmissing
##
## REML criterion at convergence: 1429.6
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.1091 -0.5065 -0.0178  0.5242  3.7932
##
## Random effects:
##  Groups   Name            Variance Std.Dev. Corr
##  Artifact RubricCritDes   0.55311  0.7437
##           RubricInitEDA   0.35239  0.5936   0.47
##           RubricInterpRes 0.17512  0.4185   0.23 0.75
##           RubricRsrchQ    0.16997  0.4123   0.58 0.44 0.71
##           RubricSelMeth   0.06816  0.2611   0.39 0.60 0.74 0.41
```

```
##           RubricTxtOrg    0.26339  0.5132   0.34 0.62 0.70 0.56 0.67
##           RubricVisOrg    0.25809  0.5080   0.35 0.73 0.68 0.52 0.41 0.76
##  Residual                 0.18916  0.4349
## Number of obs: 810, groups:  Artifact, 90
##
## Fixed effects:
##                     Estimate Std. Error t value
## (Intercept)         2.013748   0.109103  18.457
## as.factor(Rater)2   0.001977   0.054887   0.036
## as.factor(Rater)3 -0.174867   0.055045  -3.177
## SemesterS19       -0.175017   0.087850  -1.992
## SexM               0.010506   0.081271   0.129
## Repeated          -0.073586   0.098522  -0.747
## RubricInitEDA      0.547054   0.095710   5.716
## RubricInterpRes    0.587091   0.100893   5.819
## RubricRsrchQ       0.460875   0.087516   5.266
## RubricSelMeth      0.164863   0.094265   1.749
## RubricTxtOrg       0.692880   0.099523   6.962
## RubricVisOrg       0.530182   0.099136   5.348
##
## Correlation of Fixed Effects:
##             (Intr) a.(R)2 a.(R)3 SmsS19 SexM   Repetd RbIEDA RbrcIR RbrcRQ
## as.fctr(R)2 -0.245
## as.fctr(R)3 -0.237  0.499
## SemesterS19 -0.361  0.008  0.000
## SexM        -0.398 -0.026 -0.035  0.302
## Repeated    -0.154  0.001 -0.003  0.079  0.009
## RubrcIntEDA -0.552 -0.001  0.000 -0.001  0.000  0.007
## RbrcIntrpRs -0.660 -0.001  0.000 -0.001  0.000 -0.009  0.734
## RubrcRsrchQ -0.626 -0.001  0.000 -0.001  0.000 -0.039  0.585  0.756
## RubricSlMth -0.689 -0.001  0.000 -0.001  0.000 -0.088  0.659  0.777  0.689
## RubrcTxtOrg -0.611 -0.001  0.000 -0.001  0.000  0.005  0.674  0.751  0.682
## RubricVsOrg -0.607 -0.001 -0.001 -0.002 -0.001 -0.021  0.715  0.745  0.668
##             RbrcSM RbrcTO
## as.fctr(R)2
## as.fctr(R)3
## SemesterS19
## SexM
## Repeated
## RubrcIntEDA
## RbrcIntrpRs
## RubrcRsrchQ
## RubricSlMth
## RubrcTxtOrg  0.725
## RubricVsOrg  0.680  0.750
```

```
comb.back_elim <- fitLMER.fnc(comb.full, log.file.name = FALSE)
```

```
## Warning in fitLMER.fnc(comb.full, log.file.name = FALSE): Argument "ran.effects" is empty, which mea
## TRUE
```

```
## ======================================================
## ===              backfitting fixed effects         ===
## ======================================================
## processing model terms of interaction level 1
```

```
##   iteration 1
##     p-value for term "Sex" = 0.887 >= 0.05
##     not part of higher-order interaction
```

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

```
##     removing term
##   iteration 2
##     p-value for term "Repeated" = 0.0919 >= 0.05
##     not part of higher-order interaction
```

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

```
##     removing term
## pruning random effects structure ...
##   nothing to prune
## ==========================================================
## ===            forwardfitting random effects       ===
## ==========================================================
##  ===       random slopes       ===
## ==========================================================
## ===            re-backfitting fixed effects        ===
## ==========================================================
## processing model terms of interaction level 1
##   all terms of interaction level 1 significant
## resetting REML to TRUE
```

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

```
## pruning random effects structure ...
##   nothing to prune
```

```
summary(comb.back_elim)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: as.numeric(Rating) ~ (0 + Rubric | Artifact) + as.factor(Rater) +
##     Semester + Rubric
##    Data: tall.nonmissing
##
## REML criterion at convergence: 1424.1
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.1200 -0.5125 -0.0173  0.5302  3.7752
##
## Random effects:
##  Groups   Name            Variance Std.Dev. Corr
##  Artifact RubricCritDes   0.55495  0.7449
##           RubricInitEDA   0.35064  0.5921   0.47
##           RubricInterpRes 0.16892  0.4110   0.23 0.75
##           RubricRsrchQ    0.16777  0.4096   0.59 0.44 0.70
##           RubricSelMeth   0.06499  0.2549   0.40 0.60 0.74 0.40
##           RubricTxtOrg    0.25615  0.5061   0.33 0.61 0.69 0.55 0.66
##           RubricVisOrg    0.25894  0.5089   0.35 0.73 0.68 0.52 0.41 0.75
##  Residual                 0.18934  0.4351
## Number of obs: 810, groups:  Artifact, 90
##
## Fixed effects:
```

```
##                     Estimate Std. Error t value
## (Intercept)        2.0084130  0.0987610  20.336
## as.factor(Rater)2  0.0003231  0.0547446   0.006
## as.factor(Rater)3 -0.1771062  0.0548892  -3.227
## SemesterS19       -0.1730357  0.0826927  -2.093
## RubricInitEDA      0.5474747  0.0957148   5.720
## RubricInterpRes    0.5864544  0.1008618   5.814
## RubricRsrchQ       0.4584082  0.0874179   5.244
## RubricSelMeth      0.1590770  0.0937771   1.696
## RubricTxtOrg       0.6930033  0.0995479   6.962
## RubricVisOrg       0.5289027  0.0990973   5.337
##
## Correlation of Fixed Effects:
##             (Intr) a.(R)2 a.(R)3 SmsS19 RbIEDA RbrcIR RbrcRQ RbrcSM RbrcTO
## as.fctr(R)2 -0.281
## as.fctr(R)3 -0.277  0.499
## SemesterS19 -0.264  0.017  0.011
## RubrcIntEDA -0.610 -0.001  0.000 -0.002
## RbrcIntrpRs -0.735 -0.001  0.000  0.000  0.734
## RubrcRsrchQ -0.701 -0.001  0.000  0.002  0.586  0.756
## RubricSlMth -0.782  0.000  0.000  0.006  0.662  0.779  0.688
## RubrcTxtOrg -0.679 -0.001  0.000 -0.001  0.674  0.751  0.682  0.728
## RubricVsOrg -0.675 -0.001 -0.001  0.000  0.715  0.745  0.667  0.681  0.750
## optimizer (nloptwrap) convergence code: 0 (OK)
## boundary (singular) fit: see ?isSingular
```

```r
## The final model fit is a boundary fit again, but we will proceed to try interactions

comb.inter <- update(comb.back_elim, . ~ . + as.factor(Rater)*Semester*Rubric)
```

```
## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv, :
## Model failed to converge with max|grad| = 0.00431172 (tol = 0.002, component 1)
```

```r
## This didn't quite converge, so we will try switching optimizers and increasing
## the number of iterations allowed...

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

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

```r
## it takes a few seconds to fit, but at least we got a converged fit.

summary(comb.inter.u)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: as.numeric(Rating) ~ (0 + Rubric | Artifact) + as.factor(Rater) +
##     Semester + Rubric + as.factor(Rater):Semester + as.factor(Rater):Rubric +
##     Semester:Rubric + as.factor(Rater):Semester:Rubric
##    Data: tall.nonmissing
## Control: lmerControl(optimizer = "bobyqa", optCtrl = list(maxfun = 2e+05))
##
## REML criterion at convergence: 1424.4
##
```

```
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.9141 -0.5141 -0.0653  0.5023  3.6609
##
## Random effects:
##  Groups   Name            Variance Std.Dev. Corr
##  Artifact RubricCritDes   0.48550  0.6968
##           RubricInitEDA   0.35257  0.5938   0.42
##           RubricInterpRes 0.14619  0.3824   0.32 0.80
##           RubricRsrchQ    0.16444  0.4055   0.66 0.43 0.72
##           RubricSelMeth   0.06297  0.2509   0.45 0.64 0.78 0.49
##           RubricTxtOrg    0.25441  0.5044   0.44 0.65 0.67 0.60 0.62
##           RubricVisOrg    0.25527  0.5052   0.35 0.73 0.68 0.57 0.35 0.76
##  Residual                 0.18839  0.4340
## Number of obs: 810, groups:  Artifact, 90
##
## Fixed effects:
##                                               Estimate Std. Error t value
## (Intercept)                                   1.739538   0.136568  12.738
## as.factor(Rater)2                             0.302995   0.155107   1.953
## as.factor(Rater)3                             0.237851   0.155863   1.526
## SemesterS19                                  -0.129077   0.250318  -0.516
## RubricInitEDA                                 0.765215   0.165241   4.631
## RubricInterpRes                               0.979228   0.162160   6.039
## RubricRsrchQ                                  0.710427   0.147386   4.820
## RubricSelMeth                                 0.462750   0.155274   2.980
## RubricTxtOrg                                  1.011251   0.160899   6.285
## RubricVisOrg                                  0.647869   0.166603   3.889
## as.factor(Rater)2:SemesterS19                 0.268014   0.303883   0.882
## as.factor(Rater)3:SemesterS19                -0.072789   0.301026  -0.242
## as.factor(Rater)2:RubricInitEDA              -0.325018   0.204108  -1.592
## as.factor(Rater)3:RubricInitEDA              -0.374190   0.205354  -1.822
## as.factor(Rater)2:RubricInterpRes            -0.469281   0.201051  -2.334
## as.factor(Rater)3:RubricInterpRes            -0.711515   0.202316  -3.517
## as.factor(Rater)2:RubricRsrchQ               -0.447050   0.189326  -2.361
## as.factor(Rater)3:RubricRsrchQ               -0.474411   0.190681  -2.488
## as.factor(Rater)2:RubricSelMeth              -0.301450   0.193678  -1.556
## as.factor(Rater)3:RubricSelMeth              -0.365656   0.194970  -1.875
## as.factor(Rater)2:RubricTxtOrg               -0.449164   0.200927  -2.235
## as.factor(Rater)3:RubricTxtOrg               -0.407754   0.202209  -2.016
## as.factor(Rater)2:RubricVisOrg                0.009042   0.205059   0.044
## as.factor(Rater)3:RubricVisOrg               -0.287443   0.206299  -1.393
## SemesterS19:RubricInitEDA                    -0.050212   0.301475  -0.167
## SemesterS19:RubricInterpRes                   0.127813   0.295706   0.432
## SemesterS19:RubricRsrchQ                      0.133874   0.267750   0.500
## SemesterS19:RubricSelMeth                    -0.089616   0.282837  -0.317
## SemesterS19:RubricTxtOrg                      0.166097   0.293176   0.567
## SemesterS19:RubricVisOrg                      0.146845   0.302496   0.485
## as.factor(Rater)2:SemesterS19:RubricInitEDA    0.020326   0.392376   0.052
## as.factor(Rater)3:SemesterS19:RubricInitEDA    0.252422   0.389961   0.647
## as.factor(Rater)2:SemesterS19:RubricInterpRes -0.266618   0.385390  -0.692
## as.factor(Rater)3:SemesterS19:RubricInterpRes -0.152392   0.383354  -0.398
## as.factor(Rater)2:SemesterS19:RubricRsrchQ    -0.217348   0.360414  -0.603
## as.factor(Rater)3:SemesterS19:RubricRsrchQ     0.354319   0.357388   0.991
```

```
## as.factor(Rater)2:SemesterS19:RubricSelMeth   -0.401035   0.370200  -1.083
## as.factor(Rater)3:SemesterS19:RubricSelMeth   -0.192670   0.367887  -0.524
## as.factor(Rater)2:SemesterS19:RubricTxtOrg    -0.542267   0.385011  -1.408
## as.factor(Rater)3:SemesterS19:RubricTxtOrg    -0.316395   0.382614  -0.827
## as.factor(Rater)2:SemesterS19:RubricVisOrg    -0.603626   0.392909  -1.536
## as.factor(Rater)3:SemesterS19:RubricVisOrg    -0.186749   0.390759  -0.478

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

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

```
## If you compare with summary(comb.inter) you will see that there wasn't much difference in the fitted

comb.inter_elim <- fitLMER.fnc(comb.inter.u, log.file.name = FALSE)
```

```
## Warning in fitLMER.fnc(comb.inter.u, log.file.name = FALSE): Argument "ran.effects" is empty, which r
## TRUE

## ======================================================
## ===              backfitting fixed effects         ===
## ======================================================
## processing model terms of interaction level 3
##   iteration 1
##     p-value for term "as.factor(Rater):Semester:Rubric" = 0.5526 >= 0.05
##     not part of higher-order interaction

## boundary (singular) fit: see ?isSingular

##     removing term
## processing model terms of interaction level 2
##   iteration 2
##     p-value for term "as.factor(Rater):Semester" = 0.598 >= 0.05
##     not part of higher-order interaction

## boundary (singular) fit: see ?isSingular

##     removing term
##   iteration 3
##     p-value for term "Semester:Rubric" = 0.0761 >= 0.05
##     not part of higher-order interaction

## boundary (singular) fit: see ?isSingular

##     removing term
## processing model terms of interaction level 1
##   all terms of interaction level 1 significant
## pruning random effects structure ...
##   nothing to prune
## ======================================================
## ===            forwardfitting random effects       ===
## ======================================================
##  ===       random slopes       ===
## ======================================================
## ===            re-backfitting fixed effects        ===
## ======================================================
```

```
## processing model terms of interaction level 2
##   all terms of interaction level 2 significant
## processing model terms of interaction level 1
##   all terms of interaction level 1 significant
## resetting REML to TRUE

## boundary (singular) fit: see ?isSingular

## pruning random effects structure ...
##   nothing to prune
```

```
summary(comb.inter_elim)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: as.numeric(Rating) ~ (0 + Rubric | Artifact) + as.factor(Rater) +
##     Semester + Rubric + as.factor(Rater):Rubric
##    Data: tall.nonmissing
## Control: lmerControl(optimizer = "bobyqa", optCtrl = list(maxfun = 2e+05))
##
## REML criterion at convergence: 1419.6
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.9280 -0.5122 -0.0447  0.4827  3.5854
##
## Random effects:
##  Groups   Name            Variance Std.Dev. Corr
##  Artifact RubricCritDes   0.50348  0.7096
##           RubricInitEDA   0.35480  0.5956   0.44
##           RubricInterpRes 0.15192  0.3898   0.35 0.82
##           RubricRsrchQ    0.17953  0.4237   0.63 0.44 0.72
##           RubricSelMeth   0.06727  0.2594   0.42 0.60 0.74 0.36
##           RubricTxtOrg    0.26069  0.5106   0.42 0.64 0.67 0.55 0.64
##           RubricVisOrg    0.25491  0.5049   0.34 0.71 0.68 0.51 0.38 0.77
##  Residual                 0.18519  0.4303
## Number of obs: 810, groups:  Artifact, 90
##
## Fixed effects:
##                                  Estimate Std. Error t value
## (Intercept)                       1.75945    0.11785  14.929
## as.factor(Rater)2                 0.36537    0.13296   2.748
## as.factor(Rater)3                 0.21421    0.13297   1.611
## SemesterS19                      -0.17780    0.08228  -2.161
## RubricInitEDA                     0.74625    0.13676   5.457
## RubricInterpRes                   1.01453    0.13479   7.527
## RubricRsrchQ                      0.74926    0.12419   6.033
## RubricSelMeth                     0.42672    0.13040   3.272
## RubricTxtOrg                      1.04967    0.13551   7.746
## RubricVisOrg                      0.68354    0.13947   4.901
## as.factor(Rater)2:RubricInitEDA   -0.30843    0.17249  -1.788
## as.factor(Rater)3:RubricInitEDA   -0.29522    0.17282  -1.708
## as.factor(Rater)2:RubricInterpRes -0.53674    0.17008  -3.156
## as.factor(Rater)3:RubricInterpRes -0.75247    0.17049  -4.414
## as.factor(Rater)2:RubricRsrchQ    -0.50157    0.16151  -3.106
## as.factor(Rater)3:RubricRsrchQ    -0.37068    0.16179  -2.291
## as.factor(Rater)2:RubricSelMeth   -0.39602    0.16467  -2.405
```

```
## as.factor(Rater)3:RubricSelMeth   -0.41324    0.16504  -2.504
## as.factor(Rater)2:RubricTxtOrg    -0.58380    0.17141  -3.406
## as.factor(Rater)3:RubricTxtOrg    -0.48649    0.17177  -2.832
## as.factor(Rater)2:RubricVisOrg    -0.14444    0.17442  -0.828
## as.factor(Rater)3:RubricVisOrg    -0.33380    0.17481  -1.910

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

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

```
## it's a little hard to compare summaries for such big models, so let's look at the highlights:

formula(comb.inter.u)
```

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

```
formula(comb.inter_elim)
```

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

```
formula(comb.back_elim)
```

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

```
summary(comb.inter.u)$varcor
```

```
##  Groups   Name             Std.Dev. Corr
##  Artifact RubricCritDes    0.69678
##           RubricInitEDA    0.59378  0.416
##           RubricInterpRes  0.38235  0.324 0.800
##           RubricRsrchQ     0.40551  0.655 0.430 0.723
##           RubricSelMeth    0.25094  0.446 0.639 0.784 0.488
##           RubricTxtOrg     0.50439  0.436 0.649 0.667 0.604 0.622
##           RubricVisOrg     0.50524  0.349 0.727 0.675 0.567 0.346 0.757
##  Residual                  0.43404
```

```
summary(comb.inter_elim)$varcor
```

```
##  Groups   Name             Std.Dev. Corr
##  Artifact RubricCritDes    0.70956
##           RubricInitEDA    0.59565  0.445
##           RubricInterpRes  0.38977  0.354 0.815
##           RubricRsrchQ     0.42371  0.631 0.440 0.716
##           RubricSelMeth    0.25937  0.424 0.601 0.737 0.364
##           RubricTxtOrg     0.51058  0.417 0.637 0.675 0.547 0.636
##           RubricVisOrg     0.50489  0.339 0.715 0.677 0.512 0.376 0.772
##  Residual                  0.43034
```

```
summary(comb.back_elim)$varcor
```

```
##  Groups   Name             Std.Dev. Corr
##  Artifact RubricCritDes    0.74495
```

```
##           RubricInitEDA   0.59215  0.467
##           RubricInterpRes 0.41100  0.230 0.749
##           RubricRsrchQ    0.40960  0.588 0.436 0.704
##           RubricSelMeth   0.25493  0.399 0.603 0.736 0.397
##           RubricTxtOrg    0.50612  0.335 0.614 0.691 0.551 0.656
##           RubricVisOrg    0.50886  0.350 0.731 0.679 0.516 0.414 0.752
##  Residual                 0.43513
```

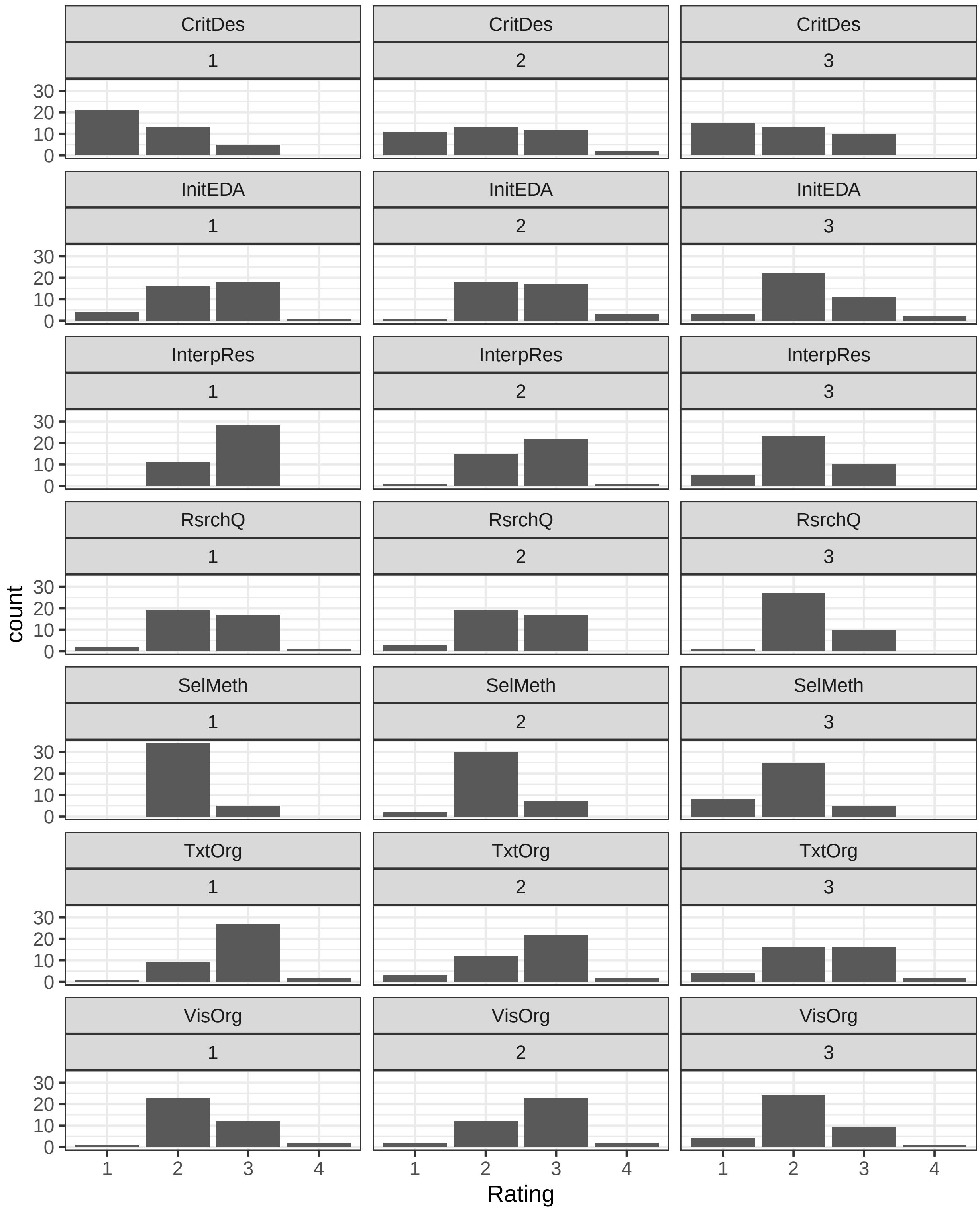
```
anova(comb.back_elim,comb.inter_elim,comb.inter.u)
```

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

```
## Data: tall.nonmissing
## Models:
## comb.back_elim: as.numeric(Rating) ~ (0 + Rubric | Artifact) + as.factor(Rater) + Semester + Rubric
## comb.inter_elim: as.numeric(Rating) ~ (0 + Rubric | Artifact) + as.factor(Rater) + Semester + Rubric
## comb.inter.u: as.numeric(Rating) ~ (0 + Rubric | Artifact) + as.factor(Rater) + Semester + Rubric + a
##                 npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
## comb.back_elim    39 1464.0 1647.2 -693.02   1386.0
## comb.inter_elim   51 1454.5 1694.1 -676.26   1352.5 33.526 12   0.000801 ***
## comb.inter.u      71 1471.4 1804.8 -664.68   1329.4 23.161 20   0.280962
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

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

## comb.inter_elim adds a rater x rubric interaction to the main-effects model comb.back_elim.This sugge
```

```
## In addition to looking at the fixed effect coefficients in
## summary(comb.inter_elim)$coef, we could also see if there's
## a pattern in an appropriate facets plot

g <- ggplot(tall.nonmissing, aes(x=Rating)) +
  geom_bar() +
  facet_wrap( ~ Rubric + Rater, nrow=7)

g
```




```
## and it does look as if the 3 raters have different ways of scoring the 7 rubrics,
## so the interaction we found in comb.inter_elim makes sense.
```

```
## Finally, we consider adding random effects to what seems like the
## best model so far, comb.inter_elim...
```

```
## The fixed-effects terms we have to work with are:
##
## as.factor(Rater)
## Semester
## as.factor(Rater):Rubric
##
## In all cases, there is more than one random effect to test (3 for raters,
## 2 for semesters, 7 for rubrics, and 21 for the interaction).  Since exactRLRT()
## can only test single random effects, we can't use it.  Instead we inspect AIC
## andBIC from anova() tables for these...


m0 <- comb.inter_elim
mA <- 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(m0,mA)
```

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

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

## Data: tall.nonmissing
## Models:
## m0: as.numeric(Rating) ~ (0 + Rubric | Artifact) + as.factor(Rater) + Semester + Rubric + as.factor(R
## mA: as.numeric(Rating) ~ (0 + Rubric | Artifact) + (0 + as.factor(Rater) | Artifact) + as.factor(Rate
##    npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
## m0   51 1454.5 1694.1 -676.26   1352.5
## mA   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
```

```
##
## AIC and BIC both like including (0 + as.factor(Rater) | Artifact) in the model

m0 <- comb.inter_elim
mA <- 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(m0,mA)
```

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

## Data: tall.nonmissing
## Models:
## m0: as.numeric(Rating) ~ (0 + Rubric | Artifact) + as.factor(Rater) + Semester + Rubric + as.factor(R
```

```
## mA: as.numeric(Rating) ~ (0 + Rubric | Artifact) + (0 + Semester | Artifact) + as.factor(Rater) + Sem
##    npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
## m0   51 1454.5 1694.1 -676.26   1352.5
## mA   54 1458.4 1712.0 -675.18   1350.4 2.1534  3     0.5412
```

```
##
## AIC and BIC do not like (0 + Semester | Artifact) in the model...


m0 <- comb.inter_elim
mA <- 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)
```

```
## Error: number of observations (=810) <= number of random effects (=1890) for term (0 + as.factor(Rate
```

```
## anova(m0,mA)
##
## There are not enough observations to fit mA here, so we need not do any
## formal model comparison...

## So, to summarize, the "final" model appears to be

comb.final <- 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)
```

```
formula(comb.final)
```

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

```
summary(comb.final)$varcor
```

```
##  Groups     Name               Std.Dev. Corr
##  Artifact   RubricCritDes      0.70456
##             RubricInitEDA      0.56385   0.318
##             RubricInterpRes    0.31953   0.142  0.674
##             RubricRsrchQ       0.42309   0.500  0.194  0.538
##             RubricSelMeth      0.19564   0.145  0.227  0.376 -0.240
##             RubricTxtOrg       0.50029   0.268  0.437  0.364  0.305  0.213
##             RubricVisOrg       0.48201   0.175  0.504  0.445  0.276 -0.160
##  Artifact.1 as.factor(Rater)1 0.11309
##             as.factor(Rater)2 0.33421  -0.488
##             as.factor(Rater)3 0.30670   0.330  0.663
##  Residual                      0.36700
##
##
##
##
##
##
##
```

```
##   0.537
##
##
##
##
```

```
summary(comb.final)$coef
```

```
##                                     Estimate Std. Error    t value
## (Intercept)                        1.7575675 0.11403884 15.4120075
## as.factor(Rater)2                  0.3660512 0.13918262  2.6300063
## as.factor(Rater)3                  0.1958650 0.12967617  1.5104163
## SemesterS19                       -0.1591929 0.07647446 -2.0816477
## RubricInitEDA                      0.7394806 0.12996198  5.6899761
## RubricInterpRes                    0.9915166 0.12771096  7.7637555
## RubricRsrchQ                       0.7261861 0.11792862  6.1578445
## RubricSelMeth                      0.4106681 0.12470221  3.2931906
## RubricTxtOrg                       1.0157886 0.12999521  7.8140465
## RubricVisOrg                       0.6542550 0.13353206  4.8996095
## as.factor(Rater)2:RubricInitEDA   -0.2997977 0.15609303 -1.9206348
## as.factor(Rater)3:RubricInitEDA   -0.2946987 0.15635429 -1.8848136
## as.factor(Rater)2:RubricInterpRes -0.5132368 0.15349003 -3.3437796
## as.factor(Rater)3:RubricInterpRes -0.7148456 0.15364513 -4.6525755
## as.factor(Rater)2:RubricRsrchQ    -0.4874143 0.14722200 -3.3107438
## as.factor(Rater)3:RubricRsrchQ    -0.3223763 0.14726598 -2.1890751
## as.factor(Rater)2:RubricSelMeth   -0.3863680 0.15031029 -2.5704694
## as.factor(Rater)3:RubricSelMeth   -0.3871301 0.14961676 -2.5874779
## as.factor(Rater)2:RubricTxtOrg    -0.5510564 0.15646236 -3.5219741
## as.factor(Rater)3:RubricTxtOrg    -0.4448931 0.15673326 -2.8385369
## as.factor(Rater)2:RubricVisOrg    -0.1049122 0.15861363 -0.6614326
## as.factor(Rater)3:RubricVisOrg    -0.2752225 0.15885162 -1.7325758
```

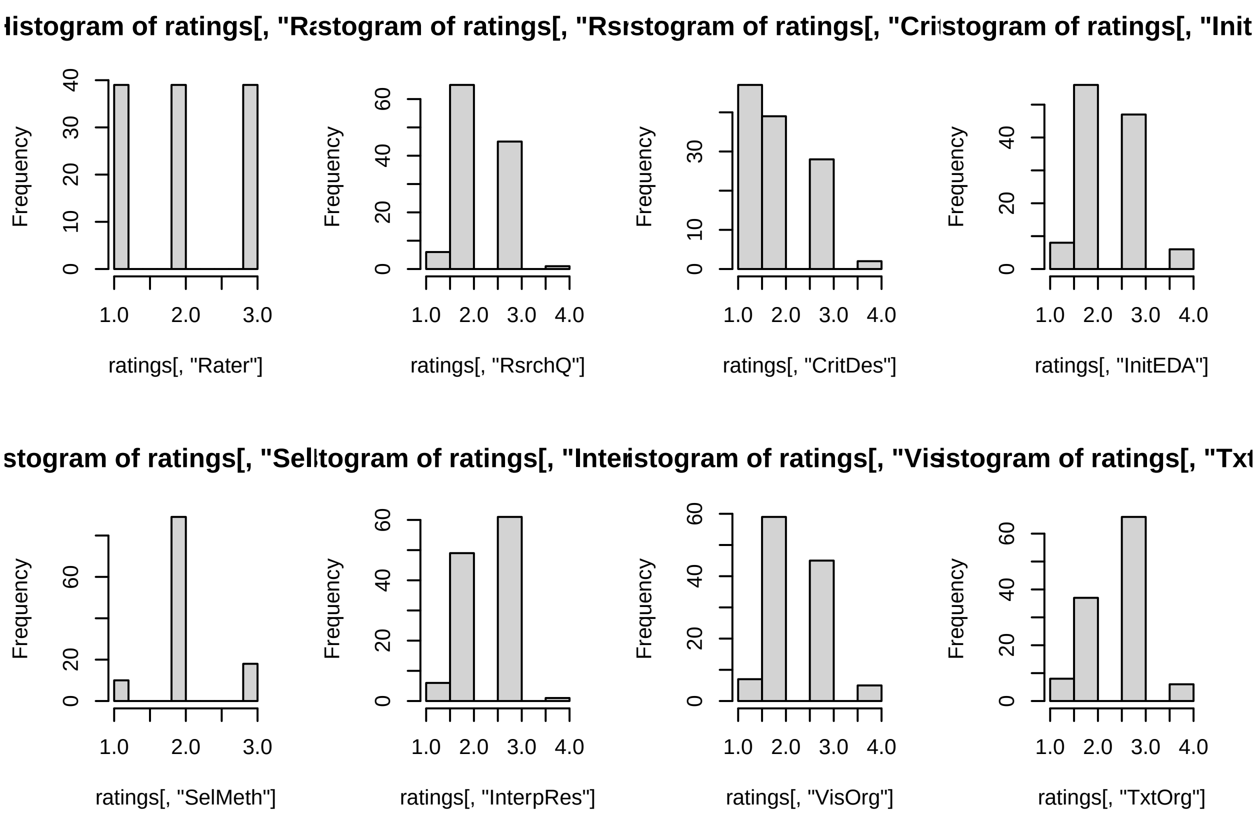
```
## if we accept comb.final as our final model, we can interpret the pieces as
## follows:
##
## (0 + as.factor(Rater) | Artifact) + as.factor(Rater)
##   * There is a kind of Rater x Artifact interaction: each Rater's
##     rating on each Artifact differs from what we would expect (from the
##     fixed effects alone) by a small random effect that depends on the Artifact
##
## Rubric + as.factor(Rater) + as.factor(Rater):Rubric
##   * There is a Rater x Rubric interaction: each Rater uses each
##     Rubric in a way that is not like, or even parallel to, other rater's
##     Rubric usage.  (we saw that in the facets plot above also).
##
## (0 + Rubric | Artifact) + Rubric
##   * There is a kind of Rubric x Artifact interaction: There are
##     different average scores on each rubric, but the rubric averages also
##     vary a bit from one Artifact to the next, by a small random effect that
##     depends on Artifact

## In all of this, the fact that Rubric scores depend on Artifact (that is,
## there is a kind of Rubric x Artifact interaction) is what we might expect:
## the artifacts aren't all of equal quality on each rubric, and so we should
## expect the average scores on each Rubric to vary from one Artifact to the next.
```

```
##
## More troubling are the Rater x Rubric interaction and the "kind of"
## Rater x Artifact interaction.  The Rater x Rubric interaction suggests
## that the Raters are not all interpreting the Rubrics in the same way.  The
## "kind of" Rater x Artifact interaction suggests that the Raters are not
## interpreting the evidence in the artifacts in the same way.  These
## interactions suggest that perhaps the raters should be trained more, to
## make the raters' ratings more similar to each other.
```

Research Question 4: More on Data EDA:

4.Why some factors from rating data are having very skewed distributions? How would the pattern affect our result and final model selection?

```
ratings <- read.csv("ratings.csv")
par(mfrow = c(2,4))
hist(ratings[,"Rater"])
hist(ratings[,"RsrchQ"])
hist(ratings[,"CritDes"])
hist(ratings[,"InitEDA"])
hist(ratings[,"SelMeth"])
hist(ratings[,"InterpRes"])
hist(ratings[,"VisOrg"])
hist(ratings[,"TxtOrg"])
```



The distribution does not make sense for some of variables but gives us a sense of how the variables are distributed.InitEDA, RsrchQ, InterpRes, VisOrg, TetOrg, SelMeth have high values of rate 3 and rate 4, and very few rate 1 and rate 2. CritDes only has roughly all numbers of rate 1.

Additional EDA on the percentage of each rubrics using margin table:

```
#Looking at the percentage:
CritDes<-table(ratings$CritDes)
addmargins(CritDes)
```

```
##
##   1   2   3   4 Sum
##  47  39  28   2 116
```

```
round(prop.table(CritDes)*100,digits=0)
```

```
##
##  1  2  3  4
## 41 34 24  2
```

```
InitEDA <- table(ratings$InitEDA)
addmargins(InitEDA)
```

```
##
##   1   2   3   4 Sum
##   8  56  47   6 117
```

```
round(prop.table(InitEDA)*100,digits=0)
```

```
##
##  1  2  3  4
##  7 48 40  5
```

```
SelMeth <- table(ratings$SelMeth)
addmargins(SelMeth)
```

```
##
##   1   2   3 Sum
##  10  89  18 117
```

```
round(prop.table(SelMeth)*100,digits=0)
```

```
##
##  1  2  3
##  9 76 15
```

```
InterpRes <- table(ratings$InterpRes)
addmargins(InterpRes)
```

```
##
##   1   2   3   4 Sum
##   6  49  61   1 117
```

```
round(prop.table(InterpRes)*100,digits=0)
```

```
##
##  1  2  3  4
##  5 42 52  1
```

```
VisOrg <- table(ratings$VisOrg)
addmargins(VisOrg)
```

```
##
##   1   2   3   4 Sum
##   7  59  45   5 116
```

```r
round(prop.table(VisOrg)*100,digits=0)
```

```
##
##  1  2  3  4
##  6 51 39  4
```

```r
TxtOrg <- table(ratings$TxtOrg)
addmargins(TxtOrg)
```

```
##
##    1   2   3   4 Sum
##    8  37  66   6 117
```

```r
round(prop.table(TxtOrg)*100,digits=0)
```

```
##
##  1  2  3  4
##  7 32 56  5
```

```r
Artifact <- table(ratings$Artifact)
addmargins(Artifact)
```

```
##
## 100 101 102 103 104 105 106 107 111 112 113 114 115 116 117 118  13  15  16  17
##   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1
##  21  22  23  24  25  26  27  28  32  33  34  35  36  37  38  39  40  45  46  47
##   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1
##  48  49   5  53  54  55  56  57   6  61  62  63  64  65  66  67  68   7  72  73
##   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1
##  74  75  76  77  78  79   8  84  85  86  87  88   9  92  93  94  95  96  01 010
##   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   3   3
## 011 012 013  02  03  04  05  06  07  08  09 Sum
##   3   3   3   3   3   3   3   3   3   3   3 117
```

```r
round(prop.table(Artifact)*100,digits=0)
```

```
##
## 100 101 102 103 104 105 106 107 111 112 113 114 115 116 117 118  13  15  16  17
##   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1
##  21  22  23  24  25  26  27  28  32  33  34  35  36  37  38  39  40  45  46  47
##   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1
##  48  49   5  53  54  55  56  57   6  61  62  63  64  65  66  67  68   7  72  73
##   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1
##  74  75  76  77  78  79   8  84  85  86  87  88   9  92  93  94  95  96  01 010
##   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   3   3
## 011 012 013  02  03  04  05  06  07  08  09
##   3   3   3   3   3   3   3   3   3   3   3
```

```r
Repeated <- table(ratings$Repeated)
addmargins(Repeated)
```

```
##
##    0   1 Sum
##   78  39 117
```

```r
round(prop.table(Repeated)*100,digits=0)
```

```
##
```

```
## 0  1
## 67 33
```

```
RsrchQ <- table(ratings$RsrchQ)
addmargins(RsrchQ)
```

```
##
##   1   2   3   4 Sum
##   6  65  45   1 117
```

```
round(prop.table(RsrchQ)*100,digits=0)
```

```
##
##  1  2  3  4
##  5 56 38  1
```

```
Sex <- table(ratings$Sex)
addmargins(Sex)
```

```
##
##  --   F   M Sum
##   1  64  52 117
```

```
round(prop.table(Sex)*100,digits=0)
```

```
##
## --  F  M
##  1 55 44
```

```
Semester <- table(ratings$Semester)
addmargins(Semester)
```

```
##
##   Fall Spring    Sum
##     83     34    117
```

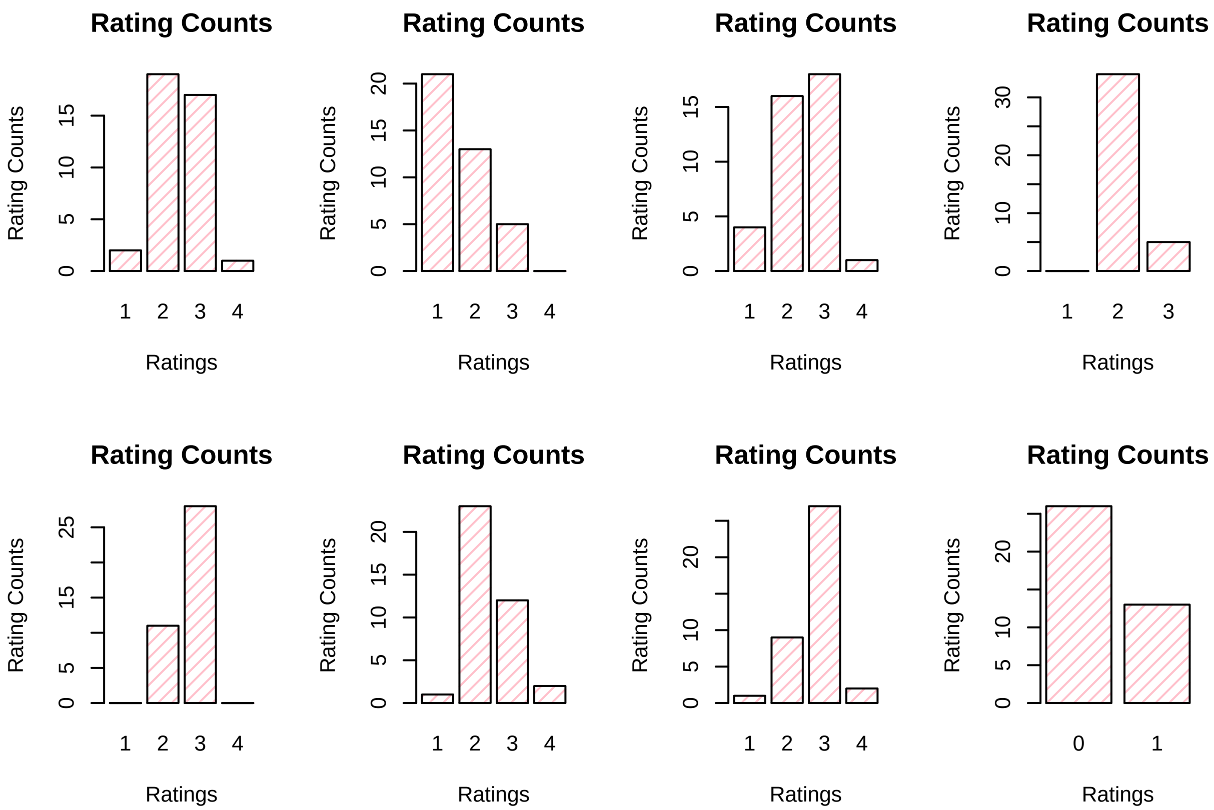
```
round(prop.table(Semester)*100,digits=0)
```

```
##
##   Fall Spring
##     71     29
```

These percentage tables help us look at the exact percentages of each factors.According to the percentage marginal tables of all the factors, we can see the difference of factors in each season. This might have something to do with the skewness in distributions, but more about whether to include that part in our model or not. There is no need to include Sex in our model because there are no much difference across difference genders. Similarly, the repeated factor: 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. Therefore, we should not include Sex and Repeated in our model.

```
#Converting all numerical factors to factor

ratings$X <- as.factor(ratings$X)
ratings$Rater <- as.factor(ratings$Rater)
ratings$Sample <- as.factor(ratings$Sample)
ratings$Overlap <- as.factor(ratings$Overlap)
ratings$Semester <- as.factor(ratings$Semester)
ratings$Sex<- as.factor(ratings$Sex)
```

```
ratings$RsrchQ <- as.factor(ratings$RsrchQ)
ratings$CritDes <- as.factor(ratings$CritDes)
ratings$InitEDA <- as.factor(ratings$InitEDA)
ratings$SelMeth <- as.factor(ratings$SelMeth)
ratings$InterpRes <- as.factor(ratings$InterpRes)
ratings$VisOrg <- as.factor(ratings$VisOrg)
ratings$TxtOrg <- as.factor(ratings$TxtOrg)
ratings$Artifact <- as.factor(ratings$Artifact)
ratings$Repeated <- as.factor(ratings$Repeated)

#Looking at the barplot of rating score=1 as an example:
par(mfrow=c(2,4))
ratings_1<- ratings %>% filter(ratings$Rater==1)
barplot(table(ratings_1$RsrchQ),main="Rating Counts",xlab="Ratings",ylab="Rating Counts",col="pink",dens

barplot(table(ratings_1$CritDes),main="Rating Counts",xlab="Ratings",ylab="Rating Counts",col="pink",de

barplot(table(ratings_1$InitEDA),main="Rating Counts",xlab="Ratings",ylab="Rating Counts",col="pink",de

barplot(table(ratings_1$SelMeth),main="Rating Counts",xlab="Ratings",ylab="Rating Counts",col="pink",de

barplot(table(ratings_1$InterpRes),main="Rating Counts",xlab="Ratings",ylab="Rating Counts",col="pink",

barplot(table(ratings_1$VisOrg),main="Rating Counts",xlab="Ratings",ylab="Rating Counts",col="pink",dens

barplot(table(ratings_1$TxtOrg),main="Rating Counts",xlab="Ratings",ylab="Rating Counts",col="pink",dens

barplot(table(ratings_1$Repeated),main="Rating Counts",xlab="Ratings",ylab="Rating Counts",col="pink",d
```



Looking at the bar plots, we see that there are differences in rating counts when rating score=1. If we do a further EDA on other rating scores, the result would probably be the same due to the fact that the rating counts for different groups are very different.