Exploring Course Ratings System for CMU Freshman Statistics

Yuqing Xu yuqingxu@andrew.cmu.edu

Abstract

This project mainly focuses on exploring the relationship between the ratings and other various factors (Rater, Semester, Sex, Repeated, Rubric) in this experiment on rating work in Freshman Statistics by raters from across the college. The data including these variables is from Junker(2021) and includes ratings on 1 project-papers that were randomly sampled from a Fall and Spring section from three raters based on seven rubrics. The general method on model selection is to use ANOVA test to select among the intercept-only models, models with fixed effects, models with interactions, and models with random effects. When considering all rubrics as a whole, Rating is related to Rubric, Rater, Semester, interaction of Rater and Semester, random effects Rater and Rubric. For further analysis, we should try to fit more models to find the best one and validity of the model selected should be a concern in the future.

Introduction

Recently the Dietrich College at Carnegie Mellon University has been experimenting with rating work in Freshman Statistics by raters from across the college in order to prepare for rating student work performed in the new program. Here in this project, we want to explore the question about when difference raters are asked to rate project papers-refered to as "artifacts" based on seven different rubrics, if there is any relationship between the final ratings and the variables described below, or I will say if there exists any variable that can significantly affect the ratings. The questions will be addressed related to the topic in this project are:

(1) Is the distribution of ratings for each rubrics 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 experiement (Rater, Semester, Sex, Repeated, Rubric) related to the ratings? Do the factors interact in any interesting ways?

(4) Is there anything else interesting to say about this data?

Data

The data file we are going to use is ratings.csv and tall.csv from Junker(2021). The data provides that for 91 project-papers that were randomly sampled from a Fall and Spring section of Freshman Statistics, three raters from three different departments were asked to rate these artifacts on seven rubrics as shown in Table 1, with rating scale for all rubrics 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. *NOTE: These are* **not** *the rubrics used by instructors or TA's in Freshman Statistics. They are* **only** *approved to be used 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. *NOTE: This is* **not** *the rating scale used by instructors or TA's in Freshman Statistics. It is* **only** *approved to be used in this experiment.*

13 of the 91 artifacts were rated by all three raters, and the remaining were rated by only one. ratings.csv and tall.csv contain same information but were organized differently. We can see below in Table 3, variables for analysis are:

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 file `ratings.csv`. Variables that are **not** expected to be useful for analysis are shown in parentheses.

From the table below, we can see how variables distribute:

	Min	1st Qu	Median	Mean	3rd Qu	Max	
Sample	1	31	60	59.9	89	118	
Overlap	1	4	7	7	10	13	
Repeated	0	0	0	0.33	1	1	
	1	2	3				
Rater	39	39	39				
	fall	spring					
Semester	83	34					
	Female	Male					
Sex	64	52					
	1	2	3	4			
RsrchQ	6	65	45	1			
CritDes	47	39	28	2			
InitEDA	8	56	47	6			
SelMeth	10	89	18	0			
InterpRes	6	49	61	1			
VisOrg	7	59	45	5			
TxtOrg$	8	37	66	6			

Table 4

Method

(1) Is the distribution of ratings for each rubrics 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?

To address the first question asking about if there is distinguishable distribution of ratings for each rubrics and each raters, we will need to extract the artifacts with all 3 raters to check the distributions seperately. Firstly, we want to create a bar plot for each rubric to see the difference in distribution of rubric with different ratings for the two seperated datasets(full and 13 artifacts). Also, a table of counts is a supplement for the plots to see if ratings are distributed similarly for all rubrics for both datasets. Secondly, to compare distributions across raters, same method is used. A bar plot is made for each rater from both full datasets and the datasets with 13 artifacts. And tables of counts are also used to help recheck the distribution of ratings based on different raters.

(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?

For the research question talking about the agreement among raters, we still seperate the dataset as the full dataset and the one with 13 artifacts. The basic idea is to use ICC value (intra-class correlation coefficient value), which measures the reliability of two different raters to measure subjects similarity. And we will calculate the ICC value for all ratings for each rubric and all rateings for each rubric from three raters pairwisely. Also, 2-way tables of counts for the ratings of each pair of raters, on each rubric recording the percent exact agreement between the two raters will be used to help to determine disagreement and agreement.

(3) More generally, how are the various factors in this experiement (Rater, Semester, Sex, Repeated, Rubric) related to the ratings? Do the factors interact in any interesting ways?

(3)(i)

We will start with addressing the 13 common artifacts with all 3 raters' ratings, adding fixed effect to the seven rubric-specific models by fitting linear mixed effect model (lmer). Variable "Repeated" will be removed because Repeated will be all the same for these 13 artifacts. Then, a backwards-elimination process is applied to the model so that only significant fixed effects are left by usiong fitLMER.fnc(). After that, we will use ANOVA test to compare the model with only intercept and the model with fixed effects, and see if more interactions are needed.

(3)(ii)

After finishing dealing with the 13 common artifacts, we will start working on the full dataset with same process. For the reason that we should use same dataset for every model fitting and comparison, missing data will be eliminated. We will add fixed effect to the seven rubric-specific models by fitting linear mixed effect model (lmer). Then, a backwards-elimination process is applied to the model so that only significant fixed effects are left by usiong fitLMER.fnc(). After that, we will use ANOVA test to compare the model with only intercept and the model with fixed effects, and see if more interactions are needed.

(3)(iii)

For those rubrics whose selected models from the previous step are not just the simple intercept-only models, we will examine each of these to see if the fixed effect make sense and if there are any interactions or additional random effects. For each of them, firstly, refit the model and check t-values for all variables to decide if they are significant; secondly, we will use ANOVA test to compare the model and model without "Rater" to see if we really need "Rater" as a factor for this rubric model; thirdly, we will add fixed-effect interactions between each pair of variables left in the best model selected from previous steps to the model, and use ANOVA test to compare the new model and original model to check if we need the interactions; finally, random effects on the models should be considered and compared to get the final model.

(3)(iv)

Finally, we combine all rubrics and consider it as a whole variable in order to find the relationship of Rubric variable and other variables. We try to add fixed effects to the "combined" intercept-only model with Rubric|Artifacts using all the data by backward elimination. Interactions are also added based on the result

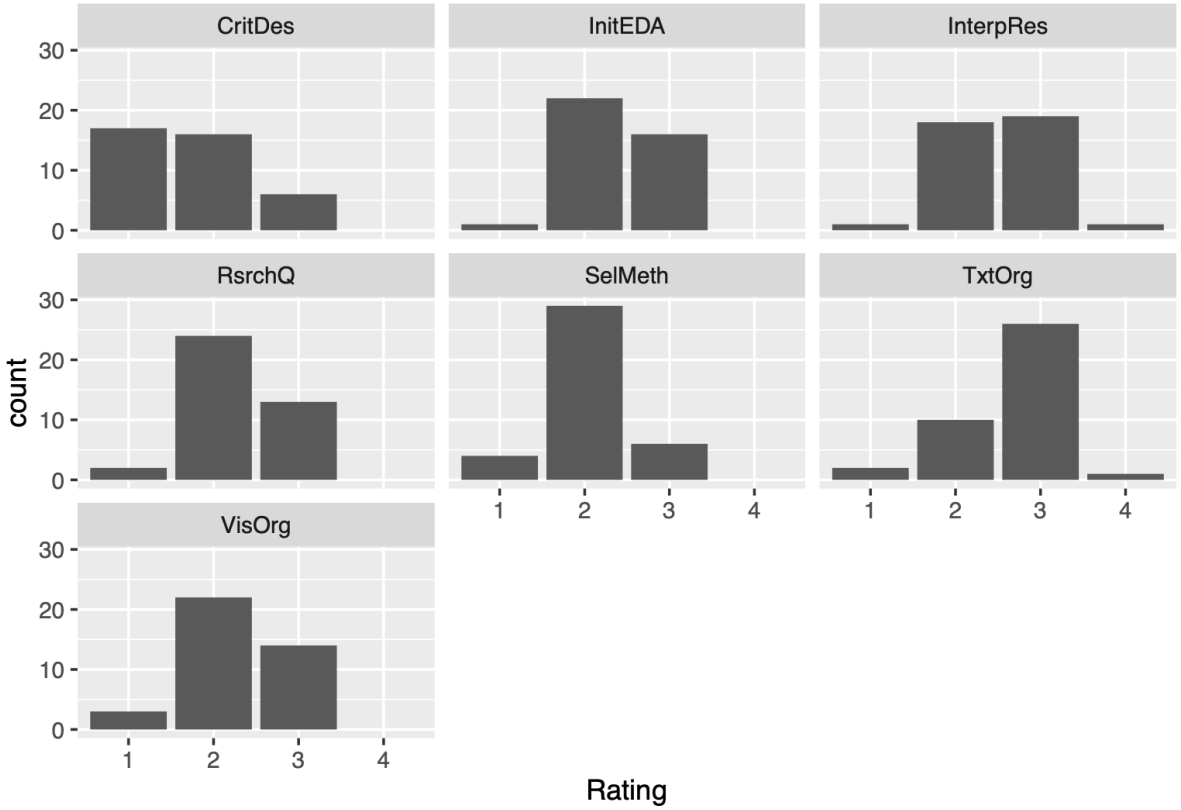
we get from previous step by combining each pair of fixed effects. Then, ANOVA test is used to select the model. And after adding random effects to the model, we will use ANOVA to select the final model.

Result

(1) Is the distribution of ratings for each rubrics 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?

Technical Appendix part (1): page 2-6

By plot shown below, which shown the distribution of 7 rubrics in the extracted dataset that contains 13 artifacts with all 3 raters' ratings, we can see that CritDes is the only one rubric which has most ratings at 1 than all the other ratings, as other rubrics have highest counts at score 2 or 3. Also, rubrics InitEDA, RsrchQ, SelMeth, and Visorg do not have any ratings at 4, but one reason might be the number of observations is small for this subset.



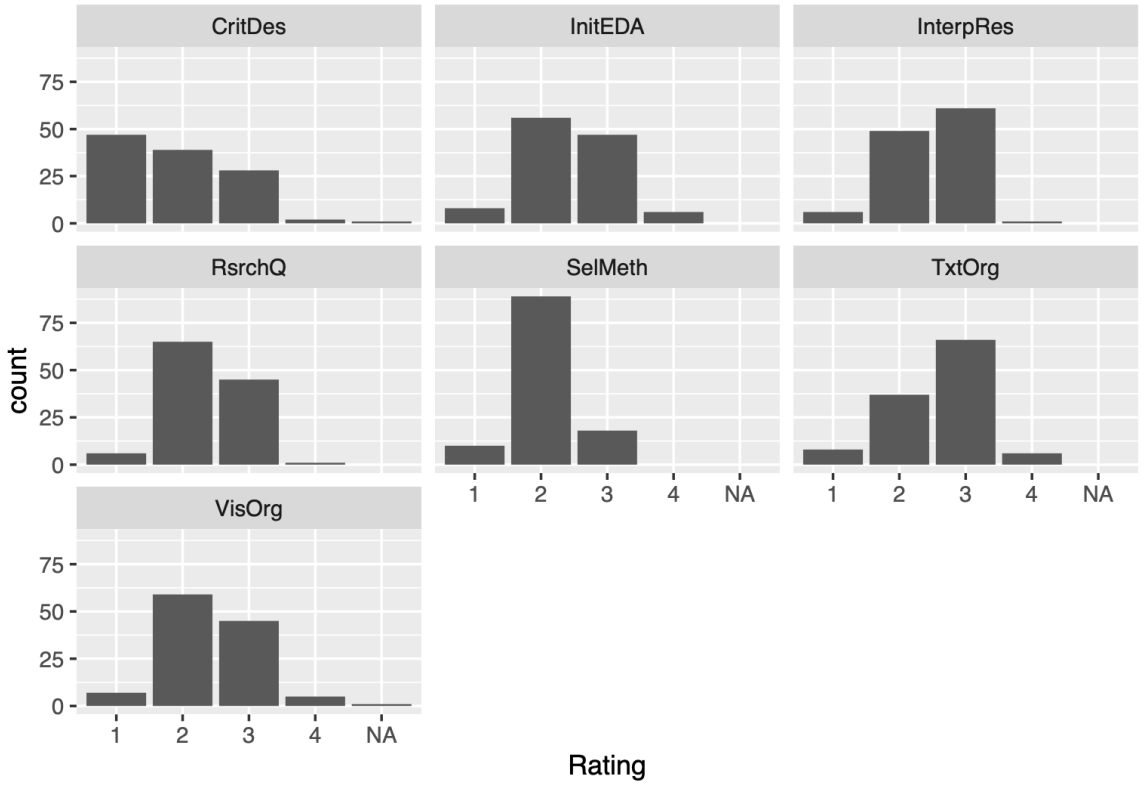
Plot 1

Also, we can find similar pattern based on the table of counts of the same dataset.

	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

Plot 2

By plot shown below, which shown the distribution of 7 rubrics in the full dataset , we can see that CritDes is the only one rubric which has most ratings at 1 than all the other ratings, as other rubrics have highest counts at score 2 or 3. Also, rubrics SelMeth does not have any ratings at 4.



Plot 3

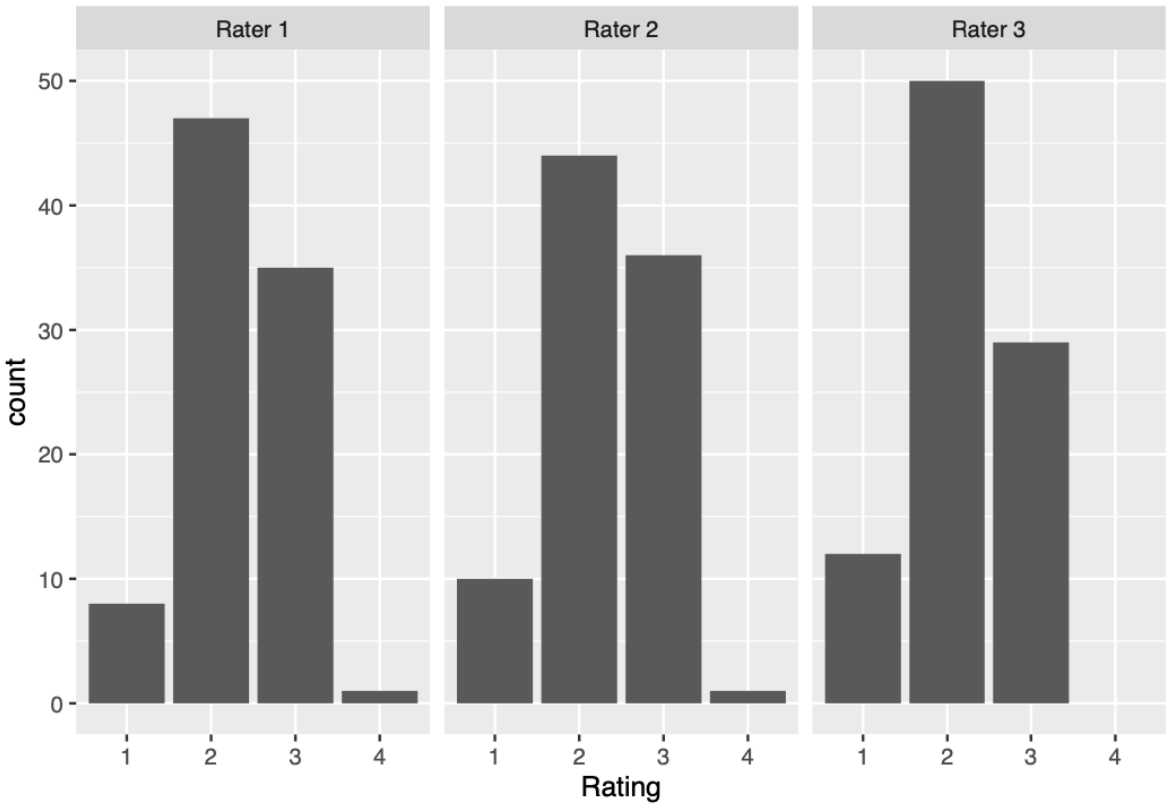
Also, we can find similar pattern based on the table of counts of the same dataset.

	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

Table 5

From everything we have above, I will say that the distribution of CritDes is distinguishable from other rubrics and distributions of other rubrics are indistinguishable. For rubric CritDes, we can see that largest proportion of students get score 0; however, for all other rubrics, most students get score 2 or 3 and only few get 0 or 4, and this difference makes CritDes distinguishable from others. Raters tend to give lower score for rubric CritDes than other rubrics.

By plot shown below, which shown the distribution of 3 raters in the extracted dataset that contains 13 artifacts with all 3 raters' ratings, we can see that they all have similar patterns that they rate at score 2 more than other scores, and they rate at score 4 least, especially that rater 3 does not give any students score 4.



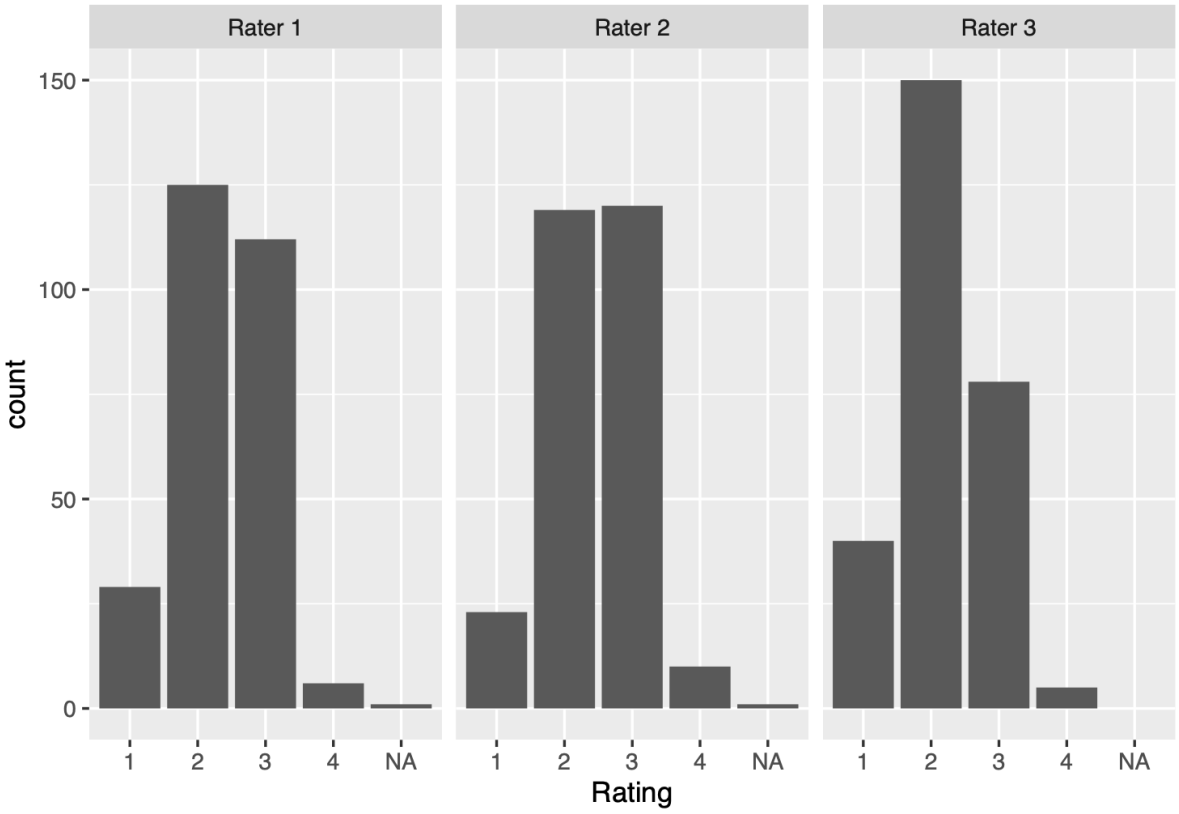
Plot 4

Also, we can find similar pattern based on the table of counts of the same dataset.

```
##          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
```

Table 6

By plot shown below, which shown the distribution of 3 raters in the full dataset, we can see that they all have similar patterns that they rate at score 2 or 3 more than other scores, and they rate at score 4 least.



Plot 5

Also, we can find similar pattern based on the table of counts of the same dataset.

	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

Table 7

From everything we have above, I will say that all 3 raters are indistinguishable as they all have really similar patterns and none of them tends to give a lower/higher score.

As we do not have any missing value in the smaller 13-rubirc dataset, we do not need to deal with it. From the table below, we can see that there are missing values for Rating. We may need to remove or address it when we use these data in the model later.

	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>

Table 8

For the one missing sex value shown at the table below, for the reason that we do not want to lose this data and also we cannot decide sex for this data easily based on what we have now, we will just leave it as a third sex category "–".

```
##   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
```

Table 9

(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?

Technical Appendix part (2): page 6-11

From the table below showing the ICC value for ratings on each rubric and the value for ratings from raters pairwisely from the subset containing only the 13 artifacts, we can see that InterpResm RsrchQ, and TxtOrg have somehow low ICC values indicating their especially low reliability on each other raters. Also, I will say that for the overall ICC value for each rubric, they are all not high.

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

Table 10

From the table below showing the ICC value for ratings on each rubric and the value for ratings from raters pairwisely from the subset containing only the 13 artifacts, we can see that InterpResm RsrchQ, and TxtOrg have somehow low ICC values indicating their especially low reliability on each other raters. Also, I will say that for the overall ICC value for each rubric, they are all not high. And ICC for all data and ICC for subset are quite similar.

```
##            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 11

Based on the tables in the Technical Appendix, I will say that most times the raters agree with each other as the percent exact agreement between each pair of raters for each rubric is not low and most time their ratings match the ratings from the other. From the tables above, even though percent exact agreement between each pair of raters is not low, the reliability among raters for each rubric is not high, we cannot say that they agree with each other.

(3) More generally, how are the various factors in this experiement (Rater, Semester, Sex, Repeated, Rubric) related to the ratings? Do the factors interact in any interesting ways?

Technical Appendix part (3)(i): page 12-17

For the reduced dataset containing 13 common artifacts with all 3 raters' ratings, to check if fixed effects are needed for each model of each rubric, the intercept-only model for each rubric is compared with the model adding fixed effect Rater, Semester, and Sex, with backwards-elimination applied. The result of ANOVA test for each pair of models is larger than 0.05, which tells that for each rubric, the intercept-only model is adequate. Thus, the final model chosen for each rubric is:

```
## $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)
```

Table 12

The result tells that for the 13 artifacts that all three raters saw, we do not need to add fixed effects to any of the models and they do not help improve the fit of data. Thus, no interactions or random effects are going to be added anymore in the later step.

Code Appendix part (3)(ii): page 17-23

For the full dataset, to check if fixed effects are needed for each model of each rubric, the intercept-only model for each rubric is compared with the model adding fixed effect Rater, Semester, and Sex, with backwards-elimination applied. The result of ANOVA test for some pair of models is larger than 0.05 and for some pair is less than 0.05, which tells that for rubric InitEDA, RsrchQ, and TxtOrg, the intercept-only model is adequate, but for all other four rubrics CritDes, InterpRes, SelMeth, and VisOrg, the model with some fixed effects is better than intercept-only model, which means that for some rubrics like CritDes, InterpRes, and VisOrg, Rater is be related to Rating, and for rubric SelMeth, Rater and Semester are related to Rating. Thus, the final model chosen for each rubric is:

```
$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
```

Table 13

For the those rubric that intercpet-model is adequate and enough, we can say that the mean ratings on these rubrics from all raters are the intercept coefficient from summary of model, 2.35 for RsrchQ, 2.44 for InitEDA, and 2.59 for TxtOrg.

The way we interpret the random effect term for each model is that for each model for each rubric, we can get the random effect (grouped by Artifact) coefficients for each of the all 13 artifacts. Take InitEDA as an example as below.

```
## $Artifact
##     (Intercept)
## 01   0.44517185
## 010 -0.03709765
## 011 -0.27823241
## 012 -0.27823241
## 013  0.20403710
## 02  -0.03709765
## 03  -0.03709765
## 04   0.20403710
## 05  -0.27823241
## 06  -0.51936716
## 07   0.44517185
## 08  -0.27823241
## 09   0.44517185
##
## with conditional variances for "Artifact"
```

Table 14

Then we can tell that the mean rating for rubric InitEDA is 2.44, and the mean rating for InitEDA on Artifact 01 is 2.44 + 0.45 = 2.89, and this interpretation works for all artifacts. The high variance of random effects tells that the rating for each artifact varies a lot, and high coefficients mean that the rating for this artifact on this rubric is a lot different than the mean rating for this rubric.

Technical Appendix part (3)(iii): page 23-32

For those rubrics CritDes, InterpRes, SelMeth, and VisOrg, whose selected models from the previous step are not just the simple intercept-only models, we will examine each of these to see if the fixed effect make sense and if there are any interactions or additional random effects that can improve the model fitting.

SelMeth: After refitting the model, from the table below giving the t-values for all variables, we can see that absolute values of t-values are large enough, indicating that the variables in this model are all significant and none of them needs to be removed.

```
##                     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
```

Plot6

With this model including variables Rater and Semester, ANOVA test is applied on this model and model without Rater. The result p-value is less than 0.05, indicating that model without Rater is not adequate for the data. Thus, old model from previous step (the one with Rater) is selected in this step.

```
## 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
```

Plot 7

After, interactions between fixed effects Rater and Semester are added to the model. In this case, Rater*Semester is added. We use ANOVA test to select between the new model with interaction and the old model. The result p-value is larger than 0.05, telling that the model without interactions is adequate. And the old model is selected.

```
## 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):S
##                        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
```

Plot 8

Finally, we will consider if random effects (Semester|Artifact), (as.factor(Rater)|Artifact) are needed. For the reason that lmer() cannot fit these two new models, we cannot add any random effects on this model. Thus, final model selected is the one below.

```
## 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
```

Plot 9

From the summary of model above, we can see that we do not have an intercept (different mean rating for rubric SelMeth from all three raters) for this model, but we have clearly different mean rating from three raters on this rubric 2.25, 2.23, 2.03 from Rater 1,2,3. Also, the mean rating on this rubric is 0.36 lower in Spring than fall semester.

CritDes: After refitting the model, from the table below giving the t-values for all variables, we can see that absolute values of t-values are large enough, indicating that the variables in this model are all significant and none of them needs to be removed. With this model including variables Rater, ANOVA test is applied on this model and model without Rater. The result p-value is less than 0.05, indicating that model without Rater is not adequate for the data. Thus, the model with Rater is selected in this step.

```
Models:
tmp.single_intercept: as.numeric(Rating) ~ (1 | Artifact)
tmp: as.numeric(Rating) ~ as.factor(Rater) + (1 | Artifact) - 1
                     npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
tmp.single_intercept    3 277.68 285.91 -135.84   271.68
tmp                     5 273.62 287.35 -131.81   263.62 8.0535  2    0.01783 *
---
```

Plot 10

After, interactions between fixed effects are added to the model. In this case, as we only have fixed effect

Rater, there is no other variables can be used for interaction with Rater. We will skip this step.

Finally, we will consider if random effect (as.factor(Rater)|Artifact) is needed. For the reason that lmer() cannot fit this new model, we cannot add any random effects on this model. Thus, final model selected is the one below.

```
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
Linear mixed model fit by REML ['lmerMod']
Formula: as.numeric(Rating) ~ as.factor(Rater) + (1 | Artifact) - 1
   Data: tall.nonmissing[tall.nonmissing$Rubric == "CritDes", ]

REML criterion at convergence: 271

Scaled residuals:
     Min       1Q   Median       3Q      Max
-1.55495 -0.50027 -0.08228  0.64663  1.60935

Random effects:
 Groups   Name        Variance Std.Dev.
 Artifact (Intercept) 0.4349   0.6595
 Residual             0.2473   0.4972
Number of obs: 115, groups:  Artifact, 89

Fixed effects:
                  Estimate Std. Error t value
as.factor(Rater)1   1.6863     0.1207   13.98
as.factor(Rater)2   2.1129     0.1219   17.34
as.factor(Rater)3   1.8908     0.1219   15.51

Correlation of Fixed Effects:
            a.(R)1 a.(R)2
as.fctr(R)2 0.244
as.fctr(R)3 0.244  0.246
```

Plot 11

From the summary of model above, we can see that we do not have an intercept (different mean rating for rubric CritDes from all three raters) for this model, but we have clearly different mean rating from three raters on this rubric 1.69, 2.11, 1.89 from Rater 1,2,3.

InterpRes: After refitting the model, from the table below giving the t-values for all variables, we can see that absolute values of t-values are large enough, indicating that the variables in this model are all significant and none of them needs to be removed. With this model including variables Rater, ANOVA test is applied on this model and model without Rater. The result p-value is less than 0.05, indicating that model without Rater is not adequate for the data. Thus, the model with Rater is selected in this step.

```
Models:
tmp.single_intercept: as.numeric(Rating) ~ (1 | Artifact)
tmp: as.numeric(Rating) ~ as.factor(Rater) + (1 | Artifact) - 1
                     npar    AIC    BIC   logLik deviance  Chisq Df Pr(>Chisq)
tmp.single_intercept    3 218.53 226.79 -106.263   212.53
tmp                     5 200.66 214.43  -95.331   190.66 21.864  2  1.787e-05 ***
---
```

Plot 12

After, interactions between fixed effects are added to the model. In this case, as we only have fixed effect Rater, there is no other variables can be used for interaction with Rater. We will skip this step.

Finally, we will consider if random effect (as.factor(Rater)|Artifact) is needed. For the reason that lmer() cannot fit this new model, we cannot add any random effects on this model. Thus, final model selected is the one below.

```
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
Linear mixed model fit by REML ['lmerMod']
Formula: as.numeric(Rating) ~ as.factor(Rater) + (1 | Artifact) - 1
   Data: tall.nonmissing[tall.nonmissing$Rubric == "InterpRes", ]

REML criterion at convergence: 199.7

Scaled residuals:
    Min      1Q  Median      3Q     Max
-2.5317 -0.7627  0.2635  0.6614  2.6535

Random effects:
 Groups   Name        Variance Std.Dev.
 Artifact (Intercept) 0.06224  0.2495
 Residual             0.25250  0.5025
Number of obs: 116, groups:  Artifact, 90

Fixed effects:
                   Estimate Std. Error t value
as.factor(Rater)1  2.70421    0.08912   30.34
as.factor(Rater)2  2.58574    0.08912   29.01
as.factor(Rater)3  2.13918    0.09027   23.70

Correlation of Fixed Effects:
            a.(R)1 a.(R)2
as.fctr(R)2 0.061
as.fctr(R)3 0.062  0.062
```

Plot 13

From the summary of model above, we can see that we do not have an intercept (different mean rating for

rubric InterpRes from all three raters) for this model, but we have clearly different mean rating from three raters on this rubric 2.7, 2.59, 2.14 from Rater 1,2,3.

VisOrg: After refitting the model, from the table below giving the t-values for all variables, we can see that absolute values of t-values are large enough, indicating that the variables in this model are all significant and none of them needs to be removed. With this model including variables Rater, ANOVA test is applied on this model and model without Rater. The result p-value is less than 0.05, indicating that model without Rater is not adequate for the data. Thus, the model with Rater is selected in this step.

```
Models:
tmp.single_intercept: as.numeric(Rating) ~ (1 | Artifact)
tmp: as.numeric(Rating) ~ as.factor(Rater) + (1 | Artifact) - 1
                     npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
tmp.single_intercept    3 227.21 235.44 -110.60   221.21
tmp                     5 220.82 234.54 -105.41   210.82 10.392  2   0.005539 **
---
```

Plot 14

After, interactions between fixed effects are added to the model. In this case, as we only have fixed effect Rater, there is no other variables can be used for interaction with Rater. We will skip this step.

Finally, we will consider if random effect (as.factor(Rater)|Artifact) is needed. For the reason that lmer() cannot fit this new model, we cannot add any random effects on this model. Thus, final model selected is the one below.

```
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
Linear mixed model fit by REML ['lmerMod']
Formula: as.numeric(Rating) ~ as.factor(Rater) + (1 | Artifact) - 1
   Data: tall.nonmissing[tall.nonmissing$Rubric == "VisOrg", ]

REML criterion at convergence: 219.6

Scaled residuals:
    Min      1Q  Median      3Q     Max
-1.5004 -0.3365 -0.2483  0.3841  1.8552

Random effects:
 Groups   Name        Variance Std.Dev.
 Artifact (Intercept) 0.2907   0.5392
 Residual             0.1467   0.3830
Number of obs: 115, groups:  Artifact, 89

Fixed effects:
                  Estimate Std. Error t value
as.factor(Rater)1  2.37794    0.09658   24.62
as.factor(Rater)2  2.64891    0.09564   27.70
as.factor(Rater)3  2.28355    0.09658   23.64

Correlation of Fixed Effects:
            a.(R)1 a.(R)2
as.fctr(R)2 0.263
as.fctr(R)3 0.265  0.263
```

Plot 15

From the summary of model above, we can see that we do not have an intercept (different mean rating for rubric VisOrg from all three raters) for this model, but we have clearly different mean rating from three raters on this rubric 2.38, 2.65, 2.28 from Rater 1,2,3.

Technical Appendix part (3)(iv): page 32-47

Finally, we combine all rubrics and consider it as a whole to check the interaction of the variable Rubric by trying to add fixed effects, interactions, and new random effects to the “combined” intercept-only model including Rubric as a random effect grouped by artifacts using all the data. $/(Rating)\ 1 + (0 + Rubric|Artifact)/$

By several round of selection using ANOVA test comparing AIC, BIC, loglikelihood test, the final model we choose is $/as.numeric(Rating)\ (0 + Rubric|Artifact) + (0 + as.factor(Rater)|Artifact) + as.factor(Rater) + Semester + Rubric + as.factor(Rater)\ :\ Rubric/$ Coefficients are shown below:

```
##                                    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
```

Table 15

By the coefficients of random effects on the model, we can tell that for each artifact, how each rater rates differently from the mean and how ratings different from the mean rating based on different rubric. We are not going to show the random variable coefficients because there are so many values there: one value for each artifact rated by one rater under one rubric. But the general rule of interpreting random effect will be the same as what we have told previously.

To interpret the fixed effects, interactions shown above, we can say that: for fixed effect Rater, what we expect is that the overall ratings from Rater 2 is 0.37 higher than Rater 2, and the ratings from Rater 3 is 0.2 higher than Rater 1. For fixed effect Semester, what we expect based on the dataset we have is that the mean rating for spring semester will be 0.16 lower than fall semester. For fixed effect Rubric, what we expect is the mean rating for rubric InitEDA will be 0.74 higher than CritDes, InterpRes will be 1 higher, RsrchQ will be 0.73 higher, SelMeth will be 0.41 higher, TxtOrg will be 1.01 higer, VisOrg will be 0.65 higher. For interaction term Rater:Rubric, what we expect is that the overall rating from Rater 2 on rubric InitEDA will be 0.3 lower than overall rating from Rater 1 on rubric CritDes. And other interpretations are similar to this.

(4) Is there anything else interesting to say about this data?

Technical Appendix part (4): page 47-54

Maybe we can also try to see if there is any tendency on the gender that if female or male tend to get distinguishable higher/lower ratings for these different rubrics and maybe later for different artifacts and other variables. From the distribution plots, roughly same amount of female and male can get 2/3 for all rubrics, but also from some rubrics, only male/female or mostly male/female get 4.0.

Discussion

The distribution of CritDes is distinguishable from other rubrics and distributions of other rubrics are indistinguishable. For rubric CritDes, we can see that largest proportion of students get score 0; however, for all other rubrics, most students get score 2 or 3 and only few get 0 or 4, and this difference makes CritDes distinguishable from others. Raters tend to give lower score for rubric CritDes than other rubrics. From everything we have above, I will say that all 3 raters are indistinguishable as they all have really similar patterns and none of them tends to give a lower/higher score.

From the ICC values calculated, the reliability among raters for each rubric is not high, we cannot say that they agree with each other. Also because raters are not distinguishable from each other, we can know that some of them disagree with each other for some rubrics and agree for some rubrics, and we do not know for which rubric they agree with.

When considering Rubric seperately, we can find that Rating is related to Rater and Semester for rubric SelMeth, and for rubrics InitEDA, VisOrg, and CritDes, Rating is related to Rater. When considering Rubric as a whole, Rating is related to Rubric, Rater, Semester, interaction of Rater and Semester, random effects Rater and Rubric. The final model is $/as.numeric(Rating)\ (0+Rubric|Artifact)+(0+as.factor(Rater)|Artifact)+ as.factor(Rater) + Semester + Rubric + as.factor(Rater) : Rubric/$

Overall, to improve the overall performance and fairness of rating, Rater should not be a significant factor that can affect the Rating, and the department should hold pre-training to raters to eliminate the possiblity that raters disagree each other too much and the overall rating for each is too different than the mean rating.

strengths: Thorough exploration is made on checking the effect of raters and rubrics on the ratings. Model is selected from a lot of and different tests to check the significance of variables and the necessity of fixed effects, interactions, and random effects.

limitations: Have not find a way to figure out the difference in the models fitted to the data from the 13 common items and the models fitting to all the data. Have not find ways to deal with missing values. Have not really check the validity of the models selected.

To further improve the model and the understanding of this project, maybe later in the future we can try other difference models after the missing values and other problems are addressed, and models should be selected also based on the validity.

Reference:

Junker, B. W. (2021). Project 02 assignment sheet and data for 36-617: Applied Regression Analysis. Department of Statistics and Data Science, Carnegie Mellon University, Pittsburgh PA. Accessed Nov 29, 2021 from https://canvas.cmu.edu/courses/25337/files/folder/Project02

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

Technical Appendix

Yuqing Xu

11/17/2021

```
library(arm)
```

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

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

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

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

```
## Working directory is /Users/abcdefg/Documents/applied linear models
```

```
library(plyr)
library(ggplot2)
```

Project 2

Data

```
ratings = read.csv("/Users/abcdefg/Documents/applied linear models/ratings.csv")

tall = read.csv("/Users/abcdefg/Documents/applied linear models/tall.csv")
```

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

# address missing values
tall$Sex[nchar(tall$Sex)==0] <- "--"
ratings.13 <- ratings[grep("O",ratings$Artifact),]
tall.13 <-  tall[grep("O",tall$Artifact),]

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
```

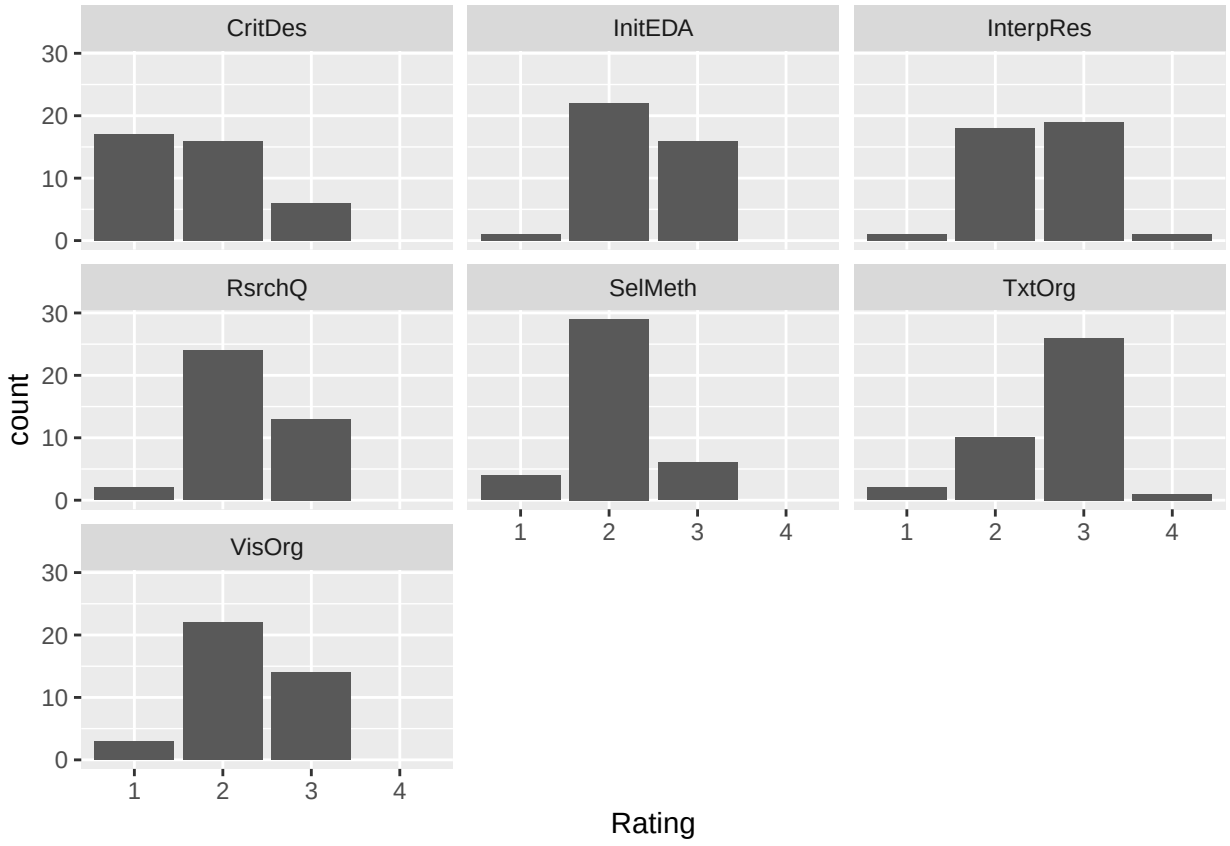
```
summary(ratings) # to two tables
```

```
##        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 SelMeth InterpRes  VisOrg   TxtOrg
##  Length:117         1: 6   1   :47   1: 8    1:10    1: 6      1   : 7   1: 8
##  Class :character   2:65   2   :39   2:56    2:89    2:49      2   :59   2:37
##  Mode  :character   3:45   3   :28   3:47    3:18    3:61      3   :45   3:66
##                     4: 1   4   : 2   4: 6    4: 0    4: 1      4   : 5   4: 6
##                            NA's: 1                             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
##
```

(1)

This can be checked by ICC values and we will also need to compare those rated by all three raters and others seperately.

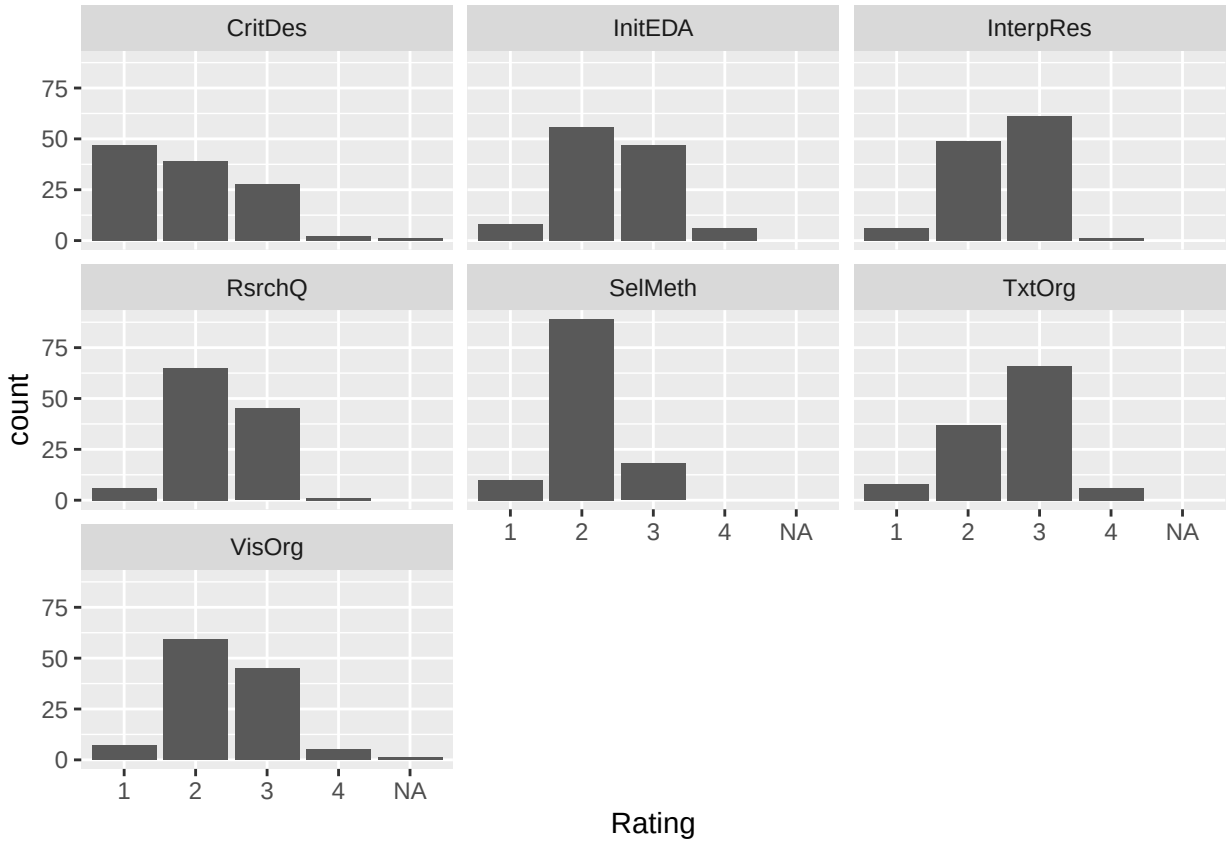
```
# compare distributions across Rubrics in two datasets
g <- ggplot(tall.13,aes(x = Rating)) +
  facet_wrap( ~ Rubric) +
  geom_bar()

g
```




```
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
```

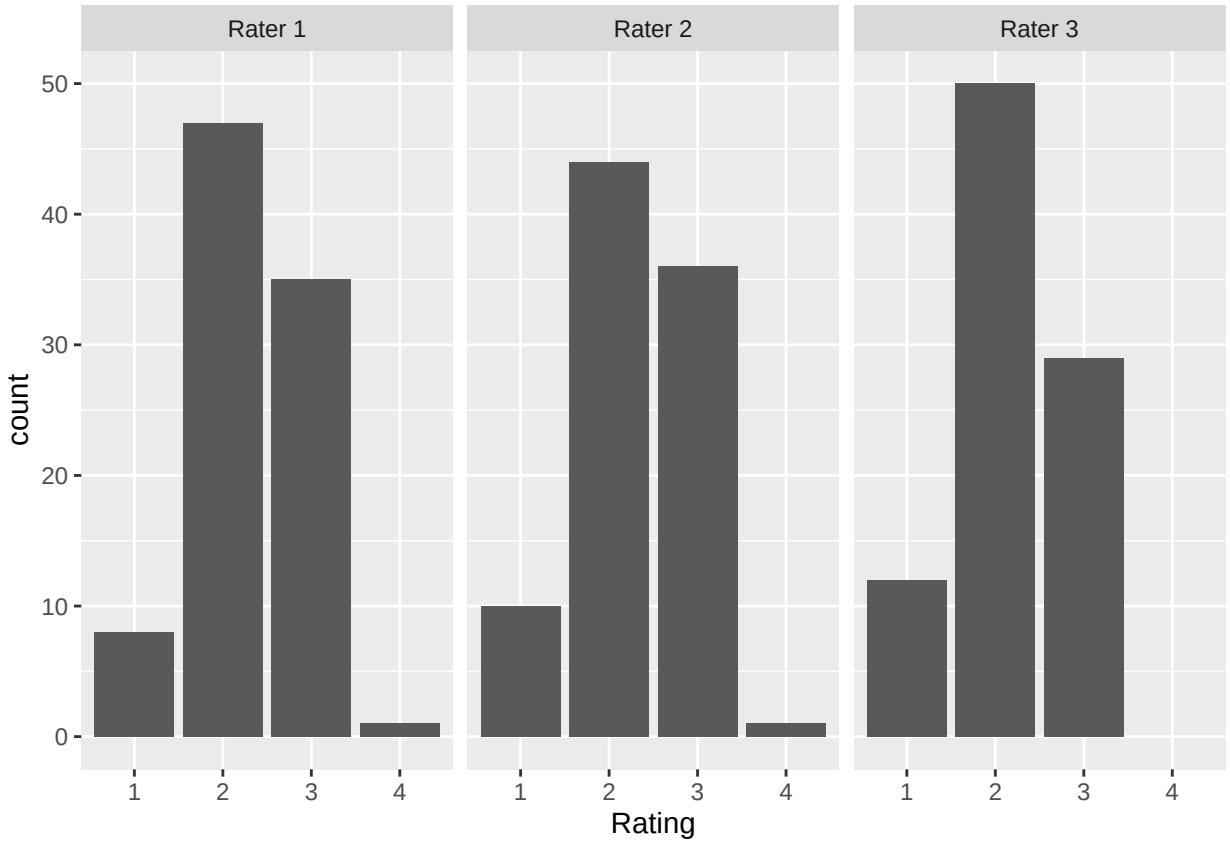
```
g <- ggplot(tall,aes(x = Rating)) +
  facet_wrap( ~ Rubric) +
  geom_bar()

g
```




```
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
```

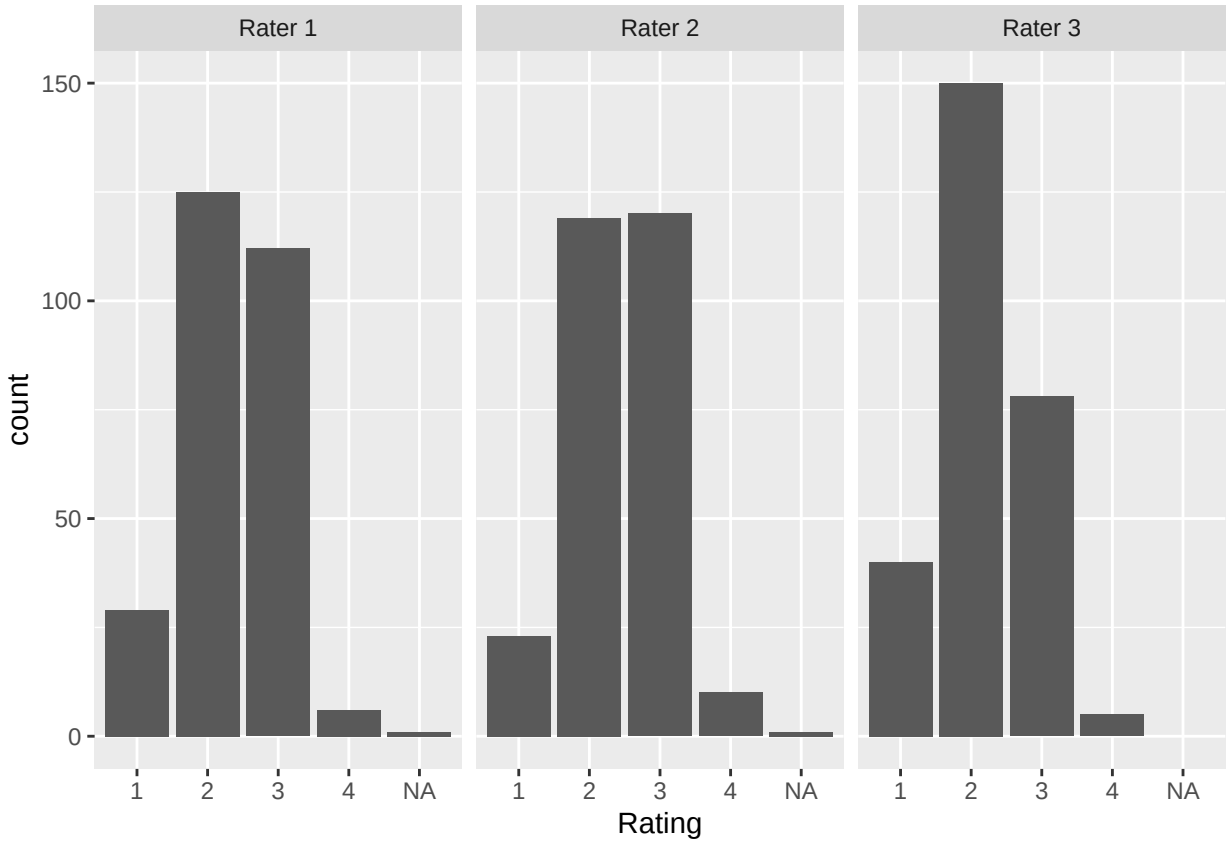
```
# compare distribution across raters
rater.name <- function(x) { paste("Rater",x) }


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

g
```




```
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
```

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

g
```




```
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
```

(2)

Measure the intraclass correlation, ICC value, to find out if the raters agree with each other.

```
#View(tall)
c <- tall[grep("O",tall$Artifact),]
```

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

RsrchQ_mod = lmer(Rating ~ 1 + (1|Rater), data = RsrchQ_ratings)
CritDes_mod = lmer(Rating ~ 1 + (1|Rater), data = CritDes_ratings)
InitEDA_mod = lmer(Rating ~ 1 + (1|Rater), data = InitEDA_ratings)
SelMeth_mod = lmer(Rating ~ 1 + (1|Rater), data = SelMeth_ratings)
InterpRes_mod = lmer(Rating ~ 1 + (1|Rater), data = InterpRes_ratings)
VisOrg_mod = lmer(Rating ~ 1 + (1|Rater), data = VisOrg_ratings)
TxtOrg_mod = lmer(Rating ~ 1 + (1|Rater), data = TxtOrg_ratings)



### RsrchQ


Repeated <- ratings[ratings$Repeated==1,]

RsrchQ_r1_r2 <- data.frame(r1 = Repeated$RsrchQ[Repeated$Rater == 1], r2 = Repeated$RsrchQ[Repeated$Rat
                           a1 = Repeated$Artifact[Repeated$Rater == 1], a2 = Repeated$Artifact[Repeated$
r1 <- factor(RsrchQ_r1_r2$r1, levels = 1:4)
r2 <- factor(RsrchQ_r1_r2$r2, levels = 1:4)
table(r1,r2)




RsrchQ_r2_r3 <- data.frame(r2 = Repeated$RsrchQ[Repeated$Rater == 2], r3 = Repeated$RsrchQ[Repeated$Rat
                           a2 = Repeated$Artifact[Repeated$Rater == 2], a3 = Repeated$Artifact[Repeated$
r2 <- factor(RsrchQ_r2_r3$r2, levels = 1:4)
r3 <- factor(RsrchQ_r2_r3$r3, levels = 1:4)
table(r2,r3)


library(performance)
RsrchQ_ratings <- c[c$Rubric == "RsrchQ",]
RsrchQ_mod = lmer(Rating ~ 1 + (1|Rater), data=RsrchQ_ratings)
summary(RsrchQ_mod)
performance::icc(RsrchQ_mod)


### CritDes


CritDes_r1_r2 <- data.frame(r1 = Repeated$CritDes[Repeated$Rater == 1], r2 = Repeated$CritDes[Repeated$
                           a1 = Repeated$Artifact[Repeated$Rater == 1], a2 = Repeated$Artifact[Repeated$
r1 <- factor(CritDes_r1_r2$r1, levels = 1:4)
r2 <- factor(CritDes_r1_r2$r2, levels = 1:4)
```

```r
table(r1,r2)


CritDes_r2_r3 <- data.frame(r2 = Repeated$CritDes[Repeated$Rater == 2], r3 = Repeated$CritDes[Repeated$R
                            a2 = Repeated$Artifact[Repeated$Rater == 2], a3 = Repeated$Artifact[Repeated$
r2 <- factor(CritDes_r2_r3$r2, levels = 1:4)
r3 <- factor(CritDes_r2_r3$r3, levels = 1:4)
table(r2,r3)


CritDes_ratings <- c[c$Rubric == "CritDes",]
CritDes_mod = lmer(Rating ~ 1 + (1|Rater), data=CritDes_ratings)
summary(CritDes_mod)
performance::icc(CritDes_mod)

### InitEDA


InitEDA_r1_r2 <- data.frame(r1 = Repeated$InitEDA[Repeated$Rater == 1], r2 = Repeated$InitEDA[Repeated$R
                            a1 = Repeated$Artifact[Repeated$Rater == 1], a2 = Repeated$Artifact[Repeated$
r1 <- factor(InitEDA_r1_r2$r1, levels = 1:4)
r2 <- factor(InitEDA_r1_r2$r2, levels = 1:4)
table(r1,r2)

InitEDA_r2_r3 <- data.frame(r2 = Repeated$InitEDA[Repeated$Rater == 2], r3 = Repeated$InitEDA[Repeated$R
                            a2 = Repeated$Artifact[Repeated$Rater == 2], a3 = Repeated$Artifact[Repeated$
r2 <- factor(InitEDA_r2_r3$r2, levels = 1:4)
r3 <- factor(InitEDA_r2_r3$r3, levels = 1:4)
table(r2, r3)

InitEDA_ratings <- c[c$Rubric == "InitEDA",]
InitEDA_mod = lmer(Rating ~ 1 + (1|Rater), data=InitEDA_ratings)
summary(InitEDA_mod)
performance::icc(InitEDA_mod)

### SelMeth


SelMeth_r1_r2 <- data.frame(r1 = Repeated$SelMeth[Repeated$Rater == 1], r2 = Repeated$SelMeth[Repeated$R
                            a1 = Repeated$Artifact[Repeated$Rater == 1], a2 = Repeated$Artifact[Repeated$
r1 <- factor(SelMeth_r1_r2$r1, levels = 1:4)
r2 <- factor(SelMeth_r1_r2$r2, levels = 1:4)
table(r1,r2)

SelMeth_r2_r3 <- data.frame(r2 = Repeated$SelMeth[Repeated$Rater == 2], r3 = Repeated$SelMeth[Repeated$R
                            a2 = Repeated$Artifact[Repeated$Rater == 2], a3 = Repeated$Artifact[Repeated$
r2 <- factor(SelMeth_r2_r3$r2, levels = 1:4)
r3 <- factor(SelMeth_r2_r3$r3, levels = 1:4)
table(r2, r3)


SelMeth <- c[c$Rubric == "SelMeth",]
SelMeth_mod = lmer(Rating ~ 1 + (1|Rater), data=SelMeth_ratings)
```

```r
summary(SelMeth_mod)
performance::icc(SelMeth_mod)


### InterpRes

InterpRes_r1_r2 <- data.frame(r1 = Repeated$InterpRes[Repeated$Rater == 1], r2 = Repeated$InterpRes[Repe
                              a1 = Repeated$Artifact[Repeated$Rater == 1], a2 = Repeated$Artifact[Repeate
r1 <- factor(InterpRes_r1_r2$r1, levels = 1:4)
r2 <- factor(InterpRes_r1_r2$r2, levels = 1:4)
table(r1,r2)



InterpRes_r2_r3 <- data.frame(r2 = Repeated$InterpRes[Repeated$Rater == 2], r3 = Repeated$InterpRes[Repe
                              a2 = Repeated$Artifact[Repeated$Rater == 2], a3 = Repeated$Artifact[Repeate
r2 <- factor(InterpRes_r2_r3$r2, levels = 1:4)
r3 <- factor(InterpRes_r2_r3$r3, levels = 1:4)
table(r2, r3)

InterpRes <- c[c$Rubric == "InterpRes",]
InterpRes_mod = lmer(Rating ~ 1 + (1|Rater), data=InterpRes_ratings)
summary(InterpRes_mod)
performance::icc(InterpRes_mod)


### VisOrg


VisOrg_r1_r2 <- data.frame(r1 = Repeated$VisOrg[Repeated$Rater == 1], r2 = Repeated$VisOrg[Repeated$Rate
                              a1 = Repeated$Artifact[Repeated$Rater == 1], a2 = Repeated$Artifact[Repeate
r1 <- factor(VisOrg_r1_r2$r1, levels = 1:4)
r2 <- factor(VisOrg_r1_r2$r2, levels = 1:4)
table(r1,r2)

VisOrg_r2_r3 <- data.frame(r2 = Repeated$VisOrg[Repeated$Rater == 2], r3 = Repeated$VisOrg[Repeated$Rate
                              a2 = Repeated$Artifact[Repeated$Rater == 2], a3 = Repeated$Artifact[Repeate
r2 <- factor(VisOrg_r2_r3$r2, levels = 1:4)
r3 <- factor(VisOrg_r2_r3$r3, levels = 1:4)
table(r2, r3)

VisOrg <- c[c$Rubric == "VisOrg",]
VisOrg_mod = lmer(Rating ~ 1 + (1|Rater), data=VisOrg_ratings)
summary(VisOrg_mod)
performance::icc(VisOrg_mod)

### TxtOrg


TxtOrg_r1_r2 <- data.frame(r1 = Repeated$TxtOrg[Repeated$Rater == 1], r2 = Repeated$TxtOrg[Repeated$Rate
                              a1 = Repeated$Artifact[Repeated$Rater == 1], a2 = Repeated$Artifact[Repeate
r1 <- factor(TxtOrg_r1_r2$r1, levels = 1:4)
r2 <- factor(TxtOrg_r1_r2$r2, levels = 1:4)
```

```r
table(r1,r2)

TxtOrg_r2_r3 <- data.frame(r2 = Repeated$TxtOrg[Repeated$Rater == 2], r3 = Repeated$TxtOrg[Repeated$Rat
                           a2 = Repeated$Artifact[Repeated$Rater == 2], a3 = Repeated$Artifact[Repeate
r2 <- factor(TxtOrg_r2_r3$r2, levels = 1:4)
r3 <- factor(TxtOrg_r2_r3$r3, levels = 1:4)
table(r2, r3)


TxtOrg <- c[c$Rubric == "TxtOrg",]
TxtOrg_mod = lmer(Rating ~ 1 + (1|Rater), data=TxtOrg_ratings)
summary(TxtOrg_mod)
performance::icc(TxtOrg_mod)
```

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

ICC.vec <- NULL
for (i in Rubric.names) {

  tmp <- lmer(as.numeric(Rating) ~ 1 + (1|Artifact), data=tall.13[tall.13$Rubric==i,])
  sig2 <- summary(tmp)$sigma^2
  tau2 <- attr(summary(tmp)$varcor[[1]],"stddev")^2
  ICC <- tau2 / (tau2 + sig2)
  ICC.vec <- c(ICC.vec,ICC)
}
names(ICC.vec) <- Rubric.names

agreement.results <- cbind(ICC.common=ICC.vec,"       a12"=0,a23=0,a13=0)

agreement.tables <- as.list(rep(NA,7))
names(agreement.tables) <- Rubric.names

for (i in Rubric.names) {
  r12 <- data.frame(r1=factor(ratings.13[ratings.13$Rater==1,i],levels=1:4),
                    r2=factor(ratings.13[ratings.13$Rater==2,i],levels=1:4),
                    a1=ratings.13[ratings.13$Rater==1,"Artifact"],
                    a2=ratings.13[ratings.13$Rater==2,"Artifact"])
  if(any(r12[,3]!=r12[,4])) { stop(paste("Rater 1-2 Artifact mismatch on rubric",i)) }
  a12 <- mean(r12[,1]==r12[,2])
  r12 <- table(r12[,1:2])  ## print this to see how much agreement there is among raters 1-2

  r23 <- data.frame(r2=factor(ratings.13[ratings.13$Rater==2,i],levels=1:4),
                    r3=factor(ratings.13[ratings.13$Rater==3,i],levels=1:4),
                    a2=ratings.13[ratings.13$Rater==2,"Artifact"],
                    a3=ratings.13[ratings.13$Rater==3,"Artifact"])
  if(any(r23[,3]!=r23[,4])) { stop(paste("Rater 2-3 Artifact mismatch on rubric",i)) }
  a23 <- mean(r23[,1]==r23[,2])
  r23 <- table(r23[,1:2])  ## print this to see how much agreement there is among raters 2-3

  r13 <- data.frame(r1=factor(ratings.13[ratings.13$Rater==1,i],levels=1:4),
                    r3=factor(ratings.13[ratings.13$Rater==3,i],levels=1:4),
                    a1=ratings.13[ratings.13$Rater==1,"Artifact"],
                    a3=ratings.13[ratings.13$Rater==3,"Artifact"])
```

```
  if(any(r13[,3]!=r13[,4])) { stop(paste("Rater 1-3 Artifact mismatch on rubric",i)) }
  a13 <- mean(r13[,1]==r13[,2])
  r13 <- table(r13[,1:2])  ## print this to see how much agreement there is among raters 1-3

  agreement.results[i,2:4] <- c(a12,a23,a13)

  agreement.tables[[i]] <- list(r12,r23,r13)

}
round(agreement.results,2)
```

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

```
##

if (F) { print(agreement.tables) }

ICC.vec <- NULL
for (i in Rubric.names) {

  tmp <- lmer(as.numeric(Rating) ~ 1 + (1|Artifact), data=tall[tall$Rubric==i,])
  sig2 <- summary(tmp)$sigma^2
  tau2 <- attr(summary(tmp)$varcor[[1]],"stddev")^2
  ICC <- tau2 / (tau2 + sig2)
  ICC.vec <- c(ICC.vec,ICC)
}
names(ICC.vec) <- Rubric.names

agreement.results <- cbind(ICC.alldata=ICC.vec,agreement.results)

round(agreement.results,2)
```

```
##           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
```

Above we get the ICC value of full dataset and the subset including only those rated by all three raters.

(3)

3(i): Adding fixed effects to the seven rubric-specific models using just the data from the 13 common artifacts that all three raters saw.

```
library(LMERConvenienceFunctions)
library(RLRsim)

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



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
```

```
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
```

```
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) {

  ## 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)
```

All models with fixed effect are not better than intercept-only models. And we will not need to try other terms on the models based on 13 common dataset.

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

Now let's try with the full data...

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

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)

  ## 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
```

Here we can see that for some rubrics, the intercept only models are adequate and for others they are not.

3(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?
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
```

```
## apparently they do.

## 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
```

```
## looks like we do, so we keep "tmp" as our best model so far...

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

tmp.fixed_interactions <- update(tmp, . ~ . + as.factor(Rater)*Semester - Semester)
## I've specified the model so that I can see (a) a different intercept for each
## rater, and (b) a different semester effect for each rater.

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):Se
##                         npar    AIC    BIC  logLik deviance Chisq Df Pr(>Chisq)
## tmp                        6 142.05 158.58 -65.027   130.05
## tmp.fixed_interactions     8 143.46 165.49 -63.731   127.46 2.592  2     0.2736
```

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

## Finally we check for random effects.  We should only add random effects that
## are also present as fixed effects.  This means, for this model, we should try
## (Rater|Artifact) and (Semester|Artifact).

## I will show how to test these with exactRLRT()...

## Testing (Semester|Artifact)...

m0 <- tmp                                       ## Null hypothesis
mA <- update(m0, . ~ . + (Semester|Artifact))   ## Alternative hypotheses
```

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

```
m  <- update(mA, . ~ . - (1|Artifact))          ## Model with only the new R.E.
```

```
## Error in h(simpleError(msg, call)): error in evaluating the argument 'object' in selecting a method
```

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

```
## Error in exactRLRT(m0 = m0, mA = mA, m = m): object 'm' not found
```

```
## Many error messages!  But note what the first one, for model mA is: there are
## more random effects than there are observations in the data set!  As explained
## on Piazza, this means 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 is needed.



## 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 testing is needed.


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

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
```

```
## You would need to do something similar with the other models that are not
## just intercept-only models (i.e the models for CritDes and InterpRes)

## ALSO, please don't forget:  I am not giving interpretations to the model fits
## or coefficient estimates here. That is something I'm leaving for you, as you
## complete your analyses and write an IDMRAD paper for the college.
```

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

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

```
## apparently they do.

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

```
## looks like we do, so we keep "tmp" as our best model so far...

## now let's check for fixed-effect interactions... Since only Rater
## is involved, we do not need this step
```

```
#tmp.fixed_interactions <- update(tmp, . ~ . + as.factor(Rater)*Semester - Semester)
## I've specified the model so that I can see (a) a different intercept for each
## rater, and (b) a different semester effect for each rater.

#anova(tmp,tmp.fixed_interactions)

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

## Finally we check for random effects.  We should only add random effects that
## are also present as fixed effects.  This means, for this model, we should try
## (Rater|Artifact).

## I will show how to test these with exactRLRT()...



## 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)
## Many error messages!  But note what the first one, for model mA is: there are
## more random effects than there are observations in the data set!  As explained
## on Piazza, this means lmer() cannot fit a model.  Thus, the model
##
##
## as.numeric(Rating) ~ -1 + as.factor(Rater) + Semester +
##                           (1 | Artifact) + (as.factor(Rater) | Artifact)
##
## isn't even possible, so no testing is needed.


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

summary(tmp)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: as.numeric(Rating) ~ as.factor(Rater) + (1 | Artifact) - 1
##    Data: tall.nonmissing[tall.nonmissing$Rubric == "CritDes", ]
##
## REML criterion at convergence: 271
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -1.55495 -0.50027 -0.08228  0.64663  1.60935
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  Artifact (Intercept) 0.4349   0.6595
```

```
##  Residual             0.2473   0.4972
## Number of obs: 115, groups:  Artifact, 89
##
## Fixed effects:
##                    Estimate Std. Error t value
## as.factor(Rater)1   1.6863     0.1207   13.98
## as.factor(Rater)2   2.1129     0.1219   17.34
## as.factor(Rater)3   1.8908     0.1219   15.51
##
## Correlation of Fixed Effects:
##             a.(R)1 a.(R)2
## as.fctr(R)2 0.244
## as.fctr(R)3 0.244  0.246
```

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

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

```
## apparently they do.

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

```
## looks like we do, so we keep "tmp" as our best model so far...

## now let's check for fixed-effect interactions... Since only Rater
## is involved, we do not need this step

#tmp.fixed_interactions <- update(tmp, . ~ . + as.factor(Rater)*Semester - Semester)
## I've specified the model so that I can see (a) a different intercept for each
```

```
## rater, and (b) a different semester effect for each rater.

#anova(tmp,tmp.fixed_interactions)

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

## Finally we check for random effects.  We should only add random effects that
## are also present as fixed effects.  This means, for this model, we should try
## (Rater|Artifact).

## I will show how to test these with exactRLRT()...



## 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)
## Many error messages!  But note what the first one, for model mA is: there are
## more random effects than there are observations in the data set!  As explained
## on Piazza, this means lmer() cannot fit a model.  Thus, the model
##
##
## as.numeric(Rating) ~ -1 + as.factor(Rater) + Semester +
##                           (1 | Artifact) + (as.factor(Rater) | Artifact)
##
## isn't even possible, so no testing is needed.


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

summary(tmp)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: as.numeric(Rating) ~ as.factor(Rater) + (1 | Artifact) - 1
##    Data: tall.nonmissing[tall.nonmissing$Rubric == "InterpRes", ]
##
## REML criterion at convergence: 199.7
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.5317 -0.7627  0.2635  0.6614  2.6535
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  Artifact (Intercept) 0.06224  0.2495
##  Residual             0.25250  0.5025
## Number of obs: 116, groups:  Artifact, 90
```

```
##
## Fixed effects:
##                   Estimate Std. Error t value
## as.factor(Rater)1  2.70421    0.08912   30.34
## as.factor(Rater)2  2.58574    0.08912   29.01
## as.factor(Rater)3  2.13918    0.09027   23.70
##
## Correlation of Fixed Effects:
##             a.(R)1 a.(R)2
## as.fctr(R)2 0.061
## as.fctr(R)3 0.062  0.062
```

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

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

```
## apparently they do.

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

```
## looks like we do, so we keep "tmp" as our best model so far...

## now let's check for fixed-effect interactions... Since only Rater
## is involved, we do not need this step

#tmp.fixed_interactions <- update(tmp, . ~ . + as.factor(Rater)*Semester - Semester)
## I've specified the model so that I can see (a) a different intercept for each
## rater, and (b) a different semester effect for each rater.
```

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

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

## Finally we check for random effects.  We should only add random effects that
## are also present as fixed effects.  This means, for this model, we should try
## (Rater|Artifact).

## I will show how to test these with exactRLRT()...



## 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)
## Many error messages!  But note what the first one, for model mA is: there are
## more random effects than there are observations in the data set!  As explained
## on Piazza, this means lmer() cannot fit a model.  Thus, the model
##
##
## as.numeric(Rating) ~ -1 + as.factor(Rater) + Semester +
##                           (1 | Artifact) + (as.factor(Rater) | Artifact)
##
## isn't even possible, so no testing is needed.


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

summary(tmp)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: as.numeric(Rating) ~ as.factor(Rater) + (1 | Artifact) - 1
##    Data: tall.nonmissing[tall.nonmissing$Rubric == "VisOrg", ]
##
## REML criterion at convergence: 219.6
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -1.5004 -0.3365 -0.2483  0.3841  1.8552
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  Artifact (Intercept) 0.2907   0.5392
##  Residual             0.1467   0.3830
## Number of obs: 115, groups:  Artifact, 89
##
## Fixed effects:
```

```
##                   Estimate Std. Error t value
## as.factor(Rater)1  2.37794    0.09658   24.62
## as.factor(Rater)2  2.64891    0.09564   27.70
## as.factor(Rater)3  2.28355    0.09658   23.64
##
## Correlation of Fixed Effects:
##             a.(R)1 a.(R)2
## as.fctr(R)2 0.263
## as.fctr(R)3 0.265  0.263
```

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

Now we try something similar with the "combined" model suggested on p. 4 of the project assignment sheet.

```
## 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
```

```
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
## R complains that we have a "boundary (singular) fit", i.e. the
## variance-covariance matrix for the random effects is singular
## (not of full rank), or nearly singular.
##
## What this typically means is that some of the random effects are highly
## correlated with one another.  We can see this in the "Random effects"
## block of summary(comb.0):
##
## * 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
##
## etc.
##
## In some ways we should not be surprised: these rubrics all represent
## features of a good research report, and we would expect that if someone
## is good at one or two of these features, they are probably good at the
## others.

## Although the random effects are highly correlated, we can still proceed with
## our variable selection...

## 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
```

```
##
## It's interesting to note that comb.full is no longer a boundary (singular)
## fit.  Adding the fixed effects changed the residuals enough that the
## variance-covariance matrix for the random effects is no longer (nearly)
## singular.

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
```

```
## 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)
```

```
## 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.
## again, boundary fit (near-singular random effects variance-covariance mtx)

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 values; we could
## probably have just proceeded wth the model comb.inter.  But
## since we have the converged model we will use it for fixed
## effects selection

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 m
## 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
```

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

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

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

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

```r
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
```

```r
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
```

```r
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
```

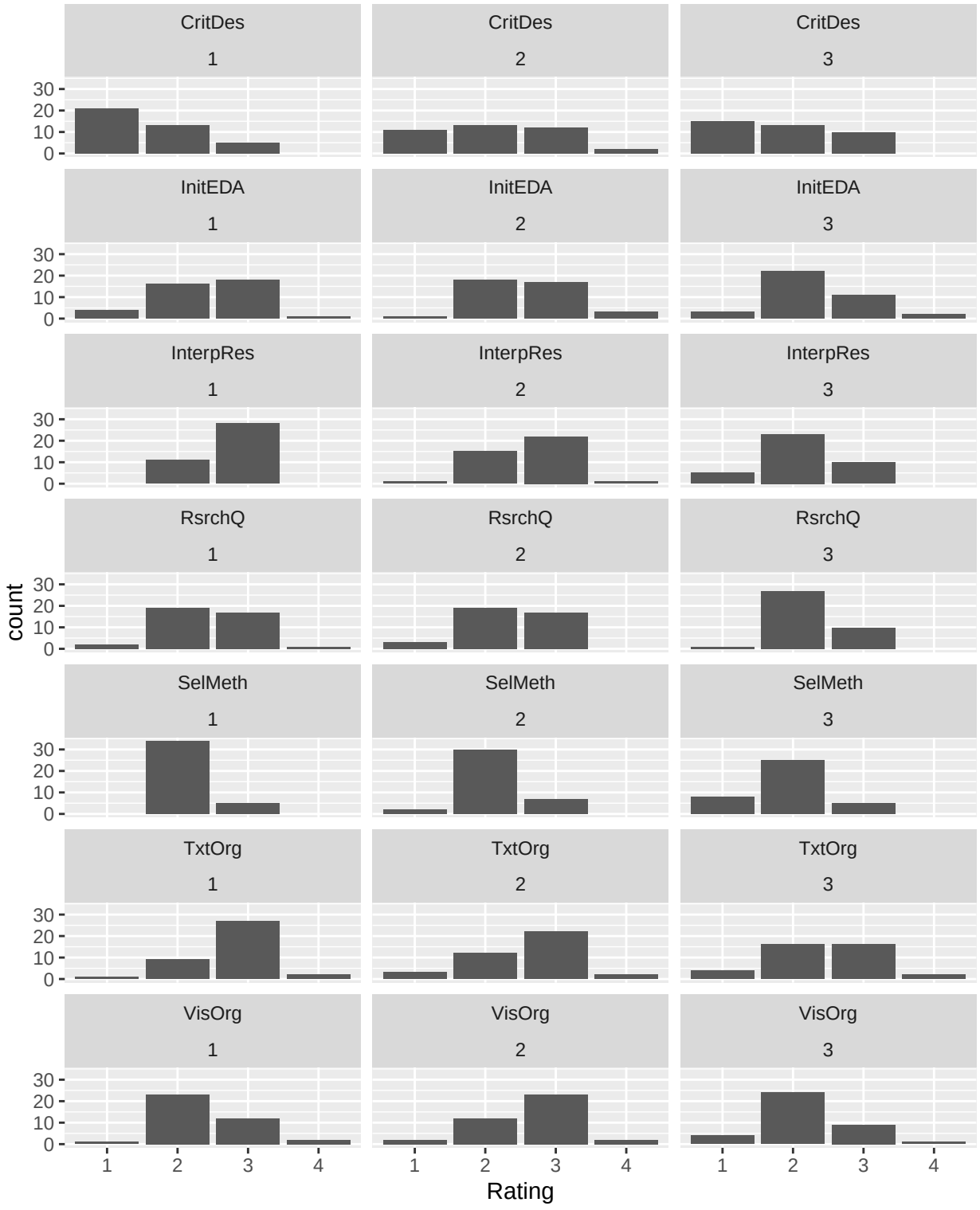
```r
anova(comb.back_elim,comb.inter_elim,comb.inter.u)
```

```
## 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.

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

```
## 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.  (Clearly it is
## not the case that one rater is simply more harsh than another, or something
## like that.  Can you describe the different patterns of scoring among the 3
## raters?  Can you relate it to coefficient estimates in summary()$coef?)
```

```
## 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
##
## We want to add each of these *without* a random intercept, to preserve the
## structure of the model (separate random interepts for each 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...

## Fitting some of these models produces various errors and warnings; I am not
## going to worry about them too much, in order to get an idea of what random
## effects I may want...

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(
## mA: as.numeric(Rating) ~ (0 + Rubric | Artifact) + (0 + as.factor(Rater) | Artifact) + as.factor(Rat
##    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(
## mA: as.numeric(Rating) ~ (0 + Rubric | Artifact) + (0 + Semester | Artifact) + as.factor(Rater) + Se
##    npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
## m0   51 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)

## anova(m0,mA)     -- Not needed!
##
## 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
##
##
##
##
```

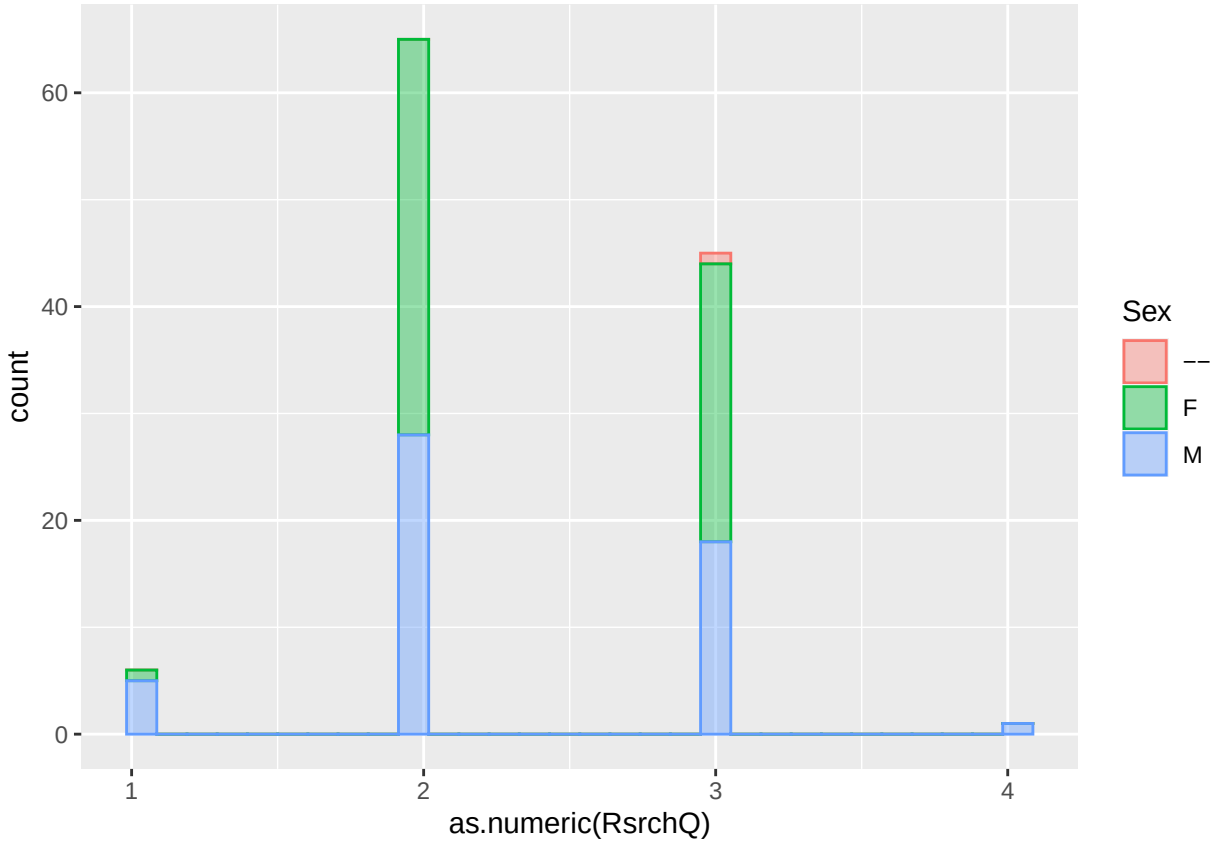
```
summary(comb.final)$coef
```

```
##                                   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.
```

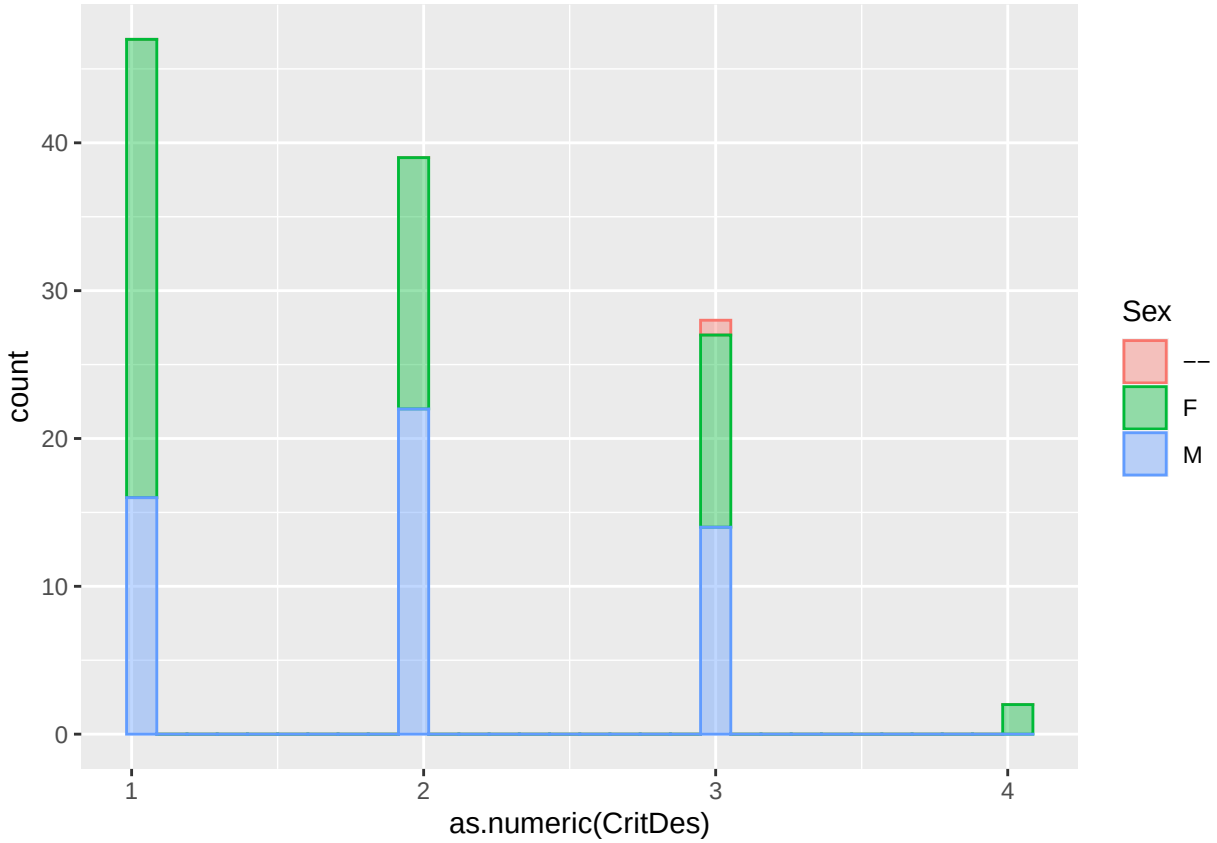
(4)

```
library(ggplot2)
ggplot(ratings, aes(x = as.numeric(RsrchQ))) + geom_histogram(aes(color = Sex, fill = Sex), bins = 30.0
```

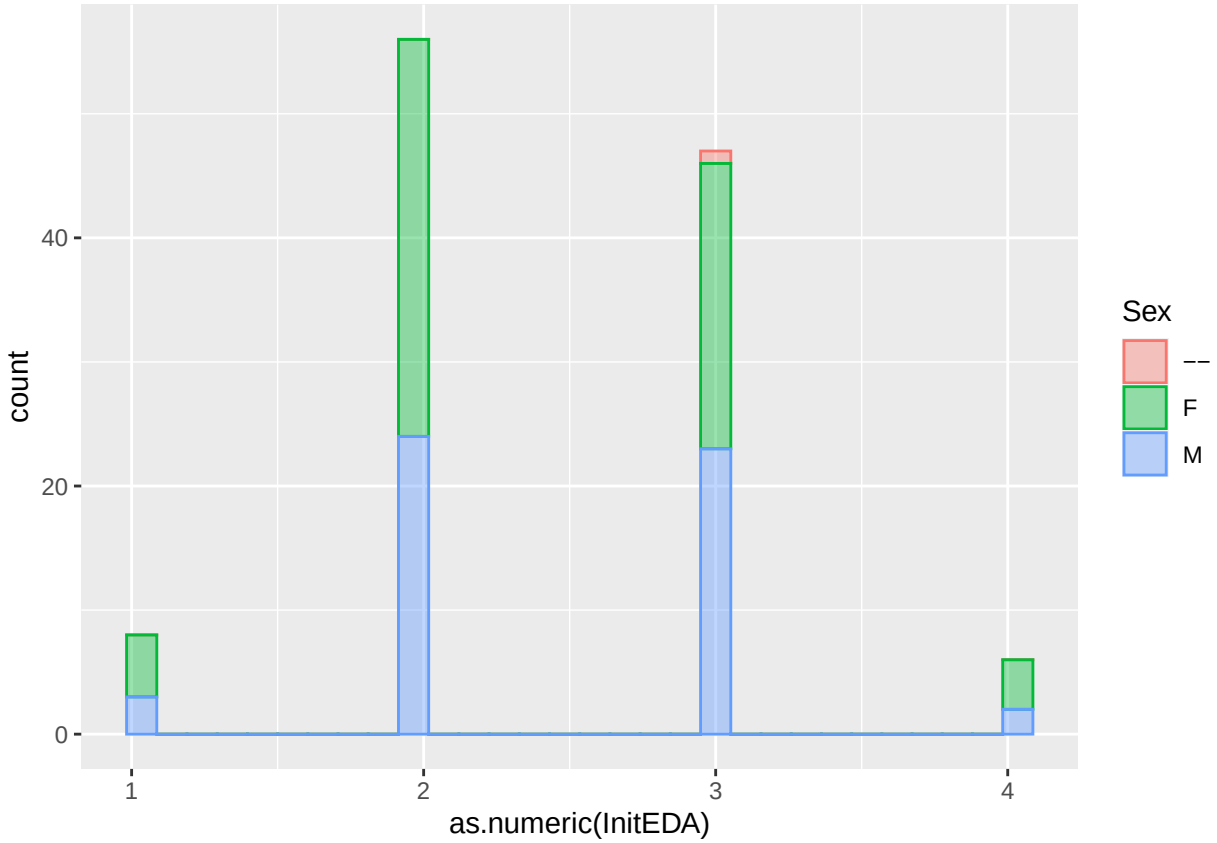



```
ggplot(ratings, aes(x = as.numeric(CritDes))) + geom_histogram(aes(color = Sex, fill = Sex), bins = 30.
```

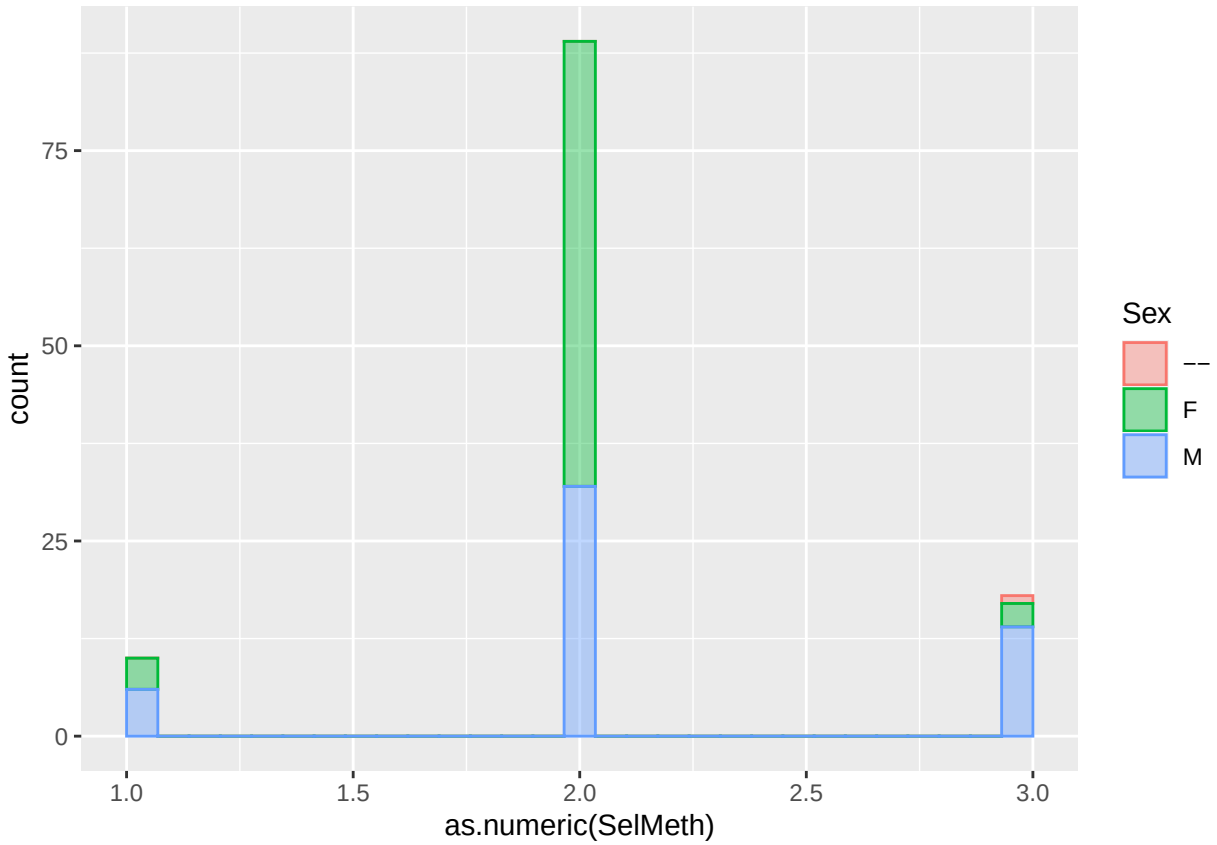
```
## Warning: Removed 1 rows containing non-finite values (stat_bin).
```



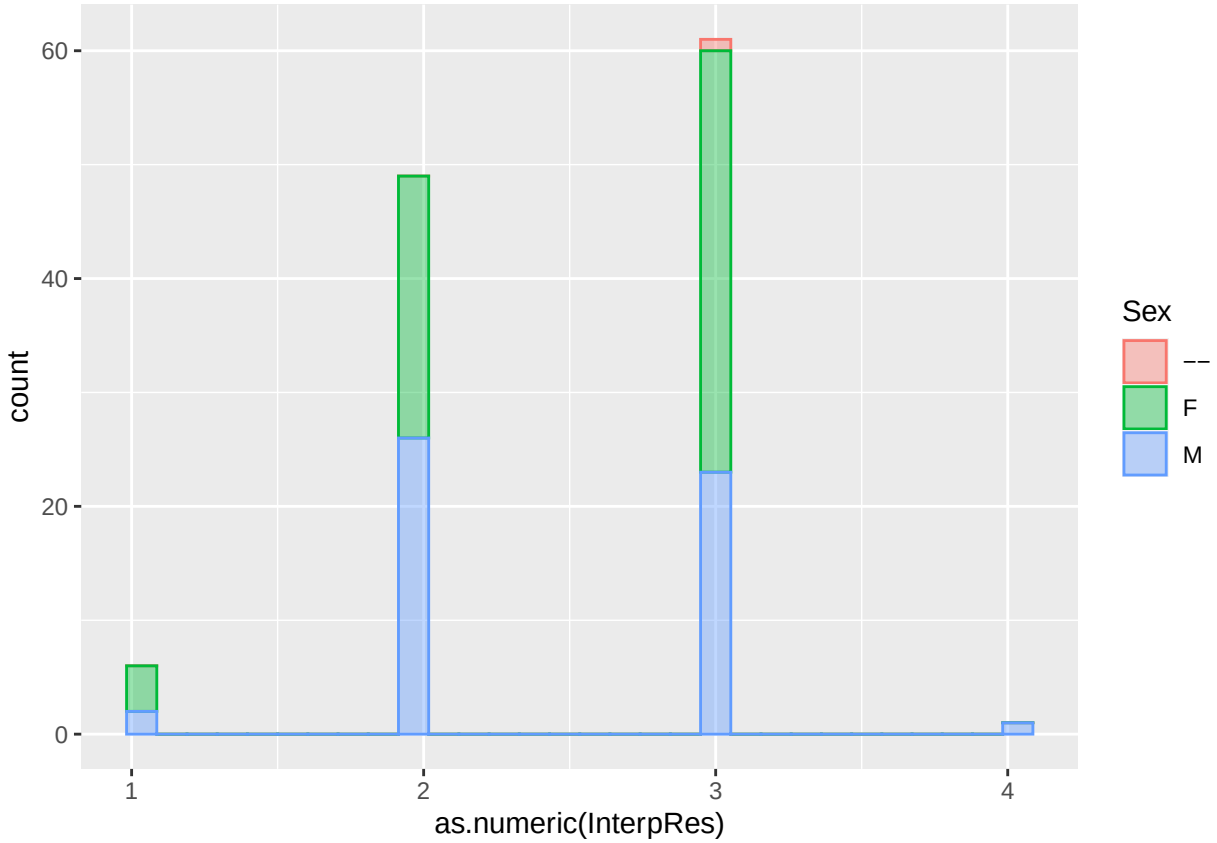

```
ggplot(ratings, aes(x = as.numeric(InitEDA))) + geom_histogram(aes(color = Sex, fill = Sex), bins = 30,
```




```
ggplot(ratings, aes(x = as.numeric(SelMeth))) + geom_histogram(aes(color = Sex, fill = Sex), bins = 30,
```

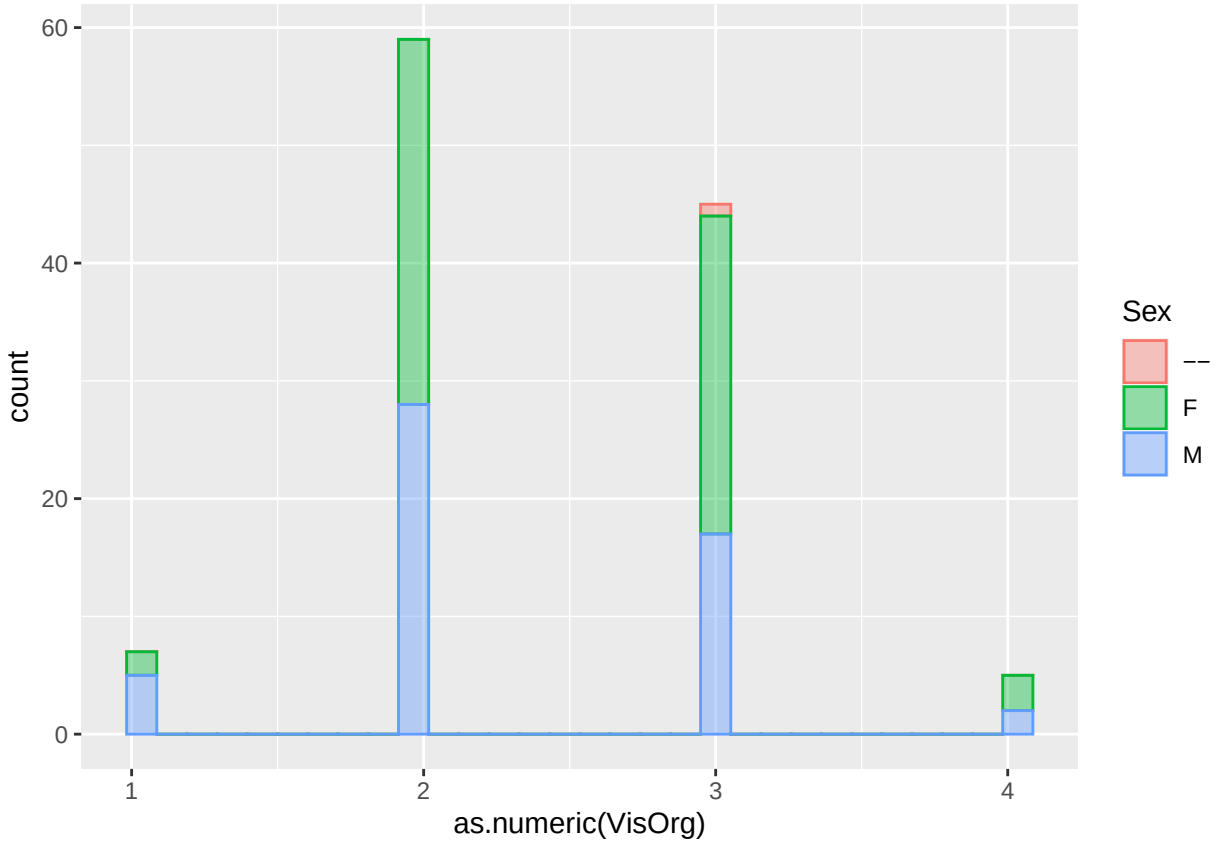



```
ggplot(ratings, aes(x = as.numeric(InterpRes))) + geom_histogram(aes(color = Sex, fill = Sex), bins = 3
```

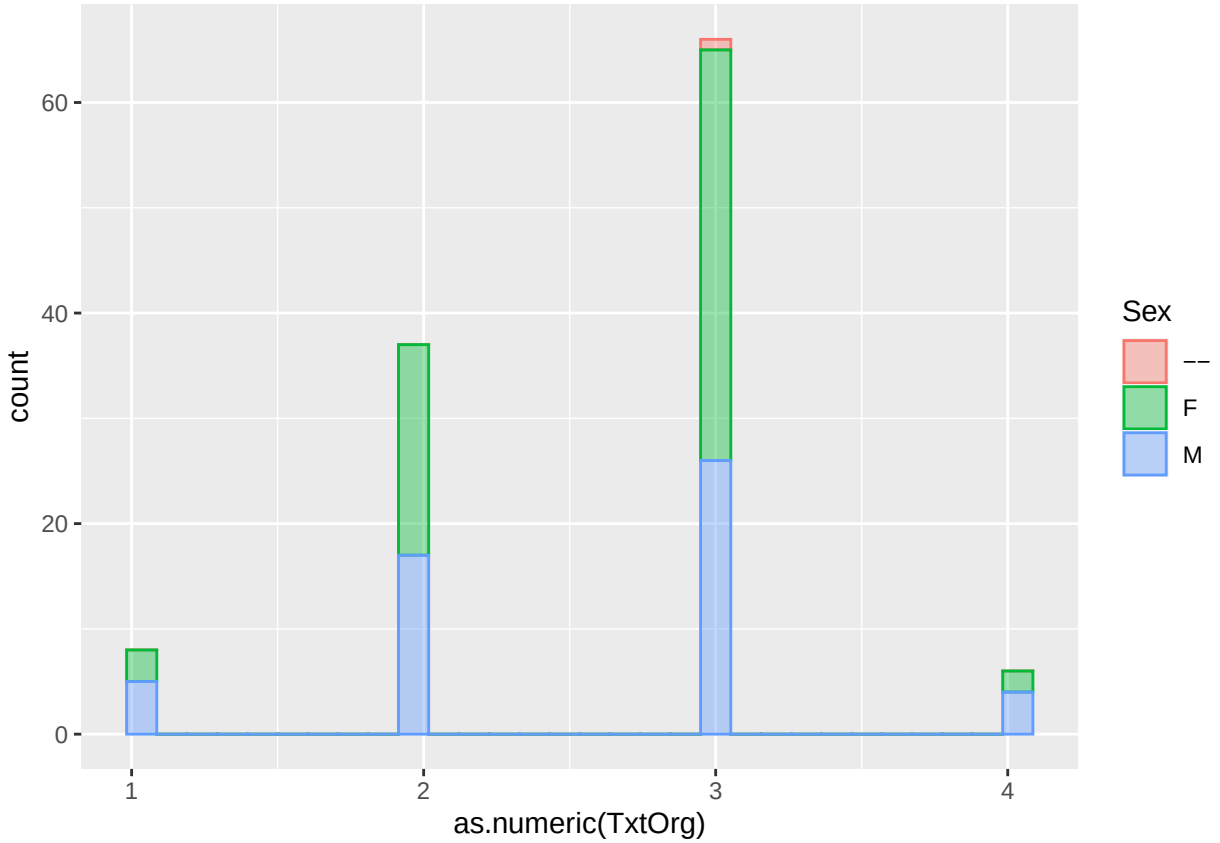



```
ggplot(ratings, aes(x = as.numeric(VisOrg))) + geom_histogram(aes(color = Sex, fill = Sex), bins = 30, a
```

```
## Warning: Removed 1 rows containing non-finite values (stat_bin).
```




```
ggplot(ratings, aes(x = as.numeric(TxtOrg))) + geom_histogram(aes(color = Sex, fill = Sex), bins = 30, a
```



Maybe we can also try to see if there is any tendency on the gender that if female or male tend to get distinguishable higher/lower ratings for these different rubrics and maybe later for different artifacts and other variables. From the plots above that roughly same amount of female and male can get 2/3 for all rubrics, but also from some rubrics, only male/female or mostly male/female get 4.0.