

Michael Kazarian, PODS Capstone, Dr. Pascal Wallisch, Will Calandra

When referring to columns, I use 0-based indexing

As can be seen in the code file, I seeded my rng according to my N-number:

```
random.seed(11447755)
```

Note: My chosen alpha level is  $0.005 = 5e-3$

## **Nan Removal**

Obviously nan removal is important, but there are several different ways one can go about it. I first try simple row-wise nan removal, but find it retains only about 13.5% of the observations, which most likely isn't ideal; see the code file for details. For this reason, I opt for column-wise removal. Now, some degree of row-wise nan removal is required for when I do multiple regression, but I could certainly retain more data for the significance tests, which involve fewer variables at a time compared with multiple regression. The one problem that column-wise removal could raise is that my conclusions from significance testing wouldn't all be equally strong. Instead, they'd vary in strength depending on the number of observations at use in the particular column/variable of analysis, i.e. depending on how much power I could bring to bear. However, this potential problem, one of power, can and will be remedied with bootstrapping. I should note that, as my handling of a given column (regarding nan removal and other matters) is naturally specific to the column and the question, I avoid discussing in detail my column-wise nan removal here, and will address it when it becomes relevant.

Separately, I have the problem, mentioned in the spec sheet, of accepting the number of average ratings from too few a number of ratings; an average value is less meaningful, as an average, if it's based on too few degrees of freedom. So I need to quantify "too few," and specifically I need to set a threshold number of ratings, in which I accept an observation only if it

equals or exceeds the threshold. I could “eyeball” this value, but this would result in an arbitrary threshold as I have no indication a priori of what a reasonable threshold would be, so I should go about it more systematically. Specifically, I consider a range of thresholds (1 through 50) and the corresponding percentage of remaining ratings, then plot this relationship:

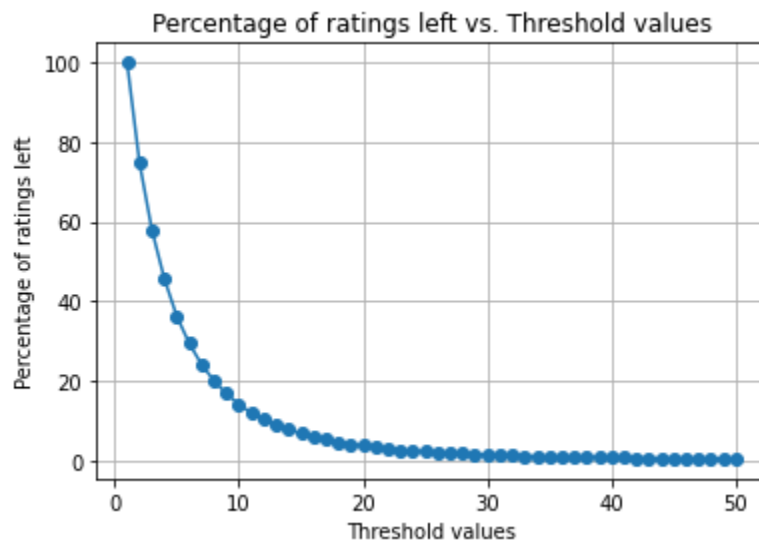
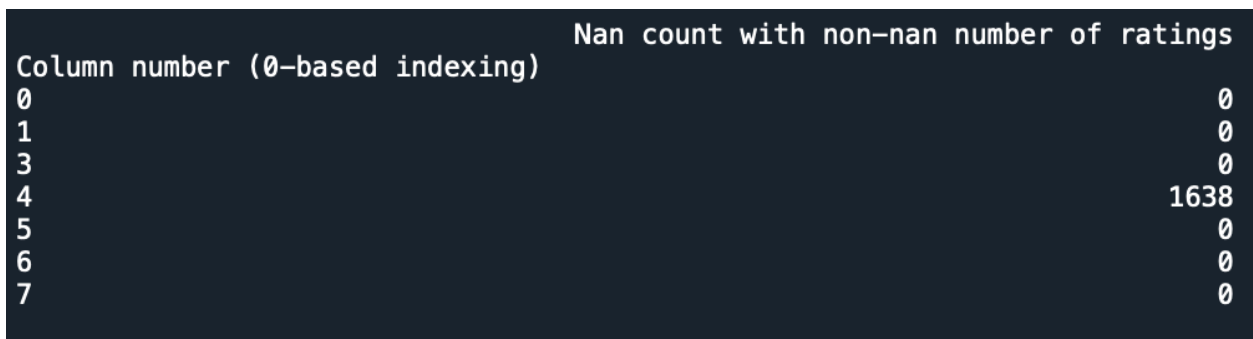


Fig 1

Based on this graph, I choose a threshold of 10 ratings. Visually, the values appear to begin declining much more slowly around here, which indicates that higher thresholds are increasingly unnecessary. At the same time, I wish for as much data as possible; hence I want a relatively low threshold. The threshold 10 fulfills both of these criteria. This visual inspection should suffice; further analysis involving, say, the drop-off rate of  $y$ , seems unnecessary. I decide against the alternative of a weighted average (weighing an average rating according to how many individual ratings make it up), as I believe that this approach implies an unfounded assumption about the data, namely that the accuracy of a rating scales linearly with its number of contributing ratings. I don't have good reason to believe this to be empirically true. Finally, I

should note that, in the graph, the percentage of ratings left can “tentatively” be interpreted as the number of observations left. This would be true if, in any column other than the number of ratings one, for every observation in which the number of ratings were non-nan, the corresponding value in that column were also non-nan. Fortunately, this is true: below see a DataFrame printed in the console which counts, in each column except number of ratings, the number of nans given a non-nan number of ratings:



Column number (0-based indexing)	Nan count with non-nan number of ratings
0	0
1	0
3	0
4	1638
5	0
6	0
7	0

Fig 2

(See the code file for details.) As the table implies, for every column except 4 (proportion of students who said they’d retake the class), I can interpret the proportion of ratings left as the proportion of observations left. I had used the word “tentatively” because, as will be seen, some questions (and not only those involving column 4) will require further reduction in the number of observations at use. However, I can use my graph just to gauge the number of observations left given a threshold of 10: 9841.

### Question 1:

Before proceeding, I must clean the data a bit more. Specifically, I must remove nans from the male and female columns each. From my earlier analysis, I know that there should be 0 nans left in either of these columns. Moreover, I must check for observations that are either both male and female or neither male nor female, because these couldn’t add any useful information.

Indeed, these observations inflate the effective number of degrees of freedom. I find there are 301 observations that are both male and female, and 2435 that are neither male nor female. I remove all of these; it's impossible to impute the latter information without making a biased assumption. I'm left with 3987 male ratings and 3118 female ratings; see the code file.

As indicated, I do a significance test, specifically a non-parametric one: I choose the U Test. With a non-parametric test, I don't need to additionally assume a normal distribution of ratings (for either group) (which would be a bad assumption; see below). With the U Test, I'm essentially comparing the distributions according to the median, which, for reasons discussed in class, is better suited to ratings; I assume this reasoning extends to averages of ratings, as in this scenario. Moreover, since both the male ratings and female ratings groups are decently large-sized, I don't believe it's necessary to correct for the disparity in group sizes; also, the U test is lenient on this. Finally, for clarity see below that the distribution of male and female ratings: (Note: though they look quite similar, they're each based on large samples, which can make significance hard to visually detect.)

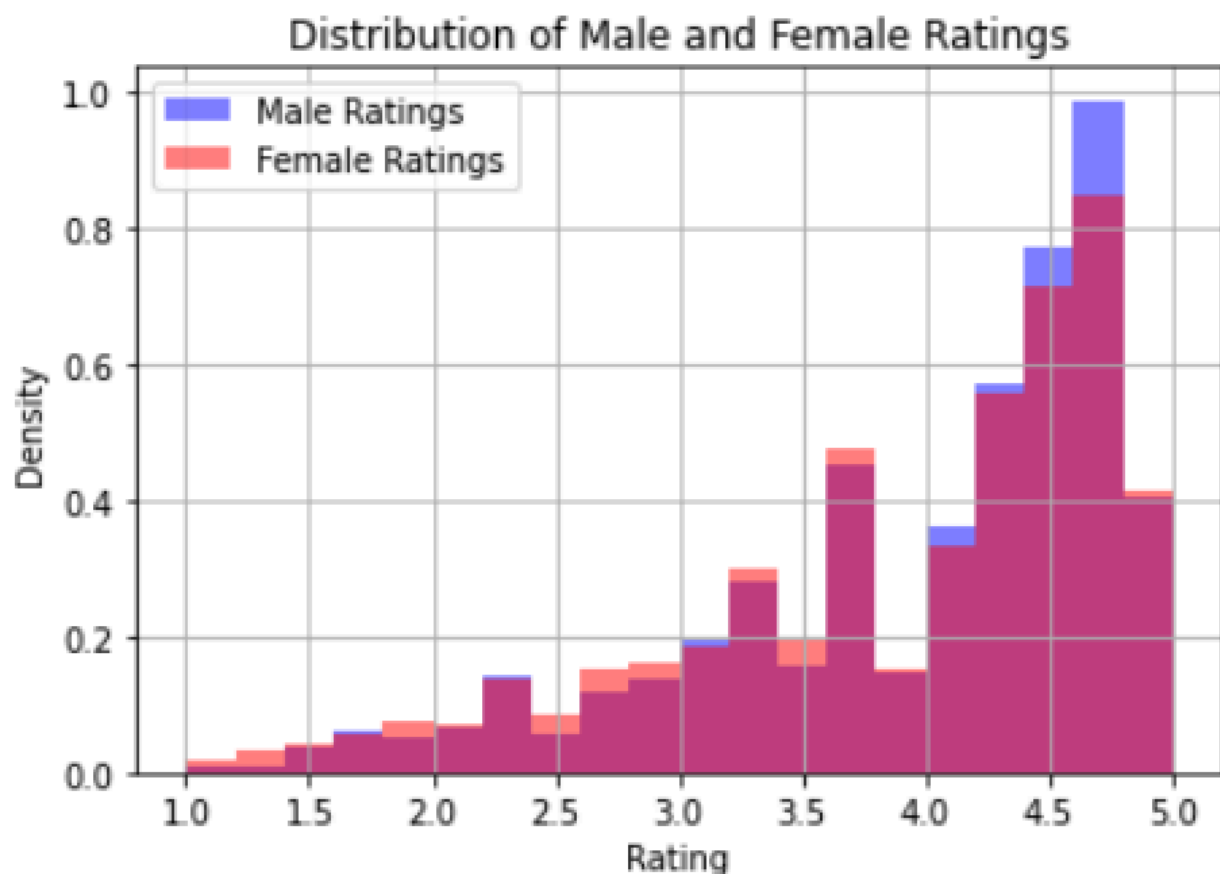


Fig 3 (above page)

Null hypothesis: if the median male rating is observed to be greater than the median female rating, this is purely due to sampling error in this data; this implies that in real life there is no pro-male rating bias. Doing the right one-sided test, I get a p-value of about  $1.4e-5$ ; this means that, assuming the null hypothesis is true, the probability of obtaining a result wherein the median male rating is greater than the median female rating by the observed amount or more is  $1.4e-5$ . This falls below the significance threshold of  $5e-3$ . Therefore I choose to reject the null hypothesis, and conclude this as evidence of a pro-male rating bias.

## **Question 2**

As indicated to answer this question I operationalize quality with average rating and number of ratings with experience. No need for further column-wise nan removal (based on my earlier analysis; see Fig 2). As indicated I do a significance test. I choose to divide the groups into “low” (5162) and “high” (4679) experience according to the median number of ratings (14). For the same reasons as in Question 1 I do the U Test, and similarly ignore the (even smaller) disparity in group sizes. Null hypothesis (I make a stronger claim): if the median average rating for the “low” experience group is observed to be less than the median average rating for the “high” experience group, this is purely due to sampling error in this data. Doing the right one-sided test, I get a p-value of about  $6.9e-6$ ; assuming the null hypothesis is true, the probability of obtaining a result wherein in the median average rating for the “low” experience group is less than the median average rating for the “high” experience group is by the observed amount or more is  $6.9e-6$ . This is less than the significance threshold of  $5e-3$ , so I choose to

reject the null hypothesis and conclude this as evidence that experience positively contributes to quality.

### Question 3

Again, no need for further column-wise nan removal (see Fig 2). I do another significance test, as I want to show evidence of an “effect” of difficulty on rating, not just a relationship. I choose to divide the groups into “low” (5309) and “high” (4532) difficulty, with the difficulty cutoff at the median of the average difficulty. For the same reasons, I do the U Test, and can safely disregard the group size disparity. Null hypothesis (again a stronger claim): if the median average rating for the “low” difficulty group is observed to be greater than the median average rating for the “high” difficulty group, this is purely due to sampling error in this data. Doing the right one-sided test, I get a p-value of about 0.0 (rounded by system); assuming the null hypothesis is true, the probability of obtaining a result wherein in the median average rating for the “low” difficulty group is greater than the median average rating for the “high” experience group is by the observed amount or more is about 0.0. In any case, the p-value is small enough (less than significance threshold of  $5e-3$ ) to safely reject the null hypothesis and conclude this as evidence that average difficulty negatively contributes to average rating. For more clarity, I compare the distributions of average rating according to low and high difficulty, and find them to be different in shape: this is fine (and expected) when using the U test in this case, especially as I did not choose not to reject the null hypothesis:

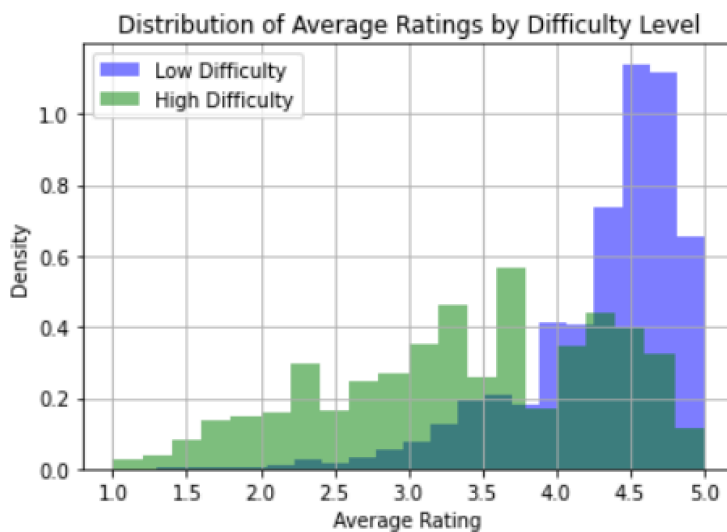


Fig 4 (above)

#### Question 4

Again, no need for further column-wise nan removal. As indicated I do a significance test, specifically the U Test, for the same reasons as earlier, but must cleverly split the data. First: null hypothesis: if the median average rating for the “low experience in online teaching” (to be defined) group is observed to be different from the median average rating for the “high experience in online teaching” (to be defined) group, this is purely due to sampling error in this data. It’ll help to first plot a histogram of num\_online\_ratings:

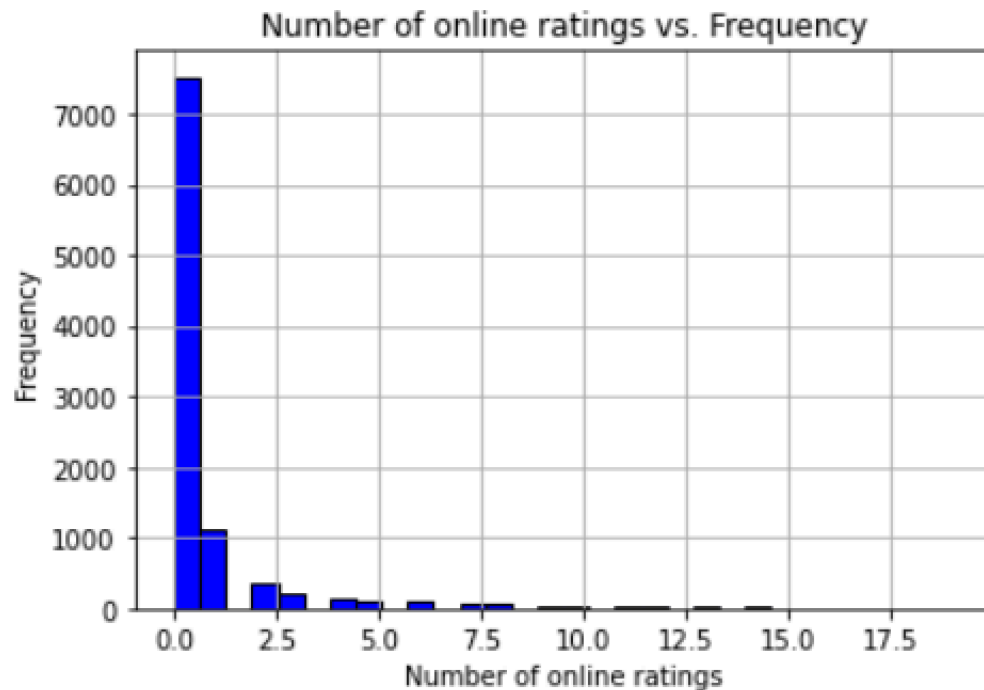


Fig 5

I can’t split it by the median as before, since the median number of online ratings (num\_online\_ratings) is 0, and I feel that number of online ratings > 0 doesn’t fully capture the “a lot” of teaching in the online modality that the question asks about. So I move up the cutoff from 50th to 95th percentile, which corresponds to num\_online\_ratings = 5 (what really matters is not the percentile but the cutoff value). Note that the max of num\_online\_ratings is 19 (not

pictured in graph), so a cutoff of 5 for “few online ratings” seems reasonable. However, this results in the groups of size 9447 (for “low experience in online teaching” or low\_exp\_online) and size 394 (for high experience in online teaching” or high\_exp\_online), so we must do bootstrapping for the smaller group to give it adequate power. Specifically, we re-sample with replacement 9447 times from high\_exp\_online (to get equally-sized groups, though this isn’t strictly necessary) over 1000 iterations, do a two-sided U Test for each iteration, and get the mean of the (1000) resulting p-values; this value is about .369: the probability of obtaining a result wherein if the median average rating for the “low experience in online teaching” group is different from the median average rating for the “high experience in online teaching” by the observed amount or more is .37.  $.37 > 5e-3$ , so I choose not to reject the null hypothesis. To have used the U Test well, I need to check that these distributions are not very different in shape; in fact they are quite similar:

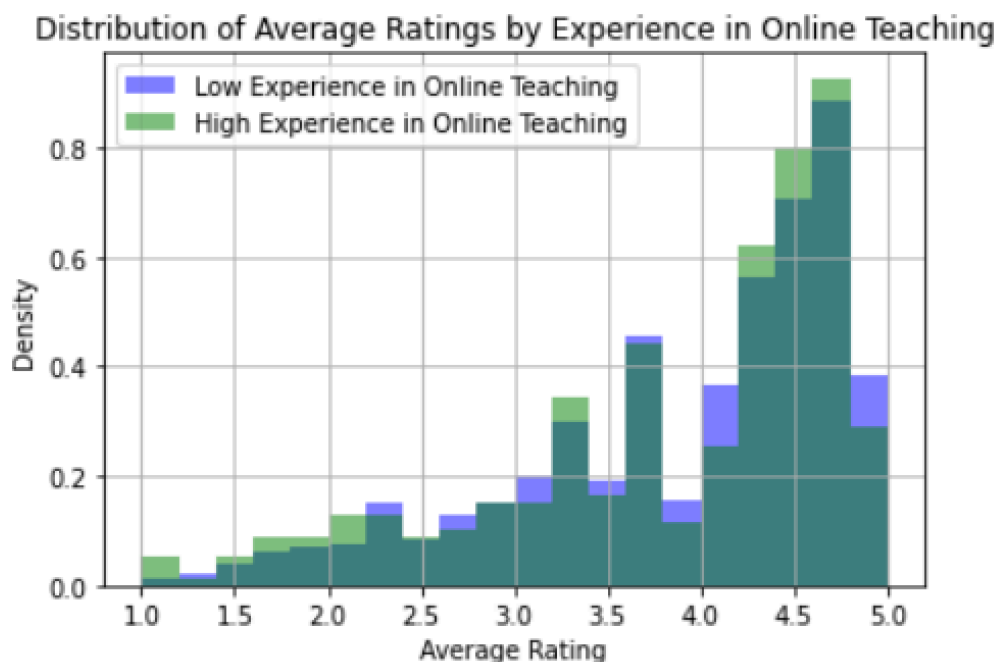


Fig 6

## Question 5



Based on Fig 2, I do now need to do some column-wise nan removal. As a result I'm left with  $9841 - 1638 = 8203$  observations for column 4 (proportion of those who would retake the class, or `prop_retake_class`) at use (refer back to Fig 2), which should be enough; imputation of values for column 4, based on, say, average ratings, might seem feasible, but I'd have to assume some empirical relationship between average rating (`avg_rating`) and proportion of retake to do this, and I believe I have enough power not to have to do this. Since this problem *only* asks about a general relationship between `prop_retake_class` and `avg_rating`, I first plot `prop_retake_class` vs. `avg_rating`:

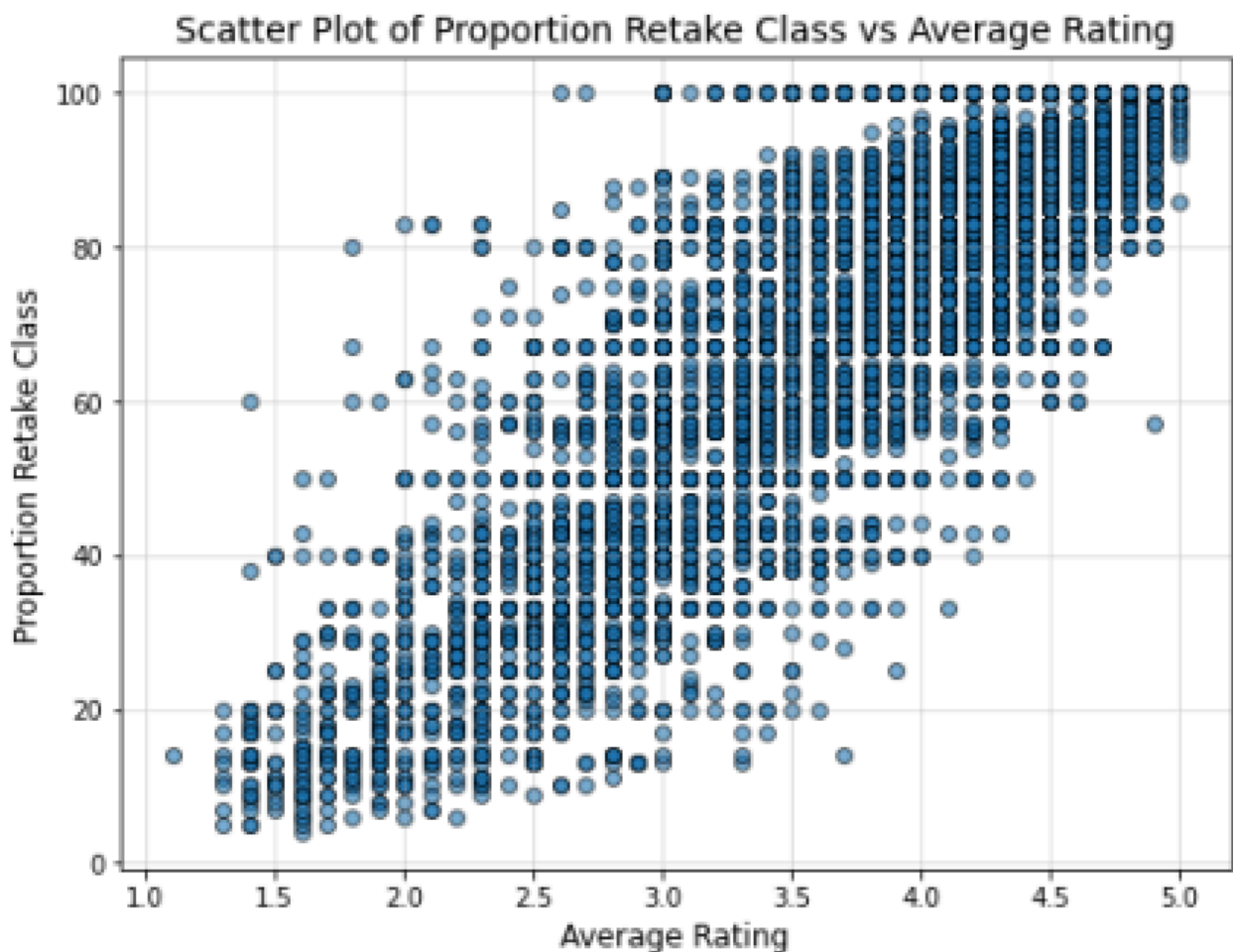
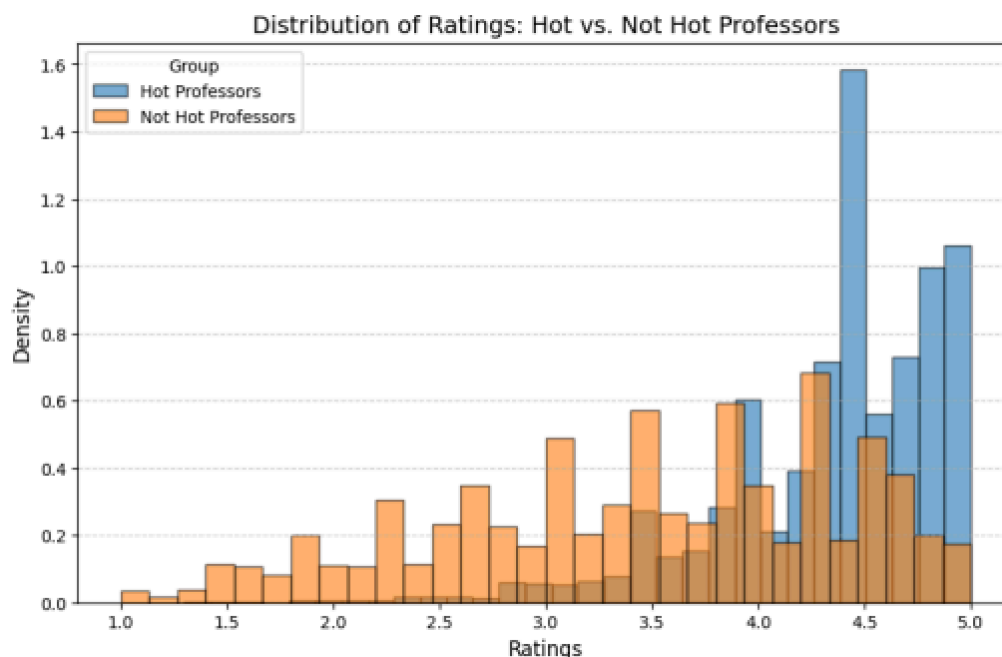


Fig 7

This relationship looks highly linear; in fact, we can use Pearson's  $r$  to measure the strength of the linear association between `prop_retake_class` and `avg_rating`, and we find it to be 0.876! This indicates a very strong, positive linear association between `prop_retake_class` and `avg_rating`, which is consistent with my intuition about this relationship (not that that matters).

### Question 6

Based on Fig 2, no need for further column-wise nan removal. Again, I use the U Test. Dividing into groups is particularly simple for this question; it's just based on whether an observation is 0 or 1 in column 3. (The difference in group sizes, again, can be safely ignored.) Null hypothesis: if the median average rating for the "hot" group is observed to be greater than the median average rating for the "not hot" group, this is purely due to sampling error in this data. Doing the right one-sided test, I get a p-value of about 0.0 (rounded by system); assuming the null hypothesis is true, the probability of obtaining a result wherein in the median average rating for the "hot" group is greater than the median average rating for the "not hot" group is by the observed amount or more is about 0.0. As  $0.0 < 5e-3$ , I reject the null hypothesis and conclude this as evidence that being rated as "hot" contributes positively to rating. As before, it's not very necessary, based on this result, to graph the distributions of these two groups, but I do it



for clarity: The distributions, expectedly, peak differently and even appear to be shaped differently:

Fig 8 (above)

### Question 7

To use linear regression to predict average rating from average difficulty, it'd first be necessary to do row-wise removal on avg\_rating and avg\_difficulty, however, based on Fig 2, they should have the same number of values, so I may proceed. I plot a scatter plot of ratings vs. difficulties (with a bit of transparency, as there are many points), along with the regression line, which I describe below:

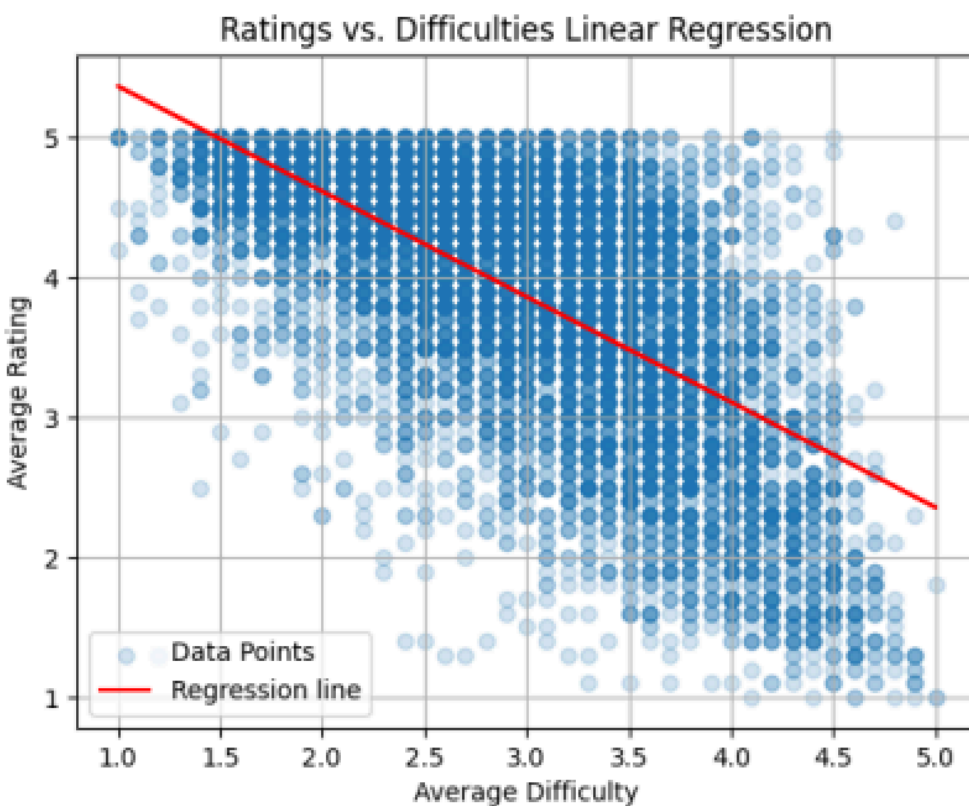


Fig 9

The slope of the regression line is about -0.75, the intercept about 6.1: according to this model, for every unit increase in average difficulty, average rating decreases by 0.75, and an

average difficulty of 0 (nonsensical as it's on a 1-5 scale) yields an average rating of 6.1 (also nonsensical for the same reason). The  $R^2$  for this model is about .42; 42% of the variation of average rating from its mean is accounted for by this model. The RMSE for this model is about 0.68; this is the average distance from the true (data) average rating.

### Question 8

The question asks to build a multiple regression model from “all available factors.” However there is one potential predictor I can safely discard from the outset based on my prior work—the number of online ratings—as this was determined not to have any significant effect on average rating. I'm left with avg\_difficulty, num\_ratings, hot, prop\_retake\_class, and male\_ratings. With these 5 predictors, a train/test split of 80/20 yields a test  $R^2$  of 0.81 and a test RMSE of 0.36 (whose meanings I explained above). However, I have concerns about multi-collinearity, and so must further narrow my selection of predictors. To this end I've included a inter-correlation heatmap for the contending predictors:

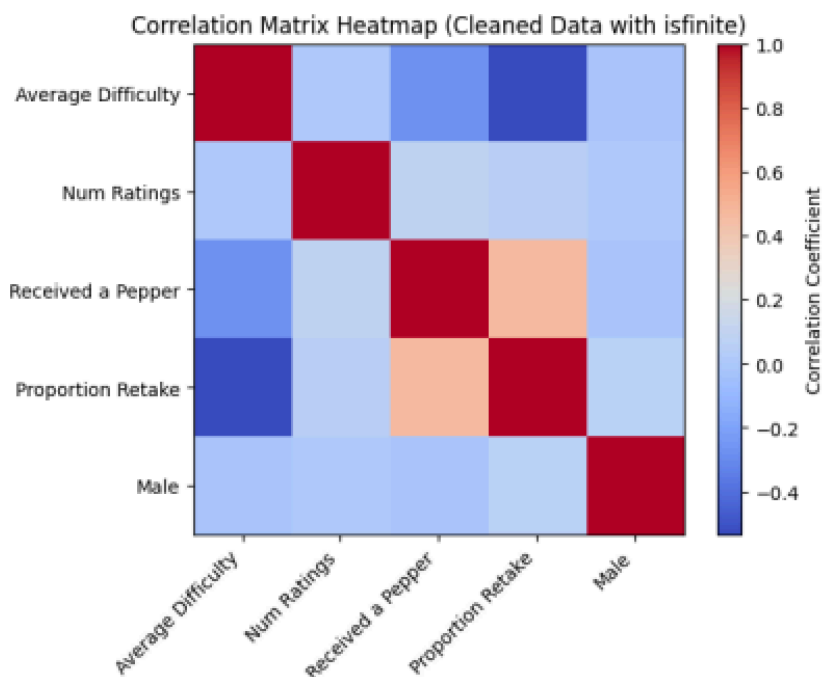
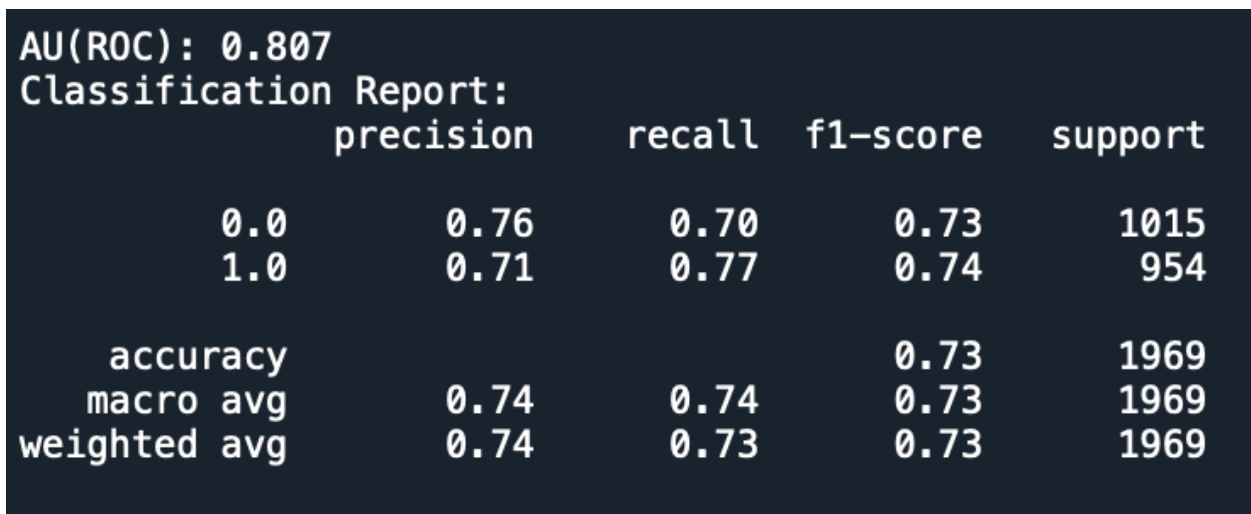


Fig 10

Based on this, I decide to remove avg\_difficulty and keep prop\_retake, as they're already moderately correlated, but I know prop\_retake to have a strong *linear* relationship with avg\_rating, which is what makes the difference. I also drop "hot" for similar collinearity concerns with prop\_retake. With this model, the test  $R^2$  is 0.77 and the RMSE is 0.4, which are of course worse in both cases than the 5-predictor model, but not too much worse; also, this model is likely to be more reliable than the 5-predictor one.

### Question 9

First we need to address class imbalances. However, we note that how we've selected for our data since the beginning, those who received a pepper (5042) and those who did not (4799) are close to being equally balanced, so we avoid major class imbalances. We use logistic regression as our classification model. Below are the AUROC score and the classification report:



```
AU(ROC): 0.807
Classification Report:
              precision    recall  f1-score   support

    0.0         0.76      0.70      0.73     1015
    1.0         0.71      0.77      0.74      954

 accuracy          0.73     1969
 macro avg         0.74      0.74      0.73     1969
 weighted avg      0.74      0.73      0.73     1969
```

Fig 11

An AUROC score of .807 means that the model does well in separating the classes, rightly ranking positive examples higher than negative ones 80.7% of the time.

### Question 10

Again we note we don't need to address class imbalances, as in 9, due to the selection of data for num\_ratings  $\geq 10$  resulting in mostly equal classes of peppers or not. Again we use logistic regression as our classification model, to get the following classification report.

Classification Report:				
	precision	recall	f1-score	support
0.0	0.76	0.68	0.72	817
1.0	0.71	0.79	0.75	824
accuracy			0.73	1641
macro avg	0.74	0.73	0.73	1641
weighted avg	0.74	0.73	0.73	1641
AUC: 0.8081335931836817				

Fig 12

We also include the ROC curve for Logistic Regression, this time:

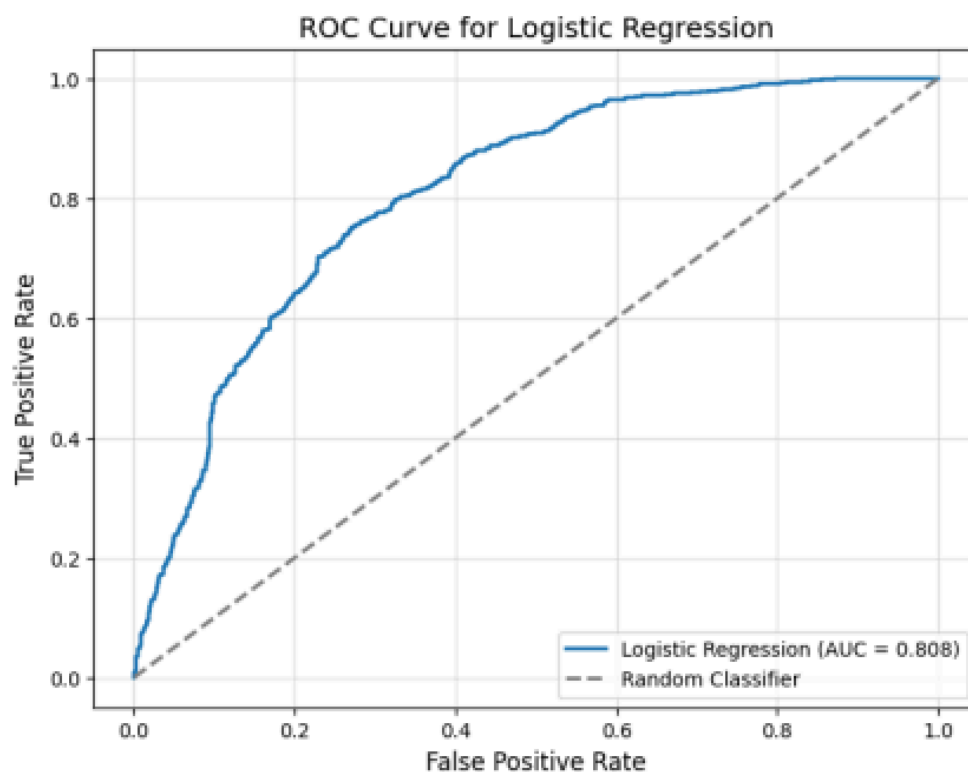


Fig 13

In comparison to the one-predictor model from question 9, this model leverages all available data. However it performs very similarly across all metrics, based on the classification report, and the AUC is almost identical. So while this model has more data to bear, it doesn't perform that much better than the one-predictor model, though its AUC indicates it is still strong.