Xtern Work Assessment

Xtern Applicant

10/15/2020

#Initial Analysis, Questions, and Assumptions

Without looking at the data too in depth, I have a few questions regarding the data itself. The Minimum\_Order attribute does not clearly display meaningful information. It’s title would suggest that this is the least amount of money a person could spend at this restaurant possible, however the values, just from skimming through the data, are usually significantly greater than the Average\_Cost for the same restaurant. If these were the cheapest possible orders a customer could make, then the average cost of an order would be significantly higher. I have ignored this attribute in my analysis due to a lack of information on how I can properly utilize this attribute of data.

Additionally, I am making some assumptions about the data so I can more efficiently utilize it. I am assuming that Votes represents how many people voted for that restaurant to be their favorite. I am also assuming that Reviews represents how many people have left reviews rating a given restaurant on a scale of 1 to 5, and that Rating represents the average score given by these reviews. Finally, I am assuming that Average\_Cost represents the average cost of a meal for 1 person at a given restaurant, and Cook\_Time is the time it takes the food to come out.

library(ggplot2)  
library(dplyr)

##   
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':  
##   
## filter, lag

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

library(tidyverse)

## -- Attaching packages ----------------------------------------------------------------------- tidyverse 1.3.0 --

## v tibble 3.0.3 v purrr 0.3.4  
## v tidyr 1.1.2 v stringr 1.4.0  
## v readr 1.3.1 v forcats 0.5.0

## -- Conflicts -------------------------------------------------------------------------- tidyverse\_conflicts() --  
## x dplyr::filter() masks stats::filter()  
## x dplyr::lag() masks stats::lag()

FoodieX = read.csv("2020-XTern-DS.csv")  
BaseRestaurants = FoodieX[which(FoodieX$Rating!="NEW" & FoodieX$Rating!="-" & FoodieX$Reviews>=1),]

#General Methodology

I have created a subset of the data called BaseRestaurants, which I will be conducting my analysis on. This set represents all restaurants who have received at least 1 review and are not considered new. By analyzing this subset, we can see how other restaurants in the area may fare dependent on given features.

#Conclusion 1: Cook\_Time and Rating

My first idea was to determine if the time it took to prepare the food had any effect on the rating a restaurant typically received. I hypothesized that there would be an inverse relationship between the time it took to create the food and the rating a restaurant received.

lmRating = as.numeric(BaseRestaurants$Rating)  
lmTime = as.numeric(gsub(" minutes", "", BaseRestaurants$Cook\_Time))  
summary(lmRating)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 2.400 3.300 3.600 3.622 3.900 4.800

summary(lmTime)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 20.00 30.00 30.00 38.36 45.00 120.00

cor.test(lmRating, y=lmTime)

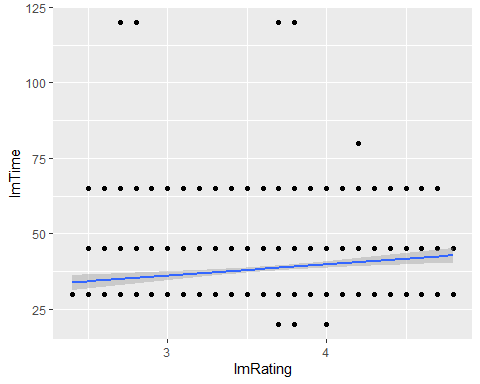
##   
## Pearson's product-moment correlation  
##   
## data: lmRating and lmTime  
## t = 5.0432, df = 1602, p-value = 5.099e-07  
## alternative hypothesis: true correlation is not equal to 0  
## 95 percent confidence interval:  
## 0.07653608 0.17289917  
## sample estimates:  
## cor   
## 0.1250124

linearModel = lm(lmTime ~ lmRating, data=BaseRestaurants)  
summary(linearModel)

##   
## Call:  
## lm(formula = lmTime ~ lmRating, data = BaseRestaurants)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -19.772 -8.279 -6.786 5.974 85.080   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 24.8418 2.6984 9.206 < 2e-16 \*\*\*  
## lmRating 3.7327 0.7401 5.043 5.1e-07 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 12.42 on 1602 degrees of freedom  
## Multiple R-squared: 0.01563, Adjusted R-squared: 0.01501   
## F-statistic: 25.43 on 1 and 1602 DF, p-value: 5.099e-07

ratingTimePlot = ggplot(BaseRestaurants, aes(x=lmRating, y=lmTime)) + geom\_point() + geom\_smooth(method="lm", se=TRUE, level=0.99)  
ratingTimePlot

## `geom\_smooth()` using formula 'y ~ x'

 I created a scatterplot to visualize the data along with a line of best fit drawn using linear regression, and found the opposite to be true. There was a positive, however very weak correlation. This had me give more thought to the nature of the data, since the time to prepare food and the rating of a restaurant seemed to have a minimal effect on each other.

This led me reconsider what the Cook\_Time attribute represented. My initial assumption of Cook\_Time was, as seen above, that it represented how long it took for the food to be served. In reality, it represented a more literal explanation of the title, exactly how long it took to prepare and serve the food. I had not factored in prep time or any steps that may occur before the food is even ordered. While the results of this analysis disprove my initial hypothesis, they do match the findings of this new assumption. It also makes logical sense, better food would require a longer time to make, and while the correlation isn’t very strong, alongside the linear regression it suggests that restaurants that take longer to prepare their food receive higher ratings.

In conclusion, my initial hypothesis that faster food is rated better was disproved due to new information regarding the data, and in reality food that takes longer to be prepared tends to receive better ratings.

#Conclusion 2: Average\_Cost and Rating

After comparing Rating to Cook\_Time, I decided to compare Rating to Average\_Cost. I was unsure what to expect from this. While quality comes with a price as well, I could see expensive food driving off business. In the end, I hypothesized that price would have a positive correlation with ratings.

signlessCost = gsub("[$ ]", "", BaseRestaurants$Average\_Cost)  
lmCost = as.numeric(signlessCost)

## Warning: NAs introduced by coercion

summary(lmCost)

## Min. 1st Qu. Median Mean 3rd Qu. Max. NA's   
## 5.00 15.00 20.00 21.35 25.00 150.00 2

cor.test(lmRating, y=lmCost, na.rm = TRUE)

##   
## Pearson's product-moment correlation  
##   
## data: lmRating and lmCost  
## t = 14.939, df = 1600, p-value < 2.2e-16  
## alternative hypothesis: true correlation is not equal to 0  
## 95 percent confidence interval:  
## 0.3061476 0.3921328  
## sample estimates:  
## cor   
## 0.3498769

linearModel = lm(lmCost ~ lmRating, data=BaseRestaurants)  
summary(linearModel)

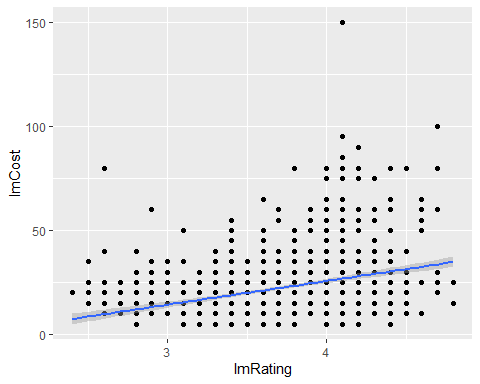
##   
## Call:  
## lm(formula = lmCost ~ lmRating, data = BaseRestaurants)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -25.207 -7.695 -2.245 3.442 123.205   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -19.8406 2.7755 -7.149 1.33e-12 \*\*\*  
## lmRating 11.3745 0.7614 14.939 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 12.77 on 1600 degrees of freedom  
## (2 observations deleted due to missingness)  
## Multiple R-squared: 0.1224, Adjusted R-squared: 0.1219   
## F-statistic: 223.2 on 1 and 1600 DF, p-value: < 2.2e-16

ratingCostPlot = ggplot(BaseRestaurants, aes(x=lmRating, y=lmCost)) + geom\_point() + geom\_smooth(method="lm", se=TRUE, level=0.99)  
ratingCostPlot

## `geom\_smooth()` using formula 'y ~ x'

## Warning: Removed 2 rows containing non-finite values (stat\_smooth).

## Warning: Removed 2 rows containing missing values (geom\_point).

 I used the same methodology as before to compare these two attributes. There was a weak positive correlation between the 2, as depicted by the linear regression on the scatterplot. This correlates directly with my initial hypothesis that the attributes would have a positive correlation, even if the correlation is weak.

In conclusion, increased average costs for eating at a restaurant mean increased ratings. This is likely caused by the food being more expensive due to it being of a higher quality, or using better ingredients that cost more.

#Conclusion 3: Location and Rating

I wanted to analyze if locations in the town had better or worse restaurants due to their location. I do not have any frame of reference for how this map will look, so I hypothesize the map will have no correlation between Rating and either Longitude or Latitude.

lmLong = as.numeric(BaseRestaurants$Longitude)  
lmLat = as.numeric(BaseRestaurants$Latitude)  
  
summary(lmLong)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## -86.00 -85.74 -85.50 -85.50 -85.24 -85.00

summary(lmLat)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 39.00 39.26 39.50 39.50 39.75 40.00

cor.test(lmRating, y=lmLong)

##   
## Pearson's product-moment correlation  
##   
## data: lmRating and lmLong  
## t = -0.030659, df = 1602, p-value = 0.9755  
## alternative hypothesis: true correlation is not equal to 0  
## 95 percent confidence interval:  
## -0.04970878 0.04818048  
## sample estimates:  
## cor   
## -0.0007659848

cor.test(lmRating, y=lmLat)

##   
## Pearson's product-moment correlation  
##   
## data: lmRating and lmLat  
## t = -1.2514, df = 1602, p-value = 0.211  
## alternative hypothesis: true correlation is not equal to 0  
## 95 percent confidence interval:  
## -0.08007161 0.01772234  
## sample estimates:  
## cor   
## -0.03124942

linearModelLong = lm(lmLong ~ lmRating, data=BaseRestaurants)  
linearModelLat = lm(lmLat ~ lmRating, data=BaseRestaurants)  
summary(linearModelLong)

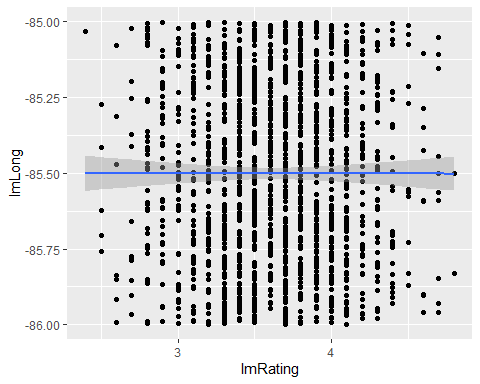
##   
## Call:  
## lm(formula = lmLong ~ lmRating, data = BaseRestaurants)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.49698 -0.24322 -0.00247 0.26011 0.50011   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -8.550e+01 6.333e-02 -1349.965 <2e-16 \*\*\*  
## lmRating -5.326e-04 1.737e-02 -0.031 0.976   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.2915 on 1602 degrees of freedom  
## Multiple R-squared: 5.867e-07, Adjusted R-squared: -0.0006236   
## F-statistic: 0.0009399 on 1 and 1602 DF, p-value: 0.9755

summary(linearModelLat)

##   
## Call:  
## lm(formula = lmLat ~ lmRating, data = BaseRestaurants)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.51366 -0.24193 0.00059 0.24901 0.50545   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 39.57904 0.06230 635.277 <2e-16 \*\*\*  
## lmRating -0.02138 0.01709 -1.251 0.211   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.2868 on 1602 degrees of freedom  
## Multiple R-squared: 0.0009765, Adjusted R-squared: 0.0003529   
## F-statistic: 1.566 on 1 and 1602 DF, p-value: 0.211

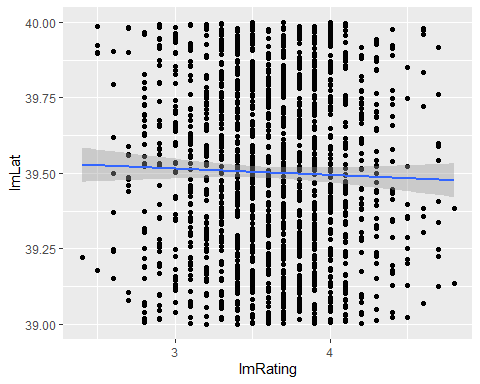
ratingLongPlot = ggplot(BaseRestaurants, aes(x=lmRating, y=lmLong)) + geom\_point() + geom\_smooth(method="lm", se=TRUE, level=0.99)  
ratingLongPlot

## `geom\_smooth()` using formula 'y ~ x'



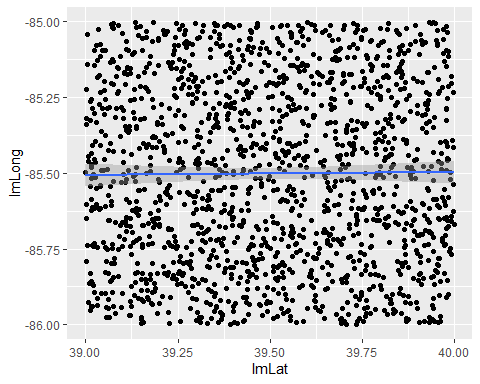
ratingLatPlot = ggplot(BaseRestaurants, aes(x=lmRating, y=lmLat)) + geom\_point() + geom\_smooth(method="lm", se=TRUE, level=0.99)  
ratingLatPlot

## `geom\_smooth()` using formula 'y ~ x'



ratingMapPlot = ggplot(BaseRestaurants, aes(x=lmLat, y=lmLong)) + geom\_point() + geom\_smooth(method="lm", se=TRUE, level=0.99)  
ratingMapPlot

## `geom\_smooth()` using formula 'y ~ x'



I used the same processes as before, and the analysis gave me the results I was expecting. For Longitude, there was no significant correlation with rating. For Latitude, there was some correlation with rating, but not nearly enough to be considered significant either. When comparing Latitude and Longitude, there was no visible correlation seen through the linear regression displayed either. I’m guessing the road system must look very peculiar, however, to break the scenario for a moment, I am assuming the data was randomly generated for this work assessment. The real coordinates for this reigon stretch far beyond a simple town and take up a small chunk of SouthEastern Indiana.

In conclusion, the restaurant’s location has no bearing on how successful it is.

#Conclusion 4: Votes and Rating

I wanted to see if there was any correlation between the number of votes a restaurant received, assuming that each vote represents 1 person picking a restaurant as their favorite, in comparison to their rating, which depicts a score of 1 to 5. I predict a strong positive correlation, since a restaurant with a higher rating would presumably be more people’s favorite restaurant to vote for.

lmVote = as.numeric(BaseRestaurants$Votes)  
summary(lmVote)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 4.0 20.0 68.5 260.2 253.0 9054.0

cor.test(lmRating, y=lmVote, na.rm = TRUE)

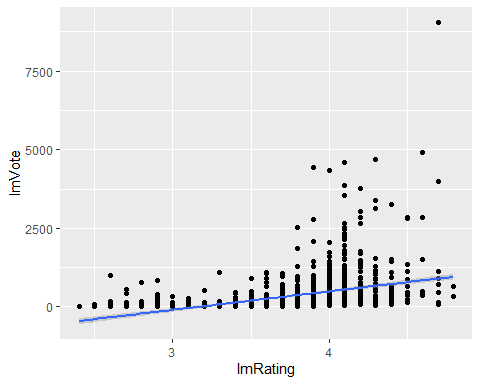
##   
## Pearson's product-moment correlation  
##   
## data: lmRating and lmVote  
## t = 19.321, df = 1602, p-value < 2.2e-16  
## alternative hypothesis: true correlation is not equal to 0  
## 95 percent confidence interval:  
## 0.3941563 0.4735828  
## sample estimates:  
## cor   
## 0.4347145

linearModel = lm(lmVote ~ lmRating, data=BaseRestaurants)  
summary(linearModel)

##   
## Call:  
## lm(formula = lmVote ~ lmRating, data = BaseRestaurants)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -812.9 -220.7 -103.6 64.0 8160.1   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -1868.08 110.89 -16.85 <2e-16 \*\*\*  
## lmRating 587.65 30.42 19.32 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 510.4 on 1602 degrees of freedom  
## Multiple R-squared: 0.189, Adjusted R-squared: 0.1885   
## F-statistic: 373.3 on 1 and 1602 DF, p-value: < 2.2e-16

ratingVotesPlot = ggplot(BaseRestaurants, aes(x=lmRating, y=lmVote)) + geom\_point() + geom\_smooth(method="lm", se=TRUE, level=0.99)  
ratingVotesPlot

## `geom\_smooth()` using formula 'y ~ x'



I used the same method I used for the other models and received similar results to my hypothesis. As seen on the linear regression and scatterplot, the ratings of a given restaurant share a moderately strong positive correlation with the number of votes they received.

In conclusion, the number of votes a restaurant has directly correlates with their rating.

#Final Analysis

Overall, all of the attributes analyzed had some level of correlation with the Rating attribute except for Longitude and Latitude. This could be due to those attributes being common business choices, such as raising prices and production time to accommodate for higher quality food, or it could be due to customer preference, as shown with ratings and votes correlating.

#Future Work

Given more time, I would like to have dug into the Cuisines attribute and determine if Cuisine had any bearing on Ratings, Votes, or Cook\_Time. I also would have liked to more work with the new restaurants, and see if I could produce a model that would have predicted their ratings based on their Average\_Cost and Cook\_Time.