Proposal
Bordeaux Wine Taste Review in Machine Learning

By: Ethan Gruis, Shiv Patel, and Benjamin Siglow

"Before Rudy Kurniawan was 30, he had risen to become a kingpin purveyor of wines that were the rarest of the rare. Then, one Friday evening in 2008, he made a mistake that shattered his career."(Hollman, 2017). This snippet from a 2017 New York Post article is of a historic swindler who sold faux bottles of wine to the wealthy. The truth is wine is a fickle thing with arguments that price is not enough to gauge just how good wine is. With our Machine Learning team of Shiv Patel, Ethan Gruis, and Benjamin Siglow recently entering into the world of wine we wanted to review how well we could predict the rating given to wine based on a data set containing rating, name, price, and extensive taste notes. With this machine learning campaign, we plan to make it so Bordeaux wine producers would be able to predict current and future bottles of wine's rating from price and taste notes. A resulting effective model would allow for production of higher scoring wines, but also accurate production numbers of inventory to accommodate the expected interest of buyers on high scoring wines. In a perfect world depending on how many required attributes it takes to accurately rate a wine we would hope to take the winter break and build a usable wine rate dashboard. In this you could put some specific notes described from the wine as well as the cost and year to best purchase your wine for both aged and aspiring afficionados.

Data Overview

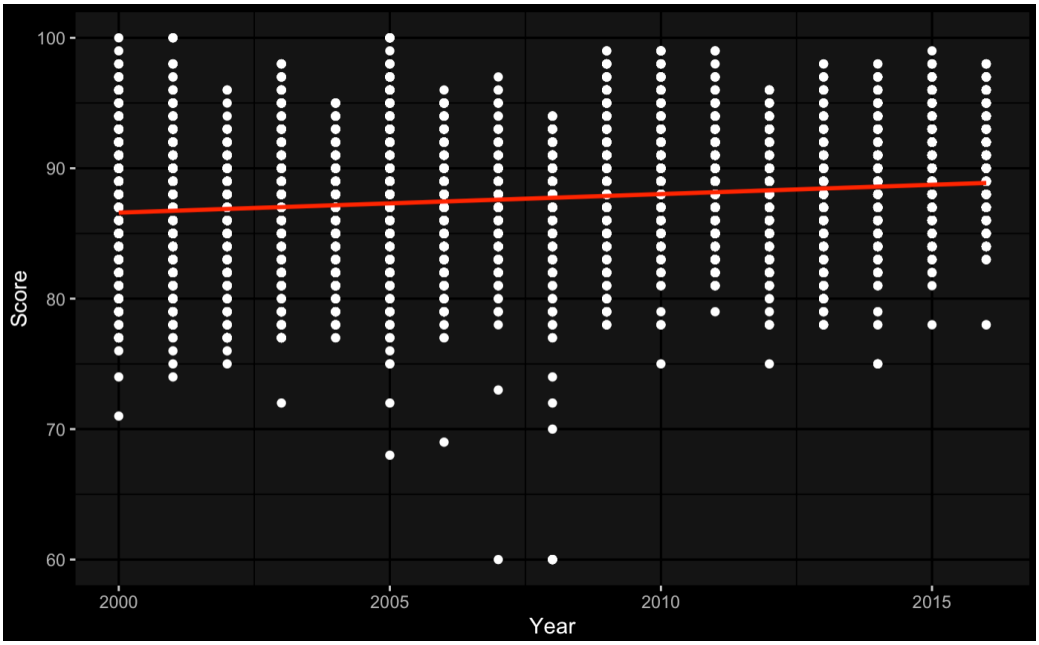
The data set we intend to use is a Bordeaux wine data set which was collected from Kaggle. A Bordeaux wine is any wine produced in the Bordeaux region of southwest France. The dataset overall has 14,349 rows each pertaining to a review of an individual wine from a particular year. The primary variables are the wine name, price, year, score, and the taste notes each having their own column. These taste notes are factors with 2 levels, being 1 and 0. With the value being 1, that wine has that taste note and 0 not having it.

Overall, there are 983 binary variables (taste notes) and these needed to be converted from an integer into a factor. We used the function "lapply" to modify the columns for tasting notes into factors. By changing these variables to factors, multiple columns became factors with one level, i.e. the entire set of wines had or did not have a taste note. Due to this these columns would no longer help in determining scores of wines, because there is no difference among this taste note. With this we removed these columns to decrease data size and allow for data processing. From there we continued to clean our data by removing "$" from the Price column and then converted it from a character into a numeric value which introduced NA values by coercion for our missing prices.

With the names in our dataset, it is not an encoding issue rather than a way the data has been exported from Kaggle. Various encoding methods to account for the special characters in the wine names were not successful. Along with the names, there are some values that we would have liked to have for our exploration. These variables would be acidity or pH, location of the vineyard and other chemical characteristics that contribute to a wine's score.

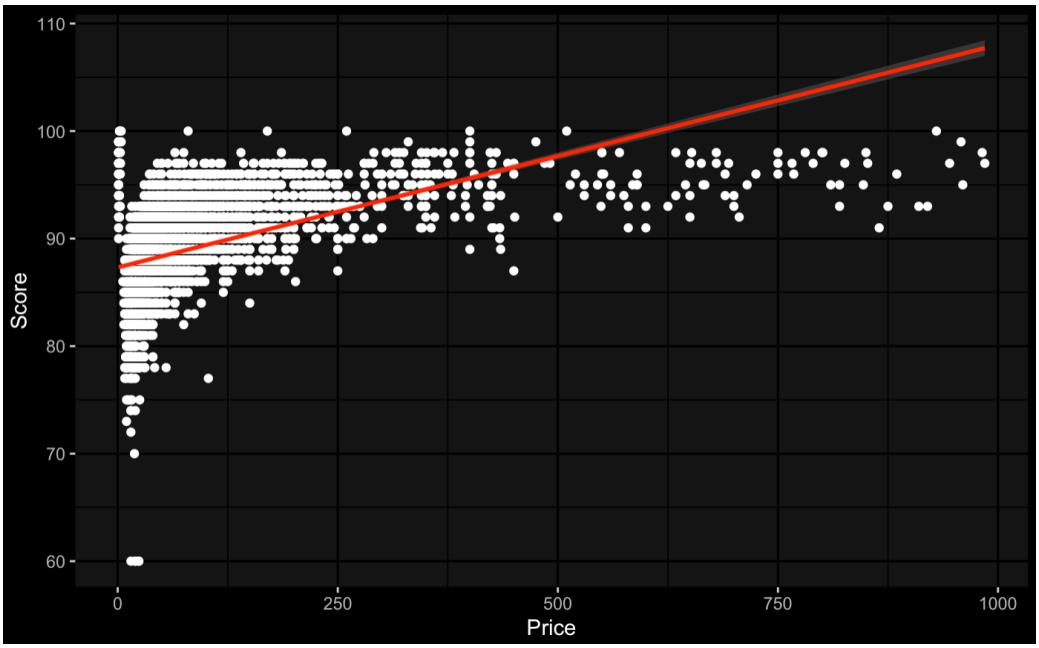
Visualization

Score vs. Time



Seeing as we are planning to predict the ratings of future Bordeaux wines we wanted to see if overall the review numbers are increasing just based on the year of wine. If this was a major increase there may be concerns of the effect of the reviewers or the overall product Bordeaux is providing. Our visualization did conclude there is a slight increase over the last 16 years, however not a large change and thus something we are not overly concerned about.

Score vs. Price



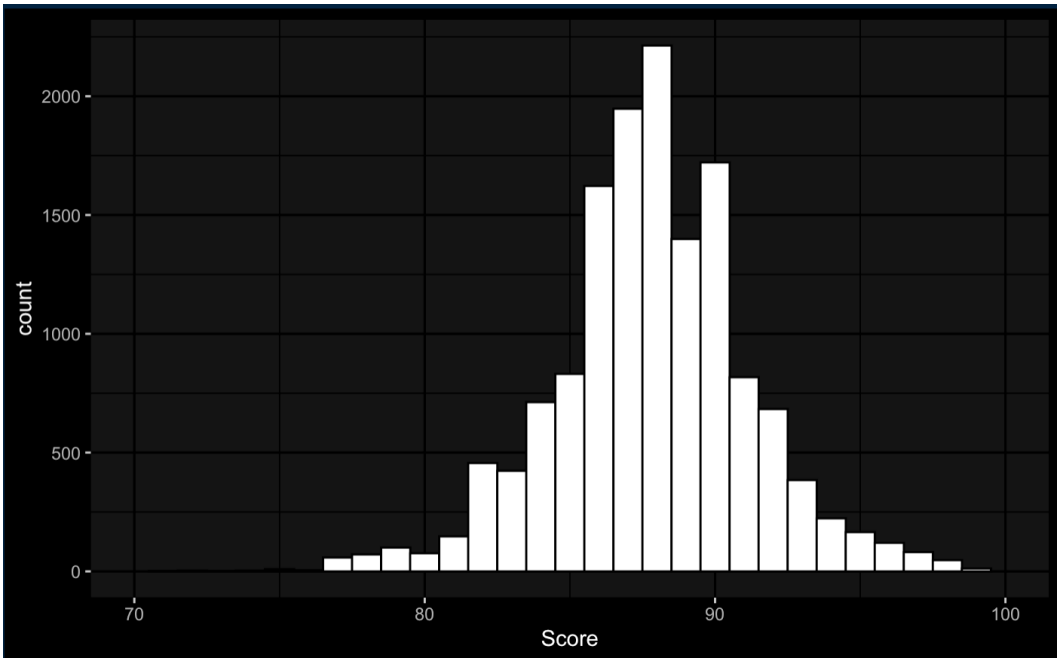
The premise of this paper was to determine the complicated rating of wine and price is not a great determiner. To get an idea of distribution of scores as well as to make sure price is not an immense predictor of the score, we graphed the price vs. score.

Linear Regression

```
Call:
lm(formula = Score ~ Price, data = wine_data)

Residuals:
     Min       1Q   Median       3Q      Max
-27.8140  -1.6276   0.1031   1.9582  12.6416

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 8.732e+01  3.638e-02 2399.93   <2e-16 ***
Price       2.071e-02  3.951e-04   52.42   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 3.015 on 9684 degrees of freedom
  (4663 observations deleted due to missingness)
Multiple R-squared:  0.2211,    Adjusted R-squared:  0.221
F-statistic:  2748 on 1 and 9684 DF,  p-value: < 2.2e-16
```

To accompany the linear graph we worked a linear regression to confirm price is not too powerful in the prediction of the overall scores. With Price as our independent variable and Score as the dependent, we are only explaining 22% of the variability. We are comfortable with this not being too large of a determiner.

Histogram



Lastly, we wanted to confirm we had a reasonable distribution of score across our wines. This is not a perfect distribution; however, we are comfortable with this layout of the scores and our ability to use this data for prediction.

Previous Work

After looking for some other projects utilizing wine data and machine learning, we found a few different examples of people using wine data to analyze wine review scores - however none immediately popped up with the same data that we're using with the many binarized note variables. We did find a few Towards Data Science articles from people who ran different analyses on different wine datasets, and while we found their findings interesting, we are certainly taking a different approach and direction with our data and problem framing than just merely looking at trends in wine data.

The first article we found (Goutay, 2018), looked at predicting wine quality based off of a string description of the wine, and the author used a Random Forest Classifier and string vectorization to achieve a 97% accuracy score, which is very impressive although probably not useful in real world application, since naturally people who create longer descriptions of a wine will enjoy wine more than a wine they give a short description of.

The second article we found had the author (Shin, 2020) predicting wine quality based on measurable chemical properties of the wine; including acidity, total sugars, pH balance, and chlorides amongst other properties. This project seemed less concerned with solving a problem in the wine industry and more concerned with broadly understanding which model would be most successful given the dataset. The author used different models; including a Decision Tree, AdaBoost, XGBoost,

Proposed Process

Upon review with Dr. Barron we plan to incorporate decision tree forests as well as effective portions of XGBoost such as cross validations. Seeing as most of are attributes are specific to binary (factors in r) linear regression seems extremely unrealistic for the purpose of better understanding of what has created the scores. In addition, after reviewing our integer value of cost and determining it is only accounting for a minor amount of our variance leaning into the more complex set of 984 factor attributes appears to be the best method. In addition to the effectiveness of trees on factorial variables it brings in two other major strengths. For one the only missing values we are currently seeing is that of the price. With the usage of a tree or trees model we can feel comfortable this will not extensively affect the overall tree. In addition, the hierarchy formed through a tree could be extremely helpful in future decision making. For example, if we believe Bordeaux is the wine we are trying to purchase, and Blood Orange notes are ranked high in the tree splits it could be the wine we choose.

With the great value of the tree accommodating our specific data set there are some specific areas we will need to determine. We need to use methods from class to allow our train data to have an even distribution of low and high scores, because there are a massive number of low scores in comparison to the high scores. In addition, it is extremely pertinent that we are careful in our pruning of the tree. We are still looking at 988 overall variables and we know multiple changes could extremely change our tree so actions such as bootstrapping, and sampling will be a major role in this project.

Overall using either a tree or random forests we believe we will be effective in producing a predictive model for Bordeaux. If this is accomplished based on the notes of flavor parameters Bordeaux should have the ability to predict the rating of their up and coming wines to best support the amount of production and distribution. We will be checking accuracy among test samples and tracking our changes along this process to track our success and confirm the realism of our methods.

Contribution

Overall, the project has been tackled equally from each team member. We played to each other's strengths and weaknesses. Ethan for example was focused more on the code, still contributing to the report, while Ben and Shiv, though they contributed to the code as well, were more focused on the analysis and report.

Bibliography

Goutay, Olivier. "Wine Ratings Prediction Using Machine Learning." *Medium*, Towards Data Science, 15 June 2018, towardsdatascience.com/wine-ratings-prediction-using-machine-learning-ce259832b321.

Hellman, Peter. "How an Illegal Immigrant Pulled off the Greatest Wine Scam in US History." *New York Post*, New York Post, 7 Oct. 2017, nypost.com/2017/10/07/how-an-illegal-immigrant-pulled-off-the-greatest-wine-scam-in-us-history/.

Shin, Terence. "Predicting Wine Quality with Several Classification Techniques." *Medium*, Towards Data Science, 8 May 2020, towardsdatascience.com/predicting-wine-quality-with-several-classification-techniques-179038ea6434.