Student: Olga Vinogradova
Introduction to Data Science. Fall 2019
Instructor: Daniel D. Gutierrez

Final Project.

Wine Collection Data.

The source of this project is the data provided by a private wine collector, a user of Cellartracker.com which is a wine inventory management software. The data was downloaded in csv format and uploaded into the R Studio Project as an Excel file.

The business consideration is to identify value correlations tied to scores from past purchases that can be used to impact future purchases. As older bottles are removed from a cellar, newer ones must be bought to replenish them, thus maintaining a certain number of aged wines being available to consume every year. The goal is to identify wines that have value (higher score for a lower price and are cheaper relative to their scores than other bottles). Essentially wine collecting is for personal consumption thus profitability analysis is secondary. It's more of a "revenue leakage" problem – to save money in the future.

The scores (80-100) are aggregate scores crowdsourced from people who have tasted the wines using cellartracker.com to log their ratings.

A collector created his own rule of thumb: he worked out a general guideline for buying wine early in his collecting: "Less than $25 was okay for an avg 90 point wine or better, less than $60 was okay for a 92 point wine or better, less than $125 was okay for a 94 point wine or better" It's been about 10 years of collecting and he wants to see from what he has bought what was a successful purchase and what was not. It will give him guidance for future purchases.

The goal of this project is to see if there is a relationship between price and rating(quality) and to give a guidance how to determine the most valuable wines in three price categories (below $25, between $25 and $60, and up to $200)

scoring:

| |
|---|
| #Levels: [80,88) [88,90) [90,92) [92,94) [94,100] |
| #Labels: Bad OK Good Great Amazing |

Data Transformation.

Initial file was well organized in csv format. I was able to upload into R Studio by importing its Excel version (xlsx format). The file has 865 observations of 22 variables:

# > names(MyWine)

# [1] "iInventory" "Value" "NativePrice" "Size" "iWine"

# [6] "Type" "Color" "Category" "Vintage" "Wine"

# [11] "Locale" "Producer" "Varietal" "Country" "Region"

# [16] "StoreName" "PurchaseDate" "Location" "Bin" "BeginConsume"

# [21] "EndConsume" "CScore"

After "data munging" process 9 variables left:

#[1] "rating" "score" "price" "has_profit" "vintage" "scale_size"

#[7] "color" "country" "wine"

Finding average price in every range of the "rating" gives us the picture of the current state:

# > avg_price_rating

| # Bad | OK | Good | Great | Amazing |
|---|---|---|---|---|
| #117.50000 | 60.90909 | 48.10813 | 80.47664 | 155.46717 |

For example, "Bad" wine is very expensive and "Good" one is much cheaper than "ok" wine.

After displaying a few plots it becomes clear that just a couple of expensive bottles effects those results.

Also I removed one observation – a $1,000 bottle that influenced the observations.

Supervised Machine Learning.

Linear regression shows that there is no strong correlation between price and score (as was assumed by the wine collector in the beginning).

To give him the guidance what score to consider while purchasing wine I divided the data into 3 price categories:

1. Below $25,
2. between $25 and 60,
3. above $60

I was considering to give top-10 of each list but it doesn't give the best value ( the price can be slightly higher but its quality value could be significantly higher than the cheaper one).

Using "Hmisc" library I cut a numeric variable into intervals for each price category.

As a result, the value exceeds normal expectation if the price range is more than the specific score but not less than the second value. Please refer to the following table:

| **Price, USD** | **Score for max value** | **Min Score** |
|---|---|---|
| [17,20) | > 91.2 | and not less 90.3 |
| 20 | >92.5 | and not less 91.2 |
| 24 | > 93 | and not less 92.5 |
| [25,48) | > 92.3 | and not less 92.5 |
| [48,52) | >92.6 | and not less 92.3 |
| [52,58] | >95 | and not less 92.6 |
| [ 60,107) | >94.5 | and not less 94 |
| [107,154) | >95 | and not less 94.5 |
| [154,200] | >96.3 | and not less 95 |