Data Mining Project

Dung Le, Logan Zetaruk, Dave Cheng

V00856156, V00711606, V00925604

CSC 503 / SENG 474

Abstract

The techniques used for data mining are more pervasive today. The purpose of this paper is to use the Wine Review Dataset as an example to illustrate the data collection and analytical process based on the concept of data mining. The method used for this project is based on Naïve Bayes with cross comparison, SVM, and weka generated decision tree.

Based on the chosen data analysis process, the result indicated several substantial association rules that are indicative for the overall understanding of the correlation between wine grapes, wine price, countries, and their rating points. The result of the study took an important step in understanding of concept of data mining as its effectiveness in organizing the data to enhance organizations' decision making. Further studies are suggested to dwell deeper into promoting and integrating the concept data mining to general public to enhance activities of daily living.

Keywords: data, data mining, association rule, wine review

Content

1. Introduction

Data Mining is a relatively new concept that is transforming the world we live bit by bit. In every corner of the globe, we are immersed in an immeasurable amount of Big Data, being generated by mundane parts of our daily lives. While Data Mining is closely related to data analytics, in essence, it is about using that information to understand patterns and trends better. To better understand Data Mining and Big Data, we have selected the "Wine Reviews" dataset from Kaggle.com (Kaggle, 2018), because this dataset has valuable information to better understand the theories behind the Data Mining concept. The scrubbed version of the Wine Reviews dataset contains approximately 130,000 rows of scrubbed data (Kaggle, 2018). With this scrubbed version, we aimed to limit the amount of errors we might encounter during the processing stage, which is in order to have a higher quality of output. For each row of the dataset, there are 13 fields of information that describes a single wine, such as "country of origin", "sale price", "points", "varieties", and "testers" (Kaggle, 2018). Through a more in-depth analysis of the dataset, we aim to find the associative relationships and correlations between data using various data mining algorithms we learned in class. In finding a more profound meaning within the dataset, we would be able to understand the true nature of how effective Data Mining is and could make sense of seemingly random occurrences.

2. Method

To better understand the potential information of raw data, we aim to apply Naives Bayes and SVM algorithm to it. This would allow us to derive valuable information and knowledge out of the dataset. For the dataset, we imported the data table from Kaggle with aims in understanding the relationship between the following list of correlations we wish to look at. After we downloaded the data set, we first scrubbed the data to better apply data mining algorithms to. As the original data contains fields that are less important, and some of them contain more than 30% null values, we removed those columns; “taster_name”, “taster_twitter_handle”, “region_2” and “designation” were removed. Also, we realized that because almost all the data in the “title” column was unique, it was therefore not important to the point of wine. Nevertheless, this attribute contains the year of production, which is useful information. Therefore, we had to extract the year from that column and added a new column to the data named “year” which will help us identify which year that wine was produced in. Since the range of wines point value is from 80 to 100, we split the values into 2 classes: 0 (lower point < 90) and 1 (higher point >= 90). Finally, we chose the points classes as the response variable, and other attributes as explanatory variables. The finalized dataset can be seen in Fig 1, this is just a random row of data taken from the csv file which we use for data mining. We closely examine the datasets with multiple software tools, such as Python, Weka, and Microsoft Excel. Thus, the dataset generated from this process will help to determine the connection between the different data mining methods and the associative relationship each dataset has.

16	US	Building on 150 years and six generations of winemaking tradition, the winery trends toward a leaner style, with the classic California buttercream aroma cut by tart green apple. In this good everyday sipping wine, flavors that range from pear to barely ripe pineapple prove approachable but not distinctive.	87	12	California	Central Coast	Chardonnay	Mirassou	2012	0

Fig. 1 sample data set row

2-1 Description vs. Data - Text Classification

1) Text processing:

The Description attribute is a text column recorded the detailed review about the wine taste.

→ E.g: “Aromas include tropical fruit, broom, brimstone and dried herbs. The palate isn't overly expressive, offering unripened apple, citrus and dried sage alongside brisk acidity.”

To feed this data into the model, firstly, we had to split the string into words, converted it into original format (using Python: NLTK.stem.WordNetLemmatizer) (e.g: Aromas -> aroma), and removed all stop words (using Python: NLTK.corpus.stopwords) (e.g: the, a, and, etc). Then, we realized that only nouns, adjectives and verbs are important to the taste of wine. We remove all other word types from the description (using Python: NLTK.pos_tag) (e.g: overly, alongside, include, etc).

→ Result after transformation: ['aroma', 'tropical', 'fruit', 'broom', 'brimstone', 'dried', 'herb', 'palate', ’expressive’, 'offering', 'unripened', 'apple', 'citrus', 'dried', 'sage', 'brisk', 'acidity']

2) Model Selection:

After getting the data ready, we feed it to the Naïve Bayes - Multinomial model and Support Vector Machine to do the text classification.:

- Naïve Bayes - Multinomial: we used our own code since it has faster running time than using package in Scikit-learn. It reduced the running time from 35 minutes to 14 minutes.
- Support Vector Machine: we converted the word into number and fit to the Scikit Learn SVM package.

We use package "train_test_split" in Scikit Learn to hold out 20% of data for testing and 80% of data for training.

3) Model evaluation:

We compared the two algorithms by using several metrics: confusion matrix, accuracy score, sensitivity, specificity, precision, AUC score, ROC curve.

a) Confusion matrix:

Naïve Bayes Confusion Matrix:

11857	1972
3488	7605

SVM Confusion Matrix:

13418	2586
1927	6991

b) Accuracy score:

Naïve Bayes Accuracy Score → 78.09164593531818

SVM Accuracy Score → 81.89150148463206

c) True Positive Rate / Sensitivity / Recall:

Naïve Bayes Recall Score → 68.55674749842244

SVM Recall Score → 78.3920161471182

d) Specificity

Naïve Bayes Specificity Score → 85.74011136018513

SVM Recall Specificity → 83.84153961509622

e) Precision:

Naïve Bayes Precision Score → 79.40900073091782

SVM Precision Score → 72.99780724652814

f) AUC score:

Naïve Bayes AUC Score → 78.33923480664497

SVM AUC Score → 80.21998540886199

g) ROC curves graph:

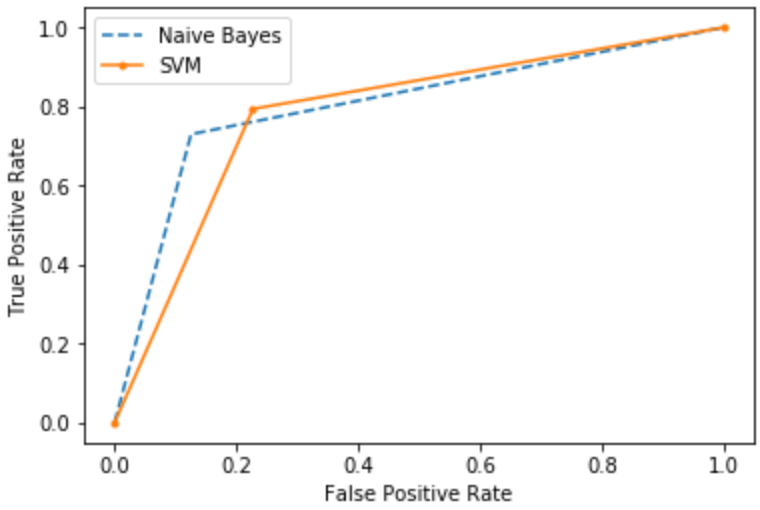

Fig. 2 ROC curves graph

2-2 Description vs. Data - Data classification

1) Processing

Data Processing was done through a few edits to the same data as above. The data was cleaned by removing single quotes (‘) and double double quotes (“”) in textedit to allow the

Weka program to properly parse the data. Another change was the removal of a couple unexpected and unnecessary new line signal (on lines 20618 and 51392) for the same reason.

In Weka, once the data was loaded, the Description and Points columns were removed to avoid interference with both the qualitative approach and to avoid a trivial classification, as the quality attribute was directly related to the points attribute. The NumericToNomial filter was then applied, and the data was processed.

2) Model Selection

NaiveBayes was chosen as the main model because it had high accuracy and ROC area, and was the easiest to compare to the qualitative data. The other algorithms run were BayesNet, OneR, ZeroR and J48. We tried to run the Weka SVM implementation, but Weka was unable to handle the large dataset with this implementation and ran out of usable memory, thus forcing us to abandon the SVM comparison.

The Weka output for NaiveBayes can be seen here:

```
=== Summary ===

Correctly Classified Instances       94228               75.6183 %
Incorrectly Classified Instances     30382               24.3817 %
Kappa statistic                          0.4952
Mean absolute error                      0.2699
Root mean squared error                  0.4246
Relative absolute error                 57.0942 %
Root relative squared error             87.3419 %
Total Number of Instances           124610

=== Detailed Accuracy By Class ===
TP Rate  FP Rate  Precision Recall F-Measure  MCC   ROC Area  PRC Area  Class
0.767    0.261    0.826     0.767  0.795      0.497 0.837      0.891     0
0.739    0.233    0.663     0.739  0.699      0.497 0.837      0.760     1
```

```
Weighted Avg.
0.756    0.250    0.763      0.756   0.758       0.497 0.837       0.841

=== Confusion Matrix ===

      a     b   <-- classified as
 58934 17937 |     a = 0
 12445 35294 |     b = 1
```

3) Model Evaluation

- Naïve Bayes Accuracy Score: 75.6183%
- Naïve Bayes Precision: 76.3%
- Naïve Bayes ROC curves graphs: 0.837

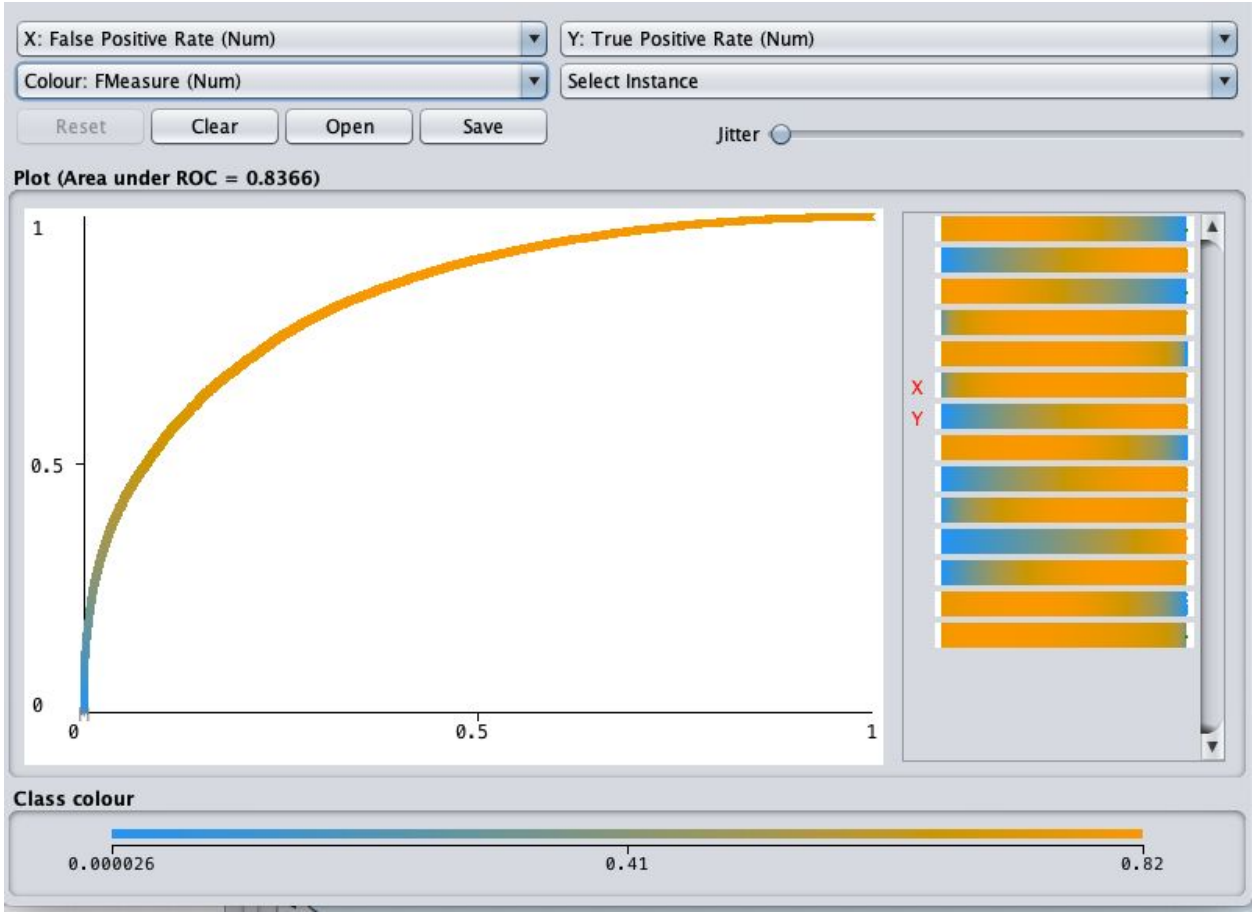

Fig. 3 Weka ROC graph (NaiveBayes)

2-3 Decision tree from data classification

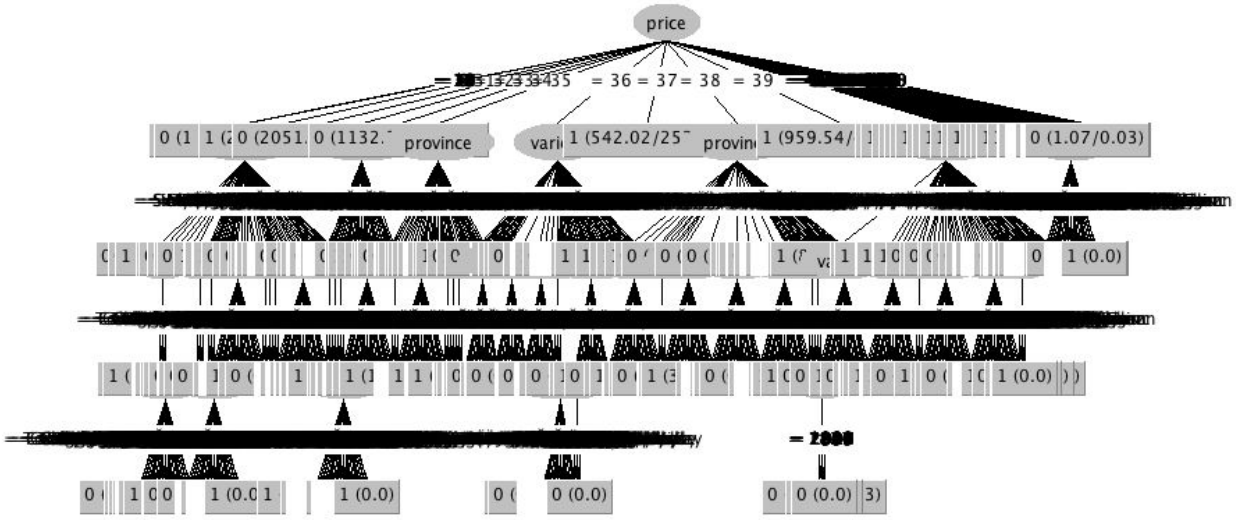

Fig. 4 Weka Decision Tree

We also generated a decision tree from the dataset using weka. While not as accurate as the Naïve Bayes algorithm, it gives a better insight into the reasoning behind the decisions. Due to the individuality of the "winery" and "region_1" columns, they were removed for this tree, as the individuality would make the tree exponentially larger. The J48 algorithm (an implementation of the C4.5 algorithm) was run on the remaining data. As seen in Fig. 4, "price" was the main starting point, typically followed by "province", then either of "variety" or "year" (or vice versa). Of course, any value could come after "price", but these trends appeared often enough to be noticed. This means that the best way to select a high quality wine is to first look at the price, then the area the grapes were grown, followed by the variety of wine you are looking for, then finally the year.

The Weka output for J48 can be seen here. Note that the expanded tree is included in the appendix (Decision Tree.model).

```
=== Summary ===

Correctly Classified Instances       92456               74.1963 %
Incorrectly Classified Instances     32154               25.8037 %
Kappa statistic                          0.4326
Mean absolute error                      0.3524
Root mean squared error                  0.4171
Relative absolute error                 74.5489 %
Root relative squared error             85.8001 %
Total Number of Instances           124610

=== Detailed Accuracy By Class ===

TP Rate FP Rate Precision Recall F-Measure MCC    ROC Area  PRC Area  Class
0.853   0.437   0.759     0.853  0.803     0.439  0.803     0.862     0
0.563   0.147   0.704     0.563  0.626     0.439  0.803     0.701     1
Weighted Avg.
0.742   0.326   0.738     0.742  0.735     0.439  0.803     0.800

=== Confusion Matrix ===

      a     b   <-- classified as
 65557 11314 |     a = 0
 20840 26899 |     b = 1
```

3. Results

For the results, we decided to do a qualitative vs. quantitative approach: i.e. description vs data. For the qualitative approach, words that are associated with wines with higher quality will be scored higher than words that are not. More high scoring words means the wine is more likely to be classified higher. For the quantitative approach, all other data sections (price, country, year, etc.) were compared and classified in Weka.

The Naïve Bayes implementation of the qualitative data had a higher accuracy than the quantitative data (78.1% vs. 75.6%). The SVM implementation of the qualitative data also had a higher accuracy than the highest quantitative data implementation, Bayes Network (81.9% vs. 75.8%). These comparisons show that the qualitative predictions are more accurate than the quantitative predictions. This may be attributed to the variety and the expressiveness of words compared to the monotonous data. This association may directly relate to the quality of the wine, as the higher quality of wine will typically have features that are relatively easy to describe. More descriptive words also signifies the experience of the wine reviewer able to express, this reinforces the high quality of the wine.

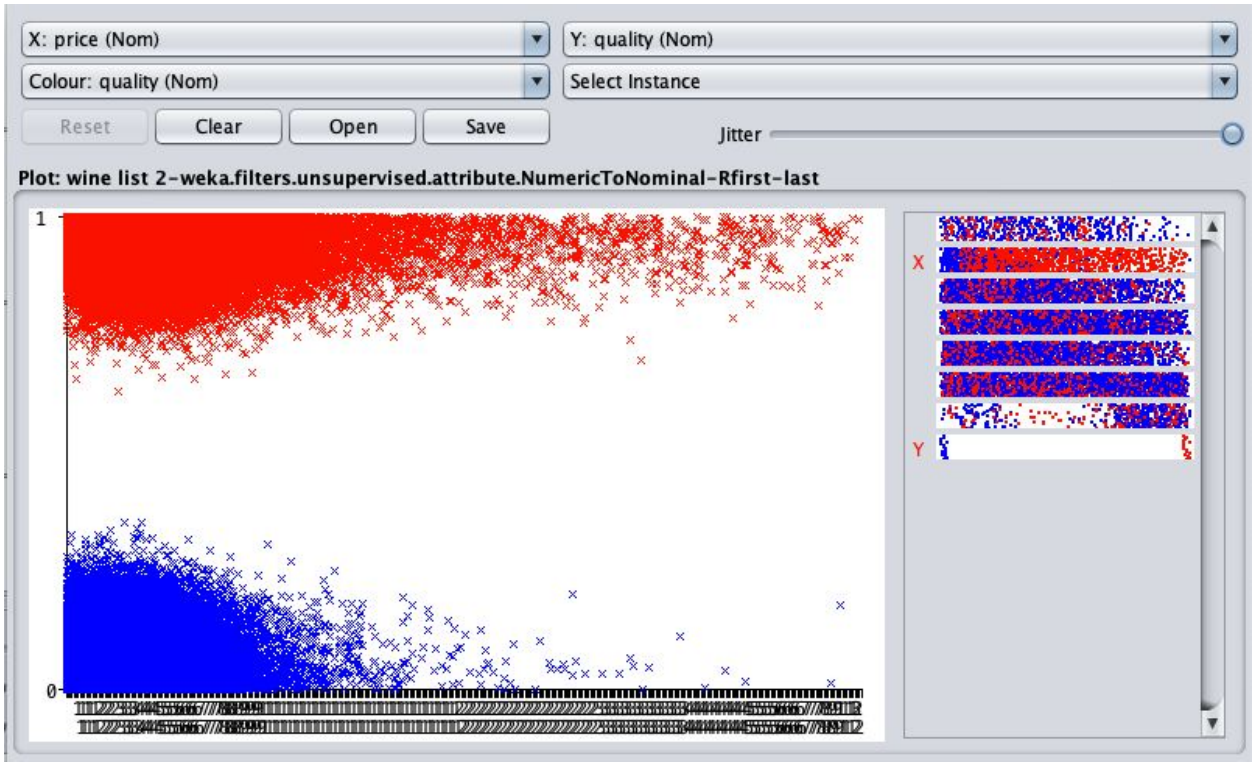

Fig. 5 Quality (y axis) vs. price (x-axis)

Another, simpler comparison is Price to Quality. As seen in Figure 5, the high quality wines (defined in Red) are more likely to have higher costs than the lower quality (Blue) wines. As cost increases along the X-axis, the appearance of both wines drop off, but there are many more high quality wines remaining. Of course, there are a few high-costed, low-quality wines, but the majority of high-costed wines will have higher quality. Spending (much) more money on a bottle of wine is generally an indication that it will also be rated higher.

We also generated a decision tree to better understand the correlation between the relationship of description vs. data, however this is generated with the Data classification approach from above. The decision tree provides a clear indication of how the algorithm chose the path that it did, and what the best way to choose a wine is.

4. Conclusion

As data mining has been continuously involved in our daily lives, understanding its concepts and gaining the ability to use them is essential to enhance our daily work. Based on Wine Review Dataset, how we apply various algorithms and explore the correlations that lie within the dataset help to determine some of the important concepts that can be used by the business consumers.

We can derive business intelligence from the description vs. data method because the richness of the information in the description column. It contains many points of interest that provided the basis for this project. We looked at the descriptive words from the “description” attribute, and then at the other attributes from the same item. It became apparent that as a

reviewer described the wine in greater detail, the higher the possibility the wine would receive a higher quality score on average, and while the other information was important, it was not as vital as a good description.

This derived information would present business opportunities to buyers who want to purchase such wines and expect a relatable product. This also signifies a true positive result from a strong correlation, meaning a person who wishes to purchase a good quality wine can do so by simply reading the description. As the description would fit closely to what they wanted, have less possibility of deviation, and leave the customer satisfied. This aspect of the dataset was further explored with a cross examination from both qualitative and quantitative approach.

5. Reference

Wine Reviews. (2017, November 27). Retrieved November 26, 2018, from https://www.kaggle.com /zynicide/wine-reviews/data