Predicting *Review Rating* for Wine Recommendation

[CSE 190 Assignment 2]

Fan Chao
University of California, San Diego
PID: A53078965
chf004@ucsd.edu

Pengbo Li
University of California, San Diego
PID: A53079916
pengboli@ucsd.edu

Renxiang Yan
University of California, San Diego
PID: A53079806
reyan@ucsd.edu

Nasha Zhai
University of California, San Diego
PID: A53082522
nzhai@ucsd.edu

ABSTRACT

In this assignment, we try to obtain a suitable model for *Wine Recommendation System* based on the data-set from *CellarTracker*. Since we know different people have different tasting note and personal stories from the collection of wine reviews, it's reasonable that we create our *Wine Recommendation System* according to these differences and make an accurate system with the suitable features we choose from the data-set. Generally, we consider *linear regression*, *Random-Forest regression* to get various models and use the mean absolute error(MAE) as the criteria of model accuracy. More specifically, we firstly describe basic properties of the data-set, then identify the review rating(points) predictive task, consider the features that may be relevant to it, describe literature and research relevant to the topics, and have an analysis results in the end.

Keywords

Wine Recommendation System; CellarTracker;
linear regression; Random-Forest

1. INTRODUCTION

CellarTracker is the world's largest collection of wine reviews from users who love wines. It also features a massive database of community wine reviews that anyone can search to find recommendations on wines. Based on this strong user platform, we come up with the recommendation system as our topic and look for the appropriate model to achieve this goal. We know the recommendation system involves predicting user responses to options. For the data-set we choose, we will look through the features like user review time, review text, review points, wine year and so on to dig for the problem if a specific user will like a kind of wine or not. As we all know, recommendation systems use different kinds of technologies. We can classify them into two broad groups. One is Content-based systems which examine properties of the items recommended. Another one is Collaborative filtering systems recommend items based on similarity measures between users and(or) items. The items recommended to a user are those preferred by similar users. For our wine recommendation system, we'll mainly consider the Content-based systems since we want to use the supervised learning to construct a good model and also avoid no use of new users(items) problem.

Number of reviews	2,025,995
Number of users	44,268
Number of wines	485,179
Users with > 50 reviews	5,957
Median no. of words per review	29

Table 1: Dataset statistics

2. DATASET IDENTIFICATION AND DESCRIPTION

The data-set we will use to build the model can be found at the website http://snap.stanford.edu/data/cellartracker.txt.gz which is from Stanford large network data set collection. The data-set we are going to explore is baisically about the wine review originated from CellarTracker, one of the world's largest collection of wine reviews. The data span a period of more than 10 years, including all 2 million reviews up to October 2012. Reviews include product and user information, ratings, and a plaintext review. We also have reviews from beeradvocate and ratebeer. We also attach the dataset statistical table from the website as Table 1 shown on this page.

2.1 Dataset *Description*

Since we have already seen the basic data-set statistics, we can then get a bit further on the content of this dataset. A sample data entry can be shown here as Table 2 for further study. We can see there are different features which might have some influences on our further predictions. Intuitively,

wine/name:	1991 E. Guigal C&#244;te-R&#244;tie La Turque
wine/wineId:	13162
wine/variant:	Syrah
wine/year:	1991
review/points:	N/A
review/time:	1171670400
review/userId:	1
review/userName:	Eric
review/text:	OMFG, this is off the charts. Smoke, gunpowder, a sexy beast of a wine that drinks incredibly well.

Table 2: Sample data entry

average-review-points	89.019805
average-review-time	1235468920.06
average-wine-year	2002.614511
average-review-text-length	39.176031

Table 3: Basic data statistics

we try to divide the data-set into two parts, one is for users' features, another is for wines' features. For users' features, we list user-average-review-points, user-average-review-time, user-average-review-year, user-average-length of review text, the number of some user-data shown on the overall data-set. For wines' features, we consider the relative features like item-average-review-points, item-average-review-time, item-average-wine-year, item-average-length of review text, the number of some item shown on the overall data-set. Moreover, we also calculate the average through all of the data-set on some features, which are shown as the Table 3. Furthermore, we can discuss the relationship between different feature and then have good sense to pick useful features for our prediction. The exploration on the relationship between different features is discussed in the next sections.

2.2 Exploratory Analysis

Aimed to obtain good features for our model and have a clear mind about what kind of prediction task we are gong to work on, we try to explore further on the properties on our data-set.

2.2.1 General Attempt

The first time we go through this data-set, we naturally think about the distribution of each feature listed in the data-set and consider the features which may vary by different users or wines may have some influence on our prediction task like user rating prediction task.

Based on this thought, we use the plot functions to figure out the distributions of these different features.

Figure 1 shows the histogram for "review/points" in train data-set. From the figure, we can see most "review/points" distributes between 80 and 100.

Figure 2 shows the histogram for "review/time" in train data-set. From the figure, we can see most "review/time" distributes between $1.0 * 10^9$ and $1.4 * 10^9$.

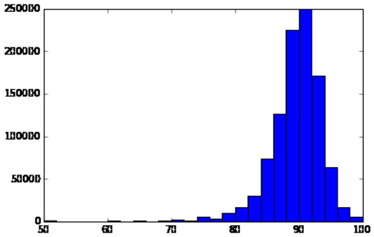

Figure 1: User review points

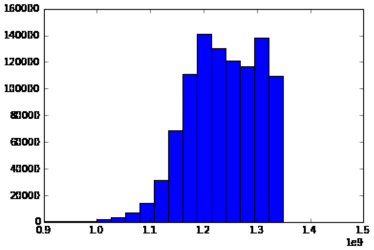

Figure 2: User review time

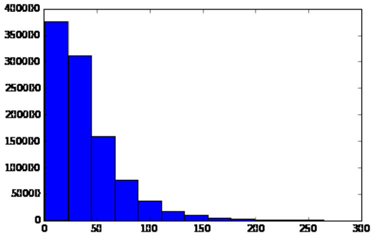

Figure 3: User review text length

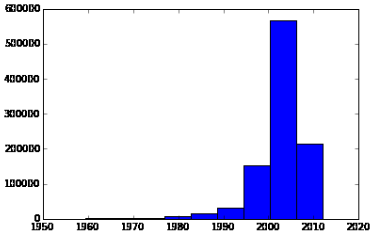

Figure 4: Wine year

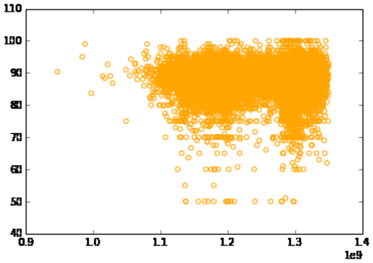

Figure 5: User average rating and user average review time.

Figure 3 shows the histogram for "wine/year" in train data-set. From the figure, we can see most "wine/year" distributes between 1980 and 2010.

Figure 4 shows the histogram for the length of "review/text" in train data-set. From the figure, we can see most length of "review/text" distributes between 0 and 200.

2.2.2 *Relationships between different features*

Since we have filtered data which means the added new features like the average wine years have been rounded and the data with NA values have been removed, we digged it further on the relationships between filtered data-set. More specifically, we have change the original data-set with 2,025,995 reviews into a data-set of 1521552 reviews, then we use the first 100,0000 as the training set and use the rest of the data(52,1552 reviews) as the test set.

Inspired by the lecture slides, we firstly perform our exploratory on the average rating points.

Figure 5 shows the relationship between the average review time and average rating corresponding to each user. The x-axis indicates the average review time, and the y-axis indicates the average rating. We can get the information: when the x-axis ranges between around 1.1-1.3, the data of rating is the most confident.

Figure 6 shows the relationship between the average year and average rating corresponding to each user. The x-axis indicates the wine's year and the y-axis indicates the rating corresponding to each user. In the figure, we can conclude that the some obvious fact: (1) the number of the point is the total number of user. (2) Normally, the highest rating ranges from year 1980-2000.

Figure 7 shows the relationship between number of text lines and average rating corresponding to each user The x-axis indicates the number of text lines, and the y-axis indicates the average rating. We can get the information: the more number of text lines user write, the more confident the rating data is.

Figure 8 shows the relationship between frequency of user and average rating corresponding to each user The x-axis indicates the frequency of user, and the y-axis indicates the rating. We can get the information: when the x-axis ranges between around 2000-3000, the data of rating is the most confident.

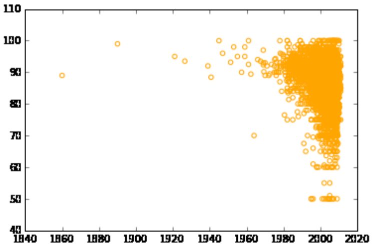

Figure 6: User average rating and user average wine year.

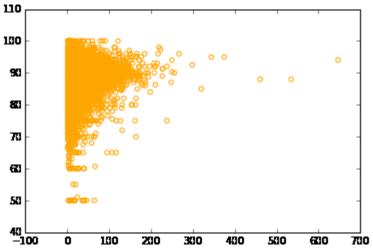

Figure 7: User average rating and user average review text length.

Similarly, we can also do this kind of experiments on the wine-associated features. Let's try to draw these plots to work on the wines' associated features.

Figure 9 is about the connection between wine average rating points and wine average review time.

Figure 10 is about the connection between wine average rating points and wine average year.

Figure 11 is about the connection between wine average rating points and wine average text length.

Figure 12 is about the connection between wine average rating points and some specific wine showing times throughout the dataset.

3. PREDICTION TASK IDENTIFICATION

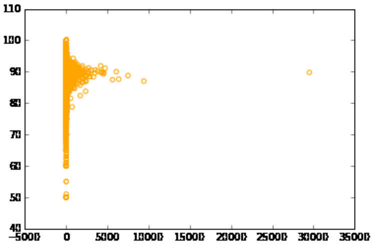

Figure 8: User average rating and user showing times throughout the dataset.

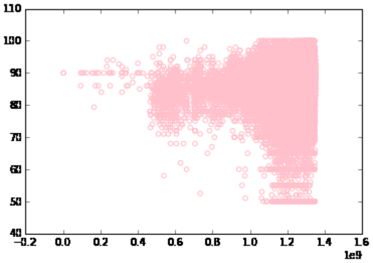

Figure 9: Wine average rating and wine average review time.

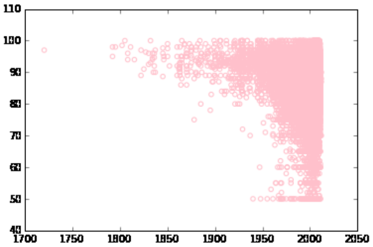

Figure 10: Wine average rating and wine average year.

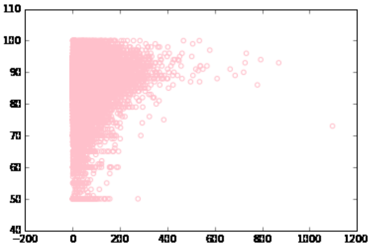

Figure 11: Wine average rating and wine average review text length.

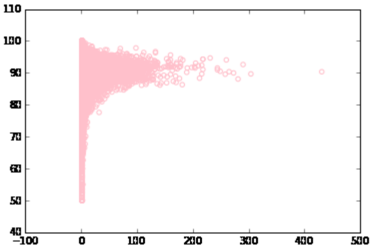

Figure 12: Wine average rating and wine showing times.

Based on the discussion above, we have made our prediction task for this assignment to be the wine recommendation system for the old and new users from the online platform CellarTracker. In an easy-to-understand language, we try to make suitable predictions on whether a chosen user will like a specific kind of wine or not. The way we judge if the user like it or not is to make the predictions on the user review score on the wine we choose and then compare the average review rating score and the predictive score. If we have the predictive score is greater than the user's average rating score on various wine, then we say the user like this wine, otherwise we say the user dislike this wine.

We also set a criteria for assessing the validity of your predictions and confirming that they are significant. The criteria we use is MAE which is short for mean absolute error. As shown in the exploratory analysis, we did preprocessing of the data-set and try to plot some figures of the features to find out which kind of feature will have significant influence on the user rating points and then we selected some suitable features from this experiments.

4. RELEVANT FEATURES

The features we use for our models are user-related average review-points, user-related average review-time, user-related average wine-year, user-related average review-text-length, user-related popularity, wine-related average review-points, wine-related average review-time, wine-related average wine-year, wine-related average review-text-length, wine-related popularity, review-time, wine-year, review-text length, totally 13 features.

5. MODEL DESCRIPTION

Based on the description above, we aim to predict the rating of the wine on the test set. We choose the following features for the prediction task: the corresponding average points, review time, year of the wine, length of text for both users and items, their individual review time, year of the wine and length of text. So the features can be represented as the following matrix.

$$X = [1, \quad points\_users, \quad time\_users, \quad wine\_users, \quad text\_users, \quad \ldots]$$

Y corresponds to the points in the training set.

5.1 Baseline Model

In the baseline, we simply predict the points of wine using the following strategy. If the item predicted in the test set appears in the training set, we use the average points of the corresponding item in the training set. Otherwise, if the item is not in the training set, we use the average of all of the items in the training set as a its prediction.

5.2 Linear Regression Model

Linear regression is an approach for modeling the relationship between a scalar dependent variable y and one or more explanatory variables (or independent variables) denoted X. A simple linear model can be represented as $Y = X\beta$, and the solution for that is $\beta = \underset{\beta}{argmin} \sum_i (y_i - x_i^T\beta)^2$

5.3 Ridge Regression Model

Ridge regression helps to penalize the size of the regression coefficients in linear model. A ridge regression model is of the form $\underset{\beta}{argmin} \sum_i (y_i - x_i^T \beta)^2 + \lambda \sum_{i=1}^{p} \beta^2$, with the solution $\hat{\beta}^{ridge} = (X^T X + \lambda I)^{-1} X^T Y$

5.4 Random Forest

A random forest is a meta estimator that fits a number of classifying decision trees on various sub-samples of the dataset, which helps to improve the predictive accuracy and control over-fitting. Here, we use the RandomForestRegressor provided by sklearn package in Python.

6. LITERATURE AND RESEARCH

The database we chose comes from SNAP and has been used in professor Julian McAuley's previous paper. What our idea to do prediction is based on the assignment1: Helpful prediction baseline provided by professor Julian McAuley and homework1: linear regression to predict review/taste. In homework1, we use linear regression model to predict review/taste corresponding to beer/ABV. In this project report, we do a major change.

First, we do some pre-processing on our dataset like baseline in assignment1, adding the average rating into the features.

Secondly, by optimizing the relation coefficients, we use ridge regression to estimate:

$$\underset{\beta}{argmin} \sum_i (y_i - x_i^T \beta)^2 + \lambda \sum_{i=1}^{p} \beta^2$$

with the solution

$$\hat{\beta}^{ridge} = (X^T X + \lambda I)^{-1} X^T Y$$

Compared to the regular linear regression, our model penalizes the size of the regression coefficients firstly. (the penalized part is as below)

$$\lambda \sum_{i=1}^{p} \beta^2$$

So that the coefficient we select is the reasonably optimal one, considering the correlation coefficient for each coefficients.

The result of our model is pretty good. Its MAE is greatly reduced compared to both of the baseline and the linear regression. However, our result is still far away from professor's paper result.

In that paper, professor Julian McAuley used modified latent factor model by introducing user experience as a function of time. It based on the assumption that "By individually learning for each user the rate at which their experience progresses, we are about to account for both types of behavior". The performance of the model is extremely good. It let us know that how much we still could do to improve our model.

7. RESULTS ANALYSIS

There are two general methods for model evaluation, Mean Absolute Error(MAE) and Mean Square Error(MSE). MAE is defined as

$$\text{MAE} = \frac{1}{N} \sum_{i=1}^{N} |y_i - \hat{y}_i|$$

as MSE is defined as

$$\text{MSE} = \frac{1}{N} \sum_{i=1}^{N} (y_i - \hat{y}_i)^2$$

Though both methods reflect to the deviation between the estimate value and true value, they still have slight difference. The method of MAE can fairly regard each item in the test data set while the method of MSE will expand the effect of those items whose estimate values are far away from their true values. However, MSE is easily computed and analyzed since it is convex. Here we select to use MAE to evaluate our models and pick out the best one.

Computing MAE on the test data set using baseline model, random forest regression model, linear square regression and ridge regression model. The results are shown in Table 4.

Model	MAE
baseline	3.04199508305
random forest regression	2.74082737675
linear square regression	2.65466068719
ridge regression	2.65217864745

Table 4: MAE on test data set using different models

Table 4 shows the baseline model with the highest MAE may be the worst one. The reason is the limit number of features that could not reflect all the information for the estimate. The ridge regression model with the lowest MAE should be the best one for predicting review rating. Compared to linear square regression, ridge regression makes some effort to eliminate multicollinearity and avoid overfitting.

8. REFERENCES

[1] J.McAuley and J.Leskovec, *From amateurs to connoisseurs: modeling the evolution of user expertise through online reviews*, WWW, 2013

[2] Wang S G and Chow S C. *Advanced Linear Models..* New York: Marcel Dekker Inc, 1994