190 Assignment

Zijian Tao
Mingjun Ji
Nan Chen

Dataset

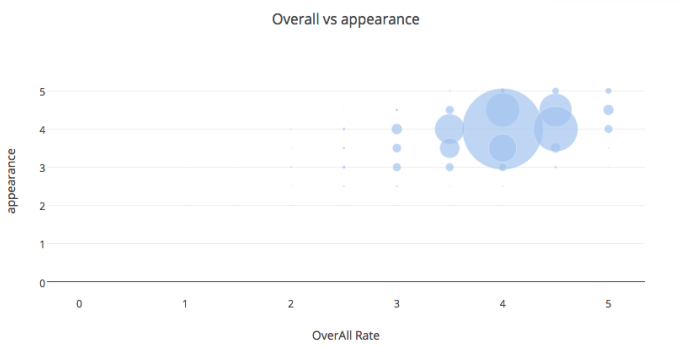
1. Identify a data set to study

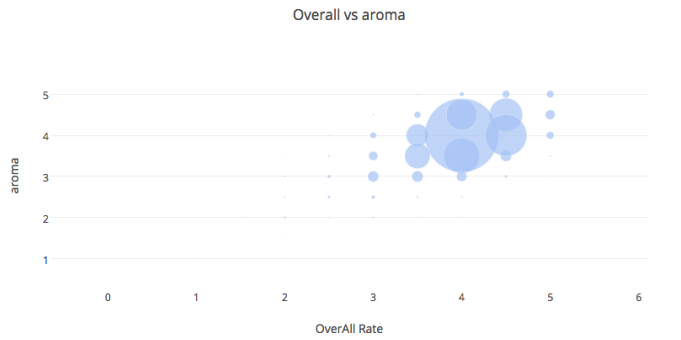
We choose the dataset ‘beeradvocate.txt’. The dataset consists of 1586614 data entries. Every entry has the following fields:

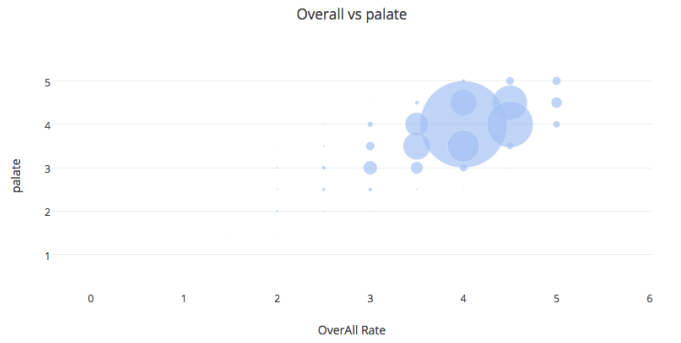
1. beer/name : Name of the beer
2. beer/beerId : Unique beer identification
3. beer/brewerId : Unique brewer identification
4. beer/style : Beer category
5. beer/ABV : Alcohol by volume
6. review/profileName: Reviewer’s profile name / user ID
7. review/time : UNIX time when review was written
8. review/aroma : Rating based on how the beer smells
9. review/palate : Rating based on how the beer interacts with the palate
10. review/taste : Rating based on how the beer actually tastes
11. review/appearance: Rating based on how the beer looks
12. review/text : Personal observations made by the review in text format
13. review/overall : Cumulative experience of the beer is encapsulated in this rating

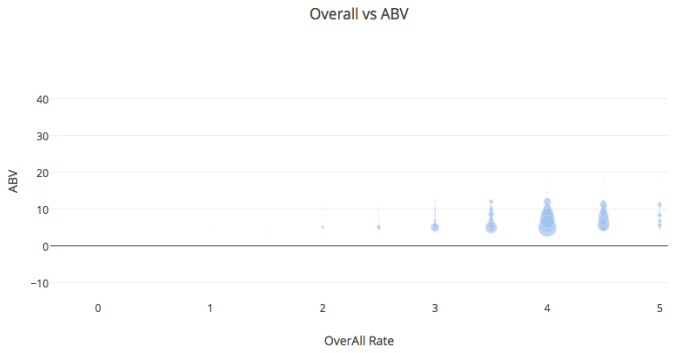
2. Exploratory task

The most relevant ratings provided are the scores of ABV, aroma, palate taste and appearance. Therefore, we first explore the relationship between the related rating and overall rating.







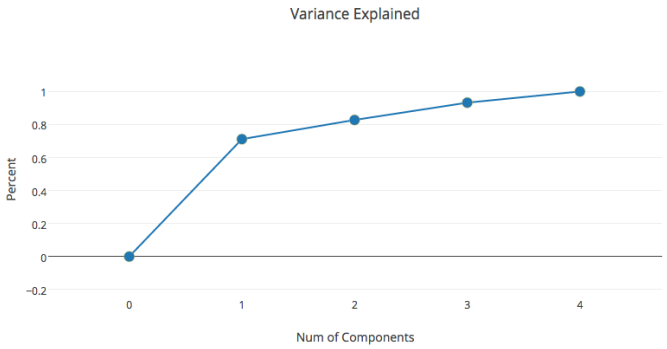


In the above graphs, the size of the circles demonstrate the occurrence of a specific x,y value combination, we could observe that the ratings for overAll with 4 has the highest occurrence, and there is a vague linear relation between the overall rate with appearance, aroma, palate, and taste. As for ABV, it seems that people tend to rate beers with smaller ABV than higher one, and the ratings for 4 is the most.

Since all of the previous four rating is correlated with overAllRating, we might need to just extract one or two of them to predict overAllRating, since containing all of them might cause double counting. So we calculate the pearson correlation coefficient between the these rates.

overAllRating	appearance	0.5074
overAllRating	palate	0.7020
overAllRating	aroma	0.6227
overAllRating	taste	0.7893

Therefore, taste and palate is more correlated with overAllRating. We might incorporate these two as predictors. Another way would be to just run principal component analysis on these features since they might have a high variance on the first eigenvector. The following is the graph of how many components versus variance explained.

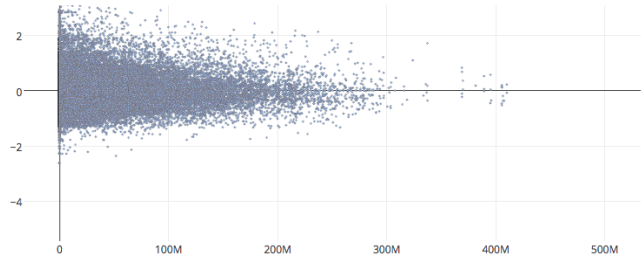


The output is quite reassuring, the first component explained more than 70% of variance, indicating a potential decent replacement of the four predictors.

Then we try to explore other predictors, like in assignment 1, such as review length. However, the correlation coefficient between review length and overAllRating is 0.05109, too small to be qualified as a predictor.

We also look specifically into the review/text field to see if we can find anything to the . We want to find if there are any relation between the choice of words from a user, to the overall rating he/she gives. Since we cannot measure the correlation between this and the overAllRating directly, we decide to use the text features separately from other numerical features such as the review/taste or review/palate first and see how well it does.

There might also be relationship between user's experience and his rating, since professor mentioned that a user with more experience in the community might rate differently from a novice one. So we generate the following graph, with x-axis representing the exact time length from the user's first appearance in the community when he rated the beer, and y-axis being the deviation of the rating from the beer's average rate.



The graph shows an interesting pattern that more experienced user tend to rate close to average. But since it's too noisy for users with no much experience, it would not be quite effective when we predict for new test data.

Predictive Task

Given an entry, we want to predict the 'review/overall' field with the other fields in the entry as well as with previous knowledge about the beer/reviewer involved in this entry. And we want to know which feature correlates most greatly with the overall ratings.

For example, provided an entry with 'beer/ABV' = 5 and the average ratings of the reviewer that reviews the beer is 0.27, we want to obtain the prediction of 'review/overall' field in this entry.

We measure the accuracy of our prediction by 'Mean Squared Error' on the test data. The sizes of training set and test set are as follows:

- Training: 100,000 entries
- Test: 10,000 entries

Literature

1. Literature relevant to the task

The dataset 'beeradvocate.txt' comes from the Online Review collection from Stanford Large Network Dataset Collection. "Learning Attitudes and Attributes from MultiAspect Reviews" [2] is a work that studies the relationship between ratings and various attributes. It provides a model called "Preference and Attribute Learning from Labeled Ground-truth and Explicit Ratings" that focuses on the modeling the influence of textual data onto the ratings.

2. Relevant uses of similar data

A similar dataset is 'ratebeer.txt'. In [2], this dataset was identified to be similar to beeradvocate.txt. It is used to provide non-English reviews to the analysis of [2]. This improves the generality of the reviewers, enhancing the model they developed,

3. Studies that solve similar tasks

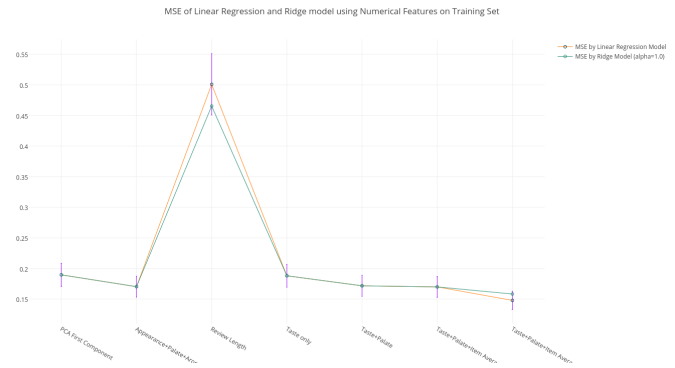
[3] introduces OPINE system that utilizes relaxation labeling to explicitly extract features from data. It tries to discover important features related to the target, just as we do. It not only calculates relations between a feature with each other one, but also the opinion regarding those features. It then discovers the pattern between opinions and features by ranking the calculated opinions.

Results

SInce we used very few features, we decided to use the most basic Linear Regression Model to train our models as we do not really think overfitting will be any issues.

In order to confirm our assumption on the issue of overfitting, we decide to use the
Ridge model with L2 norm regularization parameter (sklearn.linear_model.Ridge) on the above feature vectors.

Below is the MSE of the Linear Regression and Ridge model with different features on the Training Set:



We also used collaborative filtering technique on features including: average overall rating, bias of a beer in respect of all users who reviewed it, and bias of a user in respect of all beers he/she has bought:

Total Average + user bias + beer bias	0.265

The λ we pick is 1. We have tried different lambda values and concluded that the MSE reaches minimal when λ = 1.

For the text analysis part, we used both the **TF-IDF model** and **Bag-of-Words model** with different Dimensionality Reduction techniques such as **Singular-value Decomposition** and **Latent Dirichlet Allocation.**

We used the **sklearn.feature_extraction.text.TfidfVectorizer** library for our TF-IDF model. We take into account features made of only a single tokens(**unigrams**) with at least **four characters**. They would also have to be words or vocabularies. The independent variable in this model is the **maximum number of features allowed.**

We used the **sklearn.feature_extraction.text.CountVectorizer** library for our Bag-of-Words model. In our Bag-of-Words model we have the exact same restrictions as the above TF-IDF model.

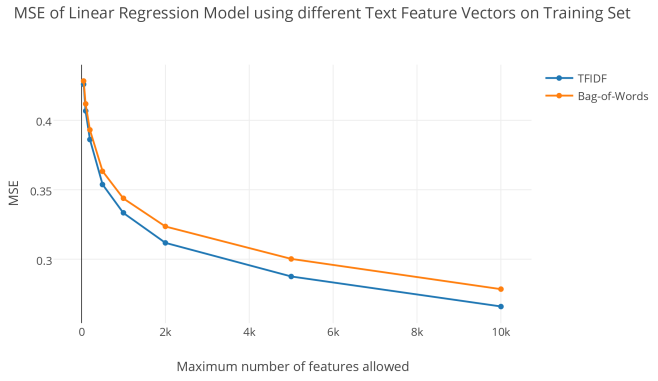
A potential independent variable for the above two models might be taking into account other **n-gram words** rather than just unigrams.

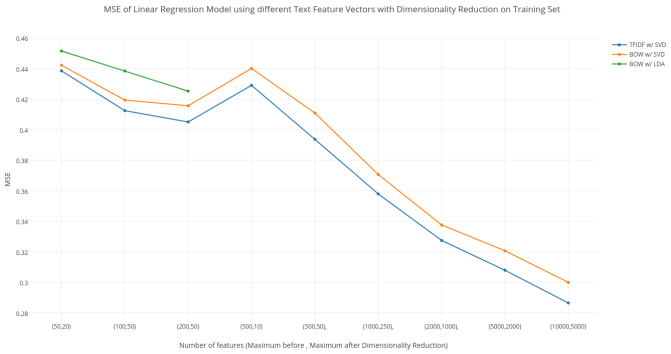
We used the **sklearn.decomposition.TruncatedSVD** library for our Singular-value Decomposition. When using this technique, the independent variable is the **number of components allowed after Dimensionality Reduction**.

We used the **lda.LDA** library from http://pythonhosted.org/lda/ as our Latent Dirichlet Allocation reduction technique. The independent variable here is also the **number of topics allowed after the dimensionality reduction**. In addition, the **number of sampling iterations** can also be changed accordingly for best results. In our model, we used the default sampling iteration 2000 for the parameter.

The reason behind our choice of parameters is that using other n-grams will potentially require much more space and a word with less than 4 characters is more likely to be a stem word (but we should really further tune this parameter in the future).

Below is the MSE of our different text models on the Training Set:





Based on the above graph, using only the text feature to predict the overAllRating is not a very good choice since even with a maximum of 10000 dimension feature vector, the TF-IDF and Bag-of-Words models still underperformed than the previous models having only the numerical values as features.

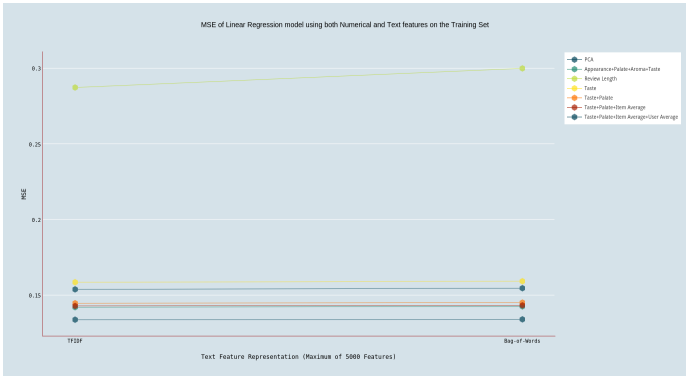
Moreover, the Dimensionality Reduction Techniques in this case are not so useful since they produced a higher MSE than using normal TF-IDF or Bag-of-Word vectors.

During the testing, the Latent Dirichlet Allocation appears to be extremely time consuming as the default sampling iteration is set to 2000 which seems to be a lot. We decided to only test it with two different inputs and it appears to be less optimal than the normal Bag-of-Words model. Therefore we remove it as an option for representing text features.

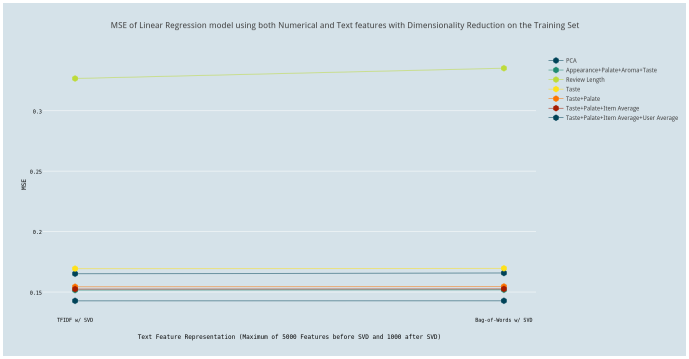
In order to really prove that the text features are not useful to our predictions, we incorporated them with each of the above features from numerical values(i.e taste, palate, etc)

Below is the MSE of our combined Linear Regression models on the Training Set:

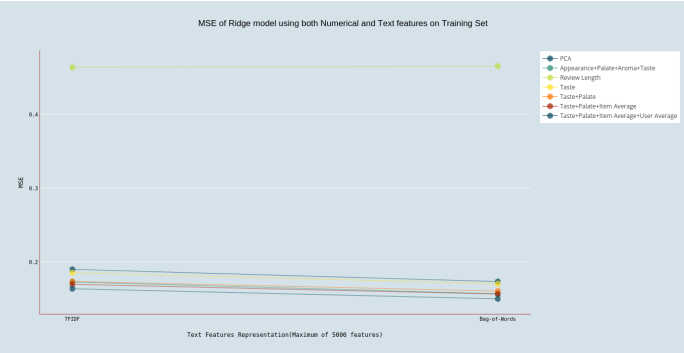
Without Dimensionality Reduction:



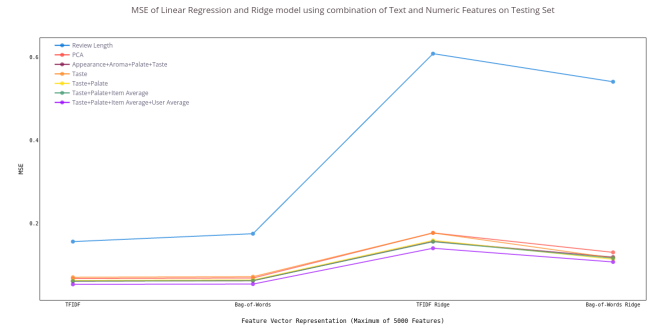
With Dimensionality Reduction:



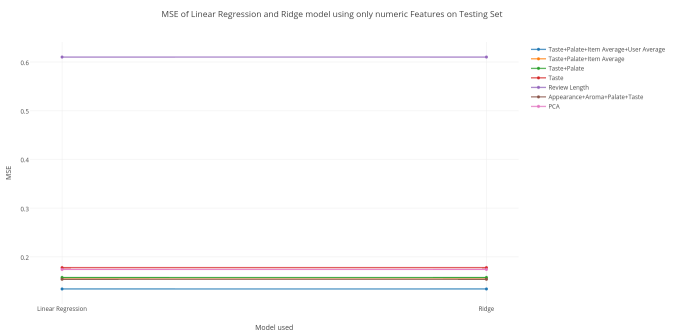
We used the same feature vectors on the Ridge model to test one last time the issue of overfitting. Below is the result:



From these graphs, we did not see any issues with overfitting and the SVD dimensionality reduction technique did not bring us any better results. However, by combining both text features and numerical features, the model received a lower MSE. Therefore, we decide to use these combined feature vectors without dimensionality reduction on our test set using Linear Regression model. Here is the result:



We also test the MSE of using only numerical features on the Testing set for comparison:



Conclusion:

Using only numerical features such as review/taste, review/palate to construct feature vectors produces pretty good results on both training and testing sets. However, given the size of training data is much larger than the testing data, we expect the MSE on the testing data to be lower.

Using only text features is not a very good choice as the MSE is still quite high with a 10000-dimension feature vector. It not only consumes large amount of space but also has the potential of overfitting.

When we combine these two categories of features together, we get better result on both the training and testing set. The difference mainly comes from the choice of numerical features included, as the TFIDF and Bag-of-Words model produce similar results given the same numerical features are used.

Issue of overfitting still seems to exist in the final model used as there is a slightly higher difference in MSE when using the Ridge model in comparison to Linear Regression model.
One way of preventing this from happening might be decreasing the maximum number of features allowed. Using a 5000-dimension vector on the

testing set containing only 10000 entries is prone to have overfitting issue. Another way is to use cross-validation techniques to make sure that the model does not over fit.

We might be implementing the Dimensionality Reduction techniques incorrectly as it consistently produces higher MSE. We have questions on what to use LDA on (beer/style given all the review/text or a BOW representation vector or just the review/text). The other explanation will be the data set somehow does not prefer low dimensional structure.

Conclusion

Having firstly found that palate rating and taste rating as suitable candidate features to make a predictor, we make efforts on proving our discovery by two linear regression models. We carefully pick combinations on several simple features for comparison, and obtained an MSE as low as 0.17, a minimal among all simple feature combinations we picked.

We then exploratively work on extracting textual features. Incorporating models of Bag-of-Words and TF-IDF and dimensionality reduction techniques of LDA and SVD, Although the MSE on picking a large number of features reaches as low as 0.29, we observe that the two textual mining models do not outperform the previous numerical model.

Finally we work on incorporating both numerical features and textual features. We received an MSE as low as 0.08 on test data, showing that the features generate an outstanding predictor.

Citations

[1] J. Bennett and S. Lanning. The Netflix prize. In KDD Cup and Workshop, 2007
[2] McAuley, J., J. Leskovec, and D. Jurafsky. “Learning Attitudes and Attributes from MultiAspect Reviews.” In 2012 IEEE 12th International Conference on Data Mining (ICDM), 1020–25, 2012. doi:10.1109/ICDM.2012.110.
[3] A. Popescu and O. Etzioni. Extracting product features and opinions from reviews. In HLT, 2005.