Recommendation for Online Retail Data

Sumit Patel

Springboard Capstone #2

Abstract

Recommendation Systems are widely used to recommend products and services to customers and clients. This system allows for the prediction of the rating or preference a user would give to an item. The dataset is for a UK-based and registered non-store online retail. The company mainly sells unique all-occasion gifts. The dataset is comprised of 8 attributes consisting of InvoiceNo, StockCode, Description, Quantity, InvoiceDate, UnitPrice, CustomerID and Country. The dataset is composed of 532,618 rows. Through exploratory data analysis and using graphlab we will identify how to make a suitable recommender systems for the dataset. Appropriately matching customers to items that they may be more inclined to purchase would increase the likelihood of another purchase using recommender systems approach.

UCI Machine Learning Repository Dataset can be found at:

https://archive.ics.uci.edu/ml/datasets/Online+Retail

Recommendation for Online Retail

Recommender systems provide recommendations through collaborative filtering, content-based filtering or hybrid approaches. Collaborative filtering approaches build a model from a user's past behavior as well as similar decisions made by other users which is then used to predict items that the user may also have interests in. Content-based filtering approaches utilize use a series of discrete characteristics of an item in order to recommend additional items with similar properties. These approaches are often combined in hybrid systems. We will train and test various recommender systems to find one that is most suitable for this dataset application suing graphlab.

Method

Data

Dataset is a UCI Learning Repository dataset that consists of 532,618 rows of online retail data and 8 columns consisting of attributes InvoiceNo, StockCode, Description, Quantity, InvoiceDate, UnitPrice, CustomerID and Country. The dataset can be found at: https://archive.ics.uci.edu/ml/datasets/Online+Retail. Our recommender system will be based on the features of CustomerID, StockCode, Description and Quantity to provide our recommendations.

Extract, Transform and Load (ETL)

The dataset exists in xslx format which we converted to csv for easier processing. After converting the columns to the necessary data types we created and SFrame. When converting to SFrame some of the columns required data type manipulation once again. We also concatenated StockCode and Description to a column named Items for simplicity. We formatted the SFrame to only have our columns of interest: CustomerID, Items and Quantity. We then restructured the SFrame to show Items ordered by CustomerID using a groupby function. This action resulted in reducing the SFrame to 273,425 rows from 532,618 rows of data. Since multiple customers are able to to order similar items we wanted to identify the total number of unique items which happens to be 5,732.

Exploratory Data Analysis (EDA)

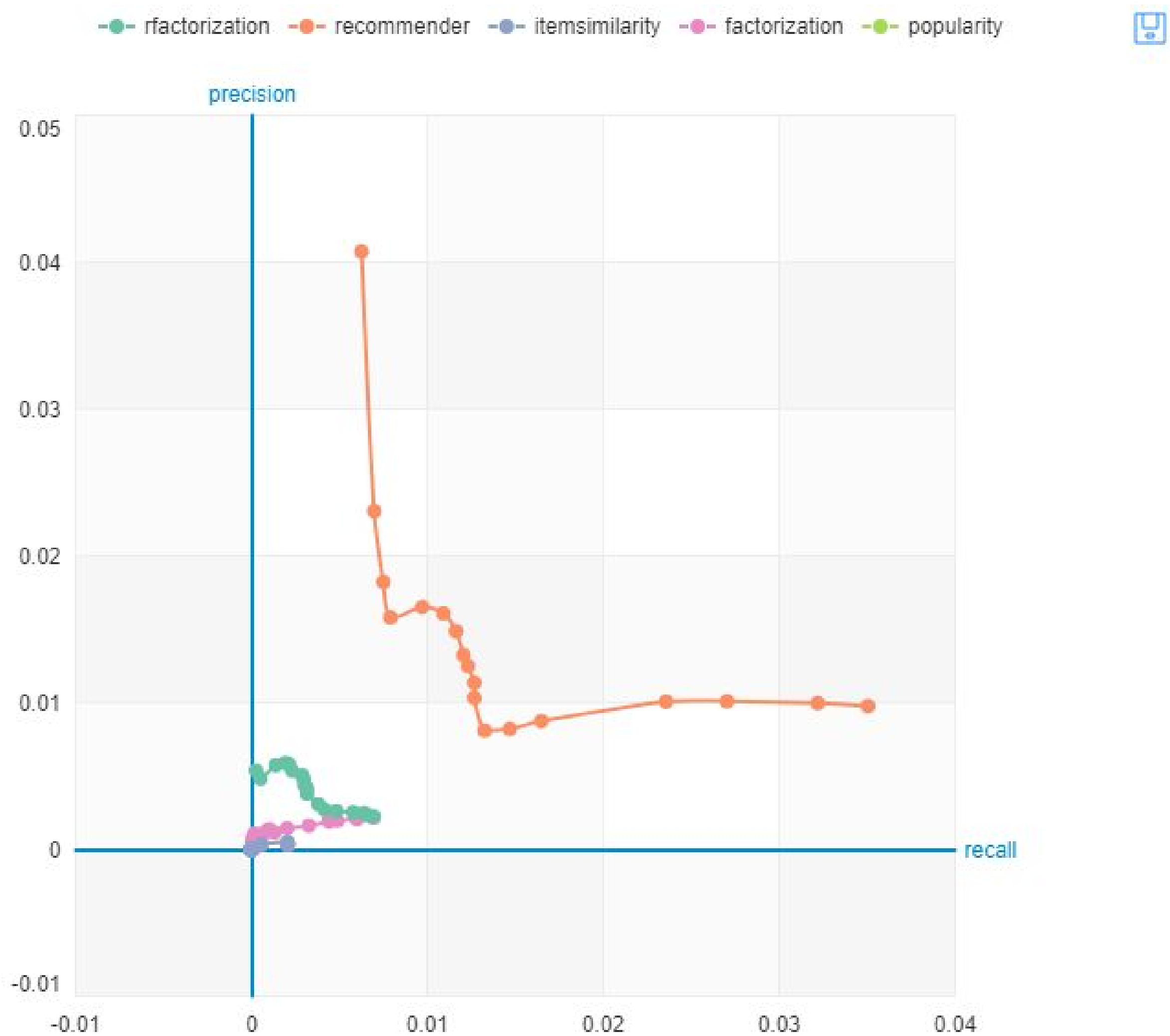
Using protocols and functions available in graphlab we were able to apply recommender create to our SFrame to find out which recommender system model would be suggested, the result was ranking factorization model. This is a collaborative recommender that learns latent factors for each user and item and uses them to rank recommended items according to the likelihood of observing those (user,item) pairs. This result is what we desire as collaborative filtering would be the ideal recommender system for our dataset. We will now test the model performance against other recommender systems and evaluating them with root mean-square error (RMSE) metric. RMSE is the measure of the differences between values (sample and population values) predicted by a model or an estimator and the values actually observed. We will be comparing factorization, ranking factorization, popularity and item similarity recommender systems for our observations. Other metrics that are supportive to our observations include precision and recall. Recall metric reflects the ratio of items that a user likes that were actually recommended. Precision is the metric of recommendation that the user actually liked. RMSE will be prioritized as this is the metric for measuring prediction errors. Ideally a low RMSE would be desired.

Figure #1 RMSE of Models on Original Dataset

Recommender System	**RMSE Overall**
Recommender	195.17391062322085
Factorization	75.30771856256175
Ranking Factorization	158.96039669096675
Popularity	173.85176507763856
Item Similarity	173.712527926137

This table shows the RMSE overall values of the models using the original dataset. We can see that these are not ideal values and that from this observation the factorization recommender system would bet the most ideal model given that is has the lowest RMSE overall value but still has about 75 error rate. We will try normalizing the data with mean and standard deviation to bring this to scale to see if this improves, as it stands this is not an ideal result.

Figure #2 Performance of Models on Original Dataset



From the above figure we can see that recommender is the most ideal precision-recall observation. From recommender we established that the system recommendation was ranking factorization. We can also see that the next best performing model is factorization. We will try observing these results with normalized data to see if there is an observable difference in performance. This does not support our RMSE observations which is understandable given that the performance declines in precision. This gives us further incentive to normalize our data.

Normalization of Data

Our observations from the original dataset showed that normalization of data may be required to scale our data to provide a normal distribution. The RMSE observed was very high across all models but least of all in factorization. On the original dataset the RMSE of ranking factorization model was less than that of the recommender model which indicates why ranking factorization was recommended by the recommender. We can see though that factorization model had the best RMSE granted that the error rate is about 75. Factorization has the same drawback as its strength which is generalization. In factoriation the users and items are represented by a fixed length vector, or a latent factor to provide the dot product of the factors to give a predictive rating. We can divide users (and items) into two groups: those in the training set and those not. Validation scores for the first group correspond to so called *weak generalization*, and for the second to *strong generalization*. In case of weak generalization, each user is in the training set. We take some ratings for training and leave the rest for testing. When assessing strong generalization, a user is either in train or test. We are mainly interested in the strong generalization, because in real life we're recommending items to users not present in the training set. We could deal with this by re-training the model, but this is infeasible for real-time recommendations (unless our model happens to use online learning, meaning that it could be updated with new data as it comes). Our working assumption will be using a pre-trained model without updates, so we need a way to account for previously unseen users.

Figure #3 RMSE of Models on Normalized Data using mean

Recommender System	RMSE Overall
Recommender	196.4236607309218
Factorization	16.294540247234956
Ranking Factorization	214.30602539491747
Popularity	146.80332573452327
Item Similarity	146.71692656629298

This table shows the RMSE overall values of the models using the normalized data with mean as the target variable instead of quantity. Using the mean of the quantities we are able to scale the quantities observations. We will observe this data using standard deviation and then using the coefficient of variance to see which provides the most optimum results. Factorization model RMSE overall was vastly improved to only show an error rate of about 16 in predictive value.

Figure #4 Performance of Models on Normalized Data using mean

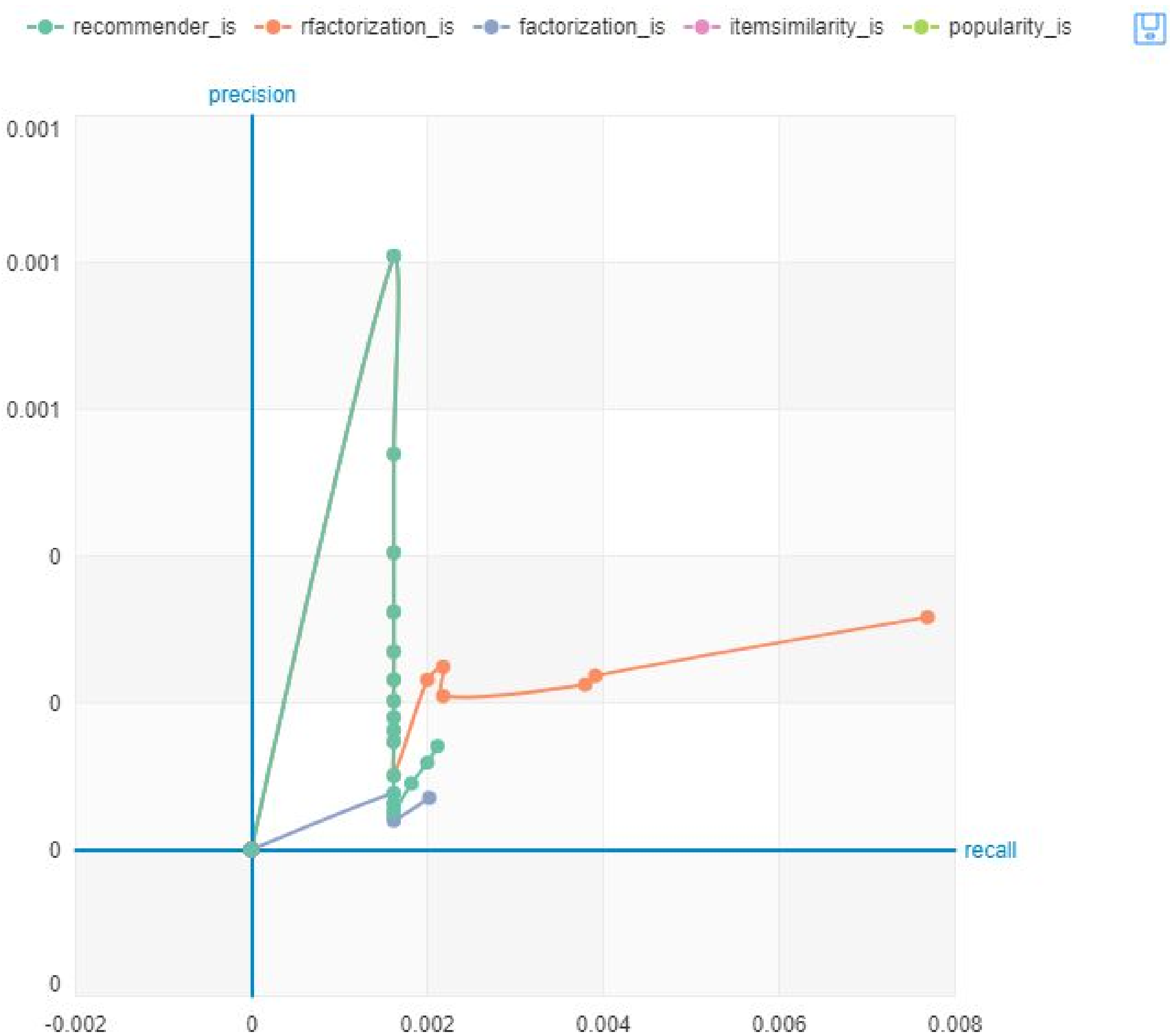
The performance of the models greatly improved by scaling the quantity using mean. Recommender suggested that ranking factorization was the recommended model and we can see that that is indeed the case in this precision-recall performance. This is a vast improvement from figure 2 observations.

Figure #5 RMSE of Models on Normalized Data using standard deviation

Recommender System	RMSE Overall
Recommender	12.194382353991005
Factorization	5.462202312884061
Ranking Factorization	12.568461217770666
Popularity	10.390886973914267
Item Similarity	10.170032002146788

Using the standard deviation we can see that the RMSE is vastly improved across all models. Factorization model still is the most ideal with the lowest error rate on prediction value with error rate of about 5.

Figure #6 Performance of Models on Normalized Data using standard deviation



The performance of the models prediction-recall ability in a graphical representation shows that recommender and ranking factorization are the ideal performers.

Model Selection

After ETL and EDA we continued to scale our data for our observations. We created and implemented training and test sets to test our various models and their performance. The metrics used to measure model performance were RMSE. We then scaled our data to normalize our results and observations using mean, standard deviation and coefficient of variance. Our observations reflected improvement from original data observations which means that our data benefited from scaling for normalization. Ranking factorization was the recommender systems recommended by the graphlab recommender function. Factorization yielded the better RMSE score but on the performance plot for precision recall we saw that ranking factorization outperformed it vastly. Factorization has the better generalization for the model but ranking factorization is the ideal choice for the pairwise relationship of our user-item recommendation.

Conclusion

Evaluation of the models primarily utilized the RMSE metric and the graphical performance of the precision recall. Our results and calculations did provide that the original dataset would benefit from normalization to scale our data. Using mean and standard deviation we were able to achieve this and observe that the models improved in RMSE. We were able to notice that the best RMSE lied with factorization and not ranking factorization, this is due to factorization using a stronger generalization for prediction. Ranking factorization is still the ideal model choice due to the latent factors for each user and item to rank recommended items according to the likelihood of observing those (user,item) pairs. This is still the best option in collaborative filtering purposes. As a recommender system factorization creates a better recommendation due to its generalized approach.

References

Daqing Chen, Sai Liang Sain, and Kun Guo, Data mining for the online retail industry: A case study of RFM model-based customer segmentation using data mining, Journal of Database Marketing and Customer Strategy Management, Vol. 19, No. 3, pp. 197â€“208, 2012 (Published online before print: 27 August 2012. doi: 10.1057/dbm.2012.17).

http://archive.ics.uci.edu/ml/datasets/online+retail