EECS 349 Final Project Report

--Black Friday Prediction

Shufeng Ren & Jiarui Li

1. Introduction and Objective

Retailers want to understand the customer purchase behavior (specifically, purchase amount) against various products of different categories. It's important for them building a model to predict the purchase amount of customer against various products which will help them to create personalized offer for customers against different products.

Our goal is to:

i. predict the customer purchase behavior when the customer demographics (age, gender, marital status, city_category), product details (product_id and product category) are given;

ii. predict purchase solely based on the customer demographics so that they can reflect those in retailer's marketing strategies that target different consumers.

2. Dataset

The dataset comes from a competition hosted by Analytics Vidhya[1] and it contains about 550,068 observations in training dataset and 233599 examples in testing dataset. The training dataset contains 12 attributes, either numerical or categorical. The attributes are as following:

Variable	Definition
User_ID	User ID
Product_ID	Product ID
Gender	Sex of User
Age	Age in bins
Occupation	Occupation (Masked)
City_Category	Category of the City (A,B,C)
Stay_In_Current_City_Years	Number of years stay in current city
Marital_Status	Marital Status
Product_Category_1	Product Category (Masked)
Product_Category_2	Product may belongs to other category also (Masked)
Product_Category_3	Product may belongs to other category also (Masked)
Purchase	Purchase Amount (Target Variable)

The testing dataset has 11 attributes and 'Purchase' is omitted for contest solution submission, therefore, we can only implement different machine learning methods on training dataset. 10-fold-corss validation is implement on all ML model validation.

3. Machine Learning Model(Task1)

1) **Preprocess:** We remap the attributes value from object(string) into integer with two

[1] https://datahack.analyticsvidhya.com/contest/black-friday/

techniques: LabelEncoder (sklearn.reprocessing package) for User_ID, Product_ID, Age, Stay_in_Current_City_Years; one-hot-encoded (pandas.get_dummies) for other nominal categorical attributes.

2) **ML Model a:** Using Collaborative Filtering wrote by ourselves. Firstly, find the Users who bought the same or similar products, then calculate Manhattan/Cosine similarity according to customer demographics and find three most similar examples in training set. Finally, calculating the mean purchase of the three most similar examples.

ML Model b: Using Random Forest in sklearn package. Firstly, using GridSearchCV in sklearn.model_selection to choose suitable estimator number with part of training data. Then using the best parameter to cauculate the average RMSE on the entire training dataset with 10-fold-cross validation.

4. Machine Learning Model(Task2)

(1) **Preprocessing:** Remapping the attributes value from category into integer. For example:

Gender: Female →0 Male →1

Age: 0-17→0 18-25→1 26-35→2 36-45→3 46-50→4 51-55→5 55+→6

City Category: A→2 B→1 C→0

Since we want to use decision tree to predict first, we have to turn the purchase number into category. We need to find dividing criterion from the following figure: 'Purchase' frequency statistics.

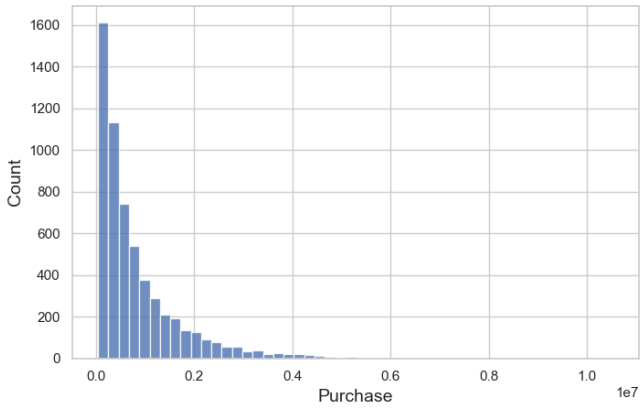

Figure1 'Purchase' Frequency Statistics

The 'Purchase' is remapped as following criterion.

Purchase: [0, 0.1]*1e7→level 4

[0.1, 0.2] *1e7→level 3

[0.2, 0.4] *1e7→level 2

[0.4, +∞]*1e7→level 1

(2) **ML Model:** Using SVM, Decision Tree and KNN in sklearn package only with the low-level information, and 10 fold-cross validation is implemented to assess the algorithm performance. Using SVM with kernel='rbf', gamma='auto', using Decision Tree with

default parameter, using KNN with n_neighbors:3, weights:'uniform', p:manhattan_distance.

5. Results

1) Taks1(Collaborative Filtering)

There is more than 550 thousand examples in training dataset, using all of them cause running-time issue on my machine because it's the generative training. Actually, when implement the code on the entire training dataset, it takes almost 49h. Therefore, only 1/100 of its data (about 5.5k) is randomly sampled for this attempts.

When using Collaborative Filtering with Manhattan similarity and k_neighbor = 3 on the random 1/100 dataset, the average RMSE for predicting Purchase in 10-Folds-Cross validation is 3644.533. When implementing Collaborative Filtering with cosine similarity and k_neighbor = 3 on the random 1/100 dataset, the average RMSE for predicting Purchase in 10-Folds-Cross validation is 3361.723. Following is the result of 10-fold-cross validation with cosine similarity: [3184.618, 4172.318, 3109.259, 3159.112, 3591.761, 4006.161, 2852.621, 3037.879, 3268.775, 3234.729]. In the examples, 34.44% examples consume over 10k and 27.73% examples with less than 6k 'Purchase'. It seems like this model's performance is not good enough to predict different situation even if customer and product information is given. For comparison, Random Forest is also tried to validate the correctness of our model.

2) Task1(Random Forest)

Firstly, selection best 'n_estimators' through 10-fold cross-validation. Setting 'param_grid = {'n_estimators':[1,3,5,10,30,50,100,150,200]}'. From figure2, we find that after Number of trees increasing by100, there is no significant accuracy improvement. To avoid overfitting, we choose Number of Tress=100

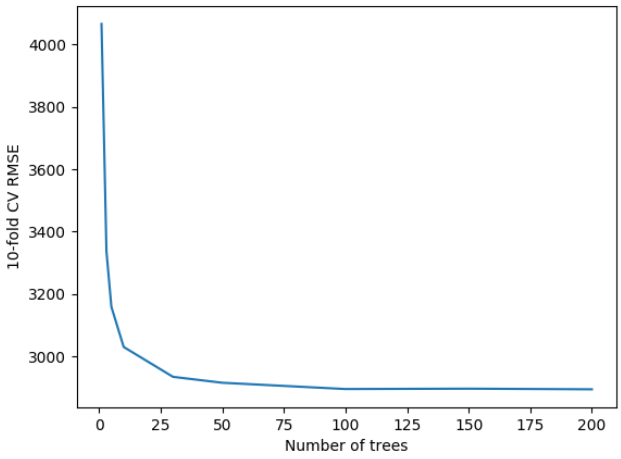

Figure2 Besting Parameter Choosing for Random Forest

When using Random Forest model with n_estimators = 100 on the random 1/100 dataset, the RMSE for predicting Purchase in 10-Folds-Cross validation is 3123.527. When using Random Forest model with n_estimators = 100 on the entire dataset, the RMSE for predicting Purchase in 10-Folds-Cross validation is 2895.407. The result using 1/100dataset is close to Collaborative

Filtering and it shows that the accuracy could improve as the number of samples increasing. Figure3 is the learning curve of Random Forest. It's reasonable to expect lower RMSE of Collaborative Filtering if all training dataset are used.

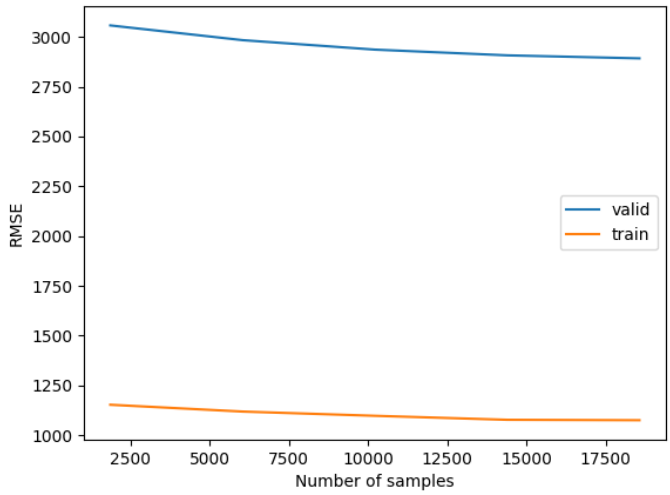

Figure3 Learning Curve of Random Forest

3) Task2(SVM/DT/KNN)

With SVM, Decision Tree, and KNN and 10-fold-cross validation is implemented to assess the algorithm performance. Only customer demographics information is used to predict 'purchase_level' (from level 1 to level 4), and all attributes are transferred to integer type. Following is the results:

	1	2	3	4	5	6	7	8	9	10	AVG
SVM	0.477	0.433	0.513	0.489	0.49	0.503	0.466	0.490	0.491	0.474	0.480
DT	0.385	0.381	0.350	0.330	0.368	0.352	0.387	0.399	0.340	0.346	0.360
KNN	0.338	0.361	0.331	0.333	0.309	0.340	0.362	0.379	0.333	0.320	0.340

It shows that SVM performs best among these three methods. The results disappointed us actually, because lower than 50%(ZeroR) accuracy shows failure to predict the consuming behavior. The failure reason will be analyzed in next part.

6. Conclusion

1) Feature Analysis

In the data preprocess, some interesting relationships among different features are revealed. Figure 4 shows features rank by correlation to "Purchase". The top three feature is: Product_Category_1, Product_ID and User_ID. Therefore, the disappointed result of task 2 could be explained: It's impossible to predict a new customer purchase behavior using features which are weak correlation to 'Purchase'.

What's more, in all customer demographics features, occupation is the most related to Purchase, other attributes such as marital_status/stay_in_current_city_years has no significant correlation to 'Purchase'. Figure 5 is the total "Purchase" by different Ages and figure 6 is total "Purchase" by different occupations, which shows the obvious difference between different group. Figure 7 is the total "Purchase" by different Stay_In_Current_City_Years with less difference between different group.

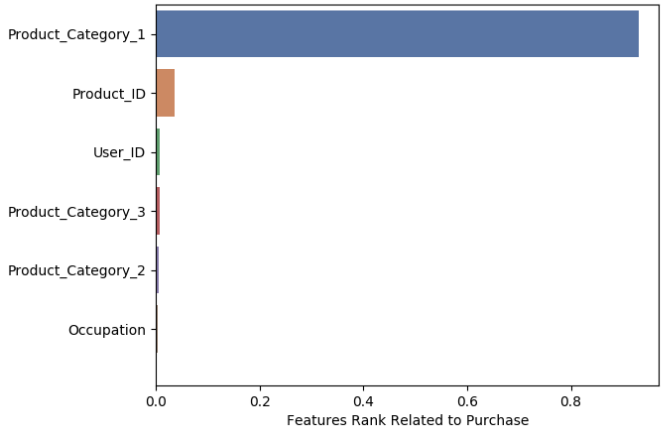

Figure4 Features Rank by correlation to "Purchase"

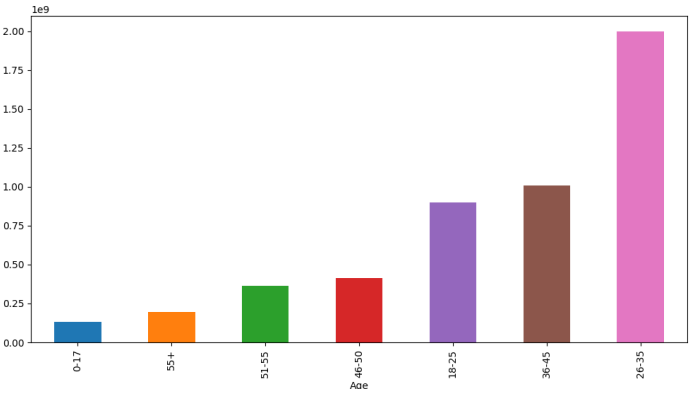

Figure5 Total "Purchase" by different Ages

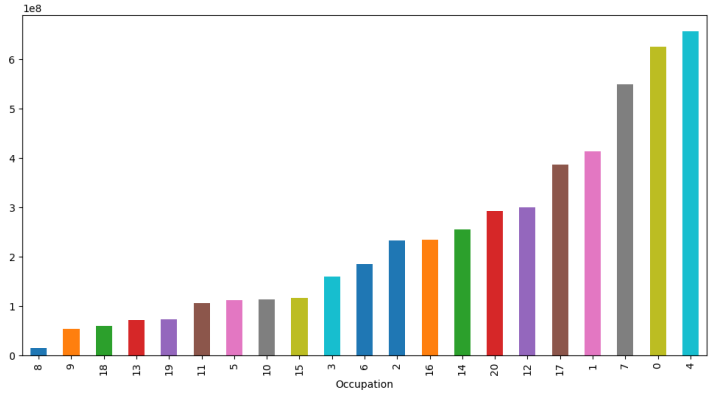

Figure6 Total "Purchase" by different Occupations

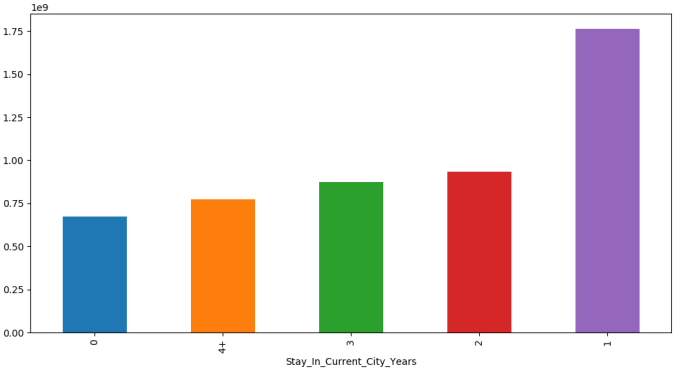

Figure7 Total "Purchase" by different Stay_In_Current_City_Years

2) ML Model Evaluation

For task 1, all attributes are considered to predict Purchase. When implementing Collaborative Filtering wrote by ourselves with cosine similarity and k_neighbor = 3, the average RMSE for predicting Purchase in 10-Folds-Cross validation is 3361.723. However, using Random Forest model with n_estimators = 200, the RMSE for predicting Purchase in 10-Folds-Cross validation is 2932. Considering the code by ourselves is not as efficiency as sklearn package, this results.

For task2, only customer demographics can be used. From features analysis, we know that customer-specific and product-specific information are most related to Purchase, therefore, it's hard to expect good prediction. The highest accuracy with SVM, DT and KNN is 51.3% when predicting Purchase which have been discretized to 1~4.

7. Future Directions

For further optimization of the Collaborative Filtering, more similarity measure should be investigated and parallel computing technique should be added to solve the running time issue. What's more, hyperparameter of Random Tree can be tuned more intensively, like using deeper trees regardless running time. Other ML models like Neural Networks could also be implemented in future work.

8. Member Contributions

Task1 && Website

Shufeng Ren: Implement Collaborative Filtering and Random Forest algorithm and build the website.

Data Preprocess && Task2

Jiarui Li: Complete the data preprocessing and implement three machine learning model for task2.