HANOI UNIVERSITY OF SCIENCE AND TECHNOLOGY
SCHOOL OF INFORMATION AND COMMUNICATION TECHNOLOGY

MACHINE LEARNING - IT3190E

PROJECT: BLACK FRIDAY SALES PREDICTION

Instructors: Assoc Prof. Khoat Than
TA Quang Hieu Pham

Students: Nguyen Quang Duc - 20204876
La Dai Lam - 20204918
Le Hong Duc - 20204874
Luu Trong Nghia - 20204888
Tran Hoang Quoc Anh - 20200044

Ha Noi, July - 2022

Contents

Abstract

Black Friday is obviously one of the busiest shopping day of the year, and to prepare for this, many companies want to make personalized offers to each person to maximize their profit. By using the data from previous sales and different models for machine learning, we will determine the purchase behavior of each customer against various products as accurate as possible.

1 Introduction

During Black Friday, many companies want to increase the sale rate while minimizing the discount to maximize profit. This can be achieved by tailoring advertisements targeted to a specific group of people. Some may only spend small amount of money on some products but are willing to pay more for other kinds. For example, cosmetics will be more profitable to advertise to women at the age of 20-40, computer gears is better to market to men at the age of 20-30.

It is supposed that our contractor "Co. DS-AI" wants to understand the customer purchase behavior, specifically the purchases amount of each customer against various products. Moreover, they provide us the datasets (which is available from Black Friday Sales Prediction[1] competition) to analyse and finally give them the appropriate solutions. In this dataset, we are basically provided with the Customer Demographic, Product Details and the total Purchase Amount from previous months.

Given the problem and the datasets, in order to solve the above problem, we utilized many techniques about data preparation and machine learning models to find the optimal and effective solution. Speaking of tools, we chose Scikit-learn,[2] Pandas[3] and Seaborn,[4] etc. to analyse the data, visualise graphs and build our machine models faster and easier.

In the report, we will present the process we tackled the aforementioned problem step by step. Firstly we defined the issue more carefully. Secondly, we explored the dataset and found interesting patterns and results. Finally, machine learning models were built and why we chose them would be presented as well as methods and techniques to increase the efficiency of each model overtime such as cross-validation, learning curve and so on.

2 Problem formulation

The first step in any project, especially machine learning project, is correctly define the problem we need to solve in order to choose a correct strategy and suitable techniques and models.

Our problem is a supervised learning problem as we need to know about the previous purchases of the products regard to each costumer in order to predict the price in the future.

We define the formula that can be described as a triple (T, E, P):

- Task (T) : Predict the suitable purchase of each product for each customer.
- Experience (E) : A list of the previous purchases behaviour of each customer against different products.
- Performance (P) : The loss error of the prediction for each observation, which is the measures for the difference between real purchase and predicted purchase.

With the motivation of exploring the methods to process data and the models applied to this situation, we tried out best to find the best answer for the this problem. Moreover, we think our models could help some people choose the suitable items that they can buy products on an everyday basis, specially on the Black Friday, as well as help some shops can improve their profits by discounting some specific products at the profitable price.

3 Data preprocessing

3.1 Datasets

The original data was obtained from the Black Friday Sales Prediction competition.[1] A company wants to understand the customer purchase behavior against various products of different categories. They shared purchase summaries of various customers for selected high volume products. Basically, there

are mainly two different data sets: the training set with train labels and test set. In this project, we are only working with the training set and making prediction on the test set. With a given training dataset, competitors are wanted to build a predictive model and apply it to test set to predict the purchase amount of customer against various products. The data set used contains customers demographics, product details and total purchase amount. The data set has these following attributes:

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 (Marked)
Product_Category_2	Product may belongs to other category also (Marked)
Product_Category_3	Product may belongs to other category also (Marked)
Purchase	Purchase Amount (Target Variable)

Table 1: Attributes Description

There are 550068 rows of data in the training set. All features are categorical , whereas, the target attribute (Purchase) is continuous. In the data set, every attribute is complete except *Product_Category_2* and *Product_Category_3*, with 173638 and 383247 null data cells, respectively. We will take a closer look at the data in the next section.

3.2 Data Exploration and Visualization

Firstly, let's explore about the customer demographics. There are 5891 distinct customers in total who have purchases at least a product. We wonder whether where is any connection between information and the amount of money that customer spent on products?

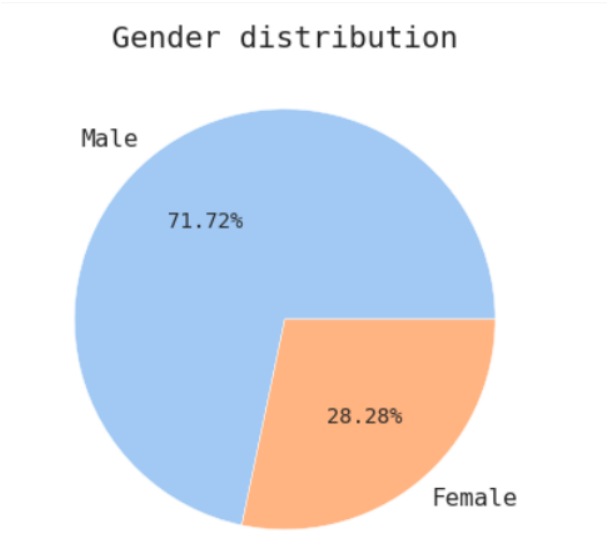

Figure 1: Gender distribution

The number of male customers is 4225, accounts for 75.31% of all customers; this number is approximately 2.5 times higher than that of female customers (1666). Customers come from 3 different cities, it could be a reason why the shopping behaviour of customers are different. The figure shows that number of customers from city A is the least and city C has the most customers where number of users

in this city is over 3 times higher than number of users in city A and as twice as many that figure in city B. Surprisingly, the total amount that customers in city A have purchased is more than 2.5 times higher than that of customers in the city C and the amount of money which users in city B have spent is just under that number from city A. We have an assumption that city A is a well-developed city, where its citizen are rich.

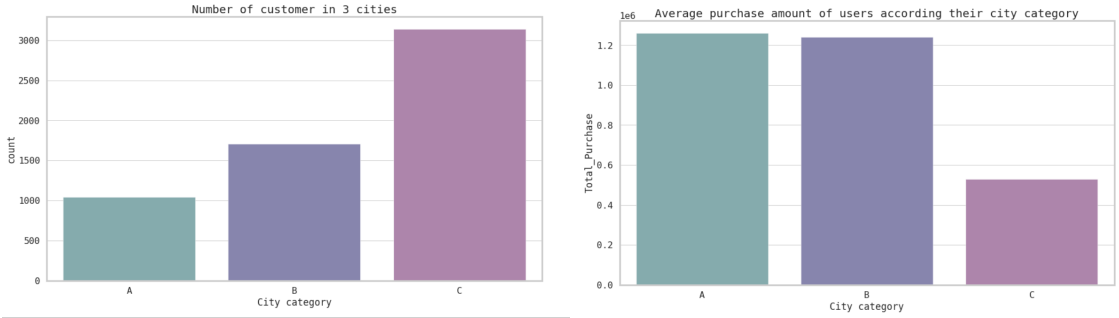

Figure 2: City distribution

The next attribute of customers is their age. There are 7 age groups of customers; the number of users of each group has a significant difference. In general, there are customers with young population age, where customers form 26-35 years old are predominate in number. Group 0-17 years old is the age group which have least users. In the bar chart represent total amount of money for each group of age, 4 groups from 18 to 55 spent most money in total. 0-17 and over 55 years old are 2 groups which have lowest purchase amount, the reason could be from their income. Customers from group aged 26-35 has a larger consumption than the rest, nearly $1,000,000. We guess that at this age, some users do not know how to spend properly and they are willing to spend a lot of money to buy products.

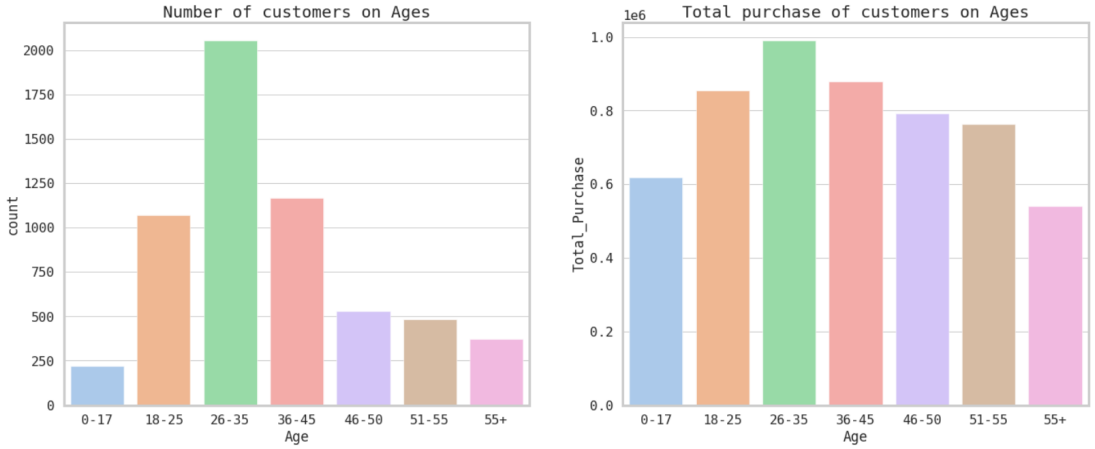

Figure 3: Age distribution

The next thing we consider is about customers occupation. The occupation of customers have been marked from 0 to 20, with no knowledge detail. We use bar chart to count and compare the number of users in each job and total purchase amount according to their job. It is clearly that numbers of people who have job 0, 4, 7 are taking the lead, whereas, it seems that job 8, 18, 19 are some of special job because there are very few people working in that field. Despite of large number working in the field, three jobs 0, 4, 7 have average values of total purchase amount, these could be common job which staffs have 'normal' income.

The time of living in a city of customers is also one of the information we need to understand. The data set divided time of living of a customer in current city into 4 groups: under a year with label 0, from 1 to under 2 years with label 1 (similarly with label 2 and label 3), and over 4 years with label 4+. It is hard to understand this field without any combination to other customers' information. From figure Figure 5 , number of customers who live 1 year in city C is surprisingly high, while the rests do not have

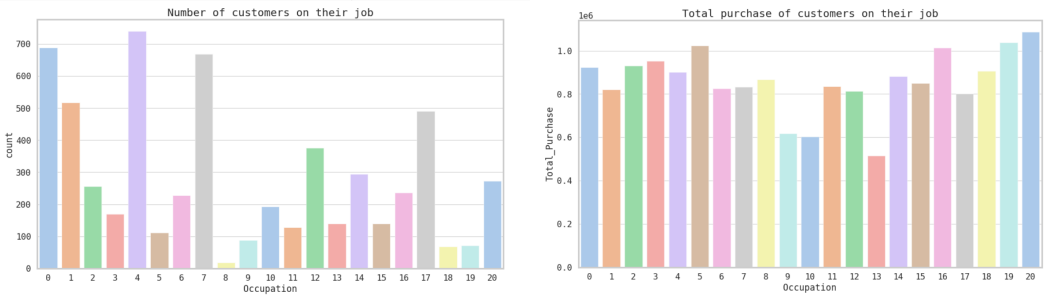

Figure 4: Occupation distribution

many things to analyze. One thing we know from figure Figure 5 that newcomers who moved to city A less than 1 year spent more on company products. Moreover, people who have lived more than 3 years in city B tend to spend more than such people in other cities.

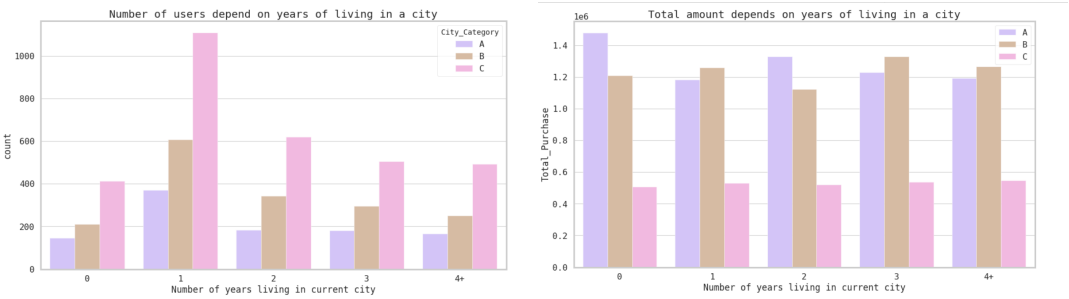

Figure 5: Number of years living in current city distribution

58% of customers are unmarried. It is easy to understand why group 0-17 years old all unmarried. Under 45 years old, in each group of age, number of users who have not married or have divorced is higher than that of users who get married. Over 45 years old, users who already get married are many than that number who unmarried in all group of age. We also wonder about whether marital status affects to user purchase behaviours, but the result from bar chart in Figure 6 do not give us many information.

We have already introduced and visualized user's attributes. Now, we take a close look in relationship

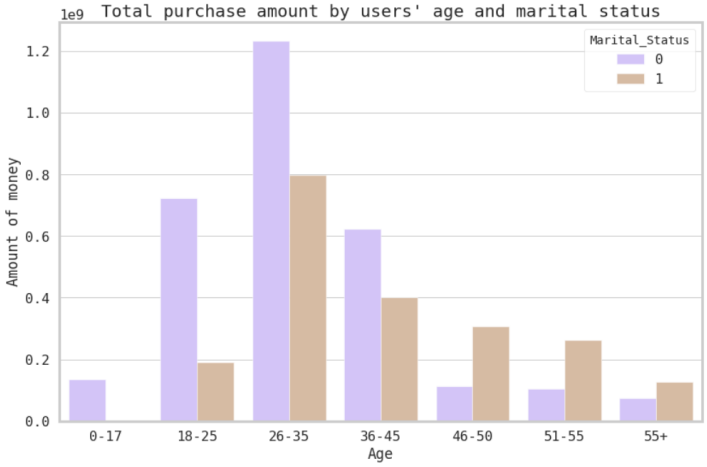

Figure 6: Total marital age

between attributes. From Figure 7, it is hard to see any data point that belongs to users in city C which have more than 200 products purchased. In general, a normal user paid for lower than 400 products, there are only some of users in city A buy more and therefore total money these customers spent for products also the highest.

Figure 7: Relationship between number of products and spending money of users

The three distribution plots shows us some helpful information. The distribution of the money of a user spend for a product on average (total money an user spent over number of distinct products he purchased) has a bell-shape. This distribution is not Normal distribution, the p_value is approximately 0. That is all we found out about customers profile. We still need to understand the data set more.

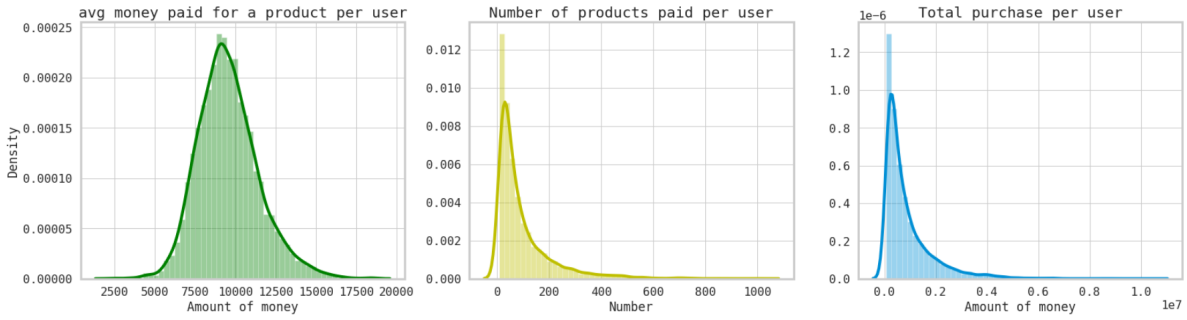

Figure 8: Distribution of average money, number of products, total money purchase by users

The second thing after user profile we analyze, it is about product details. A product detail contains product id, product category 1, product category 2, product category 3. A product belongs to at least one category that fill into product category 1. Product category 2 and product category 3 are optional, that means some of products do not filled these fields. Since category 2 and category 3 have a large number of null values, we assume that product category 1 is the main category of a product and concentrate on analyze this field for understanding more about products. Similar to user occupation, category 1 is also marked. We just know from bar chart Figure 9 that there are 2 distinct products belongs to category id 9 or category id 19. We take a guess that 9, 19, and 20 are 3 particular categories without competition from suppliers, or the company may not offer many models of these types. There are 1047 kind of products in category 8, the second rank belongs to category 5 with 967 different products.

The category which is the third highest number of kind of products is category 1. From bar chart represent total amount of money spent according categories, category 1 is far superior to the rest, with over$1,910,000,000 of products have been purchased. The majority of categories have total consumption under $200,000,000. Although category 8 has lots of types of products, the total amount for this category is lower than $\frac{1}{2}$ of that for category 1. However, we cannot eliminate product category 2 and product category 3 just because of lack of information. We consider a versatile score which involves in all 3 product category fields with an assumption that a product is more versatile if it belongs to more categories.

Clearly, number of products which belong to only 1 category is majority, over 50%. Percentage of very versatile product is 14.54%, the lowest of 3 types. However, users spent most for products which very versatile, the evidence is showed. The total amount paid for the most multi-use products (very versatile) is significantly higher than the other 2 groups, and more than 2 times that of products versatile type 2, which ranks second. We also plot the distribution of average amount of money paid for a product (total amount for a specific product over number of users purchased that product), The distribution may

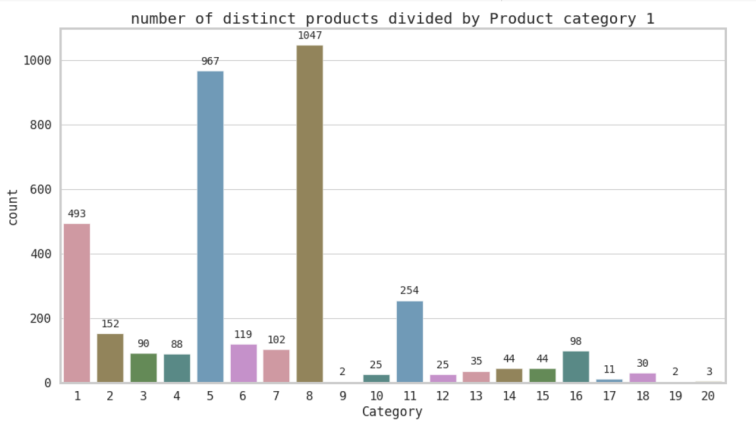

Figure 9: Number of distinct products by Product Category 1

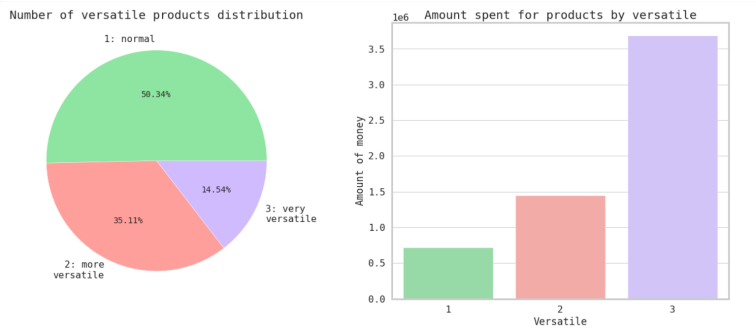

Figure 10: Versatile Score Distribution

be the combination of 2 normal distribution. There are 2 peaks, the highest density is about $6000 and the second peak is about 13500. We will use this distribution and the versatile score as 2 new derived attributes involved in products to apply to our model in next section.

4 Machine Learning models

In this section, we will reason why we selected each following model and introduce theories of these models which hopefully can help us to overcome this hard-core problem. We will start from very simple model like Simple Linear Regression then go through many more complex models like Decision Tree, K-Nearest Neighbors, etc. and finally Ensemble Learning where we actually compose different models to have a better result.

4.1 Simple Regression - a baseline model

A baseline model is essentially a simple model that acts as a reference in a machine learning project.[5] We simply implement a baseline model for serving as a benchmark for other complex models.

4.1.1 Ordinary Least Square Linear Regression (OLS)

Indeed, we started by choosing Ordinary Least Square Linear Regression as our baseline model because of its simplicity and easy implementation. This also is used in statistics to compare the dependent variable to the independent variables so we also could compare the efficiency of different data preprocessing processes. Moreover, there are also several advantages and disadvantages we need to take into account when choosing this model.

Advantages:

- The complexity is minimal in comparison with other regression models.

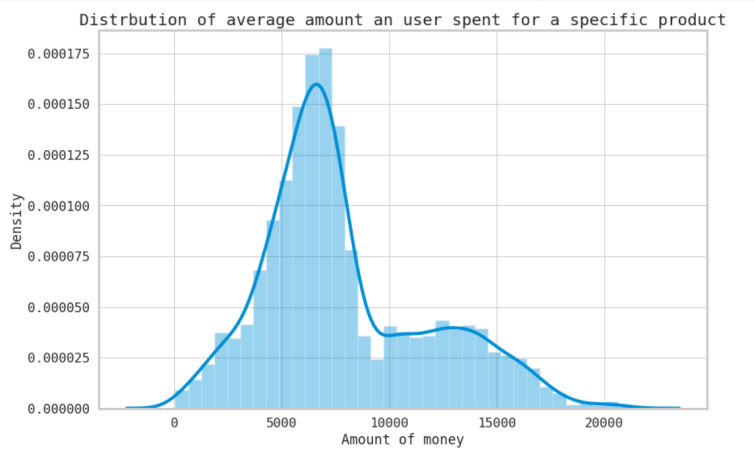

Figure 11: Distribution of amount of a specific product purchased

- The model is sensitive to outliers so we can observe the performance of the data preprocessing step (so we can compare different data preprocessing techniques).

Disadvantages:

- The model does not work in some cases as it depends on the singularity of the matrix obtaining from the train set.
- The model tend to be overfitting if the data is generally good. However in case the data is bad, underfitting may occur.

4.1.2 Polynomial Regression

Linear regression in the above section seemed not working too effectively actually underfitting so we come up with using polynomial regression which own better expression of the relationship between independent variables and dependent variable in order to increase the dimension of the features. There are also several benefits and drawbacks of this model.

Advantages:

- The algorithm can fit a wide range of curvature which is beneficial for increasing performance and could avoid underfitting.

Disadvantages:

- The computation time comparing to linear regression is much slower.
- We have to choose good degree for the algorithm while concentrate on the bias-variance tradeoff.

4.2 Decision Tree Learning

Decision tree is very powerful tools for both regression and classification problems. This algorithm can use set of if-else rule to handle well for data that contain all categorical features like this problem. Moreover, decision tree is the base algorithm for other advanced decision-making and ensemble learning techniques.

Decision tree is the function having tree structure that each path from root to leaf representing a decision. Each branch in the tree show the value of the attributes corresponding to the branch staring node.

Advantages:

- Decision tree does not require excellent data preparation to have high performance
- Decision tree have great interpretation, it can be visualized for better intuition.

Disadvantages:

- Since decision tree can be seen as a piecewise approximations tree, this algorithm may overfit the training set if the data is reliable, otherwise it will suffer from underfitting .
- The performance of decision tree relies on the choice of its hyper-parameters.

4.3 Ensemble Learning

4.3.1 Random Forest

As our decision tree model after hyperparameter tuning suffers from overfitting so we think about using Random Forest model. It is also the first technique we want to introduce about ensemble learning which is bagging as it could reduce the variance of the previous model and one of the most popular application .

Random Forest is constructed from a variety of decision trees which has its attributes randomly picked and is built from boostrap sampling. The prediction of random forest is the average of all tree prediction so it will

Advantages:

- Random forest can avoid the overfiting case of decision tree since it take the average prediction of all tree. Moreover, the result will be more accurate comparing to that of only one tree since its variance is lower.

Disadvantages:

- Random forest use many decision tree therefore the algorithm will be more time-consuming.
- Having similar problem like decision tree, random forest has many hyper-parameters to adjust so making the right selection for them may result in the model effectiveness.

4.3.2 XGBoost

However, the Random Forest is still suffer from overfitting as it reduces its variance but not the bias so we came up the idea about using boosting model.

In ensemble learning, bagging can deal with learners that have high variance and low bias while boosting is remarkably effective in case of high bias and low variance learners. Bagging has all the learners be trained simultaneously then combine their result so in case all the individual results are poor, the overall will be terrible too. On the other hand, boosting can easily overcome this weakness by training all learner sequentially where current leaner try to be more useful than prior ones by learning from their mistakes.

When it comes to boosting, there are two approaches which are Adaptive Boosting (AdaBoost) and Arcing. In comparison with AdaBoost, Arcing is more flexible as its ability to provide appropriate solution for general loss function. Indeed, XGBoost, an outstanding implementation of Gradient Boosting, had showed its effectiveness by being one of the most credible and powerful tool of lots of ML developers for solving a wide range of problems.

Advantages:

- XGBoost can utilize the power of the machine by using parallel processing which can enhance its computation time.

Disadvantages:

- Generally, XGBoost training time is quite long in comparison with other boosting algorithms
- XGBoost is sensistive to outlier because the every next learners will always try to correct the shortcoming of its predecessors.

4.4 K-Nearest Neighbors

In this problem, we worked with datasets that contain many users and products so instanced-based learning is quite adequate suggestion.

Instanced-base learning will predict the target value base on the similarity of new instance to the existing ones.

Advantages:

- The KNN algorithm is quite flexible as there is various ways to measure the distance
- The KNN algorithm can learn non-linear boundaries which is suitable for solving many problems.

Disadvantages:

- The prediction time of KNN is considerably long when the size and dimension of the data set is large.
- The KNN algorithm is sensitive to outliers hence it require great data preprocessing.

4.5 Collaborative Filtering

In this problem, our final target is understanding the customer behavior by learning and predicting the purchase for one particular product. However, in this section, we will assume that the purchase amount is the price of just one product unit . The cost of the product can be seen as a way of grading the product because if a customer assess a product to be great, he/she will tend to buy it with higher price. Base on this idea, we attempt to use recommendation system for this problem and the prices can be inferred from the ratings. Since the categories of the product is missing and the preprocessing step do not ensure to give accurate value to classify the product, we will use collaborative filtering instead of content-based algorithms.

The idea and implementation are inspired from lesson of well known machine learning expert - Mr.Andrew Ng. Comparing to what he said in that lecture, we tried to estimate proper parameters for each product corresponding to the attributes of a user like age, job, etc.

Advantages:

- We do not need domain knowledge of the data since the model will learn it automatically.

Disadvantages:

- The model may suffer from cold-start problem which is when it tries to predict for new items.
- For products that having small rating .i.e not popular, the system can not recommend them but the more common ones.

5 Experiments

After our explanation about data exploration and visualization and the learning models we chose to predict the target variable (purchase), in this section, we will discuss about the experimental process we used to improve the result and chose the best model for this solution. The section also discusses the reasons why we used several techniques in our source code on the Github repositories. We also noted everything we could in the Jupyter notebook file as our profound and more clear explanation.

Overall, after preprocessing the datasets, we chose one suitable evaluation metrics to evaluate the performances of this regression problem, then we will choose more complex (or simpler) model based on each previous selected model until find an acceptable model with various following techniques.

5.1 Evaluation metrics selection

Evaluation metrics selection is an important step as we need a value or a number to compare between different models.

Among different evaluation metrics for regression problem such as R Square, MSE, MAE, etc., we decided to choose root-mean-square error measure (RMSE)[6] for our problem. The formula of RMSE is

$$RMSE = \sqrt{\frac{\sum_i^N ||y_i - \hat{y}_i||^2}{N}}$$

At first we wanted to use R Square metric as it gives us a concrete value between 0 and 1 (just like accuracy score in classification problem), however it only suitable for linear models and not for non-linear models (such as Decision Tree, Random Forest, etc.) Therefore, at the end, compared to MAE, we chose RMSE as it has benefit of penalizing large errors (the more purchase is far from the real one, the more it is penalized).

5.2 Model assessment

In order to achieve a better model, we also utilized two evaluation techniques: K-fold cross-validation and hold-out. In following subsections, we will explain why and how we applied this to each machine learning model.

5.2.1 K-fold cross-validation

In this problem, we mostly used 5-fold cross-validation for evaluating each model parameter and selecting the best set of parameters by searching a pre-defined finite sets.

For example, for the Decision Tree algorithm, we chose the best-performance hyperparameter by using GridSearchCV (exhaustive search) as you could find in the source code.

However, it is time-consuming so for instance-based models or high time complexity models, we opted to use the second technique which is hold-out.

5.2.2 Hold-out (random splitting)

We used popular split $|D_{train}| = \frac{2}{3}|D|$ which is suitable for our dataset size.

5.3 Learning curve - the way we identify overfitting and underfitting models

After training and testing a new model, we always concern whether the model is underfitting or overfitting. One of the most common way to detect this problem is using learning curve.[7] By giving intuitive graph, developers are able to recognize the change in model performance when adding more data examples, which is important for their inference about the model.

After each model selection, we always drew a learning curve to detect whether the model is underfitting or overfitting then decided to choose what model we should use next.

5.4 Results and our explanation

After applying all techniques and trying different, here are the summary of results of different models with the best parameters settings we have tried. In this table, settings column is the best performance parameter we have tried.

	Model	RMSE_mean	RMSE_std	Training time(s)	Prediction time(s)	Parameters	Evaluation technique
0	OLS Linear Regression	4614.7	8.76	0.172	0.015		5-fold cross-validation
1	Polynomial Regression	4163.05	10.74	15.318	0.068	degree = 3	5-fold cross-validation
2	Decision Tree	2733.33	7.85	2.879	0.038	max_depth = 15, min_sample_leaf = 20	5-fold cross-validation
3	Random Forest	2738.68	6.34	71.483	4.137	n_estimators= 100, min_samples_split= 5, max_features= sqrt	5-fold cross-validation
4	XGBoost	2618.79	10.3	84.291	1.143	n_estimators= 200, min_samples_split= 9, subsample = 0.8	5-fold cross-validation
5	K-Nearest Neighbors	2995.14		0.395	1530.0	k = 15	Hold-old (2:1)
6	Collaborative Filtering	4311.77		67.713	338.071		Hold-out (2:1)

Figure 12: Result table of different models

Because of the time complexity of K-Nearest Neighbors based on the number of instances in the dataset (which is half of a million) so it takes time to predict so we just chose a simple hold-old technique in order to save time. Moreover, the reason we decided the hold-old technique for collaborative filtering as we built this model from scratch, which is not efficient (high running time at both training time and prediction time as in the Figure 12).

The OLS Linear Regression error was the highest one but it was also the algorithm having the fastest training time. Because of its simplicity, it was not supposed to have an excellent model for every complex problem. It showed that using non-linear algorithms can improve the performance since the rest model have smaller error than that of the linear model. We believe it was caused by choosing an inadequate degree for the model so our function approximation was not close to the general function.

Working with models that have more hyper-parameters, it took lots of time for us to find the proper set of them but the result was well-deserved. There was a fall from above 4000 to below 2700 when using Decision Tree, Random Forest, and XGBoost. In comparison with the Decision Tree, the Random Forest successfully decreased the standard deviation as it consists of numerous trees. As we expect, XGBoost was the most efficient model with the smallest error.

The KNN algorithm had the longest overall processing time as it had to do lots of computation in predicting phase. Though the collaborative filtering algorithm manually designed by us was quite simple which resulted in having a long predicting time, its performance is still better than other simple models like linear. In problems that involved customers and products, with better techniques, we believed that the collaborative filtering algorithm was still considerable and stood a chance to beat other ones.

6 Conclusion

So far, we have introduced and solved our prediction problem by different techniques in data analysis and machine learning models. The best model, in general, is the XGBoost model which is also the classic boosting algorithm in many competitions whose result is nearly twice as high as that of our baseline model.

From the project, it is the first time we have done a machine learning project on such a scale so it taught us many lessons. Firstly, we need to really interpret the problem such as the main demand of this in order not to answer the wrong question. Secondly, for the data mining process, we need to fully understand each feature (especially the target feature) as at first, we did not correctly understand it. Finally, model selection and parameter tuning can have a tremendous effect on the result of our solution.

The most challenging step when we solved the problem is when we needed to choose what strategic process to choose the suitable data preprocessing techniques and reasonable machine learning models in order to divide the workload between each group member. And finally, we decided to follow an aforementioned process.

As we can see from the table of results of different models, ensemble learning, especially boosting algorithms yield much better results in term of errors for this type of problem. Therefore, if we have more time on this project, we definitely will research some boosting models such as LightGBM,[8] CatBoost,[9] etc. as well as try out different new techniques in data preprocessing for the given dataset.

References

[1] Vidhya A. Black Friday Sales Prediction;. Available from: https://datahack.analyticsvidhya.com/contest/black-friday/.

[2] Scikit-learn;. Available from: https://scikit-learn.org/stable/.

[3] Pandas;. Available from: https://pandas.pydata.org/.

[4] Seaborn; 2021. Available from: https://seaborn.pydata.org/.

[5] Nair A. Baseline Models: Your Guide For Model Building;. Available from: https://towardsdatascience.com/baseline-models-your-guide-for-model-building-1ec3aa244b8d.

[6] Root-mean-square deviation; 2021. Available from: https://en.wikipedia.org/wiki/Root-mean-square_deviation.

[7] Muralidhar K. Learning Curve to identify Overfitting and Underfitting in Machine Learning; 2021. Available from: https://towardsdatascience.com/learning-curve-to-identify-overfitting-underfitting-problems-133177f38df5.

[8] Light Gradient Boosting Machine; 2022. Available from: https://github.com/microsoft/LightGBM.

[9] Gulin A. CatBoost; 2022. Available from: https://catboost.ai/.