MAT 555 PROJECT

Burak Çakan

A study with the real data of trendyol.com

Table of Contents

INTRODUCTION

Throughout this project, a joint dataset of sales-transactions, customer information and product information retrieved from trendyol.com, which is the leading e-commerce company in Turkey, was used. As a time-interval, first 4 months of 2021 are selected for the dataset. The aim is to predict some features of the customers like saving credit card, gender and elite member flag. At first, the only aim was to just predict saved card information, which tells the company whether customers trust the company or not, because it is one of the most cared KPI for the company. In business, ascending of the ratio of the customers who saved credit card is an important indicator for growing of the company and the profit, and vice versa is also valid. If the model fails in predicting saved card flag, other features, gender and elite member flag, will be used in the model as the target variable instead. To clarify, elite member flag is a kind of loyalty badge that if the customer satisfies some conditions like spending more than 4000 TL in the last three months and using Trendyol Wallet at least in one purchase. Furthermore, these elite members have some privileges like free shipping for all products and having an exclusive customer service.

DATASET

The company, trendyol.com, uses Google Cloud to store its data. Hence, I choose Google Colab for this study. At the company side, Google Bigquery SQL Machine is used to process the data. I have used it to bring the needed information together for the study from more than 10 different tables in the cloud. Then, I used "google.colab" library in order to authenticate entering cloud with "auth" package and to run the SQL code on python script to transport dataset from cloud to python with the help of "bigquery" package.

As it is mentioned in Introduction, time range of the dataset is from 2021-01-02 to 2021-04-15. Because the size of the full data is more than 1 TB that it is far beyond the limit of processing and transferring in Python in this range, I implemented a methodology to sample the data cleverly. This methodology is as follows:

- Using transaction records of only the customers having more than 50 distinct orders in the specified time range above.

- In each order, transaction of all the products are not included, only the most expensive product is taken into consideration. (When I implement just the first condition and take all products in every order, size of the data is more than 50 GB.)

After implementing these two conditions on cloud environment, I have reached the final dataset to use in this study. Size of this dataset is 676.58 MB having the number of instances of 6,959,857 and 11 attributes. The attributes and their description are explained below.

a. Attributes

Attributes in this study are order_number, order_date, Product_Category, Business_Main_Category, order_price, customer_number, customer_gender, customer_age, elite_member_flag, saved_card_flag and BRAND_NAME. Basically, there are three types of attributes which are categorical, discrete and categorical. A **categorical variable** is a variable that can take on one of a limited, and usually fixed, number of possible values, assigning each individual or other unit of observation to a particular group or nominal category on the basis of some qualitative property (Yates et. al., 2003). Gender can be a good example for categorical variables. **Discrete variables** are numeric variables that have a countable number of values between any two values (Minitab, 2019). In other words, a variable formed by counting a set can be called as discrete variable. Discrete variables should be finite and numeric. Lastly, **Continuous variables** are numeric variables that have an infinite number of values between any two values (Minitab, 2019). Date of transactions can be a good example for continuous variables.

In the light of this information, categorical attributes of this study are customer_gender, Product_Category, Business_Main_Category and BRAND_NAME, discrete attributes are order_number, customer_number, customer_age, elite_member_flag and saved_card_flag, and continuous variables are like order_price and order_date. Descriptions of these attributes are explained below:

- ***order_number:*** It is the unique identification number for each order. 1 is assigned for the first order and increases one by one with each order. This number hashed with FingerPrint Algorithm on BigQuery (Tigani and Naidu, 2014) for the information security policy of the company. Data type of this attribute is integer and there are 6959857 distinct order_number in the dataset.

- ***order_date:*** It shows the date of the order purchased by the customer. Data type of this attribute is object. The date range is between 2021-01-02 and 2021-04-15.

- ***Product_Category:*** It is the category of the product. It is like a subcategory of Business_Main_Category. For example, Product_Category of Toblerone is Chocolate. There are 2773 distinct product categories. Some of them are like “Çocuk Öykü”, ”Pantolon”, ”Sweatshirt” etc. Data type of this attribute is object.

- ***BRAND_NAME:*** It shows the brand of the purchased product. In the chocolate example, Toblerone is the brand name. There are 63808 distinct brands in the dataset. Data type of this attribute is object.

- ***Business_Main_Category:*** It is the main category of the purchased products. There are 17 distinct categories. They are FMCG, Branded Tekstil, UNKNOWN, Kozmetik, Ev, Private Label, Aksesuar & Saat & Gözlük, Çocuk, GM, GAS, Ayakkabı & Çanta, Consumer Electronics, Mobilya, Home Electronics, Digital Goods, Aksesuar & Lüks, Kadın Tekstil. In the chocolate example, FMCG is the business main category for Toblerone. Data type of this attribute is object.

- ***customer_number:*** It is the unique identification number for each customer. This number hashed with FingerPrint Algorithm on BigQuery (Tigani and Naidu, 2014) for the information security policy of the company. There are 83834 distinct customer numbers in the dataset. Data type of this attribute is integer.

- ***customer_gender:*** It shows the gender of the customer. It is “Bay” or “Bayan” in the dataset. Data type of this attribute is object.

- ***customer_age:*** It shows the age of the customer. It is calculated with the birth date information of the customer that the customer can enter any birth date. It ranges between -47 and 2020 that means it is not a reliable attribute. Data type of this attribute is object.

- ***elite_member_flag:*** It is the elite status of the customer. As it is mentioned in Introduction part, it shows the loyalty of the customer to the company. There are 2 distinct values which are 1 for elite member and 0 for not elite member. Data type of this attribute is integer.

- ***saved_card_flag:*** It shows whether the customer has saved his/her credit card information to the site. There are 2 distinct values which are 1 for saved credit card and 0 for not saved credit card. Data type of this attribute is integer.
- ***order_price:*** It is the total price paid for the specified product in the specified order_number. Total price paid for each product is nearly 1,5x10^9 TL. Minimum value is 0.1 TL and the maximum value is 472991.2 TL. Data type of this attribute is float.

b. Data Cleansing

I have implemented some data cleansing operations to have a reliable dataset. Importantly, three fundamental problem is observed by investigating the dataset. First one is the unreliable values like negative and more than 100 of customer_age attribute that I included it in preliminary models, then I discarded it from train and test splits for the final models. Second one is that customer_gender attribute has a value named as “Unknown” and values except “Bay” and “Bayan” are not considerable for this attribute for the quality of the model. Lastly, I eliminated some instances whose brand name has a value of “Null”. I developed main and final models after implementing these cleansing operations and the data get clean afterwards.

EXPERIMENTS

As clearly seen in Google Colab repository for this study, I have many attempts in order to reach a final model and a good prediction. Therefore, I have arranged all experiments into three groups in order to explain better. These groups are preliminary phase, progress phase and the final phase. According to the requirement and progress of each phase, different classifiers, evaluation metrics and sample size. In all of the models, 25 % of the 4-months data are used as test and the others are used as train split.

a. Preliminary Phase

Preliminary phase is the phase of initial trials in order to understand the behavior of the features by trying smaller models. In the scope of this phase, any evaluation metrics like accuracy are not used. Decision tree classifier is selected because it can handle both categorical and continuous data with a good performance at all. Patel and Prajavati (2018) emphasized the advantages of this method by saying the ability to selecting the most biased feature and comprehensibility nature, ease of classifying and interpreting results easily and the usage for both continuous and discrete data sets. In more detail, decision tree algorithms are used to split the attributes to test at any node to determine whether splitting is “Best” in individual classes (Patel and Prajavati,2018). The resulting partitioned at each branch is PURE as possible, for that splitting criteria must be identical (Patel and Prajavati, 2018). As an impurity measure, entropy is used and 3 and 5 are selected to use as the number of maximum depth feature of decision tree classifier. In the following sections, number of maximum depth and impurity measure will be optimized with an iterative heuristic method.

I have two small models with a sample size of 0.01 having Product_Category, Brand_Name and Business_Main_Category as the predictor variables and having customer_gender and elite_member_flag as target variables respectively in two models.

According to the results, Appendix-1 shows the tree of first model, having target variable as customer gender, and there is a good prediction in the half of the leaf nodes, but other half has a high level of entropy. Appendix-2 shows the tree of first model, having target variable as elite member flag, nearly all of the leaf nodes have an entropy value, which is very close to 1. Thus, model of the prediction of the elite member flag absolutely does not work.

In the third model, which has more variables than the other two, sample size is decreased to 0.005 because of the increase of the dimensionality. Product_Category, Brand_Name, Business_Main_Category, elite_member_flag, order_price, and customer_gender are used as predictor variables and saved_card_flag, which is the most important feature for the company, is used as target variable. The tree of the model, which is shown in Appendix-3, has good predictions in a few of leaf nodes.

Although the tree in the first model looks better than other two, I decided to enhance the third model because of the importance of this feature in the following section.

b. Progress Phase

Progress Phase is the phase of trial of different methods. There are four different models, which two of them are in decision tree classifier, one in logistics regression and the other one is in sklearn's MLP classifier, through this phase. Accuracy score is implemented as the evaluation metric and the sample size is decreased to 0.001 because of the high number of samples and iterations in all of the models.

Before starting the first model, I tried to develop an iterative heuristic to find the optimal max_depth and impurity measure for the maximum accuracy. Figure 1 shows the result for this heuristic and it clearly indicates that the gini index in the max_depth of 5 performs the best.

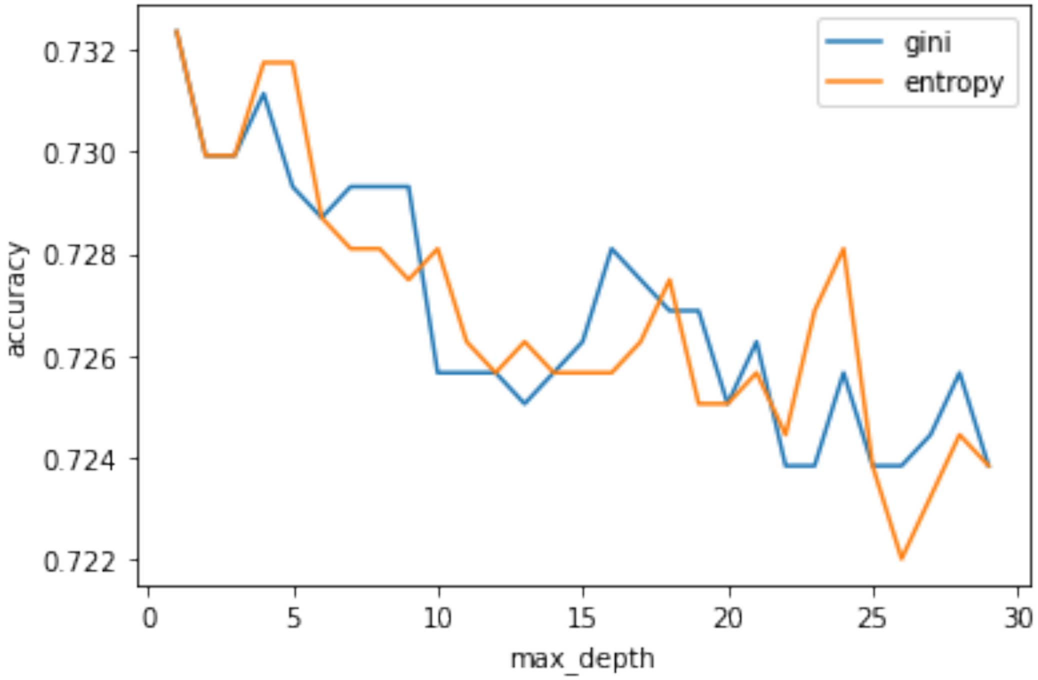


Figure 1: Iterative Heuristic for Decision Tree Algorithm

The first model is set up with the optimal parameters found above and accuracy score, which is easy to interpret, is selected for the evaluation metric. Like the previous model, Product_Category, Brand_Name, Business_Main_Category, elite_member_flag, order_price, and customer_gender are used as predictor variables and saved_card_flag is used as target variable. After a single run, the accuracy score is **0.73** and according to the tree, which is as indicated inAppendix-4, only one leaf node has enormous samples and others have very few samples. It is can be said that model is not brilliant at that moment.

Only one run with only one sample is used for the model thus far. But, for the other three models in this phase, 1000 different iteration will be implemented for randomly selected samples and 95% confidence interval of the accuracy score will be calculated for 1000 iterations.

The second model's parameters, predictor variables and target variable are the same as the first model in this phase. The only difference is the implementation of 1000 iterations. After these iterations, 95% confidence interval of accuracy score in decision tree classifier is **(0.7144, 0.7148).**

Logistic regression, which can be a good method for binary classification is tried in the third model. As the parameters of logistics regression, "ovr" for "multi_class" which, is the recommended one for binary classification, and "saga" for "solver" are selected. Because it is a regression model, R-squared is selected for evaluation metric. R-squared value provides a measure of how well observed outcomes are replicated by the model, based on the proportion of total variation of outcomes explained by the model (Steel & Torrie, 1960). 95% confidence interval of R-squared values for the third model is **(-2.274, -2.207)** and that means regression line couldn't interpret the trend and data.

After trying decision tree classifier and logistics regression so far, I wanted to investigate a model of neural network. As a beginning for neural network algorithms, I choose using sklearn library's MLP classifier without giving any parameters for the fourth model in progress section. As a result, 95% confidence interval of accuracy scores are (0.63, 0.65) in MLP classifier. This result is worse than decision tree classifier. However, it is a fact that results could be better with the right parameters and libraries in neural network algorithm.

In the following section, neural network algorithm will be focused, deeply investigated and implemented in all models.

c. Final Phase

Neural network algorithms are difficult to describe with a simple definition (Zupan, 1994). Maybe the closest description would be a comparison with a black box having multiple input and multiple output which operates using a large number of mostly parallel connected simple arithmetic units (Zupan, 1994). Many variants of learning algorithms have been proposed, from complex gradient computations, to dropout methods, but the baseline learning algorithm still consists in recursively computing the gradient by using the back-propagation algorithm and performing (stochastic) gradient descent (Denoyer & Gallinari, 2014). These great successes mainly come from the ability of DNNs to compute high-level features over data (Denoyer & Gallinari, 2014). Making complex computations with a high level of speed could encourage many researchers to specialize on neural network algorithms. Unlike MLP classifier in the previous section, I will use keras, which is a deep learning API for tensorflow library of the python, for three models in final phase.

I choose to implement Sequential class, which could be called as standard neural network algorithm, and to implement use Dense class, which is the fully connected layers of NN, in all models. There is not a simple rule to find the optimal number of nodes and it can be very complex. Thus, I choose to have 300 nodes in hidden layer, I calculate the number of columns of the predictors after one-hot encoding, which is for categorical and discrete variables, and assign it to the input_dim parameter, which equals to the number of nodes in input layer. To have efficient computation and better gradient propagation, ReLU (Rectified Linear Unit) is used as activation function except for output layer. Sigmoid activation function is promoted for the output layer for the binary classification and I choose it to use it for the output layer. Besides these parameters, binary_crossentropy, which is one of the most suitable one for binary classification, is selected as loss function, adam, which is fast and one of the best among adaptive optimizers in most of the cases, is selected as optimizer and accuracy is selected as the evaluation metrics. Lastly, number of iterations in this section is 100 for all models.

In the first model with keras's sequential model, Product_Category, Brand_Name, Business_Main_Category, elite_member_flag, order_price, and customer_gender are used as predictor variables and saved_card_flag is used as target variable. %95 confidence interval of accuracy values is **(0.72, 0.738)** which is slightly better than decision tree model for these predictors and the same target variable. However, the accuracy value is not good enough for the study. Therefore, I decided to try other two features, which are elite_member_flag and customer_gender, for the following two models.

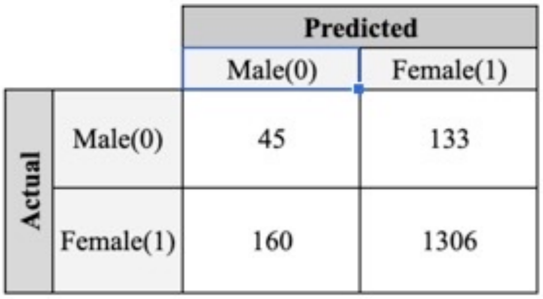
The second model have elite_member_flag as the target variable and others as the predictor variables. %95 confidence interval of accuracy values is **(0.69, 0.71).**

The third model have customer_gender as the target variable and others as the predictor variables. %95 confidence interval of accuracy values is **(0.87, 0.88).** According to this result, the study has reached a successful model at the end.

ANALYSIS ON STUDY

As it is mentioned in previous sections, saved_credit_card feature was the most important feature, but the data or the model is not good enough to reach a satisfactory model for that feature. If the only aim is to predict saved_credit_card, the study fails. Like saved_credit_card feature, the models of elite_member_flag are also unsatisfactory and the worst in this study. After finding the right parameters in neural network algorithm, we finally reached a good model by using customer gender as target variable. In business terms, gender of any customer can be estimated in high accuracy with the transaction records on e-commerce industry.

According to the confusion matrix for the final model, which is shown in Figure 2, most of the customers are female in line with the actual statistics and females could be classify better than males. Estimated precision in predicting gender of male customers is 45/(45+133)=0.25, which is very bad for this study, and estimated precision in predicting gender of male customers is 1306/(1306+160)=0.89, which is slightly better than the confidence interval's upper limit. Because most of the customers are female, low level of precision in estimating gender for male customers could not affect much. Nevertheless, the final model is not a good model and attributes are not sufficient for predicting male customers.



		Predicted	
		Male(0)	Female(1)
Actual	Male(0)	45	133
	Female(1)	160	1306

Figure 2: Confusion matrix for the final model

CONCLUSION

Through this study, a sample dataset from e-commerce industry is taken into consideration and a list of candidate target variables are defined as saved_card_flag, customer_gender and elite_member_flag. Despite the fact that the initial focus and effort was on saved_card_flag, the successful model is established when the target variable was customer gender. That means we could estimate the gender of the customer with using the features of the product, membership status and order price in high accuracy. In real life, most of the time, we could estimate the gender of the customer by looking at the shopping cart especially in the clothing shop. Nevertheless, it is hard to estimate the gender of the customer by looking at shopping cart in a supermarket in real life. Trendyol.com, which is the owner of the data, is selling products in a large scale of sectors including clothing, supermarket, meal delivery, electronic, furniture and activation codes of games or licenses etc.

I aimed to use classification model which is both efficient and adaptable for mixed type data sets. Thus, I mainly focused on neural network algorithms and decision tree classifier. In the final model, I have used python tensorflow library's keras.Sequential algorithm.

Along with this study, I handled two major obstacles. One of them was to take required permissions from the company to extract the dataset from company's cloud service to an external python environment. I solved this problem by anonymizing the sensitive customer/database information for taking permission and using a special library to be able to extract directly from the cloud environment. Second obstacle was the size of the data that I solved it by taking small samples from full dataset.

For the further improvement, new attributes could be added to the study in order to check whether it increases the accuracy score or not and to enhance the model for predicting the gender of male customers. Besides decision tree and neural network algorithms, ensemble learning models could be tried.

REFERENCES

Denoyer, L., & Gallinari, P. (2014, October 2). Deep Sequential Neural Network. https://arxiv.org/abs/1410.0510.

Minitab. (2019). Retrieved from https://support.minitab.com/en-us/minitab-express/1/help-and-how-to/modeling-statistics/regression/supporting-topics/basics/what-are-categorical-discrete-and-continuous-variables/

Patel, H. H., & Prajapati, P. (2018). Study and Analysis of Decision Tree Based Classification Algorithms. *International Journal of Computer Sciences and Engineering* , *6*(10), 74–78.

Steel, R. G. D.; Torrie, J. H. (1960). *Principles and Procedures of Statistics with Special Reference to the Biological Sciences*

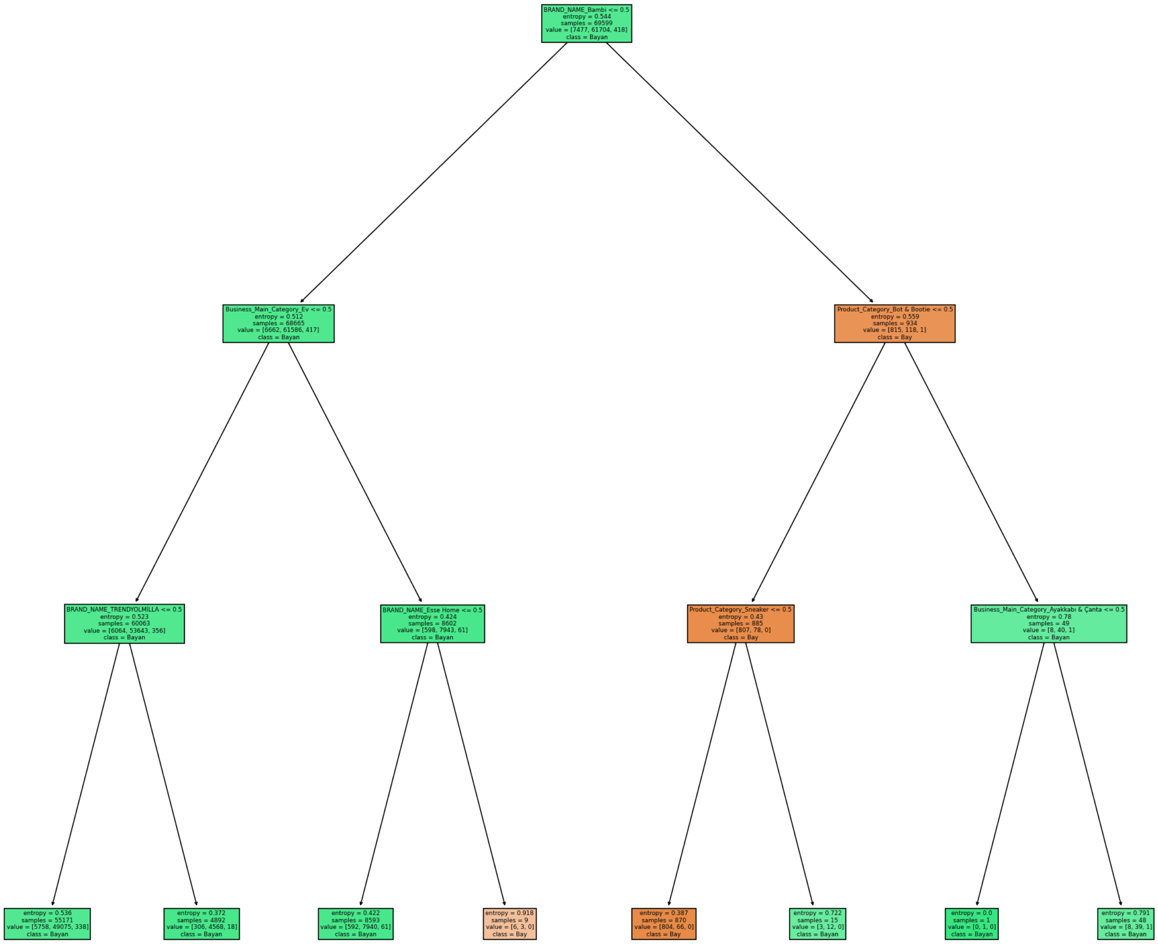
Tigani, J., & Naidu, S. (2014). *Google BigQuery analytics*. Wiley.

Yates, D. S., Moore, D. S., & McCabe, G. P. (1998). *The practice of statistics: Ti-83 graphing calculator enhanced*. W.H Freeman.

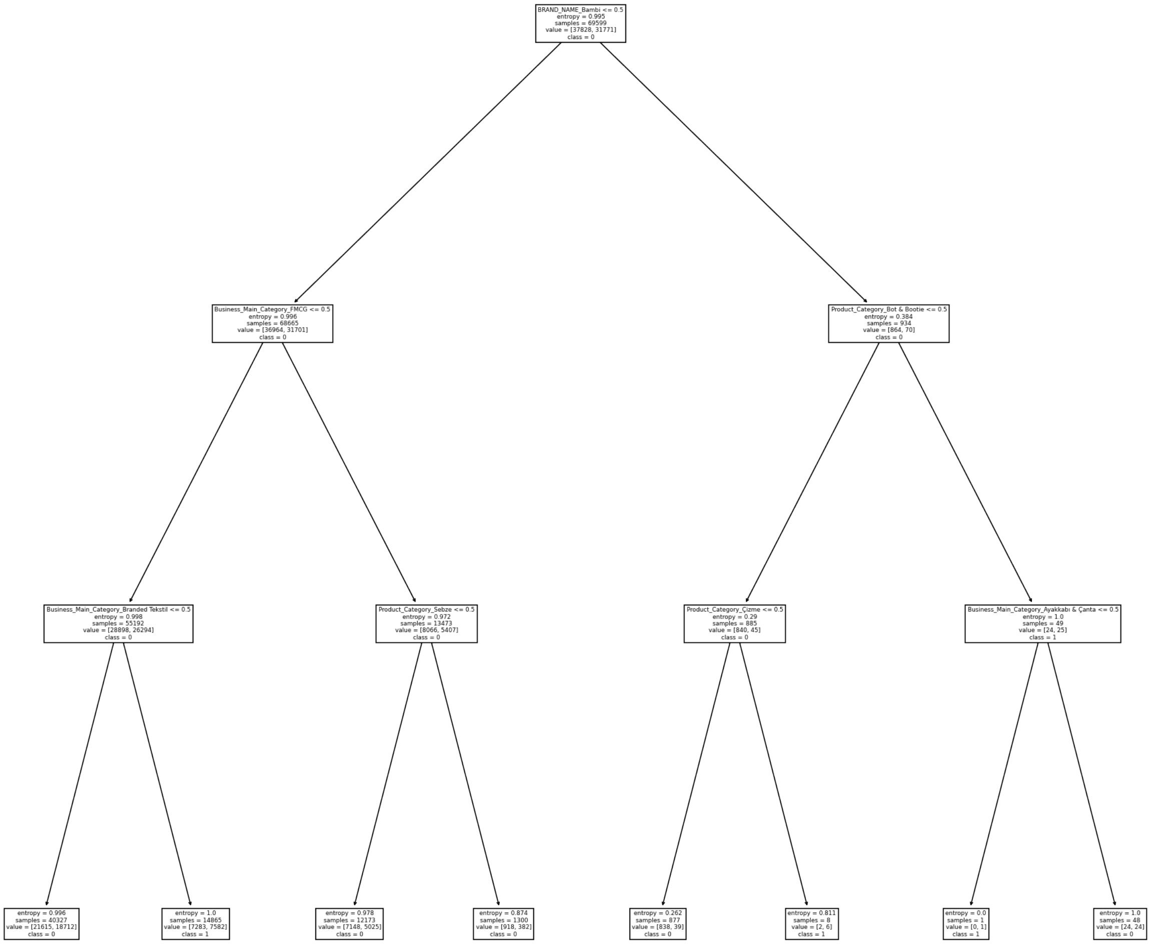
Zupan, J. (1994). Introduction to Artificial Neural Network (ANN) Methods: What They Are and How to Use Them. *Acta Chimica Slovenica*, *41*(3), 327–352.

APPENDICES

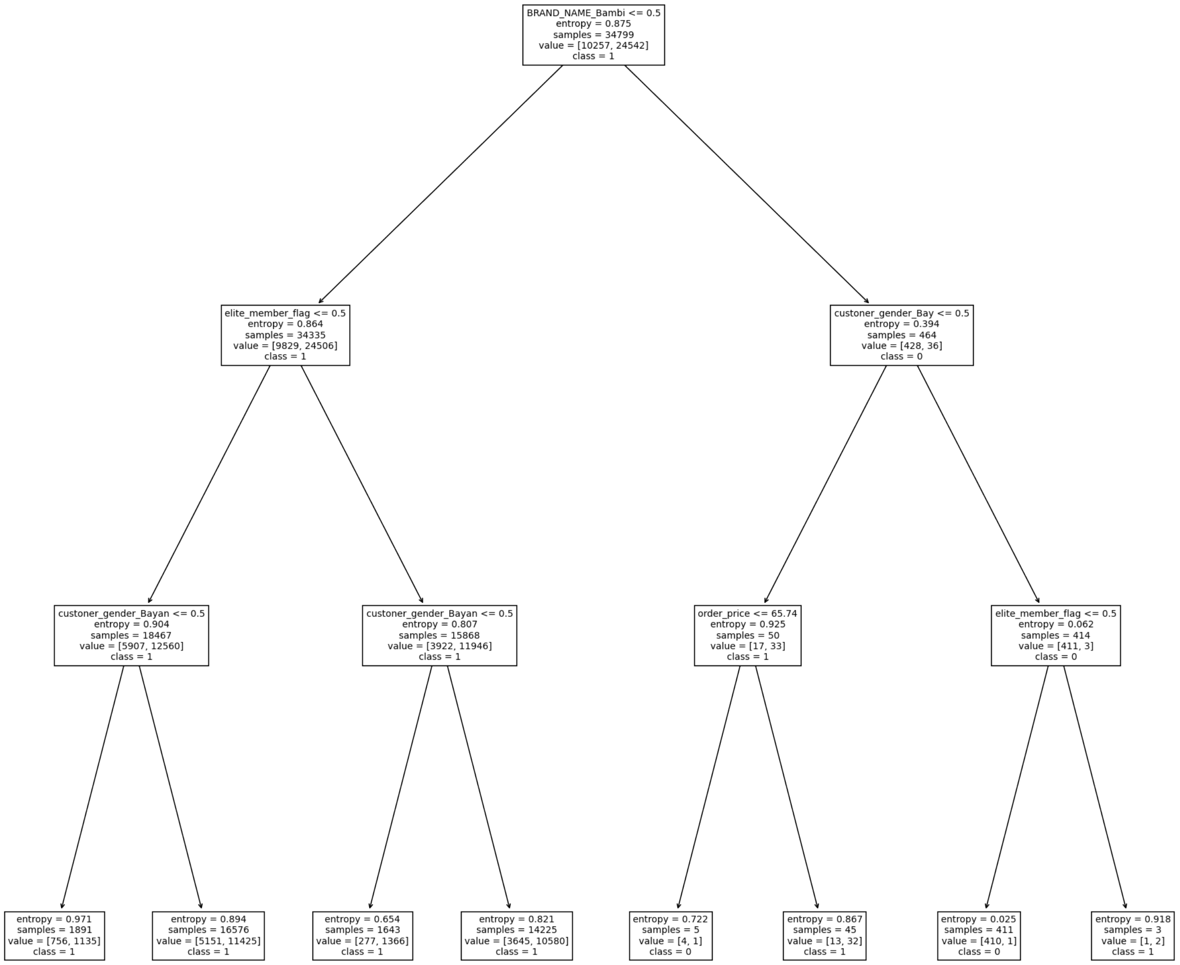
Appendix-1: Preliminary Model-1(Target variable=customer gender)



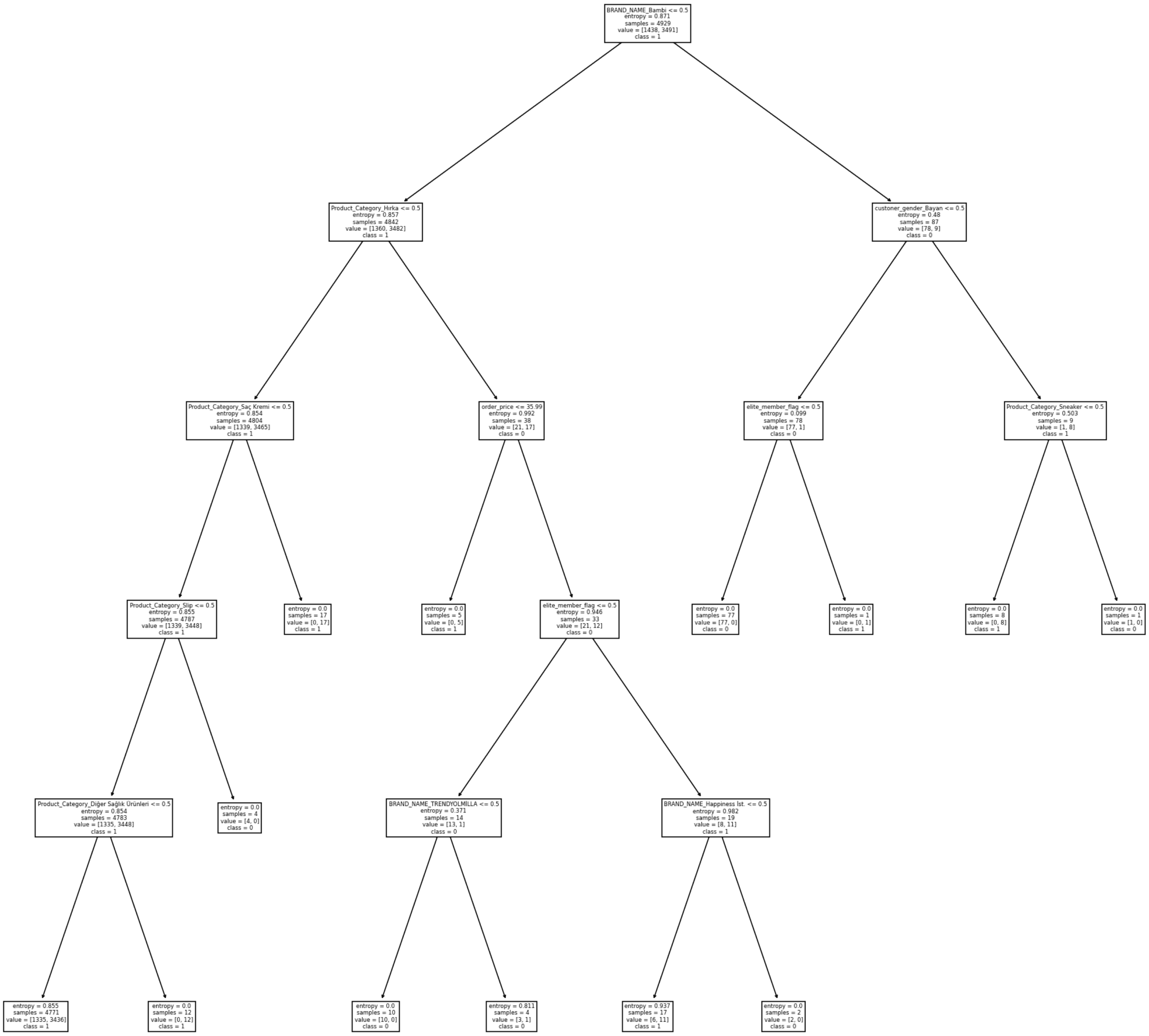
Appendix-2: Preliminary Model-2(Target variable=elite member flag)



Appendix-3: Preliminary Model-3 (Target variable=saved card flag)



Appendix 4: Progress Model-1 (Target Variable=saved card flag)



Appendix 5: Python script of the study

https://colab.research.google.com/drive/1lkwGePyMMfUJBDnpYtBqyTykxNhEr0cg?usp=sharing