University of Nottingham
UK | CHINA | MALAYSIA

MACHINE LEARNING AND PREDICTIVE ANALYTICS
BUSI4373

SPR1 2022-2023

Coursework – Churn prediction for FoodCorp

Student ID: 20494451

Executive Summary:

The customer is said to have churned when they have the tendency to terminate using a company's service over a period of time. It is used to measure customer retention, which is a prominent factor that FoodCorp should consider in building a loyal customer base since they are facing customer churn problems. If a customer has not availed any services from the company in an extended period of time, then, the customers are inactive, and the company has considered them to have churned. This can be a problem for them because it means they are losing revenue and may need to find new customers to replace those who have churned. This report aims to examine how customers behave and predict when they might stop using FoodCorp's services. By doing so, valuable insights can be gained to implement a targeted marketing strategy that will help retain customers who are likely to churn.

A churn prediction system has been developed using Python, specifically utilizing the sklearn framework for predictive analytics. The system predicts active customers who are at risk of churning and recommends preventative actions. The models taken are SVC and Random Forest. Relevant features have been identified and created within the business domain, contributing to the accuracy of the prediction system. Temporal prediction problems have been addressed, ensuring that customers identified as active are not already considered churned.

The given dataset used for predicting churn is broad and covers various aspects of customer behavior such as purchase history, store-related data, details of the product and individual receipts. The dataset contains 1,187 active customers of which 27.8% are churners and the rest 72.1% are non-churners over the span of 608 days. Churning windows was taken to analyze the behavior pattern of customers and number of days they are churning. The models taken are SVC and random forest. The accuracy of random forest is 71.4% which is more than the accuracy of SVC which is 62%. The random forest model is taken because the data fits the model perfectly. By utilizing this strategy, supermarkets may put into place focused measures to keep clients who are at the risk of churning, foster loyalty, eventually leading to business development.

The purpose of this report is to train the model with the highest possible degree of precision, recall, and accuracy to avoid revenue loss, the expenses associated with sales and marketing will be decreased by accurately identifying and targeting churners and gain insightful information for a marketing plan that will reduce client churn.

Current levels of churn:

In basic sense, churn is when the customer stops taking a particular service from one company and moves on to another company since he is offered better service. It is based on the frequency and recency of a customer's visits. In the given FoodCorps's report, Figure 1 shows that beyond 49 days 75% of the customers return for their consecutive visit. Figure 2 shows that after 40-60 days of inactivity, the number of customers who return does not rise and the curve flattens and comes to a point of no returns. As a result, it would be better to take the churn value as 16 since a small portion of the customer base may be actively engaged with by the company so marketing costs could be saved, which could help to lower churn. The optimal churn definition is set to 16 days, which means that customers who don't return to the supermarket within that timeframe will be classified as churned. This time frame helps them to identify customers who may churn sooner. A customer could be retained by FoodCorp by providing loyalty cards that have enticing offers such as a discount or free product and working on improving its overall shopping experience by improving customer service.

Data description:

FoodCorp provided the data of 4,786 customers which consisted of their first name, last name, no. of times customers visited(frequency), date of birth, details of purchase like region of stores where the customers made a purchase and date of when the purchase was made. The data is there of 608 days which has a ref_date of 22nd March 2022. The dataset was loaded on databricks using pandas dataframe, the tables are 5 in number named customers, products, receipt_lines, receipts, and stores. There are 1,187 active customers in the given dataset.

The Customers database includes information about each customer's unique customer id, first and last names, and birthdate. Product codes, product details, department codes, product names, category codes and details, and subcategory codes and details are all included in the products table. The receipt lines database includes the receipt id, receipt_lines id, product code, amount of purchased items, and product value. Receipt ID, purchase date, customer ID, store number, and till number are all contained in the receipts table. The attributes of the Stores table, which include store_code, address, postcode, latitude, and longitude of the store location, provide details about the 4 stores from which the data have been gathered.

	receipt_id	total_quantity	total_value	product_variety	purchased_at	customer_id	day
33785	34225	26	28.35	23	2022-03-22	676	0
13983	14164	11	15.27	11	2022-03-22	2746	0
24794	25121	7	7.55	6	2022-03-22	14954	0
105127	106471	18	18.05	15	2022-03-22	11181	0
30520	30923	14	20.68	13	2022-03-22	2746	0
...	...	...	...	...	...	...	...
93574	94788	3	10.88	1	2020-07-22	970	608
42736	43303	2	29.36	2	2020-07-22	638	608
112699	114129	8	7.30	8	2020-07-22	4228	608
64326	65156	11	6.18	6	2020-07-22	9091	608
92229	93429	7	6.98	6	2020-07-22	5858	608

Figure 1: Selected features

Information on customers, receipts, products, receipt lines, were supplied in 5 tables. The features that have been chosen are stored in the receipt lines database and are organised by receipt_id into a new table called receipt_total that displays the total value for each receipt id. The new dataframe receipt_total is joined with the receipts table to bring all the information together and determine the most recent customers.

Recency, frequency, and monetary (RFM) analysis is a technique used to segment customers based on their purchasing behavior and determine their value to a business.

Recency: Recency refers to the time elapsed since a customer's last purchase. In this dataset, the "purchased_at" column indicates the date of each transaction. By calculating the recency for each customer, it is understood how recently they have made a purchase.

Frequency: Frequency measures how frequently a consumer uses a good or service over a predetermined period of time. It shows how many purchases or commitments a consumer has made. Weeks, months, or any other pertinent time length might be used.

Monetary: Monetary represents the total value or number of purchases made by a customer. It indicates the customer's spending habits and their overall monetary contribution to the business. The "total_value" column in the dataset provides the information needed to calculate the monetary aspect.

There were 5 new features that were taken and used in the analysis. Feature engineering was performed, and the following new features were chosen: “total_value”, “receipt_id”, “customer_id”, “purchased_at” and “day”. After analyzing the monetary part of RFM which is **“total_value”** column, it is seen that data on the amount of money that customers are spending identifies customers that are high value, and it also shows the financial standing of the business. Each transaction or purchase has a specific identification, represented by the "receipt_id"

column. It can be helpful for keeping track of and referencing individual transactions, particularly if further in-depth investigation or transaction-level analysis is required. Now, looking at the recency of RFM, “purchased_at” column shows a particular date when each transaction took place. By analysing this, the date of purchase might reveal time series patterns when a customer's behaviour changes. Each client is individually identified by their "customer_id" field. The ability to aggregate and categorise transactions depending on certain customers is provided. Customer segmentation into churners and non-churners, identifying loyal customers, and personalising marketing strategies are all made possible by analysing customer behaviour and preferences based on their transactions. "Day" refers to the number of days the customer visited the store prior to the current date.

Two features were dropped- **“total_quantity”** and **“product_variety”.** The number of goods a client has purchased is shown in the "total_quantity" column. Although this information can be useful for logistics and inventory management, it is not directly related to the particular churn prediction analysis. The "product_variety" field shows the type or variety of product that a client has purchased. There are a number of reasons why this column was excluded. For instance, product variety is not a key component because the study is largely focused on customer behaviour and churn segmentation rather than product-specific information. These characteristics didn't provide the analysis of any noteworthy data points.

Temporal dataset:

	f1_spending	f1_quantity	f1_frequency	f1_churn	f2_spending	f2_quantity	f2_frequency	f2_churn	f3_spending	f3_quantity	f3_frequency	f3_churn
0	394.85	302	12	0	454.34	314	13	0	458.79	350	13	0
1	101.03	50	5	0	135.24	64	7	0	123.76	53	7	0
2	11.74	10	1	0	0.00	0	0	1	0.00	0	0	1
3	10.02	14	3	0	20.69	21	2	0	24.33	24	4	0
4	33.52	21	5	0	0.00	0	0	1	18.88	11	1	0
...	...	...	...	...	...	...	...	...	...	...	...	...
1182	16.94	21	1	0	30.45	27	2	0	24.40	27	2	0
1183	71.15	70	6	0	45.26	44	3	0	29.46	29	2	0
1184	46.37	20	3	0	35.52	17	3	0	75.58	27	4	0
1185	41.42	25	2	0	52.17	52	2	0	39.17	44	2	0
1186	3.30	3	1	0	6.88	9	2	0	51.79	54	5	0

Figure 2: creating a temporal dataset

There are three parameters which are 'ref_date', 'tws', and 'ows' that are a part of temporal dataset. The 'tws' parameter represents the size of the tumbling window, which is used to analyze consumer patterns over specific intervals of time. The 'ows' parameter represents the churn window, whereby assumptions are made to ascertain if a consumer has churned or not. In the tumbling window phase, there are 3 phases which are 'f1', 'f2', and 'f3' and here, training of the data occurs. Henceforth, during the 'ows' timeframe, predictions are made. Here, now test and train the model is done where, tws and ows is 30 days and 16 respectively. The function computes the customer expenditure, quantity, and frequency over these various time periods. The

variable 'X', which stands for the input characteristics for the machine learning model, is then used to store the results of these computations. Additionally, the variable 'y', which acts as the output feature for the machine learning algorithm, stores the churn information, indicating whether a client has churned or not.

Random forest:

The Random Forest Classifier function from the Sci-kit Learn package is imported to create a random forest classifier. The train dataset can be obtained by choosing the data in the range from now - 2 * output_window_size to now - output_window_size, while the test dataset can be obtained by choosing the data in the range from now - output_window_size to now. The training dataset is then used to deploy the random forest classifier using the model. Compared to the model.Fit() function model.predict() function operates the data is divided into two groups: the training set, which comprises the remaining 80% of the data, and the test set, which comprises 20% of the data. The first model under consideration, the Random Forest Classifier Algorithm, is then trained using the train data. Machine learning classification tasks are handled using the Random Forest Classifier algorithm, a sort of ensemble learning method. During the training phase, several decision trees are built, and after that, the class that represents the mode of the classes of all the individual trees is produced.

SVC (Support Vector Classifier)

SVC is a non-probabilistic binary linear classifier that divides the data into many classes using hyperplanes. It aims to find the hyperplane that best divides the classes by maximizing the distance between the nearest points from each class, known as Support vectors.

The SVC class was initially imported from the sklearn.svm module. The SVC technique is implemented using a radial basis function kernel and a gamma value of 1 after getting the test and training sets. For the purpose of allowing for results replication, the random state parameter is set to 42. For preparing for and conducting tests, use the fit () and predict() methods. Accuracy_score() returns the accuracy, which is then saved in the accuracy_svc variable. Using classification report(), the precision, recall, and f1-score for each class are computed.

Cross- Validation

Cross-validation involves statistical technique for comparing and evaluating machine learning algorithms that involves splitting the data into two sections: one for learning or training a model and the other for model validation. By performing cross validation, data can be utilized in a better way as it gives out more insights about how the algorithms perform. K-fold method segregates the data into K-sections, here, cross validation(K) =5 which means the data is divided into 5 folds and the model will be trained and evaluated on each fold separately, resulting in 5 performance metrics that can be analyzed to assess the model's performance.

Evaluation:

The models show that random forest outperforms SVC, which has an accuracy of 67.1%, with an accuracy of 71.4%. This indicates that random forest's higher accuracy makes it an effective classifier. According to the confusion matrix of the model, the winning classifier has accuracy of 0.99 for class 0, recall of 0.72 for class 0 and 0.48 for class 1. Class 0 has an F1-score of 0.83. The classifier's total accuracy is 0.71, meaning that 71% of the time it guesses the class labels correctly.

The SVC classifier has an overall accuracy of 0.67 for classes 0 and 1, a precision of 0.99 for classes 0 and 1, a recall of 0.67 for classes 0 and 1, and an F1-score of 0.80 for classes 0. According to the confusion matrix of the model, the F1-score for class 1 is low, coming in at 0.03. The classifier's total accuracy is 0.67, meaning that it correctly guesses the class labels in about 67% of cases. Overall accuracy indicates that Random Forest outperforms SVC by a little margin. The Random Forest model will thus be used to forecast client attrition. Random forest is anyhow better than SVC because:

1. **Handling large datasets:** Random Forest is more efficient in handling large datasets compared to SVC. Random Forest builds multiple decision trees in parallel, which can process large amounts of data more quickly. On the other hand, SVC requires storing support vectors in memory, which can become impractical for large datasets.
2. **Dealing with categorical features:** Random Forest is capable of effectively handling categorical features without requiring explicit encoding. It can handle categorical variables naturally, which can be advantageous when dealing with datasets that contain categorical information. SVC, on the other hand, typically requires encoding categorical variables before training.

Hyperparameter Tuning

- **Grid Search**

GridSearchCV (Cross-Validation) is a technique used for hyperparameter tuning in machine learning models. It is an exhaustive search algorithm that systematically evaluates the performance of a model across different combinations of hyperparameter values specified in a grid. This process helps in finding the best combination of hyperparameters that maximizes the model's performance on the given dataset, leading to improved accuracy and generalization. In the GridSearchCV process, you define a set of hyperparameters and their possible values.The 'GridSearchCV' class which is used to perform the grid search belongs to 'sklearn.model_selection'. The cross-validation fold is set to 5, the parameter grid and estimator are the inputs and estimator is random forest classifier. The model is trained and evaluated for each viable grid hyperparameter combination when the training data is supplied to the grid.fit() function. Grid then returns an appropriate set of hyperparameters.The corresponding score is returned by best_params_ and grid.best_score_.

After then, the measures are used to assess the accuracy.accuracy_score(). Additionally, a report of the precision, recall, and f1 score for each class in the target variable is produced using the classification report() function. The np. mean() method is then used to print the model's accuracy. Hyperparameter tuning is necessary to optimize the performance of machine learning

models, improve generalization, and find the best settings for hyperparameters. It ensures better accuracy, avoids overfitting or underfitting, automates selection, and enhances the model's robustness and reproducibility. So, random forest model is performed again, and the accuracy has increased to 72%.

Insights - Section 1:

Marketing suggestions for FoodCorp:

(1) **Cost-Effectiveness of Retention vs. Acquisition:** The statement suggests that by focusing on retention based on the analysis made above, Food Corp can reduce its spending on customer acquisition. This insight is valuable because customer acquisition is often associated with high costs, such as advertising, promotions, and sales efforts. By prioritizing retention initiatives, the company can potentially allocate more resources towards customer satisfaction, loyalty programs, and personalized marketing, which can yield a higher return on investment.

(2) **Referral Programs:** According to the non-churners data, there are existing customers who are satisfied, and loyal customers can serve as advocates for Food Corp's brand. Implementing referral programs can encourage existing customers to refer friends, family, and colleagues, thereby increasing the customer base without substantial acquisition costs. By offering incentives or rewards for successful referrals, Food Corp can leverage the power of word-of-mouth marketing and create a network of loyal customers who actively promote the brand.

(3) **Leveraging Customer Feedback:** Customer retention should be improved since churner's are disengaged with the organization. As a result, Food Corp can leverage customer feedback as a valuable resource. By actively soliciting and analyzing feedback, the company can gain insights into the reasons for churn and identify areas for improvement. This can help them develop targeted strategies to address customer pain points, enhance the overall customer experience, and ultimately reduce churn.

(4) **Tailored Retention Campaigns:** customers are at risk of churning can help Food Corp create tailored retention campaigns. By segmenting customers based on their preferences, demographics, purchase history, and engagement patterns, the company can deliver personalized offers, incentives, and communications. This targeted approach can effectively engage customers, increase loyalty, and reduce the likelihood of churn.

(5) **Enhancing Customer Experience:** Improving the overall customer experience can significantly impact retention as seen in the data below. Food Corp should focus on optimizing every touchpoint along the customer journey, from website navigation and product selection to checkout and post-purchase support. By delivering a seamless and delightful experience at each stage, the company can foster stronger customer relationships and increase the likelihood of customer loyalty.

Section-2:

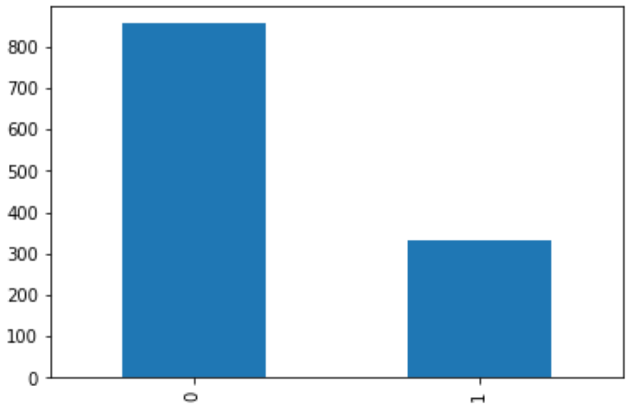

Figure 3: Churners and non-churners

Based on the study, 1,187 are the total number of active customers. Out of which, 331 have churners who have churned after 16 days and 856 are non-churners. Figure 3 shows that the number of customers who are at risk of churning labelled as 1 is greater than the number of customers who did not churn labelled 0. As a consequence, focusing on retention would enable Food Corp to reduce its spending on acquisition.

Pen-profiles:

Churners: Churners, on average, spend £209.56, which is significantly lower than non-churners. They exhibit a lower average amount spent, indicating potential dissatisfaction or decreased engagement with the company's offerings. Churners make an average of 14.50 visits, implying a lower frequency of interactions with the brand. Additionally, churners tend to purchase an average quantity of 6.28 units, suggesting reduced interest in product offerings compared to non-churners.

Non-Churners: Non-churners, on the other hand, have a substantially higher average amount spent of £25,215.76. This indicates their strong loyalty and engagement with the company, as they invest significantly more in their purchases. Non-churners make an average of 108.64 visits, demonstrating their active and frequent interactions with the brand. They also have a higher average quantity bought, with 15.04 units purchased on average, indicating a higher level of interest and satisfaction with the products.

```
Average amount spent by churners = 209.5585196374622
Average amount spent by non churners = 25215.755280373833
Average number of visits by churners = 14.49546827794562
Average number of visits by non churners = 108.64485981308411
Average quantity bought by churners = 6.283987915407855
Average quantity bought by churners = 15.035046728971963
```

Figure 4: Information on churners and non-churners