**TEMASEK POLYTECHNIC**

**SCHOOL OF INFORMATICS & IT**

**AY2022/2023 APRIL SEMESTER**

**DIPLOMA IN APPLIED ARTIFICIAL INTELLIGENCE**

**Machine Learning for Developers**

Project Report (10%)

Name: Kenzie Tan Si En

Admin No.: 2102928E

Practical Class: P02

Lecturer: Shuyun Cheng

# **Abstract**

Finding out whether a customer would churn before he/she actually does is crucial in customer retention, however it also poses as a challenging task. The focus of this project is to apply machine learning techniques that can predict as accurately whether a customer will churn.

# **Business Understanding**

## About the company

The name of the company was kept anonymous, this has some limitations on us as we are unable to find out how the company works which would help us in exploratory data analysis and other decisions on features (described in detail later). I shall call the business telecom X.

## Why is it important

Simply put, retaining customers continuously bring in profit from re-subscribing to telecom X’s contracts. Although it may sound trivial, according to research from Forrester (2009), it costs 5 times as much to acquire a new customer than to keep one.

## What needs to be achieved

Finding out what are the differences between customers who churn and those who did not.

Building an accurate predictive model which is able to classify whether customers are more likely to churn or not.

# **Data Exploration & Preprocessing**

## About the data

The dataset I have chosen comes from Kaggle <https://www.kaggle.com/datasets/blastchar/telco-customer-churn>, which is a subset of the data taken from IBM. With 7043 rows by 21 columns.

The columns can be split into 4 different groups (customerID excluded):

***Demographic info about customers*** – gender, age range, and if they have Partners and Dependents

***Services that each customer has signed up for*** – PhoneService, MultipleLines, InternetService, OnlineSecurity, OnlineBackup, DeviceProtection, TechSupport, StreamingTV, StreamingMovies

***Customer account information*** – tenure, Contract, PaymentMethod, PaperlessBilling, MonthlyCharges, TotalCharges

***Whether customer have left within the last month (Predictor variable)*** – Churn

# Data Preparation

**There are no duplicate rows in the data**

**Dealing with missing values**

After checking for anomalous values, I discovered that there were 11 rows in TotalCharges with values of “ “. From my observations, as there are no other columns with missing values, and I am unable to think of any relation on why these values are missing, I shall classify them as missing completely at random. I have decided for these rows, I would be dropping them. Only 11 of the 7043 rows has missing values, almost no effect in data loss, 11 rows would unlikely affect EDA and predictive modelling much.

Other considerations:

did not impute mean/median as data imputed would very likely be inaccurate from the actual value, variance would be underestimated and distribution slightly distorted. Multiple imputation by chained equations or k-nearest neighbours’ (KNN) imputation would cause bias in predictive modelling later.

**Data transformation**

**Text

Description automatically generated**

Most (not all) of the services that customer signed up for columns has values of “Yes”, “No” and “No internet service”. Since “No internet service” implies the same as “No” for customers, I will be replacing the values of “No internet service” with “No”.

Reason: doing so reduces dimensionality as we do not need to one-hot encode all service columns, binary data can be label encoded without many drawbacks, reducing dimensionality would help in models such as K-Nearest Neighbours who are susceptible to increased dimensionality and noise.

# **Exploratory Data Analysis**

1. Analysing customer tenure and contract

**1.i) Distribution of customers by tenure**

Chart, histogram

Description automatically generated

It is noticible that there have been quite many customers for about 1-3 months (approximately), and > 70months. It is also quite notable that every month interval, for the next 2 or so months, it is the highest for that 10th interval. This is most likely due to the contracts of customers, most customers with only 1-3 months tenure likely because of the month-to-month contract, the increase of customers after 2 or so months every 10th interval was likely due to the one-year and two-year contracts (12/24-month interval)

**1.ii) Customers by tenure, among 3 different contracts**

Chart, histogram

Description automatically generated with medium confidence

Distribution of customers among contracts:

* Most customers opted for the month-to-month contract
* The one year and two year contract have around the same number of people opted  
    
  Majority of customers who signed up for month-to-month contract stayed with the company for 1-3 months, while most of the customers with 2 year contracts tend to last for more than about 65 to 70 months, even though 2 year contracts are only 24 months long, which implies that customers are more likely to recontract with telecom X, compared to one year contract and month-to-month contract.

**1.iii) Trend between tenure and number of services a customer subscribed**

This is the newly created column which sums the number of services the customer subscribed to, plotted against the average tenure of the customers

Chart, bar chart

Description automatically generated

Every customer has at least 1 service subscribed  
The number of services the customer subscribed to generally follows a direct proportion with customer's average tenure in months, with the exception of customers with 3 services subscribed. This means encouraging customers to sign up for more services could increase customer retention with telecom X.

However, it seems that the threshold for increase in customer retention based on threshold starts to increase for > 4 services

1. Analysing customer demographics

**2.i) Distribution of gender and senior citizens in customers**

Chart, bar chart

Description automatically generated

Gender distribution in customers is almost the same (~ 50% for both male and female)  
Noticibly, both the distribution for senior citizens are also about evenly distributed among both genders  
Most of our customers are younger people

1. Analysing churn and its relationship with other variables

**3.i) Finding out customer churn rate**

Graphical user interface, application, Word

Description automatically generated

The churn rate is 26.57%, which is a large percentage, considering 1 in every 4 customers leave telecom X. The data could be skewed towards a higher churn rate, when sampling this subset of data.

This is important to note as data modelling on a skewed dataset could lead to more false negatives, but we will make do with the data we have

**3.ii) Relationship between tenure and churn**

Chart, box and whisker chart

Description automatically generated

Customers who did not churn generally stays with telecom X longer, however there are some anomalies among those who churned but has a high tenure with telecom X

**3.iii) Comparing churn between senior citizens and younger people**

Chart, bar chart

Description automatically generated

A higher percentage of seniors’ churn from telecom X compared to the younger population, the churn rate for seniors is almost 2x of the churn rate compared to those who are not seniors

­­­­­­

**3.iv) Comparing churn between monthly and total charges**

Chart, histogram

Description automatically generated

Generally, for monthly charges, customers who pay a lower monthly charge are more likely to not churn, compared to customers who pay > 70 are more likely to churn  
  
For total charges, there is not much difference between whether customers will churn or not, however, it is notable that customers who paid < 1700 are more likely to churn, those who did not churn are a little more skewed towards higher total charges compared to those who churned

# **Predictive Modelling**

## Shortcomings & Assumptions

These predictive models attempt to classify a Bernoulli outcome, where we assume each observation is independent from each other.

**Multicollinearity:**

We do not know what the company is, this means that we are unsure how the different contracts may affect the number of services subscribed (usually not the case in real life). Now we are assuming that all these different features are independent and mutually exclusive from each other, which may have affected the accuracy of the model.

I have tried using factor analysis, however it dataset is not suitable for it as it failed the Kaiser-Meylin-Oklin (KMO) and Bartlett test, hence I was unable to find the “hidden factors” which may have correlated between features.

**Data Imbalance:**

A problem that we will face is the imbalance of data as the ratio of customers who churned to those who did not churn are around 27:73. This would likely result in a higher overall accuracy as a classifier generally focuses on the majority class as it has a higher weight in the data (Thabtah et al., 2020). I will describe in more detail on how I would address this problem later on.

## Handling Categorical Values

We need to handle categorical values, mainly non-numerical data, through encoding as machine learning models used later on will not be able to interpret non-numerical data such as strings.

For columns with binary values, I used label encoding, replacing values with 0 and 1, this does not have much negative effect as using label encoding values for only 2 values, not many ‘levels’ to encode, hence tree algorithms will not have any difficulty finding features in splitting (Martin, 2019).

However, for categorical columns with more than 2 unique values, I have opted for one-hot encoding, reason being, as explained in label encoding, there are now more “levels” within the column which may pose difficulty for tree algorithms to “find” a feature to split on. Moreover, label encoding values for columns with more than 2 unique may create an accidental, level or hierarchy, which would be misinterpreted by machine learning algorithms. This may cause unwanted extra weights on certain values or features.

## Features Selection

Chi-Squared Test

I chose the Chi-Squared Test as a test of statistical significance between categorical features in relation to a categorical predictor, which uses degrees of freedom and ultimately calculate each columns p-value (McHugh, 2013). Chi-squared test is important in our case where we are completely unsure of whether our features are independent (as described in “Shortcomings and assumptions”), the purpose of this chi-squared test is to test for a feature’s independence using p-values. As for me I set the threshold p-value of accepting the null hypothesis – that a feature is not significant, for features whose values are > 0.05

A picture containing text

Description automatically generated

I will be removing PhoneService and gender columns as they have high p-values.

I have considered using ANOVA test, however the unequal variances in my groups would not be the method of choice in this case.

## Feature Scaling

K-Nearest Neighbours (KNN) – as it is a distance-based model, I will be scaling my data using 0-1 normalization. As KNN does not make assumptions about the distribution of the data when modelling, I have decided not to use standardization.

For naïve bayes, it does not need feature scaling as it is based on probability instead of distance-based calculation (Pareek, 2022).

Same goes for AdaBoost and Random Forest, as they use decision trees, there is no need for feature scaling as it uses rule-based (probabilistic) approach instead of distance calculation.

## Choosing metrics to evaluate model performance

**What to consider:**

The goal for prediction of churn for telecom X customers is to prevent customer churn before they do happen, telecom X needs to identify correctly and as most accurately as many customers who will actually churn (high precision) and minimize the number of false negatives (high recall).

Precision is the % of all churns that the model correctly identifies

Recall is the % of identified churn that ends up churning

**What I chose:**

As we value precision and recall, we shall include f1-score, which is the harmonic mean of precision and recall, it is useful as we are seeking a model which performs reasonably well across both metrics.

**Other considerations:**

Accuracy – in the case of our dataset, the number of customers who did not churn dominates those who churned. As many machine learning models are designed around the assumption of a balanced class distribution, as accuracy does not distinguish the number of correctly classified examples of different classes, it may lead to erroneous conclusions (Galar et al., 2012) and mislead us to a higher accuracy, usually close to the percentage of the majority, in our case closer to 80%.

## Hyperparameter optimization

Hyperparameter tuning for the machine learning models I will be done in the same way, the difference being the parameters of each model.

**Stratified k-fold cross validation**

K-fold cross validation significantly reduces bias as we are using most of the data for fitting and reduces variance as data is also used in the test set (Gupta, 2018). On-top of the normal k-fold cross validation, I used the stratified version to ensure that certain data imbalances are accounted for and well represented in both training and testing data.

Other considerations:

In our case, stratified cross validation is much more suitable than normal k-fold cross validation as each fold contains a balanced set of sample data when used for hyperparameter optimization. Stratified k-fold just differs to normal k-fold in this aspect.

**GridSearch**

For hyperparameter tuning, I have decided to use SciKit-Learn’s Grid Search to iteratively find the best parameters for the models, I have set the scoring parameter for Grid Search to “f1” as we are using f1 score as the main metrics to compare across models.

Other considerations:

RandomSearch is also an option when instead of GridSearch, however, a drawback of random search is that it yields high variance during computing, since the selection of parameters is completely random, luck plays its part. Considering the drawback of grid search, which is exponential increase in dimensionality – which in turns requires higher computing power, is a drawback we can accept as there are certain parameter ranges where we can estimate how the range should be before it goes absurd.

1. KNN

To see iterations for hyperparameter tuning of KNN, see Appendix A.

**Results:**

|  |  |
| --- | --- |
| Before (default parameters) | After |
|  | Chart, treemap chart  Description automatically generated |
|  | Table  Description automatically generated |

Overall, parameters given by grid search did much better in both precision and recall of both true and false predictions but more so in predicting whether customers will actually churn.

**Drawbacks of KNN:**

Although we can use KNN with both binary and continuous data, there are some important considerations we should note.

The results are going to be heavily informed by the binary splits relative to the dispersion among the real-valued results (for 0-1 scaled, unweighted vectors) like tenure, monthly charges and total charges.

In our case, most of our features are binary, 22 of the 25 features. A downside in using KNN in our case is that our numerical features may be outvalued and weigh less than it should be.

1. Naïve Bayes

I included Naïve Bayes as it addresses the shortcomings of using KNN. KNN’s accuracy is susceptible to noisy/irrelevant features as the dimensions of data increases, and heavily informed by features following a Bernoulli distribution (Ray, 2019).

As Naïve Bayes follows a conditional probabilistic approach, it is insensitive to irrelevant features and handles the shortcomings of binary data faced by KNN.

**Type of Naïve Bayes:**

Sci-kit learn library offers 5 types of naïve bayes algorithms: gaussian, multinomial, complement, Bernoulli and categorical. Out of these 5, I will be choosing Bernoulli Naïve Bayes as most of my features are binary.

**Hyperparameter tuning:**

To see iterations for hyperparameter tuning of Bernoulli Naïve Bayes, see Appendix B.

**Results:**

|  |  |
| --- | --- |
| Before (default parameters) | After |
|  | **Chart, treemap chart  Description automatically generated** |
|  | **Table  Description automatically generated** |

Overall, the default parameters for naïve bayes did better than the parameters given by grid search.

**Drawbacks of Naïve Bayes:**

Assumes that all features are independent (Vadapalli, 2022), however as described in “Shortcomings and assumptions” above, it is quite unlikely that these columns are independent. To my knowledge, it is quite likely that the type of contract has a relationship with services the customer signed up for.

## Tree Algorithms

**Why I chose these algorithms:**

As for Random Forest and AdaBoost, I have chosen these 2 models as they follow the ensemble learning approach, which are in theory more effective at handling data imbalance (Feng et al., 2018), the only difference between these 2 is that Random Forest uses bootstrap aggregation (bagging) whereas AdaBoost uses boosting.

Random Forest’s bagging aims to decrease variance whereas AdaBoost’s boosting aims to decrease bias (Kurama, 2021).

1. Random Forest

To see iterations for hyperparameter tuning of Random Forest, see Appendix C.

**Results:**

|  |  |
| --- | --- |
| Before (default parameters) | After |
|  | Chart, treemap chart  Description automatically generated |
|  | Table  Description automatically generated |

Overall, parameters given by grid search did better in both precision and recall of both true and false predictions.

1. AdaBoost (Adaptive Boosting)

To see iterations for hyperparameter tuning of AdaBoost, see Appendix D.

**Results:**

|  |  |
| --- | --- |
| Before (default parameters) | After |
|  | **Chart, treemap chart  Description automatically generated** |
|  | **Table  Description automatically generated** |

Overall, parameters given by grid search did better, specifically in identifying customers who actually churned, only in that aspect.

## Best Model Performance

The top 2 performing models are K-Nearest Neighbours and Random Forest. KNN did better in identifying actual customers who churn, whereas Random Forest did better in identifying actual customers who did not churn.

I will be choosing KNN as identifying actual customers who churn is much more important, telecom X needs to know who the customers are who will churn to take action before these customers’ churn, hence, I chose KNN.

# **Conclusion**

Through this project, I have discovered a whole ton of ways we can handle different types of problems in data, such as data imbalance etc. It takes quite a lot of effort when choosing a solution and going down the rabbit hole of solving a problem since there are many other options with their pros and cons. Almost at every step, I had to research what are the options and which is the best to apply to my dataset. As I have few experiences in handling data, and this is my biggest project yet, I realised that I am severely lacking in statistical knowledge of making decisions at every step of the way, especially in model building.

As this dataset I did have majority categorical values, on top of that an imbalanced dataset and assumptions, I had to learn background knowledge on the business part of a telecommunications company to address some assumptions that underlies within the data patterns. I also had to learn new techniques in addressing multicollinearity – such as factor analysis, hypotheses tests for statistical significance – Chi-Squared tests, ANNOVA tests, learning about new metrics evaluating model performance – area under receiver operator characteristic (AUC ROC) curve, f1-scores etc.

Overall, I enjoyed this process of exploring new techniques and learning through code, gained a newfound respect towards data engineers and statisticians who build these processes and technologies. I felt that I put in quite a bit of effort and I am overall proud of my work.

# Appendix

1. Hyperparameter optimization - KNN

Graphical user interface, diagram, application

Description automatically generated

1. Hyperparameter optimization – Bernoulli Naïve Bayes

**Chart, bubble chart

Description automatically generated**

1. Hyperparameter optimization – Random Forest

**Diagram

Description automatically generated**

1. Hyperparameter optimization - AdaBoost

**Chart

Description automatically generated**

# References

Conner, D. R. (1998). *Leading at the Edge of Chaos: How to Create the Nimble Organization* (1st ed.). Wiley.

Feng, W., Huang, W., & Ren, J. (2018). Class Imbalance Ensemble Learning Based on the Margin Theory. *Applied Sciences*, *8*(5), 815. https://doi.org/10.3390/app8050815

Forrester. (2009, April 1). *Listen to the Voice of the Customer*. CRM Magazine. <https://www.destinationcrm.com/Articles/Web-Exclusives/Viewpoints/Listen-to-the-Voice-of-the-Customer-53239.aspx>

Galar, M., Fernandez, A., Barrenechea, E., Bustince, H., & Herrera, F. (2012). A Review on Ensembles for the Class Imbalance Problem: Bagging-, Boosting-, and Hybrid-Based Approaches. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, *42*(4), 463–484. https://doi.org/10.1109/tsmcc.2011.2161285

Gupta, P. (2018, June 21). *Cross-Validation in Machine Learning - Towards Data Science*. Medium. https://towardsdatascience.com/cross-validation-in-machine-learning-72924a69872f

Kumar, N. (2019, February 23). *Advantages and Disadvantages of Random Forest Algorithm in Machine Learning*. The Professionals Point. http://theprofessionalspoint.blogspot.com/2019/02/advantages-and-disadvantages-of-random.html?\_sm\_au\_=isVvssT0L7FZr17QBkGLcKQRFMcTt

Kurama, V. (2021, April 9). *A Guide To Understanding AdaBoost*. Paperspace Blog. <https://blog.paperspace.com/adaboost-optimizer/#:%7E:text=AdaBoost%20is%20an%20ensemble%20learning,turn%20them%20into%20strong%20ones>.

Martin, D. (2019, August 25). *Encoding categorical variables*. Github. https://kiwidamien.github.io/encoding-categorical-variables.html

McHugh, M. L. (2013). The Chi-square test of independence. *Biochemia Medica*, 143–149. https://doi.org/10.11613/bm.2013.018

Ray, S. (2019). A Quick Review of Machine Learning Algorithms. *2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon)*. https://doi.org/10.1109/comitcon.2019.8862451

*Telco Customer Churn*. (2018, February 23). [Dataset]. IBM. https://www.kaggle.com/datasets/blastchar/telco-customer-churn

Thabtah, F., Hammoud, S., Kamalov, F., & Gonsalves, A. (2020). Data imbalance in classification: Experimental evaluation. *Information Sciences*, *513*, 429–441. https://doi.org/10.1016/j.ins.2019.11.004

Vadapalli, P. (2022, June 27). *Naive Bayes Classifier: Pros & Cons, Applications & Types Explained*. upGrad Blog. https://www.upgrad.com/blog/naive-bayes-classifier/