TABLE OF CONTENTS

1.0. Introduction

This report covers a critical analysis of existing subscribers in a daily newspaper company. The dataset adopted for use in this report, comprises of personal information of the company's digital subscribers. The newspaper company is perceived to be a market leader but has been faced with the challenge of customer retention. The company is therefore interested in developing strategies to manage subscriber churn rate. A proposed initiative is to offer 25% discount to inactive subscribers. In a bid to achieve this, the company is looking to develop a model that can predict possible customer/subscriber churn, in order to minimize financial loss and cost incurred in gaining new market entrance/confidence. Upon achievement of this goal, the company will be able to minimize subscriber churn and achieve overall business success.

Details of the adopted dataset can be found in the appendix of this report.

2.0. Classification Models

The dataset was first prepared by converting categorical columns to numerical columns using the map and dummies method. This was done to ensure the data was in the right format before model development. The features label was then defined by dropping two columns. The *'Subscription ID'* column was dropped because it was perceived to be insignificant in determining whether a subscriber would churn or not. The *'Subscriber'* column was also dropped because it is the target variable.

Six different classification models were developed to evaluate the performance of the dataset using different algorithms for further selection of the best performing model. Precision scoring parameter was selected because it is a more fatal error in this scenario compared to recall. Precision represents churned subscribers predicted as active subscribers, and the business objective would be more favourable if churned subscribers are not wrongly predicted as active subscribers. Therefore, the models were built to minimize false positive prediction errors. The implemented classification models are as follows:

a) **Random Forest Classification Model**
The model is built using multiple fully grown decision trees which are used to make predictions. The number of decision trees *'n_estimators'* are tuned using grid search, and the optimal number of trees returned. This model is interpretable due to its ability to generate a features importance list, which is used to assess variables that are more significant to subscriber churn. This model uses cross validation to avoid overfitting by optimizing the number of decision trees that are randomly selected, to ensure variance in performance. It was evident that the model did not underfit, possibly because of the volume of the dataset.

The final model was built using 3 variables which were, *'Year of Residence'*, *'Children'* and *'Recruitment Channel_Website'*, as they appeared to be the most significant variables contributing to subscriber churn. Upon implementation of this model, the optimal number of decision trees returned was 5, and the model performance gave a value of 0.7618. This model can be interpreted as a good performing model. The model was able to successfully minimize false positive prediction errors by 76%.

b) **Adaptive Boosting (AdaBoost) Model**
This is a boosting ensemble model that grows multiple decision stumps in sequence. Each decision stump is built with a single variable selected from a list of variables based on the

estimation of the highest information gain score. The stumps are built sequentially with different levels of say on the final prediction, until the required number of stumps have been reached. This model avoids overfitting by controlling the extent at which decision trees are grown, through the use of a single variable as opposed to splitting the data with multiple variables.

Upon implementation, the optimum number of decision stumps was 3, with an overall performance of 0.7966. This model is equally interpretable, as it returns a list of the most significant variables which were, *'Household Income of 100,000 – 199,999', 'Household Income of 300,000 – 399,999',* and *'Year of Residence'*. This goes to show that these variables contributed majorly to subscriber churn rate. The benefit of this model is that it places little emphasis on linearity; hence, they perform well on datasets with linear and non-linear relationships.

c) Support Vector Classification (SVC) Model

This model is based on the separation and classification of classes within a dataset using a hyperplane. The position of the hyperplane is dependent on datapoints closest to it; therefore, other instances like dimensionality rarely affect the model. The kernel function was further tuned using grid search, to realize the optimal function. This is to ensure the transformation of the data from a low dimension to higher dimension, for the dataset to be linearly separable.

Also, the regularization parameter 'C', was implemented for the purpose of generalization, by allowing data points outside the margin, to avoid overfitting. The Radial Basis Function (RBF) was returned as the best kernel function along with a regularization score of 0.001. Overall, the SVC model performance, yielded a score of 0.8227. SVC is a black-box model and performs well on general business applications. It is best used for problems that do not require interpretability. In this model, predator variables cannot be explained in relation to the target variable.

d) Logistic Regression Model

This is a binary classification model that follows a linear design and is an extension of linear regression model; hence, the name regression. It separates classes in a dataset with the use of a line or plane or hyperplane. It uses a logistic function to limit the output to two classes and an optimization function to find optimal values of coefficients, at which the objective function is minimized. The elasticnet regularization parameter (combination of lasso and ridge) was also included to shrink the coefficients and avoid overfitting.

Upon implementation of this model, it yielded a performance score of 0.8086. The elasticnet penalty showed that the model was built on 0% lasso regularization and 100% ridge regularization, meaning the coefficients can be shrunk by a larger margin but cannot be shrunk to zero. This model is also an interpretable model with the use of coefficient estimations to show the relationship between independent variables and the output.

e) Naïve Bayes Classification (NBC) Model

This model uses probabilistic calculations for predictions. The model assumes that variables are independent which might not necessarily be the case for most datasets. This model works well on large datasets, as it places little emphasis on dimensionality. The multinomial naïve bayes classifier was implemented in the code, using a multinomial distribution for the features in the dataset. A 10-fold cross validation was carried out to estimate precision scores for each fold. The mean score was further estimated to show the average performance on each split.

The cross-validation process is also a way to avoid overfitting and check for underfitting. An estimation of the prior probability in the bayes theorem formular is another way to control overfitting. The estimation of Maximum A Posterior (MAP) and integration of Maximum Likelihood Estimation (MLE) in the algorithm for general business applications, also avoids overfitting in NBC models. The model implementation resulted in a performance score of 0.791. This model is also relatively interpretable by showing the probability of a variable belonging to a class.

f) Multi-Layer Perceptron (MLP) Model

This model is built on neural networks with an input layer, hidden layer, and output layer. The model has five hyperparameters, two of which are related to the hidden layer (number of hidden layers and number of nodes in each hidden layer). Two other hyperparameters are related to the optimization algorithm (number of iterations and learning parameter). The last hyperparameter is the activation function which can either be rectified linear unit (relu), logistic regression function or the hyperbolic tangent function (tan h).

Overfitting can be avoided by applying constraints to the network structure, through the reduction of hidden layers and elements within each hidden layer. The reduction of model complexity helps to avoid overfitting. Therefore, only three hyperparameters were tuned with the exception of the number of hidden layers and number of nodes in each layer. By setting the hidden layer parameters to (50,), we have minimized the model complexity to one layer and 50 nodes to reduce the model complexity. The MLP model yielded a performance score of 0.7931 and a corresponding logistic activation function. This model is however a complex model and takes a long time to run due to the number of hidden layers. It is also a non-interpretable model and useful in complicated business applications.

3.0. Model Selection

Upon critical analysis of the business objective, model performance is the key focus. The business is majorly interested in identifying and classifying churning customers for the purpose of subscription renewal and customer retention. Therefore, the model that results in a high performance, will be recommended for deployment. The support vector classification model proved to be the best performing model and will be proposed for prediction of subscriber churn in the news media company.

This model is chosen for deployment because of the little emphasis placed on interpretability and understanding of possible features causing subscriber churn. This model is also less prone to overfitting due to the implementation of a regularization penalty in the algorithm, to enable the generalization of instances in the data, thereby giving a better performing precision score. Linearity is also not an issue in this model, as it transforms the data to a higher dimension for classification rather than assuming a linear form for the data. Therefore, it can be used for future news media data that hold both linear and non-linear relationships. Lastly, this model is great at ignoring outliers in cases where noisy datasets are applied.

4.0. Cost-Benefit Analysis

Cost-benefit analysis is simply an approach to determine the projected cost of undergoing a project and the estimated benefits that will be derived from the project implementation. In the news media business application, the proposed project is to deploy a model to classify and predict subscribers and non-subscribers, for the extension of a specified discount rate to its customers to drive customer retention.

The cost-benefit analysis was further estimated using the best performing model (SVC), with the use of both precision and recall scoring parameters. Parameters used in the estimation are as follows:

Table 1: Confusion Matrix

		Predicted Subscribers	
Actual Subscribers		**YES (1)**	**NO (0)**
	YES (1)	True Positive	False Negative
	NO (0)	False Positive	True Negative

Where,

True Positive (TP): Actual Subscribers correctly predicted as Actual Subscribers.

True Negative (TN): Churned Subscribers correctly predicted as Churned Subscribers.

False Positive (FP): Churned Subscribers wrongly predicted as Actual Subscribers.

False Negative (FN): Actual Subscribers wrongly predicted as Churned Subscribers.

The model performance for both scoring parameters were generated from the code and given below:

Scoring Parameters	**Model Performance Score**
SVC Precision	0.8227
SVC Recall	0.925

Also, the following information on the news media company were given.

Total Number of Paying Subscribers	1,000,000.00
Subscription cost per annum	€240.00
Annual Churn Rate	30%
Proposed Discount Rate	25%
Percentage of Active Subscribers Eligible for Discount	100%
Percentage of Churned Subscribers Eligible for Discount	40%
Cost of acquiring new customers	€150.00
Running cost of classifiers	€0

1) **Expected Cost per year if no model is deployed**

All components of the confusion matrix (TP, TN, FP, FN), are first estimated to develop a new confusion matrix.

Total Subscribers in the dataset	4286
Total Active Subscribers in the dataset	3000
Total Churned Subscribers in the dataset	1286

Therefore,

TP + FN = 3000
FP + TN = 1286

The recall score is first used to derive the TP value.

The formular for estimating Recall is:

$$\frac{TP}{TP + FN} = 0.925$$

$$\frac{TP}{3000} = 0.925$$

TP = (0.925) * (3000)

TP = 2775

Therefore,

TP + FN = 3000

2775 + FN = 3000

FN = 3000 – 2775

FN = 225

Also, Precision is estimated using the formular below:

$$\frac{TP}{TP + FP} = 0.8227$$

$$\frac{2775}{2775 + FP} = 0.8227$$

2775 = 0.8227(2775 + FP)

2775 – 2282.9 = 0.8227FP

492.1 = 0.8227FP

$$FP = \frac{492.1}{0.8227}$$

FP = 598.15

FP ≈ 598

Therefore,

FP + TN = 1286

598 + TN = 1286

TN = 1286 – 598

TN = 688

These values are then used to construct a new confusion matrix as seen below.

		Predicted Subscribers	
Actual Subscribers		**YES (1)**	**NO (0)**
	YES (1)	TP = 2775	FN = 225
	NO (0)	FP = 598	TN = 688

The expected cost was further estimated through the steps below.

Churned Subscribers	30% of 1,000,000.00	300,000.00
Active Subscribers	(1,000,000.00) - (300,000.00)	700,000.00
Percentage of Active Subscribers eligible for discount	100% of 700,000.00	700,000.00
Percentage of Churn Customers eligible for discount	40% of 300,000.00	120,000.00
Proposed Discount Rate	25% of €240	€60.00
Discounted Subscription cost in EUR	€240.00 - €60.00	€180.00
Revenue		
Expected Revenue from Active Customers	700,000.00 * 180.00	126,000,000.00
Expected Revenue from Churned Customers	120,000.00 * 180.00	21,600,000.00
Total Revenue		**€147,600,000.00**
Cost		
Cost of re-acquiring Churned Subscribers	(300,000.00) * (150)	45,000,000.00
Loss on expected revenue from Churned Subscribers	(300,000.00) * (240)	72,000,000.00
Cost of not deploying the model		**€117,000,000.00**

2) Expected Savings per year if model is deployed

The expected cost per year if model is deployed is first estimated. Then a difference between both costs is taken to derive the expected savings per year.

		Predicted Subscribers	
Actual Subscribers		**YES (1)**	**NO (0)**
	YES (1)	TP = 2775	FN = 225
	NO (0)	FP = 598	TN = 688

The above matrix was used to estimate the proportion of subscribers in the dataset. The revenue and cost were then taken to analyse the benefit and cost from each portion.

True Positive: These active subscribers were predicted correctly; therefore, the news media company will earn the full subscription fee of €240 on all subscribers in this category.

True Negative: All churned subscribers were predicted correctly; therefore, the news media company will only earn partial revenue (€240 – €60 = €180) from 40% of the subscribers in this category, due to the availed discount of 25%. The remaining 60% will be recorded as a cost to the company, as the expected discounted revenue will not be earned from these subscribers.

False Positive: Churned subscribers wrongly predicted as active subscribers will result in no revenue for the company as the subscribers were falsely represented. This however implies that the expected subscription fee on the supposed active subscribers will be recorded as a full cost to the company.

False Negative: Active subscribers wrongly predicted as churned subscribers will yield a partial revenue of (€240 – €60 = €180), because the subscribers were assumed to have churned and were offered the 25% discount. Since all active subscribers are expected to avail the discount, the expected subscription fee will then be €180 per active subscriber. The cost incurred here will be the discount amount (€60) offered to the active subscribers. If these categories of subscribers were not wrongfully labelled, then the company might not have extended this discount to the active subscribers and still earn the full subscription fee of €240 as opposed to €180.

The table below, shows the estimation of each value.

Expected Savings Per year if the Model is deployed		
TP =	647,456.84	
FP =	139,524.03	
FN =	52,496.50	
TN =	160,522.63	
Total Paying Subscribers	**1,000,000.00**	

Revenue of TP	(647,456.84) * (240)	**155,389,640.69**
Cost of TP	No cost Incurred	-
Revenue of TN	(160,522.63 * 0.4) * (180)	**11,557,629.49**
Cost of TN	(160,522.63 * 0.6) * (180)	**17,336,444.24**
Revenue of FP	No revenue earned	-
Cost of FP	(139,524.03) * (240)	**33,485,767.62**
Revenue of FN	(52,496.50) * (180)	**9,449,370.04**
Cost of FN	(52,496.50) * (60)	**3,149,790.01**
Cost of Deploying Model	Cost of TN + Cost of FP + Cost of FN	**€53,972,001.87**
Cost of acquiring Churned Subscribers	(160,522.63 * 0.6) * (150)	**€14,447,036.86**
Total Cost of Deploying Model	(Cost of Deploying Model) + (Cost of acquiring Churned Subscribers)	**€68,419,038.73**
Total Cost of not Deploying Model		**€117,000,000.00**
Expected Savings	(Total cost of deploying model) - (Total Cost of not deploying model)	**€-48,580,961.27**

This shows that the news media company will save €48,580,961.27 by implementing the Support Vector Classification model for the classification of subscriber churn.

5.0. Conclusion

Findings from this report have shown that, of all the six classification models developed, Support Vector Classification (SVC) model performed better for the subscriber churn classification problem. Upon estimation of the cost-benefit analysis, it was realized that the news media company would save €48,580,961.27 by deploying the SVC classification model to predict subscriber churn. This estimation was beneficial to understand possible losses that could be avoided by adopting a churn classification system.

6.0. References

Akash Desarda (2019). Understanding AdaBoost. [online] Medium. Available at: https://towardsdatascience.com/understanding-adaboost-2f94f22d5bfe.

notast, R. on (2019). Explaining Predictions: Interpretable models (logistic regression) | R-bloggers. [online] Available at: https://www.r-bloggers.com/2019/06/explaining-predictions-interpretable-models-logistic-regression-2/ [Accessed 17 Apr. 2022].

neylicious.github.io. (n.d.). Naives Bayes Classifier. [online] Available at: https://neylicious.github.io/ml/2017/09/17/naive.html [Accessed 18 Apr. 2022].

www.kaggle.com. (n.d.). Logistic Regression vs AdaBoostClassifier | Data Science and Machine Learning. [online] Available at: https://www.kaggle.com/questions-and-answers/182637 [Accessed 17 Apr. 2022].

7.0. Appendix

The dataset is stored in a csv file titled *NewspaperChurn.csv*.

Total number of columns = 8

Total number of rows = 4286

S/N	Column Name	Column Description
1	Subscription ID	Unique ID per subscription
2	Household Income	Range of income per household
3	Home Ownership	Forms of home ownership per subscriber
4	Children	Presence or absence of children
5	Year Of Residence	Number of years resident per subscriber
6	Age Range	Age range per subscriber
7	Recruitment Channel	Forms of recruitment channel
8	Subscriber	Status of subscribers.