Nhi Le

D209 Prediction analysis

western governors university

September 11, 2023

**Part I: Research Question**

**A1. Proposal of question**

Which customers are at high risk of churn? What are the features that significantly impact churn?

**A2. Defined goal**

From this analysis, the stakeholders can understand what factors affect churn to see where the services can be improved. Moreover, they will be able to predict which customers have high risks of retention. Then, the customer support team can contact those customers and have strategies to improve their experience with the company.

**Part II: Method Justification**

**B1. Explanation of prediction method**

I will be using decision trees as the prediction method in this analysis. This method will help me to create a model that predicts churn by using a tree-like pattern of decisions. The decision trees mimic human thinking when humans are making decisions. This method can handle both categorical and continuous variables. By using this method, the expected outcome is to see what features affect churn. Moreover, a tree consists of 2 major components: decision node – the point where you decide, and leaf node – the output of said decision (GeeksforGeeks, 2023). The expected outcome is to find the best parameters for the model.

**B2. Summary of method assumption**

One assumption of decision trees method is that in the beginning, the whole training set is considered as the root (Sharma, 2021). Secondly, records are distributed recursively based on attribute values (Sharma, 2021). Thirdly, the method prefers feature values to be categorical, if the values are continuous, they will need to be discretized prior to building the model (Sharma, 2021).

**B3. Packages or libraries list**

Here are the packages and libraries I will use in Python:

* NumPy is used for working with arrays.
* Pandas is used for working with a data set. Example, we can use .read.csv() to load the data set to Python. Or I can use .info() to get the information of the data set.
* Matplotlib is a comprehensive library for creating visualizations. I will use matplotlib.pyplot submodule for creating histograms. They will help me detect outliers. This package also helps me to create bivariate visualization such as scatter plots.
* Seaborn is also used to create boxplots to detect outliers and create univariate/ bivariate visualizations.
* Scikit-learn is used for machine learning, especially for decision trees in this analysis. Example, Train\_test\_split can be used to split the data set into training and testing sets, DecisionTreeRegressor to create Decision Tree classifier object, classification\_report to create classification report.

**Part III: Data Preparation**

**C1. Data preprocessing**

My data pre-processing goal is to detect missing data, duplicate data, and outliers, then decide to treat them with appropriate methods. Moreover, I want to use all the independent variables for the analysis. To make that happen, I will need to encode categorical values into numerical values.

**C2. Data set variables**

The dependent variable used to answer the research question was ‘Churn’. It is a categorical variable as the values are ‘Yes’ or ‘No’.

The independent variables used to answer the research question:

* The 10 continuous variables including:
* ‘Children’, ‘Age’, ‘Income’ are demographic variables on billing statement for each customer.
* ‘Email’ is numeral variable to record the number of marketing or correspondence emails sent.
* ‘Contacts’ is numeral variable for how many times customer contacted technical support.
* ‘Outage\_sec\_perweek’ shows system outages in the customer’s neighborhood’s average of seconds per week.
* ‘Bandwidth\_GB\_Year’ (the average yearly amount of data used, in GB, per customer) is a continuous variable.
* ‘Tenure’ is numerical variable to record how many months the customer has been with the provider.
* ‘MonthlyCharge’ is the monthly charge for the customer.
* ‘Yearly\_equip\_failure’ is numeral variable to show the number of time customer’s equipment failed and needed to reset or replaced last year.
* The 23 categorical variables including:
* 8 categorical variables reflect customer’s satisfaction ratings on a scale of 1 to 8 (1 = most important, 8 = least important): ‘item1’ – Timely response, ‘item2’ – Timely fixes, ‘item3’ – Timely replacements, ‘item4’ – Reliability, ‘item5’ – Options, ‘item6’ – Respectful response, ‘item7’ – Courteous exchange, ‘Item8’ – Evidence of active listening.
* ‘Gender’ is categorical variable to reflect the gender of customer.
* ‘Techie’ has Yes/No value. This categorical variable reflects if the customer thinks that they are good at technology.
* ‘Contract’ is categorical variable on what kind of contract customer has ‘Month-to-month’, ‘One Year’, or ‘Two Year’.
* ‘Tablet’ is categorical variable answering if the customer has a tablet.
* ‘Port\_modem’ is categorical variable answering if the customer has a portable modem. The values are ‘Yes’ or ‘No’
* ‘InternetService’ is categorical variable that shows customer’s internet service provider.
* ‘Phone’, ‘Multiple’, ‘OnlineSecurity’, ‘OnlineBackup’, ‘DeviceProtection’, ‘TechSupport’, ‘StreamingTV’, ‘StreamingMovies’, ‘PaperlessBilling’ are services that the company provides. The values of these variables are ‘Yes’ or ‘No’ to reflect if the customer signed up for. These are categorical variables.

**C3. Steps for analysis**

Data preparation steps:

* Import dataset churn\_clean.csv into Jupyter Notebook.
* Get information (column names, data types), and statistical details (count, min, max, mean, std, percentile) of the dataset.
* Detect duplicates and delete the duplicated records if there are any.
* Find missing data and impute missing data with meaningful measures of central tendency (mean, median, or mode).
* Find outliers and treat them by removing them, retaining them, excluding them, or imputing them with the median.
* Run univariate and bivariate visualizations to see the spread of data.
* Rename the Item1 – Item8 columns to easily recognized names (For example: ‘Item1’ renamed to ‘TimelyResponse’).
* Drop variables that will not be needed for the analysis.
* Create dummy variables for categorical variables.
* Encode categorical values to numerical values: For those variables with Yes/No values, the dummy value is 1 for Yes and 0 for No. For the Gender variable, it has Male, Female, and Nonbinary. The DummyFemale is 1 when Gender is Female and else it is 0. Contract has 3 values: Month-to-month, One Year, and Two year. DummyMonthtoMonth is 1 when Contract is Month-to-month, else it is 0. InternetService has 3 values: Fiber Optic, DSL, and None. DummyFiberOptic is 1 when InternetService is Fiber Optic, else it is 0.
* Spot-check the statistical details of the dataset to make sure categorical values are encoded correctly.
* Drop those categorical values from the data set.
* Use heatmaps to detect independent variables with high correlation with others and drop them.
* Extract the prepared dataset as CSV file named ‘churn\_prepared1.csv’.

A screenshot of a computer

Description automatically generated

A screenshot of a computer code

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A white rectangular object with black text

Description automatically generated

A screenshot of a computer

Description automatically generated

A line of dots with black text

Description automatically generated

A screenshot of a computer screen

Description automatically generated

A screenshot of a computer screen

Description automatically generated

A screenshot of a computer code

Description automatically generated

A screenshot of a graph

Description automatically generated

A screenshot of a graph

Description automatically generated

A screen shot of a graph

Description automatically generated

A screen shot of a graph

Description automatically generated

A graph with a bar and a line

Description automatically generated with medium confidence

A screenshot of a graph

Description automatically generated

A screen shot of a computer

Description automatically generated

A screenshot of a computer screen

Description automatically generated

A screenshot of a computer screen

Description automatically generated

A screen shot of a computer screen

Description automatically generated

A screenshot of a computer screen

Description automatically generated

A screenshot of a graph

Description automatically generated

A screenshot of a computer screen

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer screen

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer code

Description automatically generated

**C4. Data preprocessing**

The prepared data set will be submitted as ‘churn\_prepared1.csv’ along with this doc file.

**Part IV: Analysis**

**D1. Splitting the data**

The csv files will be submitted as ‘X\_train\_2.csv’, ‘X\_test\_2.csv’, y\_train\_2.csv’, and ‘y\_test\_2.csv’ along with this doc file. The data set was split into training (80%) and testing (20%) sets.

A screenshot of a computer program

Description automatically generated

**D2. Output and intermediate calculations**

After splitting the data into training and test data sets, I fitted the data sets into the DecisionTreeClassifier model by using the default parameters and created a new array called ‘y\_pred’ to make predictions on the test data set. The analysis technique I used was to tune some of the model hyperparameters to get a better model with higher accuracy score. To do that, I would use GridSearch class, it helped me to find the best hyperparameters and apply cross validation. I used accuracy scores and the mean squared error of the models to compare which model is better.

The results from GridSearch showed that the best parameters were: Fitting 5 folds for each of 50 candidates, totaling 250 fits: {'criterion': 'entropy', 'max\_depth': 10, 'min\_samples\_leaf': 50}.

A screenshot of a computer program

Description automatically generated

A screenshot of a computer code

Description automatically generated

A screenshot of a computer

Description automatically generated

**D3. Code execution**

The code was included in D2 above, and in ‘D209(2) NL.ipynb’ file submitted along with this doc file.

**Part V: Data Summary and Implications**

**E1. Accuracy and MSE**

My final decision tree model with the best parameters: {'criterion': 'entropy', 'max\_depth': 10, 'min\_samples\_leaf': 50}. The accuracy of the model was 87.90%, out of 2000: True Positive + True Negative = 1345 + 413 = 1758, False Positive + False Negative = 145 + 97 = 242. The mean squared error (MSE) of the model is 0.121. The MSE measures the average of the squares of the errors. With this low MSE at 0.121, it indicates that this is a good model.

Accuracy score and mean squared error of the model with the best parameters after tuning hyperparameters:

A screenshot of a computer

Description automatically generated

**E2. Results and implications**

The accuracy score of the initial model with default parameters was 83.45%. After tuning hyperparameters, I created a new model with the best parameters. The accuracy score of the new tables improved to 87.90%. Along with that, the mean squared error reduced from 0.1655 to 0.121. This indicates that the new model is better than the initial model. By using the decision tree model with the best parameters, stakeholders can predict which customers are at high risks of churn. In my opinion, the accuracy score is not as high as what I expected (to be over 90%). Therefore, I think the model is not significantly practical. However, the stakeholders can still use this model to see what features affect churn and have plans to reduce retention.

**E3. Limitation**

The limitation of my analysis is that the nature of decision trees method has overfitting problems. Overly complicated trees do not generalize the input well. Overfitting occurs when the tree reaches a particular level of complexity, which likely happens in a large tree (EDUCBA, 2023). To avoid overlifting, we need to prune, establish the minimum samples required at a leaf node, or set the maximum depth of the tree (EDUCBA, 2023). Since my model is large and has many features, the model is complex. Therefore, its limitation is that stakeholders with little data analytics experience will have a hard time understanding it.

**E4. Course of action**

As looking at the decision tree model, I see that customers with Month-to-month contracts had high risk of retention. The recommended course of action is that Sales should keep customers on One-Year or Two-Year contracts to help to reduce churn. Secondly, high monthly charge is also a reason to contribute to high churn. The stakeholders should create pricing plans to make the services more affordable for customers. For example, loyal customers can be offered recurring or one-time discounts. Customers in the region with lower region’s avarage income can get cheaper prices.

**Part VI: Demonstration**

**F. Panopto recording**

<https://wgu.hosted.panopto.com/Panopto/Pages/Viewer.aspx?id=8ad30e4e-c6ad-4eec-9757-b07a01757c9f>

**G. Sources of third party code**

Navlani, A. (2023, February 23). *Python decision tree classification tutorial: Scikit-Learn Decisiontreeclassifier*. DataCamp. https://www.datacamp.com/tutorial/decision-tree-classification-python

Piepenbreier, N. (2022, April 17). *Decision tree classifier with Sklearn in python • datagy*. datagy. https://datagy.io/sklearn-decision-tree-classifier/

**H. Sources**

GeeksforGeeks. (2023d, August 20). *Decision tree*. GeeksforGeeks. https://www.geeksforgeeks.org/decision-tree/

Sharma, A. (2021, March 1). *Machine learning 101: Decision tree algorithm for Classification*. Analytics Vidhya. https://www.analyticsvidhya.com/blog/2021/02/machine-learning-101-decision-tree-algorithm-for-classification/

*Decision tree limitations: Learn the limitations of decision trees*. EDUCBA. (2023, March 13). https://www.educba.com/decision-tree-limitations/