**CLASSIFICATION ANALYSIS (NVM3 TASK 1)– D209**

**Performance Assessment**

**Western Governors University**

**By: Christian LeBlanc**

***Part I: Research Question***

**A-1**

**“Can a KNN model predict a customer discontinuing service within the last month?” I will explore this question by using the k-nearest neighbor (KNN) method.**

**A-2**

An important goal of this data analysis is to use the KNN method to help identify causes of customers discounting service to minimize it.

***Part II: Method Justification***

**B-1**

The k-nearest neighbor (KNN) method predicts the label of a data point by looking at the “k” amount of the closest labeled data points. Setting “k” to a number will get that number of labeled data points closest to an unlabeled data point. Then it will assign the label of the majority of data points in the “k” closest points to the unlabeled data point. It is important to point out that “k” can be changed as necessary to create the most effective model. Once the most effective model is found then “k” will be constant for all unlabeled data points. This method uses distance to determine the nearest neighbors which is most commonly and as a default Euclidean distance.

**B-2**

The KNN algorithm assumes that similar things exist in close proximity (Medium, 2019). This means that for KNN to be useful we are assuming that data points with the same labels will be found closely in distance to other data points with the same label.

**B-3**

I will be using the Python langue for my analysis. Pandas is imported to allow for me to import the csv data that I will be using. From pandas I imported CategoricalDtype to help with memory efficiency with the categorical data I am using. Numpy was the next import. It was imported to have access to certain calculations. Matplotlib.pyplot and seaborn are imported to allow for visualizations that may be needed. The multiple imports from sklearn were needed to run my KNN will follow. From sklearn import preprocessing provides a variety of tools for preprocessing and transforming data in preparation for KNN. From sklearn.feature\_selection import SelectKBest, f\_classif allows for feature selection. From sklearn.neighbors import KNeighborsClassifier is used for classification in the KNN. From sklearn.model\_selection import GridSearchCV allows for 5 fold cross validation and finding the best parameter. From sklearn.metrics import confusion\_matrix to show the confusion matrix for the model. From sklearn.metrics import roc\_auc\_score is used to get the Area Under the Curve score. From sklearn.metrics import roc\_curve is used to plot the Receiver Operating Characteristic. From sklearn.metrics import classification\_report is to print a summary report for the KNN.

***Part III: Data Preparation***

**C-1**

**One data preprocessing goal that is needed to be done is encoding categorical data. This is needed due to KNN method measures the distance between data points, so it requires data to be numeric. This will require changing binary categories from “True” or “Yes” to 1 and “False” or “No” to 0. Categorical variables with more than two responses require similar changes to numerical values done with dummy variables for each of the possible responses.**

**C-2**

The following are the variables that will be used for my analysis:

* Children- Numeric
* InternetService- Categorical
* Contract- Categorical
* Gender- Categorical
* Techie- Categorical
* Multiple- Categorical
* OnlineBackup- Categorical
* DeviceProtection- Categorical
* TechSupport- Categorical
* StreamingTV- Categorical
* StreamingMovies- Categorical
* Bandwidth\_GB\_Year- Numeric
* Churn- Categorical

**C-3**

The following steps were taken to prepare the data for analysis:

* + Import the packages and libraries needed. A screen shot of a computer program

    Description automatically generated
  + Load the churn\_clean csv file. 
  + Check the columns for 10,000 non-null count and datatypes. 
  + Check the variables that I will be using. The numeric columns are checked for outliers and the value counts for expected data dictionary responses for the categorical columns. I did find an outlier for Children, but 10 children is not an unreasonable response so I prefer keeping it in to ensure the most accurate analysis.

* + Create Boolean mapping and convert columns that need to be changed into Boolean columns. A screenshot of a computer program

    Description automatically generated
  + Create a new dataset with just the variables listed in C-2A screen shot of a computer

    Description automatically generated
  + Separate the explanatory variables (x) from the response variable (Y). A computer screen shot of a computer code

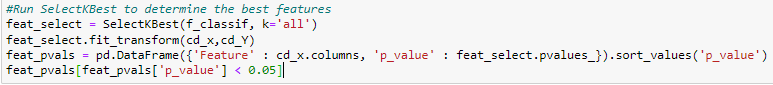
    Description automatically generated
  + Create dummy variables for my categorical variables. A screen shot of a computer code

    Description automatically generated
  + Insert dummy variables in with my explanatory variables and drop columns that dummy variables came from. A screen shot of a computer code

    Description automatically generated A screenshot of a computer program

    Description automatically generated
  + Create a Heatmap to view correlations in the data. A screen shot of a computer program

    Description automatically generated
  + Drop variables with strong correlation. A screenshot of a computer code

    Description automatically generated
  + Run SelectKBest to determine the best features. 
  + Drop features that were not in SelectKBest. A white background with red text

    Description automatically generated
  + Standardize the explanatory variables (x) to put all on the same scale for KNN to measure distance. 

**C-4**

Please see attached d209clean1.csv file for copy of clean data.

***Part IV: Analysis***

**D-1**

Please see attached csv files named Xtrain, Xtest, ytrain, and ytest for the split data.

**D-2**

The K-Nearest Neighbors requires to name a value for ‘k’. To find which ‘k’ returns the most accurate model I used GridSearchCV.

A screenshot of a computer code

Description automatically generated

This provided that ‘k’ = 15 results in the most accurate model from the range of ‘k’ from 1 to 30. From here I checked the score for this parameter.

A close-up of a message

Description automatically generated

I can begin with the KNN classification using 15 as my ‘k’. I fit the KNN to my training data and run a confusion matrix with the 2000 customers that are in the testing data.

A screenshot of a computer program

Description automatically generated

I kept the default weight of all neighbors being equal because my research question is looking for factors that can predict Churn and giving any of them more weight would not produce the wanted results. I used the standard KNN algorithm with Euclidean distance for the same reason.

**D-3**

Below I will provide screen shots of the code that I used starting after splitting and saving the data as that is when section D2 starts at. I am also uploading my full code on a Jupyter notebook if more is needed. I used segments of code found in [D209 Data Mining 1 Task 1 Cohort](https://westerngovernorsuniversity.sharepoint.com/:p:/r/sites/DataScienceTeam/_layouts/15/doc2.aspx?sourcedoc=%7B945F58A7-B99E-4D7A-BEC0-9BB216B4D2BD%7D&file=D209%20Data%20Mining%201%20Task%201%20Cohort.pptx&action=edit&mobileredirect=true).

A screenshot of a computer program

Description automatically generated

A screen shot of a graph

Description automatically generated

***Part V: Data Summary and Implications***

**E-1**

The training accuracy is 0.85725 and the testing accuracy is 0.829. This show the total number of correct predictions divided by the total of all the predictions. The Area Under the Curve (AUC) for my classification model is approximately 0.888. The AUC of 1.0 indicates that all predictions were perfect, while an AUC of 0.5 a model that is as good as a random and 0.0 indicates that all predictions were wrong. My score of approximately 0.888 shows that my model is closer to perfect predictions than it was at random.

**E-2**

The results from my KNN classification model seems to show that it could be useful in predicting Churn. The AUC of 0.888 is closer to making perfect predictions than it is to making random one. A testing accuracy score of 0.829 has my model predicting the correct result of Churn 82.9% of the time in the test set. Both metrics point to the model being useful. I used GridSearchCV to do my hyperparameter tuning resulting in 15 neighbors was the best when going through the results from 1 to 30. I am unsure if using a wider range would make the model better or just slow down my code and still result in 15. I am not sure if including more labeled data will improve my model. I cannot see data points such as latitude and longitude of a customer being a useful indicator of Churn, but I have worked with data enough to know that sometimes unexpected things have a correlation.

**E-3**

The biggest limitation I believe is that this analysis is a snapshot in time for the customer’s information. I think having multiple snapshots of a customer’s activity would produce better results on a topic such as churn. I realize that this would possibly lead to different types of analysis that may be out of the scope of this class, but if this was work related and I was asked about limitations of my results this would be my first thing I would bring up.

**E-4**

Based on the AUC and the accuracy I would recommend this k-NN model to predict customers that are at churn risk. I believe this could be useful for two different departments to have. The sales team could come up with ways such as promotions to keep the at-risk customers with the company. I also think informing the finance department would be useful as they could have an estimate of the money coming in from customers to set a better monthly budget for the company.

***Part VI: Demonstration***

**F**

Uploaded it to the Panopto drop box titled titled “Data Mining I - NVM2 | NVM3 | D209 (student creators) [assignments].” Link: <https://wgu.hosted.panopto.com/Panopto/Pages/Viewer.aspx?id=a3d65a5b-c96d-4617-b97c-b103016fa885>

**G**

Elleh, F. (2023, August 15). D209 Data Mining 1 Task 1 Cohort. [D209 Data Mining 1 Task 1 Cohort.pptx (sharepoint.com)](https://westerngovernorsuniversity.sharepoint.com/:p:/r/sites/DataScienceTeam/_layouts/15/Doc.aspx?sourcedoc=%7B945F58A7-B99E-4D7A-BEC0-9BB216B4D2BD%7D&file=D209%20Data%20Mining%201%20Task%201%20Cohort.pptx&action=edit&mobileredirect=true)

**H**

Medium. (2018, Sep 26). Building a k-Nearest-Neighbors (k-NN) Model with Scikit-learn. <https://towardsdatascience.com/building-a-k-nearest-neighbors-k-nn-model-with-scikit-learn-51209555453a>

**I**

The content in this Performance Assessment is set up and presented with the highest professional standards.