D209 Data Mining 1 Performance Assessment

Task 1: Classification Analysis

Author: Eric Colwell

Western Governors University

This report deals with the rubric for Western Governors University course D209 Data Mining 1 Task 1 and answers all the items in rubric order.

# **Part I:**

**A1:**

The research question for this analysis is “Using the k-nearest neighbors method, is it possible to predict customers that are at risk of churn using other variables from the churn dataset?

**A2:**

The objective of this data analysis is to create a machine learning model using k-nearest neighbors (KNN) to determine which customers are a churn risk. For this analysis I have chosen Churn as the dependent or target variable. For the initial model, the independent or predictor variables are Population, Area, Children, Age, Income, Marital, Gender, Outage\_sec\_perweek, Email, Contacts, Yearly\_equip\_failure, Techie, Contract, Port\_modem, Tablet, InternetService, Phone, Multiple, OnlineSecurity, OnlineBackup, DeviceProtection, TechSupport, StreamingTV, StreamingMovies, PaperlessBilling, PaymentMethod, Tenure, MonthlyCharge, and Bandwidth\_GB\_Year. Variables not included in the initial model are CaseOrder, Customer\_id, Interaction(UID), City, State, County, Zip, Lat, Lng, TimeZone, and Job. These variables have high cardinalities and would not add predictive power to the model. Also, the variables Item1 through Item8 are not included.

# **Part II:**

**B1:**

The KNN is used to predict categorical data by using the closest labeled data point to classify it. The prediction can vary depending on how many ‘neighbors’ the model is set to. In the example figure below, if k is equal to 3, then the data point to be predicted would be labeled as Class B (Purple) because 2 of the 3 nearest known data points are Class B. If k is equal to 6 the point to be predicted would be labeled as Class A (Yellow) because 4 of the 6 points nearest known data points are Class A. This analysis is to determine customers that are at churn risk. The ‘Churn’ variable is categorical with only Yes or No possibilities. This makes KNN an appropriate technique for the analysis.

Diagram

Description automatically generated(Jaroli, 2019)

**B2:**

One assumption of KNN is that similar things are in proximity to each other. While this assumption isn’t always true, it is an efficient way to classify data. By testing different settings for k, and checking different metrics, I can determine which setting for k shows the most accurate results (Elleh, 2022).

**B3:**

The benefits of using the Python language for this analysis are that it is a capable tool for data science, statistics, data exploration, data analysis, and predictive analytics. Also, I am more familiar with Python. For these reasons I chose this programming language for the KNN analysis. I imported pandas to import the dataset, data manipulation, and analysis. I imported numpy for numerical calculations. I imported matplotlib.pyplot for plotting data. I imported stats from scipy for statistical analysis and outlier detection. I imported missingno to visually check for missing values. I imported train\_test\_split for data splitting and model testing. I imported variance\_inflation\_factor from statsmodels to check VIF scores. I imported classification report and confusion matrix from sklearn.metrics to get outputs for each. I imported python\_version to show the version I am working with. I imported SelectKBest for variable reduction. I imported StandardScaler to scale the data. I imported KNeighborsClassifier for KNN model creation. I imported roc\_curve and roc\_auc\_score to get outputs for ROC and AUC. I imported warnings to ignore the warnings.

# **Part III:**

**C1:**

My data preparation goals follow the Data Analytics Life Cycle. One important preprocessing goal is to scale or standardize the data. For KNN this step is crucial for the models accuracy. If the features have different scales, the model will be less accurate and give more significance to the variables of larger scale.

**C2:**

Below are the summary statistics for the target variable and all predictor variables including screenshots of output showing the count of instances (10,000), mean (average), standard deviation (amount of variation of a set of data), minimum (least amount), quartile ranges (measure of variability of the data), maximum (largest amount) for continuous variables. Unique values, and value counts for categorical variables.

**Churn (Target Variable):** Datatype is object, qualitative, categorical, and has no null values. The value reflects whether the customer discontinued service within the last month (Yes, No).

Graphical user interface, text

Description automatically generated

**Population:** Datatype is int64, quantitative, continuous, and has no null values. The value reflects the population within a mile radius of the customer, based on census data (range from 0 to 111,850).

Table

Description automatically generated

**Area:** Datatype is object, qualitative, categorical, and has no null values. The value reflects the area type (Rural, Urban, Suburban), based on census data.

Table

Description automatically generated

**Children:** Datatype is float64, quantitative, continuous, and has no null values. The value reflects the number of children in the customer’s household as reported in sign-up information (ranging from 0 to 10).

Table

Description automatically generated

**Age:** Datatype is float64, quantitative, continuous, and has no null values. The value reflects the age of the customer as reported in sign-up information (ranging from 18 to 89).

Table

Description automatically generated

**Income:** Datatype is float64, quantitative, continuous, and has no null values. The value reflects the annual income of the customer (or invoiced person) as reported at time of sign-up (ranging from roughly 348 to 258,900).

Text

Description automatically generated

**Marital:** Datatype is object, qualitative, categorical, and has no null values. The value reflects the marital status of the customer as reported in sign-up information (Divorced, Married, Never Married, Separated, Widowed).

Text

Description automatically generated

**Gender:** Datatype is object, qualitative, categorical, and has no null values. The value reflects the customers self-identification as male, female, or nonbinary.

Text

Description automatically generated

**Outage\_sec\_perweek:** Datatype is float64, quantitative, continuous, and has no null values. The value reflects the average number of seconds per week of system outages in the customer’s neighborhood (range from roughly 0.1 to 21.2).

Text

Description automatically generated

**Email:** Datatype is float64, quantitative, continuous, and has no null values. The value reflects the number of emails sent to the customer in the last year (range from 1 to 23).

Table

Description automatically generated with medium confidence

**Contacts:** Datatype is float64, quantitative, discrete, and has no null values. The value reflects the number of times the customer contacted technical support (range from 0 to 7).

Table

Description automatically generated

**Yearly\_equip\_failure:** Datatype is float64, quantitative, discrete, and has no null values. The value reflects the number of times customers equipment failed and had to be reset/replaced in the past year (range from 0 to 6).

Table

Description automatically generated

**Techie:** Datatype is object, qualitative, categorical, and has no null values. The value reflects whether the customer considers themselves technically inclined (Yes, No).

Text

Description automatically generated

**Contract:** Datatype is object, qualitative, categorical, and has no null values. The value reflects the contract term of the customer (Month-to-month, One Year, Two Year).

Text

Description automatically generated

**Port\_modem:** Datatype is object, qualitative, categorical, and has no null values. The value reflects whether the customer has a portable modem (Yes, No).

**Text

Description automatically generated**

**Tablet:** Datatype is object, qualitative, categorical, and has no null values. The value reflects whether the customer owns a tablet such as iPad, Surface, etc. (Yes, No).

**Graphical user interface, text

Description automatically generated**

**InternetService:** Datatype is object, qualitative, categorical, and has no null values. The value reflects the customers internet service provider (DSL, Fiber Optic, or None).

**Text

Description automatically generated**

**Phone:** Datatype is object, qualitative, categorical, and has no null values. The value reflects whether the customer has a phone service (Yes, No).

**Text

Description automatically generated**

**Multiple:** Datatype is object, qualitative, categorical, and has no null values. The value reflects whether the customer has multiple lines (Yes, No).

Text

Description automatically generated

**OnlineSecurity:** Datatype is object, qualitative, categorical, and has no null values. The value reflects whether the customer has an online security add-on (Yes, No).

**Text

Description automatically generated**

**DeviceProtection:** Datatype is object, qualitative, categorical, and has no null values. The value reflects whether the customer has device protection add-on (Yes, No).

**Text

Description automatically generated**

**TechSupport:** Datatype is object, qualitative, categorical, and has no null values. The value reflects whether the customer has a technical support add-on (Yes, No)

**Text

Description automatically generated**

**StreamingTV:** Datatype is object, qualitative, categorical, and has no null values. The value reflects whether the customer has streaming TV (Yes, No).

**Text

Description automatically generated**

**StreamingMovies:** Datatype is object, qualitative, categorical, and has no null values. The value reflects whether the customer has streaming movies (Yes, No).

**Graphical user interface, text

Description automatically generated**

**PaperlessBilling:** Datatype is object, qualitative, categorical, and has no null values. The value reflects whether the customer has paperless billing (Yes, No).

Text

Description automatically generated

**PaymentMethod:** Datatype is object, qualitative, categorical, and has no null values. The value reflects the customer’s payment Method (Bank Transfer, Credit Card, Electronic Check, Mailed Check).

Text

Description automatically generated

**Tenure:** Datatype is float64, quantitative, continuous, and has no null values. The value reflects the number of months the customer has stayed with the provider (range roughly from 1 to 72).

Table

Description automatically generated with medium confidence

**MonthlyCharge:** Datatype is float64, quantitative, continuous, and has no null values. The value reflects the amount charged to the customer monthly. This is an average for the customer. For brand new customers, this value is the average for other customers who fit the new customers profile (range from roughly 80 to 290).

Text

Description automatically generated

**Bandwidth\_GB\_Year:** Datatype is float64, quantitative, continuous, and has no null values. The value reflects the average amount of data used, in GB, in a year by the customer (range from roughly 155 to 7159).

Text

Description automatically generated

**C3:**

I imported the churn.csv file using pandas.

Text

Description automatically generated

Next, I checked the dataset shape.

Graphical user interface, text, application

Description automatically generated

There are zero null values, 10,000 rows, and 50 columns.

A picture containing chart

Description automatically generated

I checked for duplicated rows and found there were none.

Graphical user interface, text, application

Description automatically generated

I made a copy of the original to preserve it.

Text

Description automatically generated

I made a new dataset that included the variables or columns that I used for the analysis.

Text, letter

Description automatically generated

I checked the uniqueness of each of those chosen variables for validity. I did this for all variables included in the initial model but am just showing one code example.

Graphical user interface, text, application

Description automatically generated

I used z-scores to identify outliers in the continuous variables, set the outliers equal to nan, then used median or mean imputation for replacement. I followed this same process for all continuous variables but am just showing one example.

Table

Description automatically generated

Chart, histogram

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Text

Description automatically generated

I made a new dataframe (df2) that had been cleaned and includes only the variables being utilized for the KNN analysis.

Text

Description automatically generated

For the data wrangling phase I used ordinal encoding for the variables with Yes/No values and set them to 0 for No and 1 for Yes.

Text, letter

Description automatically generated

Then I used pd.get\_dummies with k-1 to encode the nominal variables.

Text

Description automatically generated

The dataset is now clean and prepared for KNN analysis. There are more steps that I took to reduce the number of variables in the model. They are shown in part D1.

**C4:**

The Jupyter notebook file (.ipynb format) containing the complete annotated code along with a copy of the cleaned dataset named ‘Eric\_Colwell\_clean\_dataset’ in .csv format for this project will be uploaded with the submission.

# **Part IV:**

**D1:**

I used the SelectKBest method to filter out variables with p-values greater than 0.05 and reduced the initial model from 39 predictor variables down to 17.

Table

Description automatically generated

Then I checked for high correlation between variables. Tenure and Bandwidth\_GB\_Year had a high correlation, so I removed Tenure because it had the higher correlation to Churn.

A picture containing graphical user interface

Description automatically generated

Next, I check the variance inflation factors for the predictor variables that remained. Due to high VIF scores I removed MonthlyCharge and Contract\_Two year.

Text

Description automatically generated with medium confidence

After removing two of the variables with high VIF scores I checked it again. Now all VIF scores are less than 10 and there are 14 predictor variables.

Text

Description automatically generated with low confidence

Then I used the StandardScaler to scale the data that will be used in the final model.

Graphical user interface, text, application, email

Description automatically generated

Next, I split the data into 70% training and 30% testing datasets. These files will be uploaded along with the submission as .csv files named Eric\_Colwell\_train\_dataset and Eric\_Colwell\_test\_dataset. The code snippet for splitting the data is shown below.

Graphical user interface, text, application

Description automatically generated

**D2:**

I used the confusion matrix and classification report to analyze the model. The screenshots below represent training and testing accuracy with neighbors set from 1 to 60.

Text, letter

Description automatically generated

Chart, histogram

Description automatically generated

Table

Description automatically generated

I ran the model with 3 different settings for neighbors. The neighbors settings I ran were 6, 8, and 10. Screenshots of the confusion matrix and classification reports for each below:

Neighbors equal to 6:

Table

Description automatically generated with low confidence

Neighbors equal to 8:

Table

Description automatically generated with medium confidence

Neighbors equal to 10:

Table

Description automatically generated with low confidence

For the 3 different models we see a slight improvement in accuracy (0.84 to 0.85) and minimal changes in the confusion matrix. For the last model, the confusion matrix showed that there were 2091 true positives, 114 false positives, 335 false negatives, and 460 true negatives.

**D3:**

Code snippets used to perform the classification analysis were shown in the screenshots above. I will upload a .ipynb Jupyter notebook with the complete code for this project along with the rest of the submission.

**Part V:**

**E1:**

The accuracy of the last model run where neighbors = 10 was 85 percent. Accuracy measures the true positives and true negatives. This means that the model correctly predicted the test data 85% of the time. The AUC (Area Under the Curve) ROC (Receiver Operating Characteristic) curve is used to visualize the performance of the model. AUC measures true positives and false positives. The graph shows the plotting of the true positive rate and the false positive rate. This model achieved an AUC score of 0.9144. A higher AUC value means a better performing model. An AUC score of 1.0 would be a model of 100 percent true predictions (Kharwal, 2021).

Graphical user interface, text, application

Description automatically generated

Chart

Description automatically generated

Graphical user interface, text

Description automatically generated

**E2:**

KNN is a simple supervised machine learning algorithm that is used to solve classification problems. With an accuracy score of 85 percent and an AUC score of 0.9144, there is room for improvement. This could possibly be achieved through using different methods for feature reduction and using different parameters for neighbors. Also adjusting the size of the training set for the model could cause the model to perform better.

**E3:**

The Churn dataset has a total of 10,000 observations. 7350 are observations where customers are not going to churn. 2650 are observations where customers are going to churn. This is a large imbalance of the data and is a limitation. This model may perform poorly on imbalanced data and could cause it to get the less common class, customers that are going to churn, wrongly predicted.

**E4:**

Based on the output from this analysis the company could use this model to predict customers who are at risk of churn. I would recommend further analysis with different feature reduction methods and hyperparameter tuning for the KNN model. This along with balancing the dataset and adjustments to the training dataset size could cause the model to perform much better allowing the company to predict which customers more accurately are at risk of churn.

**Part V:**

**F:**

The Panopto video recording URL will be uploaded as part of the submission.

References

Elleh, F. (2022). D209 Data Mining I Webinar. Retrieved from <https://wgu.hosted.panopto.com/Panopto/Pages/Viewer.aspx?id=b73b6274-ef01-4d1b-a59f-aed100228a93>

Jaroli, H. (April 8, 2019). Retrieved from

<https://datascienceplus.com/k-nearest-neighbors-knn-with-python/>

Kharwal, A. (February 3, 2021). Retrieved from <https://thecleverprogrammer.com/2021/02/03/roc-and-auc-in-machine-learning/>