Performance Assessment: Classification Analysis

Gabriela Howell

Master of Science Data Analytics, Western Governors University

D209 – Data Mining

Professor Elleh

June 9, 2024

Part I: Research Question

A. Describe the purpose of this data mining report by doing the following:

1. Propose one question relevant to a real-world organizational situation that you will answer using one of the following classification methods:

Using the Medical dataset my research question is, " Which factors most significantly contribute or relate to patients diagnosed with anxiety?" This is an issue among Americans. According to a 2024 chart from the CDC, 17.4% of adults in America have anxiety. The method of classification I will be using is K-Nearest Neighbors classification to answer this question.

2. Define one goal of the data analysis. Ensure that your goal is reasonable within the scope of the scenario and is represented in the available data.

The purpose of this data analysis is to find out which factors from the medical dataset are most closely linked to anxiety diagnoses in patients. I will explore different attributes in the dataset to identify the strongest correlations with anxiety. By doing this, I hope to gain insights that can improve early detection and intervention strategies for anxiety, that way healthcare facilities can efficiently help those with anxiety.

Part II: Method Justification

B. Explain the reasons for your chosen classification method from part A1 by doing the following:

1. Explain how the classification method you chose analyzes the selected data set. Include expected outcomes.

K-Nearest Neighbor (KNN) classification is used to detect similar attributes in the closest neighbors to a target variable. Which then determines the most frequent classification value among its k-neighbors and outputs a classification prediction based on those variables. The results are expected to show how the target variable relates to the k-neighbors and provide accuracy summaries for the model. In this case, for anxiety, the goal is to identify which factors significantly influence anxiety.

2. Summarize one assumption of the chosen classification method.

The key assumption of K-Nearest Neighbor classification is encapsulated by the phrase "birds of a feather flock together," meaning similar items are located near each other, in this instance on a graph. The model predicts the label of an unknown point by examining the labels of nearby data points. If similar points are not close to each other, the KNN model will be useless because the neighbors won't aid in classifying the unknown point.

3. List the packages or libraries you have chosen for Python or R, and justify how each item on the list supports the analysis.

I chose Python to compose my K- Nearest Neighbor modeling. Which consists of several packages.

- Pandas: Used to load and analyze my dataset

- NumPy: Used for numerical operations to fine-tune array calculations
- Seaborn: Used for visualizations such as boxplots for outliers
- Matplotlib: Used for visualization in data distribution and plotting ROC
- Scikit-learn: Used for machine learning, such as;
    - train_test_split: Used to split the data into training and testing sets.
    - GridSearchCV: Used for hyperparameter fine-tuning.
    - StandardScaler: Used for standardizing the features.
    - KNeighborsClassifier: Used for K-Nearest Neighbors classification.
    - confusion_matrix, classification_report, accuracy_score, roc_auc_score, roc_curve: Metrics used for overall model evaluation.
    - SelectKBest, f_classif: Used for feature selection to help with
- Statsmodels.stats.outliers_influence: variance_inflation_factor: Used to check for multicollinearity in the data.

Each of these libraries is the bread and butter of making KNN classification work in Python.

Part III: Data Preparation

C. Perform data preparation for the chosen data set by doing the following:

1. Describe one data preprocessing goal relevant to the classification method from part A1.

A classification goal before data processing is to standardize the features. This requires placing all variables on an equal scale. To do this I will need to strip off the mean and scale to unit variance. Standardization of the variables is vital for KNN classification this is because it allows the algorithm to accurately calculate the distances between data points, that way identifying

nearest neighbors is possible. Without standardization, features with larger scales could disproportionately influence the results, leading to inaccurate classifications.

2. Identify the initial data set variables that you will use to perform the analysis for the classification question from part A1, and classify each variable as numeric or categorical.

- Nominal Categorical Variables: Marital, Gender, ReAdmis, Soft_drink, Initial_admin, HighBlood, Stroke, Arthritis, Diabetes, Hyperlipidemia, BackPain, Allergic_rhinitis, Reflux_esophagitis, Asthma, Services
- Numeric Variables:
    - Discrete: Population, Children, Doc_visits, Full_meals_eaten, vitD_supp, Item1, Item2, Item3, Item4, Item5, Item6, Item7, Item8
    - Continuous: Age, Income, VitD_levels, Initial_days, TotalCharge, Additional_charge

3. Explain each of the steps used to prepare the data for the analysis. Identify the code segment for each step.

The steps for preparing the data for analysis are as follows: first, assemble the relevant variables. Then, identify and handle any duplicated and missing values. Additionally, examine and address any outliers. The cleaning process began with the separation of the target variable 'Anxiety' from the feature dataset 'X', which included dropping irrelevant columns such as identifiers and geographical details. Numerical and categorical data were then analyzed separately, with summary statistics calculated for numerical features and frequency counts obtained for categorical ones. Boolean values were converted to integers, and unique values were inspected across all columns to ensure data integrity. Nominal categorical variables underwent

one-hot encoding to prepare them for model training, while discrete and continuous numerical variables were identified for further analysis.

Next, use StandardScaler to standardize all the explanatory variables. This makes sure that features with bigger scales do not dominate distance calculations in algorithms like K-Nearest Neighbors (KNN), K-Means clustering, and Principal Component Analysis (PCA), leading to respectable results (Scikit-learn, 2024). Then using SelectKBest, significant features for predicting anxiety were identified, with 'TotalCharge' and 'Area_Urban' showing significant p-values. Next, to address multicollinearity, the Variance Inflation Factor (VIF) was processed for selected features, confirming no significant issues. Finally, the feature selection process was completed, and the dataset was prepared for subsequent modeling tasks. Finally, verify that the data has been properly updated.

4. Provide a copy of the cleaned data set.

The cleaned dataset will be provided and named 'D209_part1_clean.csv'.

Part IV: Analysis

D. Perform the data analysis and report on the results by doing the following:

1. Split the data into training and test data sets and provide the file(s).

The saved files are named: 'X_train_task1.csv', 'X_test_task1.csv', 'Y_train_task1.csv', and 'Y_test_task1.csv'.

2. Describe the analysis technique you used to appropriately analyze the data. Include screenshots of the intermediate calculations you performed.

Several steps are involved in analyzing the data. First, I clean up the dataset to remove any errors. Then, I split the data into two parts: one for training the model and one for testing its accuracy.

```
8]: # Split the dataset into training and testing sets
    X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size=0.2, random_state=42, stratify=Y)
```

Next, using GridSearchCV to find the best number of neighbors for our model, which turns out to be 46. After that, I trained the KNN model on the standardized data.

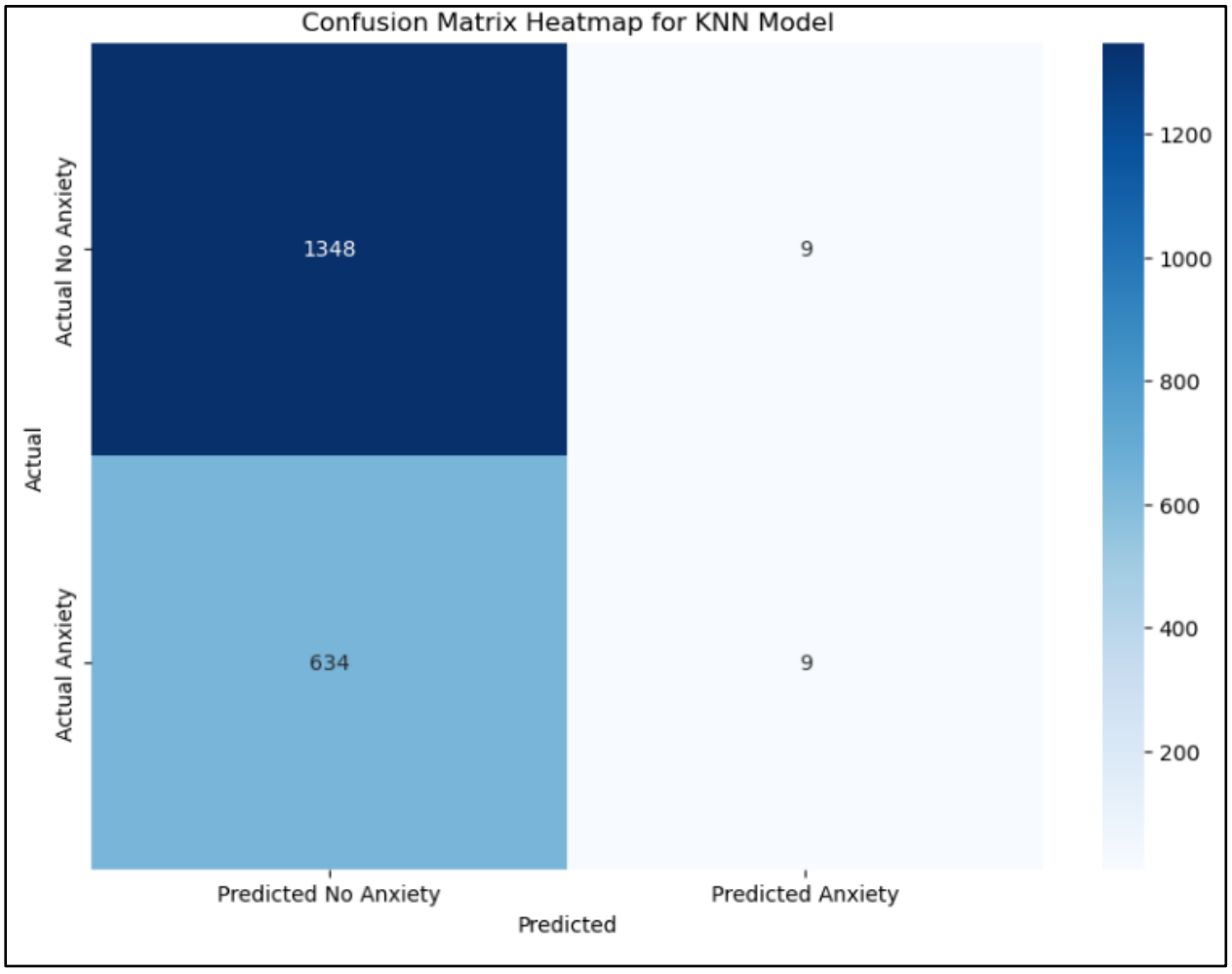
```
]: # Determine what is the best number of neighbors to use for KNN classification
   param_grid = {'n_neighbors' : np.arange(1, 50)}
   # Instantiate the KNeighborsClassifier object
   knn = KNeighborsClassifier()
   # Use GridSearchCV object, searching across the provided parameter grid and 5 fold cross validation
   knn_cv = GridSearchCV(knn, param_grid, cv=5)
   # Fit to training data
   knn_cv.fit(X_train, Y_train)
   # Find best parameter from GridSearchCV
   knn_cv.best_params_

]: {'n_neighbors': 46}
```

To see how well the model works, I compare its predictions to the actual values using a confusion matrix. To better understand I included a Heatmap for the confusion matrix. I also use

a ROC curve to see how good the model is at detecting true positives versus false positives.

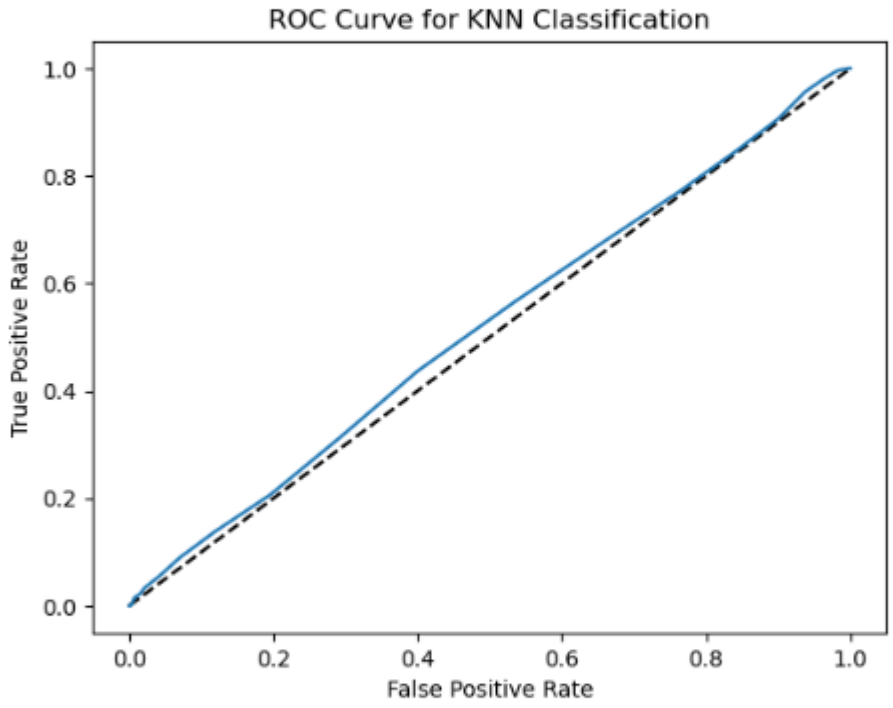
```
: # Generate confusion matrix and accuracy score of model
final_matrix = confusion_matrix(Y_test, Y_pred_test)
print("The confusion matrix for this KNN model:")
print("Predicted No Anxiety | Predicted Anxiety")
print(f"                {final_matrix[0]} Actual No Anxiety")
print(f"                {final_matrix[1]} Actual Anxiety")
print(f"The training accuracy of this KNN classification is {train_accuracy}.")
print(f"The testing accuracy of this KNN classification model is {test_accuracy}.")
```

```
The confusion matrix for this KNN model:
Predicted No Anxiety | Predicted Anxiety
                [1348    9] Actual No Anxiety
                [634   9] Actual Anxiety
The training accuracy of this KNN classification is 0.6795.
The testing accuracy of this KNN classification model is 0.6785.
```



Additionally, I calculate the Area Under the Curve (AUC) score to measure the model's overall performance. Lastly, a classification report gives a microscope look into how well the model predicts both classes of anxiety.

```
# Generate AUC score and print
Y_pred_prob = knn.predict_proba(X_test)[:, 1]
fpr, tpr, thresholds = roc_curve(Y_test, Y_pred_prob)
plt.plot([0, 1], [0, 1], 'k--')
plt.plot(fpr, tpr)
plt.xlabel('False Positive Rate')
plt.ylabel('True Positive Rate')
plt.title('ROC Curve for KNN Classification')
plt.show()
print(f"The Area Under the Curve (AUC) score is: {roc_auc_score(Y_test, Y_pred_prob)}\n")
print(classification_report(Y_test, Y_pred_test))
```




```
The Area Under the Curve (AUC) score is: 0.518839013421565

              precision    recall  f1-score   support

           0       0.68      0.99      0.81      1357
           1       0.50      0.01      0.03       643

    accuracy                           0.68      2000
   macro avg       0.59      0.50      0.42      2000
weighted avg       0.62      0.68      0.56      2000
```

3. Provide the code used to perform the classification analysis from part D2.

The Python code will be provided in the attached file named Gab-D209_Part1.ipynb.

Part V: Data Summary and Implications

E. Summarize your data analysis by doing the following:

1. Explain the accuracy and the area under the curve (AUC) of your classification model.

The training model has an accuracy of 67.95%, whereas the testing accuracy is 67.85%. Between the two you can see the training did slightly better. The Area Under the Curve (AUC) score, used to assess the model's ability to distinguish between classes, was 0.519. These results suggest that while the model performs moderately well in predicting anxiety levels, there is room for improvement, particularly in its ability to differentiate between anxiety and non-anxiety cases.

2. Discuss the results and implications of your classification analysis.

The results from the KNN model's performance metrics reveal noteworthy insights. From one perspective, it displays higher precision, recall, and F1-score for the 'No Anxiety' group than the 'Anxiety' group. However, the confusion matrix highlights a concerning number of false negatives (634) for anxiety cases, indicating that the model ignores several cases of anxiety. Furthermore, the low AUC score implies that the model struggles to distinguish between anxiety and non-anxiety cases effectively. Overall, while the model excels at identifying those without anxiety, its limitations in detecting anxiety cases.

3. Discuss one limitation of your data analysis.

One limitation in my data analysis is the limited predictive power due to the imbalance in the dataset, which contains more occurrences of 'No Anxiety' compared to 'Anxiety'. This imbalance is problematic because K Nearest Neighbors (KNN) is sensitive to unrelated variables, increasing computation time. The MyEducator article highlights that, "Computational effort of the algorithm increases greatly as more predictors, p, are considered and when the number of training records increase. The algorithm must compute the distance and find the nearest neighbors in all the training data for each prediction. This takes time and can be especially slow when there are a large number of training records that must be examined for each record to be predicted." This imbalance can lead to biased predictions and increased computational effort especially when handling large datasets.

4. Recommend a course of action for the real-world organizational situation from part A1 based on your results and implications discussed in part E2.

Since the model showed more data on no anxiety, I think it would be great to gather more information on anxiety. So that hospitals and healthcare providers can use the analysis insights to create focused interventions or support systems for patients with anxiety. These may involve specialized mental health services, educational resources, or preventive measures. By analyzing more data, organizations can adjust their strategies and allocate resources to recognize similarities among patients and learn more about anxiety.

Part VI: Demonstration

F. Provide a Panopto video recording that includes a demonstration of the functionality of the code used for the analysis and a summary of the programming environment.

My video link is here: Gabriela_Howell_D209_Task1

G. Record the web sources used to acquire data or segments of third-party code to support the analysis. Ensure the web sources are reliable.

GeeksforGeeks. (2023, April 18). *Label encoding in Python*. https://www.geeksforgeeks.org/ml-label-encoding-of-datasets-in-python/

Machine learning with scikit-learn: Python. campus.datacamp.com. (n.d.). https://campus.datacamp.com/courses/supervised-learning-with-scikit-learn/classification-1?ex=1

11.5 videos. MyEducator. (n.d.-c). https://app.myeducator.com/reader/web/1421a/11/j571v/

H. Acknowledge sources, using in-text citations and references, for content that is quoted, paraphrased, or summarized.

Centers for Disease Control and Prevention. (2024, May 16). *Mental health - household pulse survey - covid-19*. Centers for Disease Control and Prevention. https://www.cdc.gov/nchs/covid19/pulse/mental-health.htm

6.3. Preprocessing Data. scikit. (n.d.). https://scikit-learn.org/stable/modules/preprocessing.html#preprocessing-scaler

11.4 advantages and disadvantages of KNN. MyEducator. (n.d.-b). https://app.myeducator.com/reader/web/1421a/11/q07a0/