JANUARY 17, 2024

PERFORMANCE ASSESSMENT

D209 TASK 1

ROBERT PATTON
WESTERN GOVERNORS' UNIVERSITY
Rpatt33@wgu.edu

Table of Contents

Part I: Research Question 2

A. Purpose of Data Mining Report 2
1. Research Question 2
2. Goal of Analysis 2

Part II: Method Justification 2

B. Reasoning for Chosen Classification Method 2
1. Classification Method (KNN) 2
2. One Assumption of KNN 2
3. Python Packages 2

Part III: Data Preparation 3

C. Data Preparation 3
1. One Data Preprocessing Goal 3
2. Data Set Identification 3-4
3. Steps for Analysis 4-5
4. Copy of Code 5

Part IV: Analysis 5

D. Data Analysis Report 5
1. Data Training and Split Sets 5
2. Analysis Technique 5-6
3. Code for Classification Analysis 5-6

Part V: Data Summary and Implications 6

E. Summarize Data Analysis 6
1. Accuracy and AUC Explanation 6
2. Results and Implications 7
3. One Limitation of Data Analysis 7
4. Recommended Course of Action 7

Part VI: Demonstration 8

F. Panopto Video Link 8
G. Web Sources 8
H. Other Sources 8

Part I: Research Question

A. Purpose of Data Mining Report

1. Utilizing the Naïve Bayes method, can we predict hospital readmission based on the health conditions reported by patients during their initial hospital admission?
2. The goal of this data analysis is to find out if a hospital can predict readmission based on specific medical conditions. This is of value to a hospital because extensive readmission rates tend to impact hospital revenue, deter current and future patients from seeking services, and can cause fines based on criteria via the Center for Medicare and Medicaid Services (Upadhyay et al. 2019).

Part II: Method Justification

B. Reasoning of Chosen Classification Method

1. The data that we are working with has 10,000 records, which is a relatively large data set. The Naïve Bayes method works very well for making predictions with larger sets of data. This will allow us to combine known data with new data to make predictions on whether a patient is more likely to be readmitted, based on the data we have on patients recently admitted. The Naïve Bayes method, a probabilistic classification method, analyzes our data by assuming all variables are independent of one another, that is, one variable does not determine another variable. For example, if a patient has answered yes to being overweight, that does not mean they are also going to have high blood pressure. In a Naïve Bayes model, all predictor variables have an equal effect on the outcome. With those characteristics, the Naïve Bayes algorithm calculates the probability of a particular class for a given set of features and selects the class with the highest probability as the predicted class (Saini 2023).
2. One assumption of Naïve Bayes is that the method assumes conditional independence between all independent variable features (Rice, 2014).
3. Python packages used are as follows:

   Pandas: Used to load in the CSV file.

   NumPy: Used for mathematical operations and array functions.

   Matplotlib.pyplot: Used to create visualizations.

   Seaborn: Another visualization tool.

Sklearn.model_selection.train_test_split: Used to split the data into training and testing sets.
Sklearn.naive_bayes.multinomialNB: Used to import the model for our analysis
Sklearn.metrics: Used to construct a confusion matrix, ROC curve with AUC, a classification report on the model, and to calculate accuracy, recall, and precision scores.

Part III: Data Preparation

C. Data Preparation

1. To prepare for implementing the Naïve Bayes method for predicting readmission, we first need to clean the data we are working with. This includes looking for any missing values or duplicates and ensuring that all our variables have numerical values using one-hot encoding.
2. Initial data classification is as follows:

ReAdmis	**Categorical**
Children	**Numeric**
Age	**Numeric**
Income	**Numeric**
VitD_levels	**Numeric**
Doc_visits	**Numeric**
vitD_supp	**Numeric**
HighBlood	**Categorical**
Stroke	**Categorical**
Overweight	**Categorical**
Arthritis	**Categorical**
Diabetes	**Categorical**
Hyperlipidemia	**Categorical**
BackPain	**Categorical**
Anxiety	**Categorical**
Allergic_rhinitis	**Categorical**

Reflux_esophagitis	**Categorical**
Asthma	**Categorical**
Initial_days	**Numeric**
_Emergency Admission	**Categorical**
_Observation Admission	**Categorical**
Risk_Low	**Categorical**
Risk_Medium	**Categorical**
_Male	**Categorical**
_Nonbinary	**Categorical**

3. Steps for preparing the data are as follows:
    - **#Import libraries and packages**

      import pandas as pd

      import numpy as np

      import matplotlib.pyplot as plt

      %matplotlib inline

      import seaborn as sns

      from sklearn.model_selection import train_test_split

      from sklearn.naive_bayes import MultinomialNB

      from sklearn.metrics import classification_report

      from sklearn.metrics import confusion_matrix

      from sklearn.metrics import (accuracy_score, recall_score, precision_score, confusion_matrix, roc_curve, auc)
    - **#Read in the CSV file**

      mdf= pd.read_csv('/Users/robertpatton/Desktop/Desktop - Robert's MacBook Pro/D208 /D208_task2_clean.csv', index_col=0)
    - **#Examine the data info**

      mdf.info()
    - **#Examine if duplicate or missing values exist**

      mdf.isnull().sum()

      mdf.duplicated()

- **#Create a list of binary columns**
- Binary_columns= ['HighBlood', 'Stroke', 'Overweight', 'Arthritis', 'Diabetes', 'Hyperlipidemia', 'BackPain', 'Anxiety', 'Allergic_rhinitis', 'Reflux_esophagitis', 'Asthma']
- **#Replace binary columns yes/no with 0/1**
- mdf[Binary_columns]=mdf[Binary_columns].replace({'Yes': 1, 'No': 0})
- **#Re-express Initial_admin, complication risk, and Gender columns to numerical values and use one-hot encoding**
- mdf=pd.get_dummies(mdf, columns=['Initial_admin'], prefix='Initial_admin', drop_first=True)
- mdf=pd.get_dummies(mdf, columns=['Complication_risk'], prefix='Comp_risk', drop_first=True)
- mdf=pd.get_dummies(mdf, columns=['Gender'], prefix='Gender', drop_first=True)
- mdf.head()
- **#Define dependent and predictor variables**
  X=mdf.drop(columns='ReAdmis')
  y=mdf['ReAdmis']
- **#Look for correlation in variables with heat map**
  coorl= mdf.corr()
  fig, ax = plt.subplots(figsize=(20,15))
  sns.heatmap(coorl, annot=True, ax=ax)
  plt.title("Correlations - Predictors and Response",fontsize=32)
  plt.show()

4. *Copy of cleaned data attached to upload.

Part IV: Analysis

D. Data Analysis Report

1. The following code was used to split the dataset into train and test sets.

   #Split data into training and test sets

   X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=16)

   #Train the model with the naive Bayes classification method

mdf_clf=MultinomialNB()

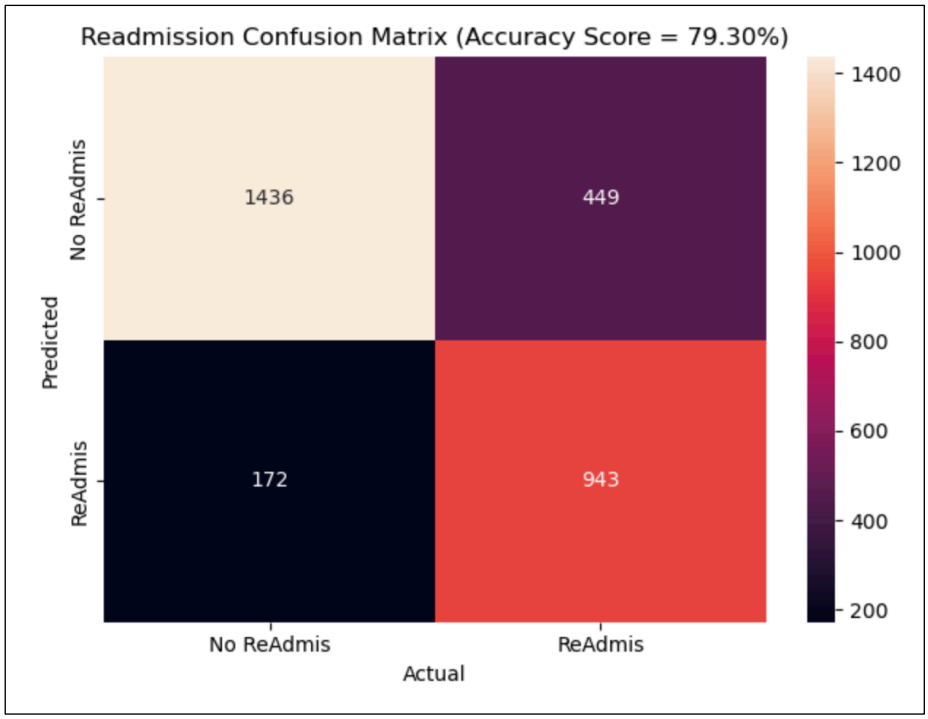
mdf_clf.fit(X_train, y_train)

#Evaluate the model and get classification report

y_pred=mdf_clf.predict(X_test)

*The files are attached to the submission for the split data.

2. With the data split into train and test sets, I was able to then use the Naïve Bayes method to help classify the potential for hospital readmission. Implementing a 70/30 test split, I was able to predict the readmission of patients based on medical conditions as reported during initial admission. A confusion matrix and classification report were constructed to visualize the output and predictions of the model, they are depicted below.



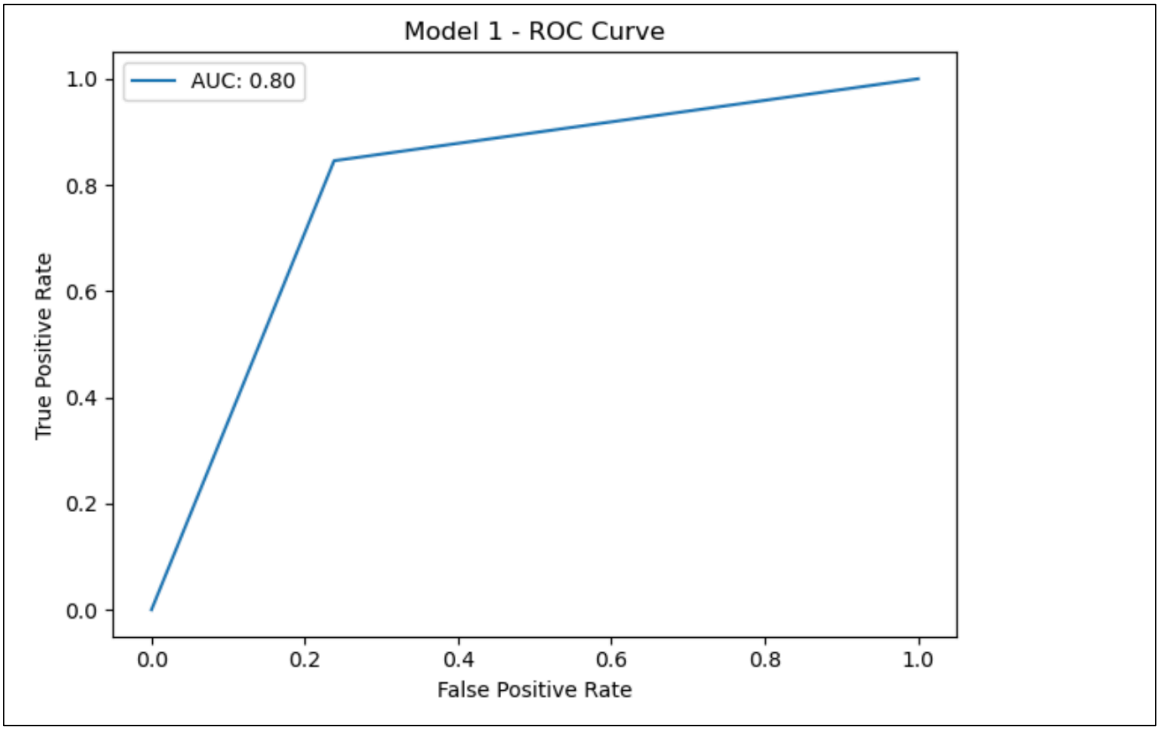

```
#Evaluate the model and get classification report
y_pred=mdf_clf.predict(X_test)
score = accuracy_score(y_test, y_pred)
print(f"Accuracy Score = {score * 100:.2f}%)")
print(f"Recall (Sensitivity): {recall_score(y_test, y_pred)* 100:.2f}%")
print(f"Specificity: {recall_score(y_test, y_pred, pos_label=0) * 100:.2f}%")
print(f"Precision: {precision_score(y_test, y_pred) * 100: .2f}%")
```

```
Accuracy Score = 79.30%)
Recall (Sensitivity): 84.57%
Specificity: 76.18%
Precision:  67.74%
```

Part V: Data Summary and Implications

E. Summarize the Data Analysis

1. Evaluating our confusion matrix above, of the 1,392 patients that were predicted as readmitted, 943 were correctly predicted for readmission while 449 of those were incorrectly predicted, bringing our precision rate to 67.74%. Of the 1,885 non-readmitted patients predicted, 1,436 were correctly predicted to not have readmission while 449 were incorrectly classified, bringing our specificity to 76.18%. For the recall, of the 1,115 predicted readmitted patients, 943 were correct and 172 were incorrectly predicted, bringing the recall value to 84.57%. Accuracy, as seen in the confusion matrix above, is calculated as follows: True Positives (943) + True Negatives (1436)/ True Positives (943) + True Negatives (1436) + False Positives (449) + False Negatives (172) = 79.30% accuracy of predicting readmission. The area under the curve looks at the true positive rate (sensitivity) versus the false positive rate (specificity). An AUC score is an overall diagnostic tool for summarizing the accuracy of a test (Mandrekar 2015). In a ROC model, an ideal curve would follow the y-axis straight up to 1.0 and over to 1.0 on the x-axis, indicating 100% accuracy. For this model, the area under the curve is at 80%, indicating an 80% chance of predicting readmission. This is still a well-performing model.



2. After determining accuracy, I created a classification report using recall, specificity, and precision to further examine the model's performance. Recall tells an analyst how well the model predicts positives out of all positive values, with the value of recall being 84.57%. Specificity tells an analyst how well the model predicts negative values out of all negatives, with the value coming to 76.18%. Based on the model's recall and specificity results, along with the accuracy, it can be said that the model does well at predicting whether or not patients will be readmitted based on the medical conditions stated at initial admission. All of these metrics combined, along with our ROC curve and AUC results, this model performed very well overall.
3. The model has produced some fairly accurate results and signifies that it can predict patient readmissions. However, hospitals tend to see varying degrees of admissions at different times of the year. We may need to collect more data on health conditions, initial admission rates during seasonal changes, and other patient information that could make a stronger case for which patients are at higher risk for readmission. For example, during the winter months when cold and flu season arrives, hospitals see a greater influx in emergency room visits and hospital admissions. Does cold and flu influence or trigger certain health conditions to become worse, requiring admission in the winter months versus other times of the year? These are some things a data analyst and hospital leaders should take into consideration.
4. To follow up on limitations, a recommended course of action would be to closely monitor trends in admission. Because the Naïve Bayes method uses known data to new data to make predictions in large datasets, an analyst and hospital corporation must ensure they are considering new cases coming in as there are many different health conditions a person may have. This dataset seems to list just the most common health conditions, so it may be important to gather data on other health conditions as well. All of that combined will allow the continued accuracy and potentially even stronger accuracy to continue in making predictions on readmission, improving the overall quality of care a patient receives upon initial admission.

Part VI: Demonstration

F. A link to my Panopto video is uploaded to the submission.

G. Third-Party Sources

Awan, A. A., & Navlani, A. (2023, March 3). Naive Bayes classifier tutorial: With Python Scikit-Learn. DataCamp. https://www.datacamp.com/tutorial/naive-bayes-scikit-learn

Saini, A. (2023, July 20). Naive Bayes Algorithm: A complete guide for data science enthusiasts. Analytics Vidhya. https://www.analyticsvidhya.com/blog/2021/09/naive-bayes-algorithm-a-complete-guide-for-data-science-enthusiasts/

Mandrekar, J. (2015, November 20). Receiver operating characteristic curve in diagnostic test assessment. Journal of Thoracic Oncology. https://www.sciencedirect.com/science/article/pii/S1556086415306043#:~:text=AUC%20can%20be%20computed%20using%20the%20trapezoidal%20rule.&text=In%20general%2C%20an%20AUC%20of,than%200.9%20is%20considered%20outstanding

H. Acknowledged Sources

Rice, D. M. (2014). Chapter 4. In Calculus of thought: Neuromorphic logistic regression in cognitive machines (pp. 95–123). essay, Academic Press, an imprint of Elsevier.

Upadhyay, S., Stephenson, A. L., & Smith, D. G. (2019). Readmission Rates and Their Impact on Hospital Financial Performance: A Study of Washington Hospitals. Inquiry : a journal of medical care organization, provision, and financing, 56, 46958019860386. https://doi.org/10.1177/0046958019860386