Saemi Ramirez

D209 Data Mining I

PA2 – Predictive Analysis

7/16/2024

WGU

1. Describe the purpose of this data mining report
   1. My research question is “Is RandomForestRegressor the best model to predict the patients TotalCharge with the features determined by RandomizedSearchCV?” and I will be using RandomForestRegression.
   2. The goal of the data analysis is to identify the how accurate the model is for the research question with determined features by RandomizedSearchCV function. This involves examining various potential predictors and determining their impact on the likelihood of a high blood pressure diagnosis. The insights gained from this analysis can help in understanding the underlying causes, improving patient risk assessments, and developing targeted interventions for prevention and management.
2. Reasons for the chosen predictive method
   1. Random Forest is an ensemble machine learning technique that can perform both regression and classification tasks by utilizing multiple decision tress and a method called Bootstrap, Aggregation, or bagging. The core concept is to combine multiple decision trees to determine the final output, rather than relying on a single decision tree (Random). The parameter ‘n\_estimators’ specifies the number of decision trees in the model, ‘max\_depth’ sets the maximum depth of each tree, and ‘max\_features’ defines the maximum number of features considered when making a split (Beheshti).  
      The expected outcome is the predicted amount of TotalCharge of a given record based on the predictor features of that record.
   2. One of the assumptions for the Random Forest model is “there should be actual values in the feature variable; this leads to prediction of more accurate results rather than a random guess by the algorithm” (Charles).

|  |  |
| --- | --- |
| **Packages/Libraries** | **Purpose** |
| Pandas | To handle the dataset/dataframe |
| Numpy | To perform mathematical operations or values |
| Matplotlib.pyplot | To visualize the data via graphics |
| From sklearn.preprocessing import MinMaxScaler | To normalize the numeric values by reducing the size of the data to match the original form |
| From sklearn.model\_selection import train\_test\_split | To split the dataset into X, y, train and test |
| From sklearn.model\_selection RandomizedSearchCV | Optimize the hyperparameters of machine learning model |
| From sklearn.ensemble import RandomForestRegressor | Create RandomForest model |
| From sklearn.metrics import mean\_squared\_error | Calculates mean-squared-error between 2 series of true and predicted values |
| From sklearn.metrics import r2\_score | Computes how well the model predicts the outcome of the dependent variables |
| From sklearn.metrics import accuracy\_score | Computes an accuracy score |

1. Data preparation for the chosen dataset
   1. One of the methods used during the data preprocessing for KNN classification is one hot encoding. The get\_dummies function was conducted with setting the drop\_first argument true to avoid multicollinearity

|  |  |
| --- | --- |
| **Variable** | **Data Type** |
| Marital | Categorical |
| Gender | Categorical |
| Initial\_admin | Categorical |
| HighBlood | Categorical |
| Complication\_risk | Categorical |
| Overweight | Categorical |
| Arthritis | Categorical |
| Diabetes | Categorical |
| BackPain | Categorical |
| Anxiety | Categorical |
| Asthma | Categorical |
| Services | Categorical |
| Age | Numeric |
| Income | Numeric |
| VitD\_levels | Numeric |
| Initial\_days | Numeric |
| TotalCharge | Numeric |

* 1. Explain each of the steps to prepare the data for analysis
     1. df.info() – check the details on each column of the data set
     2. df.head() – check the first 5 rows of the dataset
     3. df.isnull().sum() – check the total number of missing values on each column
     4. df.duplicated() – check for any duplicated information in the data set
     5. df[‘Age’].describe() – check statistical information of the numeric values
     6. df[‘Marital’].value\_counts() – check the total number of each unique values
     7. initial\_model.head() – create another data frame with selected predictors and checking the first 5 rows of the information
     8. one\_hot = pd.get\_dummies(initial\_model, drop\_first=True) – conducted one hot encoding method using get\_dummies and dropping the first column
     9. one\_hot.info() – check the details on each column of one\_hot
     10. one\_hot.head() – checking the first 5 rows of the information in one\_hot
     11. one\_hot.rename() – update the invalid column names with space to underscore(\_)
     12. one\_hot.info() – check the column names
     13. for loop – update the data type of the columns from bool to int64
     14. one\_hot.info() – check the details on each column after data type changed
     15. one\_hot.head() – check the first 5 rows of the updated data set
     16. pd.DataFrame(MinMaxScaler().fit\_transform(one\_hot), columns=one\_hot.columns) – to normalize the numeric values to set min = 0 and max = 1
     17. final\_data.to\_csv() – save the data frame to csv format in local drive
  2. Submitted “final.csv”

1. Data analysis and report
   1. One\_hot\_scaled dataset was split to X\_train, X\_test, y\_train, y\_test with 70% of training set and 30% of test set. Each set are submitted as ‘XTest.csv’, ‘XTrain.csv’, ‘yTest.csv’, and ‘yTrain.csv’.
   2. Analysis Techniques used
      1. Converted the categorical variables into dummy variables after features selected.  
         A screenshot of a computer program

         Description automatically generated
      2. Split the data into 70% training set and 30% testing set and saved them.  
         A screenshot of a computer program

         Description automatically generated
      3. Instantiated the RandomForestRegressor object and identified optional values for parameters using hyperparameter tuning.  
         A screenshot of a computer program

         Description automatically generated
      4. Instantiated the RandomizedSearchcV() object and performed RandomizedSearchCV and training the model.  
         A screenshot of a computer

         Description automatically generated
      5. Printed best parameters used and the best score for the top-performing model  
         A screenshot of a computer code

         Description automatically generated
      6. Computed the accuracy, MSE, RMSE and R-Squared scores of the training set  
         A screenshot of a computer program

         Description automatically generated
      7. Computed the accuracy, MSE, RMSE and R-Squared scores of the test set  
         A screenshot of a computer program

         Description automatically generated
      8. Plotted the graph of features importances sorted by highest values to lowest  
         A graph with a green line

         Description automatically generated
   3. Submitted “Saemi Ramirez D209 PA2 Predict Analysis - Submit.ipynb”
2. Summarize the data analysis
   1. The accuracy score of the training set is approximately 86% and the test set is approximately 85.5% which represent the RandomForestRegression is a highly accurate model. Both MSE and RMSE in training (0.68% and 2.6%) and test set (0.73% and 2.7%) are all close to 0 which means high accuracy of prediction of the model.
   2. To answer the research question, the RandomForestRegressor is the best model to predict the patients TotalCharge with following features from the most important: Initial\_days, Initial\_admin\_Emergency\_Admission, Income, Complication\_risk\_Medium, and VitD\_levels.   
      It is great that the RandomForestRegression model is highly accurate, however, this might indicate overfitting, where the model has learned the noise and specific patterns in the training data rather than generalizable trends. This can lead to poor performance on new or unseen data. The random forest model itself can be complex and difficult to interpret. This might make it challenge to understand the underlying relationships between features and the target variable.
   3. After graphing the features importance, Initial\_days was the one outstood out of all the other features tested. It could’ve been better if the data had different variables that would be more important to be considered for the patients with the high blood pressure or not.
   4. The hospital organizations can conduct the comprehensive data analysis to understand the relationships and patterns between the total charges and the identified factors. This can help in identifying the key drivers of costs and potential areas for intervention. The community health programs can focus on preventive care to reduce hospital admissions and emergency visits. Programs could include health education, vaccination drives, and regular health screenings. Given the relation to vitamin D levels, promote vitamin D supplementation and education about its importance in the community. This could potentially reduce complications and hospital admissions related to vitamin D deficiency.
3. Panopto Link: <https://wgu.hosted.panopto.com/Panopto/Pages/Viewer.aspx?id=8231e97b-c68a-48ad-8f55-b1ab00abbc3c>
4. Resources for third-party code

Naik, Kunaal. *Random Forest Hyperparameter Tuning using RandomisedSearchCv | Machine Learning Tutorial*. YouTube. (October 27, 2020). <https://www.youtube.com/watch?v=SctFnD_puQI>.

1. Resources for in-text citation

Beheshti, Nima. *Random Forest Regression*. Medium. (March 2, 2022). <https://towardsdatascience.com/random-forest-regression-5f605132d19d>.

Charles, Chibuike. *Random Forest; Assumptions, Advantages, Disadvantages and Applications*. Medium. (June 2, 2023). <https://medium.com/@chibuike.odo.c/random-forest-assumptions-advantages-disadvantages-and-applications-2881f4ea14b6>.

*Random Forest Regression in Python*. GeeksforGeeks. (December 6, 2023). <https://www.geeksforgeeks.org/random-forest-regression-in-python>.