Predictive Data Mining Models in Python

**Written by Sonia Madin**

*Performance Assessment II*

*D209: Data Mining*

*Professor Festus Elleh*

*Western Governors University*

November 3rd, 2024

**Table of Contents**

A1. Proposal of Question ----------------------------------------------------------------------------------- 3

A2. Defined Goal -------------------------------------------------------------------------------------------- 3

B1. Explanation of Prediction ------------------------------------------------------------------------------ 3

B2. Summary of Method Assumption -------------------------------------------------------------------- 4

B3.Packages or Libraries List ---------------------------------------------------------------------------- 4

C1. Data Preprocessing ------------------------------------------------------------------------------------- 6

C2. Data Set Variables -------------------------------------------------------------------------------------- 7

C3. Steps for Analysis --------------------------------------------------------------------------------------- 8

C4. Cleaned Data Set -------------------------------------------------------------------------------------- 13

D1. Splitting the Data -------------------------------------------------------------------------------------- 13

D2. Output and Intermediate Calculations -------------------------------------------------------------- 14

D3. Code Execution ---------------------------------------------------------------------------------------- 15

E1. Accuracy and MSE ------------------------------------------------------------------------------------ 16

E2. Results and Implications ------------------------------------------------------------------------------ 17

E3. Limitation ----------------------------------------------------------------------------------------------- 17

E4. Course of Action --------------------------------------------------------------------------------------- 18

F. Panopto Recording -------------------------------------------------------------------------------------- 18

G. Sources for Third-Party Code ------------------------------------------------------------------------- 18

H. Sources --------------------------------------------------------------------------------------------------- 18

I. Professional Communication --------------------------------------------------------------------------- 18

References --------------------------------------------------------------------------------------------------- 19

***A1. Proposal of Question***

*Is it possible to predict the patient’s vitamin D levels and identify contributors using the random forest method based on the data provided?*

The research question will examine the key contributors that are affecting the patient’s vitamin D levels. Identifying factors contributing to a low vitamin D level can help hospitals keep a closer eye on patients and administer the proper vitamin D before it causes health problems. According to Yale Medicine, low vitamin D levels can lead to an increased risk of bone pain, muscle pain, and falls (Yale Medicine, 2024). The results of this analysis could provide incoming patients with better treatment timeliness and reliability. The hospital can benefit from the information by receiving greater positive survey results and lower readmission rates.

***A2. Defined Goal***

The primary goal is to develop a random forest model that can help the company predict the patient’s vitamin D levels. The continuous variable is identified as ***VitD\_levels*** in the data set which is the patient’s vitamin D levels in ng/mL (Medical Data Considerations and Dictionary. (n.d.). This will require us to find the key features causing low vitamin D levels. With this information, we can provide a course of action based on the analysis performed.

***B1. Explanation of Prediction***

Our primary goal for this analysis is to develop a model that can predict the patient’s vitamin D levels and identify the patients with low vitamin D levels to provide better care and increase survey results for the organization. Random Forest Regression will be used to analyze the continuous variable ***VitD\_levels*** because it is good at handling complex relationships and reduces overfitting (Dutta, 2019). This analysis will use Random Forest because one of the advantages is that it is less sensitive to the training data and more accurate than the decision tree algorithm (Dutta, 2019). Random Forest is an ensemble technique that can perform regression with the help of multiple decision trees. Ensemble learning combines the results of multiple models for a more accurate model (Dutta, 2019). I chose to perform my analysis with Random Forest machine learning model because of its accuracy compared to other models with the idea that we can provide quality results to the organization. A prediction provided for vitamin D levels from the Random Forest regressor is the mean of the prediction of multiple trees in the forest.

***B2. Summary of Method Assumption***

An assumption of Random Forest is the decision trees in the forest are independent from each other (Sruthi, 2021). This is almost nearly impossible because the trees trained used the same data set. One can ensure that the trees are independent of each other through bootstrap sampling. The basic idea of bootstrap sampling is measuring a small random sample and multiplying it until it is equal to the data being observed. This way the parameter through bootstrapping is comparable to the actual parameter (*Bootstrap Sampling | Bootstrap Sampling in Machine Learning*, 2020). Bootstrap sampling in machine learning is also called bagging.

***B3. Packages or Libraries List***

Python is the programming language because of its ease of use and wide use in the industry. The following packages and libraries will be used in the Random Forest model:

|  |  |
| --- | --- |
| Packages & Libraries | Usage |
| pandas | This library will be used to import the initial CSV file, give summaries about the data in multiple ways, aid in the cleaning process by detecting duplicates and null values, and in the re-expression of nominal categorical variables. |
| numpy | Numerical Python, or Numpy, is a library that will be used to find array objects for calculations |
| seaborn | Visualize the boxplots, count plots that detect the outliers and the correlation matrix |
| matplotlib.pyplot | Matplotlib is a library that will be used for visualizations like a scatterplot |
| from sklearn.preprocessing import OneHotEncoder | Transform categorical variables into numerical variables |
| from sklearn.feature\_selection import SelectKBest, f\_classif | For feature selection |
| from sklearn.model\_selection import train\_test\_split | To split data into train and test sets |
| from sklearn.ensemble import RandomForestRegressor | Fits a number of decision trees |
| from sklearn.metrics import mean\_squared\_error as MSE | Find the mean squared error metric for the model |
| from sklearn.metrics import r2\_score | Get R2 the coefficient of determination |
| from sklearn.metrics import mean\_absolute\_error | Finding the mean absolute error score |

***C1. Data Preprocessing***

There are many things one must do before we begin to run the model and one of those items is one-hot encoding. This transformation step turns categorical variables into dummy variables so the data created in the model can be meaningful. One-hot encoding allows for categorical data to be transformed into numerical data. Furthermore, a variable such as ***Gender*** cannot be transformed using ordinal encoding because the parameters suggest an order and we cannot say male is greater than female as an example. Thus we apply one-hot encoding where order is not a characteristic of the technique. Using ***get\_dummies()***  from **pandas** will allow us to consider the features in our machine-learning model (*Elleh*, n.d.).

***C2. Data Set Variables***

The variables chosen for this research question are based on health conditions and patient demographics that could help predict the patient’s vitamin D levels. The initial independent variables and their data class are identified below.

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Independent Variable | Data Type | Data Class |
| 1 | ***Children*** | Continuous | Numerical |
| 2 | ***Age*** | Continuous | Numerical |
| 3 | ***Gender*** | Categorical | Categorical |
| 4 | ***VitD\_levels*** | Continuous | Numerical |
| 5 | ***Doc\_visits*** | Continuous | Numerical |
| 6 | ***Full\_meals\_eaten*** | Continuous | Numerical |
| 7 | ***Soft\_drink*** | Categorical | Categorical |
| 8 | ***HighBlood*** | Categorical | Categorical |
| 9 | ***Stroke*** | Categorical | Categorical |
| 10 | ***Complication\_risk*** | Categorical | Categorical |
| 11 | ***Overweight*** | Categorical | Categorical |
| 12 | ***Arthritis*** | Categorical | Categorical |
| 13 | ***Diabetes*** | Categorical | Categorical |
| 14 | ***Hyperlipidemia*** | Categorical | Categorical |
| 15 | ***BackPain*** | Categorical | Categorical |
| 16 | ***Anxiety*** | Categorical | Categorical |
| 17 | ***Allergic\_rhinitis*** | Categorical | Categorical |
| 18 | ***Reflux\_esophagitis*** | Categorical | Categorical |
| 19 | ***Asthma*** | Categorical | Categorical |

***C3. Steps for Analysis***

1. The first step is to import all the libraries and packages in the Random Forest machine-learning model.
2. Following that step, we will load the data set in CSV form into our data frame
3. In the data cleaning process, we will profile the data to examine a summary of the data set we are working with. We can see a mix of quantitative and qualitative variables and 10,000 observations.
4. The second step in data cleaning is checking for duplicates and none were found.
5. Next, checking for null values shows that none are present.
6. The last step in data cleaning is to determine if outliers are present and treat them accordingly. After analyzing all outliers we will choose to retain outliers given that the Random Forest model handles outliers very well.
7. Next is to drop any variables that will not be needed during this analysis.
8. Transform categorical variables through ordinal encoding or one-hot encoding to numerical format to make the variables meaningful in the Random Forest model. Ordinal encoding will transform Yes/ No values to 1/0, respectively. One-hot encoding will transform the variables into dummy variables in order to ensure the datatype has changed to *int.* The random forest algorithm requires we drop a variable so we will use

***drop\_first = True*** in the line of code.

1. Check for correlation in the quantitative data using a heatmap to reduce correlated variables that will carry extra weight in distance (*Elleh*, n.d.). The heatmap shows low correlation because the values are not near 1 or -1.
2. Check for correlation in the data for qualitative data using a heatmap to reduce correlated variables that will carry extra weight in distance. The heatmap shows low correlation because the values are not near 1 or -1.
3. ***SelectKBest*** will be used for feature selection to ensure that only meaningful features are used for better model performance (AK, 2023). After standardizing the explanatory variables and finding the p-values, we determined to keep all features. ***Gender\_Nonbinary***, is the only variable that has a p-value less than 0.05. This means that this is the only variable that is statistically significant. Although this selection method increases the model’s efficiency, we will keep all variables for the purpose of testing more variables in the model.

A screenshot of a computer

Description automatically generatedA screenshot of a computer program

Description automatically generatedA screenshot of a computer program

Description automatically generatedA screenshot of a computer

Description automatically generatedA screenshot of a computer program

Description automatically generatedA screenshot of a computer

Description automatically generatedA screenshot of a computer code

Description automatically generated

***C4. Cleaned Data Set***

*See code attached:* ***clean\_medical\_D209\_T2.csv***

***D1. Splitting the Data***

*See files attached:*

***t2\_Xtrain.csv***

***t2\_Xtest.csv***

***t2\_ytrain.csv***

***t2\_ytest.csv***

***D2. Output and Intermediate Calculations***

The analysis technique utilized for predicting the continuous variable was the Random Forest Regressor because it combines the predictions of many decision trees. Predicting a continuous variable requires us to use regression because it can aggregate predictions through the average of the trees’ predictions. No intermediate calculations were performed throughout this analysis. Although hyperparameter tuning can improve the model we will choose to not take this step for the moment. We start by converting those categorical variables into dummy variables and make sure to drop *n-1* variable, as it is a requirement for this technique (Elleh, n.d.). The data set will be split into 70% training and 30% testing data sets. The hold-out method allows the model to be trained in the training data set and evaluated in the testing data set. Next, we will create a random forest regressor with 400 trees (Train an RF Regressor | Python, 2024). Next, is fitting the random forest regressor to the training set and then predicting the test set labels. To measure the accuracy of the model, we will find the mean squared error, the root mean squared error, and the r-squared score. As a final analysis, we will create a horizontal bar plot that shows the most predictive features (Train an RF Regressor | Python, 2024).

***D3. Code Execution***

*See code attached:* ***T2\_D209\_RF.ipynb***

***E1. Accuracy and MSE***

There is no single calculation for accuracy when looking at a random forest regression model so we will analyze three calculations to determine if the model is reliable in predicting the patient’s vitamin D levels. One thing to remember is that when working with regression, we are looking for a 0.0 error to have the best score. The mean squared error is the sum of the squared difference between the actual output and the predicted output value. A low mean squared error shows a model that can predict values more accurately (Beheshi, 2020). The RMSE is the square root of the average squared distance between the actual and the predicted score (Elleh, n.d.). The coefficient of determination, also known as R-squared, measures the relationship between the residual and the total sum of squares (Elleh, n.d.). R-squared provides another accuracy metric and tells us how well the model fits the data.

***E2. Results and Implications***

The MSE value was calculated as 4.226 and the RMSE is 2.06. The numbers for the MSE value can vary from zero to infinity. With the MSE metric, we can say that the model will predict the patient’s vitamin D level well. The value for the RMSE is close to 0, so the model can predict more accurately. Let’s determine other accuracy metrics to ensure the model we made is reliable. The closer the R-squared value is to 1, the better the model fits the data. The value calculated for R-squared is 0.0001744. This means the model fits the data only at a .017% and therefore unacceptable (Elleh, n.d.). We also created a horizontal bar plot that shows the features that most contribute to the patient’s vitamin D levels. Some implications of using the model created is that if used to predict the patient’s vitamin D levels, it could result in poor predictions based on the accuracy metrics we have discussed. The MSE, RMSE, and R-squared accuracy metrics do not show an agreement in the validity of the performance of the model. We also cannot guarantee that ***Age,*** as shown in the plot, is a key feature that is contributing to lower vitamin D levels for patients. The consequences of using the model created cannot guarantee lower readmission rates, greater positive survey results, or ensure the patient with treatment timeliness.

***E3. Limitation***

A limitation of random forest is that it cannot extrapolate (Thompson, 2019). This means that the model can only make predictions based on the average of the previous values. This can limit the predictions to the highest and lowest values in the training data set. As an example, this limitation could affect the data that has a trend over time and can over or under-predict based on the trained data set (Thompson, 2019). This would be a problem if we were looking for any trends in the data.

***E4. Course of Action***

Based on our model’s accuracy metrics, the model created cannot be used to predict a patient’s vitamin D levels without other suggestions to improve the model. A recommendation for the organization would be to begin collecting more existing health conditions to add to the data set since the data collected didn’t give us a reliable model. Exploring further existing health conditions can allow the model to have more data within the model. The analysis could then be further improved by hyperparameter tuning, which allows for selecting the best hyperparameters so the model can perform more accurately. The organization can then have confidence in the new model created to lower readmission rates, have positive survey results, and ensure treatment timeliness.

***F. Panopto Recording***

*See Panopto video attached.*

***G. Sources for Third-Party Code***

*See attached references below.*

***H. Sources***

*See attached references below.*

References

AK, A. (2023, October 15). *Clever Cuts: Uncovering the Power of SelectKBest for Feature*

*Selection in Machine Learning*. Medium. <https://medium.com/@abelkuriakose/clever-cuts-uncovering-the-power-of-selectkbest-for-feature-selection-in-machine-learning-c8d20d75c82f>

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

*Bootstrap Sampling | Bootstrap Sampling In Machine Learning*. (2020, February 12). Analytics

Vidhya. <https://www.analyticsvidhya.com/blog/2020/02/what-is-bootstrap-sampling-in-statistics-and-machine-learning/>

Dutta, A. (2019, June 14).*Random Forest Regression in Python - GeeksforGeeks.* GeeksforGeeks. <https://www.geeksforgeeks.org/random-forest-regression-in-python/>

Elleh, F. (n.d.). D209 Data Mining 1 Task 2 Cohort [PowerPoint Slides].

Medical Data Considerations and Dictionary. (n.d.). *Western Governors University.* <https://access.wgu.edu/ASP3/aap/content/g9rke9s0rlc9ejd92md0.html>

Thompson, B. (2019, December 20). *A limitation of Random Forest Regression*. Medium.

<https://towardsdatascience.com/a-limitation-of-random-forest-regression-db8ed7419e9f>

*Train an RF regressor | Python*. (2024). Datacamp.com; DataCamp.

<https://campus.datacamp.com/courses/machine-learning-with-tree-based-models-in-python/bagging-and-random-forests?ex=8>

Yale Medicine. (2023). *Vitamin D Deficiency*. Yale Medicine. <https://www.yalemedicine.org/conditions/vitamin-d-deficiency>