Stats 101C Final Project

Predictive Analysis of Obesity

By Chinmay Varshneya, Abhinav Madabhushi, Nathaniel Albano, Vienna Truong, &
Connor Coyne

Abstract

Obesity can lead to many health, social, and economic complications. Being able to predict obesity can be valuable to start early intervention and prevention and reduce healthcare costs. Our objective was to use statistical classification models in order to make accurate predictions on the obesity status of our patients given various measurements of their health and lifestyle. We were given a training data set with 32,014 observations. Through careful data analysis, data imputation, and utilizing various modeling techniques, we were ultimately able to create a KNN model with 7 features on our non-linear data to produce a 1.0 accuracy on a testing data set with 10,672 observations. Although there are some limitations (as discussed in later sections), we feel that our model is efficient and accurate in being able to make accurate obesity predictions. Further testing is necessary on new, unseen data in order to ensure the reliability of our model.

Introduction

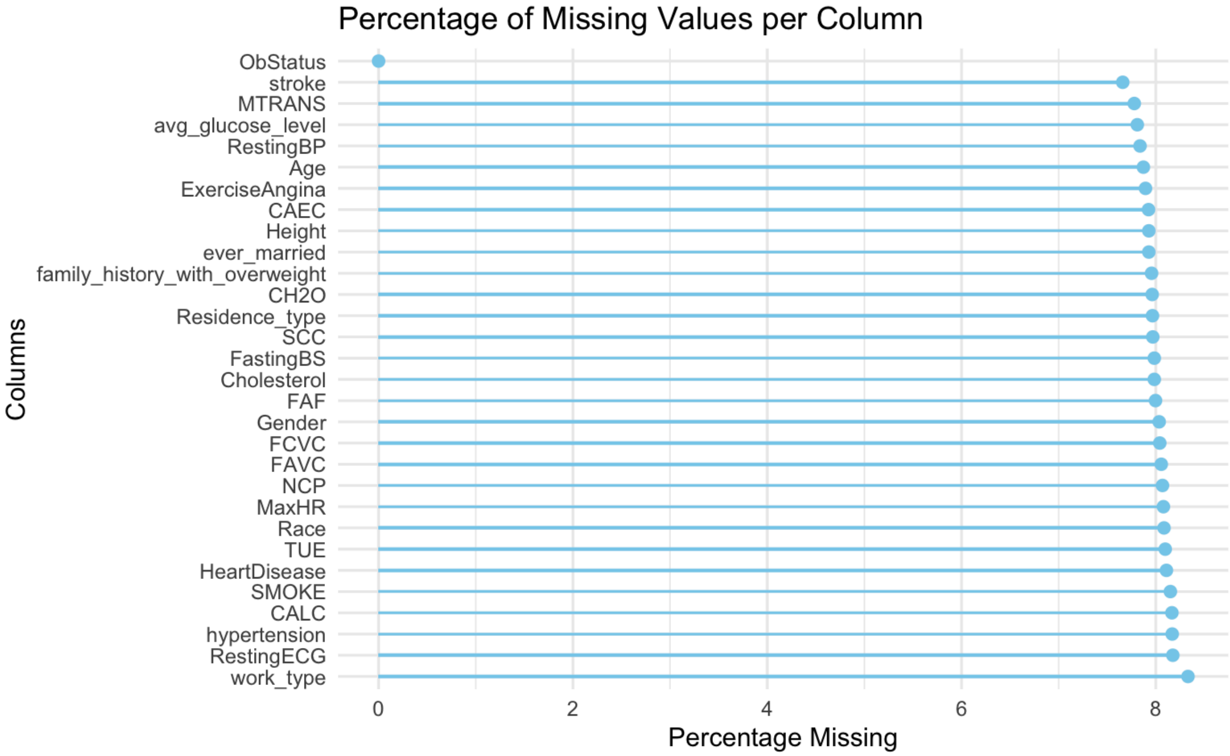
Obesity is one of the most prevalent health issues in the United States with roughly 40% of the population being obese. Obesity can lead to many negative side effects, including shorter life spans, increased cost (for the individual and for the state), negative social perception, and much more. Thus, how to identify the issues that are contributing to the growing obesity rate in the US is an urgent issue for both individuals with obesity and the state which is spending hundreds of billions of dollars a year on just healthcare costs. Identifying obesity is not our focus here as that is solely dependent on weight, but identifying the factors that are associated and might cause obesity. By being able to predict obesity, not only can we identify people who are at high risk, but we can also isolate some of the predictors that are heavily associated with obesity.

In this Kaggle Project, an Obesity training data set was used that included 32014 individuals with 29 predictor variables and 1 outcome variable. The testing data set that was used to measure the accuracy of our model included 10672 individuals with 29 predictors. There are 11 numerical predictors and 18 categorical predictors. The numerical predictors included basic descriptors such as age and height as well as health data including resting blood pressure, cholesterol, and average glucose level. The categorical predictors are similar, as they contain basic descriptors such as gender, race and work type, and also health data such as if they have heart disease, if they've ever had a stroke, and if they have a family history of being overweight. Our mission was to create a model to predict the target variable ObStatus (are they obese or not) by selecting the fewest basic descriptors and health data possible while still being accurate.

Data Analysis

Data Cleaning & Pre-Processing

In order to proceed with our analysis so that we may reduce dimensionality and begin applying various models, we cleaned and pre-processed our data. The first step in this process is to handle the missing values. As reflected by our exploratory data analysis there are a significant number of "N/A" values in many of the variables.



As a result, it did not make sense to simply remove rows with missing values. We chose a simple approach to imputation where categorical variables were filled with the mode of that column and numerical variables were filled with the median.

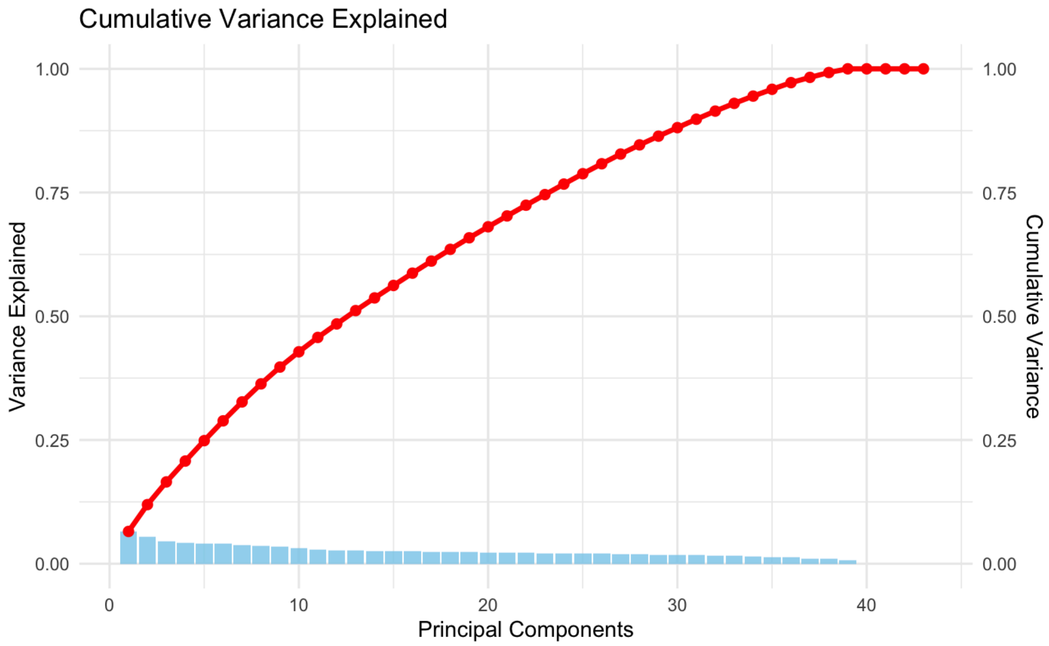
We also created a copy of this cleaned dataset where all the data was made numerical. This allows us to conduct principal component analysis to reduce the dimensionality of the data and thus simplify the models that we test. In order to pre-process the data in the aforementioned manner, we applied various forms of data encoding to convert the categorical variables. For the columns which contained two opposite category levels we used binary encoding. Examples of this are Yes/No, Obese/Not Obese, Rural/Urban, and Male/Female. One level became a 1 and the other became a 0.

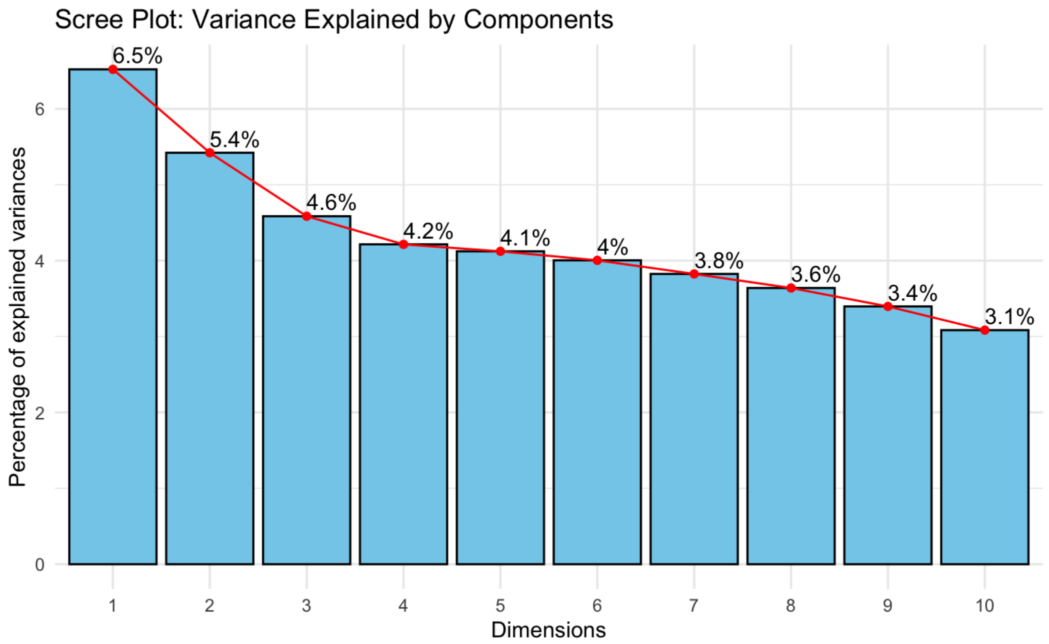
Alternatively, the columns that contained a hierarchy of levels were better suited to natural encoding. As the levels increased in value, they were mapped to larger integers. This occurred in the columns “CAEC” and “CALC” which contained the following: No/Sometimes/Frequently/Always. These were mapped to 1/2/3/4.

Lastly, the columns in which there is no relationship within the factor had one-hot encoding applied to them. The original column is replaced by n columns where n is the number of factor levels. Each new column is a binary column indicating whether the original category was present in that row or not. This form of encoding was applied to the variables "MTRANS", "Race", "RestingECG", and "work_type".

Variable Selection

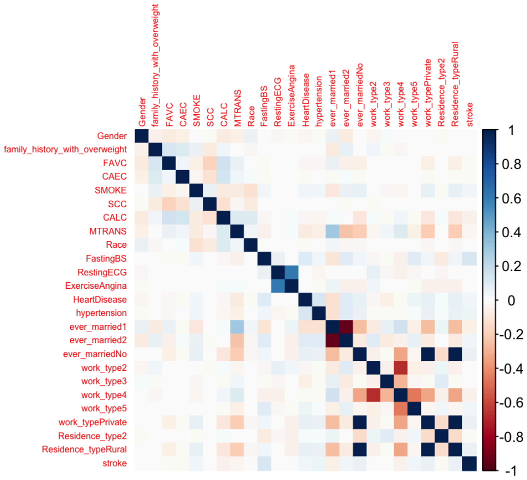
Having completed data cleaning and preprocessing, we conducted principal component analysis. Note that due to the necessary pre-processing, we increased the dimensionality of the data from 29 variables to 42 variables. Below are the results:

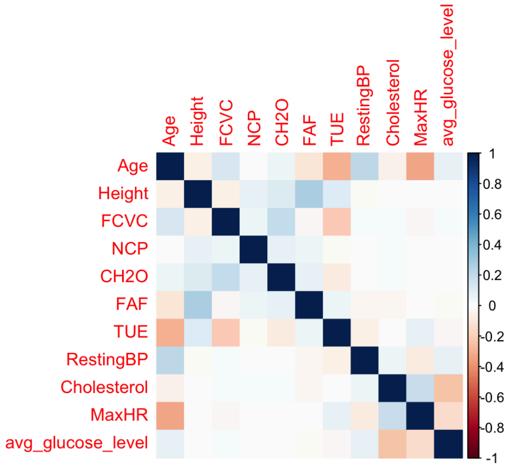




Evidently, the analysis was not successful in reducing the dimensions of the data. The first principle component only captures 6.5% of the variance in the predictors. Overall, in order to capture 95% of the variance in the data, 35 principle components are required. More dimensions than are originally present are needed in order to capture a large amount of variance in the data through PCA, leading us to move forward without using PCA on our data.

We also looked to test for multicollinearity between the predictors to see if we could remove any highly correlated variables. This also required using pre-processed data where all the categorical variables were encoded numerically. We spit out the originally categorical and originally numerical variables for our analysis.

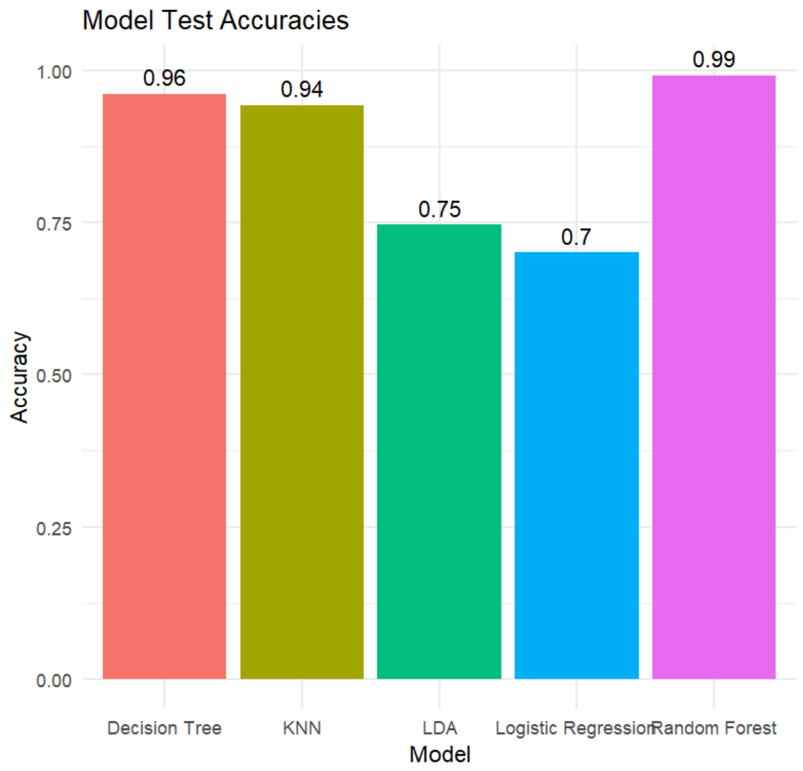




Evidently, there are no highly correlated variables on either side. The categorical variables (on the left) demonstrate some correlation between two factors of marriage, work, and residence type but that does not lend itself to further analysis since they come from the same predictors. The numerical variables (on the right) clearly demonstrate no large amounts of correlation between predictors.

Methods and Models

The datasets given to us included the training datasets with the labels for Obesity Status and a testing dataset without the labels for Obesity Status for which we had to submit a Kaggle prediction. To test our model, we initially divided the training (old) dataset into two parts: 80% training data (new) and 20% validation data. We initially trained 5 different models on the training data (new) and tested it on the validation data to get the accuracy. The following graph outlines the accuracies obtained from the validation data.

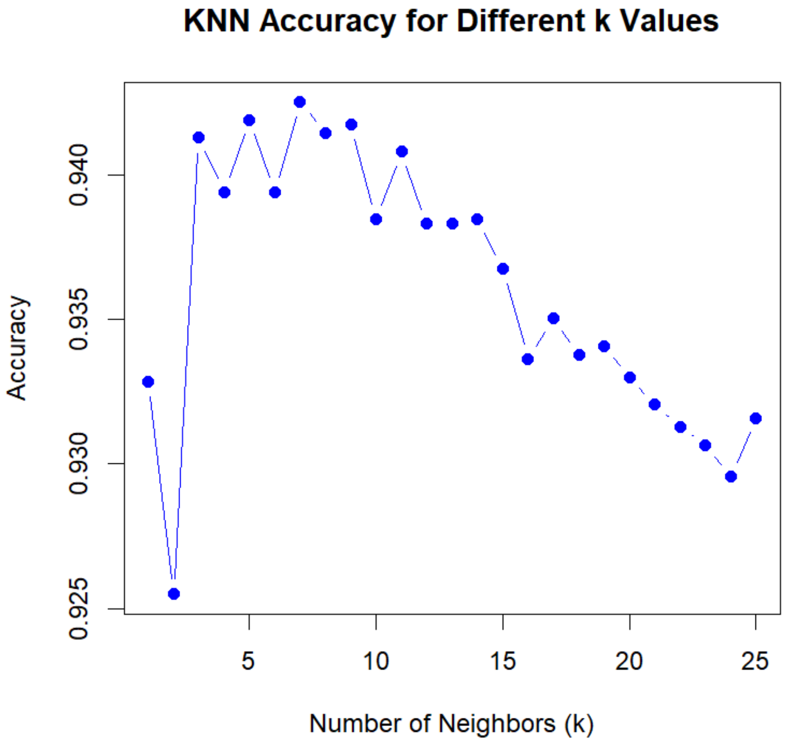


According to the graph above, the random forest model achieved the highest accuracy, followed by Decision Trees, K-nearest neighbors, and finally Linear Discriminant Analysis model and Logistic Regression. This highly suggests that our data has a non-linear boundary. Also, note that this model was without tuning any hyperparameters of the models and using all the predictors. The next few sections outline further analysis of the 3 best models, including hyperparameter tuning, feature importances, and feature reduction techniques.

A. K-nearest neighbors (KNN) model

The first model that was implemented was the KNN model. The KNN model takes into account the Obesity Status of the K nearest neighbors beside the data point which we want to predict the status for and uses the highest frequency class of the K nearest neighbors to make its prediction. The KNN model in base R uses the Euclidean distance to measure the distance between data points. The first piece of analysis here would be to find the optimal value of K that maximizes the accuracy of the model. We would need to find a middle ground between model complexity (overfitting) and simplicity (underfitting). This problem is often called the bias-variance tradeoff.

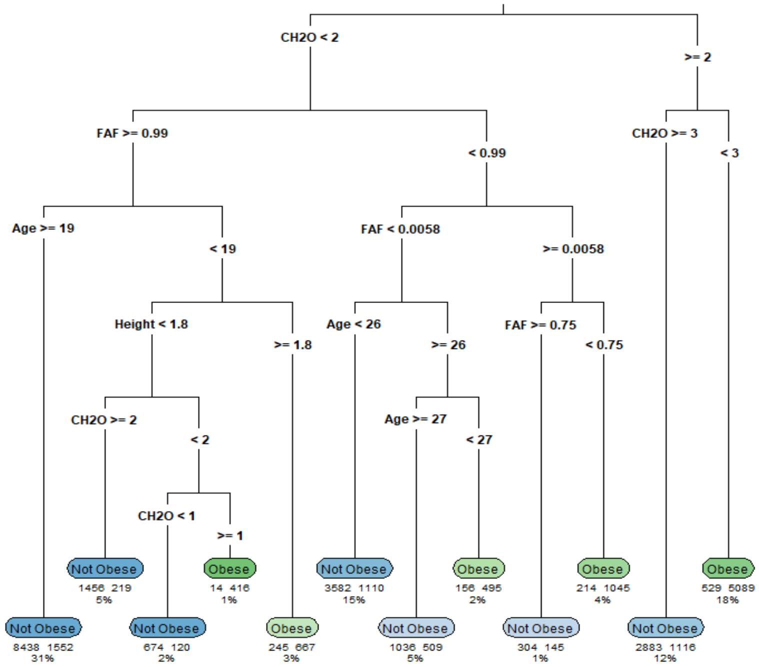
To find the best value of K, we used the "elbow method". The KNN model using different values of K was fitted on the training data (new) and then tested on the validation data. The accuracies obtained on the validation data using the different values of K were then graphed to find the optimal K.



According to the graph above, the best value of K seems to be K = 7 which has the highest accuracy on the validation data equivalent to 94.5%. The KNN model with K = 7 was then fitted on the training data (old) and used to predict the Obesity Status for the testing data and a prediction was made to Kaggle. The score obtained was 94.527%.

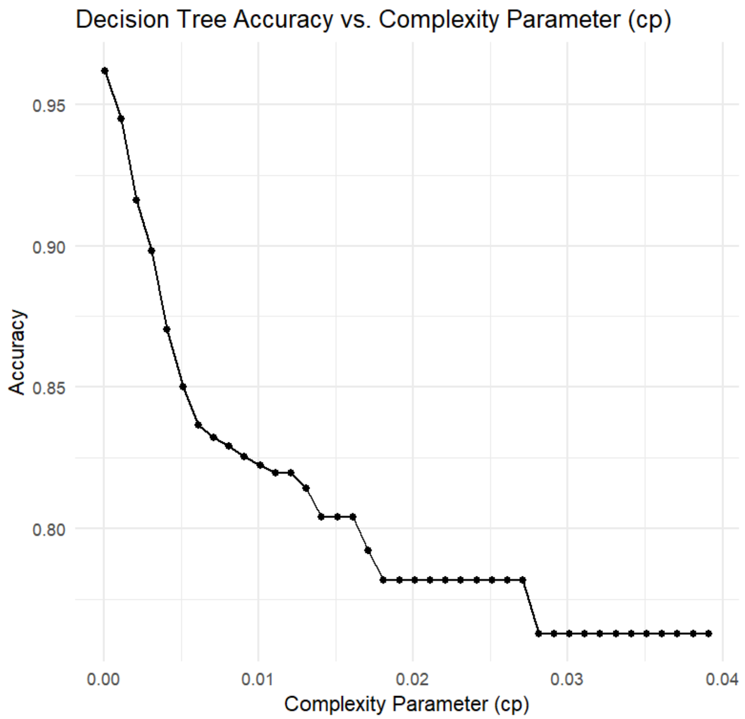
B. Decision Trees

The second model that was implemented was the most basic tree model. The tree model splits the data into subsets based on a rule, attempting to create the most homogenous (same prediction class) subsets. Initially, the decision tree model achieved a validation set accuracy of 0.96. The splits that the decision tree made were visualized to analyze the importance of the features.



Several factors such as the CH20 level, FAF, age, and height seem to be the most important variables in predicting Obesity Status. Later, when using the Random Forest model to predict the Obesity Status, we will do a more detailed analysis of feature importances.

To improve the accuracy of the model, we tried to tune the hyperparameters of the decision tree model. In particular, we tried to vary the value of cp (complexity) which also finds the middle ground between model complexity (overfitting) and simplicity (underfitting). A similar elbow method was used like above and the results were plotted.

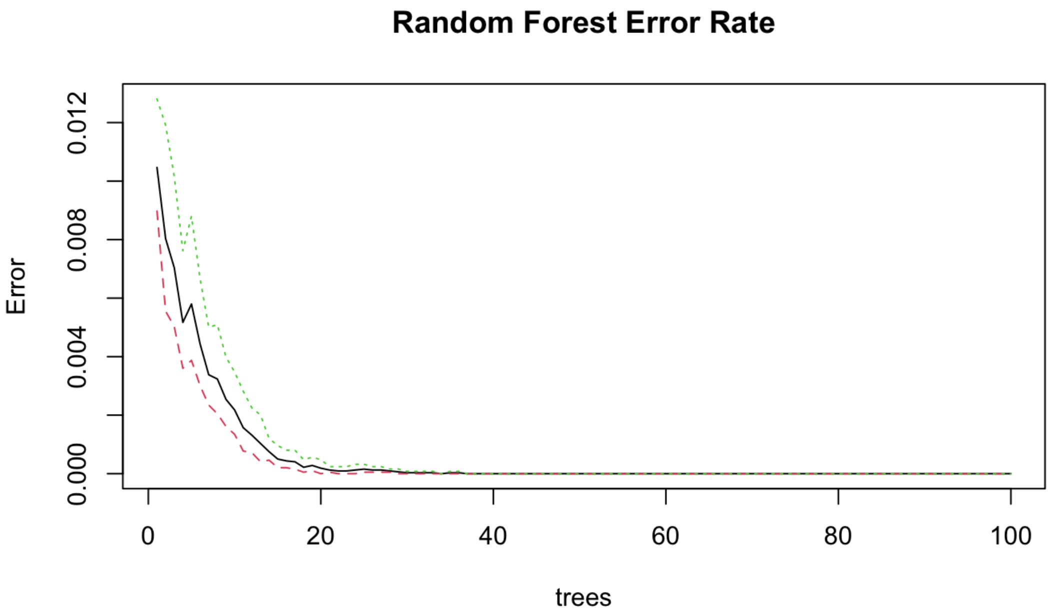


It turns out that the model with the least cp value (highest complexity) had the highest validation data accuracy. This means that the larger the number of splits in the tree, the better the model performs. This is usually not the case, as the higher the complexity, the more the model will overfit and get bad results on the testing data. This tree model with a low cp value was then fitted on the training (old) data and was used to make predictions on the testing data. The predictions were submitted to Kaggle to get a testing accuracy of 96.673%.

C. Random Forest

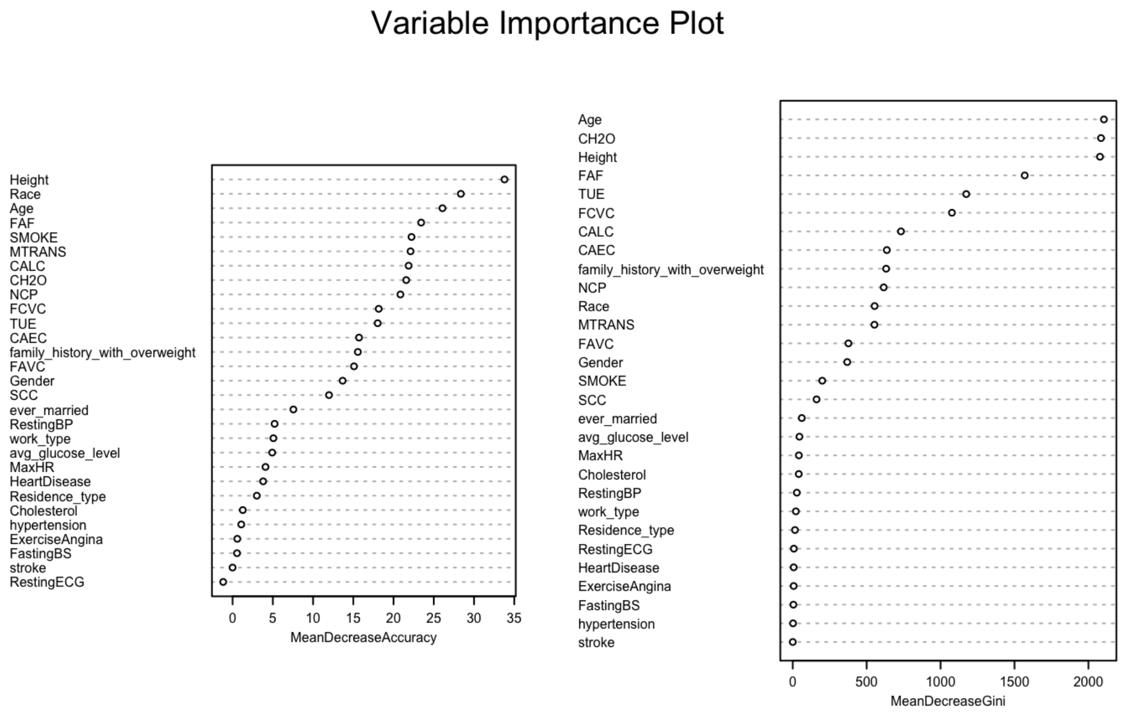
The third model implemented was the Random Forest model. This predictor builds multiple decision trees to generate more accurate predictions. Generally, this outperforms a single decision tree, so to improve our testing accuracy Random

Forest was used.



The first model included all predictors with 100 trees, although the above graph indicates that we could have lowered the number of trees to around 50 for similar results. Using three-fold cross-validation, mtry (a number of variables randomly sampled to build a tree) of 14 was selected. This model was fitted on the training data and then used to predict testing obesity status with an estimated misclassification rate of 0.99. This submission on Kaggle performed better with an accuracy of 100%.

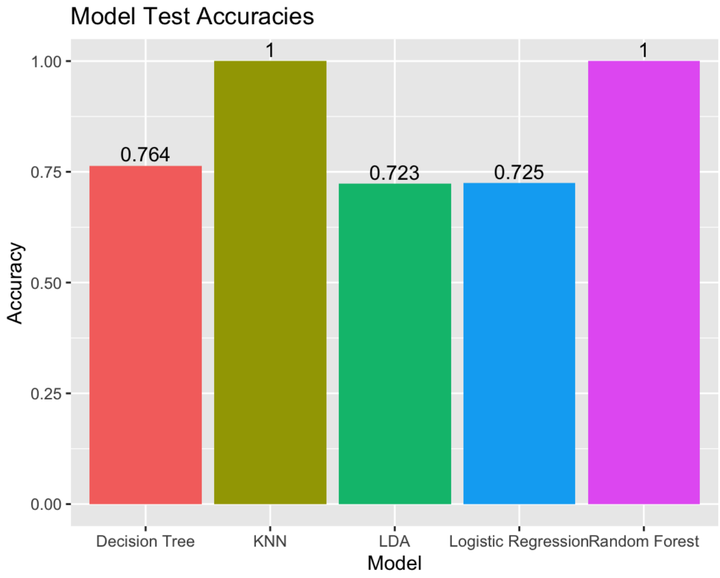
While this model performed the best of ours and earned a 100% testing accuracy, it is highly complex. To reduce the complexity of the model, we used the Variable Importance Plot to find the best subset.



We sorted by MeanDecreaseGini and chose the best 7 predictors (Age, CH2O, Height, FAF, TUE, FCVC, and CALC) to include in our model. We ran another Random Forest using this model, which produced a validation set accuracy of 1. Similarly, this submission on Kaggle earned 100% testing accuracy.

D. Other Reduced Models

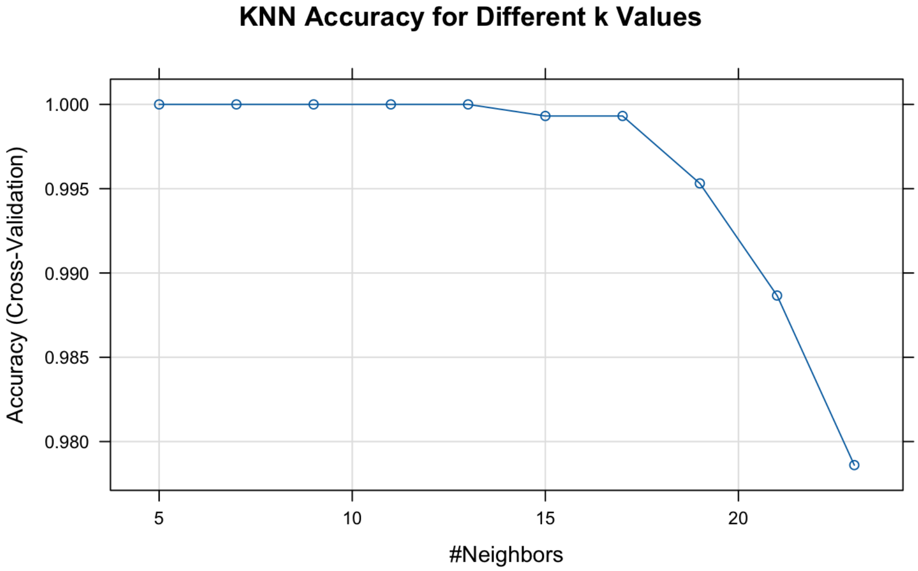
Using the reduced subset from the variable importance plot, we generated several other models.



Aside from the Random Forest, a tree, KNN, LDA, and logistic regression model were all tried with the 7 best predictors from variable importance. The tree, LDA, and logistic regression models' testing errors all decreased. However, the KNN model increased to a test accuracy of 1.

E. Final Model

Given the improvement due to the decrease in subset size and simplicity relative to the Random Forest model, the new KNN model with reduced features was chosen as our final model.



Based on the graph above, the best k-value for this model is k=5. This k is the smallest value that produces an accuracy of 1. This new model was fitted on the training data to predict obesity status in the testing data. Both our validation set and Kaggle test accuracy were 100%.

Discussion and Limitations

Although Random Forest also achieved a test accuracy of 1.0 with the same predictors, we ultimately decided to use the KNN version as our main model. This was due to its computational efficiency and simplicity in the design of its algorithm in comparison to Random Forest. This is not to say that our model is the most efficient, however. To begin with, KNN, although generally better than Random Forest, can still be computationally expensive. Furthermore, because we selected our features based on the variable importance, MeanDecreaseGini plot from our Random Forest model, they may not be the best combination of features for our KNN model, even if they are for our Random Forest. Our research into whether or not we are violating any assumptions when using features selected from our random forest variable importance plot in our KNN model was inconclusive. However, our team concluded that due to the nonlinear relationship between our data and the algorithms of both KNN and decision trees, we are not violating any major assumptions that would lead to our final KNN model breaking. The biggest problem is the fact that our model is now based on our testing data, given that we changed our model due to the results of our variable importance plot. Further testing on new data is necessary. Finally, although we did do light preliminary research, we still lack the domain knowledge of a physiological and health profession that could grant us further insight into the importance and interactions of our predictors.

Conclusion

Nevertheless, with our focus being solely on prediction accuracy, the work we did was able to achieve a 1.0 accuracy on the test data using two different types of models. Ultimately, the model we chose as our best model was the KNN model with the aforementioned 7 Random Forest predictors. Despite its limitations, we feel that given the size of the testing data set, we were able to create an accurate predictive model that avoided overfitting. However, we would love to use further testing data in the future to compare our model against, along with being more thorough in our research of the domain in order to create a more efficient model in the future.

References

Itagi, et. al.. Effect of obesity on cardiovascular responses to submaximal treadmill exercise in adult males. J Family Med Prim Care. 2020 Sep 30;9(9):4673-4679.

Robinson, et. al. Screen Media Exposure and Obesity in Children and Adolescents. Pediatrics. 2017 Nov;140(Suppl 2):S97-S101.

Longo-Silva, et. al. Association of largest meal timing and eating frequency with body mass index and obesity, Clinical Nutrition ESPEN, Volume 60, 2024, Pages 179-186, ISSN 2405-4577,