University of Virginia

Project 2 - Predicting the Likelihood of a Stroke with Predictive Models

Divya Kuruvilla and Glory Gurrola

DS 3001

Professor Johnson

30 November 2023

Summary

The purpose of this project was to build algorithmic models that predict the likelihood of a person having a stroke based on specific factors and variables, such as age of the individual or if the individual smokes or not. If a stroke can be predicted based on specific variables and factors, then preventive measures can be taken, such as lifestyle changes or even medication changes and/or adjustments. Individuals and healthcare professionals may also know what factors to look out for when assessing the likelihood of a stroke. Detection of stroke is vital for the health and wellbeing for each individual; early detection can lead to interventions that reduce the severity of a stroke, if it does happen to occur. This is important as the impact of a stroke can range from mild to severe impairment. The use of predictive models for stroke risk assessment can focus on prevention and early intervention, allowing individuals and the healthcare system to benefit when an effective predictive model is created. Thus, a good predictive model will accurately predict future outcomes of the target variable, in this case if the person will have a stroke or not, based on the specific variables of the testing data set with minimal errors.

The method used to determine the best predictive model started with first preparing and cleaning the variables in the data set. The number of missing values in each column were found, and the columns not relevant to data analysis were removed. Then, graphs and visualizations were created to analyze each of the variables and their possible relationships with the other variables. Finally, the different models were built. The models we decided to build were a numeric linear model, categorical linear model, a combined (numeric and categorical) linear model, K nearest neighbor (KNN) model, and a classification decision tree model. We decided to build a separate numeric linear model and categorical linear model as there were variables in the dataset that were strictly numeric or categorical. Thus, we thought it would be interesting to analyze if the numeric variables influence the target variable more than the categorical variables (or vice versa). We also decided to build two non-linear models, a KNN model and classification decision tree, to see if two different types of supervised machine learning algorithms are better at predicting outcomes for future cases. To determine how effective each model was, we computed specific metrics, such as R^2 and RMSE values, for each model. With these metrics, we then compared each model to find the one with the lowest RMSE value.

From this project, we concluded that the best model was the combined linear model. This model had the lowest RMSE value from the other models we tested. With a low RMSE value, we concluded that the model fits the data well and has more precise predictions. The worst model was the classification decision tree model; this model had the highest RMSE value and a R^2 value that was negative.

Data

There were 12 variables found in the dataset. Of these 12 variables, the target variable was the variable ‘stroke.’ The variables used to predict the target variable were ‘age’, ‘avg_glucose_level’, ‘bmi’, ‘ever_married’, ‘gender’, ‘heart_disease’, ‘hypertension’, ‘id’, ‘residence_type’, ‘smoking_status’, and ‘work_type’. All 11 variables were used to analyze the

risk of a person having a stroke. We found all these variables to be important when predicting the likelihood of a person having a stroke. For instance, the health of an individual (i.e., their average glucose level, body mass index, if they have heart disease or not, if they have high blood pressure or not) is just as important as outside factors that may affect the individual (such as being married, the type of work they do, if they smoke or not, or even where they live). These outside factors may be stress-inducing or affect the individual in a certain way that may detrimentally impact their health and wellbeing. Therefore, we found it important to factor in all these variables when trying to predict the likelihood of a person having a stroke. Additionally, there was a training data set that was used to build the predictive models, and there was a testing data set that was used to test the models. However, before the training and testing data could be split into the appropriate X and y data sets, the data had to be cleaned.

One of the variables, 'id', was the study identification number. The identification number could be private information, and it is not relevant to assessing the likelihood of a person having a stroke. Therefore, this variable was dropped from the training and testing data frame as it wouldn't be useful when creating models. In addition, when the columns of the data were printed out, there was found to be a "Unnamed: 0" column that was also dropped for its irrelevance. Next, the 'Residence_type' variable was renamed to be all lowercase to fit in with the rest of the variables; this was more of a stylistic choice to keep all of our variables uniform. After these initial steps of cleaning, the missing values were handled. A challenge arised when we found that there were 150 missing values for the 'bmi' variable in the training data set, and 42 missing values for the 'bmi' variable in the testing data set. To resolve this issue, the NaN values were imputed with the mean of the 'bmi' variable. We decided to not drop or discard the missing values because that could lead to a loss of variable data that might be useful in the future when building our models. Imputing the 'bmi' mean for the missing values helped maintain the size of the dataset as well.

An additional problem with the data that we had to fix was the outliers. After creating the box and whisker plots of BMI and Average Glucose Level, it was evident that there was an abundance of outliers for both of those variables. For the linear predictions, we knew that the outliers would have a bad impact on their accuracy of the prediction, so we wanted a way to process the outliers without losing the data. In order to accomplish this we utilized winsorization which transformed the outliers to be replaced with the upper or lower range within the quartiles so that they would not have extreme values and skew the predictive models. We chose to do this method as opposed to replacing the values with the mean because we did not want to overfit the data and get rid of any patterns that could be essential to the predictive models. Overall, since the data was provided in a mostly clean fashion, we only had to make a few adjustments when cleaning the data set. After all these changes, there were no missing values to be found, the variables were uniform, and the unnecessary columns were dropped. Now, the testing and training data frames can be used to make graphs and build predictive models.

To prepare the data to be used for creating the linear models, we identified which columns were numeric and which columns were categorical. By separating these variables, we

were able to create a numeric linear model, a categorical linear model, and a combined (numeric and categorical) linear model. However, our next challenge occurred when we tried to build the K nearest neighbor (KNN) model. To apply the max min normalization function, we had to make sure all the columns were numeric. Since there were four non-numeric columns, ‘ever_married’, ‘gender’, ‘residence_type’, ‘smoking_status’, ‘work_type’, we had to map the string values to its integer representation. For example, the ‘ever_married’ variable had two possible values: ‘Yes’ or ‘No.’ We mapped the ‘Yes’ to 1 and the ‘No’ to 0, and used this same process for the other non-numeric variables. After this was done, we were able to use the new copy of the training and testing data sets to create the KNN model and the classification decision tree model.

Results

After preparing all of the data for the models, we decided to create some graphs in order to have a better visualization of the data and understanding of the relationships between the different variables. By preparing these graphs, it allowed us to make more informed decisions about what models might be the best type of models for accurately predicting if someone is going to have a stroke or not. The first graphs that we created were box and whisker plots of the numerical variables: BMI, Age, and Glucose level. The graphs appears as follows:

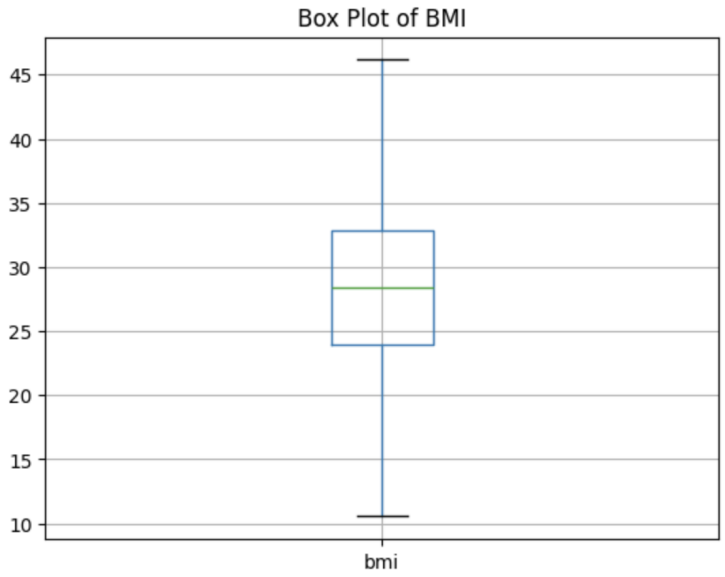

Figure 1: BMI Box Plot

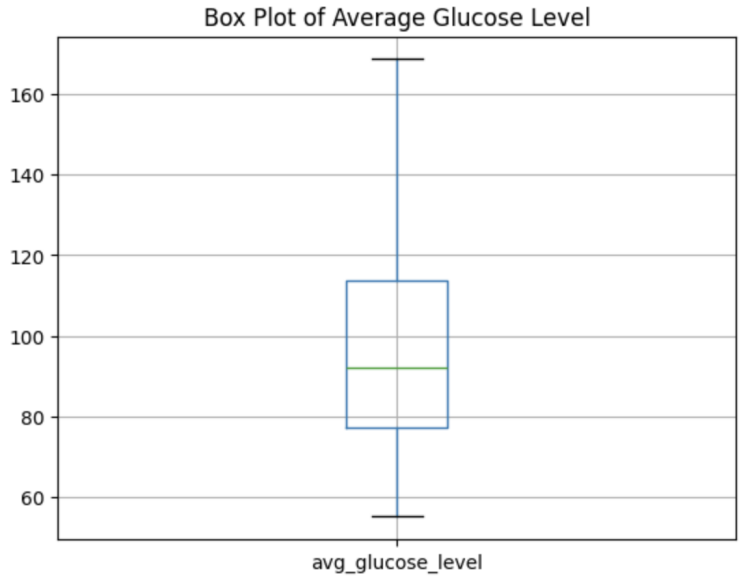

Figure 2: Average Glucose Level Box Plot

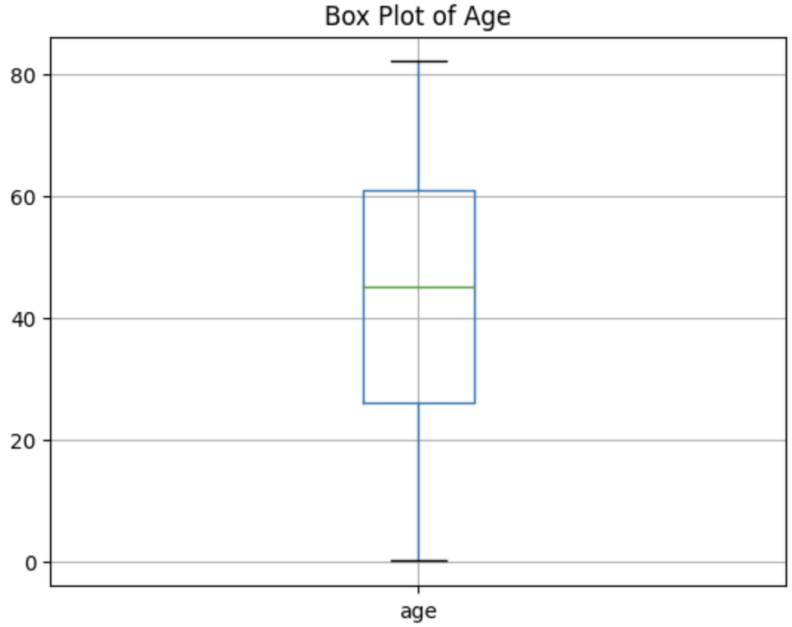

Figure 3: Age Box Plot

By creating the box and whisker plots, we were able to visualize the effect that the outliers were having on the variables in the data and allowed us to realize that we needed to account for the outliers in our data cleaning. This led us to winsorize BMI and Average Glucose Levels so that those variables would not negatively impact the predictive models. After winsorizing the variables, the box and whisker plots revealed information to us about the nature of the numerical data relating to the spread and range of each variable.

The next step that we took, in order to gain a better understanding of the data we were using to make our models, was performing a cross tabulation of different variables to see how certain variables correlate with the trends existing within other variables. In order to do this, the first graph that we made was a box and whisker plot with respect to stroke. This plot displays two side by side box plots, each relating to the 'stroke' variable, which represents whether someone has experienced a stroke or not (stroke = 0 means no stroke and stroke =1 means has had a stroke). From examining both of these box plots side by side, it is evident that the spread of ages for those that have had a stroke is much higher than those that have not had a stroke. The average sits over 60 years old for those who had a stroke, while the average age of those that have not had a stroke is around 40 years old. Seeing these side by side plots provides context as to what the predictive models and algorithms will be affected by, and what variables could have a large impact on the algorithm for predicting whether someone will have a stroke or not.

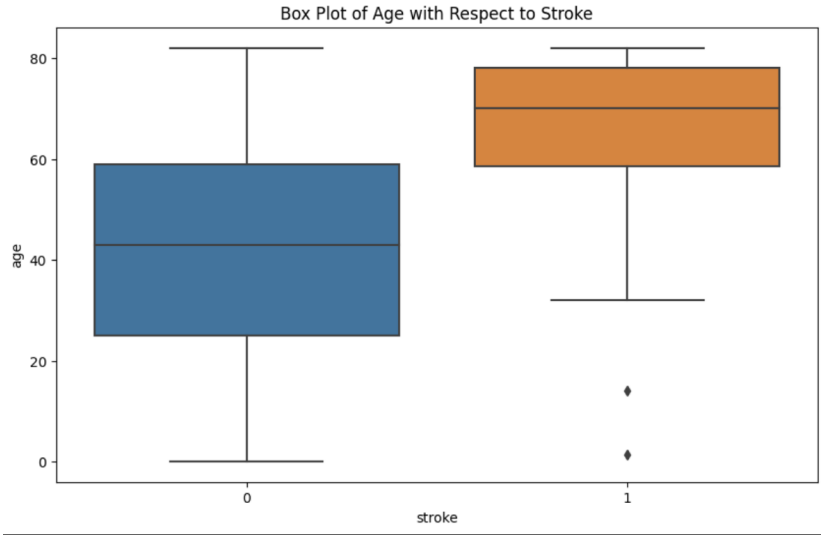

Figure 4: Box Plot of Age with Respect to Stroke.

The final step that we took in order to visualize and display the data was to create a pair plot relating all of the numerical data (age, average glucose level, bmi) to the stroke variable. Pair plots can be a very useful tool for exploring data because it provides a good summary of the relationships between multiple variables and allows visualization of the patterns and trends that were present in the data frame. Some of the key highlights from this pair plot was seeing how age affects the likelihood of having a stroke. In all of the graphs where age is a factor, the points are clustered where age is higher. Based on the average glucose level and bmi variables, it was not entirely evident whether one had a high correlation with stroke because there were lots of data points spread across the plot. It could be the case that there is a higher density in certain regions of the plot, however it is not visibly clear due to the presence of other data points. To better visualize if there was a correlation between the stroke status and average glucose level, we then created a separate kernel density plot. This kernel density plot was very important for our understanding of the data frame because it clearly visualized that there was a much higher

density of people having strokes with an average glucose level of 50-125, and that the no stroke data was evenly distributed over two areas in the plot. This plot gave us a better idea of how average glucose level would come into play when creating our predictive models, because it allowed us to visualize trends present before implementing the algorithms.

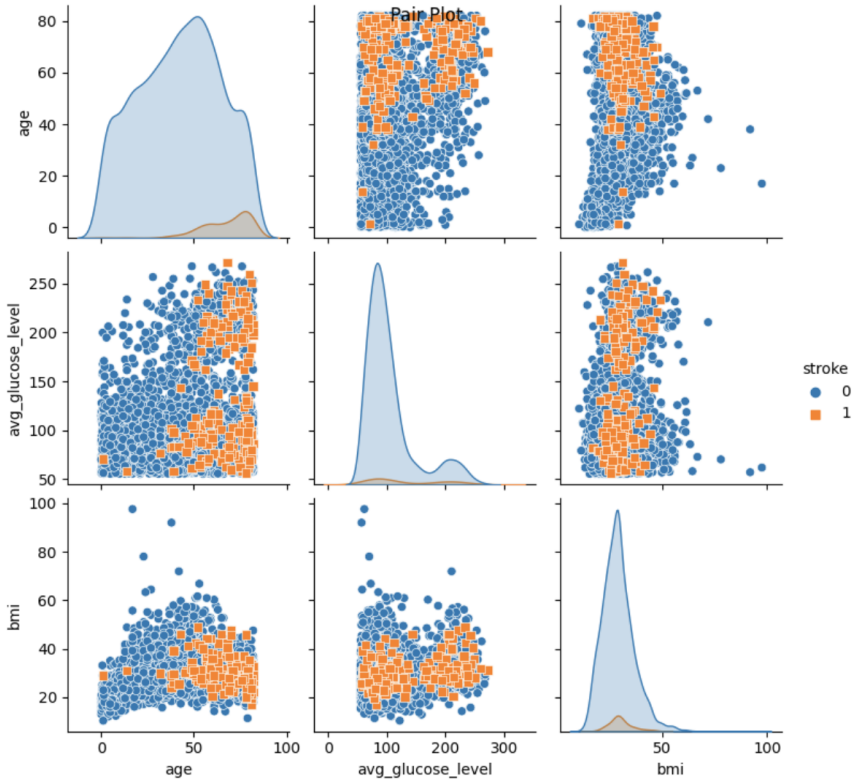

Figure 5: Pair Plot of BMI, Average Glucose Level, and Age with Respect to Stroke

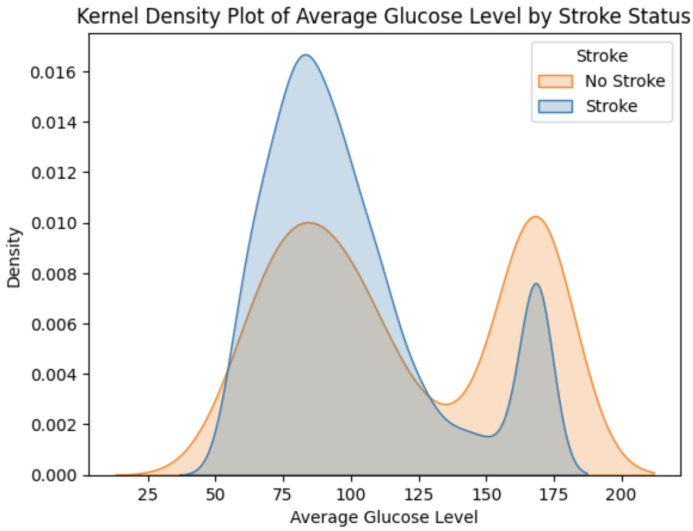

Figure 6: Kernel Density Plot of Average Glucose Level with Respect to Stroke

After creating graphs and visualizations that aid in our understanding of the different variable relationships, we then focused on building our predictive models to find the best model. The methodology used to determine the best model was a trial and error approach; we decided to build different models and compare them with each other. We created five different predictive models to compare with one another: the first was a numeric linear model, the second was a categorical linear model, the third was a combined (numeric and categorical linear model), the fourth was a KNN model, and the last model was a classification decision trees model. To determine how effective each model was, we calculated the R^2 and RMSE value for each model. Additionally, residual plots and plots of the True Values vs. Predicted Values (which can be found labeled by each model in the Appendix) were created for each model. By finding the metrics and making the appropriate plots, our methodology allowed us to interpret each model to determine which model was the best one. The model with the lowest RMSE on the testing data was determined to be the best model, as a low RMSE value indicates that the model fits the data well.

The first model we made was a numeric linear model where the target variable, ‘stroke’, was regressed on the numeric variables, ‘age’, ‘avg_glucose_level’, ‘bmi’, ‘hypertension’, and ‘heart_disease’, alone. The linear regression model was created, fitted, and then used to predict on the test set. The R^2 value was 0.0737 and the RMSE value was 0.2075. The kernel density plot for residuals was also found.

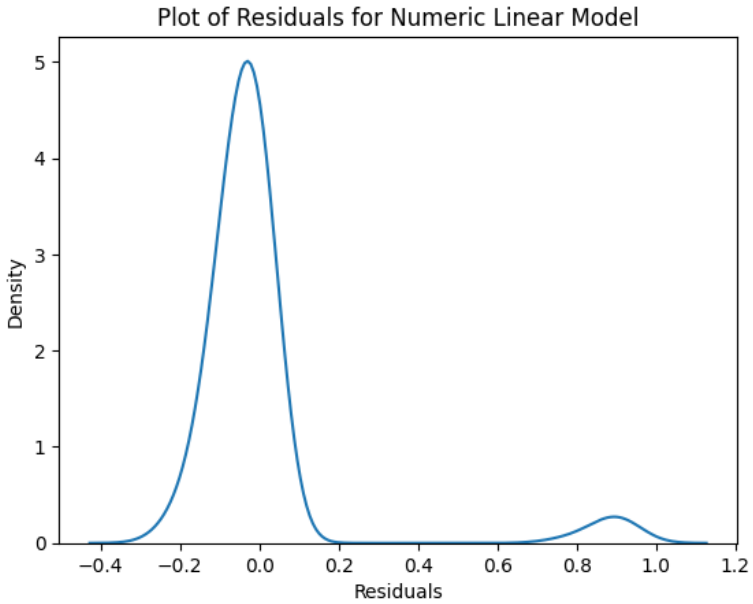

Figure 7: Kernel Density Plot of Residuals for the Numeric Linear Model

From this residual plot, it was observed that the distribution of residuals is not a normal distribution, as there is a heavy skew and a peak around -0.2 and 0. The non-normality of residuals indicates that this model is an inadequate model. The errors the model was making were not uniform, or randomly consistent across the variables; instead, the errors were skewed

and heavily focused around -0.2 and 0. Thus, this model wasn't an effective model and we decided to try another tactic.

Next, we moved onto building a categorical linear model, where the target variable was regressed on the categorical variables, 'ever_married', 'gender', 'residence_type', 'smoking_status', and 'work_type', alone. To do this, the categorical columns (for the training and testing data sets) needed to be one-hot encoded. The new encoded columns were then placed into a new, expanded dataframe, and then the linear regression model was created, fitted, and used to predict on the test set. The R^2 value was 0.0252 and the RMSE value was 0.2129. The kernel density plot for residuals was also found.

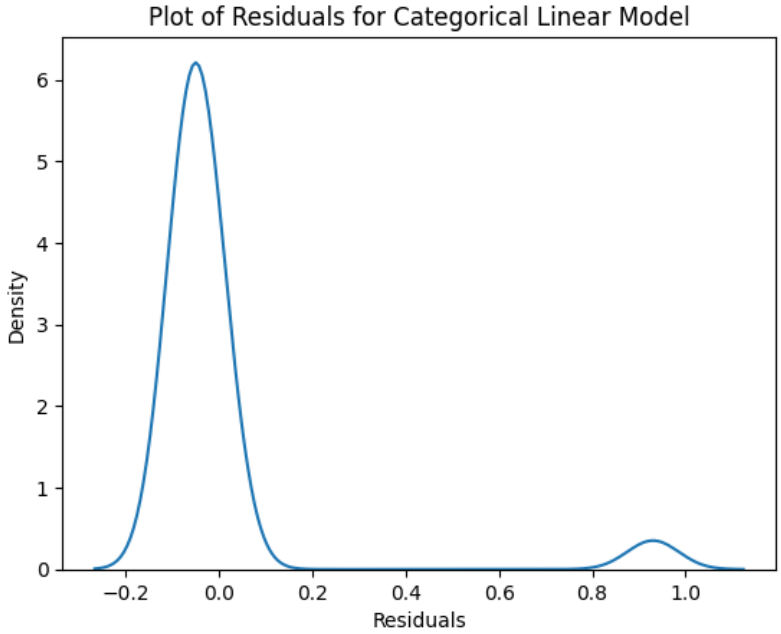

Figure 7: Kernel Density Plot of Residuals for the Categorical Linear Model

The categorical linear model did not perform significantly better than the numeric linear model. The RMSE value was higher than the RMSE value for the numeric linear model. As observed in the residual plot, the distribution of residuals is not a normal distribution, as there is also a heavy right skew and a peak around -0.2 and 0. The two linear models we built were not the best, so we decided to try to build one more linear model, a combined model of numeric and categorical columns, before moving onto non-linear models.

We decided to create one other linear model with hopes that having a combined model, with all the numeric and categorical columns, will perform better. This new model was created, fitted, and then predicted on the test set. The R^2 value was 0.0809 and the RMSE value was 0.2067. The kernel density plot for residuals was also found.

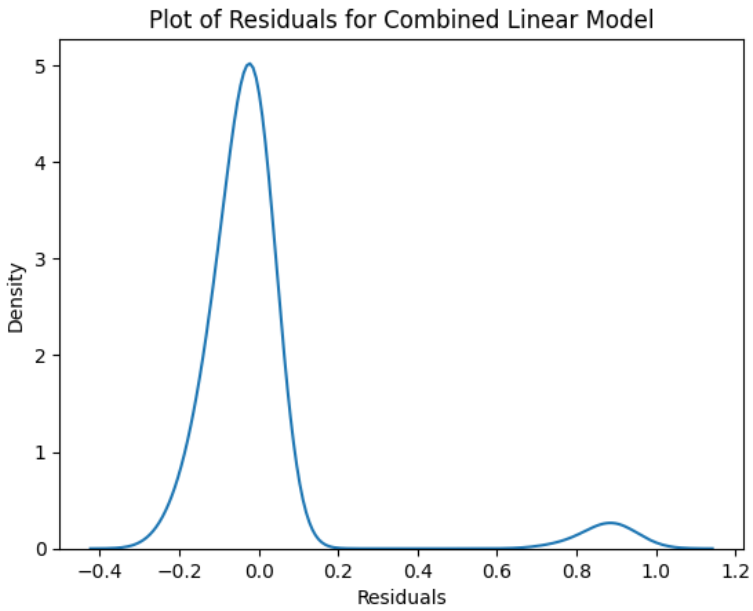

Figure 8: Kernel Density Plot of Residuals for the Combined Linear Model

Even though the distribution of the residuals was not normally distributed, and there was a peak at -0.2 and 0, as found in the other residual plots for linear models, the RMSE value of the combined model was the lowest when comparing with the numerical linear model and the categorical linear model. Thus, we concluded that the combined linear model performed the best of the linear models. Although the RMSE value was the lowest for the combined linear model, the difference of the RMSE value from the other two linear models was not significant enough to conclude the linear model is the best predictive model to determine the likelihood a person will have a stroke. Therefore, instead of building a linear model using polynomial degree expansion, we decided to move past linear models and build two other models, such as a KNN model and classification decision tree model, to determine if those models are better at predicting than the linear models.

The fourth model we built was a model using k nearest neighbors (KNN). The maxmin normalization function was applied to each column of X, and we picked an arbitrary value for the optimal k, which we decided on k=50. We then created, fitted, and then predicted the values on the test set. The R^2 value was 0.0531 and the RMSE value was 0.2098. The Sum of Squares of Errors (SSE) and residual plot was also found.

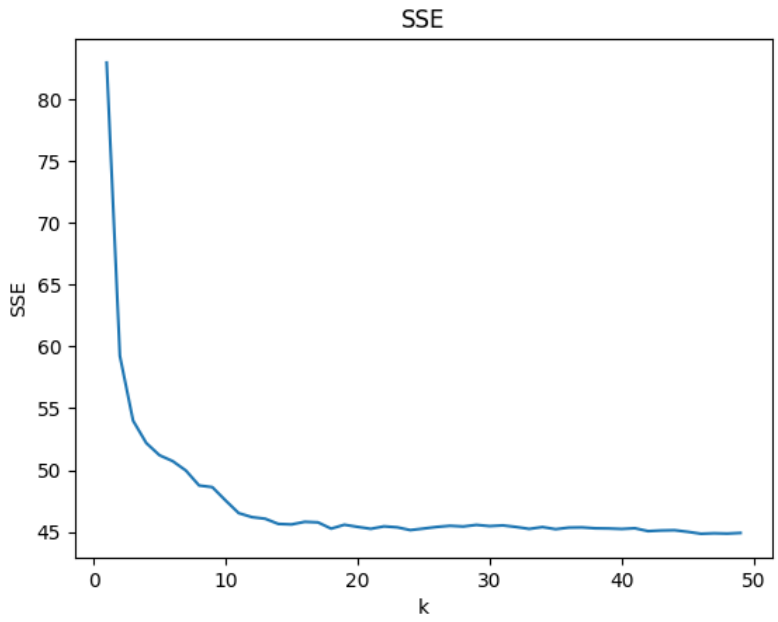

Figure 8: SSE Plot

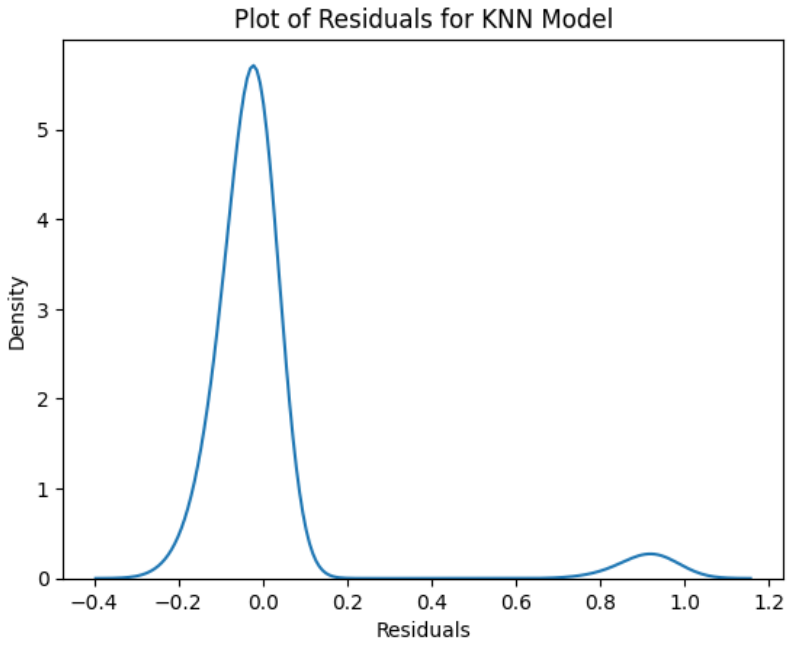

Figure 9: Kernel Density Plot of Residuals for the KNN Model

In the SSE plot found, it was observed that the SSE drops as k increases from 1 to 10, and then the SSE starts to flat line and decreases more slowly after that. The elbow for this plot seems to be around k=10, implying that increasing the number of clusters beyond k=10 does not help reduce the SSE significantly. Therefore, a valid number of clusters that can be chosen for this data is k=10. The residual plot for the KNN model is similar to the other three linear models made: there is non-normality in the distribution of the residuals and a peak around -0.2 and 0.

The RMSE value for the KNN model was also higher than the “better” model we found earlier, which was the combined linear model. Thus, it did not seem that the KNN model performed any better than the other models already created.

The next and final model we tried was the classification decision tree model. The model was created, fitted, and predicted on the test set. The R^2 value was -0.4549 and the RMSE value was 0.2601. A visualization of the decision tree, and the kernel density plot for residuals was also found.

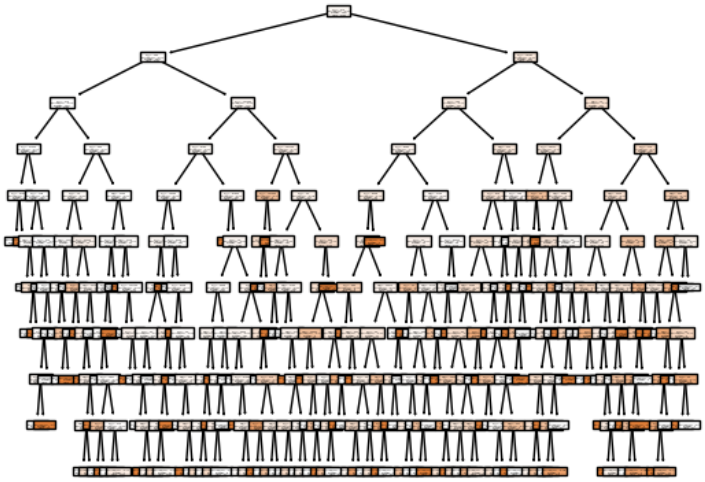

Figure 10: Visualization of the Classification Decision Tree

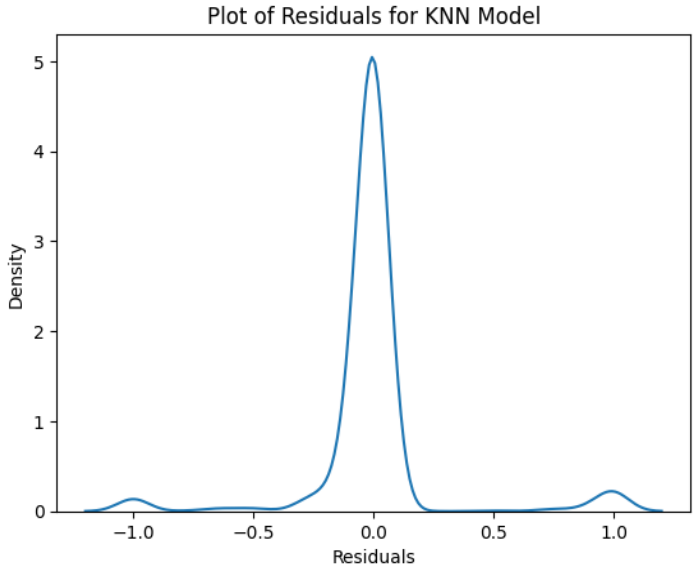

Figure 11: Kernel Density Plot of Residuals for the Classification Decision Tree Model

From the R^2 and RMSE values found, it was observed that the classification decision tree model was not any better than the KNN model and the linear models that were built. The

RMSE value for this model was the highest out of any models, and the R^2 value was negative, implying that the model did a poor job of fitting the model. Thus, this model can be seen as the model that did the worst.

After much trial and error, it was found that the combined linear model was our “best” model because it had the lowest RMSE value. However, when looking at the residual plot for the combined model, and the True Values vs. Predicted Values Plot, found in the appendix, it wasn’t concluded that the linear model was the most efficient model. The distribution of these plots were not normally distributed, and there seemed to be skew in the residual plot.

Conclusion

The primary objective of this project was to practice developing predictive algorithms to predict the likelihood of a person having a stroke. Being able to detect or predict a stroke are crucial for taking proactive preventative measures, as the impact of a stroke can range from mild to life threatening. Thus, being able to predict and take these early critical decisions can be beneficial to thousands of individuals. The dataset that we used to create our predictive models contained 12 variables where the stroke variable was our target variable, and the predictor variables were age, average glucose level, BMI, marital status, gender, heart disease, hypertension, residence type, smoking status, and work type. In order to prepare our data for the algorithms, we implemented several data wrangling techniques that involved handling missing values, dropping irrelevant columns, and addressing our outliers using winsorization. After cleaning the data, exploratory data analysis was implemented where we produced plots such as box plots, pair plots, and kernel density plots to understand the relationships between significant variables and identify outliers and patterns present in the data. The most notable variable relating to stroke that we gathered from the preliminary data analysis was that age appeared to have the most significant impact on the likelihood of an individual having a stroke, where higher age correlated to more strokes. Following the data wrangling and visualization, we proceeded with building our predictive models for our question that we were exploring.

The next step in our exploration of the data was using machine learning algorithms to create predictive models that will predict the likelihood of a person having a stroke. The models that we created were linear regression models, KNN, and decision tree. There were three regression models created: the first used numerical variables, the second used categorical variables and the third used a combination of the numeric and categorical variables. The RMSE and R^2 values are as follows:

Model	RMSE	R^2
Numeric Linear Regression	0.20786	0.07518
Categorical Linear	0.212875	0.02518

Regression		
Combined Linear Regression	0.20695	0.078627
KNN	0.209801	0.053138
Decision Tree	0.25628	-0.41285

A low RMSE value was wanted for the best predictive model as a low RMSE value implied the model was able to predict the target variable more accurately. Although all our linear models were in the same range of being in between "0.2..", the combined linear model had the lowest RMSE value, so we concluded that the combined model's predictions were better. Additionally, the R^2 value was also computed. The R^2 value represents the percentage of the dependent variable variation that a model explains, and it is better to have a higher R^2 value as that implies there are smaller differences between the observed data and the fitted values. The combined linear model also had the highest R^2 value. Therefore, we concluded that our best model was the combined linear model. As mentioned before, even though this model had the lower RMSE value, the model was not the most accurate or efficient. The distribution of the residual plot and the True Values vs. Predicted Values plots (found in the appendix) were not normally distributed, implying that the model was making systematic errors and there was a skew in the underlying data even after the outliers for the 'avg_glucose_level' and 'bmi' variable were handled. On the other hand, our worst model was the classification decision tree model; the RMSE value was the highest and the R^2 value was the lowest as it was negative. Thus, after much trial and error, we were able to conclude a best and worst model.

One criticism some might have is that our RMSE values were similar. Although all of our RMSE values were rather similar, our data wrangling was carefully considered as there were several outliers that could have heavily influenced the predictive models, so we did our best to make sure we weren't overfitting the data and manipulating it to the point it would negatively influence our predictive models. After concluding this project, we would like to further explore other predictive models, such as K-means, random forest, or neural networks, to try and reduce our RMSE value and to get our predictions to be even more accurate and try different strategies to handle outliers.

Appendix

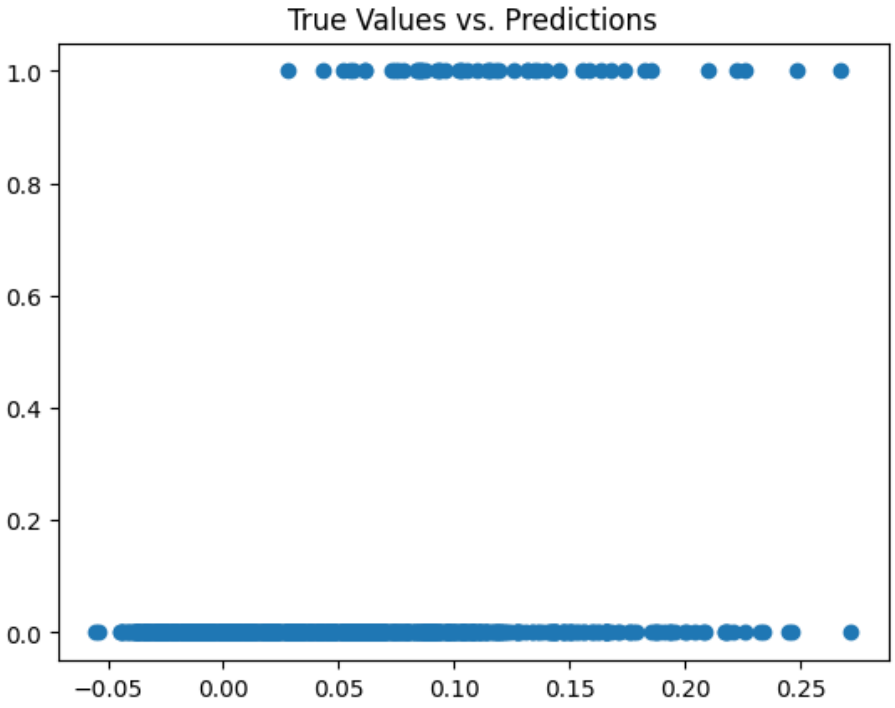

Figure 1: True Values vs. Predictions Plot for Numeric Linear Model

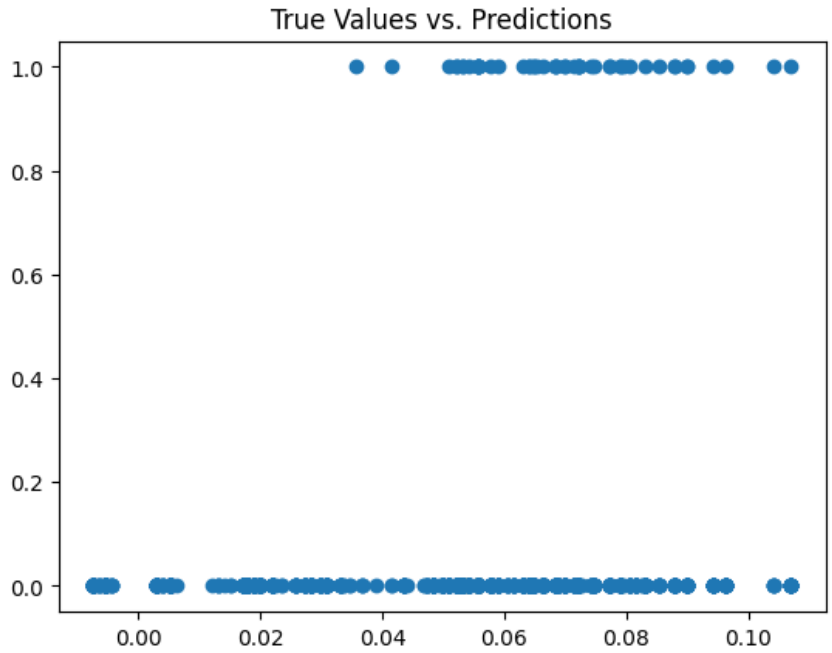

Figure 2: True Values vs. Predictions Plot for Categorical Linear Model

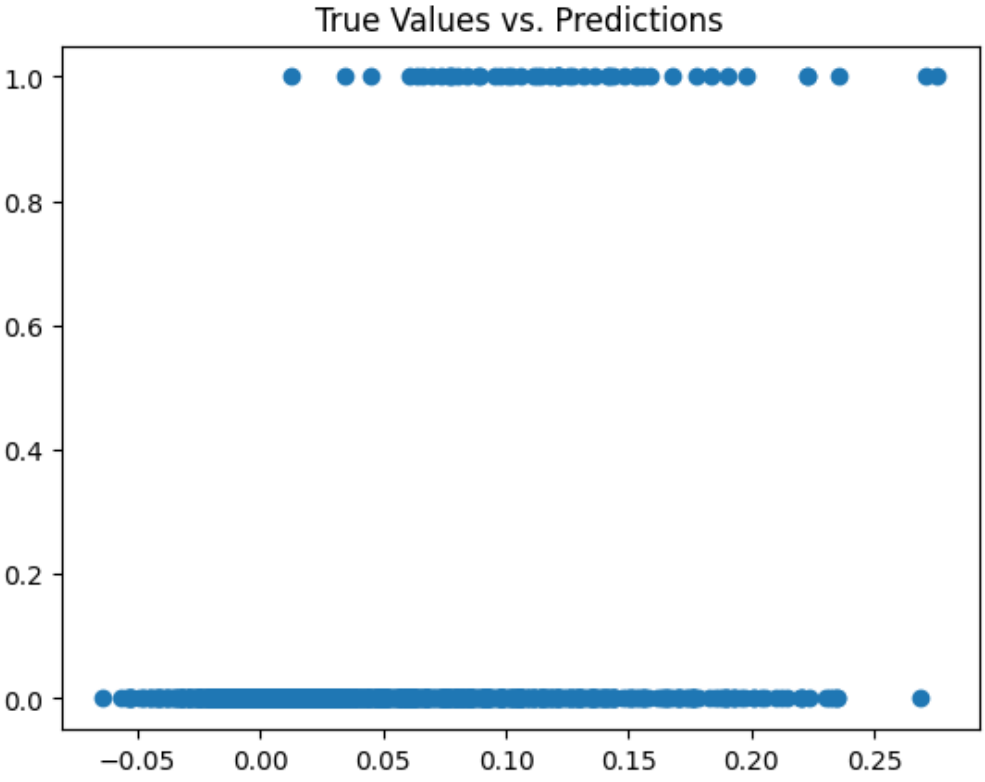

Figure 3: True Values vs Predictions Plot for Combined (Numeric and Categorical) Linear Model

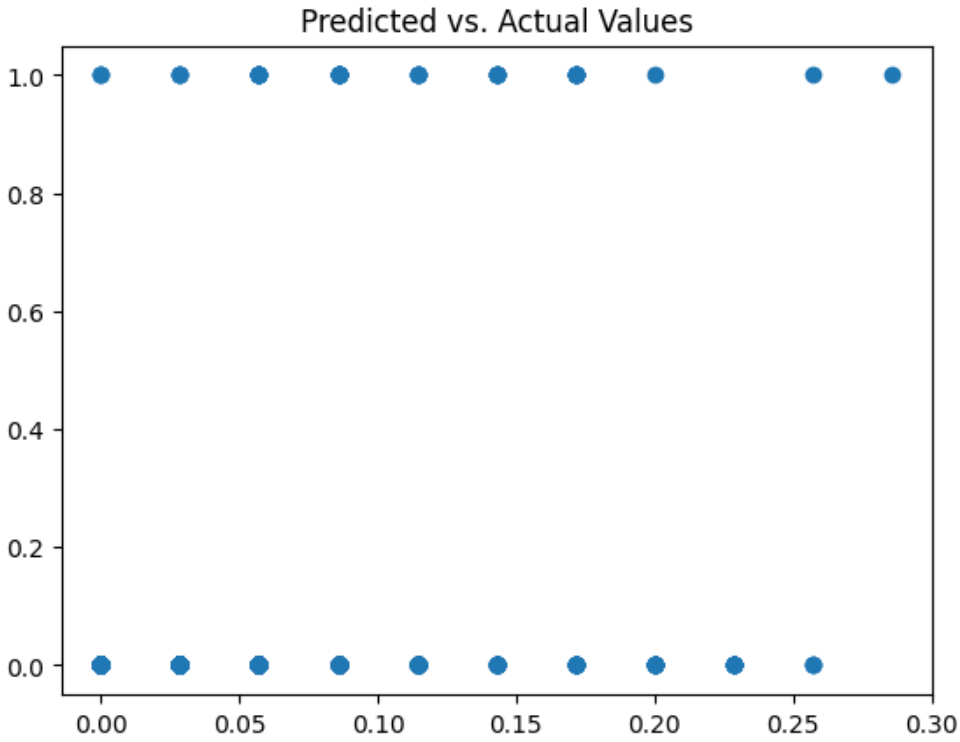

Figure 4: Predicted vs. Actual Values for KNN Model

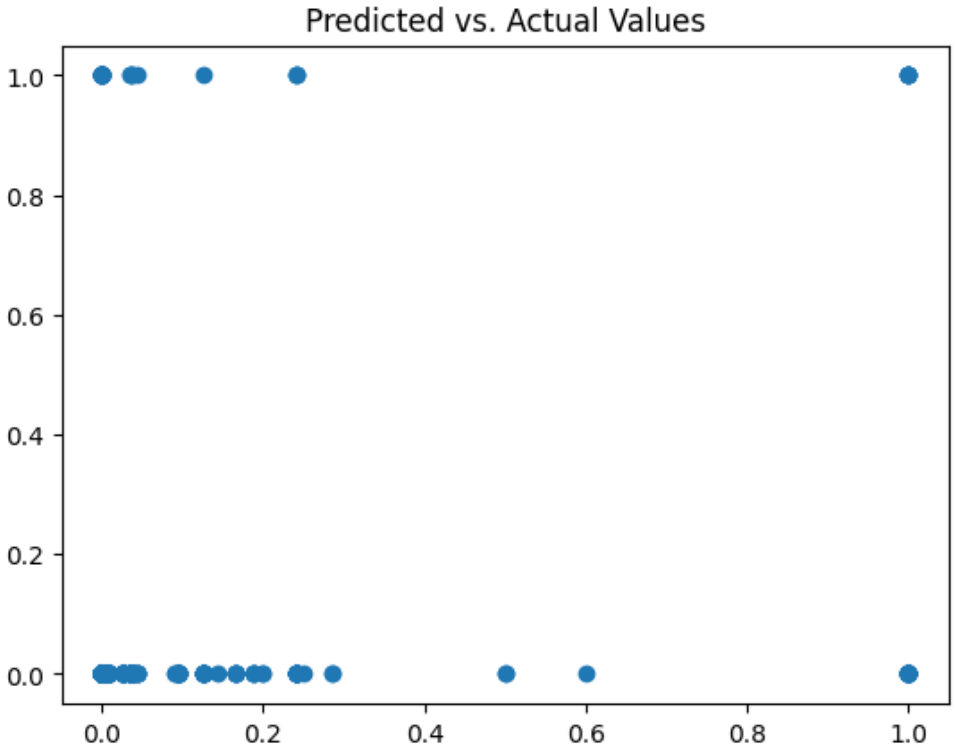

Figure 5: Predicted vs. Actual Values for Classification Decision Tree Model