Stroke Prediction

Aiden Clark - CS 4300

December 11, 2021

Contents

Introduction

Relevant Information: This data set is used to potentially predict strokes based on a series of parameters like age, gender, smoking status, and various diseases. The Center for Disease Control and Prevention (CDC) cites strokes as the fifth leading cause of death in the United States. The World Health Organization (WHO) cites strokes as the second leading cause of death globally [4].

Strokes are very hard to predict. Mainly because strokes are the result of various syndromes and diseases. This data-set attempts to predict strokes based off relevant information about the patient.

The data-set is sourced from Kaggle. The source is confidential, and is used for educational purposes only.

Google Colab Link: `https://colab.research.google.com/drive/1rb5a8zgvoBT8OMAZYEyoXOAnLqVHPMc9?usp=sharing`

1 Dataset

The "Stroke Prediction Dataset" was obtained from Kaggle data set database. The data was obtained from a confidential source meant for educational purposes only. The data was published by Fedesoriano- a data scientist at Kaggle. As mentioned above, the attributes take in measure relevant information about the patient.

The original data has been modified to better suit the machine learning technology of Tensorflow Keras. For instance, the ID numbers were ranked numerically instead of the original unique identifiers. The status of the diseases, work types, residence types, and smoking status have all been given numerical values in place of the string values.

NOTE: a small number of data was voided from the final cleaned data-set because of missing attributes.

1. ID - numbered identifier .
2. Gender - Male (0), Female (1) or Other (2).
3. Age - age of the patient.
4. Hypertension - 0 if the patient doesn't have hypertension, 1 if the patient has hypertension.
5. Heart disease - 0 if the patient doesn't have any heart diseases, 1 if the patient has a heart disease.
6. Ever Married - No (0) or Yes (1).
7. Work Type - Children (0), Government job (1), Never worked (2), Private (3) or Self-employed (4).
8. Residence Type - Rural (0) or Urban (1).
9. Average Glucose Level- the average glucose level in blood.
10. BMI - body mass index.
11. Smoking Status - formerly smoked (0), never smoked (1), smokes (2) or Unknown (3).
12. Class Output Label: stroke: 1 if the patient had a stroke or 0 if not.

1.1 Visualization of the distribution of each input features

The attributes are logically graphed. This includes scatter plots, distribution graphs, and histograms. Featuring their high and low values, and their distributions.

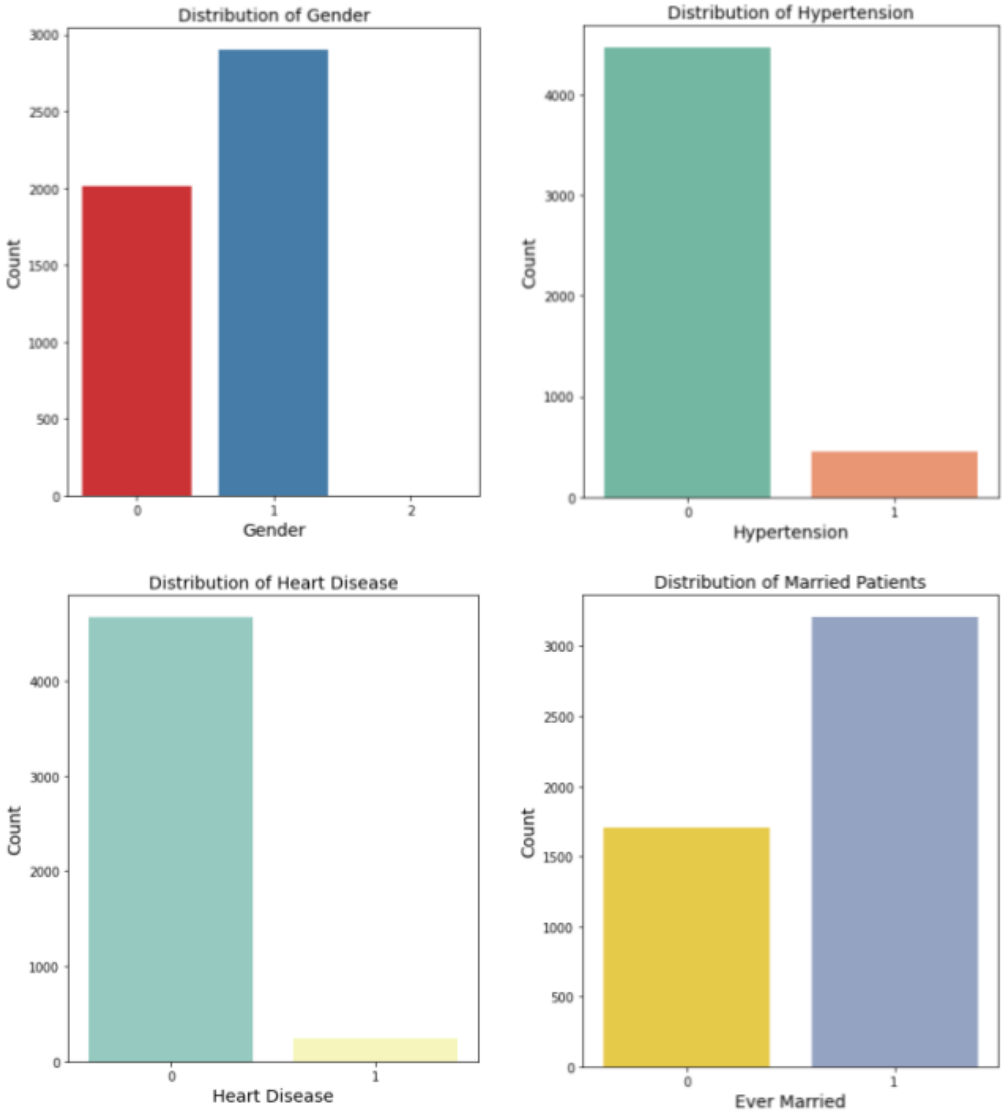

Figure 1: Distributions of gender, hypertension, heart disease, and marital status

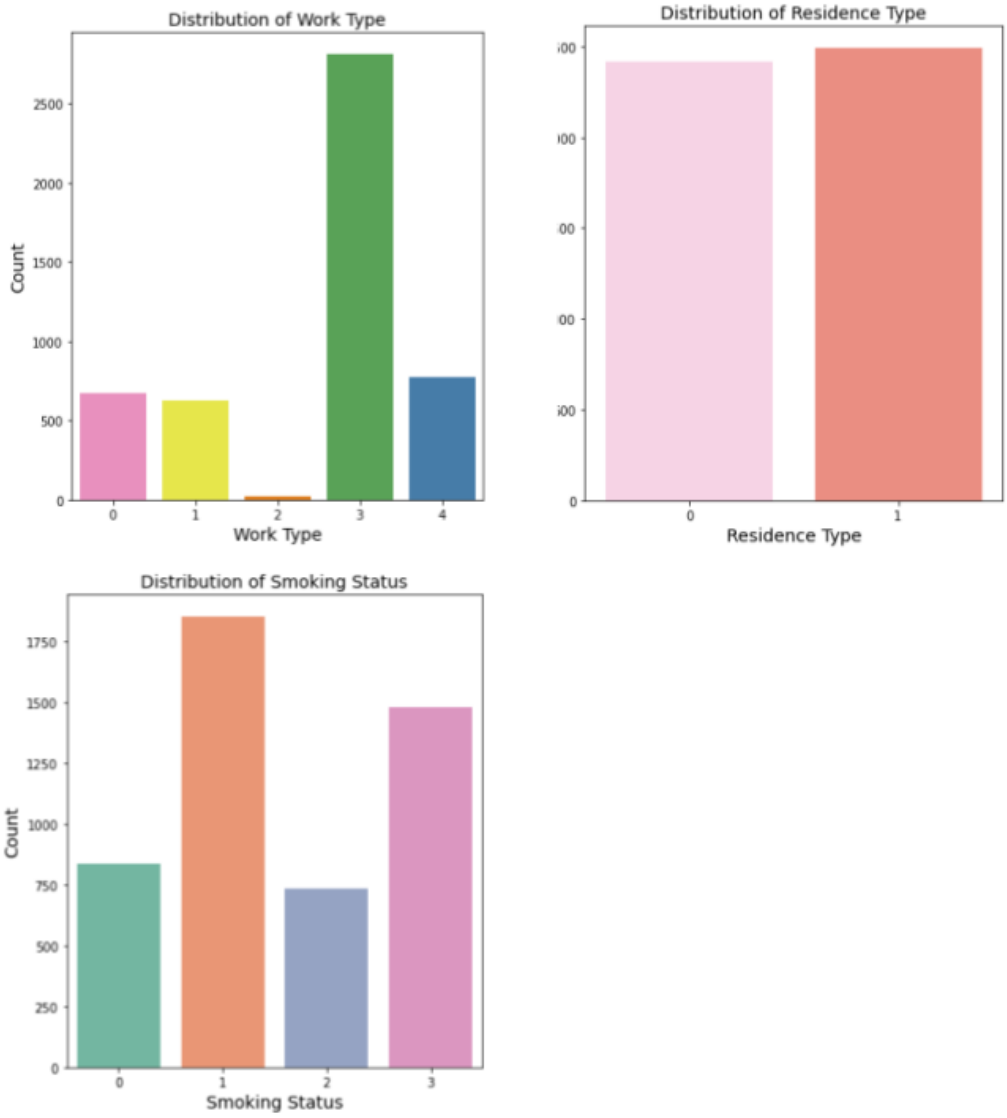

Figure 2: Distributions of work type, residence type, and smoking status

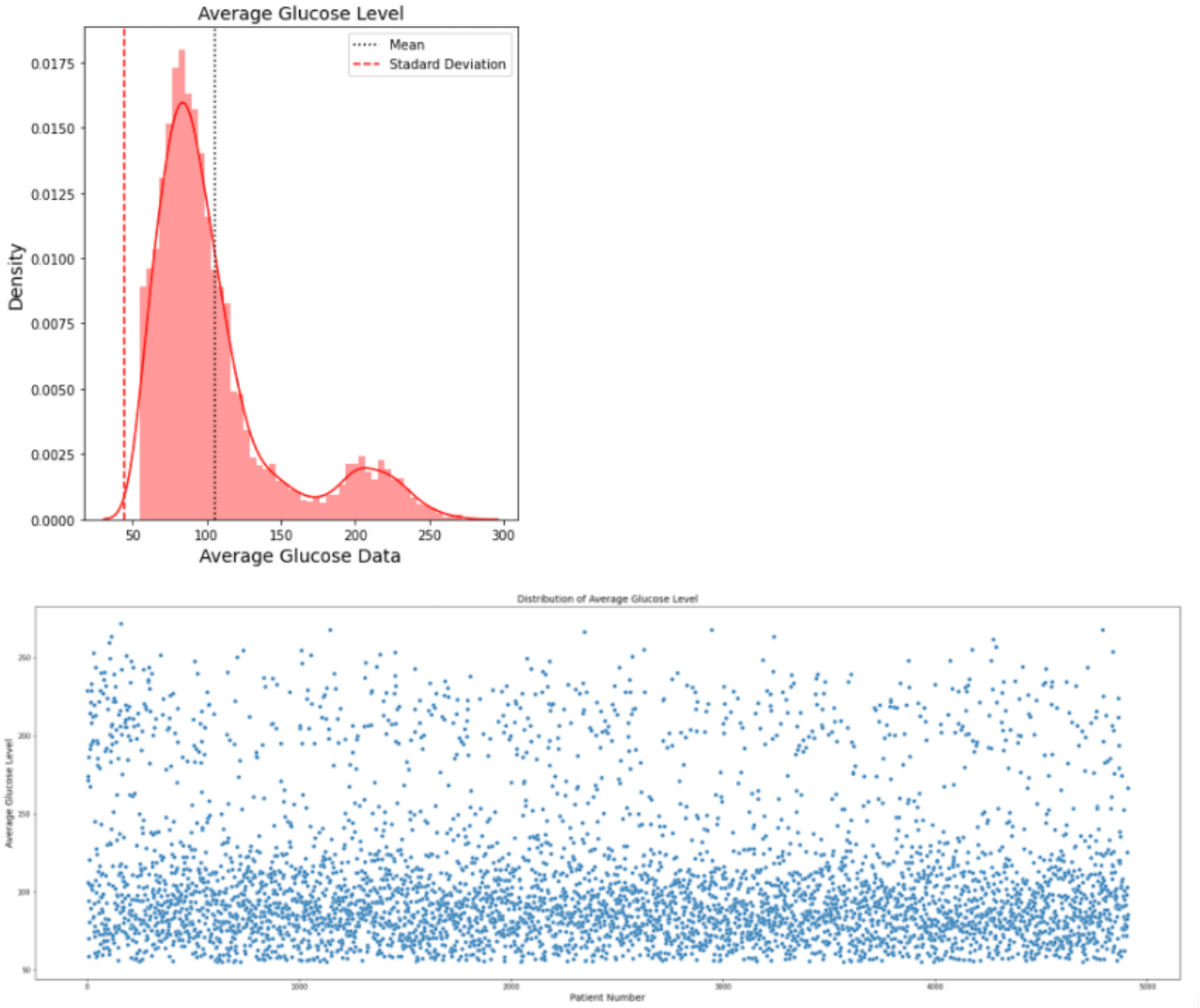

Figure 3: Distributions of Patients Average Glucose Levels

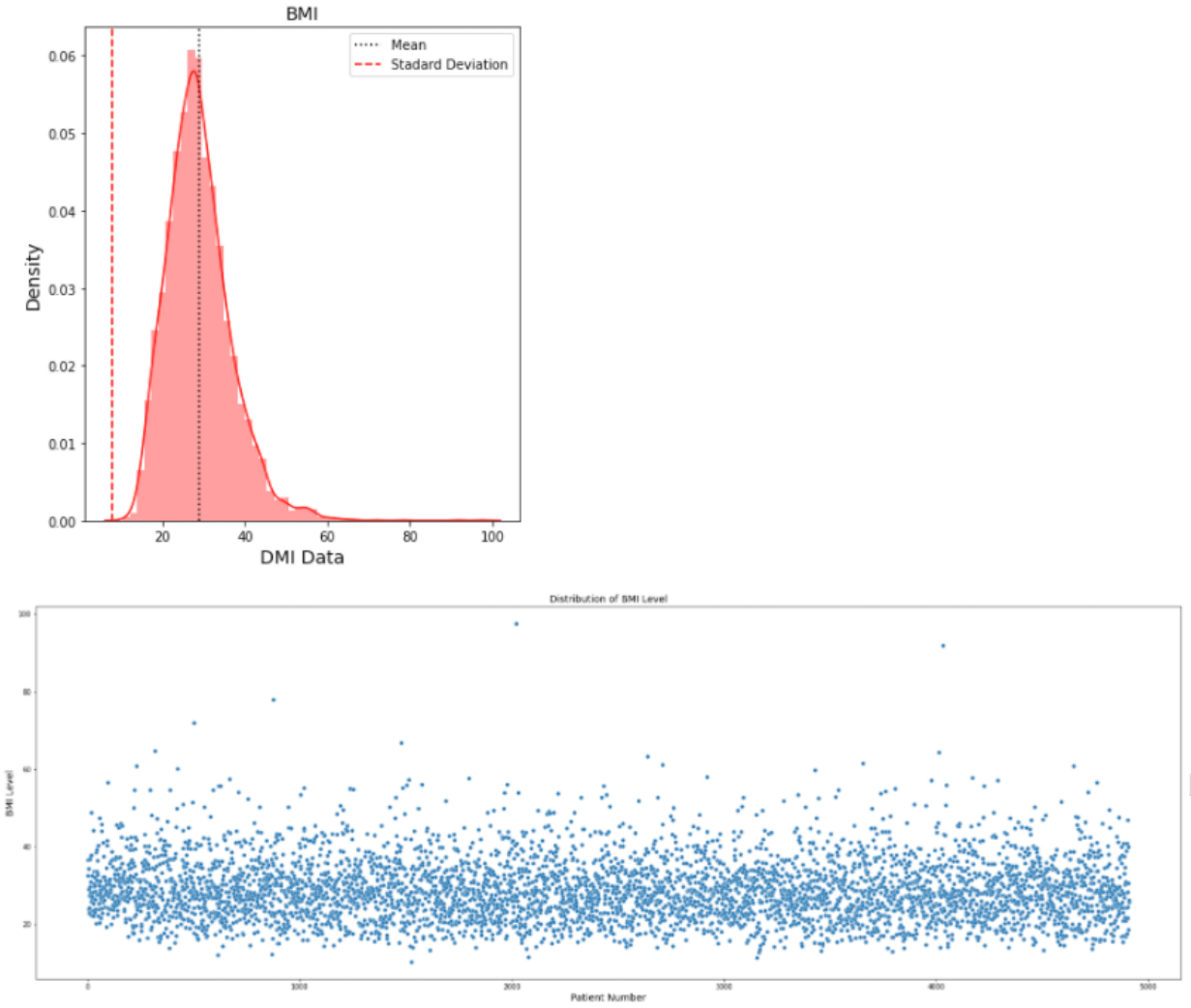

Figure 4: Distributions of Patients BMI

1.2 Distribution of Output Labels

This is a large under sampling of the stroke target class. There are 248 instances of strokes in the data set. Substantially smaller than the non-stroke instances. This creates problems when reproducing the training and validation. Most of the time the models will fail to predict strokes. A small sample size in the training does not help.

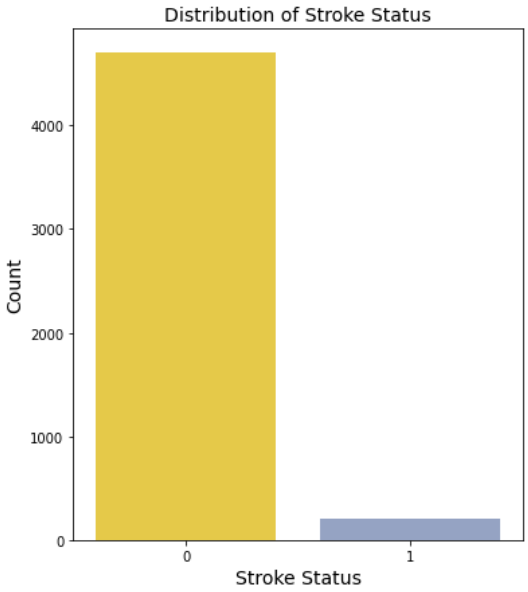

Figure 5: Output Data of Stroke Classifier

2 Data Processing

2.1 Data Splitting

There are two types of approaches that were used in this data to compensate for the under sampling. One, the data used was shuffled at random and was split into training and validation. 80% of the data set was partitioned for training whilst the other 20% was partitioned to validation with no over sampling. The second approach is over sampling with SMOTE to get better results for validation. The following training and validation data comes from the second approach.

2.2 Data Normalization

Data normalization makes training less sensitive to the scale of features. It makes it easier to solve for coefficients. Normalization also aims to turn all values in zeros or ones. Hopefully eliminating outliers, but keeping them visible to the normalized data. Min-max normalization and mean normalization, are two normalization techniques. The following data is min-max normalized.

Mean Normalization Formula

$$X_{new} = \frac{X - X_{min}}{X_{max} - X_{min}}$$

Min-Max Normalization Formula

$$X_{new} = \frac{X - X_{mean}}{X_{max} - X_{min}}$$

2.3 Normalized Data

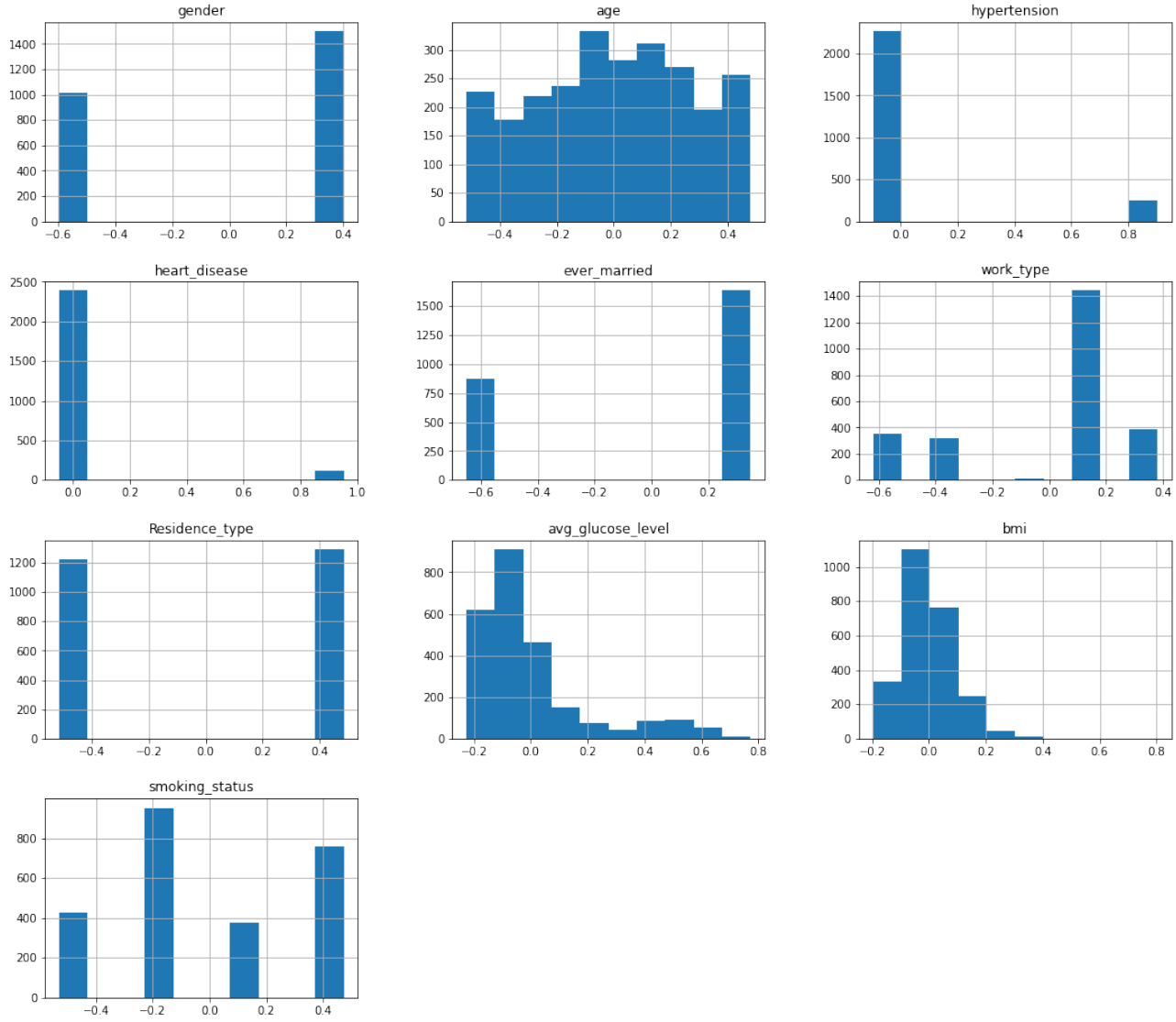

Figure 6: Output Data After Min-Max Normalization

From the visualized data and the data normalization it is obvious that heart disease and hypertension are related to the number of stroke instances.

3 Using SMOTE to Combat Under-sampled Classifier Class

SMOTE Stands for Synthetic Minority Oversampling Technique. The goal is to over sample the minority stroke class. SMOTE doesn't add duplicates of the class but instead syntheses from existing samples [2].

The trouble with the current imbalanced data set is that the learning techniques are producing poor performance. The sample size is simply too small to have accurate predictions. On top of that, this limited data set is predicting strokes, which is also a difficult task.

3.1 How SMOTE Works

SMOTE selects examples in the data set that are close in feature space. It then draws a line between examples in the feature space. It will then draw a new sample from a point along that line.

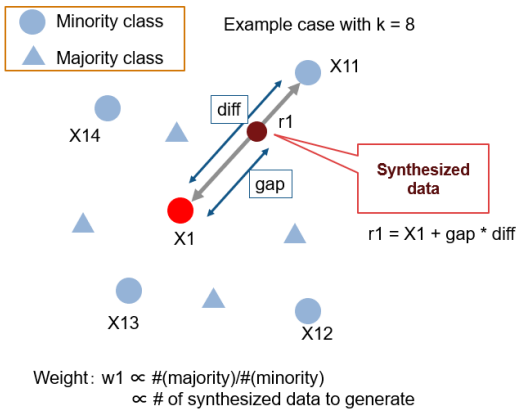

Figure 7: SMOTE Visualization [6]

	Values Before Oversampling	Vales After Oversampling
XTRAIN	3462	6592
YTRAIN	3462	6592

Stroke Classifier	Values Before Oversampling	Vales After Oversampling
0	3296	3296
1	140	3296

4 Feature Removal

Feature removal is the process of constructing combinations of selected features to get the best possible accuracy for the models. This section will rank and remove features for optimal accuracy.

4.1 Feature Scoring

Feature	Score
Age	2432.270102
Average Glucose Level	1546.894603
Heart Disease	62.921251
Hypertension	44.702661
Work Type	17.497775
Married Status	15.258000
Smoking Status	13.734744
BMI	5.804488
Residence Type	0.075376
Gender	0.001395

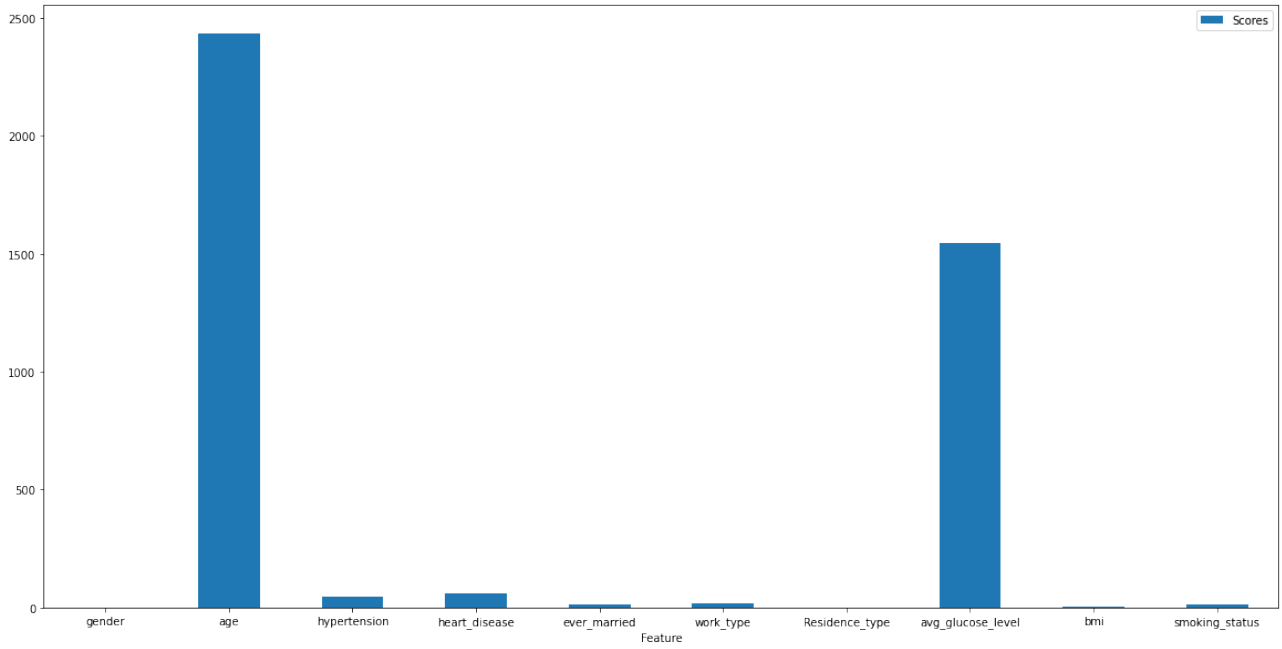

Figure 8: Graph Comparing Feature Importance

	Accuracy
Baseline Model	95.60

4.2 Feature Ranking

Each of the features received a ranking from one to six. Six being the most important where 1 is the least important.

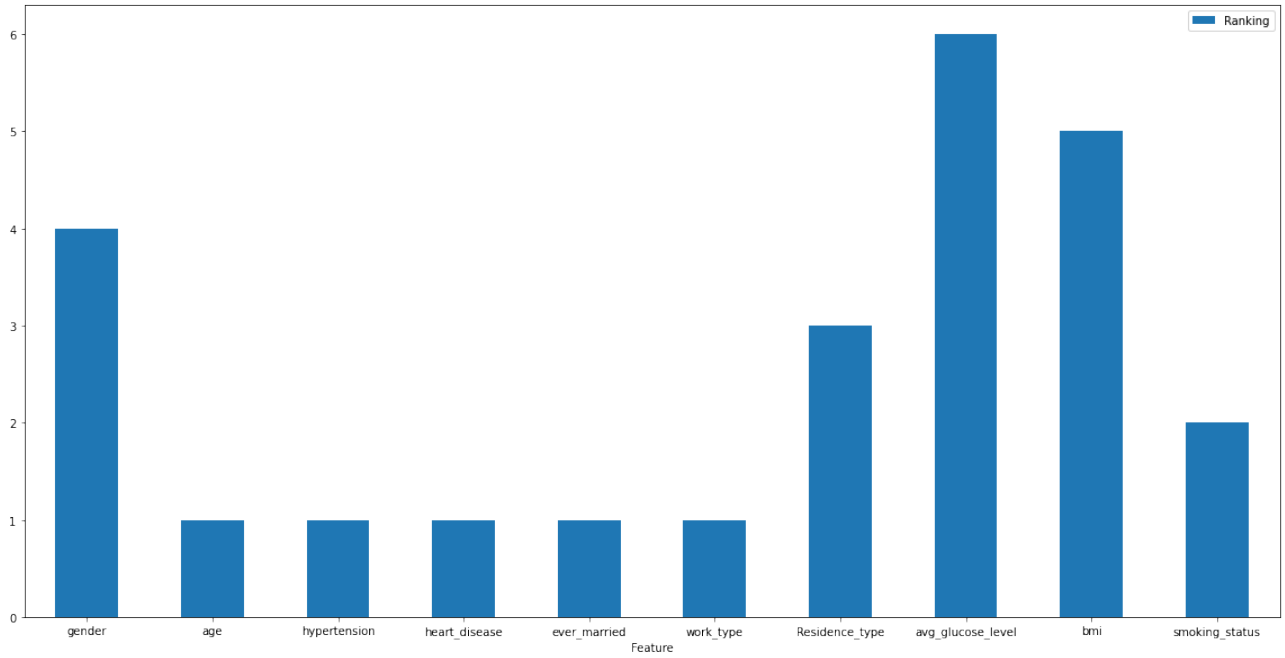

Figure 9: Feature Ranking

4.3 Recursive Feature Elimination

Recursive Feature Elimination (RFE) is a feature selection process that fits a model and removes the weakest features until highest accuracy is reached [1]. RFE produced eight optimal number of features:

1. Age
2. Hypertension
3. Heart disease
4. Ever Married
5. Work Type
6. Average Glucose Level
7. BMI
8. Smoking Status

	Accuracy
Baseline Model (with selected features)	68.84

5 Modeling

The data was evaluated using neural network architecture.

Neural networks are comprised of individuals cells that are called neurons. They aim to mimic the structure of a brain. This section will analyze the different activation functions on their effectiveness on predicting outcomes.

5.1 Baseline Model Performance

The first model that was tested was the baseline model. It is a logistic regression model to act as the control for the other models tested.

	Accuracy	Loss
Baseline	69.72	60.90

5.2 Learning Curve of Baseline Model

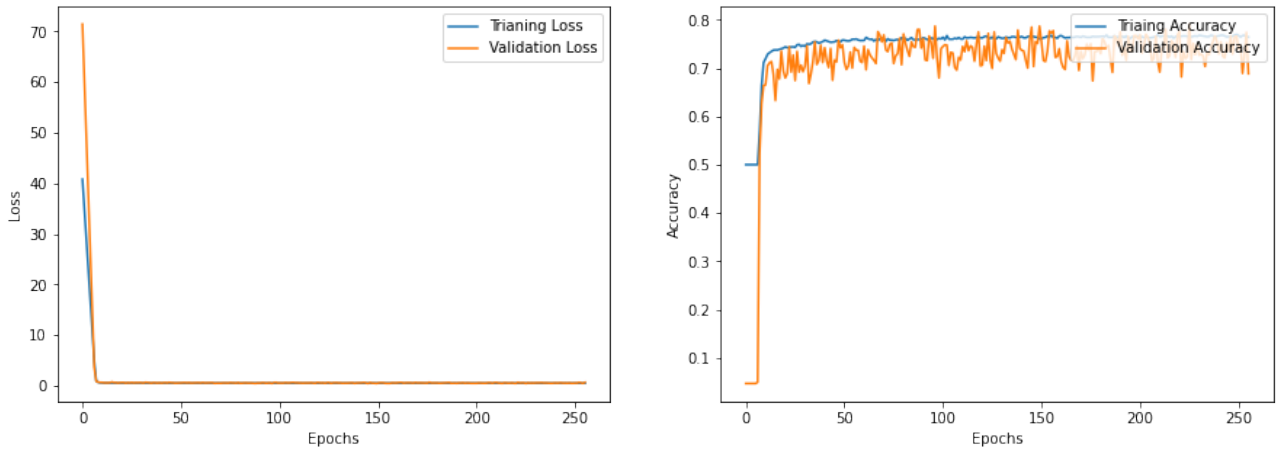

Figure 10: Baseline Model Learning Curve

5.3 Activation Models

In these trials the linear activation function preformed better than the sigmoid activation function. The linear activation function used mean absolute error (MAE) to calculate loss. Even though the linear activation function provided better results, data that is taken from normal distribution is better suited. As opposed to binary classification, which is what this data-set used. Loss cannot always be minimized binary examples. The following learning curves is a visual representation of the loss function.

Activation Function	Accuracy	MAE	Loss
Linear (Last Neuron)	-	29.28	16.40
Linear (All Neurons)	-	29.50	13.69
Sigmoid (Last Neuron)	77.05	-	46.36
Sigmoid (All Neurons)	77.39	-	43.28

5.4 Learning Curves of Linear Activation Model (Last Neuron)

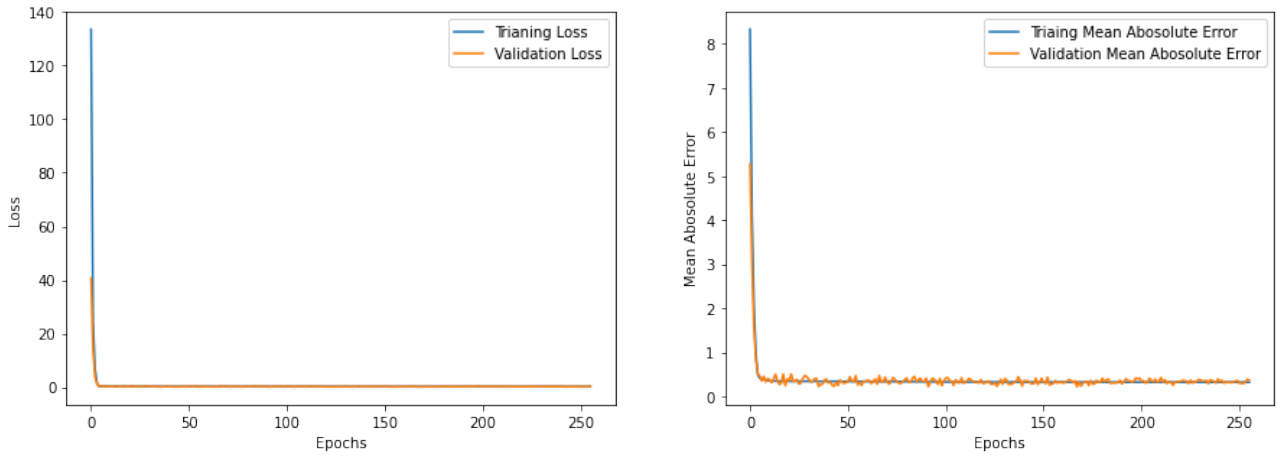

Figure 11: Graph Comparing Loss/MAE to Epochs

5.5 Learning Curves of Linear Activation Model (All Neurons)

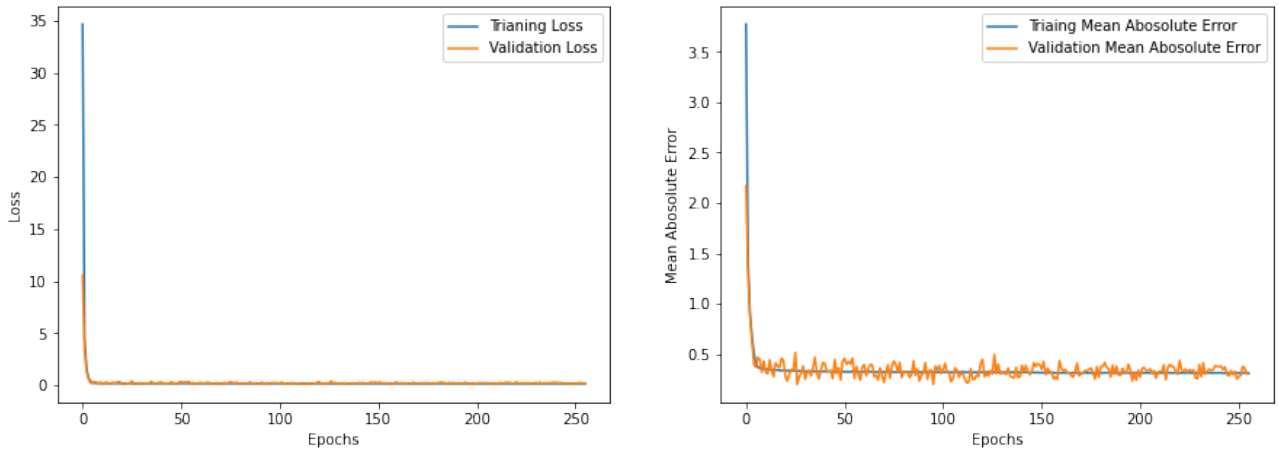

Figure 12: Graph Comparing Loss/MAE to Epochs

5.6 Learning Curves of Sigmoid Activation Model (Last Neuron)

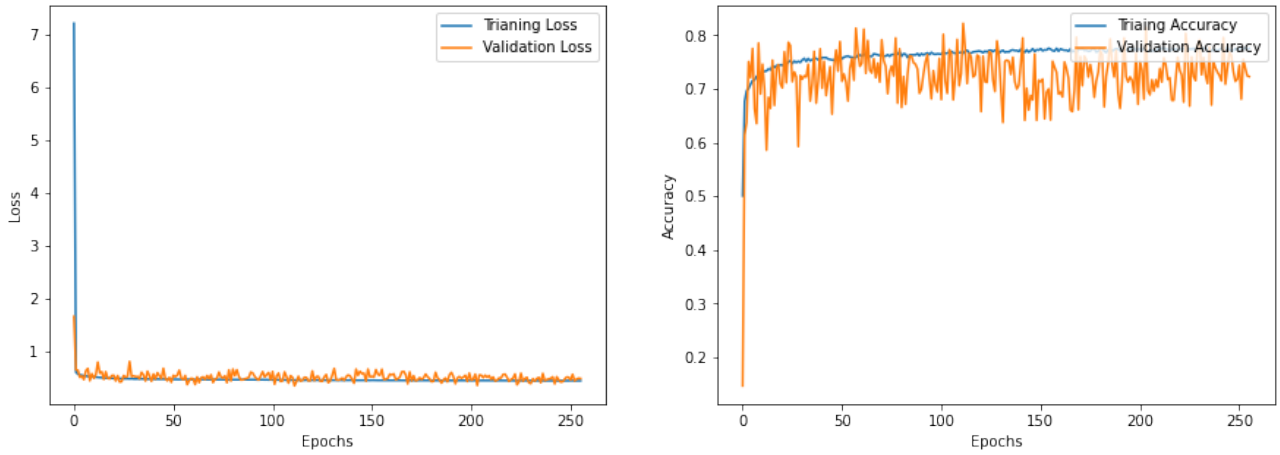

Figure 13: Graph Comparing Loss/Accuracy to Epochs

5.7 Learning Curves of Sigmoid Activation Model (All Neurons)

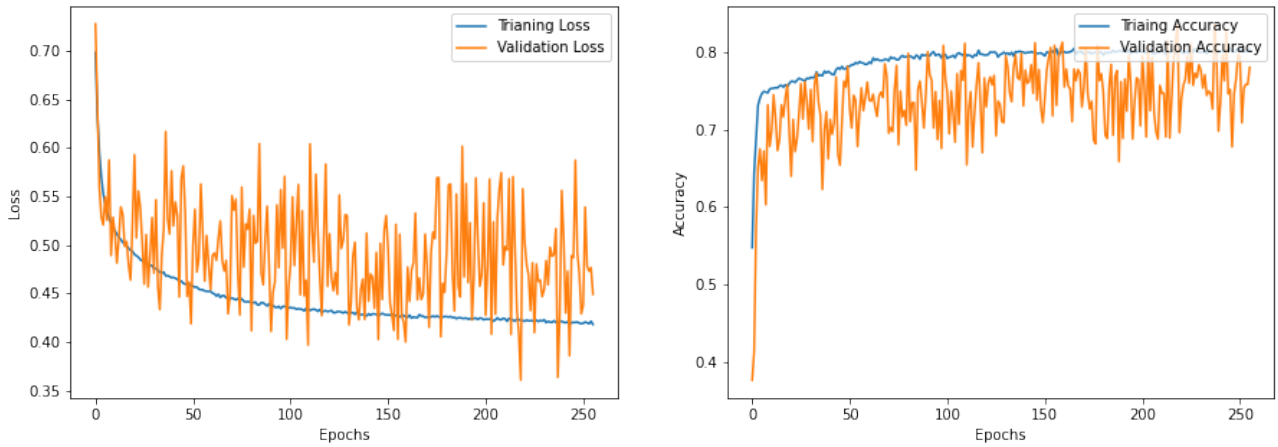

Figure 14: Graph Comparing Loss/Accuracy to Epochs

5.8 Learning Curves of Overfit Model

In this example the first layer has 1 neuron. The second layer has 40 neurons, and the last layer has 80 neurons. As expected there appears to be an over-fitting problem with the model on the accuracy curve.

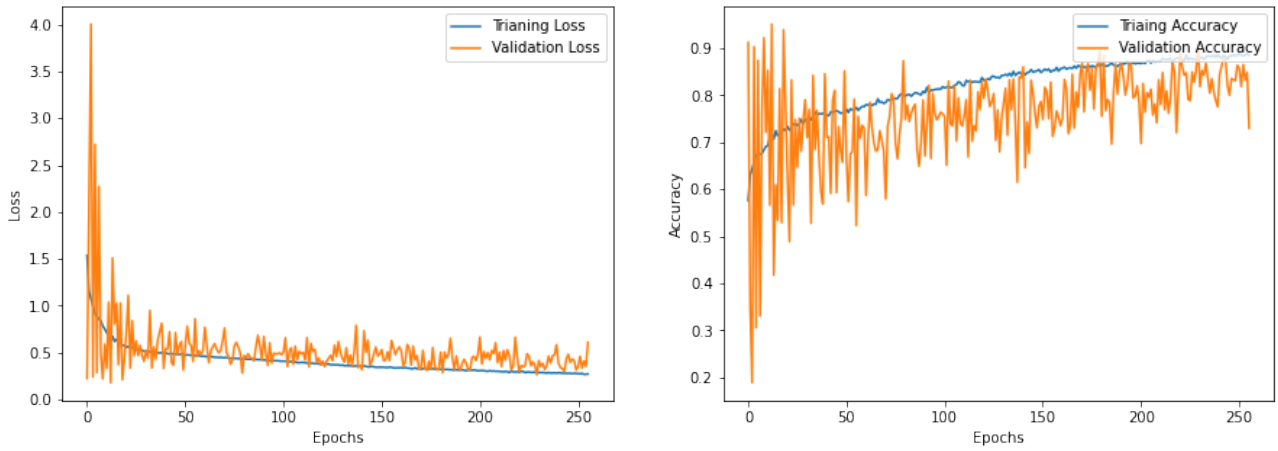

Figure 15: Graph Comparing Loss/Accuracy to Epochs

	Accuracy	Loss
Overfit	68.84	63.56

6 Evaluation

These models were evaluated off three types of classifications: precision, recall and f1 score.

1. Precision - measures the percent of positive target identifications. In this instance, we are looking for positive identifications of fire based off of the data provided.
2. Recall - measures the percentage of positive target identifications that were *identified* correctly .
3. F1 Score - binary classification to measure the tests accuracy.

6.1 Training Data Evaluation

Model	Accuracy	Precision	Recall	F1 Score
Baseline	75.17	11.54	76.43	20.06
Linear (Last Neuron)	71.33	71.33	71.33	71.3329
Linear (All Neurons)	89.29	89.29	89.29	89.2899
Sigmoid (Last Neuron)	71.83	71.83	71.83	71.8277
Sigmoid (All Neurons)	78.87	78.87	78.87	78.8708
Overfit	83.93	83.93	83.93	83.9348

6.2 Validation Data Evaluation

Model	Accuracy	Precision	Recall	F1 Score
Baseline	74.54	13.22	79.71	22.68
Linear (Last Neuron)	71.22	71.22	71.22	71.2152
Linear (All Neurons)	87.71	87.71	87.71	87.7122
Sigmoid (Last Neuron)	71.62	71.62	71.62	71.6225
Sigmoid (All Neurons)	78.00	78.00	78.00	78.0041
Overfit	81.19	81.19	81.19	81.1948

6.3 Evaluation Using ROC Curve

AUC / ROC curves is a performance measure for classification problems. In these graphs, ROC (Receiver Operating Characteristic) curves is a probability curve with the AUC (Area Under the Curve) is the measure of how capable the outputs can be separated [5]. Essentially, these graphs tell us how effective the models are at predicting class outputs. In this case the graph is predicting 0 and 1. The y-axis measures true positive rates, and the x-axis measures false positive rates. The higher the AUC value is the better the model is at predicting strokes based off of the data set.

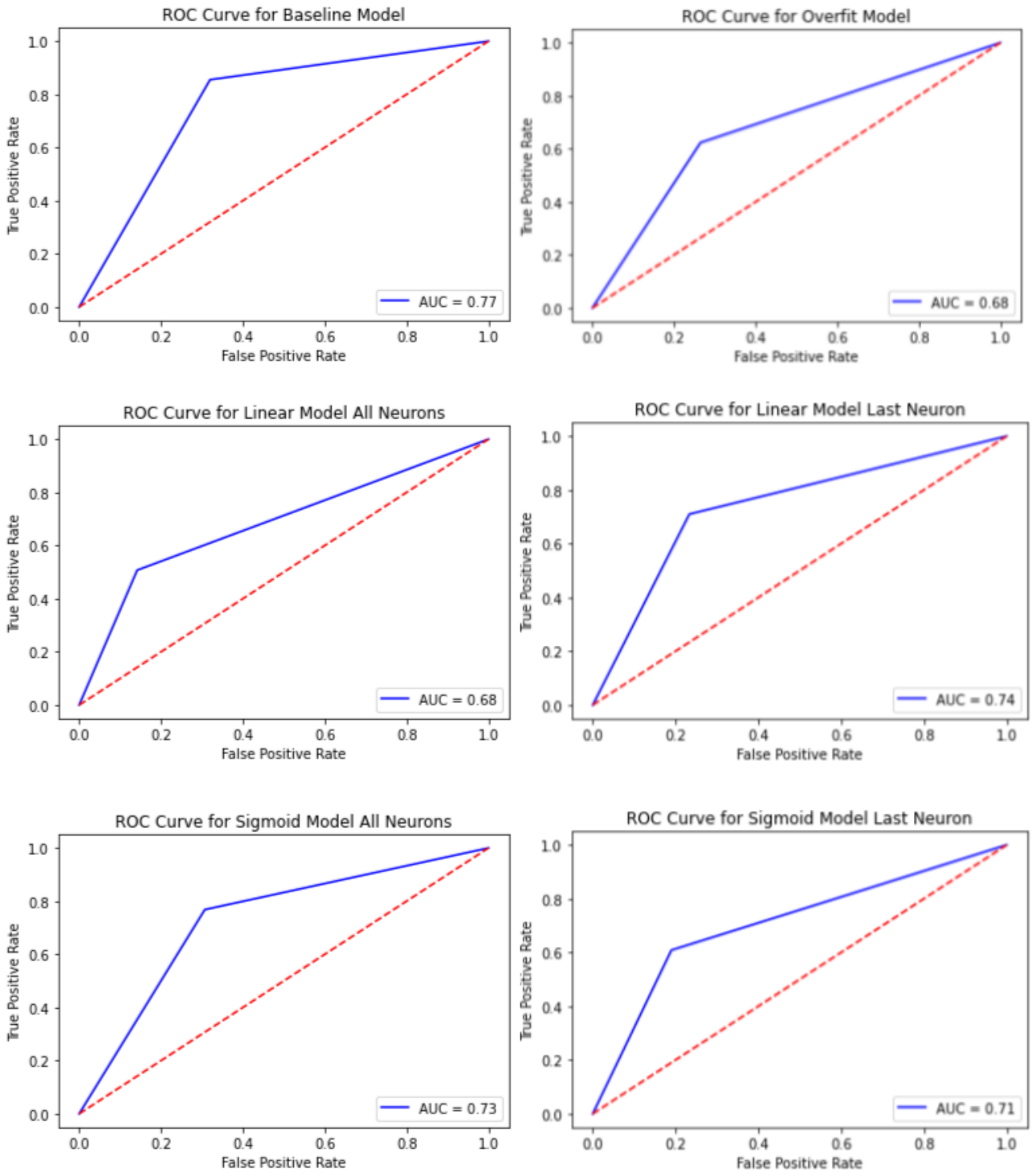

Figure 16: All Model ROC Curves

7 Future Improvements

There are few ways that this data could become more concrete. More data is required. Not only a larger data set, but more instances of the stroke classifier. Using the over sampling SMOTE technique was effective at creating more instances, but the accuracy was greatly decreased.

On top of that, more feature parameters could be useful. Even potentially specifying what type of stroke the patient suffered from. There are five main types of strokes ischemic stroke, hemorrhagic stroke, transient ischemic attack, cryptogenic stroke, and brain stem stroke [3]. All of which can be specified in a more in depth data set.

Artificially expanding the entire data set could also be a potential solution to improve model accuracy. Although, artificially expanding the data does not necessarily mean that the accuracy would improve. In the case of using SMOTE to over sample the stroke data, it made the accuracy worse.

8 Conclusion

Predicting strokes is a difficult task. It is an ongoing problem within the medical community. It is also disappointing to see the accuracy's of the models so low. In the learning curves it is obvious that the validation accuracy's is fluctuating. This could be because of the lack of data points, complexity of the predictions, hyper-parameters, or even the early-stopping. Even when the early-stopping was stopped earlier the accuracy was even more decreased. Neural network models are also hard to fine tune. Models with more complexity seemed to have greater accuracy, but the validation accuracy fluctuated.

Overall, this model can be used to determine how at risk patients are for stroke. There is still more fine tuning that could be done to increase the accuracy. However, using a bigger and more stroke specific data set could prove to be more effective at predicting strokes.

References

[1] Brownlee, J. (2020, August 27). Recursive feature elimination (RFE) for feature selection in Python. Machine Learning Mastery. Retrieved December 11, 2021, from https://machinelearningmastery.com/rfe-feature-selection-in-python/.

[2] Bownlee, J. (2021, March 16). Smote for imbalanced classification with python. Machine Learning Mastery. Retrieved December 11, 2021, from https://machinelearningmastery.com/smote-oversampling-for-imbalanced-classification/.

[3] Centers for Disease Control and Prevention. (2021, August 2). Stroke Types. Centers for Disease Control and Prevention. Retrieved December 11, 2021, from https://www.cdc.gov/stroke/typesofstroke.htm.

[4] Centers for Disease Control and Prevention. (2021, May 25). Stroke facts. Centers for Disease Control and Prevention. Retrieved December 11, 2021, from https://www.cdc.gov/stroke/facts.htm.

[5] Narkhede, S. (2021, June 15). Understanding AUC - roc curve. Medium. Retrieved December 11, 2021, from https://towardsdatascience.com/understanding-auc-roc-curve-68b2303cc9c5.

[6] Smote: Overcoming class imbalance problem using smote. Analytics Vidhya. (2021, January 6). Retrieved December 11, 2021, from https://www.analyticsvidhya.com/blog/2020/10/overcoming-class-imbalance-using-smote-techniques/.