Detection of Heart Disease

Kanaan Sullivan
Computer Science 5300

April 24th, 2024

Contents

Abstract

This report covers whether or not it is possible to accurately predict the chance of heart disease within a population over their next ten years. Using a dataset from Keras based on a medical study from Farmington, Massachusetts and is composed of fifteen different fields to use in machine learning modeling. The data will be cleaned, converting all rogue data into numerical values. Then the data will be normalized so that all of it matches onto a single axis more reliably. There will be comparison through multiple models such as Binary Classification, a Logistic Regression mode and several different Neural Network configurations. These models will be ran against one another for checks in accuracy and loss. Afterwords the winning model will have its features re-evaluated for necessity in the model before removing the ones that are low enough to warrant removal. The final model shown will be one that has a new dataset with those unnecessary input features removed. After this the model will be evaluated against the original model it was selected from to see if there was any improvement over the base model.

1 Introduction

Dataset and Problem: The dataset used for this report is the ”Logistic Regression To Predict Heart Disease” Dataset found on Kaggle. The problem attempting to be solved is with the inputs given in this dataset can we create a Neural Network to correctly predict heart disease within the next ten (10) years.
Motivations: As someone who has one and a family with a history of heart disease this problem is fairly close to the heart. As such it feels right to select this dataset keeping in mind personal struggles with the subject.

2 Dataset

The ”Logistic Regression To Predict Heart Disease” is sourced from Kaggle Data Science[1]. It is a set of data from an ongoing cardiovascular study being performed on the people of Farmingham, Massachusetts. The base dataset contains 15 input fields and over 4000 records. The output field is binary in being a 0 or 1 as for the chance if the record has a chance of heart disease over the next ten years.

2.1 Input Fields

Below are the input fields of the dataset and a short definition.

1. Sex: Is the patient male or female (Binary, 0 for Female, 1 for Male).
2. Age: How old is the patient (in years), continuous.
3. Education: How much education the patient has received (Rated 1-4 for years).
4. currentSmoker: Is the patient a smoker (Binary, 0 for non-smoker, 1 for smoker).
5. cigsPerDay: If the patient is a smoker, how many cigarettes do they smoke per day (Flat number of how many cigarettes are smoked).
6. BPMeds: Is the patient on blood pressure medication (Binary, 0 for not on medication, 1 for being on medication).
7. prevalentStroke: Does the patient have a history of a stroke (Binary, 0 for no history, 1 for past stroke history).
8. prevalentHyp: Does the patient have a history of being hypertensive (Binary, 0 for no history, 1 for past hypertensive history).
9. diabetes: Is the patient diabetic (Binary, 0 for non-diabetic, 1 for diabetic).
10. totalChol: Total cholesterol level of the patient.
11. sysBP: The systolic blood pressure of the patient.
12. diaBP: The diastolic blood pressure of the patient.
13. BMI: The Body Mass Index (BMI) of the patient.

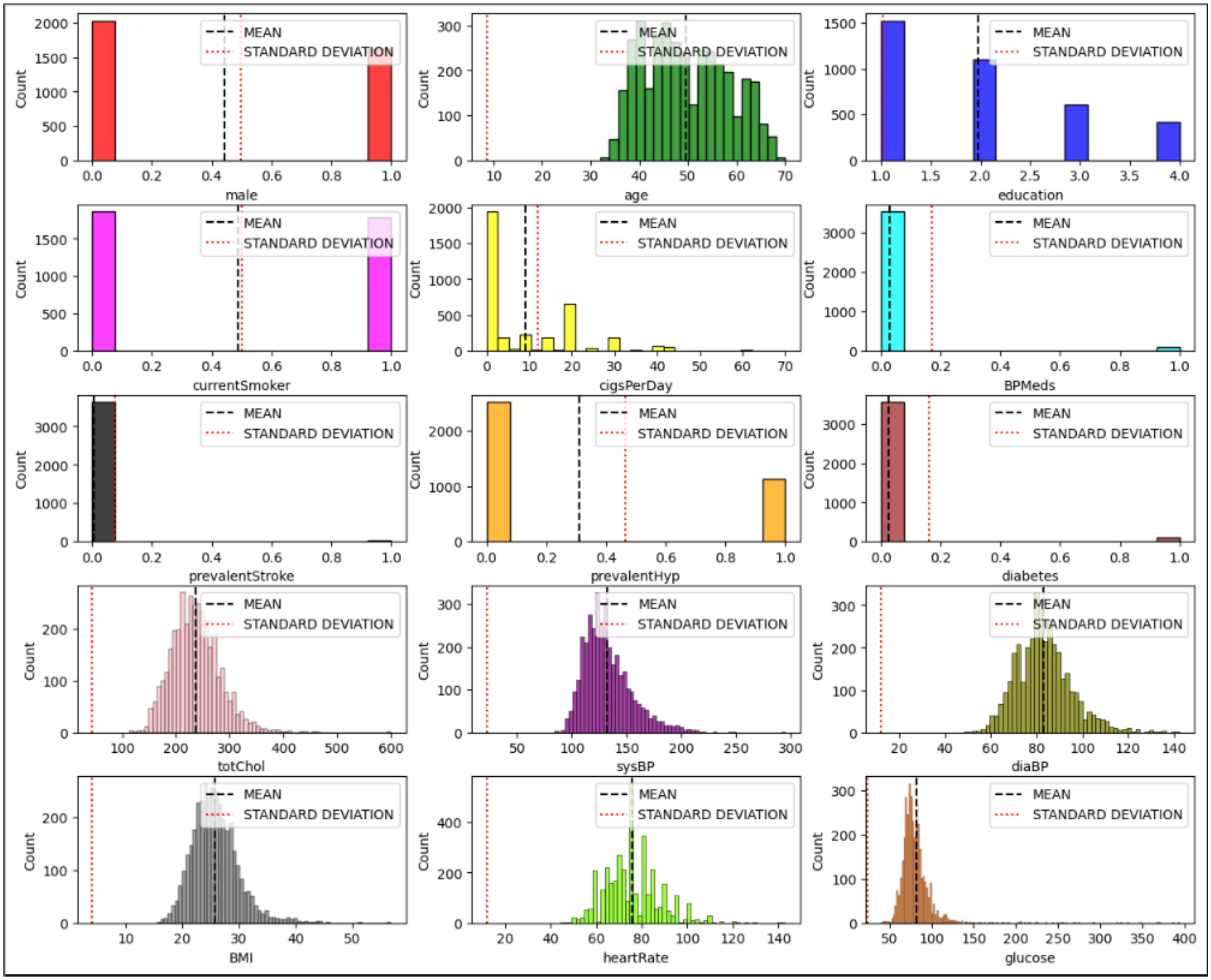
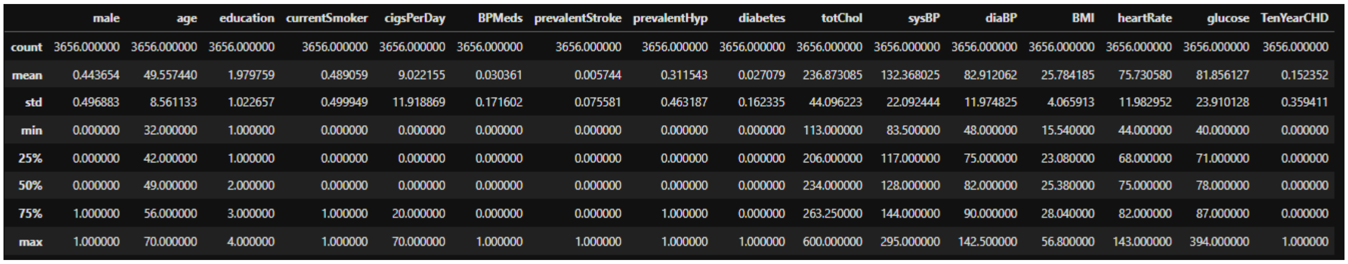


Figure 1: Input Data Histograms

14. heartRate: The heart rate of the patient, continuous.

15. glucose: The glucose level of the patient, continuous.

2.2 Visualization of Input Data Distribution



	male	age	education	currentSmoker	cigsPerDay	BPMeds	prevalentStroke	prevalentHyp	diabetes	totChol	sysBP	diaBP	BMI	heartRate	glucose	TenYearCHD
count	3656.000000	3656.000000	3656.000000	3656.000000	3656.000000	3656.000000	3656.000000	3656.000000	3656.000000	3656.000000	3656.000000	3656.000000	3656.000000	3656.000000	3656.000000	3656.000000
mean	0.443654	49.557440	1.979759	0.489059	9.022155	0.030361	0.005744	0.311543	0.027079	236.873085	132.368025	82.912062	25.784185	75.730580	81.856127	0.152352
std	0.496883	8.561133	1.022657	0.499949	11.918869	0.171602	0.075581	0.463187	0.162335	44.096223	22.092444	11.974825	4.065913	11.982952	23.910128	0.359411
min	0.000000	32.000000	1.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	113.000000	83.500000	48.000000	15.540000	44.000000	40.000000	0.000000
25%	0.000000	42.000000	1.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	206.000000	117.000000	75.000000	23.080000	68.000000	71.000000	0.000000
50%	0.000000	49.000000	2.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	234.000000	128.000000	82.000000	25.380000	75.000000	78.000000	0.000000
75%	1.000000	56.000000	3.000000	1.000000	20.000000	0.000000	0.000000	1.000000	0.000000	263.250000	144.000000	90.000000	28.040000	82.000000	87.000000	0.000000
max	1.000000	70.000000	4.000000	1.000000	70.000000	1.000000	1.000000	1.000000	1.000000	600.000000	295.000000	142.500000	56.800000	143.000000	394.000000	1.000000

Figure 2: Input Features Statistics

2.3 Distribution of Output Data

The output data included in the dataset is a binary class with 0 being for no detected Chance of Heart Disease and a 1 for if this chance was detected within the next 10 years (10YearCHD).

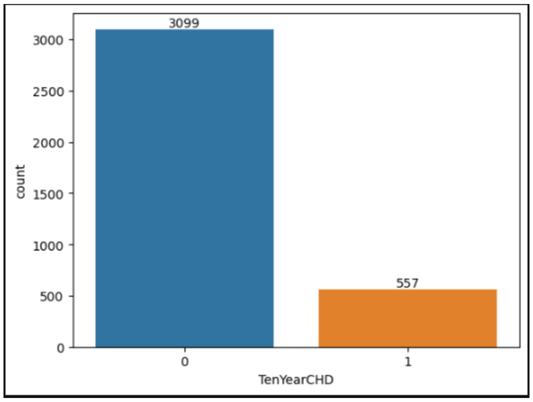


Figure 3: Output data statistics. Showing a 85% chance of no heart disease and a 15% of Heart Disease

3 Data Processing

3.1 Data Normalization

Data Normalization allows a tighter set of constraints to be me made on data. It removes the impact of scale and puts all input fields into the same scale. This "Normalizes" the data and allows for faster processing due to the smaller scalar. After normalization all values of the dataset are between 0 and 1. The process used for this normalization was the Min Normalization function. Which can be defined from the equation below.

$$Xnormal = \frac{X - Xmin}{Xmax - Xmin}$$

3.2 Visualization of Normalized Data

Below are the visualizations of the normalized input fields

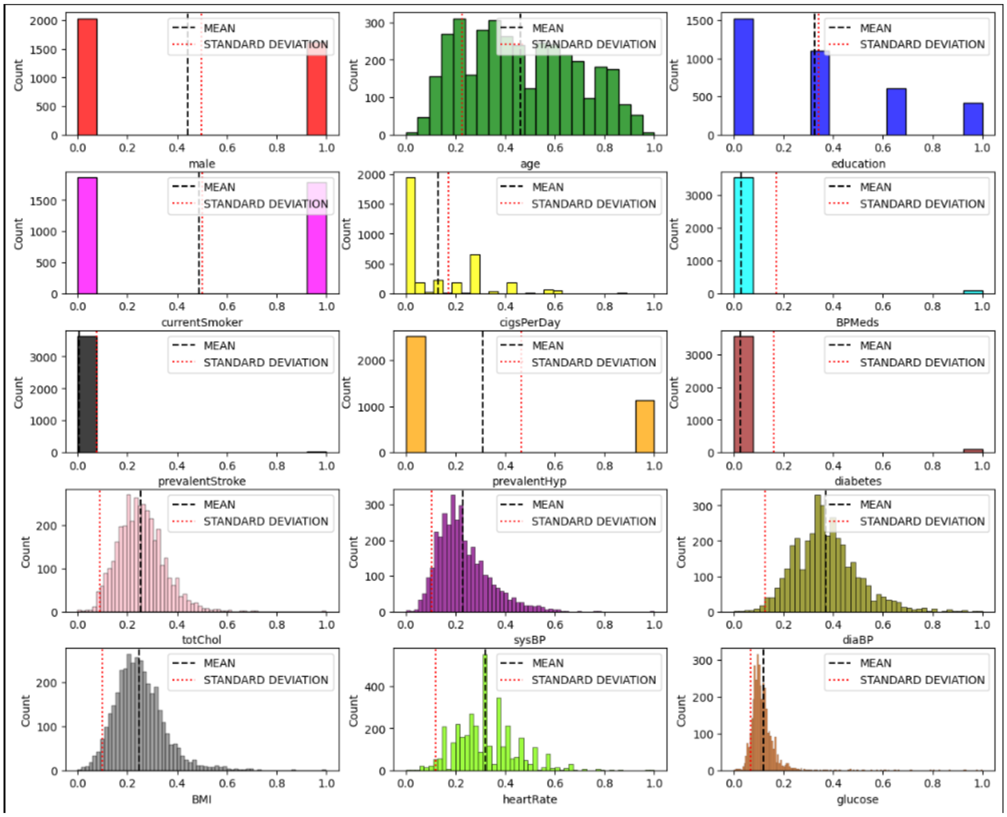
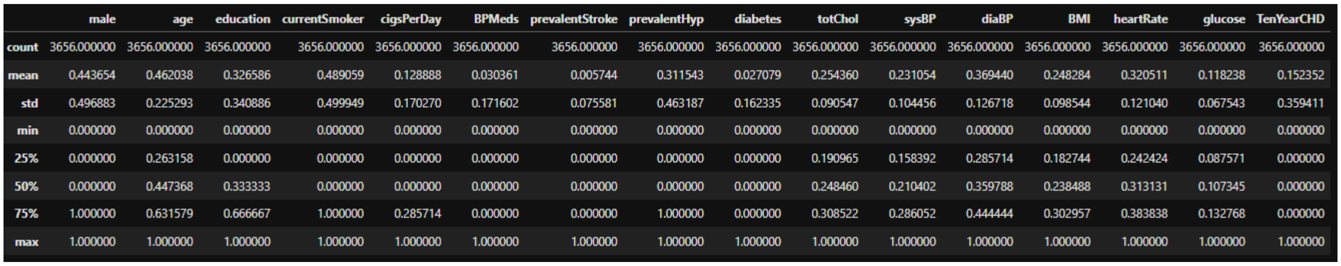


Figure 4: Normalized Input Field Histograms



	male	age	education	currentSmoker	cigsPerDay	BPMeds	prevalentStroke	prevalentHyp	diabetes	totChol	sysBP	diaBP	BMI	heartRate	glucose	TenYearCHD
count	3656.000000	3656.000000	3656.000000	3656.000000	3656.000000	3656.000000	3656.000000	3656.000000	3656.000000	3656.000000	3656.000000	3656.000000	3656.000000	3656.000000	3656.000000	3656.000000
mean	0.443654	0.462038	0.326586	0.489059	0.128888	0.030361	0.005744	0.311543	0.027079	0.254360	0.231054	0.369440	0.248284	0.320511	0.118238	0.152352
std	0.496883	0.225293	0.340886	0.499949	0.170270	0.171602	0.075581	0.463187	0.162335	0.090547	0.104456	0.126718	0.098544	0.121040	0.067543	0.359411
min	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
25%	0.000000	0.263158	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.190965	0.158392	0.285714	0.182744	0.242424	0.087571	0.000000
50%	0.000000	0.447368	0.333333	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.248460	0.210402	0.359788	0.238488	0.313131	0.107345	0.000000
75%	1.000000	0.631579	0.666667	1.000000	0.285714	0.000000	0.000000	1.000000	0.000000	0.308522	0.286052	0.444444	0.302957	0.383838	0.132768	0.000000
max	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000

Figure 5: Normalized Input Features Statistics

4 Modelling

The problem attempting to be solved, the prediction of whether a patient will need to worry about heart disease within the next ten years is a Binary Classification Problem. To fit this data a Logistic Regression model works best for initial testing.

4.1 Overfitting Initial Logistic Regression Model

The first step was to over-fit the model using the entire dataset to properly hone in on what would be the correct size for the Neural Network later on in the next phase. Initial over-fitting was not successful getting stuck usually around 82-87% accuracy. Over-fitting would not be satisfied until near 100% accuracy was reach. Nodes were added to the model in increments of two initially ramping up to five when the former did not change much. Eventually the over-fitting stabilized with 6 layers with 40 initial inputs around an average accuracy of 99% with ending statistics of
Accuracy: 99.97%
Precision: 99.82%
Recall: 100.00%
F1-score: 1.00

4.2 Model Selection and Evaluation

All models shown were set to run for 400 Epochs with Early Stopping and Model Check-pointing
Baseline Model: A baseline binary classification Neural Network model with a single layer and single input/output.
Logistic Regression Model: A logistic regression model of the same number of layers and input/outputs as the Baseline model.
Neural Network NN_HighModel: A Neural Network consisting of five layers with the inputs of 32-16-8-2-1
Neural Network NN_MedModel: A Neural Network consisting of four layers with the inputs of 16-8-2-1
Neural Network NN_Low: A Neural Network consisting of 3 layers with the inputs of 8-2-1
Neural Network NN_Min: A Neural Network consisting of 2 layers with the inputs of 2-1

Model Name	Accuracy (T)	Accuracy (V)	Loss (T)	Loss (V)
Baseline	86.52%	82.82%	33.14%	43.32%
Logistic Regression	84.78%	85.44%	38.64%	35.53%
Neural Network (32-16-8-2-1)	96.62%	76.49%	7.86%	218.40%
Neural Network (16-8-2-1)	91.28%	81.03%	28.60%	95.25%
Neural Network (8-2-1)	85.53%	83.97%	34.98%	41.38%
Neural Network (2-1)	84.64%	82.99%	36.19%	39.67%

Table 1: Each model made and their Training Accuracy, Validation Accuracy, Training Loss, and Validation Loss

As shown here the Logistic Regression Model performed the overall best. With the highest Validation Accuracy and lowest Validation Loss. The reason for the accuracy decrease and increasing loss of the models over time is due to Over-fitting occurring mid way despite check pointing and early stopping Below are the learning curves for each model.

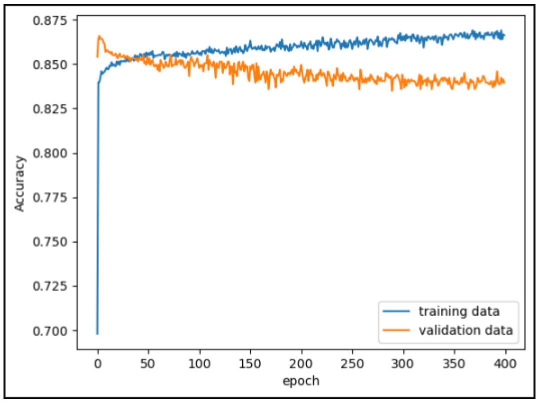


Figure 6: Baseline Model Learning Curve

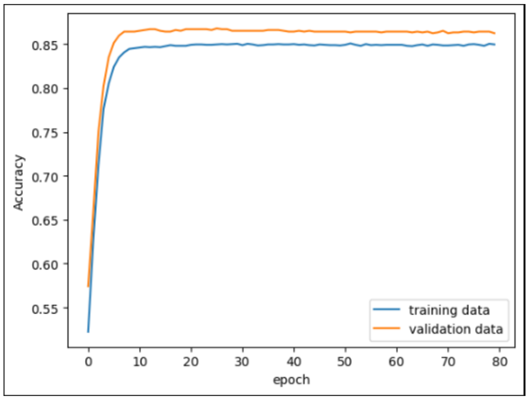


Figure 7: Logistic Regression Learning Curve

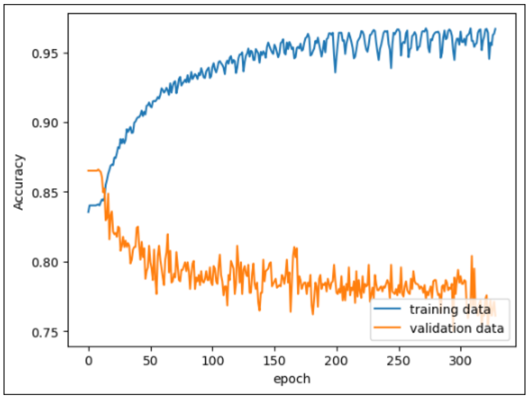


Figure 8: Neural Network (32-16-8-2-1) Learning Curve

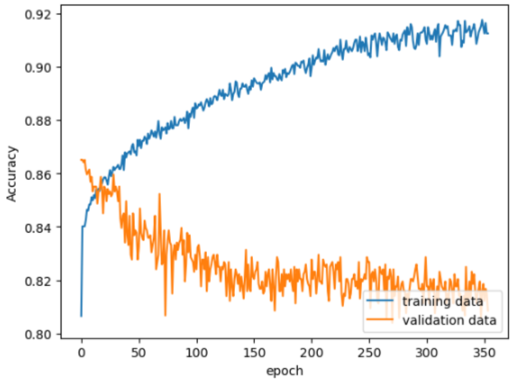


Figure 9: Neural Network (16-8-2-1) Learning Curve

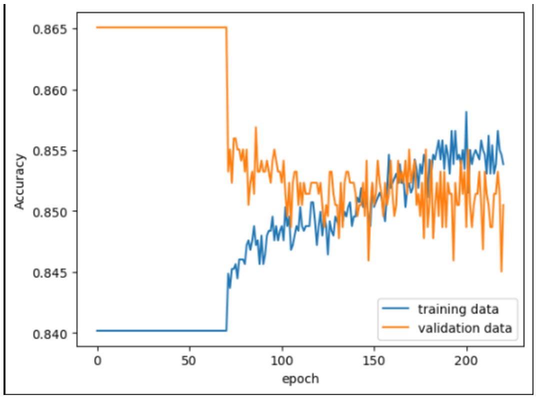


Figure 10: Neural Network (8-2-1) Learning Curve

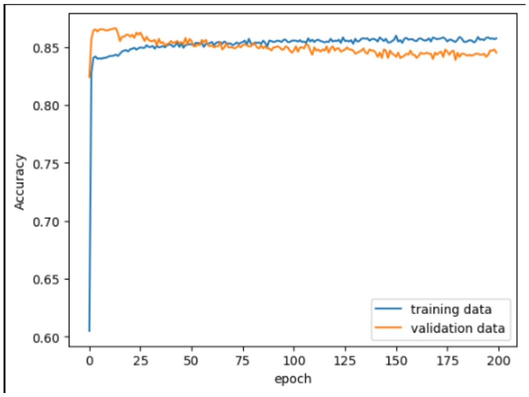


Figure 11: Neural Network (2-1) Learning Curve

4.3 Feature Reduction

With the Logistic Regression model now selected it was now time for the reduction of input fields to the model itself. This was done by taking the Logistic Regression model and feeding the inputs one at a time to it to gauge which features are most important to the model. Afterwards the fields were taken out one at a time and compared for their accuracy to one another. A graph showing feature accuracy can be found below. The least important features were found to be: education, age and gender. In the final model these will be removed.

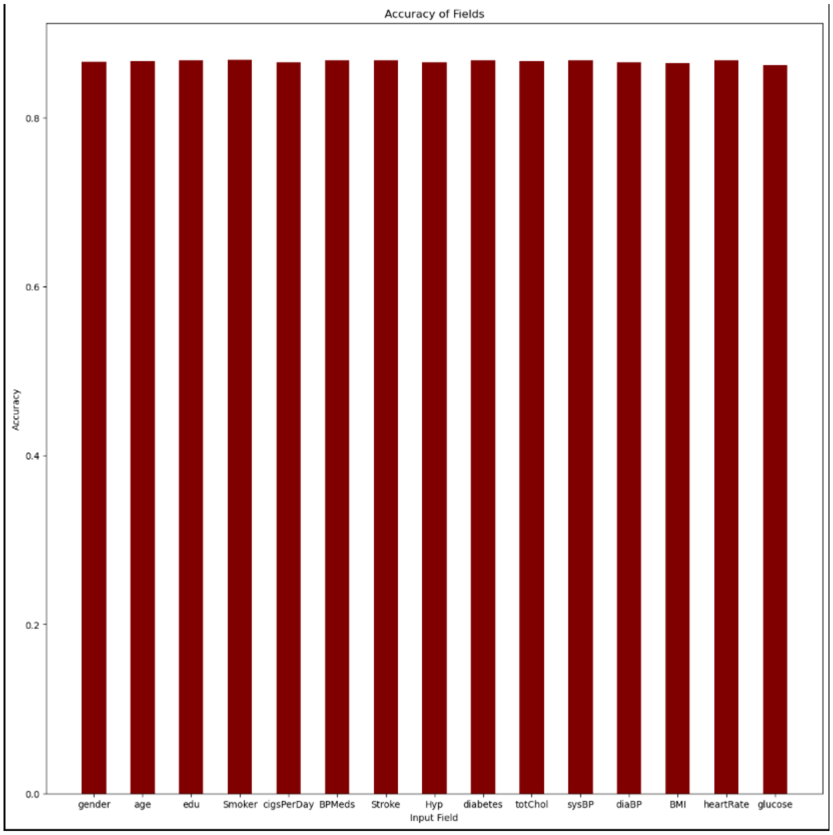


Figure 12: Feature Accuracy

4.4 Final Model

The final model is a logistic regression model with the removal of the education, age and gender input fields. The result was an improvement from the original Logistic Regression model. There was an improvement on the validated accuracy of the original model by 1.6% and a lessening in loss by 0.22%.

Model Name	Accuracy (T)	Accuracy (V)	Loss (T)	Loss (V)
Logistic Regression	84.78%	85.44%	38.64%	35.53%
Logistic Regression Final	84.24%	87.04%	41.62%	35.31%

Table 2: The original Logistic Regression and the Final Model with their Training Accuracy, Validation Accuracy, Training Loss, and Validation Loss

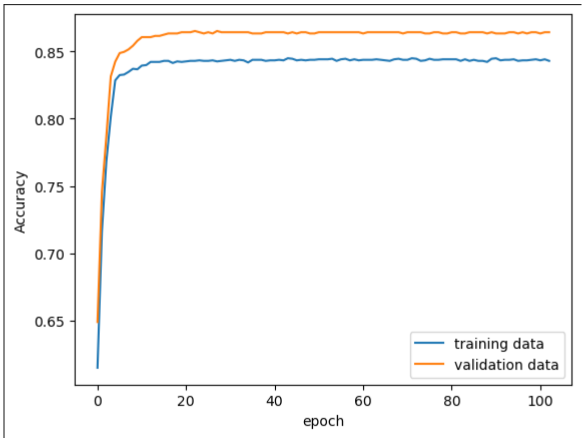


Figure 13: Final Model Learning Curve

5 Conclusion

From the data shown throughout the phases it can be shown that the the chance of heart disease within the next ten years can be accurately predicted off of the data given. It is likely that the original dataset would have also worked fine for the model but the exclusion of gender, age, and education still improved upon the accuracy of the end product. This indicates that some data in those nodes leads to false info or over-fitting that are seen in the other models. These models had vast over-fitting problems coming from the multiple layers used in them. As shown in their figures the training data would vary wildly from the validation data. It is possible running these models with adjusted datasets used for the logistic regression model could lead to better results with additional epochs. For now however the model shown here is satisfactory for the purpose given by the dataset, additional testing is required to see if there is a way to continue the improvement of the model's accuracy

References

[1] Dileep & Naveen. Logistic regression to predict heart disease. https://www.kaggle.com/datasets/dileep070/heart-disease-prediction-using-logistic-regression.