Predicting the Presence of Diabetes with Binary Classification

Author: Jesse Gempel

University of Missouri-St. Louis
College of Arts and Sciences

Introduction to Artificial Intelligence

December 4, 2024

Preface

My Google Colab notebook for this report can be accessed through this link:
https://colab.research.google.com/drive/1MFd4Ofi9-pZQvhDDgkfJCtJJtsKoLXL8?usp=sharing.

Abstract

Diabetes is a common disease that affects roughly 12 percent of the American population[1]. It occurs in two different ways: Type I diabetes attacks the immune system, preventing the body from producing insulin; type II diabetes causes the body to intercept any amount of insulin, each though it may produce enough of it[2]. In either case, the human body's inability to utilize insulin is caused by one's excessive consumption of glucose. It can be easy for an individual to consume too many sugary foods and beverages. This issue is particularly noteworthy for Americans because a concerning amount of food relies on high fructose corn syrup and excess sugar. A simple task that we do everyday can cause a person to end up like the 1.6 million people who passed away from diabetes in 2021[3]. Many factors, such as exercise, drug use, and cholesterol can help infer the presence of diabetes. For this reason, an artificial intelligence project based on predictive neural networks can help one survey the correlation and importance of these factors.

The intent of this project is to predict the presence of diabetes among hospital patients. The process of making these predictions can be defined as a binary classification problem, with a **0** for *non-diabetic* or a **1** for *diabetic/pre-diabetic*. Real-life patient data will be analyzed with logistic regression-based neural networks to make such predictions.

Seven different models will be created with unique neural network architectures. These models will then be analyzed by observing the several different metrics, such as accuracy, F1-score, precision, ROC score, and AUC score. These analyses will be used to determine which of the seven models performs the most optimally. Once the ideal model is discovered, each individual feature will be trained on the particular model, and features that hinder its performance will be incrementally removed from training. These steps will be applied to determine a patient's presence (or non-presence) of diabetes.

1 Data Exploration

The "Diabetes Health Indicators Dataset" was gathered from a data collection website called Kaggle[4]. It was also archived from the UC Irvine Machine Learning Repository[5]. The dataset contains 70,692 samples: exactly 50.0% of the dataset's samples are classified as *non-diabetic*, while the remaining 50.0% are labelled as *diabetic/pre-diabetic*.

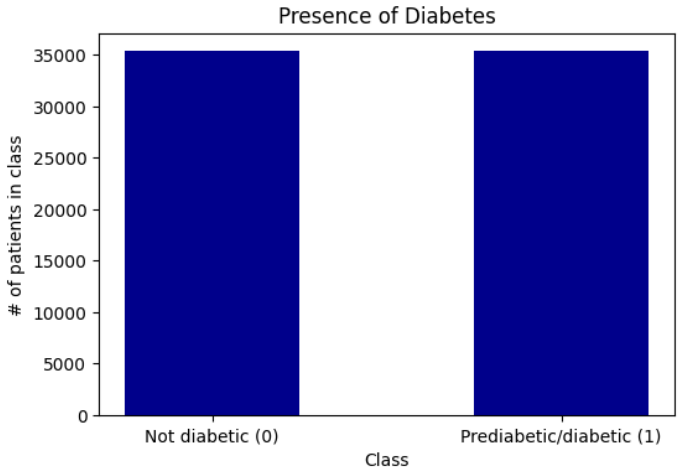


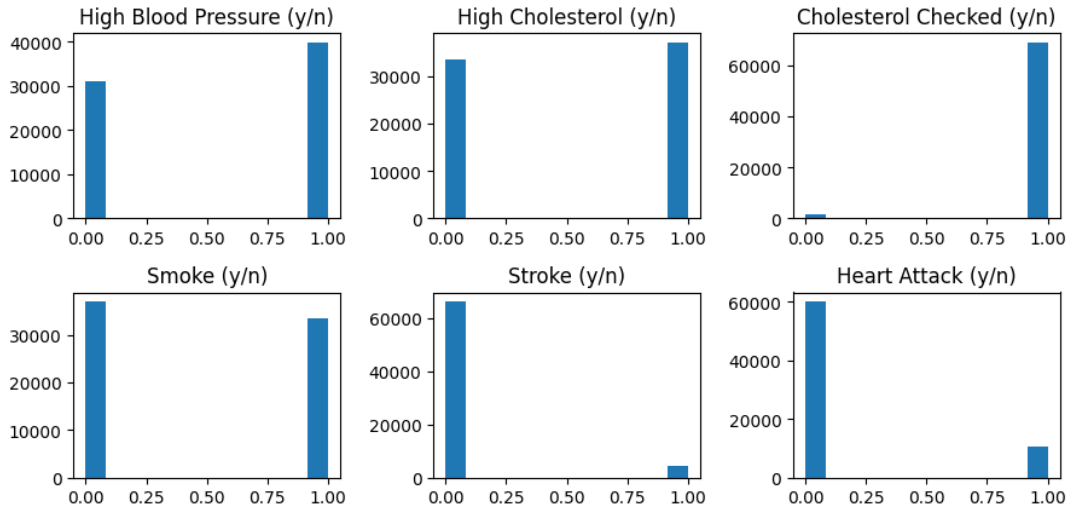
Figure 1: A breakdown of the quantity of features in each class. The **Not diabetic (0)** class contains 35,346 samples, while the **Prediabetic/diabetic (1)** class contains 35,346 samples as well.

Within each sample is a series of 21 features and 1 target variable. Each feature represents a piece of information gathered from a patient by a medical professional at a hospital. The target variable represents the presence of diabetes; it is the job of neural network models to correctly predict whether a patient has diabetes based on the information given from the 21 features.

	Attribute	Description
1	**Diabetes_binary**	Does patient have diabetes? (**TARGET VARIABLE**)
2	HighBP	Is patient's **blood pressure** high?
3	HighChol	Is patient's **cholesterol** high?
4	CholCheck	Did patient get a **cholesterol check** in the past 5 years?
5	BMI	What is the patient's **body mass index**?
6	Smoker	Does patient **smoke**?
7	Stroke	Did patient have a **stroke**?
8	HeartDiseaseorAttack	Did patient ever have a **heart attack**?
9	PhysActivity	Did patient **exercise** in the last 30 days?
10	Fruits	Does patient regularly eat **fruit**?
11	Veggies	Does patient regularly eat **vegetables**?
12	HvyAlcoholConsump	Does patient drink a lot of **alcohol**?
13	AnyHealthcare	Does patient have **health coverage?**
14	NoDocbcCost	Did patient need to **see a doctor** but couldn't?
15	GenHlth	How is patient's **overall health**? (1 to 5)
16	MentHlth	How many days (0-30) did patient have poor **mental health**?
17	PhysHlth	How many days (0-30) did patient have poor **physical health**?
18	DiffWalk	Does patient struggle with **climbing up stairs**?
19	Sex	What is patient's **sex**?
20	Age	What is patient's **age**?
21	Education	What is patient's **education**?
22	Income	What is patient's **income**?

Figure 2: A complete list of features from the "Diabetes Health Indicators Dataset," each with descriptions[6]. **Diabetes_binary** serves as the target variable that a model needs to predict.

Out of the 21 features, 14 of them contain binary data, while 7 of them contain continuous data. The features' distributions are visualized in Figure 3 and Figure 4 below:



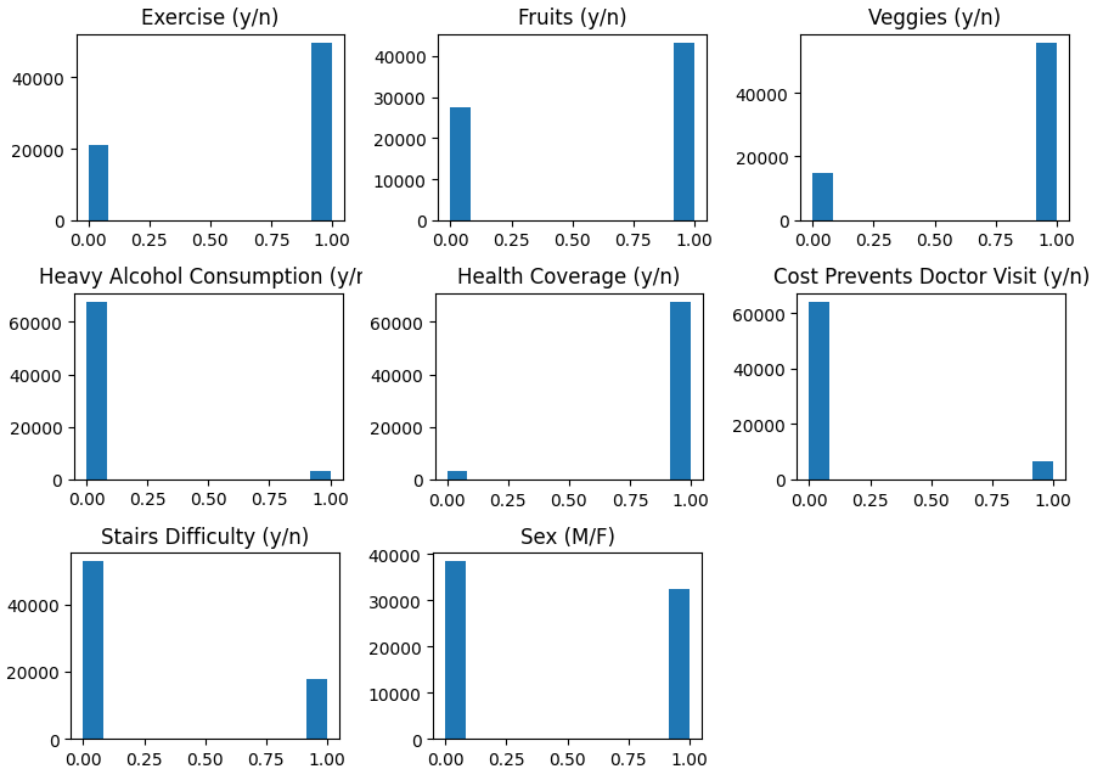
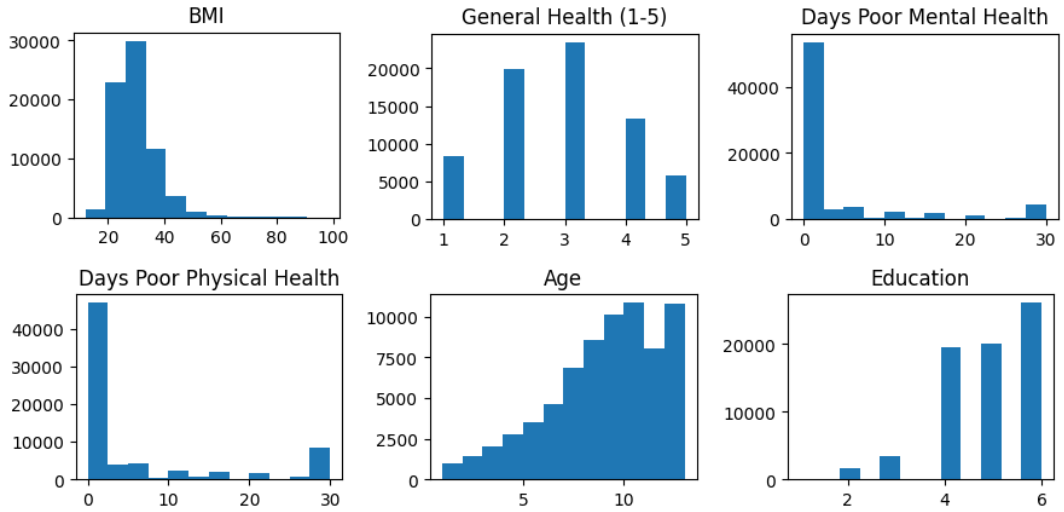


Figure 3: Histograms visualizing the distribution of the 14 features from the "Diabetes Health Indicators Dataset." These particular features are *binary*, meaning that **0** resembles *no* and **1** resembles *yes*. For the **Sex (M/F)** feature, 0 represents "female" while 1 represents "male." The features' names were changed slightly from Figure 2's names to provide a cleaner data visualization.



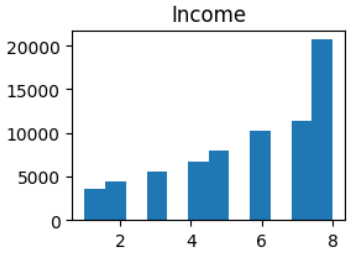


Figure 4: Histograms visualizing the distribution of 7 more features from the "Diabetes Health Indicators Dataset." Unlike the 14 features in Figure 3, these histograms consist of more than two values.

The last three features from Figure 4–Age, Education, and Income–contain some seemingly nonsensical values. However, the values of Age (1 through 13), Education (1 through 6), and Income (1 through 8) are categorical. The specific categorizations of these feature quantities can be observed by navigating to these websites: *ResearchGate*[7] for the **Age** categorization, the *National Addiction & HIV Archive Program*[8] website for the **Education** classification, and the *Resource Center for Minority Data*[9] website for the **Income** categorization. These websites are located in the References section under Numbers 7, 8, and 9.

Data Normalization

The 21 features underwent a normalization calculation to facilitate the performance for all potential models. The target variable **Diabetes_binary** was not included in the calculation due to its variables already containing binary values. The calculation for each feature's value was conducted by using a *z-score* calculation, which is represented like this:

$$z - score = \frac{x - \mu}{\sigma}$$

where x is a feature's current value inside a row, μ is the standard mean of the feature, and σ is the feature's standard deviation. The *z-score* serves as the result that replaces the raw value inside the feature's instance. This calculation is performed iteratively for every value inside of each feature. This process reduces the minimum-maximum range for each of the 21 features, which will tentatively allow each model to perform more quickly with more robust metrics.

2 Data Separation into Training and Validation Sets

Separation of data served as the next phase of creating a model that can predict the presence of diabetes. The "Diabetes Health Indicators Dataset" was divided into two additional sets for training and validation. Separating the dataset was essential towards ensuring the efficiency and accuracy of finding the ideal classification model.

The number of rows (or samples) in the training and validation sets are smaller than in the original set. This scenario occurs due to **70%** of the original data's placement into the *training set*, and the remaining **30%** into the *validation set*. That is why 49,485 rows of data are used for training and 21,207 rows are used for validation purposes. This separation process is illustrated graphically in Figure 5.

```
X_training shape:       (21207, 21)
Y_training shape:       (21207,)
X_validation shape:     (49485, 21)
Y_validation shape:     (49485,)

X shape:                (70692, 21)
Y shape:                (70692,)
```

Figure 5: A depiction of the shapes of the training and validation sets for the input features X and the output target Y. Each coordinate represents the number of rows and columns in a separated dataset respectively. NOTE: the target Y contains missing second values, meaning that only one column exists for each target variable.

3 Random Baseline Classifier

The random baseline classifier is a concept that involves comparing the effectiveness of neural network models. These models are compared to benchmark values that dictate the minimum performance that a model should achieve to be useful. A minimum accuracy metric can serve as an example of such a benchmark value. In this particular study, the random baseline classifier relies on the percent distribution of samples in each class. Since neural networks in this study attempt to solve a binary classification problem, the baseline relates to the percentage of samples in the *larger* class. For instance, if 70% of the data samples belong to the larger binary class, then the random baseline classifier accuracy would be 70%.

As mentioned in the Data Exploration section, the two classes–Diabetic/Pre-diabetic and Non-diabetic–both contain exactly 50.0% of the dataset's samples. Since no class is larger than the other class, it is safe to say that the **random baseline classifier** metric will be **50.0%**.

4 Model Selection and Evaluation

To select the ideal model for binary classification, *seven* different models with *five* unique layer counts will be analyzed. The architectures will be applied as follows:

1. A **single-layer** network with one neuron. This network will serve as a *simple logistic regression model* that will be used as a "baseline."
2. A **5-layer** network (with a 128-64-32-16-1 architecture).
3. Two **4-layer** networks (with 64-32-16-1 and 16-8-4-1 architectures).
4. A **3-layer** network (with a 8-4-1 architecture).
5. Two **2-layer** networks (with 4-1 and 2-1 architectures).

It is true that two 2-layer models and 4-layer models exist in this study. Since this report will only discuss five of the seven models used, we will select the best 2-layer model and 4-layer model used for this classification problem. This approach will facilitate the selection and analysis of the seven total network models.

All seven models (including the five best ones in this report) will run 200 epochs. For each model, **relu** activation was applied to every layer except the last one, whereas **sigmoid** activation was used for the final layer. Every model also included the two callbacks: ModelCheckpoint and EarlyStopping. Model checkpointing stores the best weights from a model after the loss function was applied for each epoch. The storage of weights depends on the minimization of the validation loss score. If a neural network's loss score does not decrease after 40 epochs, then the model will stop running. This action is due to the setting of EarlyStopping's patience parameter to 40 epochs.

The **metrics** used in this experiment are accuracy, loss, recall, F1-score (which is a combination of precision and recall), ROC (receiver operating characteristic), and AUC (area under the curve).

4.1 Simple Logistic Regression Model

The **simple logistic regression model** contains a single sigmoid layer that only operates with one neuron. Due to the model's simplicity—and therefore, ineffectiveness—it serves as a baseline for accuracy. It acts as a benchmark that facilitates the comparison of a multi-layered neural network's performance. To determine the usefulness of a more complex model, its validation accuracy must be able to surpass the simple logistic regression model's accuracy of **75.107%**.

It is worth noting that this baseline from the simple logistic regression model is not to be confused with the random baseline accuracy from Section 3. The *random baseline accuracy* is set to 50%, whereas *this* model's baseline accuracy is 75.107%. Both baseline accuracies will be used throughout this study.

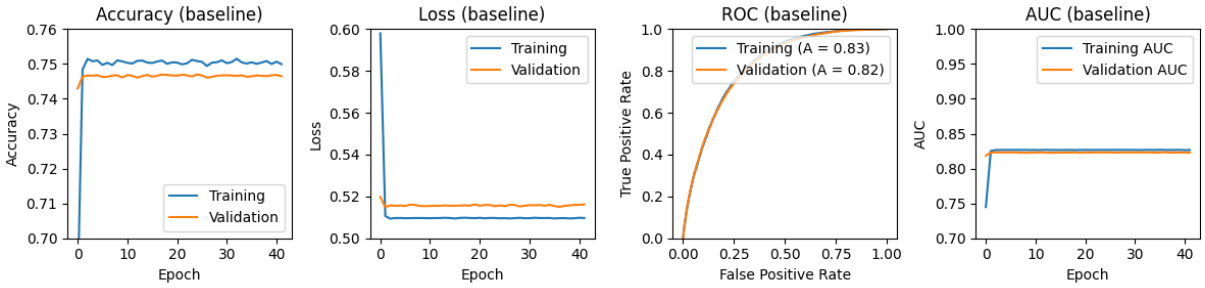


Figure 6: Accuracy, loss, ROC, and AUC measures for a single-layer, single-neuron model. After 42 epochs, EarlyStopping ceased execution of this model.

For the first 3 epochs, the training and validation accuracies and AUC scores skyrocketed, but then remained constant throughout the model's execution. A similar scenario occurred with the loss scores, except they decreased significantly before refusing to change. It is worth noting that the training and testing metrics for accuracy, loss, and AUC remain relatively equidistant throughout most epochs. This single-layered model experienced underfitting as expected due to the model's minuscule size.

4.2 5-layer Neural Network

The 5-layer neural network, which contains 13,697 parameters, deals with an architecture of these neuron counts: 128, 64, 32, 16, and 1. It was the only 5-layer model used for this classification problem.

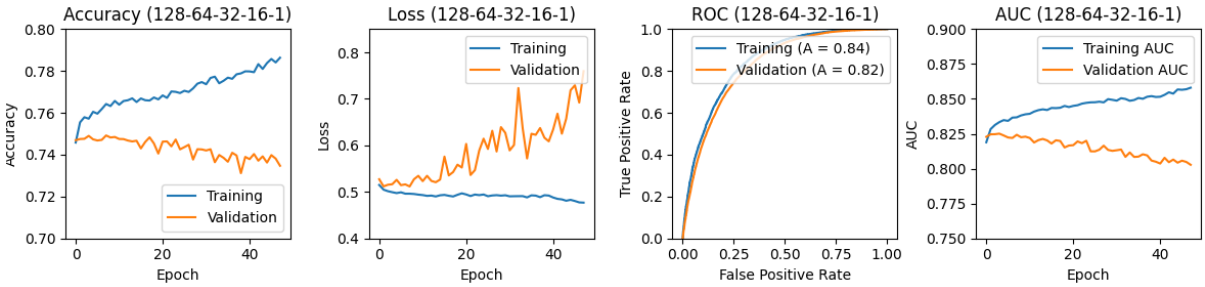


Figure 7: Accuracy, loss, ROC, and AUC measures for a 5-layer model with a 128-64-32-16-1 architecture. EarlyStopping ceased execution of this model once 48 epochs were completed.

Out of all the tested models, this 5-layer network experienced the most severe amount of overfitting. The validation accuracy and AUC (area under the curve) score consistently decreased, whereas the validation loss increased drastically. Also, the training and validation curves for those three metrics diverged quite significantly. Since these observations infer the presence of serious overfitting, we cannot claim this network configuration as ideal.

4.3 4-layer Neural Network

Two 4-layer networks were used for this classification problem: one with a 64-32-16-1 model architecture, and one with a 16-8-4-1 architecture. The latter model experienced slightly better accuracy, loss, ROC, and AUC metrics than the former model. For validation scores, the **accuracy** was 75.029% (versus 74.807%), the **loss** was 0.509 (versus 0.512), and the **ROC** and **AUC** scores were 0.8254 (versus 0.8248). For these reasons, we analyze the network with the **16-8-4-1** configuration. This network utilizes 529 parameters.

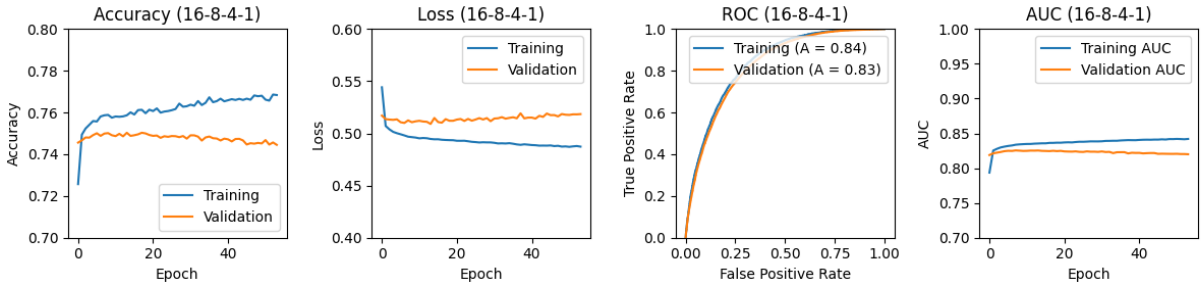


Figure 8: Line chart illustrations of the accuracy, loss, ROC, and AUC metrics from a 4-layer model. This model ran with a 16-8-4-1 architecture for 54 epochs until EarlyStopping prematurely ended its execution.

This 4-layer network behaved similarly to the 5-layer network in Section 4.2. However, there seems to be significantly less overfitting in this model. The slopes of the validation accuracy and AUC curves decrease much more slowly. Similar behavior occurs with the validation loss curve, except that it slowly increases instead. The validation curves also diverge away from the training curves for those three metrics, albeit more gradually. These behaviors dictate the presence of overfitting, so we hope to find a more ideal model later on.

4.4 3-layer Neural Network

The 3-layer neural network, which contains 217 parameters, is characterized with an architecture of these neuron counts: 8, 4, and 1. It was the only 3-layer model used for this machine learning problem.

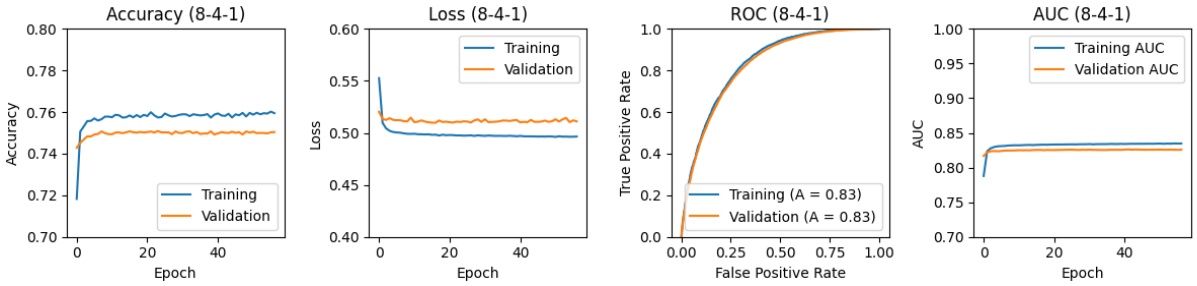


Figure 9: Illustration of a 3-layer model's performance metrics, which include the accuracy, loss, ROC, and AUC scores. This model utilized an 8-4-1 architecture, and it operated for 57 epochs.

This neural network behaves similarly to the simple logistic regression model in Section 4.1. For most of the epochs, the accuracies, losses, and AUC scores for both training and validation data remained constant. Differences between training and validation for those three metrics tended to remain constant as well. This relationship is illustrated by the training and validation curves for accuracy, loss, and AUC remaining equidistant from one another.

The curves' behaviors are concerning because they illustrate underfitting in the 3-layer model. However, the 3-layer model surpasses the baseline model in performance. This discovery applies to the validation metrics of **accuracy** (75.031% versus 74.637%), **loss** (0.510 versus 0.515), **recall** (0.809 versus 0.766), **F1-score** (0.765 versus 0.751) and both **ROC** and **AUC** (0.826 versus 0.823).

4.5 2-layer Neural Network

Two 2-layer networks existed for this machine learning problem. The first model worked with a 4-1 architecture, and the second model dealt with a 2-1 architecture. The neural network with the 4-1 layer configuration resulted in better validation scores of **accuracy** (75.089% versus 74.576%), **loss** (0.510 versus 0.519), **recall** (0.811 versus 0.761) and **F1-score** (0.765 versus 0.750). This network also obtained **ROC** and **AUC** scores both equal to 0.825 (versus 0.821). For these reasons, we select the **4-1** network configuration over the 2-1 configuration. This model contains a total of 93 parameters.

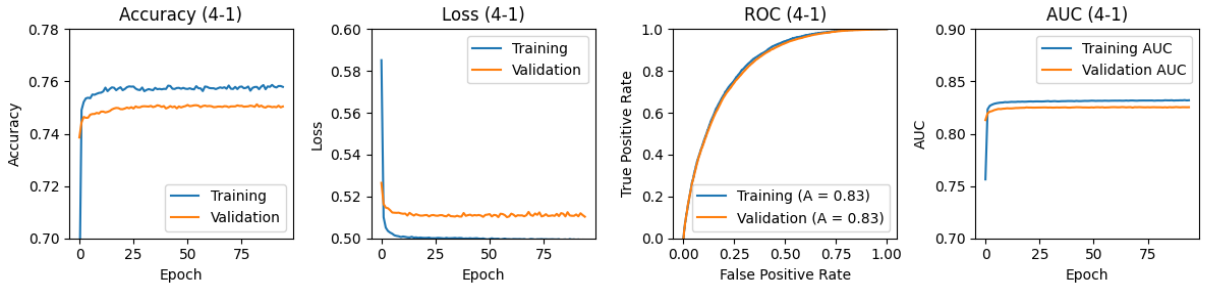


Figure 10: The accuracy, loss, ROC, and AUC performance metrics from the 2-layer model with a configuration of 4-1. This model ran for 95 epochs before EarlyStopping prematurely stopped the its execution.

This neural network behaves similarly to both the baseline model in Section 4.1 and the 3-layer model in Section 4.4. Like all the other models, this neural network began with a sharp increase in accuracy and AUC scores as well as a sharp decline in loss. Implementing the remaining epochs resulted in the accuracy, loss, and AUC scores to remain constant. This model, just like the baseline and 3-layer models, experienced underfitting. This relationship is apparent by the graphs' general lack of slope and the constant distance between the training and validation curves.

5 The Ideal Model

Model Type	Training Accuracy	Validation Accuracy	Training Loss	Validation Loss
Random baseline accuracy	50.000000	50.000000	0.000000	0.000000
Simple logistic regression	75.107276	74.636759	0.508748	0.514906
(128, 64, 32, 16, 1)	76.738813	74.919673	0.484438	0.511778
(64, 32, 16, 1)	75.687273	74.806507	0.492042	0.512016
(16, 8, 4, 1)	76.229547	75.028797	0.491184	0.509038
(8, 4, 1)	75.899467	75.030817	0.496260	0.509715
(4, 1)	75.809874	75.089421	0.498650	0.510120
(2, 1)	75.121422	74.576134	0.512188	0.518767

Figure 11: A tabular representation of the final accuracy and loss scores from all seven models.

After examining all seven neural networks—five of which were discussed in Section 4 of this report—all the models proved to be non-ideal in performance. After few epochs, the validation accuracy, loss, ROC, and AUC scores refused to improve. The underfitting models (i.e., the 2 and 3-layer models) obtained metric scores that remained constant. Meanwhile, the validation metrics of the overfitting models (i.e., the 4 and 5-layer models) would periodically worsen. Every model in this classification

problem rapidly stopped learning how to properly classify the presence of diabetes in hospital patients.

The 5-layer model struggled to avoid overfitting while the 2-layer models with 4-1 and 2-1 architectures could not abstain from underfitting. For these reasons, the neural networks with configurations of 128-64-32-16-1, 4-1, and 2-1 cannot act as an ideal model. The 3-layer 8-4-1 model underfitted the data to a much lesser degree, whereas the 4-layer 16-8-4-1 model overfitted the data *gradually*. Therefore, those two models serve as candidates for becoming labelled as the ideal model.

Upon analysis of the table in Figure 11, it appears that the 8-4-1 neural network model obtains better accuracy and loss scores than the 16-8-4-1 model. The three-layer model obtained a *validation accuracy* of 75.031% (versus 75.029% from the four-layer model). Also, the three-layer model's ROC and AUC scores come out to 0.82550 and 0.82549 (versus 0.82539 and 0.82539 from the 4-layer model). Therefore, we accept the **3 layer model** with a configuration of **8-4-1** as the ideal model for this classification problem.

6 Importance of Features

As shown in Figure 2 of this report, the "Diabetes Health Indicators Dataset" contains 21 features—some of which are more important than others. In this section of the report, we implemented 21 iterations of the classification problem's ideal model—the 3-layer model with an 8-4-1 configuration. Each iteration corresponds to a single feature; that one specific feature serves as the *input* that becomes fed into the model. This step is repeated for each of the dataset's features to produce validation accuracy scores. These scores will then correspond to those individual features. A *high accuracy* indicates that a feature is essential for classifying the presence of diabetes in a hospital patient. A *low accuracy* indicates that the feature is not important, meaning that a neural network could perform better without including it as input.

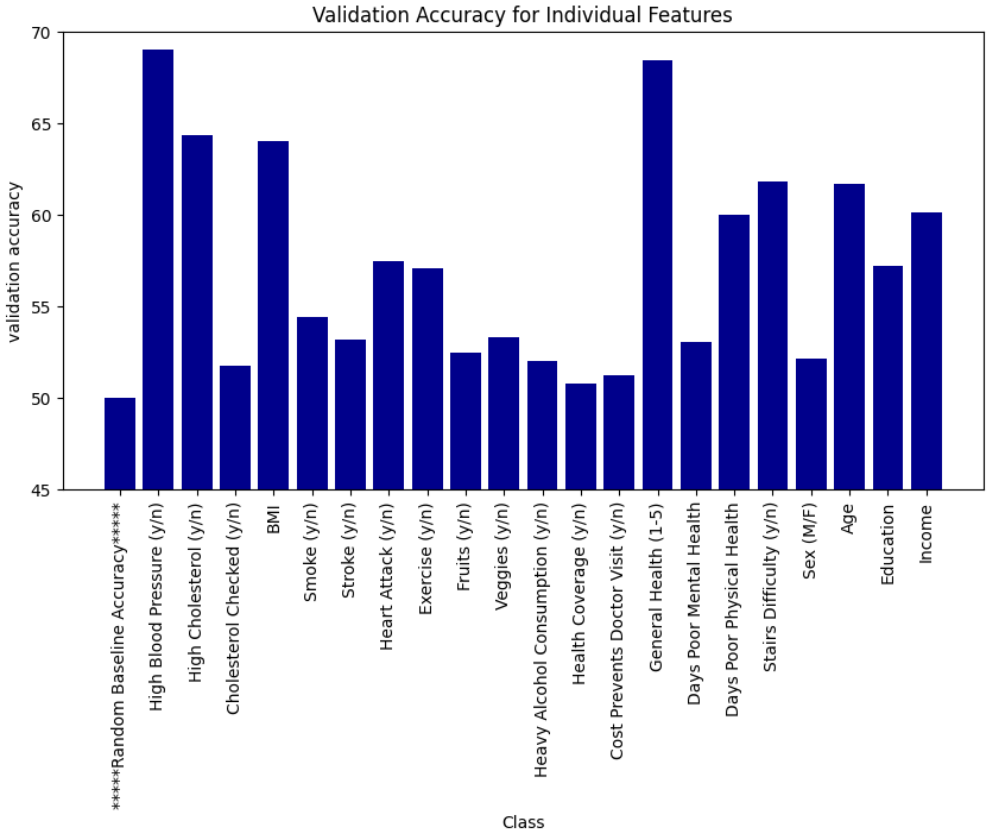


Figure 12: A bar chart visualization illustrating validation accuracies from a model that used *only one* specific feature as input.

After the model repeatedly ran on individual input features, the four highest validation accuracies derived from "High Blood Pressure," "General Health," "High Cholesterol," and "BMI." Those accuracy measures suggest that those four features are most important for classifying diabetes. Diabetes is a disease that is reflective an a patient's overall health and eating habits. People who experience high blood pressure or cholesterol are clearly more at risk of developing diabetes than those who are healthy in those aspects. A high BMI signifies that a person is overweight, which also serves as a correlation to diabetes. Having a poor diet (i.e., one with high sugar, cholesterol, carbs, or fat) and avoiding exercise serve as key factors for the presence of diabetes. Therefore, it makes sense that those four features result in relatively high accuracies.

The three lowest accuracies derived from the features "Health Coverage," "Cost Prevents Doctors Visit," and "Cholesterol Checked." It turns out that in individual's ability to seek healthcare carries a weak correlation with diabetes. Whether a person can afford to visit a doctor, with or without insurance, does not seem to affect the risk of diabetes—at least not to a large extent. People who do not visit a doctor due to cost or lack of insurance may or may not be healthy. The same scenario can be applied to those who were not tested for cholesterol. The input feature "Cholesterol Checked" is not to be confused with "High Cholesterol." The inability or refusal to get one's cholesterol checked says nothing about one's overall health; the presence of high cholesterol does. Those results suggest that a person's accessibility to healthcare serves as a weak indicator for diabetes. Diabetes strongly correlates with habits related to diet and exercise—not healthcare access—at least when analyzing validation accuracies.

7 Removal of Features

The next step is to implement the ideal model from Sections 5 and 6 on multiple input features. In this step, all 21 features will be considered, but one input feature will be removed from the model for each implementation. The removal of an input feature depends on its corresponding validation accuracy. An individual feature from Section 6 that produced a *low accuracy* will be removed from the model before one with a *high accuracy*. The objective is to remove one feature from the model that is considered relatively unimportant in each iteration. By employing this technique, we can determine whether the removal of nonessential features can increase the validation accuracy of the ideal model.

As illustrated in Figure 13 below, the first implementation of the model will run every feature as input *except* "Health Coverage." The next implementation will perform the same task, except the "Cost Prevents Doctor's Visit" feature will be removed along with "Health Coverage." This process will repeat until the model only has two features left to use as input.

```
[[12 'Health Coverage (y/n)' 50.43750631504496]
 [13 'Cost Prevents Doctor Visit (y/n)' 51.24583207032434]
 [3 'Cholesterol Checked (y/n)' 51.96928362129939]
 [11 'Heavy Alcohol Consumption (y/n)' 51.98545013640498]
 [18 'Sex (M/F)' 52.24007274931798]
 [9 'Fruits (y/n)' 52.4704455895726]
 [15 'Days Poor Mental Health' 52.69879761543902]
 [6 'Stroke (y/n)' 53.25856320096999]
 [10 'Veggies (y/n)' 53.37577043548551]
 [5 'Smoke (y/n)' 54.42861473173689]
 [8 'Exercise (y/n)' 57.227442659391734]
 [20 'Education' 57.34464989390724]
 [7 'Heart Attack (y/n)' 57.49419015863393]
 [16 'Days Poor Physical Health' 59.88885520864908]
 [21 'Income' 60.07274931797515]
 [19 'Age' 61.51561079114883]
 [17 'Stairs Difficulty (y/n)' 61.93594018389411]
 [4 'BMI' 64.06183692027886]
 [2 'High Cholesterol (y/n)' 64.36900070728504]
 [14 'General Health (1-5)' 68.63898150954834]
 [1 'High Blood Pressure (y/n)' 69.04314438718804]]
```

Figure 13: A list of 21 features that will be removed in order of corresponding validation accuracy. The validation accuracy values are located to the right of each feature.

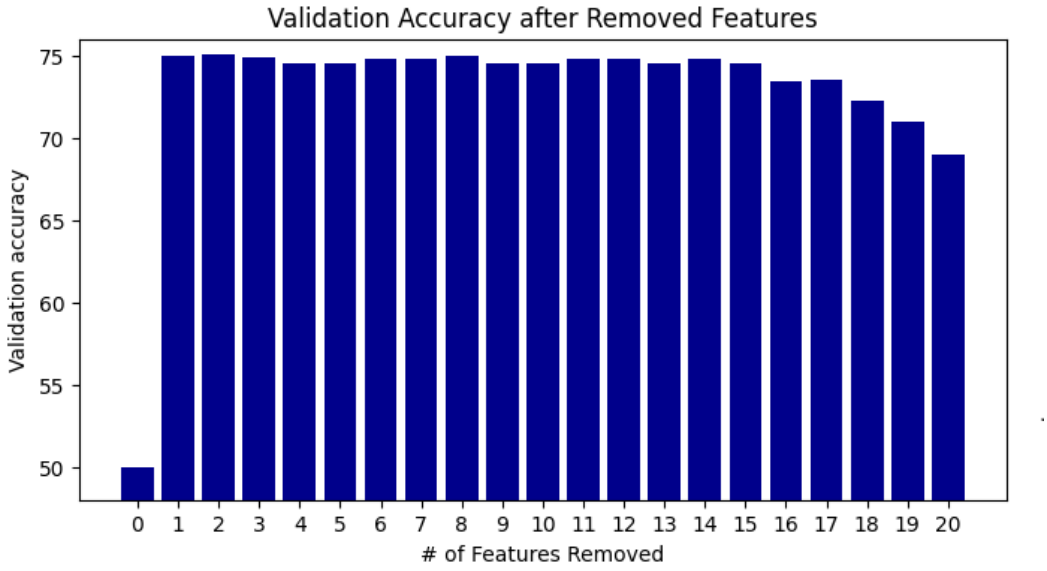


Figure 14: A bar chart visualization illustrating validation accuracies from a model with a specific number of removed features. The chart bar with 0 features removed is the *random baseline classifier* with a baseline accuracy of 50%. It is used to compare the usefulness of the other model executions in this figure.

As illustrated in Figure 14, the removal of the first 15 features did not significantly affect the performance of the model. Those removed features focused on a mix of different categories, including fruit and vegetable consumption, stroke, education, and income—among other attributes. Diabetes serves as a disease that can typically be avoided through healthy habits. Some of those removed features, such as sex and health coverage cost, do not relate to health factors that a patient can control. Therefore, it makes sense that validation accuracies remained consistent after removing the first 15 features.

The "Age" feature served as the sixteenth feature that was removed from the model. The validation accuracy dropped from 74.450% to 73.386% upon the feature's removal. Removing the following features—"BMI" (feature 18), "High Cholesterol" (feature 19), and "General Health (feature 20)—all resulted in significant decreases in performance.

In "Section 5: The Ideal Model," the 3-layer neural network with an 8-4-1 model architecture obtained a validation accuracy of **75.031%** with *every* input feature intact. It turns out the the model's accuracy improved the most by removing the first two features: "Health Coverage" and "Cost Prevents Doctors Visit." Removing only *two features* results in a validation accuracy increase from 75.031% to **75.093%**.

8 Conclusion

The metrics acquired by each model—as well as the iterations from the best possible model—suggest that neural networks may not be suitable for classifying the presence of diabetes in hospital patients. Every model was able to surpass the random baseline accuracy of 50.0%. The simple logistic regression model served as a baseline neural network for this study. The intent was to develop neural network models that surpass the baseline metrics of accuracy, loss, recall, F1, ROC (receiver operating characteristic) and AUC (area under the curve) scores. Almost every model barely surpassed all of the simple logistic model's metrics. One model (the two-layer neural network with a 2-1 architecture) could not even exceed any of the baseline scores. These observations do not support the notion that neural networks can effectively diagnose a person with diabetes.

However, machine learning can still help one determine correlations between diseases and everyday life factors. One example is the consumption of fruits and vegetables. This study allows one to infer

that eating fruits and vegetables may not be as useful for preventing diabetes as one may suggest. Also, a surprisingly strong correlation exists between diabetes and one's ability to climb stairs. These relationships may not have been discovered without implementing neural networks or extracting input features. This statement applies not only to diabetes. It also applies to other facets of health care, such as cancer prevention and ruling out possible diseases that one may have. Perhaps experimenting with non-neural network-based models can help improve the quality of healthcare, and by extension, human life.

9 References

[title page] University of Missouri–St. Louis. (n.d.). (2024). *Horizontal university logo*. Retrieved from https://www.umsl.edu/branding/logos/index.html

[1] American Diabetes Association. (2024). *Statistics About Diabetes*. Retrieved November 18, 2024, from https://diabetes.org/about-diabetes/statistics/about-diabetes

[2] National Institute of Diabetes and Digestive and Kidney Diseases. (2023). *What is diabetes?*. U.S. Department of Health and Human Services. Retrieved November 18, 2024, from https://www.niddk.nih.gov/health-information/diabetes/overview/what-is-diabetes

[3] World Health Organization. (2024, November 14). *Diabetes*. U.S. Retrieved November 18, 2024, from https://www.who.int/news-room/fact-sheets/detail/diabetes#:~:text=Diabetes%20is%20a%20chronic%20disease,hormone%20that%20regulates%20blood%20glucose.

[4] Teboul, A. (2021, November 8). *Diabetes Health Indicators Dataset* [Data set]. Kaggle. Retrieved November 18, 2024, from https://www.kaggle.com/datasets/alexteboul/diabetes-health-indicators-dataset

[5] *CDC Diabetes Health Indicators*. (n.d.). UC Irvine Machine Learning Repository. Retrieved November 18, 2024, from https://archive.ics.uci.edu/dataset/891/cdc+diabetes+health+indicators

[6] Teboul, A. (2021, November 8). *Diabetes Health Indicators Dataset Notebook* [Jupyter notebook]. Kaggle. Retrieved November 18, 2024, from https://www.kaggle.com/code/alexteboul/diabetes-health-indicators-dataset-notebook

[7] Unknown author. (n.d.). *Variable AGE5YR: Fourteen-level age category (20)* [Image]. ResearchGate. Retrieved November 19, 2024, from https://www.researchgate.net/figure/Variable-AGE5YR-Fourteen-Level-Age-Category-20_tbl3_340098871

[8] The National Addiction & HIV Data Archive Program. (n.d.). *EDUCA (Education level completed)* [Variable]. Regents of the University of Michigan. Retrieved November 19, 2024, from https://www.icpsr.umich.edu/web/NAHDAP/studies/34085/datasets/0001/variables/EDUCA?archive=NAHDAP

[9] Resource Center for Minority Data (n.d.). *INCOME2: Income Level* [Variable]. Regents of the University of Michigan. Retrieved November 19, 2024, from https://www.icpsr.umich.edu/web/RCMD/studies/34085/datasets/0001/variables/INCOME2?archive=RCMD