Data 630: Machine Learning Summer 2023

Assignment # 4– Neural Networks

Simon A Hochmuth

Email: shochmuth3@student.umgc.edu

Professor: Ami Gates

Introduction

Objective:

The analysis conducted on the Pima Indian dataset had the goal of using neural networks to classify and understand whether a patient will experience diabetes in their lifetime. A neural network algorithm was used to try to apply machine learning techniques on the data to predict whether diabetes was likely. Once the model was made the overall objective is to decide whether the model had an acceptable about of statistical significance surrounding the predictions made.

Problem Domain:

Diabetes is a disease that plagues over 11.3% of the population of the United States, with a severe impact on minority communities. It costs the country between $200 to $300 million dollars a year, which is why it tends to hurt poorer communities more (DRIF 2022). The focus of analysis of this dataset is to understand how diabetes affects individuals within the Pima Indian minority group. The goal is to use neural networks to build predictive models to support this minority group and hopefully build the quality of life of the people with diabetes.

Method Rationale:

After the data is preprocessed, the goal is to build a neural network model that can be used to classify whether a person has diabetes. The function that will be used is the *nnet()* in R Studio. When classifications are completed, the goal is to then measure the accuracy of the model through statistical analysis to show whether the neural network model was successful in producing the desired objective.

Analysis:

Data:

The data contains 9 variables and 768 observations of women with or without diabetes with Pima Indian ethnicity. The dataset was pulled from a Kaggle repository and was originally compiled by the National Institute of Diabetes and Digestive and Kidney Diseases. The data contains one target variable which states whether someone has diabetes, their diabetes pedigree, age, number of pregnancies, then biological factors like insulin, BMI, or blood pressure. All data is numerical except the target variable which is categorical (UCI 2016).

Exploratory Analysis:

As described earlier the data contains 9 variables and 768 observations, and there are no missing values within the dataset. Overall, the data is relatively clean, so the next step was to explore each variable to understand the skew within the dataset. First the target variable was analyzed, showing that a little over a third of the population observed tested positive for diabetes, as shown in figure 1.

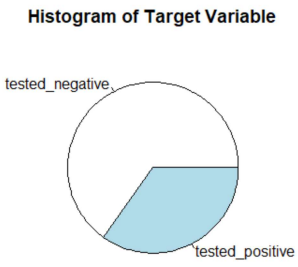

Figure 1: shows distribution of the target variable.

Next, the rest of the numerical values were analyzed to understand if there were any issues or bias within the data. First it was shown that many of the women observed were younger and in their 20s, as shown in figure 2.

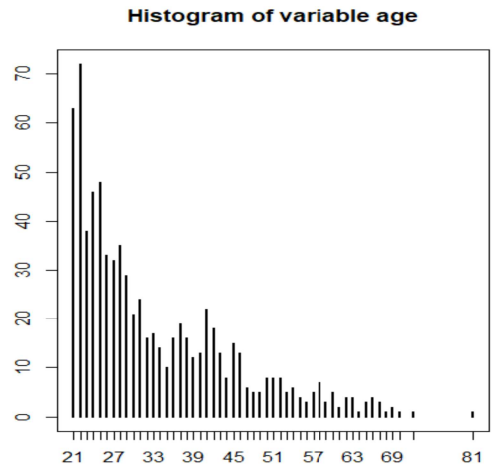

Figure 2: Shows histogram of the variable age.

The variables pedigree and pregnancy seemed to follow a trend that was expected, and no issues were seen within the data. However, after reviewing the rest of the numerical values, it was shown that there was a large percentage of observations that had the value zero. This made sense for pregnancies become someone could have no children, however it is impossible for someone who is alive to have zero insulin, blood pressure, or skin thickness, as shown in figure 3 below. Figure 4 below, shows that insulin and skin thickness had the value zero for 48% and 29% of the total observations.

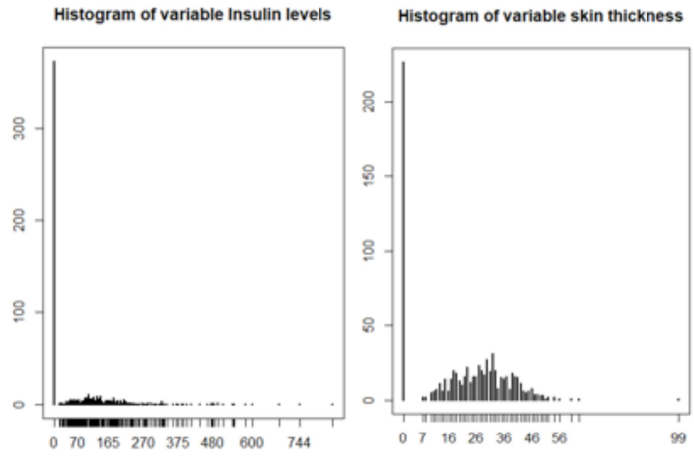

Figure 3: Shows two variables with a large amount of zero values.

```
> res
     preg      plas      pres      skin      insu      mass      pedi       age     class
14.4531250 0.6510417 4.5572917 29.5572917 48.6979167 1.4322917 0.0000000 0.0000000 0.0000000
```

Figure 4: Shows the total percentage of total observations that are zero by column.

Preprocessing:

As discussed in the exploration section, the variables insulin and skin thickness had a large amount of zero values for the observations. For that reason, these two variables were removed due to the amount of missing data. The other variables had less than 5% of the total observations equal to zero, and these were replaced with the mean, as shown in figure 5 below.

```
pima$plas[pima$plas == 0]<-mean(pima$plas)

    preg     plas     pres     mass     pedi      age    class
14.45312  0.00000  0.00000  0.00000  0.00000  0.00000  0.00000
```

Figure 5: Formula to replace zeroes with the mean and total percent of columns equal to zero.

Next, the target variable was converted into a factorized variable with the groups tested positive and tested negative. Lastly, the data was then converted into a test and training dataset which will then be used to fit and test the model.

Algorithm Intuition:

The *nnet()* function in R Studio was used to create a Neural Network model for this data. A neural network has a building block called a neuron which follows the structure shown in figure 6 below. This neuron will combine with other neurons to build a network that is interconnected as shown in figure 7. Together several inputs will feed each neuron, and together they will provide a desired output, where the output in this analysis will be whether a woman has diabetes or not. A network can have any number of layers; however, the labels will be the input, output,

and hidden layers. Hidden layers are in between the input and output layer and are created and used by the model to find the output desired. (Turing. 2022).

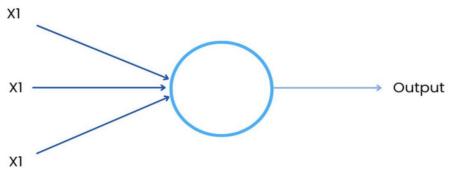

Figure 6: structure of a single neuron (Turning 2022).

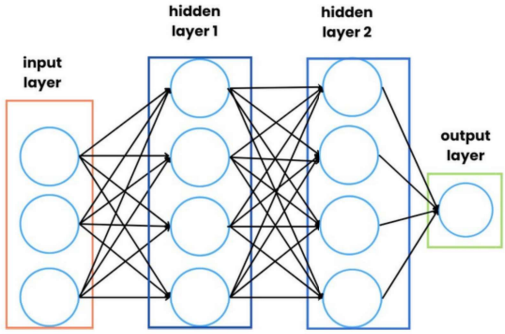

Figure 7: structure of a neural network (Turning 2022).

Model Fitting:

The first step towards fitting this model was to split the data into a test and training set, after all preprocessing was completed. Once the model was made, the accuracy and performance of the model could be analyzed through tuning the parameters. Once tuning was complete values for the parameters *size*, *decay*, and *maxit* were chosen. The model was then refitted on the training set and tested, with the new parameters chosen.

Result:

Output and Model Properties:

The *nnet()* model that was used had the parameters size=6, decay=0, and maxit=100 as shown in figure 8 below.

```
#6. Use the training data to build the model
fit <- nnet(class~., train.data, size=6, decay=0, maxit=100)
```

Figure 8: Final formula used to fit the neural network model.

It then was used to predict if someone would test positive for diabetes on the test dataset, which resulted in the output shown in figure 9. The model had six variables that were used as inputs, with six hidden layers that were interconnected with the output layer for the feature class.

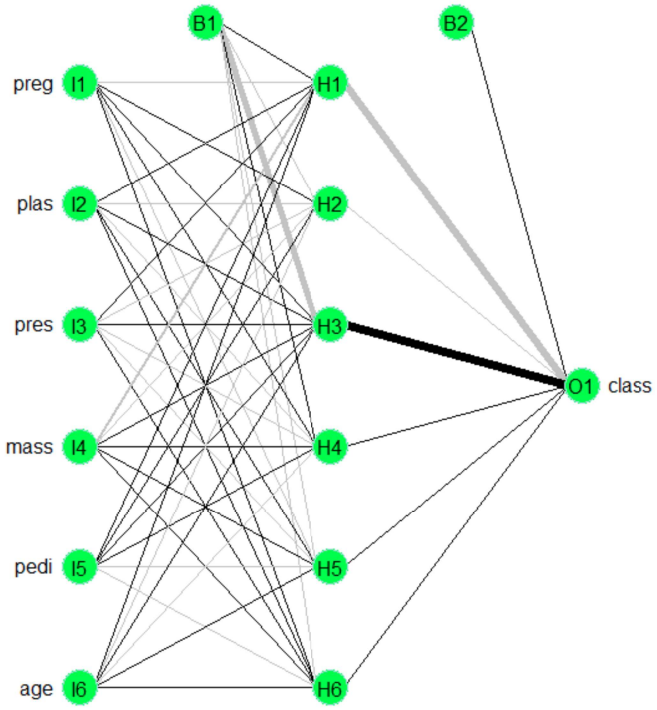

Figure 9: Shows neural net diagram for the diabetes dataset, without values.

The neural network discussed above then produced predictions on the test dataset shown in the confusion matrix of figure 10 and 11 below. Where the predictions on the test dataset, in figure 11, show that 58 women were correctly predicted to have been positive for diabetes and 126 were correct in predicting a negative test. Where 50 predictions were incorrectly labeled. Over-all, it looks as though the model was successful in predicting whether a woman had diabetes

based on the outputs for both the test and training set. The next step is to statistically evaluate the model to see if that is a correct assumption.

predictions	tested_negative	tested_positive
tested_negative	302	73
tested_positive	51	108

Figure 10: Confusion matrix for the training dataset

predictions	tested_negative	tested_positive
tested_negative	126	29
tested_positive	21	58

Figure 11: Confusion matrix for the test dataset

The features were also analyzed to show the importance each had on the dataset, as shown in figure 12 below. The figure shows that blood pressure and diabetes pedigree were the most important of the variables, while pregnancy and age were on the lower end. Overall, the importance for each feature was relatively close, and future work could include dropping the least important variables to see how the model accuracy is affected.

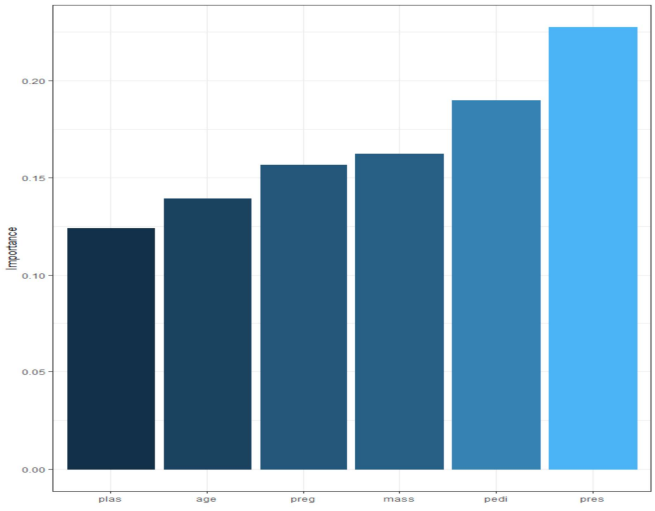

Figure 12: Diagram showing feature importance. Variables from left to right are: pregnancy, age, plasma, blood pressure, mass, and pedigree.

Evaluation:

Next, the statistical analysis was performed on the model to show how the model performed. The test dataset had a confusion matrix and performance metrics shown in figure 13. The model had 78% accuracy in predicting the target variable, and a very low p-value which showed the predictions were not due to random chance. The sensitivity and specificity were 85% and 66%, which is closer to 1 than 0. This means that the model did well in correctly identifying the positive and negative class. Overall, the model was successful in predicting whether a woman had diabetes based on the statistical metrics provided in figure 13.

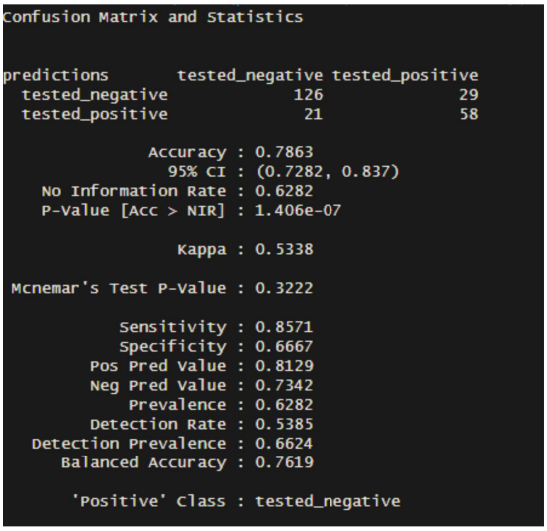

Figure 13: Confusion matrix and statistics surrounding the test dataset.

Conclusion:

Summary:

The model had 78% percent accuracy on the test data, and once a statistical evaluation was completed it was decided this was not due to random chance. This means that the model was successful in predicting whether a woman had tested positive for diabetes, however it did incorrectly label a few values. Future work would be to try and grow the accuracy of the model to 100%. The objective of this analysis was successful, and this model can be used to start working

on predicting diabetes within the Pima Indian Tribe. However, there are some limitations and future work that will need to be worked on to build confidence in the model itself.

Limitations:

One limitation of the dataset was the usefulness of the features. In this dataset, insulin and skin thickness were measured by medical professionals in the field because of its importance in finding diabetes. However, a large portion of the data had missing values for these, meaning these features could not be used in the model. There is a chance the model could have been more accurate if these two features were included. Another limitation is the amount of data that was collected. Only 768 observations were made, and this could lead to a bias in the data due to the lack of variations in observation. The confidence in the model would increase dramatically if it was trained and tested on far more observations.

Improvement Areas:

As discussed previously, the features insulin and skin thickness were dropped from the dataset due to the number of missing values. Medical professionals who created this dataset valued the insights these two variables provide when predicting whether someone has diabetes. In the future one big step could be to ensure that each of the features is correctly measured and provided to add value to the overall model. Next, work can be done to increase the total number of observations from 768. This will provide more data and variations for the model to learn and test with, thus building the overall confidence others have in the model itself.

References:

UCI Machine Learning. (2016). *Pima Indians diabetes database*. Kaggle. https://www.kaggle.com/datasets/uciml/pima-indians-diabetes-database

DRIF. (2022, May 27). Diabetes statistics. Diabetes Research Institute Foundation. https://diabetesresearch.org/diabetes-statistics/

Turing. (2022, May 2). *How neural network models in machine learning work?* Hire the World's Most Deeply Vetted Remote Developers | Turing. https://www.turing.com/kb/how-neural-network-models-in-machine-learning-work#types-of-neural-networks