Roy Phelps
July 15, 2023

NEURAL NETWORK

Table of Contents

Introduction

Neural. Networks have proven to be valuable tools in medical research and diagnostics, particularly in the identification and classification of malignant and benign cells. These artificial intelligence models are capable of learning complex patterns from large datasets, enabling them to analyze and interpret various features of cell morphology and behavior. By leveraging the power of neural networks, researchers and healthcare professionals can enhance their ability to accurately distinguish between malignant and benign cells, aiding in the early detection, diagnosis, and treatment of cancer. That makes neural network analysis on distinguishing between malignant and benign cells, the focus of this analysis. This technology holds great promise in improving patient outcomes and revolutionizing the field of oncology. As described by (Patel, 2020), the use of neural networks for cell classification has demonstrated significant advancements, showcasing the potential of this approach in clinical practice.

According to the Center for Disease Control and Prevention (CDC), there were 1.7 million new cancer cases in 2019 and over 600,000 people died in the U.S alone (Facts, 2023). That equates to for every 100,000 people, 439 new cases were filed and 147 people died. This data frames the cancer problem in the United States well.

The ability of neural networks to learn from diverse data sources and uncover complex relationships makes them a relevant and powerful method for cancer diagnosis and hold promise for enhancing early detection and personalized treatment strategies.

Analysis and Model Demonstration

The neural network algorithm is a class of machine learning algorithms inspired by the structure and function of the human brain. Neural net algorithms, specifically the package *neuralnet* in R, is a type of feedforward neural network that consists of an input layer, one or

more hidden layers, and an output layer. It uses a series of interconnected nodes, or artificial neurons to process and transform input data (Ian Goodfellow, Deep Learning, 2016). It uses a series of interconnected nodes, or artificial neurons, to process and transform input data. Each node applies a weighted sum of the inputs, passes it through an activation function, and outputs a result. The algorithm learns from training data by adjusting the weights and biases of the connections between nodes through a process called backpropagation. The *nerualnet* algorithm has been widely used in various application, including image and speech recognition, natural language processing, and pattern recognition. Using the algorithm as pattern recognition, will be the focus of this analysis.

Parameter tuning, also known as hyperparameter optimization, is an essential step in building machine learning models, including neural networks (Bergstr, 2012). Parameters are the setting or configurations of the model that are not learned from the data but are set manually or pre-defined. In neural networks, these parameters include the number of hidden layers, the number of nodes in each layer, the learning rate, the activation functions, and more. Selecting the set that yields the best model, is crucial in the accuracy. Training and evaluating the model on a validation set, makes the accuracy reproducible on a large data set.

When building a neural net model, it is important to evaluate its performance to assess the effectiveness and generalization capabilities (Danny Hernandez, 2019). Various evaluation metrics can be used, depending on the specific task. For classification problem, metrics like accuracy, precision, recall, and F1 score are commonly used. Model evaluation provides insights into the strengths and weaknesses of the neural network model and helps in identifying areas for improvement. Regularization techniques, such a L1 regularization, can also be employed to prevent overfitting and enhance the model's generalization capabilities.

Exploratory Data Analysis

To learn patterns and relationships in determining if a cell is malignant or benign, the data comes from (Dr. William H. Wolberg, 1995) for the neural network. The predicting field is categorical and has two outcomes, benign or malignant represented by *B* and *M* respectively which is named **diagnosis** and is unique. All other variables, eleven of them, are real-valued and computed for each cell nucleus. They are as follows, **ID**, represents the record ID for each entry and is unique. **radius**, representing the average distance from the center of the nucleus to its boundary. **Texture** quantifies the variation in gray-scale intensity levels of the nucleus. **perimeter** refers to the total length of the nucleus boundary. **area** represents the total area occupied by the nucleus. **Smoothness** quantifies the variation in the local lengths of the nucleus boundary. **Compactness** is a measure of how closely the nucleus is packed together. **Concavity** measures the severity of the concave portions of the nucleus. **Concave** quantifies the number of concave portions of the nucleus. **Symmetry** represents the symmetry of the nucleus shape. Finally, **fractal** which measures the complexity of the nucleus boundary.

It is important to note that each one of these variables, except for **diagnosis** and **ID,** has two more sets associated with them. For example, the variable **radius** has a dataset called **radius2** and **radius3** that has real-valued data assigned to them. Another example is **texture** and has its counterparts **texture2** and **texture3** with real-valued data in each. This increases the overall data for the creation of the neural network. In total, there are 30 variables that will contribute to the patterns and relationships in the model.

Loading the data int R and assigning it a data frame of name *di*, the function *summary(di)* is initiated to see if there are any missing data, of which are none. This also is utilized to understand the structure, summary statistics, and distribution of the key variables. Initiating the

str(di) function, provides a concise summary of the data frame. It displays the structure of the dataset, showing the variable names, their data types, and the first few observations of each variable. This helps in understanding the types of variables present and their potential relationships. The *head(di)* function displays the first six rows of the data frame that allows examination of the structure and format of the data. The target variable in this analysis is **diagnosis**, that represents if a cell is benign or malignant. It has the character values of *B* and *M* respectively. All other variables are independent and have the type of numeric values.

Data Information and Preprocessing

After the exploration process, some of the variables need to be addressed in the pre-processing stage for the R *nerualnet* algorithm can be deployed on the dataset. The variable **ID** is a unique identifier and does not contribute to the analysis; therefore, it is removed using the command *di$ID<-NULL*. Another transformation is the target variable **diagnosis**, which is in the type character represented by *B* and *M*. The nomenclature for these two are benign and malignant respectively. Converting this to an integer type of 0 for benign and 1 for malignant, will be simpler in scaling and standardizing the data. The *neuralnet* package in R works best if the data is scaled and standardized (Verma, 2020).

Results and Model Evaluation

The objective of this analysis is to provide relationships between the target variable and the variables in a supervised learning environment.

Firstly, initializing a training set and a test set that can be reproducible on a smaller sample of the data will be useful for determining the model's performance. Using the *set.seed(12345)* function then splitting the data into training and test sets, *ind <- sample(2, nrow(di), replace = TRUE, prob = c(0.7, 0.3))*

train.data <- di[ind == 1,]

test.data <- di[ind == 2,], will accomplish this.

Next, the target variable is **diagnosis**, which is a result of 0 or 1 binary, 0 for benign and 1 for malignant. This is compared to all the other independent variables. The general equation is

$$Dependant \sim independant(activation\ function)(regularization)(hidden\ layers)$$

There are some parameters tuning in the *neuralnet* algorithm that will improve the model's performance which are activation function, regularization, and the number of hidden layers.

Then, the activation function used in the model is the logistic function, also known as the sigmoid function and it is defined as $f(x) = \frac{1}{1+e^{(-x)}}$. The activation function is applied to the output of each neuron in a neural network layer, allowing the network to introduce non-linearity and make complex mappings between the input and output. The logistic activation function, denoted as "logistic" or "sigmoid," has an "S"-shaped curve and maps any real-valued number to a value between 0 and 1 (Topper, 2023).

Next, the L1 regularization prevents overfitting and controls the model's complexity by penalizing higher terms in the model (Yildrim, 2020). It forces the uninformative weights to be zero by subtracting a small amount of the weight at each iteration. This is accomplished by the command *rep = 3* in the R's *neuralnet* package.

Finally, to increase the number of hidden layers and neuron to capture more complex patterns in the data, the command *hidden = c(3)* is initiated. This adds 1 hidden layer with 3 neurons to the model.

There are some options that can be called in the *neuralnet* package in R. They can be seen in the accompanying R code in the section *network properties.* Importantly the output of

the command *nn$net.result[[1]][1:10],* gives the first ten predicted probabilities and they are all at 99%. This is a strong model prediction.

To visualize how this model looks, in *Figure 1,* the neural network is displayed.

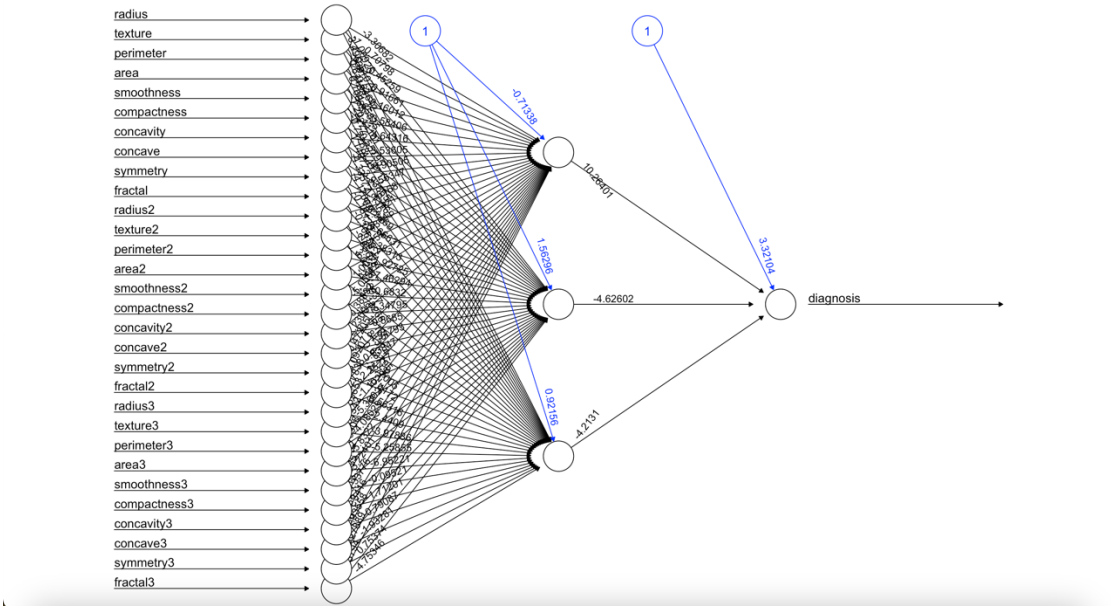

Figure 1: Neural Network

Here, the independent variables are processed through the algorithm to assign weights to the connections in each neuron. There is one hidden layer and 1 output. To visualize this another way, in *Figure 2*, the hidden layer *B1* and the output *B2* are shown.

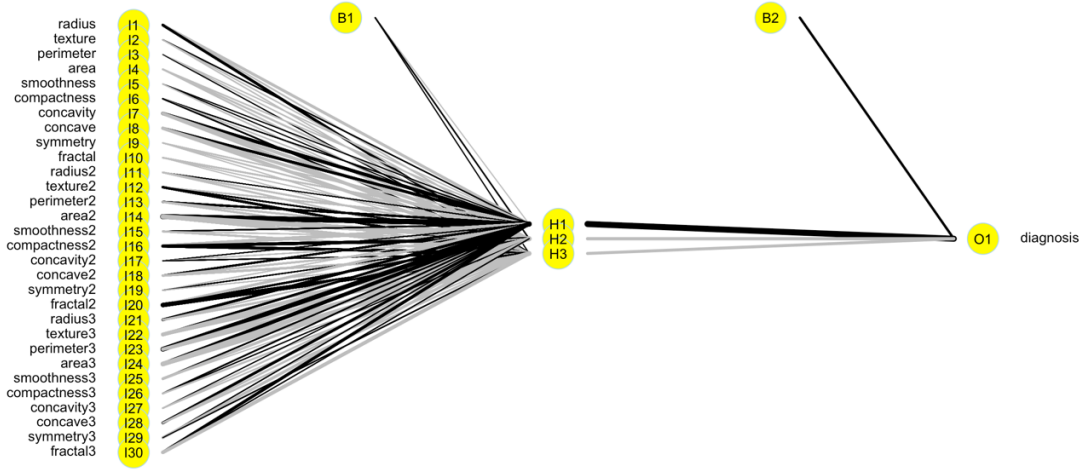

Figure 2: Neural Network 2

Since there is only one hidden layer and one output, the plot command *garson(nn)* is implemented resulting in showing the importance for each variable that contributes to the neural network, *Figure 3*. Here, the variables **area2, area3,** and **parimeter3** are the top three that contributes to the diagnosis.

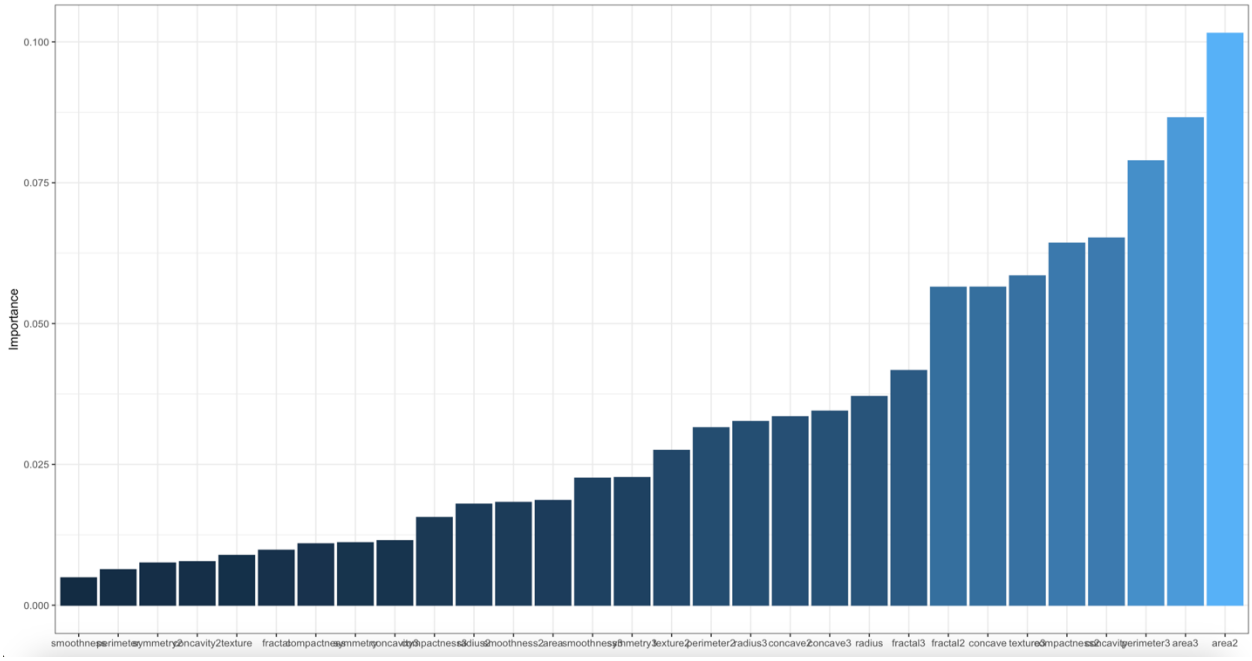

Figure 3: Importance

To determine the performance of a model, the confusion matrix is used. In this models' case, the value for the training set is .99 and the value for the test set is .97. This means that the model accurately classified 97% of the data.

Another method to test for the accuracy of the model is the F1 score (Kundu, 2022). It calculates the precision and recall into a single value, providing a harmonic mean of the two measures. It ranges between 0 and 1, where and F1 score of 1 indicates a perfect mode, while a score of 0 indicates poor performance. This this model's case, the F1 score on the training set is 99.6% and the F1 score on the test set is 96.6%.

Conclusion

There were some interesting findings upon conducting this analysis. For example, the strength of the model's performance and accuracy capabilities with and F1 score of 97% and 96.6% on the training and test sets respectively. The use of a neural network in determining if a cell is cancerous or not, is clearly beneficial in a patients care.

Another interesting finding was the relationship between the **area** and **perimeter** of a cell can have a significant importance in determining if a cell is cancerous, *Figure 4*. These parameters have a significant linear relationship with the analysis and can be a driver for further analysis.

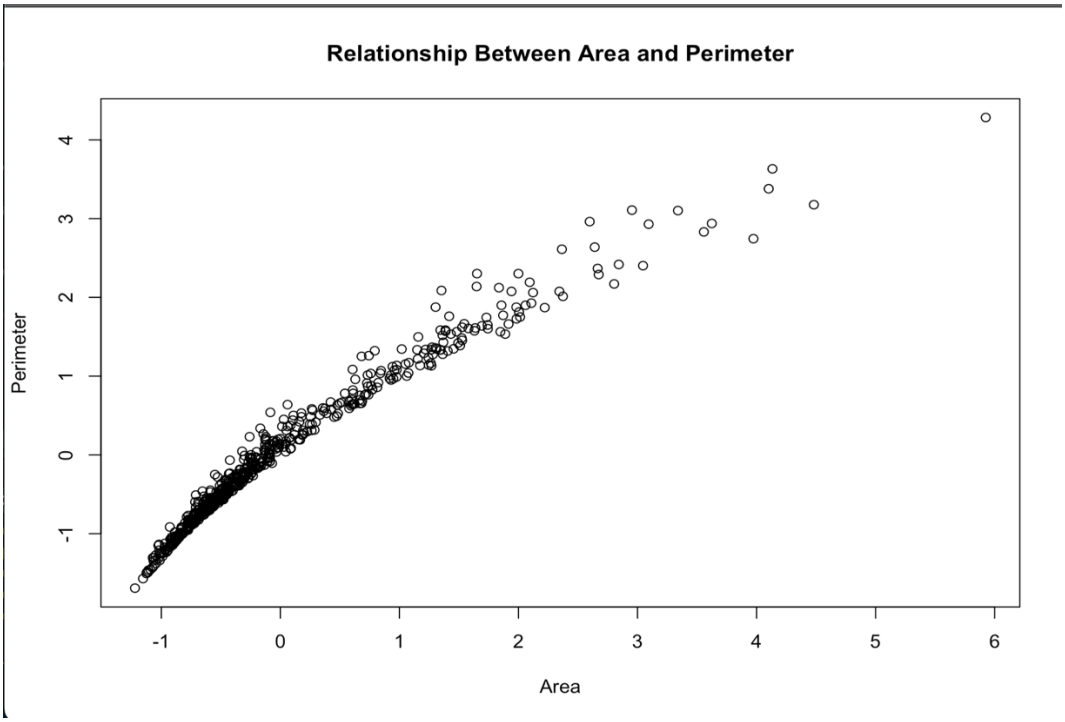

Figure 4: Area vs Perimeter

The *neuralnet* package in R demonstrated high accuracy with an impressive accuracy rate of 99% on the training set and 97% on the test set. This indicates the model accurately classified most of the data instances.

Using the *caret* package in R, revels some more interesting statistics *Figure 5*. The no information rate shows the accuracy rate that can be achieved by a simple model that always predicts the majority class. In this case, the no information rate is 0.6031, meaning that if the model predicts the majority class (class 0) for all instances, it will achieve an accuracy of 60.31%. The Kappa statistic measures the agreement between the predicted and actual classes. A value of 1 represents perfect agreement. In this case, the value is 0.9456 indicating strong agreement beyond what would be expected by chance. Balanced accuracy calculates the average of sensitivity and specificity, providing an overall measure of classification accuracy that accounts for imbalanced class distribution. In this case, the balanced accuracy is 0.967, or 96.7%.

```
Confusion Matrix and Statistics


testPred   0   1
       0 117   5
       1   0  72

               Accuracy : 0.9742
                 95% CI : (0.9409, 0.9916)
    No Information Rate : 0.6031
    P-Value [Acc > NIR] : < 2e-16

                  Kappa : 0.9456

 Mcnemar's Test P-Value : 0.07364

            Sensitivity : 1.0000
            Specificity : 0.9351
         Pos Pred Value : 0.9590
         Neg Pred Value : 1.0000
             Prevalence : 0.6031
         Detection Rate : 0.6031
   Detection Prevalence : 0.6289
      Balanced Accuracy : 0.9675

       'Positive' Class : 0
```

Figure 5: Carat Package

Overall, the findings from the analysis highlight the potential of neural networks in improving patient outcomes and revolutionizing the field of oncology. By leveraging the power of artificial indolence and machine learning, researchers and healthcare professionals can

enhance their ability to accurately identify and classify aner cells, leading to dearly detection and personalized treatment strategies.

Limitations and Improvements

One limitation is the possibility of data imbalance. The description does not mention weather the dataset used for the analyses is balanced or imbalanced in terms of the number of benign and malignant cells. Imbalanced datasets can introduce bias in the model's performance evaluation, as high accuracy or F1 score may be achieved by simply predicting the majority class.

Another limitation is interpretability. Even though neural networks can provide some accurate predictions, they sometimes often lack interpretability. Understanding the reasoning behind the model's prediction and relationships between the variable with the specific features it relies on can be challenging, especially in complex neural network architectures.

An area for improvement would be dataset diversity. It would be beneficial to expand the analysis to multiple datasets from different sources to assess the model's generalization capabilities across different populations. This would enhance the reliability and applicability of the findings.

In conclusion, it would be beneficial to and crucial to interpret the results and findings with caution.

References

Bergstr, J. &. (2012). *Randome search for hyper-parameter optimaztion.* Journal of Machine Learning Research.

Danny Hernandez, T. B. (2019). *Measuring the Algorithmic Efficiency of Neural Networks.* Retrieved from CDN: https://cdn.openai.com/papers/ai_and_efficiency.pdf

Facts, U. (2023, March 28). *US cancer rates and trends: how have cancer rates and mortality changed over time?* Retrieved from USA Facts: https://usafacts.org/articles/how-have-cancer-rates-changed-over-time/?utm_source=bing&utm_medium=cpc&utm_campaign=ND-HealthSafety&msclkid=6445212d732314b6c1cd4581f3f1fa62

Ian Goodfellow, Y. B. (2016). *Deep Learning.* MIT Press.

Ian Goodfellow, Y. B. (2016). *Deep Learning.* MIT Press.

Kundu, R. (2022, September 13). *Confusion Matrix: How To Use It & Interpret Results*. Retrieved from v7 labs: https://www.v7labs.com/blog/confusion-matrix-guide

Patel, N. G. (2020). *Application of Artigicial Neural Networks in Breast Cancer Classification Using Fine Needle Aspiration Cytology.* Journal of Pathology Informatics.

Topper, N. (2023, July 10). *Sigmoid Activation Function: An introduction*. Retrieved from built-in: https://builtin.com/machine-learning/sigmoid-activation-function

Verma, A. (2020, 7 20). *Building A Neural Net from Scratch Using R - Part 1*. Retrieved from R Views: https://rviews.rstudio.com/2020/07/20/shallow-neural-net-from-scratch-using-r-part-1/

Wolberg, W. M. (1995, October). *Breast Cancer Wisconsin (Diagnostic)*. Retrieved from UCI Machine Learning Repository: https://doi.org/10.24432/C5DW2B

Yildrim, S. (2020, May 8). *L1 and L2 Regularization Explained*. Retrieved from twards data science: https://towardsdatascience.com/l1-and-l2-regularization-explained-874c3b03f668