UNIVERSITI TUN HUSSEIN ONN MALAYSIA

FACULTY OF COMPUTER SCIENCE AND INFORMATION TECHNOLOGY

(FSKTM)

SEMESTER II 2024/2025

DATA MINING

BIT 33603

SECTION 03

LAB ASSIGNMENT 07

TITLE

CLASSIFICATION WITH NEURAL NETWORK IN R

LECTURER'S NAME

DR. ROZITA BINTI ABDUL JALIL

NAME	TUAN KHALIDAH SYAZWANA BINTI TUAN MOHD KASMAWI
MATRIC NUMBER	AI220118
DATE SUBMISSION	May 06, 2025

Topic: Classification with Neural Network in R

Objectives:

1. Learn how to preprocess data and prepare packages for neural network classification in R.
2. Develop a neural network classification model using the neuralnet package.
3. Evaluate model performance using confusion matrix and metrics like accuracy, precision, recall, and F-score.
4. Visualize the architecture of the trained neural network.

Duration: 2 hours

Assessment Question:

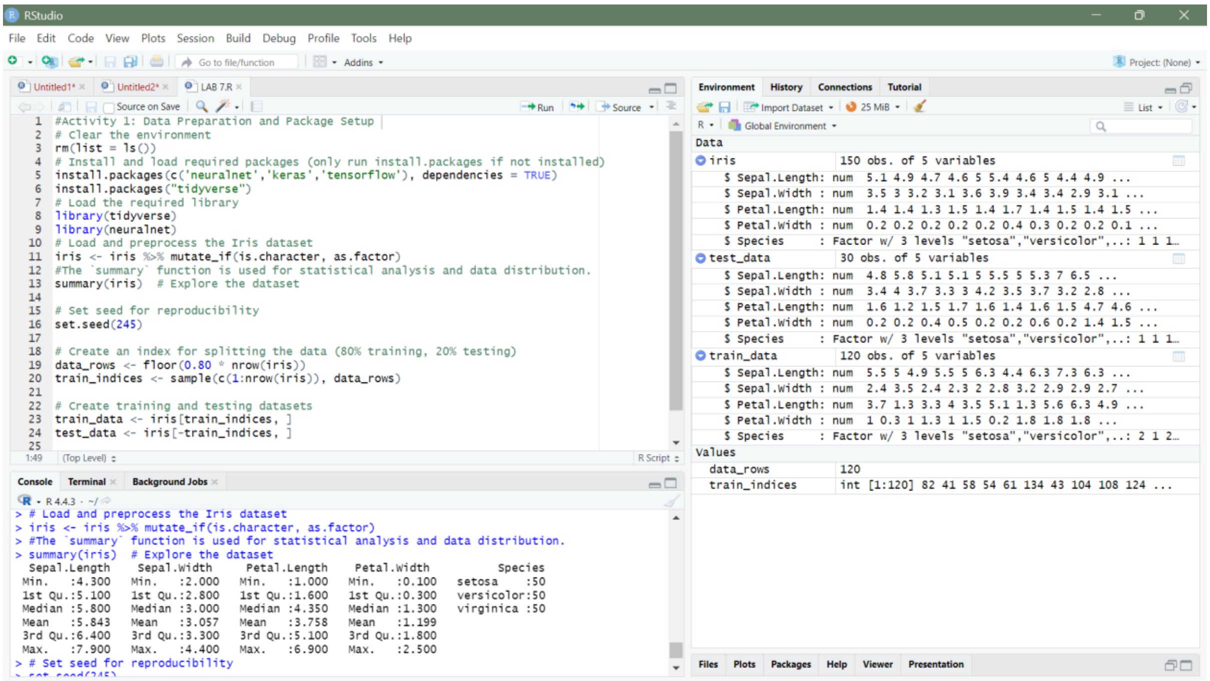
1. Run the provided code in R (Activity 1-4) and understanding the classification using Neural Network.
2. Submit the visualizations as image/data snapshots for each activity along with a brief explanation of the insights gained.

Activity 1: Data Preparation and Package Setup

Instruction:

In this activity, you will prepare your R environment by installing and loading necessary packages. You will also load and inspect the Iris dataset, then split it into training and testing sets (80:20 ratio)

Source code:



Justification:

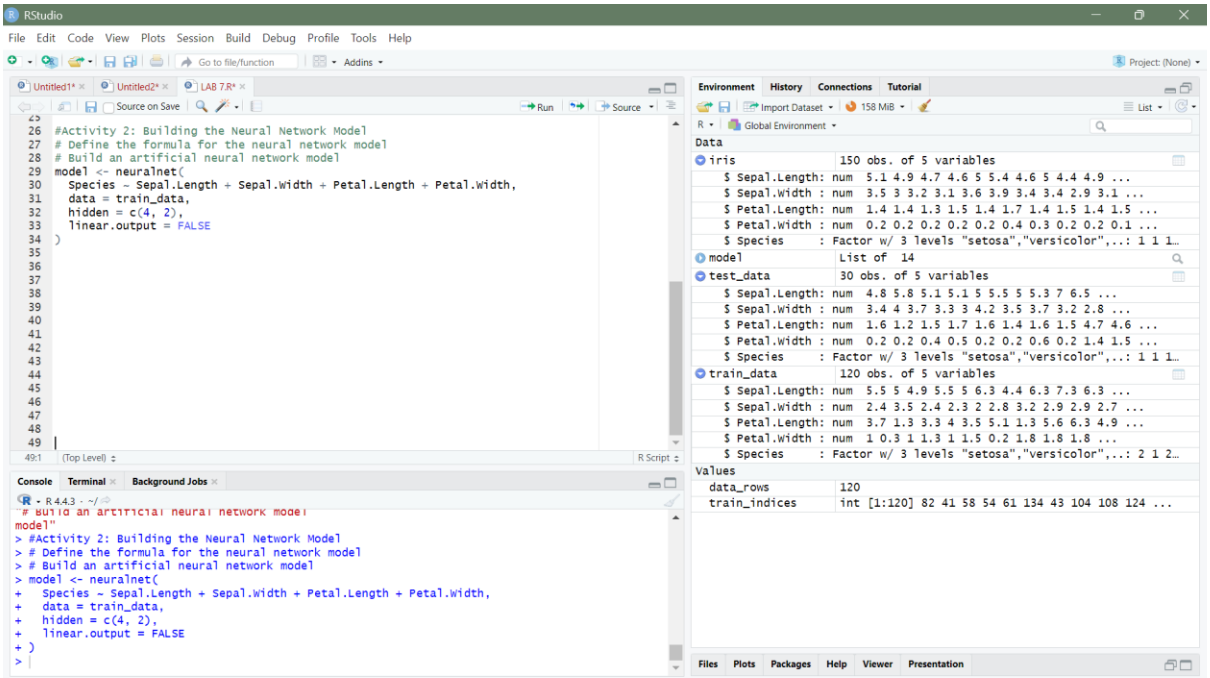
This activity is important because it gets everything ready before we build a model. We start by cleaning the R environment to avoid errors. Then we install and load the needed libraries like neuralnet and tidyverse. We use the iris dataset, which is popular for testing classification models. To make sure the model works properly, we convert character columns to factor type. Finally, we split the data into training and testing sets. This allows us to train the model with one part of the data and test its accuracy using the other part later.

Activity 2: Building the Neural Network Model

Instruction:

Now you will build a feedforward neural network with two hidden layers. You will use the training dataset created earlier to train the model.

Source code:



Justification:

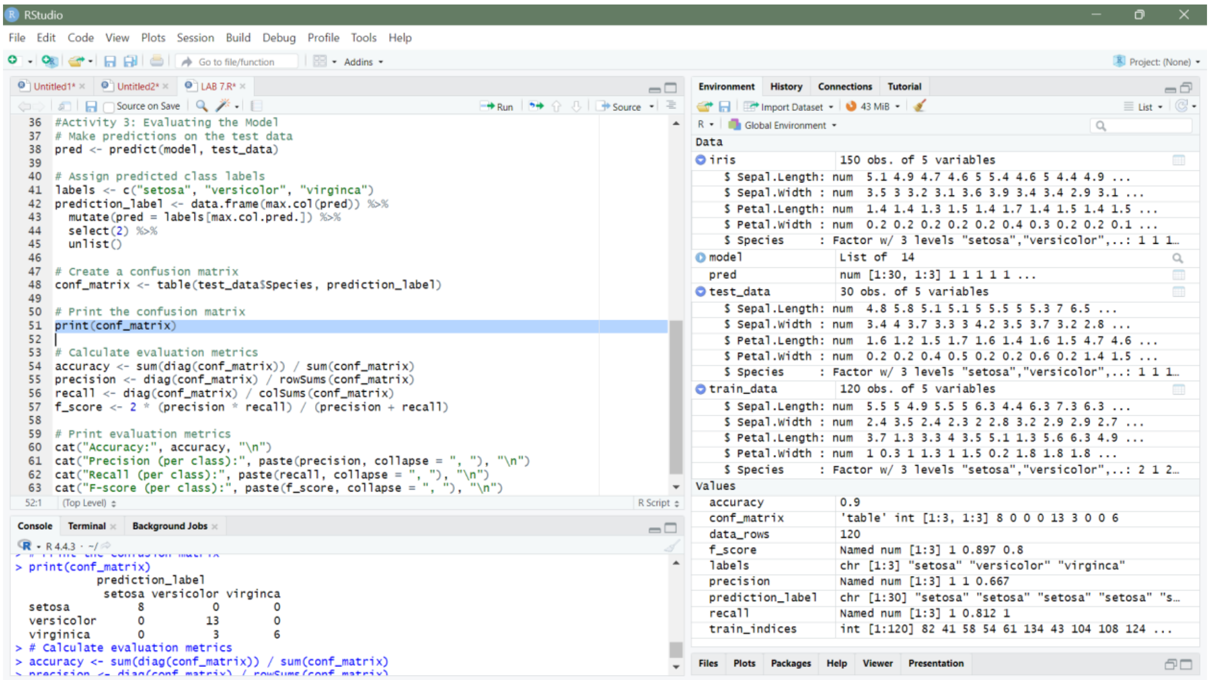
In this activity, we build a neural network model using the training data. This model tries to learn the relationship between the flower measurements (like petal and sepal length) and the flower species. The neuralnet() function builds this model with two hidden layers. Hidden layers help the model understand complex patterns. This step is very important because it teaches the computer how to predict the species based on the input data. The better the model learns from the training data, the more accurate it will be when making predictions on new, unseen data.

Activity 3: Evaluating the Model

Instruction:

In this activity, you will evaluate the performance of your trained neural network using the testing dataset. You will generate predictions and compute the confusion matrix along with key classification metrics.

Source code:




```
> # Print evaluation metrics
> cat("Accuracy:", accuracy, "\n")
Accuracy: 0.9
> cat("Precision (per class):", paste(precision, collapse = ", "), "\n")
Precision (per class): 1, 1, 0.666666666666667
> cat("Recall (per class):", paste(recall, collapse = ", "), "\n")
Recall (per class): 1, 0.8125, 1
> cat("F-score (per class):", paste(f_score, collapse = ", "), "\n")
F-score (per class): 1, 0.896551724137931, 0.8
>
```

Justification:

After building the model, we need to check how well it performs. In this activity, we use the test data to make predictions and compare them with the actual labels. We create a confusion matrix to see where the model was right and where it was wrong. Then, we calculate the accuracy, precision, recall, and F-score to understand the performance in more detail. These values help us know if the model is good or needs improvement. Evaluation is necessary to make sure the model is reliable before using it in real-world tasks.

Activity 4: Visualizing the Neural Network Architecture

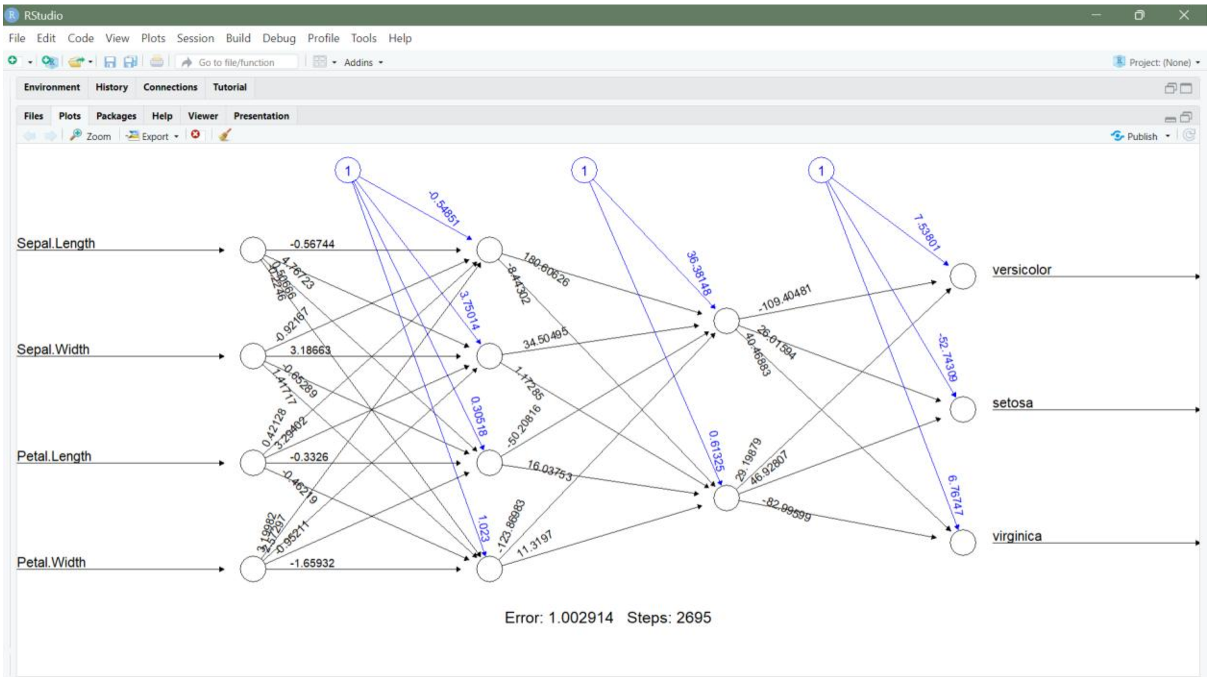
Instruction:

Lastly, you will visualize the structure of the trained neural network. The plot() function helps you see how many nodes and connections the model has.

Source code:

```
#Activity 4: Visualizing the Neural Network Architecture
# Visualize the neural network architecture
plot(model, rep = "best")
```

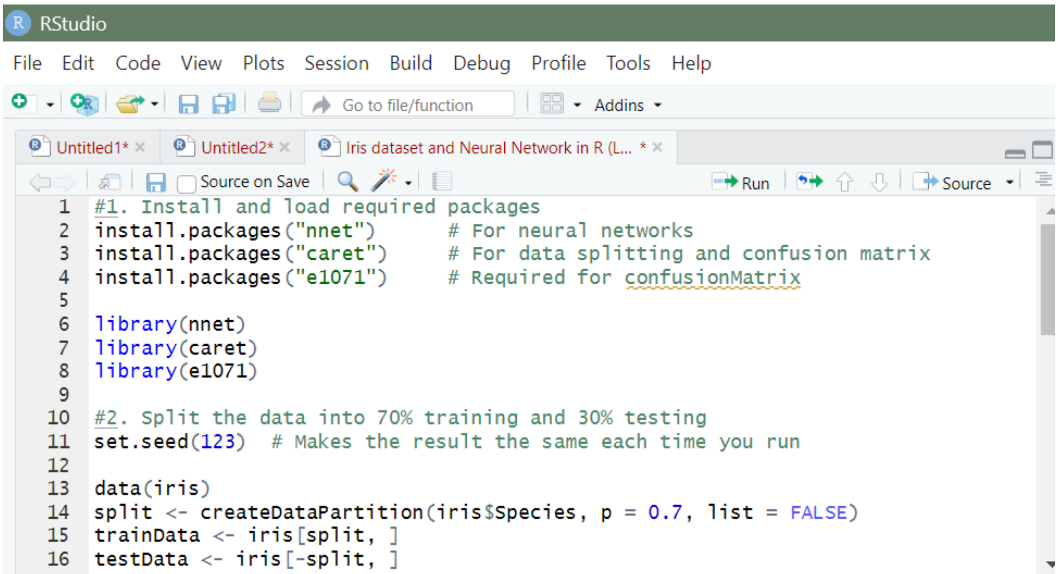
```
> #Activity 4: Visualizing the Neural Network Architecture
> # Visualize the neural network architecture
> plot(model, rep = "best")
>
```



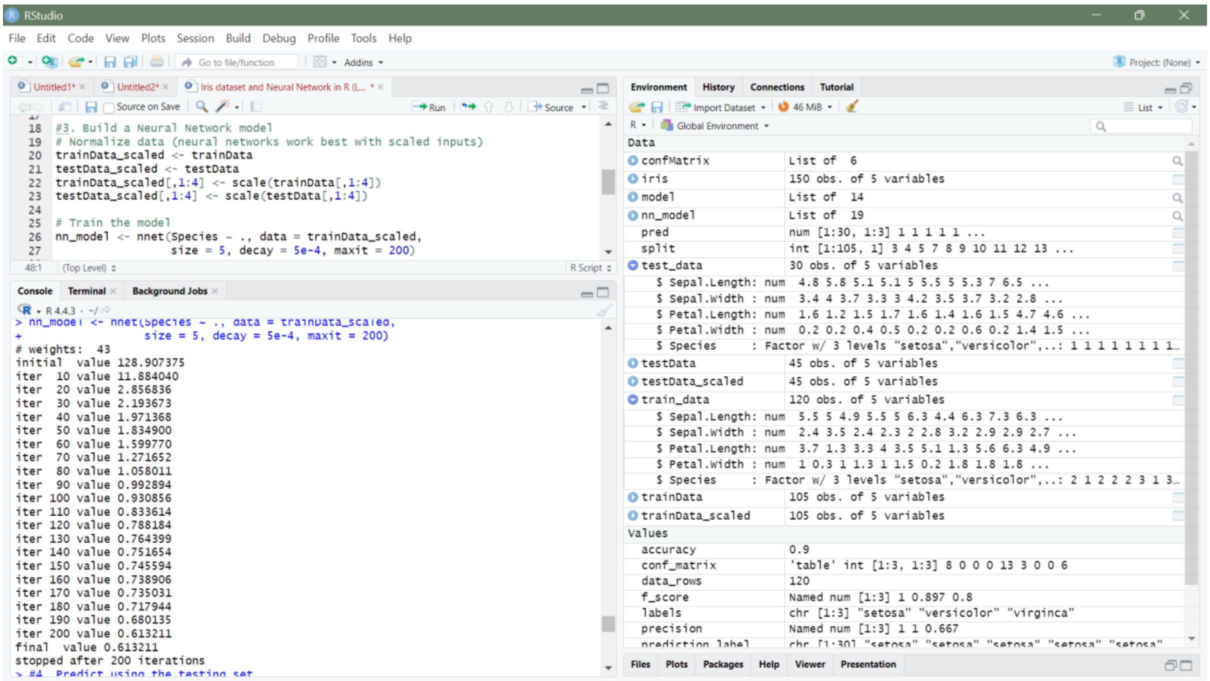
Justification:

This activity helps us see the structure of the neural network we built. By using the plot() function, we can visualize how the input values connect to hidden layers and how those connect to the output. This makes it easier to understand what's happening inside the model. It also helps in identifying if the network has too many or too few layers or nodes. Visualization is useful for both learning and debugging, and it can help us explain the model to others more easily. It's a helpful step for better understanding and presentation of the model.

3. Using the iris dataset in R, perform a classification task using the Neural Network algorithm with the following steps:
    1. Split the data into 70% training and 30% testing. Make sure to use set.seed() so your result is reproducible.

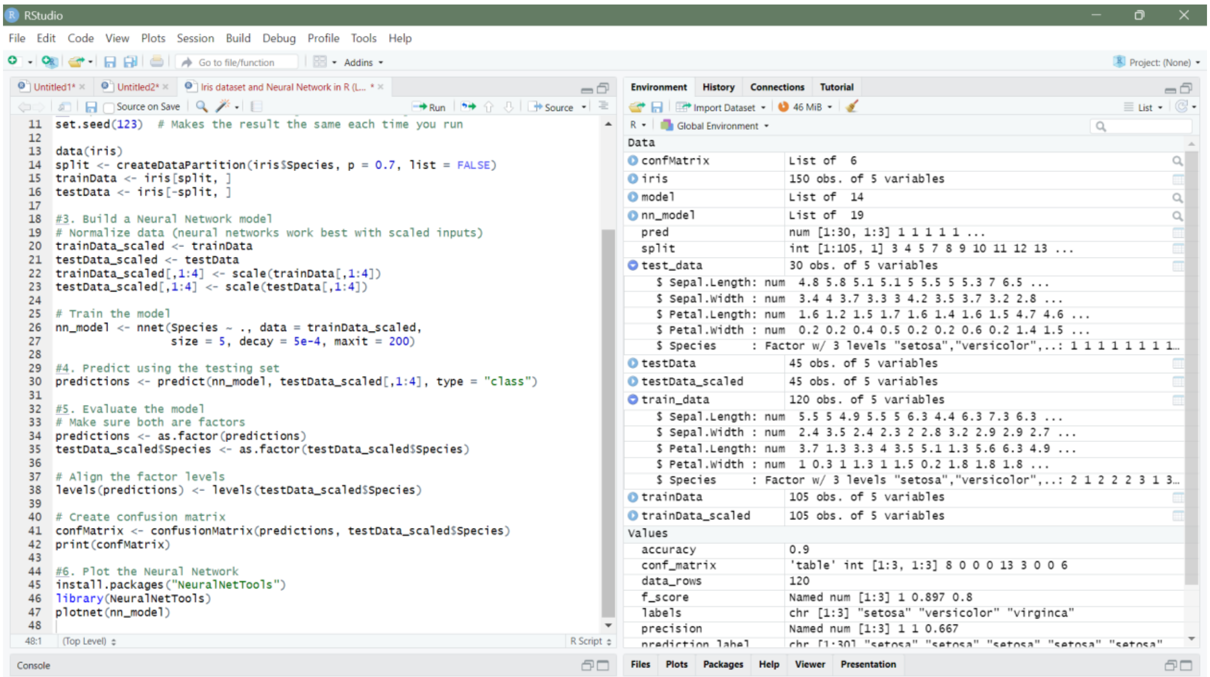


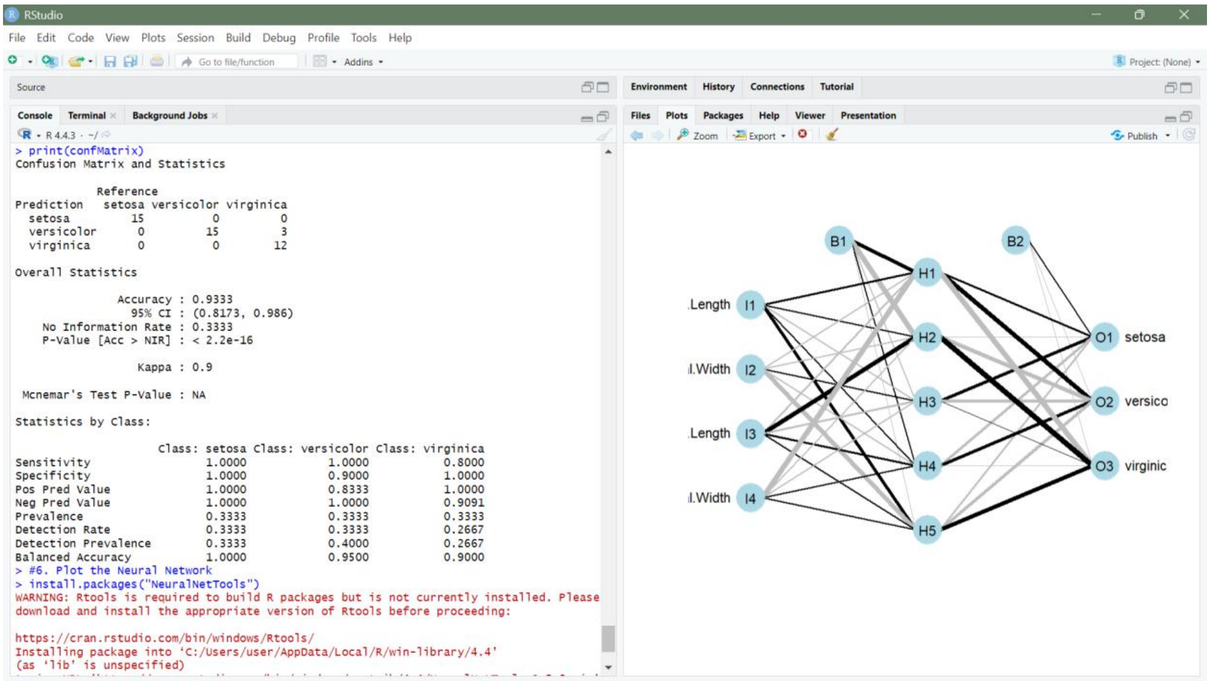
    2. Build a Neural Network classifier using the training set to predict the flower species.



    3. Evaluate your model using the testing set and report:
        1. Confusion matrix
        2. Accuracy
        3. Precision, Recall, and F-score for each class

4. Plot the Neural Network and observe its structure.





Self-Assessment/Study (No need to submit your answers):

1. **Why we need to convert the character columns in the Iris dataset using mutate_if before training the model.**

   Neural networks need numbers, not text. So, we convert character data to factor (category) format that can be used in training.

2. **What does the parameter hidden = c(4, 2) represent in the neuralnet function?**

   This sets the structure of the neural network. It means 2 hidden layers: one with 4 neurons and one with 2 neurons.

3. **From the confusion matrix output, identify which species is most accurately predicted.**

   Look at the confusion matrix. The species with the highest correct predictions (diagonal) is predicted the best.

4. **How can we improve the neural network's performance further (in terms of accuracy or generalization)?**

   You can tune the hidden layers, train with more data, normalize the data, or try other machine learning methods.

5. **Interpret the F-score result for each class. what does it tell us about model reliability?**

   F-score shows how balanced precision and recall are. A higher F-score means the model is more reliable for that class.