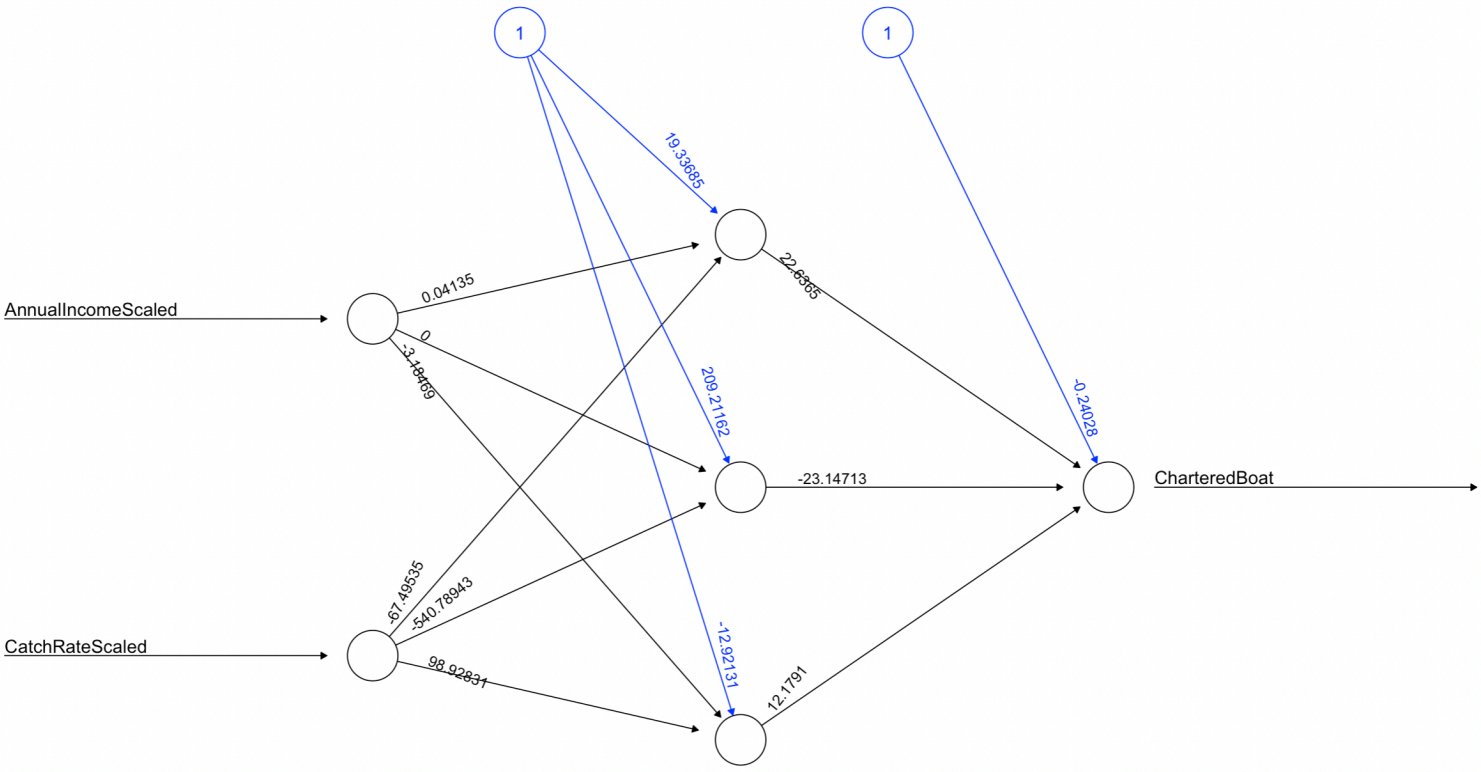
1. From R:
    1. All code from your R script (Code should be presented single-spaced in a fixed-width font. Adjust the font size so that no lines of code extend to the next line in the document)

```
# YUEH-TING WU
# MIS 545 Section 02
# Lab12WuY.R
# In this R programming, import a csv file and generate a neural
network model
# to make a prediction

# Install tidyverse and neuralnet packages
# install.packages("tidyverse")
# install.packages("neuralnet")

# Load the tidyverse and nerualnet libraries
library(tidyverse)
library(neuralnet)

# Set the working directory to Lab12 folder
setwd("~/MIS 545/Lab12")

# Read FishingCharter.csv into a tibble called fishingCharter
fishingCharter <- read_csv(file = "FishingCharter.csv",
                           col_names = TRUE,
                           col_types = "lnn")

# Display fishingCharter in the console
print(fishingCharter)

# Display the structure of fishingCharter in the console
str(fishingCharter)

# Display the summary of fishingCharter in the console
summary(fishingCharter)

# Scale the CatchRate and AnnualIncome feature from 0 to 1
fishingCharter <- fishingCharter %>%
  mutate(CatchRateScaled = (CatchRate - min(CatchRate)) /
           (max(CatchRate) - min(CatchRate)))
fishingCharter <- fishingCharter %>%
  mutate(AnnualIncomeScaled = (AnnualIncome - min(AnnualIncome))
/
           (max(AnnualIncome) - min(AnnualIncome)))
```

```
# Randomly split the dataset into fishingCharterTraining (75% of
records) and
# fishingCharterTesting (25% of records) using 591 as the random
seed
set.seed(591)
sampleSet <- sample(nrow(fishingCharter),
                    round(nrow(fishingCharter) * 0.75))

fishingCharterTraining <- fishingCharter[sampleSet, ]
fishingCharterTesting <- fishingCharter[-sampleSet, ]

# Generate the neural network model to predict
CharteredBoat(dependent variable)
# using AnnualIncomeScaled and CatchRateScaled (independent
variables)
fishingCharterNeuralNet <- neuralnet(
  formula = CharteredBoat ~ AnnualIncomeScaled +
CatchRateScaled,
  data = fishingCharterTraining,
  hidden = 3,
  act.fct = "logistic",
  linear.output = FALSE
)

# Display the neural network numeric results
print(fishingCharterNeuralNet$result.matrix)

# Visualize the neural network
plot(fishingCharterNeuralNet)

# Use fishingCharterNeuralNet to generate probabilities on the
# fishingCharterTesting data set and store it in
fishingCharterProbability
fishingCharterProbability <- compute(fishingCharterNeuralNet,
                                     fishingCharterTesting)

# Display the probabilities from the testing dataset on the
console
print(fishingCharterProbability$net.result)

# Convert probability predictions into 0/1 predictions and store
this into
# fishingCharterPrediction
fishingCharterPrediction <-
  ifelse(fishingCharterProbability$net.result > 0.5, 1, 0)
```

```
# Display the 0/1 predictions on the console
print(fishingCharterPrediction)

# Evaluate the model by forming a confusion matrix
fishingCharterConfusionMatrix <-
table(fishingCharterTesting$CharteredBoat,
                                        fishingCharterPrediction)

# Display the confusion matrix on the console
print(fishingCharterConfusionMatrix)

# Calculate the model predictive accuracy
fishingCharterPredictiveAccuracy <-
sum(diag(fishingCharterConfusionMatrix)) /
  nrow(fishingCharterTesting)

# Display the predictive accuracy on the console
print(fishingCharterPredictiveAccuracy)
```

2. A screenshot of your neural network visualization



2. Answer the following question in a sentence: What is meant by the number of "steps" in the neural network visualization?

It means that neural networks require it to converge on a solution.

3. Answer the following question in a sentence: What are the disadvantages in using a neural network to build a supervised model for this context?

To build a supervised model needs labeled data and its dataset need to contain both input and output variables, but a neural network model don’t have labeled data and it just contains input variables.