**Build a Neural Network model for 50\_startups data to predict profit.**

**Inferences from the Data Set:**

Data Set talks about the profit of the 50-startups and dependent variable is continuous with respect to around 4 Independent variables & 50 observations.

4 Independent variables are

R.D.Spend

Administration

Marketing.Spend

State

Along in 4 Independent variables one variable is categorical data i.e; states. We have to create the dummy variables for that variable

Measures of central tendency explains about mean, median and mode of the data set.

Measures of dispersion explains about the standard deviation variation and range of the data set.

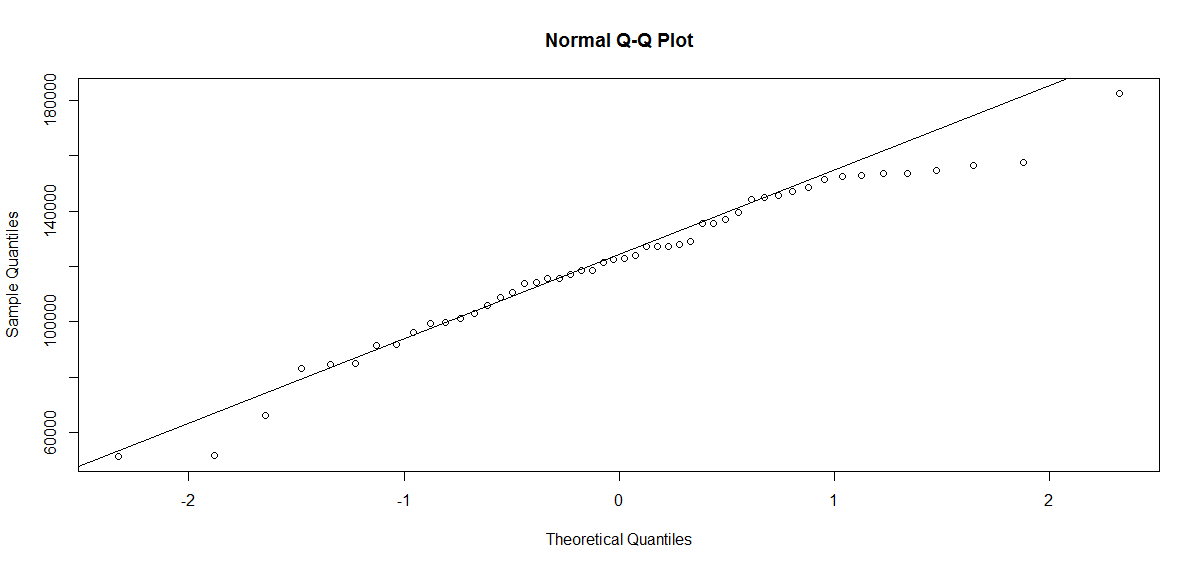
Third moment business decision and Fourth moment business decision explains about the skewness and kurtosis of the data set.

R.D.Spend of the data shows as positive skewness i.e; data skewed to the right and kurtosis is the degree of peakness of distribution.

**Probability distributions of variables:**

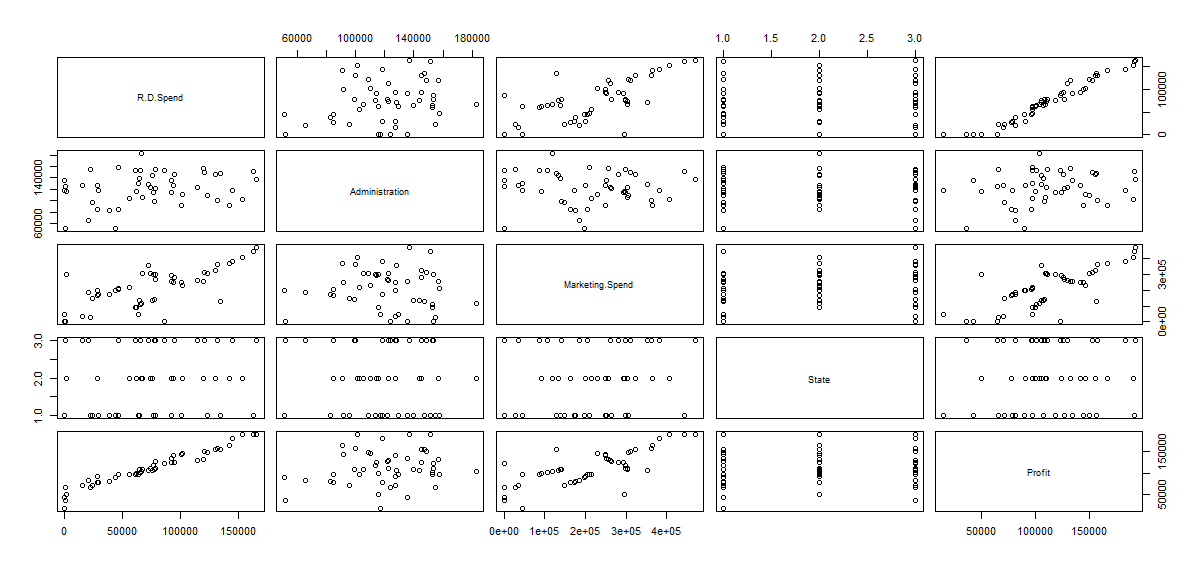
Normal Quantile-Quantile Plot: talks about the data is normally distributed or not, if not have to apply transformation to get the data normally distributed.

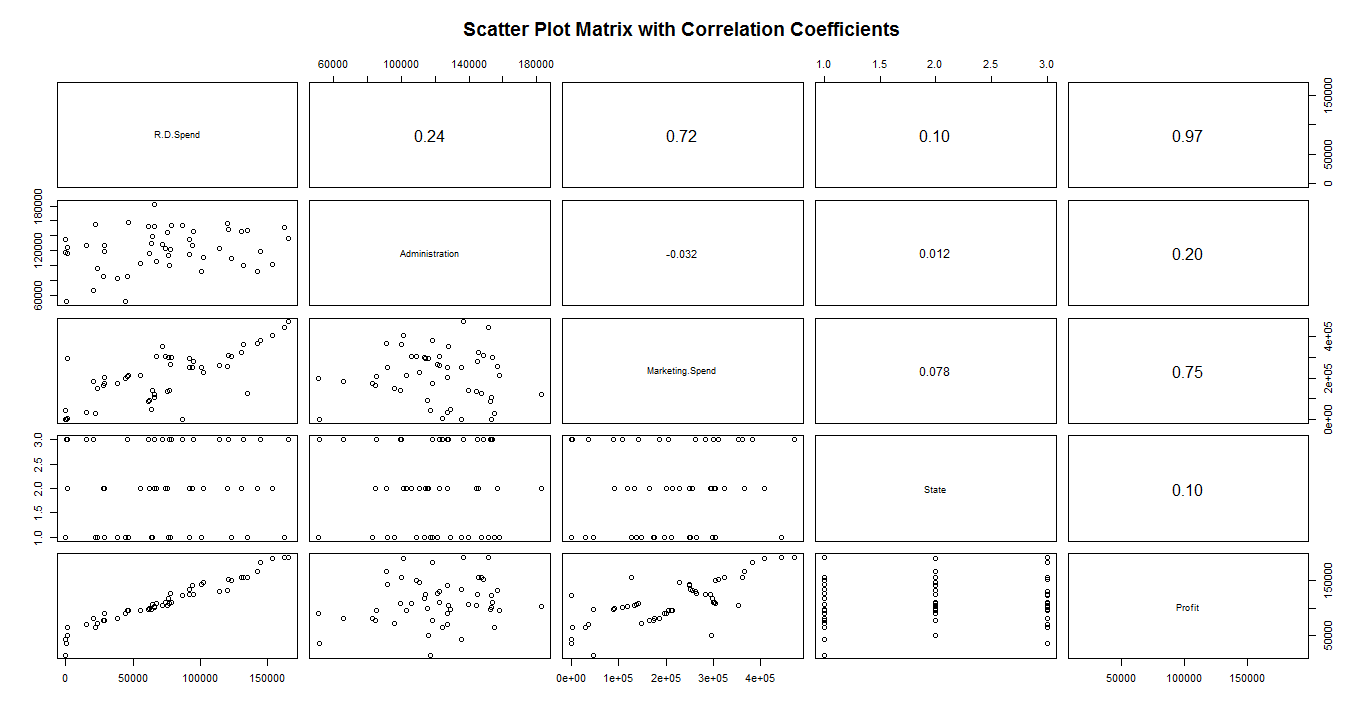
The data is normally distributed for **R.D.Spend**



**scatter plot:**

pairs(startup) # Scatter plot for all pairs of variables





The above scatter plot explains about the relationship between the input and output variables.

Relation between R.D.Spend and profit there is strong positive correlation we can observe.

And also observe that R.D.Spend and Marketing.Spend have strong positive correlation and Administration, State have low correlation to output variables.

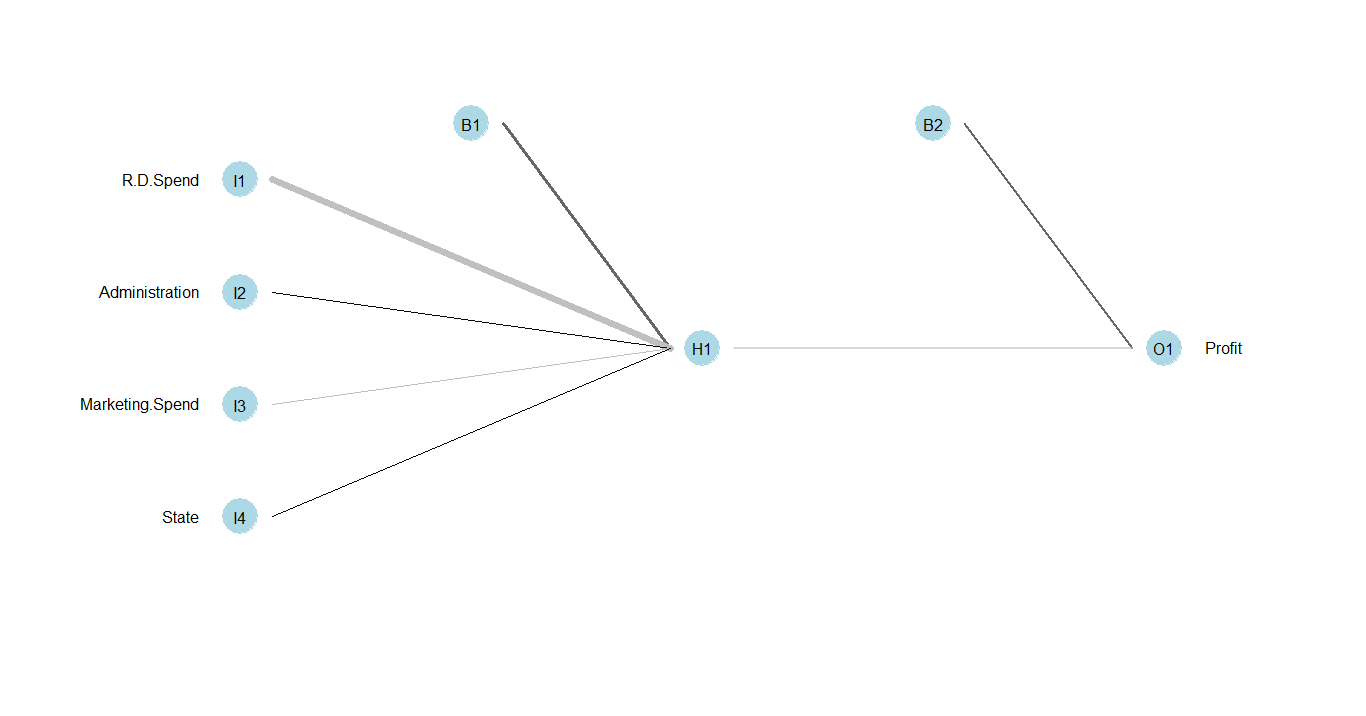
**# Data Partition**

Split the data into train and test data i.e; 70% and 30%

**# The neural network model**

Build the neural network model on the training dataset by loading the library called **neuralnet.**

And we can see the structure of the dataset by using the model.



**# Evaluating model performance**

Evaluate the model using the startups of the model and test data we can get the **model\_results** using the library **compute**

We can predict the model using the model\_results

And we can see the accuracy of the model using the correlation between predicted profit to actual profit of test data.

**# Predicted profit Vs Actual profit of test data.**

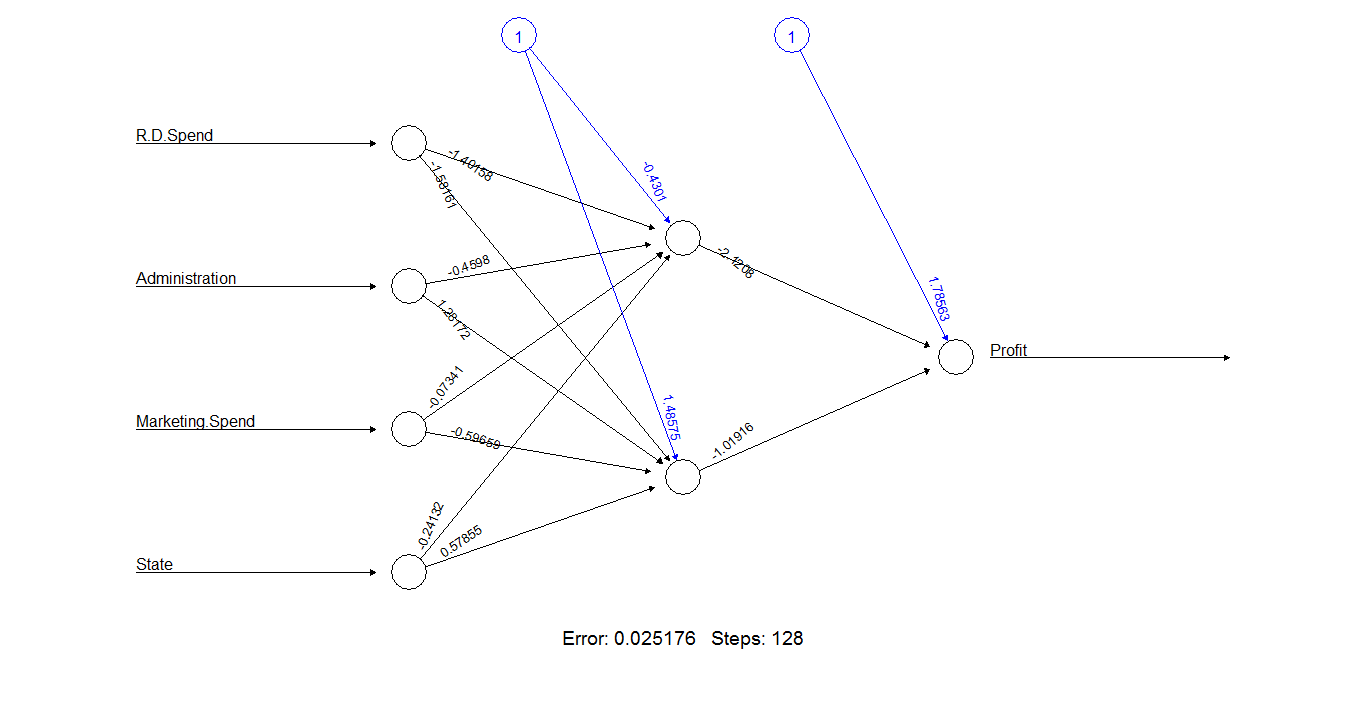
cor(predicted\_profit,startups\_test$Profit)

**#0.9556348**

**# Improve the model performance :**

We can improve the model performance by adding the **hidden layers** to that model.

Below diagram shows the hidden layers of the model, adding of 2-hidden layers.



Repeat the above steps and we get the model\_results and predicted profit ,

So we get the accuracy of the model by applying correlation between predicted profit to actual profit of test data.

**# Predicted profit Vs Actual profit of test data.**

cor(predicted\_Profit2,startups\_test$Profit)

**#0.9639338**

**Final result:**

Observing correlation between predicted profit to actual profit of test data before and after adding hidden layers, it is increased observing the model performance increased.

Error has reduced and training steps had been increased as the number of neurons under hidden layer are increased.