title: "Feed Foward Neural network"

author: "Jashaina Thomas"

date: "3/2/2020"

**### Stock Price Forecasting Using Time Series Analysis, Machine Learning and single layer neural network Models**

A single hidden layer neural network is the simplest neural networks form. In this single hidden layer form there is only one layer of input nodes that send weighted inputs to a subsequent layer of receiving nodes. This nnetar function in the forecast package fits a single hidden layer neural network model to a timeseries. The function model approach is to use lagged values of the time series as input data, reaching to a non-linear autoregressive model.

```{r }

library('QuantTools')

library('forecast')

Boeing <- get\_yahoo\_data( 'BA', '2018-01-01', '2019-12-01' )

Boeing[,"Stock"] <- "Boeing"

#Boeing

#Hidden layers creation

alpha <- 1.5^(-10)

hn <- length(Boeing$close)/(alpha\*(length(Boeing$close)+30))

#Fitting nnetar

lambda <- BoxCox.lambda(Boeing$close)

dnn\_pred <- nnetar(Boeing$close, size= hn, lambda = lambda)

#Fitting nnetar

dnn\_forecast <- forecast(dnn\_pred, h= 30, PI = TRUE)

plot(dnn\_forecast)

```



Plot the model test prediction in a black line with the real train set stock close price in red dashed line.

```{r }

plot(dnn\_forecast)

lines(Boeing$close,col="red",lty = 2)

```

