1) **Walk me through your project**
   As we know that ML models can be used to make predictions. We can also use such models to make stock market predictions

- In this project we built a model based on **Artificial Neural Networks** to successfully predict the stock prices and compare it with the actual prices of a particular firm. As far as data is concerned, we used **multi-dimensional data** of five companies listed on the **National Stock Exchange (NSE) of India** from various sectors for the last six months. The basic of stock market prediction is to predict the future stock using the market statistics of the past few years. We used Artificial Neural Network (ANN), as it is the one of the most powerful tools to predict and analyze data. Then to train the network faster, we reduce the multi-dimensionality of the data by using **Principal Component Analysis (PCA)**. Principal Component Analysis (PCA) is used to reduce the data dimension because a large dataset needs more time to train in a neural network. Then we implement ANN to prepare the data set.After specifying these experimental parameters, we implemented the Backpropagation. We can test our network using existing data and how well it can predict using different plotting and using various diagrams.We calculate the error rate and how much data can be predicted in proximity to the original data.
- So the key steps involved in the project were

Extraction of data from NSE archive of five companies

Pre-processing of data - applying PCA to reduce the dimensionality

Training of ANN models with the data

Testing of the trained model

- **What is ANN?**

  An ANN is based on a collection of connected units or nodes called artificial neurons, which loosely model the neurons in a biological brain. Each connection, like the synapses in a biological brain, can transmit a signal to other neurons. It is perfect for finding out the unknown relation among different data variables and recognizing complex patterns.

  The workings of ANN are extremely similar to those of biological neural networks, although they are not identical. ANN algorithm accepts only numeric and structured data.

  Convolutional Neural Networks (CNN) and Recursive Neural Networks (RNN) are used to accept unstructured and non-numeric data forms such as Image, Text, and Speech.

There are three layers in the network architecture: the input layer, the hidden layer (more than one), and the output layer. Because of the

numerous layers are sometimes referred to as the MLP (Multi-Layer Perceptron).

It is possible to think of the hidden layer as a “distillation layer,” which extracts some of the most relevant patterns from the inputs and sends them on to the next layer for further analysis. It accelerates and improves the efficiency of the network by recognizing just the most important information from the inputs and discarding the redundant information.

3. The activation function is important for two reasons: first, it allows you to turn on your computer.

- This model captures the presence of non-linear relationships between the inputs.
- It contributes to the conversion of the input into a more usable output.
- Finding the “optimal values of W — weights” that minimize prediction error is critical to building a successful model. The “backpropagation algorithm” does this by converting ANN into a learning algorithm by learning from mistakes.
- 5. The optimization approach uses a “gradient descent” technique to quantify prediction errors. To find the optimum value for W, small adjustments in W are tried, and the impact on prediction errors is examined. Finally, those W values are chosen as ideal since further W changes do not reduce mistakes.

What type of data?

We use the National Stock Exchange, India, as our data source and select five companies from different categories and collect their last six months’ data.

- **Data is extracted from Prowess dx of Centre for Monitoring Indian economies (CMIE).**
- **We include the Stocks Trading data of NSE of current Sensex companies**
- **We have included the daily stocks trading for past 6 months**

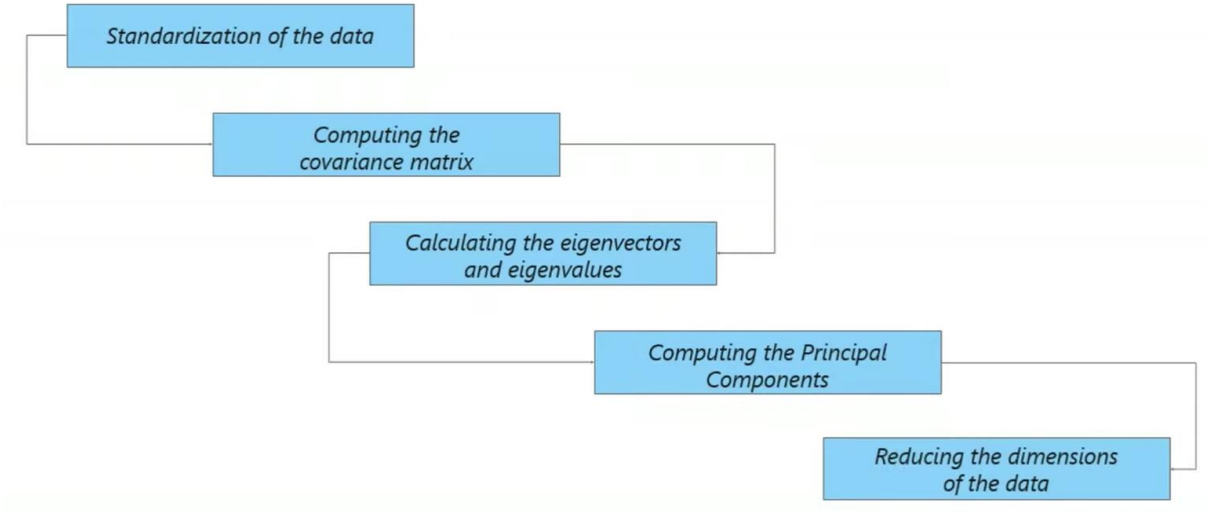
- **Data set includes several variables like *opening price, high price, trade quantity etc.***

  https://docs.google.com/spreadsheets/d/1M5_W7K4tut_v0w2Zzu3guWj1iDeocXMJ1bwCs340kNM/edit?usp=sharing

What is pca?

Principal Component Analysis (PCA) is a dimensionality reduction method that is used to minimize the dimension of large datasets.

This is done by transforming several variables into a smaller one which yet contains a lot of facts in the large set.

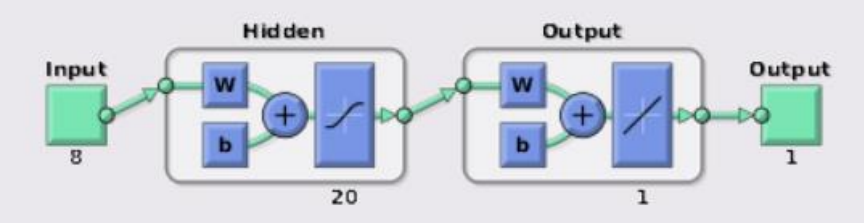


How we trained the model?

- After PCA, we get reduced data which is used as the input of ANN.
- The available data is divided into two sets: training set, and testing set.
- Then we specify the parameters of our Neural Network.
- After specification of the parameters, we move to implementation of our Neural Network.
- Initially, we assign random values to the **hyperparameters**(like weights and the biases).
- Then, in the **feed forward step**, input patterns are propagated through the network one by one, and actual outputs are calculated.
- Then, comparing actual and target outputs, gives us the error.
- Weight updates take place through the **back propagation** to minimize the error in the cost function. Thus,

Updated Weight = weight(old) + [(learning rate)*(output error)*output(neurons i)* output(neurons i+1)*(1 - Output(neurons i+1))]

- When all the data is passed through the model and hyper parameters are set accordingly, this is called an **epoch**. Multiple epochs are performed to maximize the efficiency of our model.



Algorithms

Data Division:	Random	(dividerand)
Training:	Bayesian Regularization	(trainbr)
Performance:	Mean Squared Error	(mse)
Calculations:	MEX	

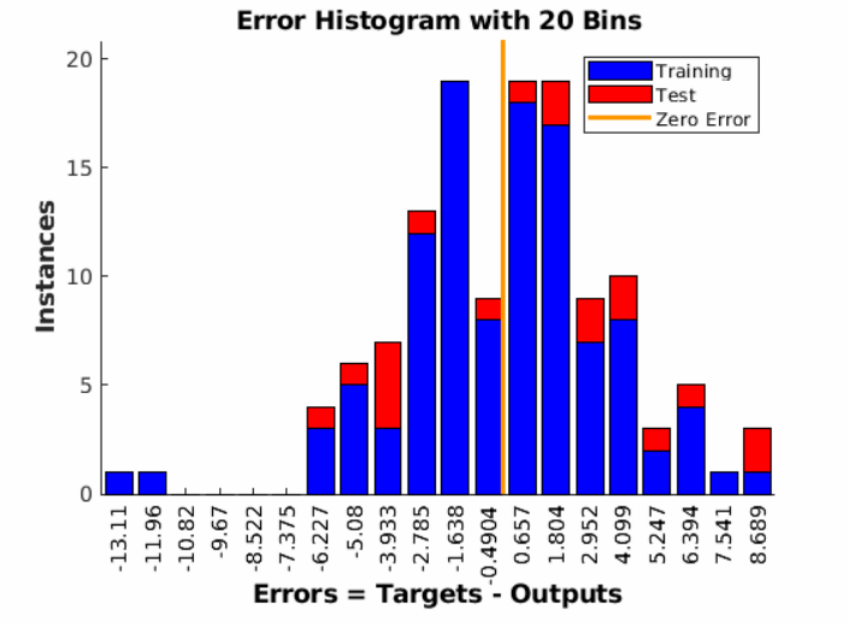
Progress

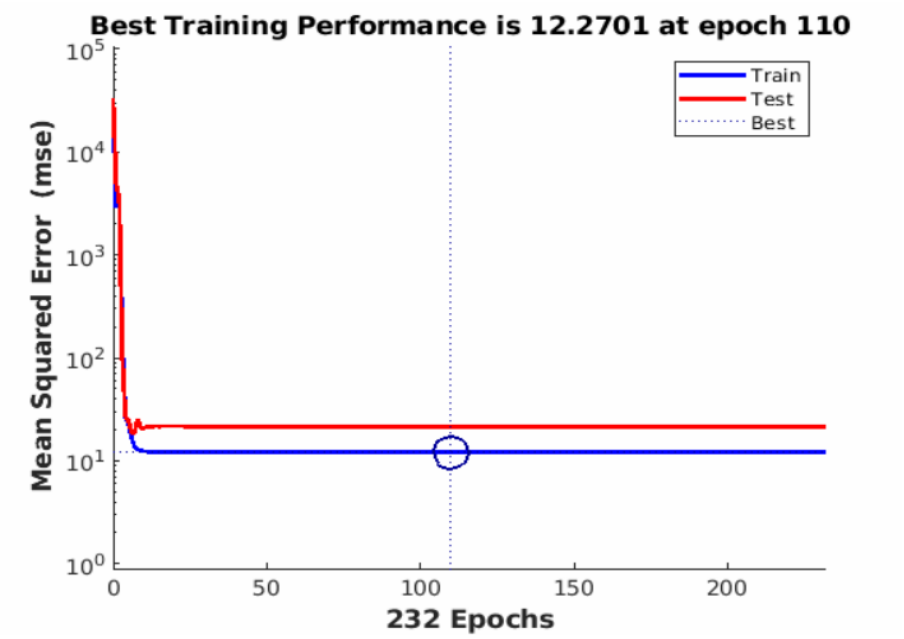
Epoch:	0	232 iterations	1000
Time:		0:00:02	
Performance:	2.82e+04	12.3	0.00
Gradient:	1.13e+05	1.98	1.00e-07
Mu:	0.00500	5.00e+10	1.00e+10
Effective # Param:	161	13.7	0.00
Sum Squared Param:	153	3.08	0.00

Plots

Performance	(plotperform)
Training State	(plottrainstate)
Error Histogram	(ploterrhist)
Regression	(plotregression)
Fit	(plotfit)

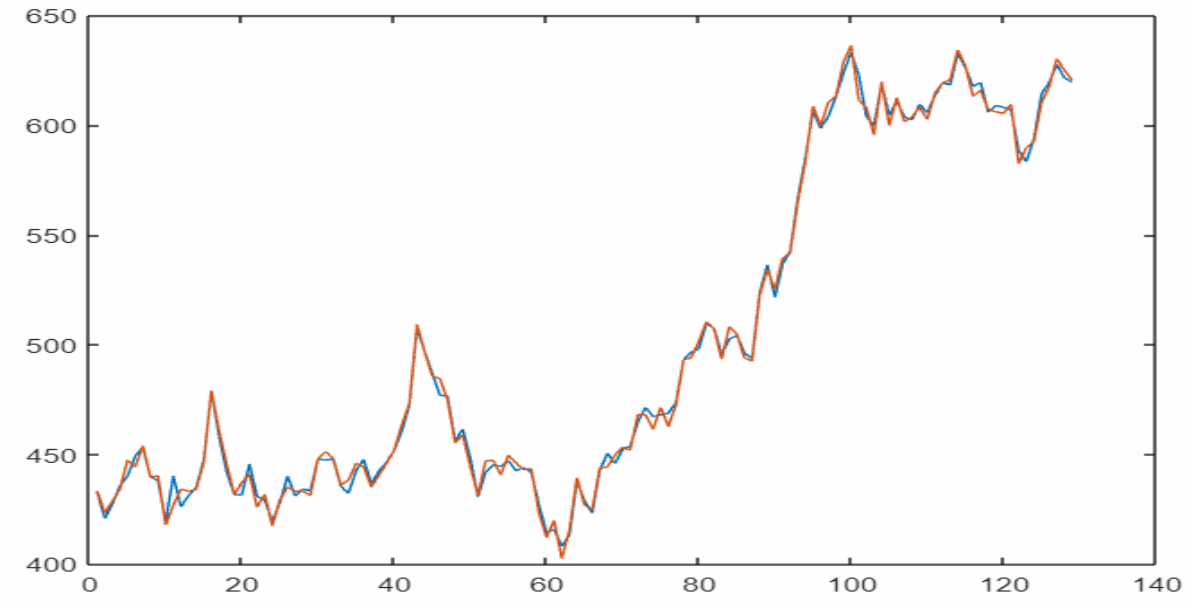
Plot Interval: 50 epochs





Testing

- **Earlier, we have divided the data into two parts, the second part would now be used for testing of our model.**
- **The resulting data is then used for plotting the graph against the actual data.**
- **Then error is calculated.**
- **A number of epochs are performed to reduce this error.**
- **Also, while choosing the epoch we should try to avoid overfitting.**



- **In our suggested system, we will develop a model to anticipate a particular firm's stock rate by training their preceding data in a neural network.**
- **First, we use Principal Component Analysis (PCA) to reduce the data dimension, which helps train our system faster. We can get the most influential features of data after the reduction of the data dimension.**
- **Artificial Neural Network, which is one of the finest neural network methods, will be used, which can reduce the error between the actual output and the desired output by using gradient descent.**
- **Since it will be quite strenuous to predict with 100% accuracy, the performance is not expected to be satisfactory every time.**