1st Possible Figure

a)

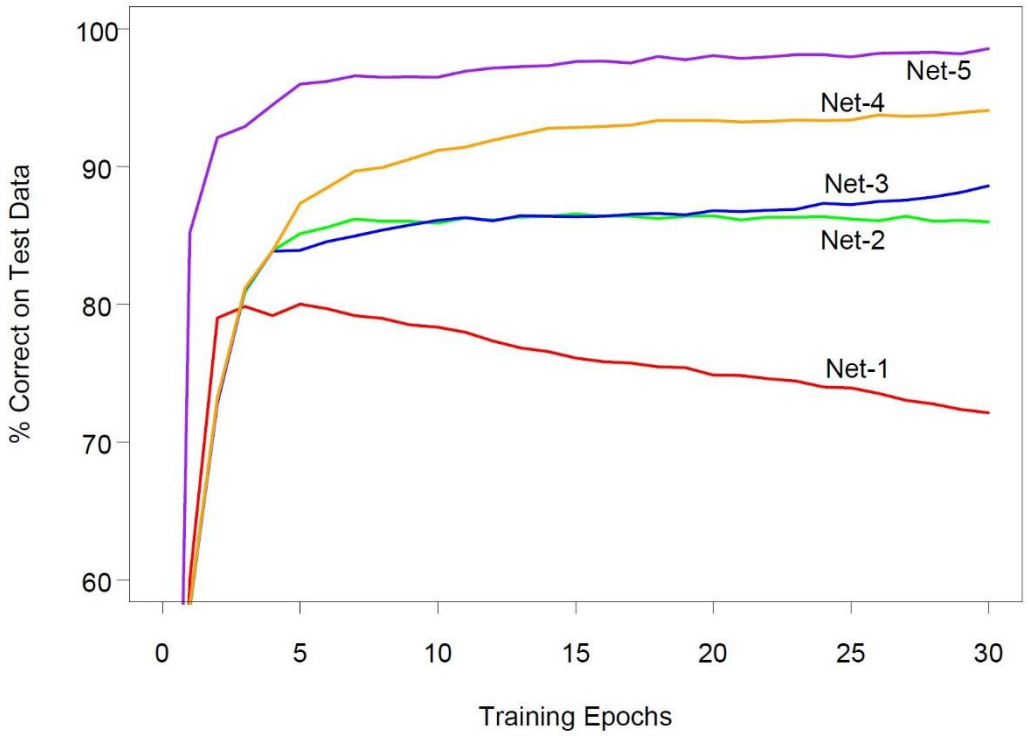

Fig. 1: Test Performance Curves, as a function of the number of training epochs for 5 different networks of Table-1 applied to the ZIP Code Data.

Ref: Hastie, et al. Elements of Statistical Learning, Figure 11.11.

Table-01

Network Index	Network Architecture
Net-1	Single Layer Network
Net-2	Two Layer Network
Net-3	Locally Connected
Net-4	Constrained Network 1
Net-5	Constrained Network 2

b)

Inputs to the Model: 16x16 grayscale image (256 pixel values)

Outputs of the Model: Ten classes (Each for 0-9 digits)

In this classification project, there would be two phase, one is training phase and the other one would be testing phase. For each of the neural network architecture, the size of epoch in training would be varied and the corresponding testing error would be calculated. Based on this data, the figure would be plotted for 5 different architecture.

Function: The network will learn a function that predicts a value to represent the estimated probability that an image x has digit class k, for k=1, 2, 3, …., 9

c) The problem is described as the "classification of handwritten digits". More specifically, classifying a digit image based on the features of the image pixels.
Data Source URL: https://web.stanford.edu/~hastie/ElemStatLearn/data.html
Training Observation: 7291, Test Observation: 2007
Each observation contains a 16x16 (256 pixel) input feature.
Each observation contains an output id (0-9) of a digit.

d) **Pseudocode:**

1. Input: 16x16 Grayscale image (256 pixel)
   Output: Class Label (0-9)
2. Set the parameters of the neural networks (such as weights, number of hidden layers, neuron connections, loss function, etc.)
3. Set the training epoch number
4. Train the network with training dataset
5. Test the network with test dataset
6. Calculate the test Error

In this project, I will use PyTorch library of Python. In that library, I can set the parameters for each of the neural networks. To generate the figure, I will trace the test error with respect to the epoch size. In this project, I will learn about the neural network architecture and also get hands on experience on the popular python library "PyTorch"

2nd Possible Figure

a)

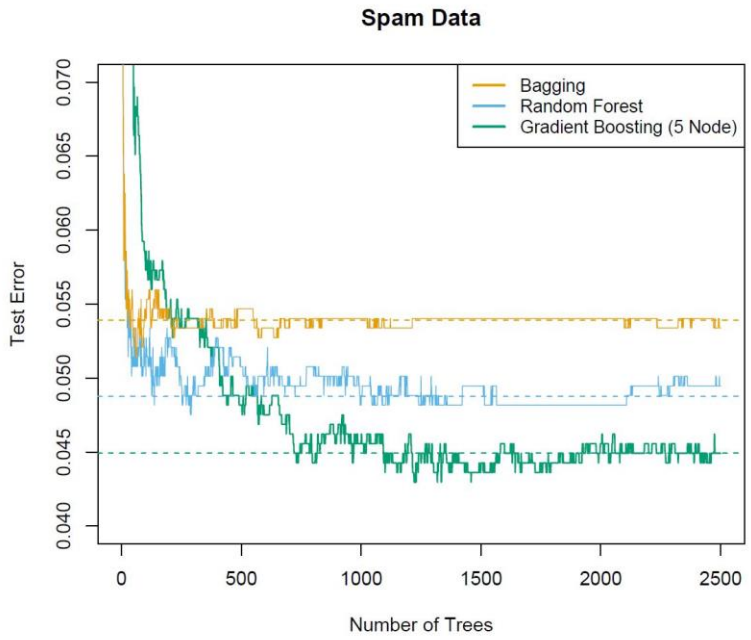

Fig. 2: Bagging, random forest, and gradient boosting, applied to the spam data. For boosting, 5-node trees were used, and the number of trees were chosen by 10-fold cross-validation (2500 trees).

Ref: Hastie, et al. Elements of Statistical Learning, Figure 15.1.

b) Inputs to the Model: 57 attributes

Output of the Model: binary (1 or 0)(Spam or Not)

Each of the three algorithms will be executed to find the test error for different tree sizes. Based on that result the above figure will be drawn.

Function: Each model will learn a function that relates between the attributes and the output (spam or not).

c) Predicting the spam emails based on different features of the emails.

Data Source URL: https://web.stanford.edu/~hastie/ElemStatLearn/data.html
Number of Observations: 4601
Number of features: 57
Output Labels: Binary (0 or 1)

d) **Pseudocode:**

1. Input: 57 features
   Output: Binary (0 or 1)
2. Set the value of n for cross validation
3. Set the parameters for respective algorithms (such as Bagging, Random Forest, Gradient Boosting)
4. Train the model for an algorithm
5. Calculate the test Error and trace the tree size

In this project, I will use the libraries such as Pandas, Numpy, Scikit-learn of python.
To generate the figure, I will trace the test error for the corresponding tree size for each of the algorithms. I will learn three different algorithms such as Random Forest, Bagging and Gradient Boosting in this project. At the same time, I will learn the popular machine learning libraries of python.