ML BOOTCAMP

PROJECT REPORT

MATTURTY sri LAKSHMI

IIT (ISM) DHANBAD

**PROBLEM STATEMENT:** The problem statement is to implement  *Linear Regression, Polynomial Regression, Logistic Regression, K-neural network, n-layer Neural Network and K-means Clustering* by using python and libraries like *numpy, pandas and matplotlib.*

**IMPLEMENTATION OF LINEAR REGRSSION:**

The provided training dataset has 50000 training examples and 20 features. We should try to fix a linear function such that it predicts the output of the each training example with minimum error. The mean normalisation is done to bring all features to same scale and to avoid one feature dominating other. The training data is then spilt into training set and validation set in the ratio of 4:1. Later, input set is concatenated with array of ones to incorporate the bias term. Theta array is used to represent the weights associated with a particular feature. Mean squared cost function is used to calculate the error. The gradient of cost function is computed to reduce the cost. The value of theta is updated using gradient descent function for each iteration. Gradient Descent is a function of learning rate and number of iterations. The value of learning rate is set at different values like “0.01, 0.03, 0.1, 0.3”. Finally, I fixed the value of learning rate as “0.3”.

At learning rate = 0.3 the cost varied with number of iterations in the following way:

Iteration= 0

Cost= 0.0071478575207311

Iteration= 1000

Cost= 1.9606086622693364e-06

Iteration= 2000

Cost= 5.998006296883324e-10

Iteration= 3000

Cost= 7.476199299546506e-13

Iteration= 4000

Cost= 5.534570448324694e-13

Iteration= 5000

Cost= 5.533890982677158e-13

Iteration= 6000

Cost= 5.533890702535662e-13

Iteration= 7000

Cost= 5.533890702378356e-13

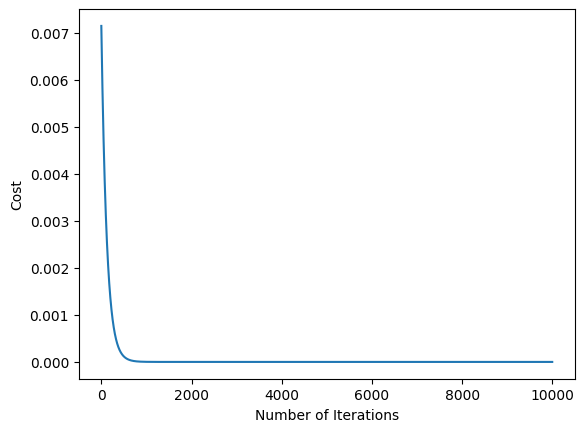
Iteration= 8000

Cost= 5.533890702378105e-13

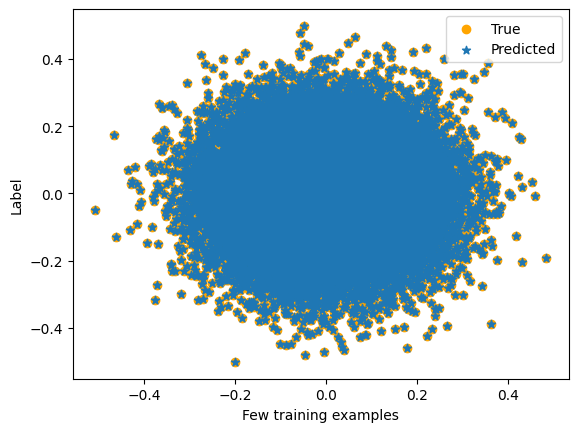
Iteration= 9000

Cost= 5.5338907023772e-13

**GRAPH 1: Number of iteration vs Cost**



**Graph 2 : The Graph is showing True values and Predicted Values**

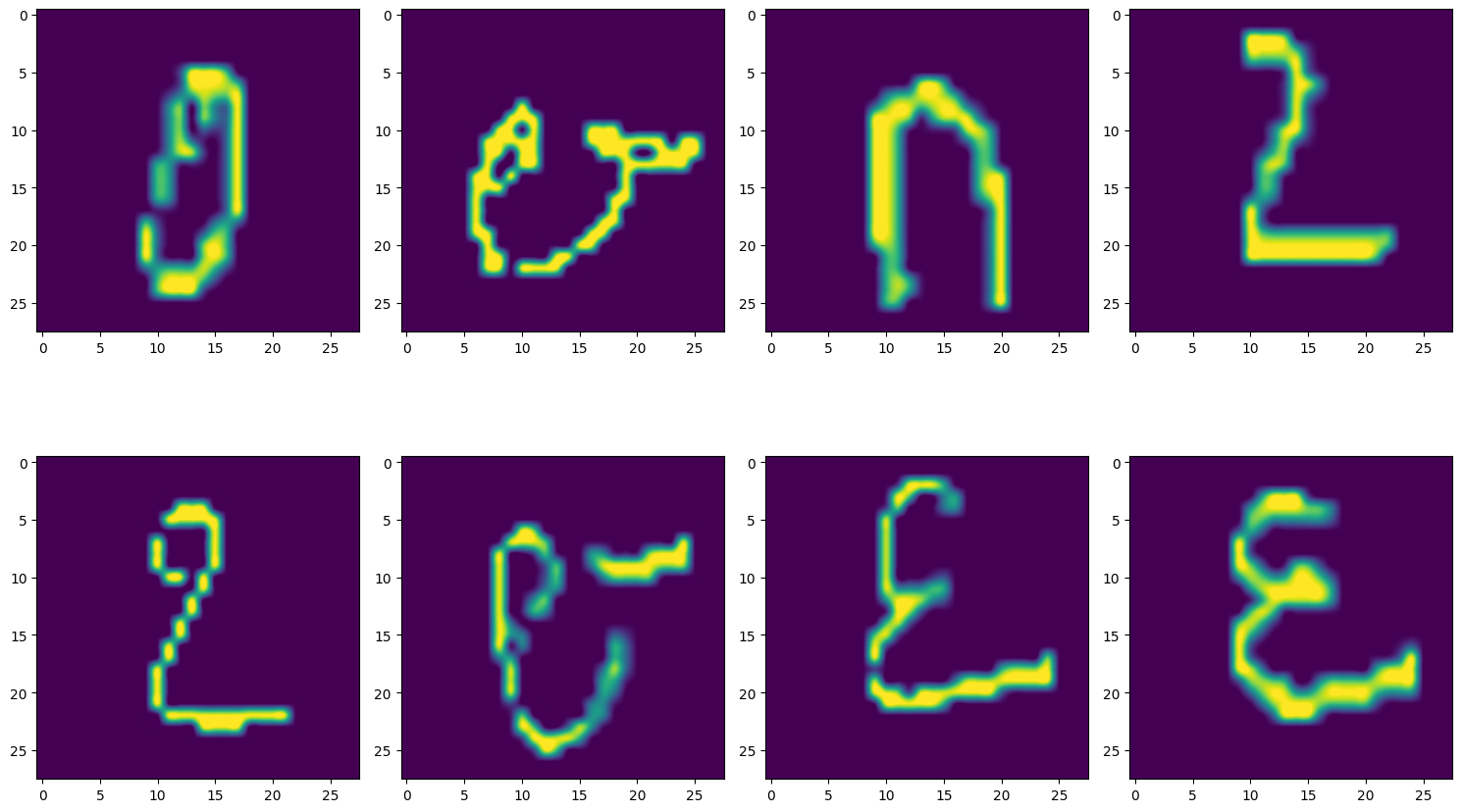


**IMPLEMENTATION OF POLYNOMIAL REGRESSION:**

The provided training dataset has 50000 training examples and 20 features. We should try to fix a linear function such that it predicts the output of the each training example with minimum error. The mean normalisation is done to bring all features to same scale and to avoid one feature dominating other. The training data is then spilt into training set and validation set in the ratio of 4:1.

**IMPLEMENTATION OF LOGISTICS REGRSSION.**

The training dataset has 30000 training examples. We need to classify the pictures. There are the first eight pictures



These are the hand written letters

The values of training set are normalised by dividing each value with the maximum value. The training is divided into two parts training set and validation set. Sigmoid function is defined. Instead of cost function loss function is used. I have defined Gradient Descent function like in the case of linear regression.

At learning rate = 0.3 the cost array is

Cost= [2.88811325e-05 2.88811325e-05 2.88811325e-05 ... 2.88811325e-05

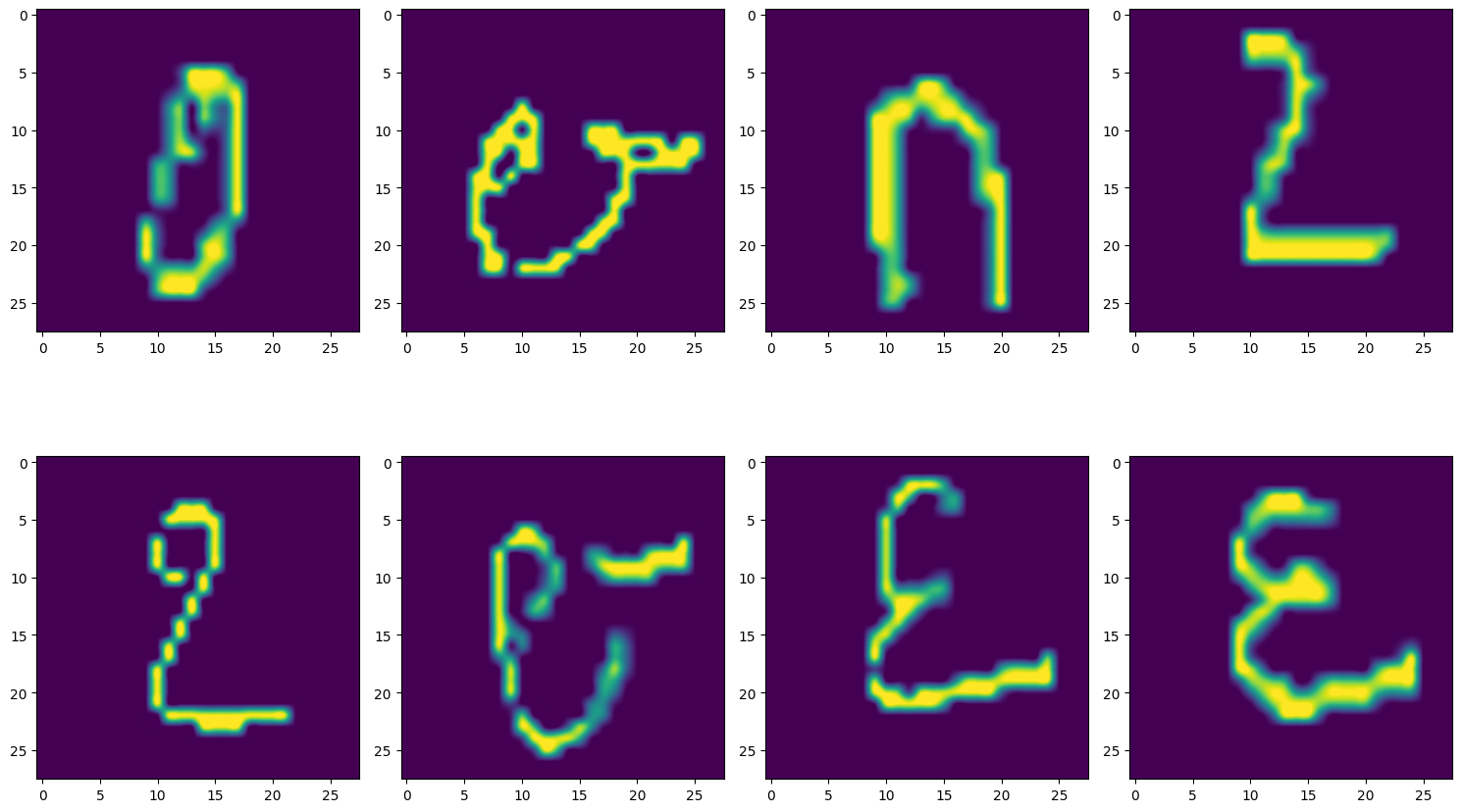
2.88811325e-05 2.88811325e-05]

The number of iterations are10000.

The accuracy is 0.8483333333333334

**IMPLEMENTATION OF K-NEURAL NETWORK:**

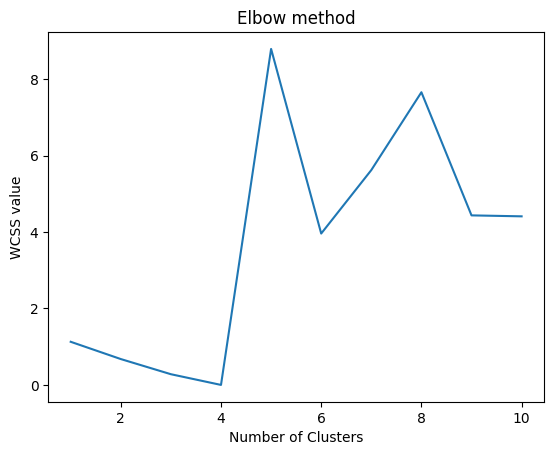
The training dataset has 30000 training examples. We need to classify the pictures. There are the first eight pictures.



The values of training set are normalised by dividing each value with the maximum value. The training is divided into two parts training set and validation set. The Euclidian distance is calculated between the each example of validation set and training set. The k nearest neighbours are found. It is sent to the classification based on the k nearest neighbours. I tried with different values of k. The accuracy I got when k=17 is 0.503. The accuracy I got when k=19 is 0.5175.

**IMPLEMENTATION OF K-MEANS CLUSTERING**

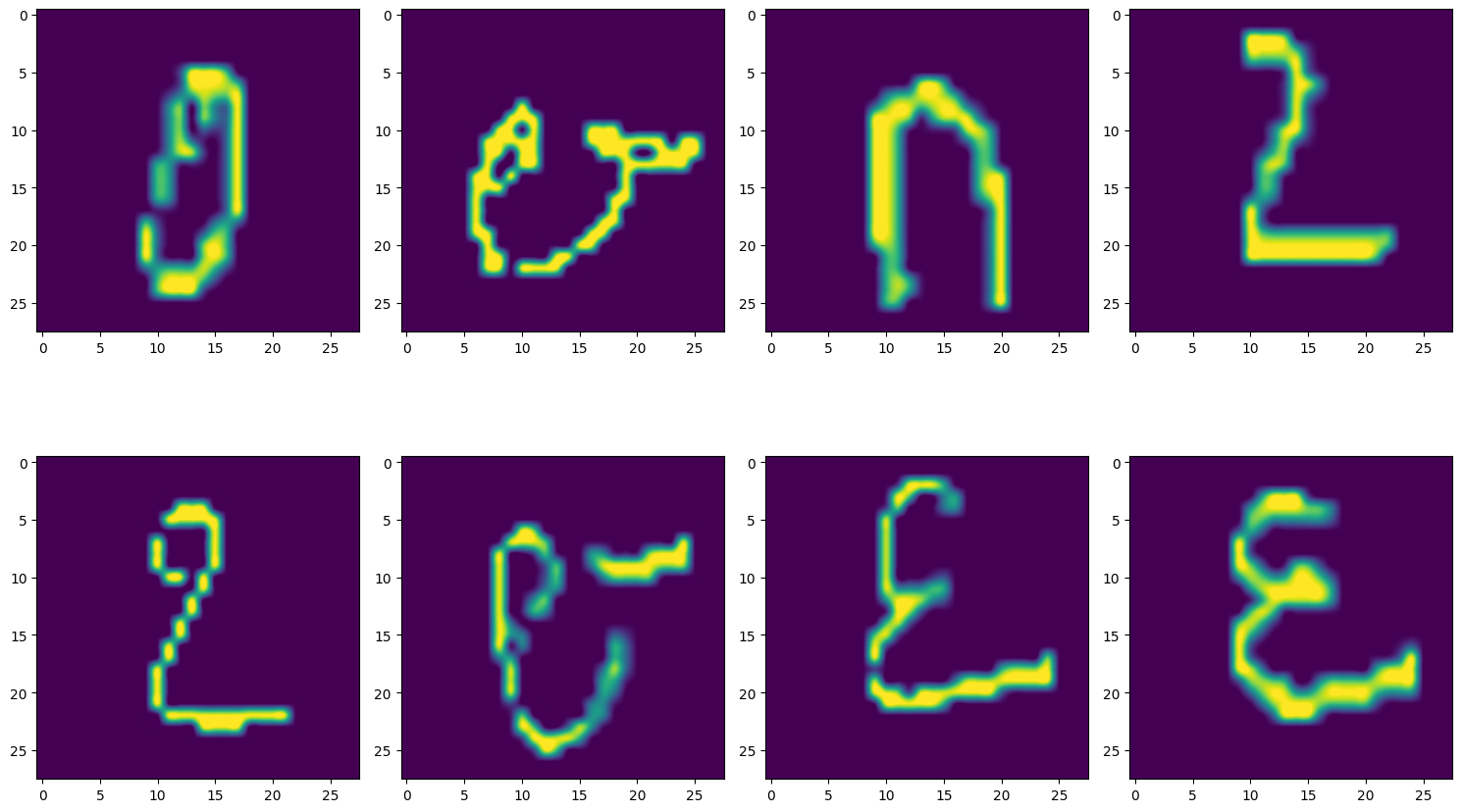
The dataset is having 12 features and 178 examples. The data is normalised using mean normalisation function. We have to classify the given data into k clusters. At first, random k points are taken as centroids. Later, euclidian distance is calculated between data point and centroid. The data point is assigned to cluster having closest centroid. The centroids are updated to the new centroids by calculating mean. The value of k is to decided by using the function Within-Cluster Sum of Squares(wcss). At the correct value of k, wcss is low.



From, the graph we can say that the best value of k is 4.

**IMPLEMENTATION OF n-layer NEURAL NETWORK:**

The values of training set are normalised by dividing each value with the maximum value.



The values of training set are normalised by dividing each value with the maximum value. As it is binary classification I have used sigmoid function as activation function. Forward propagation is used to find values of weights and biases where as backward propagation is used to reduce the cost and assign new values to weights and biases.