WOC 4.0
Report

Aryan Rajani
21JE0172

Overview

This project has 6 machine learning models - Linear Regression, Polynomial Regression, Logistic Regression, K- Nearest Neighbors (KNN), K- Means Clustering and
Neural Networks.

Linear Regression

The dataset contained various training examples and values of the various features corresponding to the training examples.

First I began with taking a small subset of the data and tried working on it, but failed in doing so. So I started off with one feature only and the labels, succeeded in that, and then broadened my scope to 2 features and started working with the vectorised approach towards the calculation of the hypothesis and cost. And finally worked with the entire dataset.

I then worked using the weights matrix obtained and applied it onto the testing dataset and got the root mean squared error.

Polynomial Regression

Using the knowledge gained in linear regression, I trained my model. The only added part was adding features which was done using np.multiply function where all possible combinations were taken up till nth degree. I implemented L2 regularization in my model to prevent overfitting as well.

Logistic Regression

It is used to model the probability of discrete outcomes and give the predicted Y values. I used the sigmoid function for this model. I used the logic behind one versus all classification to alter the Y labels array to make a matrix of 26 x 88800 (number of training examples). Trained the model using the logistic cost function. The final piece was to use the weights matrix on the testing examples dataset and used it calculate the required accuracy

Accuracy = 75.567%

K- Nearest Neighbors

It is used to predict the correct class for the test data. The model first calculates the distance of each testing data with training datas. Then it checks the k nearest points to predict. The predicted Y is compared with the given Y values to get accuracy. For the test data I have taken only the first 1000 testing examples

Accuracy = 79.1%

K- Means Clustering

This model groups the given data into clusters based on the similarity between them. I used the logic of repeating the testing examples and finding the difference between the testing example and coordinates of the various clusters, and further used it to calculate the distance of the testing example and cluster centers. Then take the minimum of the distance and assign the appropriate cluster.

Neural Network

This model tries to predict outputs for the given inputs. It consists of 1 input layer, 1 hidden layer and 1 output layer. We initialized thetas and biases, then used forward and back propagation to get optimum values. Forward and backward propagation powers neural networks.

Train Accuracy = 53.108%
Test Accuracy = 51.635%