Project For:

CyberLabs

Project Name:

Machine Learning Bootcamp

# PROJECT REPORT

by Sanskriti Agrawal

- Name: Sanskriti Agrawal
- University: IIT (ISM), Dhanbad (826004)
- Major: Mathematics and Computing
- Year of Study: First Year

## Description

The purpose of this project was to make our own Machine Learning Library. I implemented 6 machine learning algorithms like Linear and Polynomial Regression, Logistic Regression, KNN, K-Means Clustering and Neural Networks.

## Technology Stack

I used **Google Colaboratory** to write my code. The language used is Python. Various python libraries like **Numpy, Pandas and Matplotlib** are also used.

## Features

1. **Linear regression:**

   It is used to predict Y values using features(X1, X2...). This is what I did to make it work.

   I made a function first to train my model and got theta values. Then theta was used to get Y values for my testing data. Then I calculated Root Mean Squared Error by comparing it with the given testing Y values.

   Root Mean Squared Error = 1.829

2. **Polynomial Regression:**

   It is used to predict Y values using given features and their nth degree. I used my knowledge of linear regression to write its function. I started with degree n=2 and tried it till n=4. I got maximum accuracy for n=3. Then I made a function to train my model and get theta values. Then predicted Y for testing data using theta. Subsequently, Root Mean Squared Error was calculated.

   Root Mean Squared Error = 2.299

3. **Logistic Regression:**

   It is used to model the probability of discrete outcomes and tell that the given data belongs to which category. It uses sigmoid function to model. My system was crashing again and again. So, I used only 62000 training examples. First, we had to do one hot encoding on test and train labels. I made a function to train my data and get theta values. Then multiply that theta matrix with the testing data and then took sigmoid of it to get hypothesis values. Then the value with maximum probability was taken to be the predicted value. Then I compared it with the given labels to get accuracy.

   For accuracy, I equated given and predicted values and counted the no. of zeros. This gave me the number of correctly predicted values.

   Accuracy = 69.446%

4. **K Nearest Neighbours:**

   We are given some testing data, which classifies coordinates into groups identified by an attribute. Now given an unclassified point, we can assign it to a group by observing its nearest neighbors.

   It is used to predict the correct class for the test data. I used only 62000 training examples and 5000 testing examples because my system was not supporting more than this and also the computational time was really high. My model first calculates the distance of each testing data with training data. Then it checks the k nearest points to predict class. The predicted class is compared with the given values to get accuracy. For accuracy, I used the same idea I used in logistic regression.

   Accuracy = 83.46%

5. **K-Means Clustering:**

   We are given a data set of items with certain features. The algorithm will group them into k groups or clusters based on the similarity between them. It uses only 60000 examples.

   Firstly, I plotted a graph of WCSS versus k (WCSS is the sum of squared distances between each point and centroid of that cluster).

   It showed a dip for k=30. So, I made 30 clusters for the dataset. First, we took random points to be the center of our clusters. Then, calculate the distance of each center with other points. The point having minimum distance with a center was assigned to the cluster belonging to that center. Then, new centers were made by taking the mean of points of that cluster. Then this process is repeated till I get the same value for centers.

   Then I gave each cluster a label by using the given values for our data Y. Though it is an Unsupervised learning algorithm and we are not supposed to calculate accuracy. Still, I wanted to check my code. So, I calculated accuracy. You can ignore that if you want.

6. **Neural Network**:

   It involves neurons, connections, weights, biases, propagation function, and a learning rule. The learning rule modifies the weights and biases of the variables in the network.

   This model tries to predict outputs for the given inputs. It consists of 1 input layer, 1 hidden layer and 1 output layer. My system was crashing again and again. So, I used only 60000 training examples. I initialized thetas and biases, then use Forward and Back propagation to get optimum values. Those values of thetas and biases were used to get predicted values for testing data. Then I calculated the accuracy using the same idea I used in Logistic Regression.

   Accuracy = 67.885%

## Week Wise Timeline

| Time Frame | Milestones |
|---|---|
| Week 1 | -Implemented Linear Regression and Polynomial Regression |
| Week 2 | -Implemented Logistic Regression<br>-Learned about KNN and K-Means Clustering |
| Week 3 (Mid-Evaluation) | -Corrected my errors in the previous models after discussing with my mentors in Mid-Evaluation |
| Week 4 | -Implemented KNN and K- Means Clustering |
| Week 5 | -Implemented Neural Networks<br>-Made report |

## About Me

Name - Sanskriti Agrawal

Place - Kasganj, Uttar Pradesh (207123)

Branch - Mathematics and Computing

Institution - Indian Institute of Technology (Indian School of Mines), Dhanbad

E-mail - 21je0830@iitism.ac.in

Linked in - https://www.linkedin.com/in/sanskriti-agrawal-116694215/