Introduction to Machine Learning (Spring 2020)
Programming Assignment 1 – Group 41
Report

Vaibhav Chhajed (vchhajed)
Yeshwanth Badineni (ybadinen)
Hemant Koti (hemantko)

Part 1 - Linear Regression

Problem 1 - Linear Regression with Direct Minimization

Q1] Calculate and report the RMSE for training and test data for two cases: first, without using an intercept (or bias) term, and second with using an intercept. Which one is better?

A1] Results

- `RMSE without intercept on train data - 138.20`
- `RMSE with intercept on train data - 46.77`
- `RMSE without intercept on test data - 326.76`
- `RMSE with intercept on test data - 60.89`

Observation

- We know that smaller the RMSE better are the results
- Therefore, RMSE with intercept on train and test data are better than RMSE without intercept on train and test data

Problem 2 - Linear Regression with Gradient Descent

Q2] Using testOLERegression, calculate and report the RMSE for training and test data after gradient descent-based learning. Compare with the RMSE after direct minimization. Which one is better?

A2] Results

- `Gradient Descent Linear Regression RMSE on train data - 48.08`
- `Gradient Descent Linear Regression RMSE on test data - 54.76`

Observation

- Gradient Descent Linear Regression RMSE on train data is slightly greater than Direct minimization.
- Gradient Descent Linear Regression RMSE on test data is slightly lesser than Direct minimization.
- Therefore, Gradient Descent Linear Regression works better on test data compared to Direct minimization.

Part II - Linear Classifiers

Problem 3 - Perceptron using Gradient Descent

Q3] Train the perceptron model by calling the scipy.optimize.minimize method and use the evaluateLinearModel to calculate and report the accuracy for the training and test data?

A3] The evaluateLinearModel function takes the inputs X, y, w and returns an n*1 matrix which has all the predicted values. To compute the accuracy, we compare the predicted y's and given y's and divide them by the input size.

- `Perceptron Accuracy on train data - 0.84 (84 %)`

Vaibhav Chhajed (vchhajed)
Yeshwanth Badineni (ybadinen)
Hemant Koti (hemantko)

- `Perceptron Accuracy on test data - 0.84 (84 %)`

Problem 4 - Logistic Regression Using Newton's Method

Q4] Train the logistic regression model by calling the scipy.optimize.minimize method, and use the evaluateLinearModel to calculate and report the accuracy for the training and test data.

A4] The logistic regression model is computed using three functions – logisticObjVal, logisticGradient, and logisticHessian. The logisticObjVal computes the logistic loss for the given data set. The logistic Gradient computes the gradient vector of logistic loss for the given data set. The logistic Hessian computes the Hessian matrix of logistic loss for the given data set. The accuracy measure after training the data set is listed below.

- `Logistic Regression Accuracy on train data - 0.83 (83 %)`
- `Logistic Regression Accuracy on test data - 0.86 (86 %)`

Problem 5 - Support Vector Machines Using Gradient Descent

Q5] Train the SVM model by calling the trainSGDSVM method for 200 iterations (set learning rate parameter η to 0.01). Use the evaluateLinearModel to calculate and report the accuracy for the training and test data.

A5] The SVM model calls the trainSGDSVM method which learns the optimal weight w. It runs for 200 iterations with a learning rate set to 0.01 thereby computing the updated weight vector in every step.

- `SVM Accuracy on train data - 0.84`
- `SVM Accuracy on test data - 0.86`

Problem 6 - Plotting decision boundaries

Q6.1] Use the results for test data to determine which classifier is the most accurate?
A6.1] The SVM model has the most accurate classification on the testing data as shown in the image below. However, the trainSGDSVM function in SVM takes random samples each time to compute the updated weight vector thereby leading to different results when ran each time on the given data set. In contrast, the logistic regression model is more stable and has good accuracy on testing data.

Q6.2] Plot the decision boundaries learned by each classifier using the provided plotDecisionBoundary function which takes the learned weight vector, which was one of the parameters. Study the three boundaries and provide your insights.
A6.2] Based on the results listed above the **perceptron** model gives the same accuracy on training and testing data.

The **Logistic Regression** gives more accuracy on testing data when compared to training data. The logistic model has stable results when running every time as compared to perceptron and SVM.

The SVM model results in better classification compared to logistic and perceptron, however, the trainSGDSVM function uses random samples which gives different results when running each time.

Introduction to Machine Learning (Spring 2020)
Programming Assignment 1 – Group 41
Report

Vaibhav Chhajed (vchhajed)
Yeshwanth Badineni (ybadinen)
Hemant Koti (hemantko)