CSE574 Assignment 1

PA Group 11
Deepti Bharadwaj(50363309)
Kaumudi Moholkar (50388592)
Alexandar Ma (37136560)

March 5, 2021

Report 1:

Q) 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?

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

RMSE with intercept on train data is better than the one without intercept as the value is smaller. However, from observation, the RMSE with intercept on test data is a little greater than RMSE on train data which shows that the train data model is slightly overfit.

Report 2:

Q) 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?

The results of problem 2 are as seen below:

Gradient Descent Linear Regression RMSE on train data - 47.95
Gradient Descent Linear Regression RMSE on test data - 55.50

In general, the gradient descent is computationally faster than the RMSE. By observations, the accuracy of the gradient descent based learning is better than the RMSE approach on the training and testing data because the model fits well on training data and does not over-fit on test data.

Report 3:

Q) 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.

Perceptron Accuracy on train data - 83.00
Perceptron Accuracy on test data - 83.00

Report 4:

Q) 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.

Logistic Regression Accuracy on train data - 83.00
Logistic Regression Accuracy on test data - 85.00

Report 5:

Q) Train the SVM model by calling the trainSGDSVM method for 200 iterations.Use the evaluateLinearModel to calculate and report the accuracy for the training and test data.

After setting the learning rate η to 0.01 The results obtained from running the SVM model containing its accuracy are as follows:

SVM Accuracy on train data - 86.00

Report 6:

Q) i) Use the results for test data to determine which classifier is the most accurate?

ii) Plot the decision boundaries learnt by each classifier using the provided plot Decision Boundary function which takes the learnt weight vector, as one of the parameters. Study the three boundaries and provide your insights.

The boundaries of perceptron and logistic regression misclassify more points than SVM. This difference can be seen reflected in the accuracies of the three models, as SVM has the highest accuracy out of the three models. The reason for this is that in SVM, we explicitly tell the function to fit the data well by separating the points.

Perceptron and Logistic regression work well on the data as long as the model finds the data is properly separated by the decision boundary. The difference in the way they classify lies in how they are calculated perceptron is based on the weights and examples whereas logistic regression is based on a threshold/cutoff value.

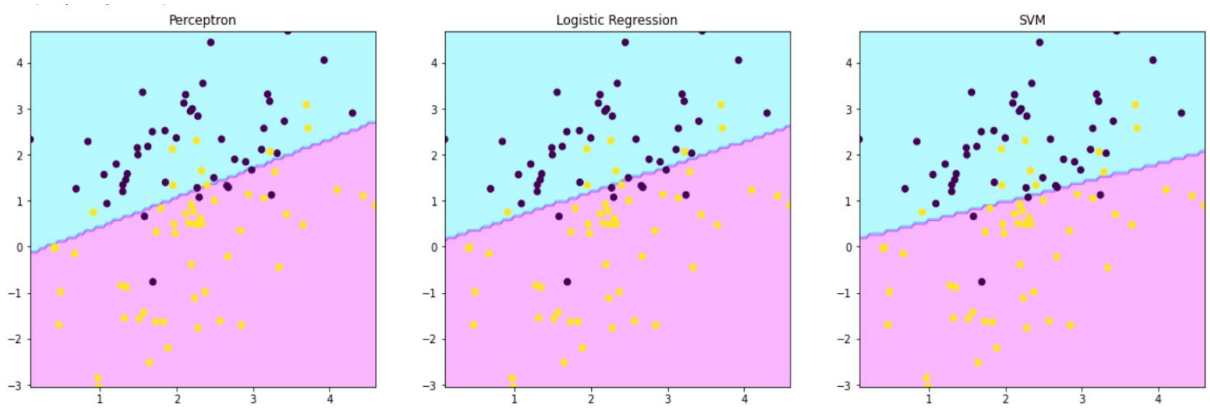

Figure 1. Graphs produced from the plotBoundaries function using training data.

Comments:

The boundary of SVM was decided based on a random sample. This would affect the produced result each run time. In order to observe uniformity a command like “np.random.seed( )” can ensure the same results.