Linear Models for Supervised Learning

Programming Assignment 1

CSE 474: Introduction to Machine Learning

Project Members (Group 26):

Akshay Singh Gehlot (agehlot@buffalo.edu)

Rupali Kotina (rupaliko@buffalo.edu)

Siddharth Suneel Bandhu (sbandhu@buffalo.edu)

Problem 1:

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?

Room Mean Squared Error(RMSE) without using an intercept:

Train Data – 138.20

Test Data – 326.76

Room Mean Squared Error(RMSE) with using an intercept:

Train Data – 46.77

Test Data – 60.89

From the above calculations we can conclude that the RMSE using an intercept is better than the RMSE without using the intercept as it is lower. Because lower RMSE has smaller error which gives us better estimate.

Problem 2:

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?

RMSE after gradient descent based learning:

Train Data – 48.09

Test Data – 54.74

Compared to RMSE after direct minimization, RMSE after gradient descent based learning is better on Test Data, and is worse for Train Data.

Problem 3:

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 both train and test data is 0.84 using the evaluateLinearModel function.

Problem 4:

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 – 0.84

Test Data – 0.86

Problem 5:

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

SVM Accuracy on:

Train Data – 0.87

Test Data – 0.88

SVM accuracy keeps changing on every execution due to the use of randomSample.

Problem 6:

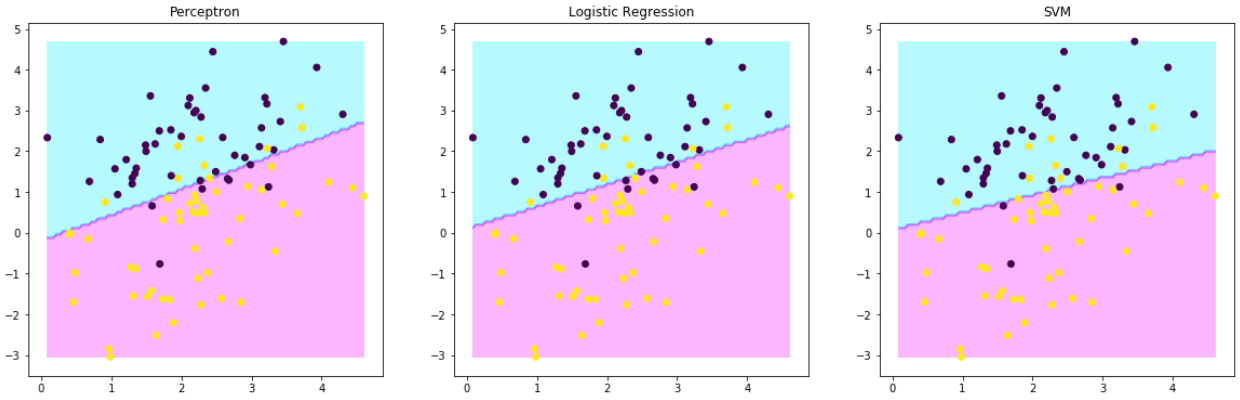
Use the results for test data to determine which classifier is the most accurate?

	Train Data	***Test Data***
Perceprton	0.84	0.84

Logistic Regression	0.84	0.86
Support Vector Machines	0.87	0.88

Thus we can see that for the given data set SVM is the most accurate followed by Logistic Regression While perceptron gives least accurate results.

Plot the decision boundaries learnt by each classifier using the provided plotDecisionBoundary function which takes the learnt weight vector, was one of the parameters. Study the three boundaries and provide your insights.



The learnt accuracies of Perceptron, Logistic Regression, and SVM it can be seen that the accuracy of SVM is higher. It is because the classifier misclassifies approximately three cases in SVM and approximately 7 cases in Perceptron and Logistic Regression. The probability to generalize well to unseen data is increased. SVM has high accuracy and low interpretability while Logistic regression has low accuracy and high interpretability.