CSE 574: Programming Assignment 1 (Group 3)

Salil Dabholkar
UBID: 50321748
salilsan@buffalo.edu

Nihar Patel
UBID: 50321748
nihardil@buffalo.edu

Sahil Kapahi
UBID: 50321748
sahilkap@buffalo.edu

Part I - Linear Regression

1. Linear Regression with Direct Minimization

RMSE with and without intercept

- **On Training Data:**
  - With Intercept: 46.77
  - Without Intercept: 138.20
- **On Test Data:**
  - With Intercept: 60.89
  - Without Intercept: 326.76

Result:

a) As can be seen above, the RMSE for both training and testing data is lower when using an intercept as compared to without intercept.
b) However, the RSME for training data is lower than the test RSME in all the cases.

Explanation:

a) Since the regression line will be forced through y = 0 when all x's are zero without the intercept term, inclusion of intercept doesn't affect the ability to estimate/predict differences in Y per unit change in a independent variables and provides a flexible way to learn the regression curve varying the bias term as required for the best fit. Thus, using an intercept is better way to learn a regression curve.
b) Training RSME is lower than test RSME as the fitted model adapts to the training data and therefore training error will be better optimized.

2. Using Gradient Descent for Linear Regression Learning

RMSE using Gradient Decent

- RMSE on train data: 48.13
- RMSE on test data: 54.70

Result:

a) The RMSE of Gradient decent is less than the RMSE of direct minimization method without intercept.

b) The RMSE of Gradient decent is slightly more than the RMSE of direct minimization method with intercept on training data but somehow lower than the RMSE with intercept on test data.

Explanation:

a) Since including the bias term in gradient descent will allow some flexibility and will not allow the regression curve to force through the zero, it gives better result (i.e. lower RSME) for gradient descent compared to direct minimization without intercept.

b) The RSME in gradient descent is slightly more on training data compared to direct minimization using intercept because the optimum in gradient descent is dependent on the learning rate and the convergence tolerance given to the optimizer. So, there is a possibility that the minimizer may converge slightly before the true optimum and hence this can be a reason for higher RSME.

   Whereas, the lower RSME on test data by gradient descent method compared to direct minimization with intercept can be explained as a result of overfitting in the direct minimization method which may decrease the performance on test data set. Gradient descent thus may lead to better results for not overfitting on the training set.

Part II - Linear Classifiers

3. Perceptron Learning

i. Perceptron Accuracy on train data: 84.00%

ii. Perceptron Accuracy on test data: 84.00%

4. Newton's Method for Logistic Regression

i. Accuracy on train data: 83.00%

ii. Accuracy on test data: 85.00%

5. SVM

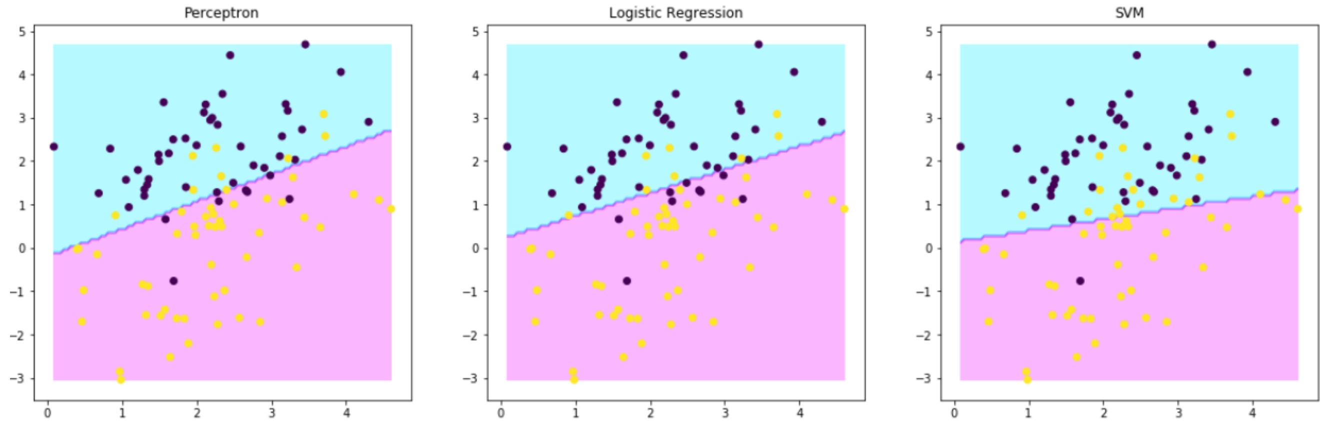
i. Accuracy on train data: 82.00%

ii. Accuracy on test data: 87.00%

6. Comparing Linear Classifiers:

I. Comparing the test data for the classifiers, it can be seen that SVM gives the best results followed by Logistic Regression and Perceptron Learning. Since the SVM tries to work on maximization of the decision boundary, it ensures to prevent overfit and thus improve performance on test data. In Logistic regression, the decision making function is non-linear although the decision boundary is linear.

II. Plotting



The decision boundary learnt for the above case by Perceptron, Logistic and SVM classifier vary slightly. Perceptron learns the boundary using linear objective function which is not the case with Logistic and SVM classifier. Logistic and SVM classifier are slightly more liberal when it comes to learning a boundary to account for better classification of the test data whereas a perceptron sticks to minimizing the error for the training data. Logistic and SVM allow have the ability to perform better on test data in some cases due to this property. As can be seen from the test results, SVM performs better on test data although the boundary it learns makes more error on training data than the perceptron as well as the logistic classifier. This is explained by the fact that SVM tries to minimize the error as well as maximize the boundary. In the similar manner, Logistic classifier makes less error on test data compared to training data when compared to perceptron. This is because Logistic Regression works on the probability model which can better classify test data in some cases though it may not minimize the training error as good as Perceptron. This explains the boundary learnt by all the three classifiers on training data.