CSE 574: Introduction to Machine Learning

Programming Assignment 1:

**Classification and Regression**

Instructor: Dr. Mingchen Gao

Date: March 14, 2018

Submitted by,

Muthuvel Palanisamy,

Person # - 50246815

**Problem 1:** Experiment with Gaussian Discriminators (10 code + 10 report = 20 points)

Implementation**:**

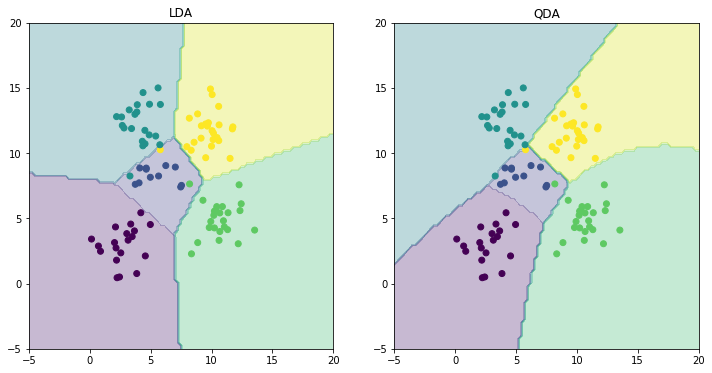
* Used NumPy function (np.mean, np.covariance) to calculate mean and covariance matrices in LDA and QDA
* Dataset: sample\_train

Results and Analysis

* **Accuracy**:

LDA – 97 %

QDA – 97 %



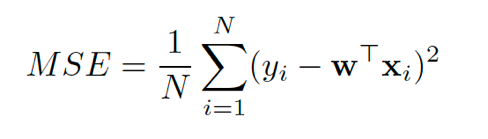
Discriminating Boundaries for LDA and QDA

* As we could see in the above picture, contour function is used to plot the discriminating boundaries of LDA and QDA
* The points with different colors correspond to different classes
* QDA is able to more accurately plot the boundaries for the different classes
* This is because QDA takes into account the covariance of each classes. This makes the boundary of a specific class to be dependent on the input values of only the specific class
* While in LDA since a single covariance matrix is used across all the test data for prediction, the boundary become a more generic model between classes
* In the end, QDA proves to be better in differentiating class boundaries

**Problem 2:** Experiment with Linear Regression (5 code + 5 report = 10 points)

Implementation:

* Performed direct parameter (w) estimation in the learnOLERegression() function
* testOLERegression computes the mean squared error (MSE) of the estimation using the below formula



Results and Analysis

* Mean Square Errors of the implementation:

MSE with intercept: 3707.84

MSE without intercept: 106775.36

* It is obvious that an intercept in the dataset [1,x] has comparatively low mean squared error and is better comparatively

**Problem 3:** Experiment with Ridge Regression (10 code + 10 report = 20 points)

Implementation:

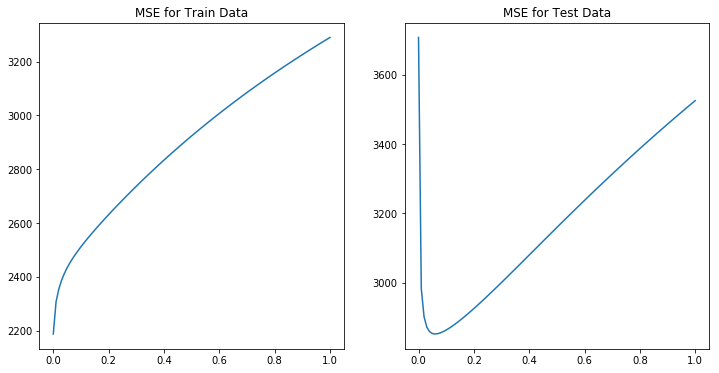
* Implemented parameter estimation for ridge regression by minimizing regularized squared loss

Results and Analysis

* MSEfor train\_data varies from 2150 to 3300 in an increasing manner for lamba = 0.01 to 0.99 (incremented in steps on 0.01)
* MSE for test\_data from 2851 to 3700
* MSE for train data is at it’s lowest (=2150) when lambda is 0.00 (No regularization)
* Hence the **optimal value** **of** **lambda** **for test data**= **0.00**
* MSE for test data is at it’s lowest (=2851) when lambda is 0.06
* Hence the **optimal value** **of** **lambda** **for test data**= **0.06**
* See the graph below for the plot of Error Vs Lamba for training and testing data

Linear Vs Ridge Regression (data with intercept)

* MSE for Linear Regression on test data= 3707
* MSE for Ridge Regression (for optimal lambda = 0.06) on test data= 2851
* We could see that introducing the lamda parameter provided a better model with weight vector that has comparivielty low MSE

****

Comparision of MSE vs Lambda for train and test data in Ridge Regression

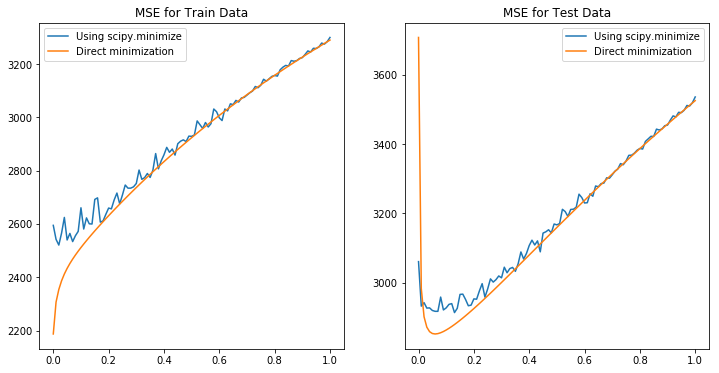
**Problem 4:** Using Gradient Descent for Ridge Regrssion Learning (20 code + 5 report = 25 points)

Implementation

* Uses scipy.optimize.minize function to implement gradient descent for ridge regression on the weights

Results and Analysis

* We could see that gradient descent is close to parameter estimation of weights in Problem 3. The gradient reached the optimum value in a few steps and also able to maintain its value without much deviations or fluctuation after a few steps
* Also, gradient descent could help avoid singularity issues in calculating inverse function during direct estimation of weights



**Problem 5:** Non – Linear Regression (10 code + 5 report = 15 points)

Implementation:

* Implemented parameter estimation for ridge regression by minimizing regularized squared loss

Results and Analysis

**Problem 6:** Interpreting Results (0 code + 10 report = 10 points)

Implementation:

* For every lambda from 0 to 0.99 (incremented in 0.01), non – linear regression in done on data with input values [1, x, x^2, …..]

Results and Analysis

* When lamba = 0

Error in train data

Error in test data

* When lambda = 0.06 (optimal value estimated from Problem 3)

Error in train data

Error in test data