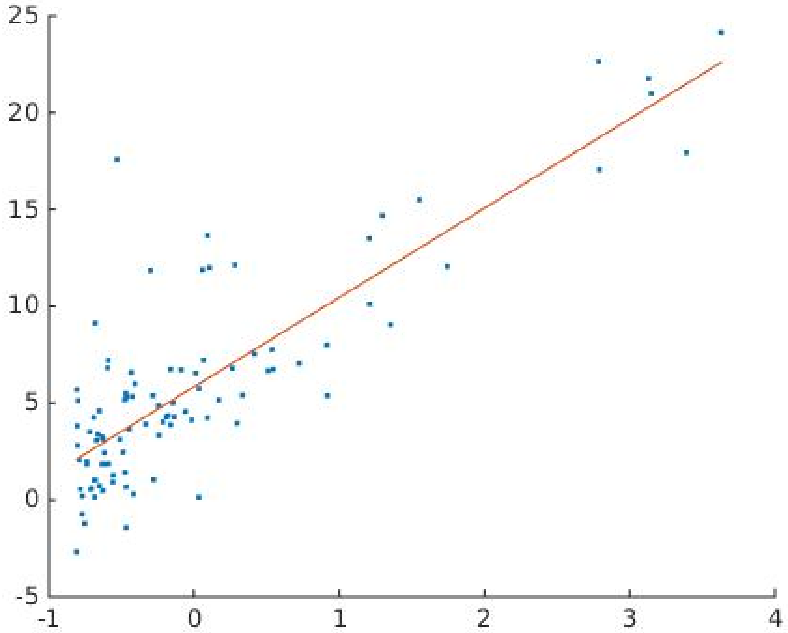
Assignment 1 - Report

Question 1. Implementation of Linear Regression using Batch Gradient Descent algorithm for Error Function minimization

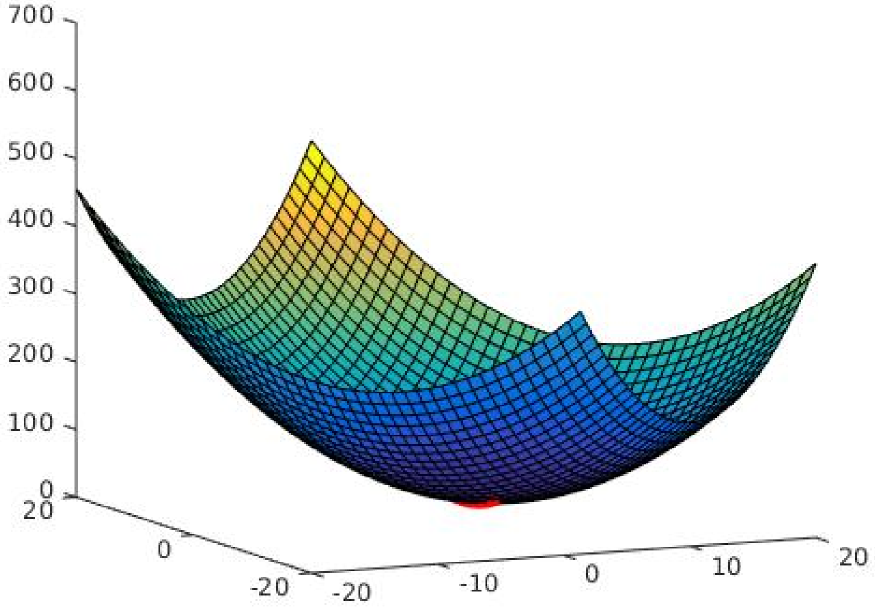
$$J(\theta) = \sum_{i=1}^{m} \frac{1}{2}\big(y^{(i)} - h_\theta\big(x^{(i)}\big)\big)^2 \tag{1}$$

Solution (. a) The Learning rate is set to 0.1 and the theta matrix is reported as below. The Stopping condition used is that the absolute value of difference between successive error functions should be less than 10^{-4} i.e $J(\theta_{old}) - J(\theta) <= \epsilon$ where $\epsilon = 10^{-4}$. The final value is : $\theta = \begin{bmatrix} 5.8392 \\ 4.6169 \end{bmatrix}$

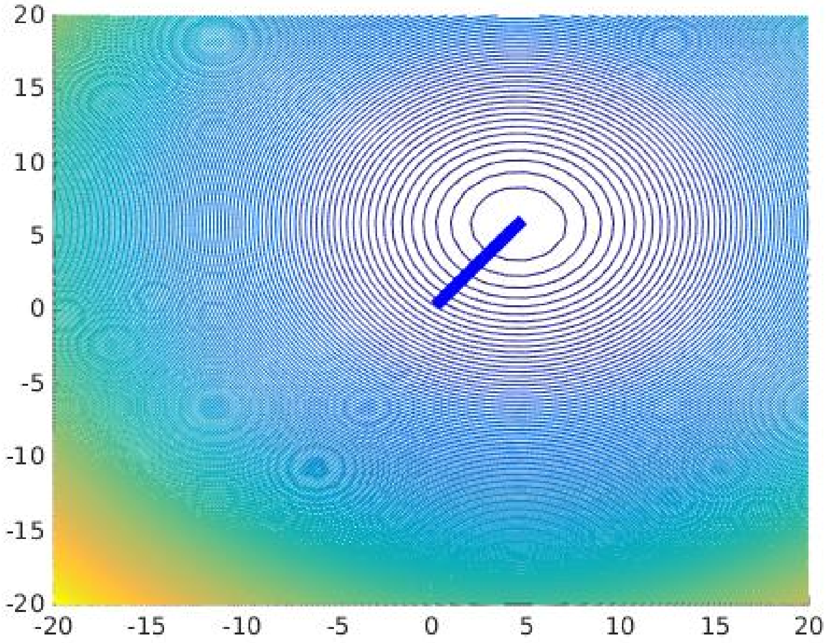
(b) The Hypothesis function is plotted as follows:



(c) The 3-D mesh plotting Error Function $J(\theta)$ on z-axis and θ_0, θ_1 on x-y plane:



(d) The contour plot of Error Function for each iteration:



(e) Part (d)is repeated for different values of 'η' (0.1,0.3,0.9,1.3,2.1,2.5). It is observed that the number of iterations for $\eta = 0.1$ to $\eta = 1.3$ keep reducing. However at $\eta = 2.1$, the algorithm does not converge and θ keep oscillating. Same is the case for $\eta = 2.5$.

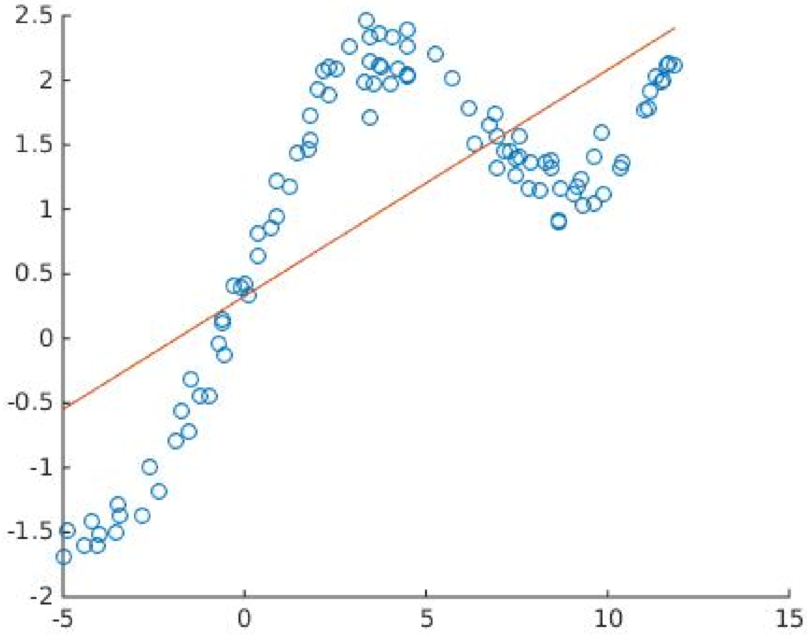
Question 2. Implementation of Locally weighted Linear Regression where the error function is described as follows:

$$J(\theta) = \sum_{i=1}^{m} \frac{1}{2} w^{(i)} \big(y^{(i)} - h_\theta\big(x^{(i)}\big)\big)^2 \tag{2}$$

Solution (. a) Linear Regression using the given normal equation for the solution of θ :

$$\theta = (X^T X)^{-1}(X^T Y) \tag{3}$$

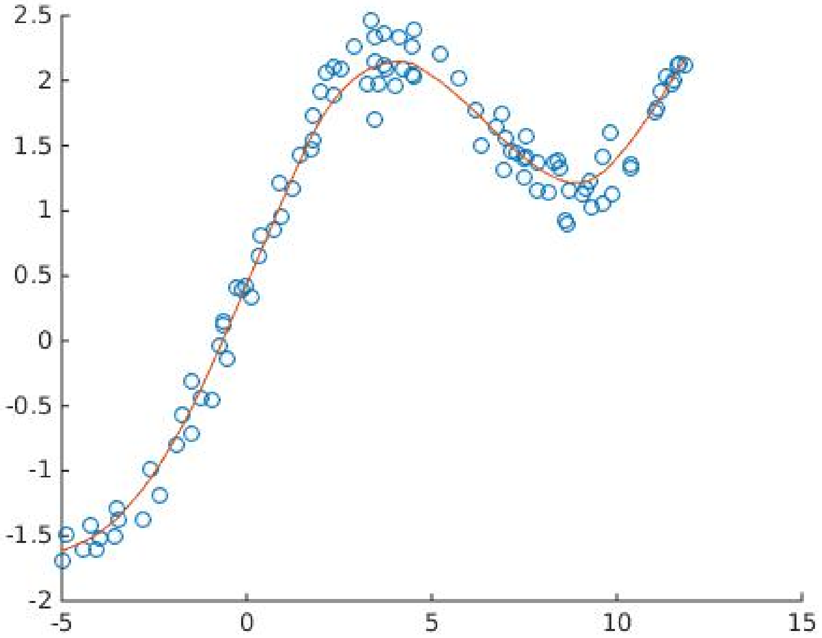
The value is: $\theta = \begin{bmatrix} 0.1753 \\ 0.3277 \end{bmatrix}$



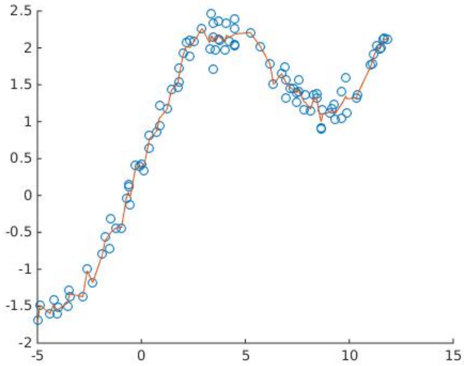
(b) Locally weighted Linear Regression has been implemented using the formula

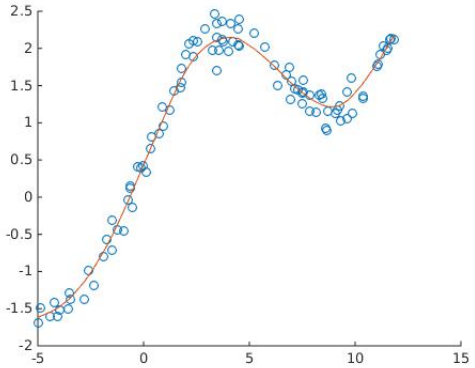
$$w^{(i)} = exp\Big(-\frac{(x - x^{(i)})^2}{2\tau^2}\Big) \tag{4}$$

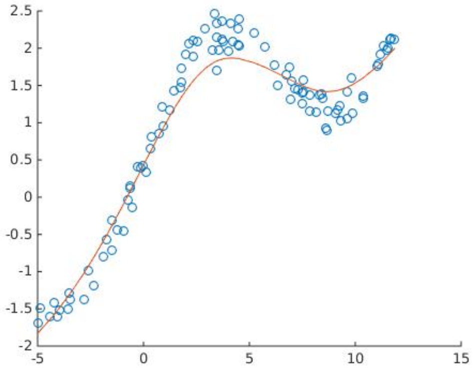
The solution for this is derived to be equal to $\theta_x = (X^T W X)^{-1} X^T W Y$. The plot for $\tau = 0.8$ is as follows

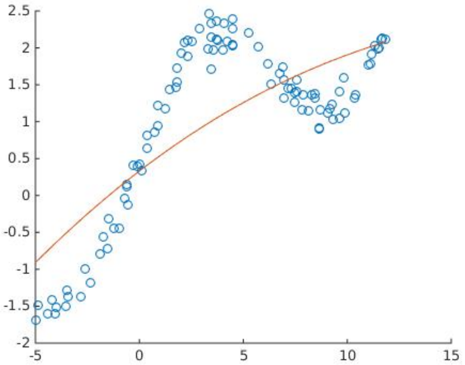


(c) The part above is repeated for different values of τ (0.1,0.3,2,10). The plots obtained are as below. It is observed that the best fit is obtained when $\tau = 0.3$.Others are either overfitting or underfitting.









Question 3. Implementation of Logistic Regression where the log-likelihood function is described as follows:

$$LL\theta) = \sum_{i=1}^{m} y^{(i)} log(h_\theta(x^{(i)})) + (1 - y^{(i)}) log(1 - h_\theta(x^{(i)}) \qquad (5)$$

Solution (. a) The function $LL(\theta)$ is to be optimized. Thus we need to find roots for the equation $LL'(\theta) = 0$ Newton Raphson's method to find root of a function is implemented to iteratively find the value of θ . Thus $\theta_{new} = \theta - H^{-1}\nabla_\theta LL(\theta)$. Here H is the Hessian which is a matrix of the form (n+1)*(n+1) where n is the dimensions (including the intercept). The equations derived to find $\nabla_\theta LL(\theta)$ and H are described as below.

$$\nabla_\theta LL(\theta) = \sum_{i=1}^{m} \big(y^{(i)} - \frac{1}{1+e^{-\theta^T x^{(I)}}}\big).x^{(i)} \tag{6}$$

$$H(j,k) = \frac{\partial(\nabla(L(\theta)_j)}{\partial(\theta_k)} = \sum_{i=1}^{m} -\frac{x_j^i x_k^i e^{-\theta^T x^{(i)}}}{(1+e^{-\theta^T x^{(i)}})^2} \tag{7}$$

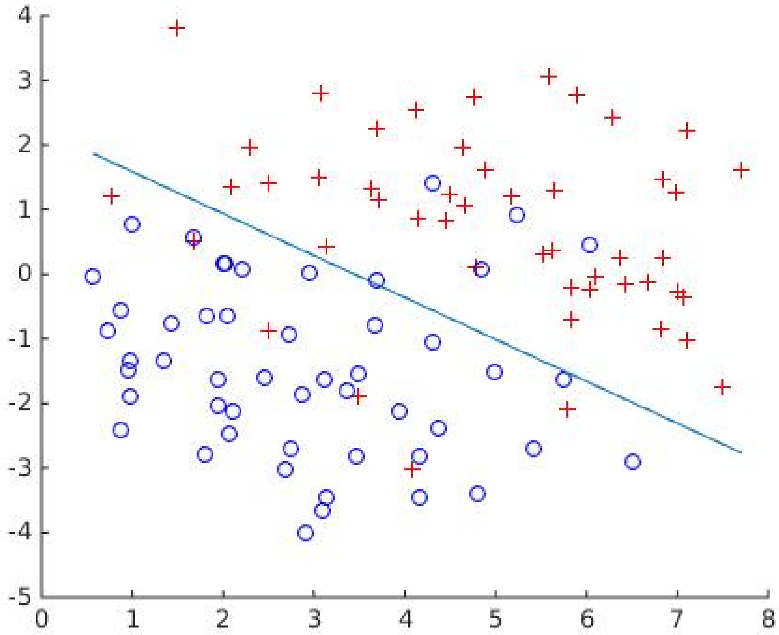
The values reported are as follows:

$$\text{H} = \begin{bmatrix} \text{-9.9097} & \text{-38.4321} & \text{2.5682} \\ \text{-38.4321} & \text{-182.1719} & \text{17.2505} \\ \text{2.5682} & \text{17.2505} & \text{-18.3116} \end{bmatrix}$$

$$\nabla_\theta LL(\theta) = \begin{bmatrix} \text{-0.0193} \\ 0.0262 \\ 0.2006 \end{bmatrix} * 1.0e^{-10}$$

$$\theta = \begin{bmatrix} \theta_0 \\ \theta_1 \\ \theta_2 \end{bmatrix} = \begin{bmatrix} \text{-2.6205} \\ 0.7604 \\ 1.1719 \end{bmatrix}$$

(b) Plot of training data and decision boundary.



Question 4. Gaussian Discriminant Analysis

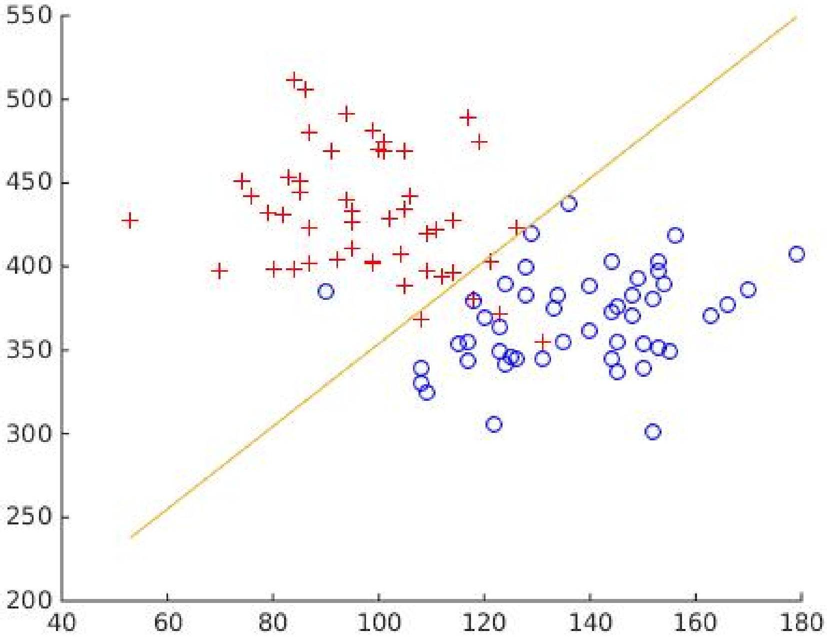
Solution (. a) Implementation of GDA where $\sum_0 = \sum_1 = \sum$ (co-variance).

The values obtained are $\sum = \begin{bmatrix} 0.287 & -0.0267 \\ -0.0267 & 1.1233 \end{bmatrix} * 1.0e^{+03}$

$\mu_1 = \begin{bmatrix} 98.3800 \\ 429.6200 \end{bmatrix}$

$\mu_0 = \begin{bmatrix} 137.4600 \\ 366.6200 \end{bmatrix}$

(b) Plot of data and linear decision boundary.



(c) Here $\sum_0 = \sum_1 = \sum$. Thus the equation of the decision boundary is linear. It is described as follows.

$$\Big((\mu_0^T - \mu_1^T)\overset{-1}{\sum})\Big)X + log(\frac{(1-\phi)}{\phi}) + \frac{1}{2}(\mu_1^T \overset{-1}{\sum}\mu_1 - \mu_0^T \overset{-1}{\sum}\mu_0) \tag{8}$$

(d)The values for $\sum_0$ and $\sum_!$ are as follows

$\Sigma_1 = \begin{bmatrix} 0.2554 & -0.1843 \\ -0.1843 & 1.3711 \end{bmatrix} * 1.0e^{+03}$

$\Sigma_0 = \begin{bmatrix} 319.5684 & 130.8348 \\ 130.8348 & 873.3956 \end{bmatrix}$

. The values of μ_0 and μ_1 remain the same as in part (a).

(e) The equation for the quadratic boundary is derived to be as $Ax^2 + Bx + C = 0$ where A,B and C are defined as follows

$A = \frac{\Sigma_0^{-1} - \Sigma_1^{-1}}{2}$

$B = \left(\mu_0^T \sum_0^{-1} - \mu_1^T \sum_1^{-1}\right)$

$C = log(\frac{1-\phi}{\phi}) + \frac{1}{2}\left(log|\sum_0| - log|\sum_1|\right) + \frac{1}{2}\left(\mu_1^T \sum_1^{-1} \mu_1 - \mu_0^T \sum_0^{-1} \mu_0\right)$

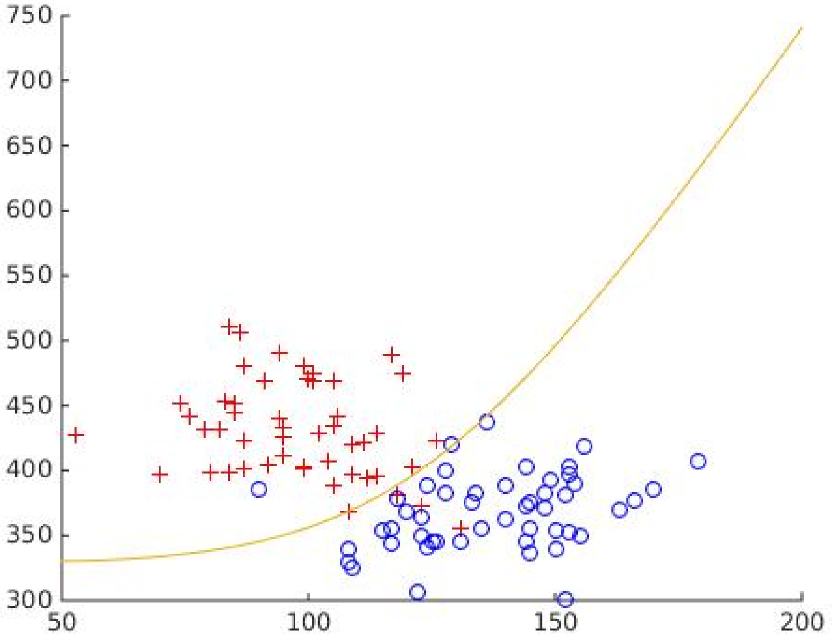
The plot of the quadratic boundary is given as below.

. The values obtained are

$A = \left[\begin{array}{cc} 0.5015 & 0.5406 \\ 0.5406 & \text{-}0.2045 \end{array} \right] * 1.0e^{-03}$

$B = \left[\begin{array}{c} \text{-}0.4015 \\ \text{-}0.0268 \end{array} \right]$

$C = 32.1141$



(f) LDA (Linear Discriminant Analysis) can only learn linear boundaries, while QDA (Quadratic Discriminant Analysis) can learn quadratic boundaries and is therefore more flexible. QDA, because it allows for more flexibility for the covariance matrix, tends to fit the data better than LDA, but then it has more parameters to estimate. The number of parameters increases significantly with QDA. Because, with QDA, we will have a separate covariance matrix for every class , if we have many classes and not so many sample points, this can be a problem.