COL774 Assignment 1 Report

Guttikonda Veera Teja

August 2023

1 Batch Gradient Descent

1.1 Calculating Theta

I Calculated theta using Linear regression Model,

$$Y{=}\theta^T X$$

I've taken $\eta = 0.05$ and $\epsilon = 10^{-15}$ before applying Newton's method I've normalized the data by applying

$$X'{=}(X\text{-}\mu)/\sigma$$

I've obtained the final theta value as follows

$$\theta{=}\begin{pmatrix}0.99662\\0.00134\end{pmatrix}$$

It took 309 Iterations for convergence.

1.2 Plot

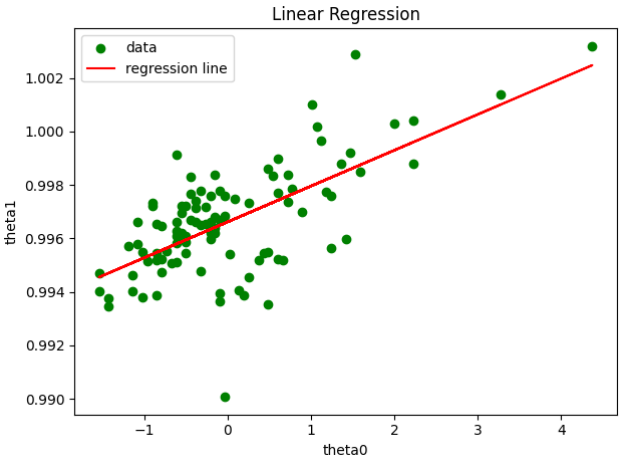


Figure 1: Linear Regression

1.3 3D Mesh plot

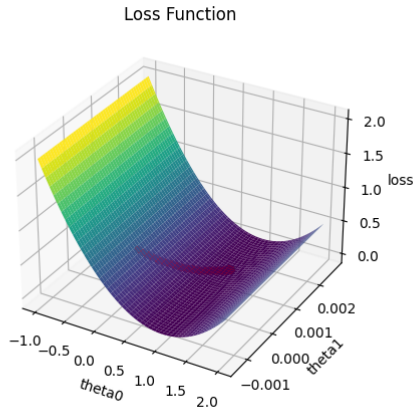
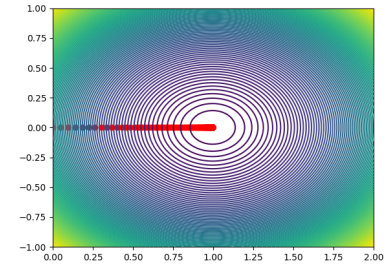
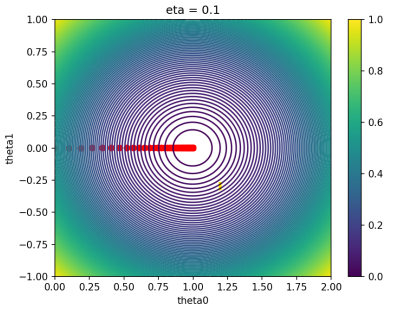


Figure 2: Caption

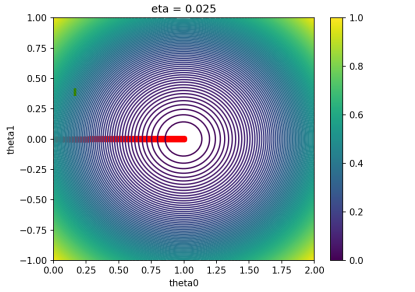
1.4 Contours



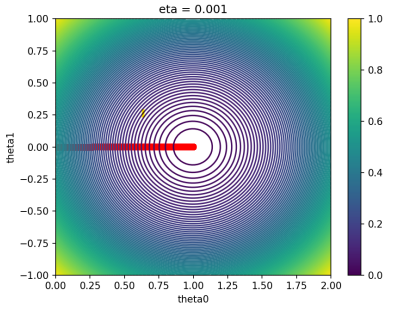
(a) $\eta = 0.05$



(b) $\eta = 0.1$



(c) $\eta = 0.025$



(d) $\eta = 0.001$

Figure 3: Arrangement of Images

1.5 Gradient Descent for different ηs

The change in learning rate changes how fast the convergence occur,but very little effect on final theta. for $\eta = 0.1$ the the iterations were less and for $\eta = 0.001$ iterations were very high

2 Stochastic Gradient Descent

2.1 Sampling the data

The data is sampled using the distributions

$$X \sim \begin{pmatrix} \mathcal{N}(3,4) \\ \mathcal{N}(-1,4) \end{pmatrix}$$

$$\epsilon \sim \mathcal{N}(0,2) \tag{1}$$

$$Y \sim \theta^T X + \epsilon$$

2.2 Stochastic Gradient Descent

I've used the following criteria for convergence

$$|J(\theta^{t+1})) - \underset{\text{last k}}{average}(J(\theta^t))| \leq \varepsilon, \text{ where } \varepsilon = 10^{-5} \tag{2}$$

I've used a Deque for maintaining the last k=1000 losses. the θ's Learned are as follows:

Batch Size	θ	Iterations	Epochs	Time
1	$\begin{pmatrix} 2.99219109 \\ 0.9831456 \\ 1.98827262 \end{pmatrix}$	254375	0	2.290367841720581
100	$\begin{pmatrix} 3.02416478 \\ 0.99684526 \\ 2.00018258 \end{pmatrix}$	5831	0	2.1171138286590576
10000	$\begin{pmatrix} 2.99252511 \\ 1.002385 \\ 1.99778123 \end{pmatrix}$	2200	22	0.9189879894256592
1000000	$\begin{pmatrix} 2.88651657 \\ 1.02508608 \\ 1.99166914 \end{pmatrix}$	1176	1176	50.1960883140564

Table 1: θs for various batch sizes

2.3 Analysis of different batch sizes

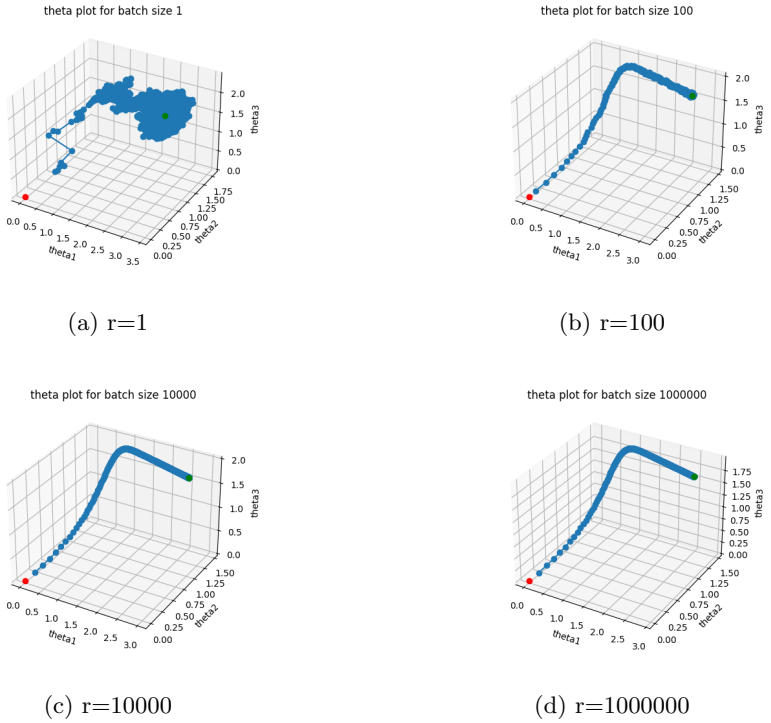
different batch sizes give different final values , θs, but they were almost near to the original θ for smaller batch sizes,i.e,r=1, it takes more time as were gradient is changing at every single batch ,i.e, every data, which results in higher time to converge i.e, more iterations. for large batch sizes,i.e r=1000000, it takes more time than others as it has to calculate the loss of many data points which takes more time to calculate loss for a single batch only. but for r=100,10000

convergence is very fast because of lesser batch size and not very less such that it won't fluctuate abruptly. iterations got decreased as r increases.

batch size r	θ	error from given data set
1	$\begin{pmatrix} 2.99219109 \\ 0.9831456 \\ 1.98827262 \end{pmatrix}$	1.00681566
100	$\begin{pmatrix} 3.02416478 \\ 0.99684526 \\ 2.00018258 \end{pmatrix}$	0.9836739
10000	$\begin{pmatrix} 2.99252511 \\ 1.002385 \\ 1.99778123 \end{pmatrix}$	0.98368742
1000000	$\begin{pmatrix} 2.88651657 \\ 1.02508608 \\ 1.99166914 \end{pmatrix}$	1.0211145668
	original θ	0.9829469215

Table 2: Caption

2.4 Plot of movement of θs for different batch sizes



(a) r=1

(b) r=100

(c) r=10000

(d) r=1000000

Figure 4: Arrangement of Images

for r=1 , the plot is rather more fluctuating than the other because of changing gradient at every data point, which changes direction very frequently than expected thus the plot is very zig-zag. for other r values the plot seems smooth with very small fluctuations. it converges very smoothly. this behaviour is due to as the batch size increases the direction of descent becomes more and more accurate. thus ninimum fluctuations.

3 Logistic Regression

I've calculated Hessian

$$H = \nabla^2(l(\theta))$$

where

$$H_{ij} = \frac{\partial^2 l(\theta)}{\partial\theta_i \partial\theta_j}$$

from notes

$$H = -X^T\left(\frac{e^{\theta^T X}}{(1+e^{\theta^T X})^2}\right)X$$

$$\theta = \begin{pmatrix} 0.401253 \\ 2.588547 \\ -2.725588 \end{pmatrix}$$

3.1 Plots

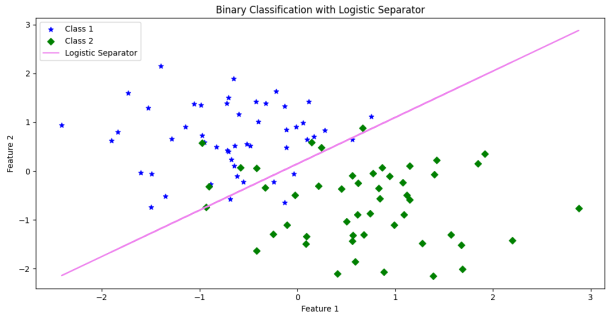


Figure 5: Logistic Regression

4 Gaussian Discriminant Analysis

I've calculated mean and covariance matrices using formulae discussed in class. $\phi, \mu_1, \mu_0, \Sigma$ can be obtained by maximizing likelihood l(θ) i.e, $\nabla_\phi l(\theta) = 0, \nabla_{\mu_1} l(\theta) = 0, \nabla_{\mu_0} l(\theta) = 0, \nabla_\Sigma l(\theta) = 0$

$$\mu_0 = \frac{\sum_{i=1}^{m} 1_{y^{(i)}=0} x^{(i)}}{\sum_{i=1}^{m} 1_{y^{(i)}=0}}$$

$$\mu_1 = \frac{\sum_{i=1}^{m} 1_{y^{(i)}=1} x^{(i)}}{\sum_{i=1}^{m} 1_{y^{(i)}=1}}$$

$$\phi = \frac{\sum_{i=1}^{m} 1_{y^{(i)}=1}}{m}$$

4.1 Linear Separator

Assuming $\Sigma_0 = \Sigma_1 = \Sigma$

$$\Sigma = \frac{\sum_{i=1}^{m} (x^{(i)} - \mu_{y^{(i)}})(x^{(i)} - \mu_{y^{(i)}})^T}{m}$$

From the given data set, I've got the following
$\phi = 0.5$
$\mu_0 = \begin{pmatrix} -0.75529433 \\ 0.68509431 \end{pmatrix}$

$$\mu_1 = \begin{pmatrix} 0.75529433 \\ -0.68509431 \end{pmatrix}$$
$$\Sigma = \begin{pmatrix} 0.42953048 - 0.02247228 \\ -0.022472280.53064579 \end{pmatrix}$$
decision boundary equation is

$$log(\frac{\phi}{1-\phi}) + (\mu_1 - \mu_0)^T \Sigma^{-1} x - \frac{1}{2}(\mu_1^T \Sigma^{-1} \mu_1 - \mu_0^T \Sigma^{-1} \mu_0) = 0$$

This is a linear boundary because there are no quadratic terms in x

4.2 Plot

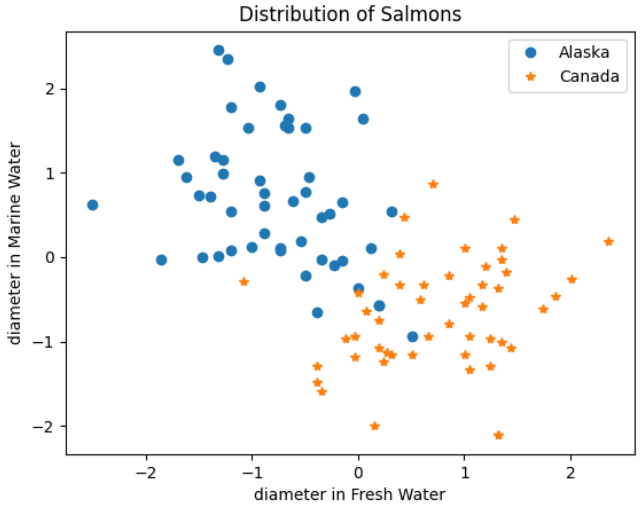


Figure 6: Data Distribution

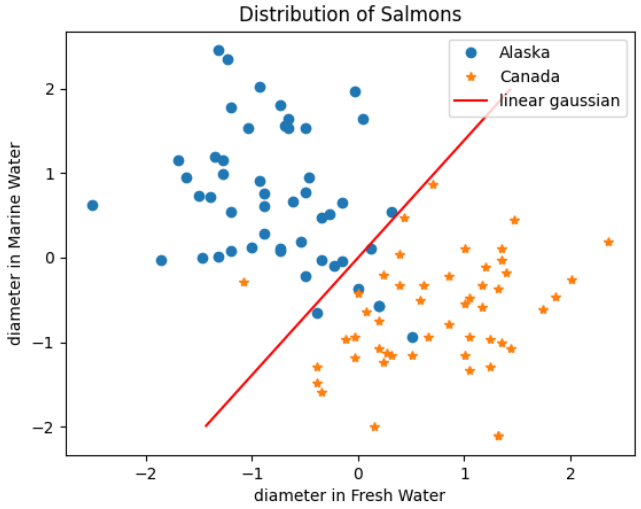


Figure 7: Linear Separator

4.3 Quadratic Separator

for more general Gaussian distribution $\Sigma_0 \neq \Sigma_1$ similar to linear case, applying $\nabla_\phi l(\theta) = 0, \nabla_{\mu_1} l(\theta) = 0, \nabla_{\mu_0} l(\theta) = 0, \nabla_{\Sigma_0} l(\theta) = 0, \nabla_{\Sigma_1} l(\theta) = 0$

$$\Sigma_0 = \frac{\sum_{i=1}^{m} (x^{(i)} - \mu_0)(x^{(i)} - \mu_0)^T 1_{y^{(i)}=0}}{\sum_{i=1}^{m} 1_{y^{(i)}=0}}$$

$$\Sigma_1 = \frac{\sum_{i=1}^{m} (x^{(i)} - \mu_1)(x^{(i)} - \mu_1)^T 1_{y^{(i)}=1}}{\sum_{i=1}^{m} 1_{y^{(i)}=1}}$$

then the decision boundary will be

$$\log\left(\frac{\phi}{1-\phi}\sqrt{\frac{|\Sigma_0|}{|\Sigma_1|}}\right) + \mu_1^T \Sigma_1^{-1} x - \mu_0^T \Sigma_0^{-1} x - \frac{1}{2}\left(x^T(\Sigma_1^{-1} - \Sigma_0^{-1})x + \mu_1^T \Sigma_1^{-1} \mu_1 - \mu_0^T \Sigma_0^{-1} \mu_0\right) = 0 \quad (3)$$

for this μ_1 *and* μ_0 will be same as above

$$\Sigma_0 = \begin{pmatrix} 0.381589 & -0.154865 \\ -0.154865 & 0.647737 \end{pmatrix} \Sigma_1 = \begin{pmatrix} 0.477471 & 0.1099206 \\ 0.109920 & 0.413554 \end{pmatrix}$$

As you can see there is an Quadratic term in x i.e, $x^T(\Sigma_1^{-1} - \Sigma_0^{-1})x$, Hence the boundary is Quadratic.

4.4 Plot for quadratic Separator

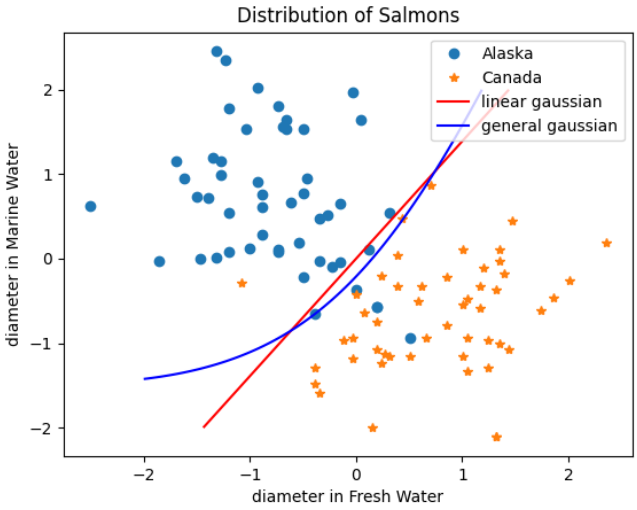


Figure 8: Quadratic Separator