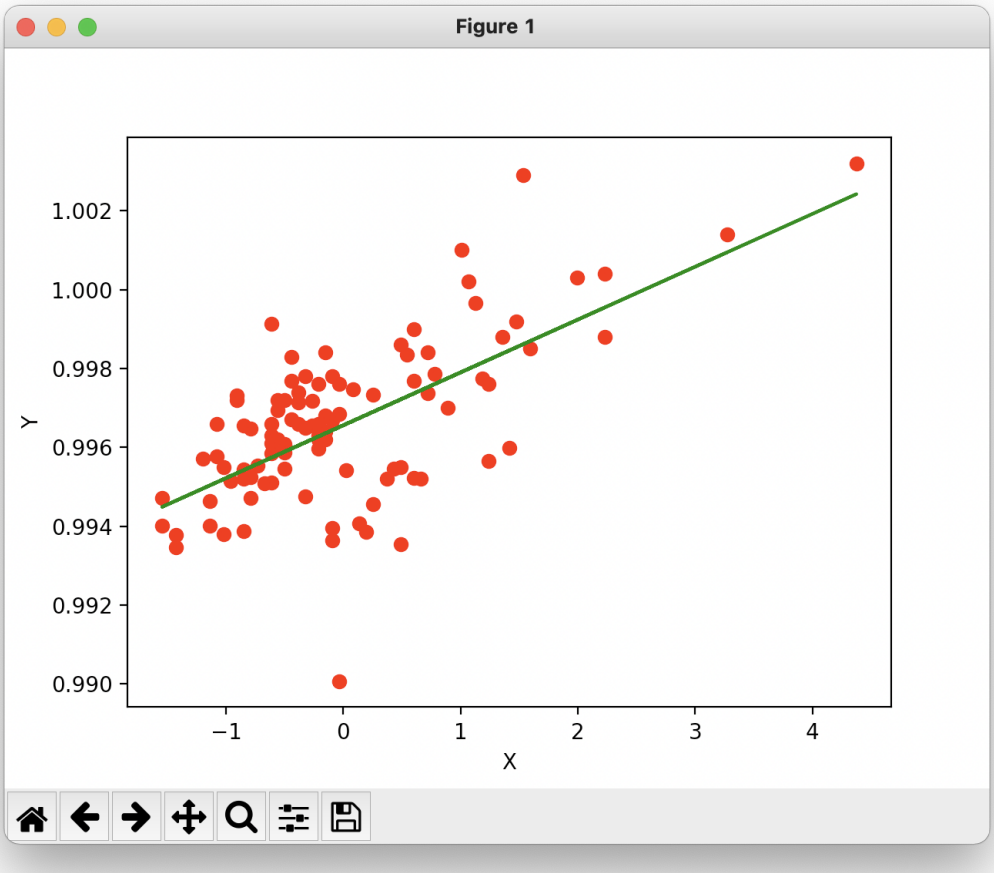
COL774 Assignment 1

Harshil Vagadia - 2019CS10356

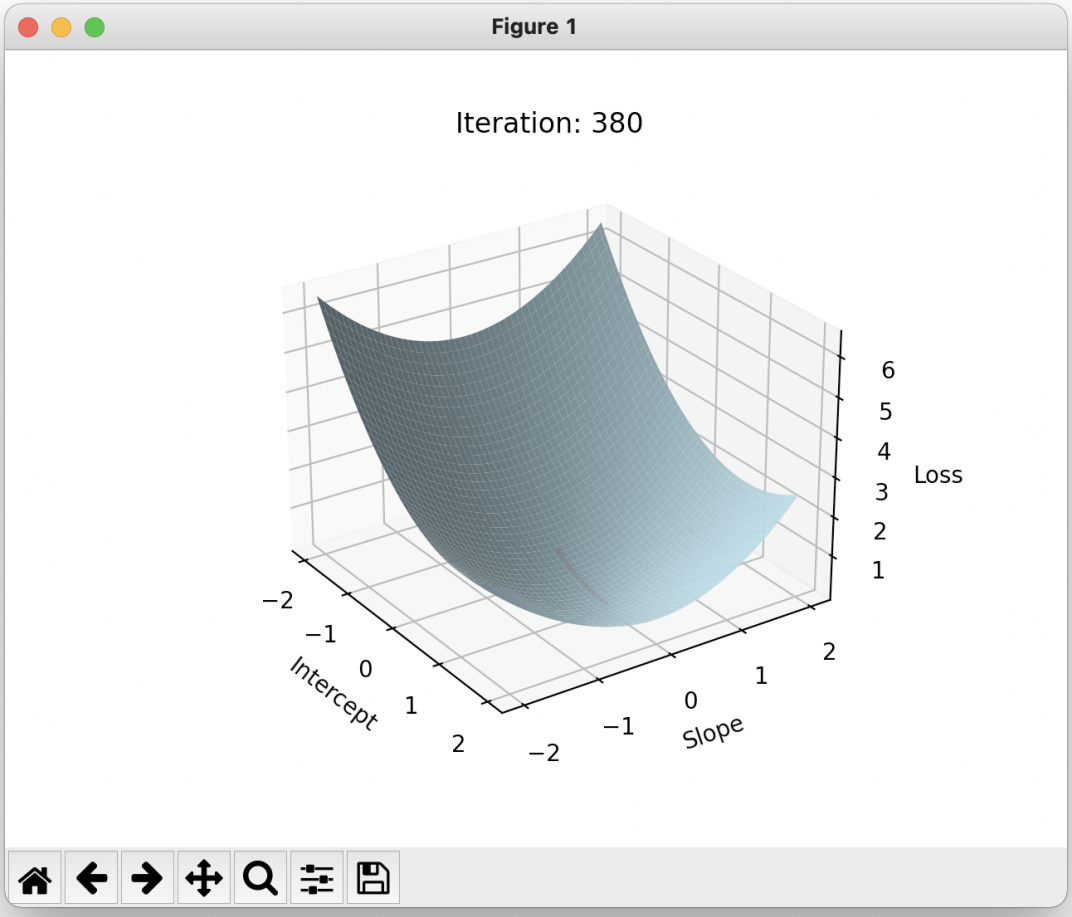
1. Linear Regression

A. This part requires implementing batch gradient descent. The parameter vector `theta` is initialized as all zeros vector (shape (3, 1)). Then the gradient was computed and subtracted from theta after multiplying by a suitable learning rate η. η was chosen to be 0.025 because the convergence was quite fast without overshooting the minima. If the hypothesis is assumed to be $y = \theta_0 + \theta_1 x$ then the final parameters came out to be $\theta_0 = 0.9965 \; and \; \theta_1 = 0.0013$. The final loss came out to be $1.19 * 10^{-6}$. The convergence criteria was that the absolute difference between current loss and previous iteration loss should be less than 10^{-9}.

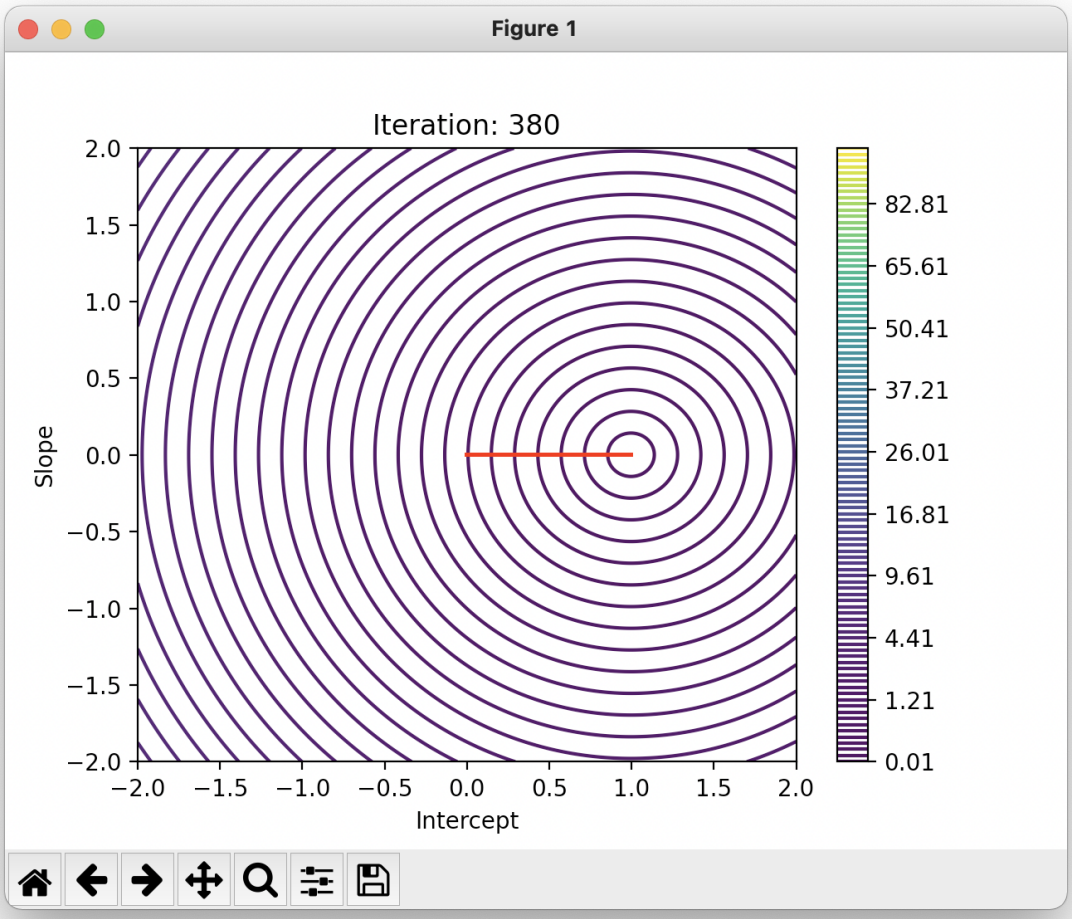
B. The linear hypothesis learned by the model



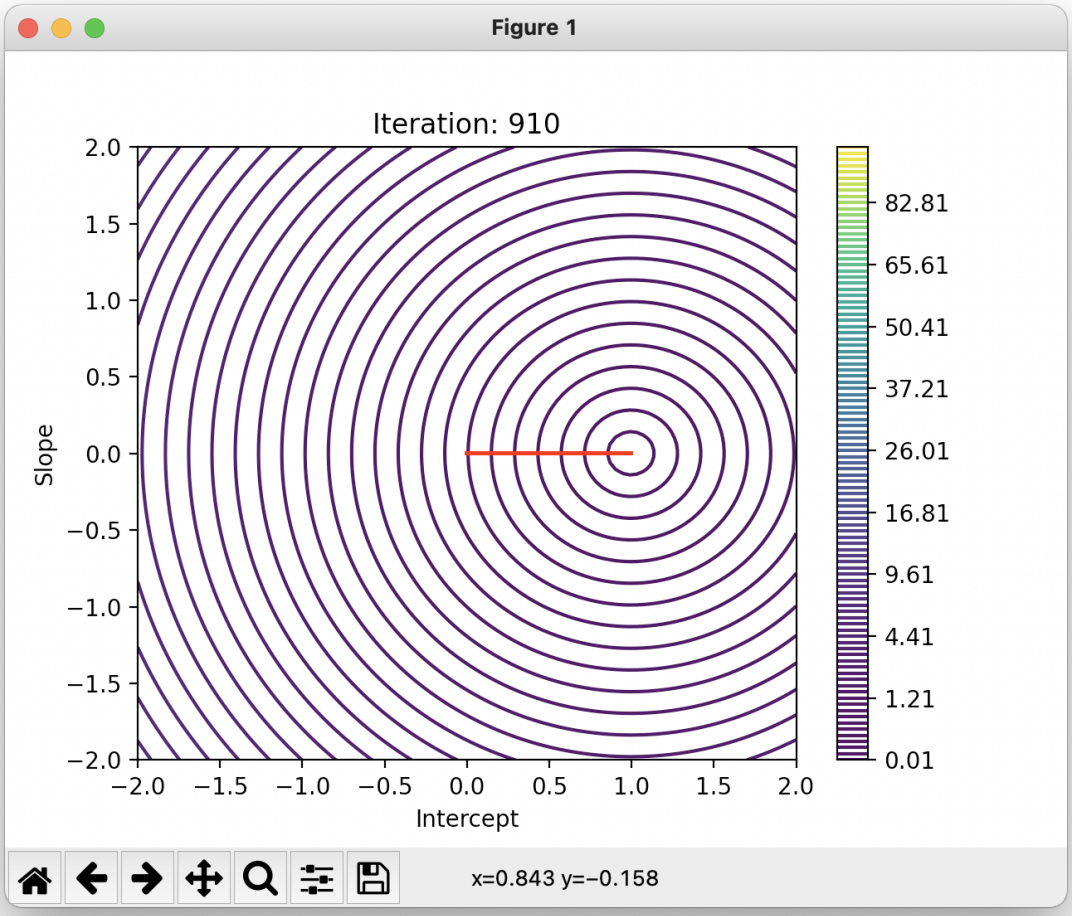
C. The variation of loss with each iteration is shown by the following graph:



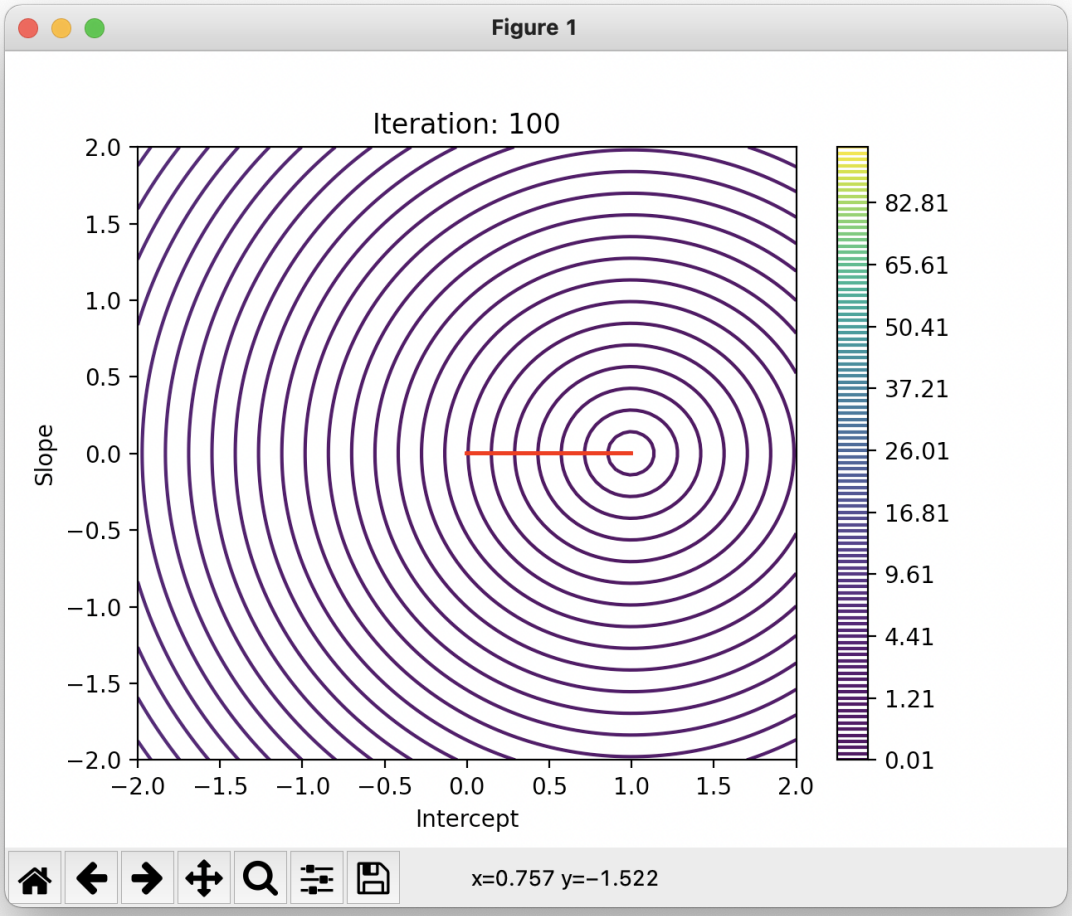
D. The contour plot for $\eta = 0.025$ is:



E. The plots for $\eta = 0.01$ is given below. The contours are the same. However it takes more iterations for theta to reach the minima as the learning rate is lower.



While for $\eta = 0.1$, the plot is:



The number of iterations taken to reach the minima are even lesser.

2. Sampling and Stochastic Gradient Descent

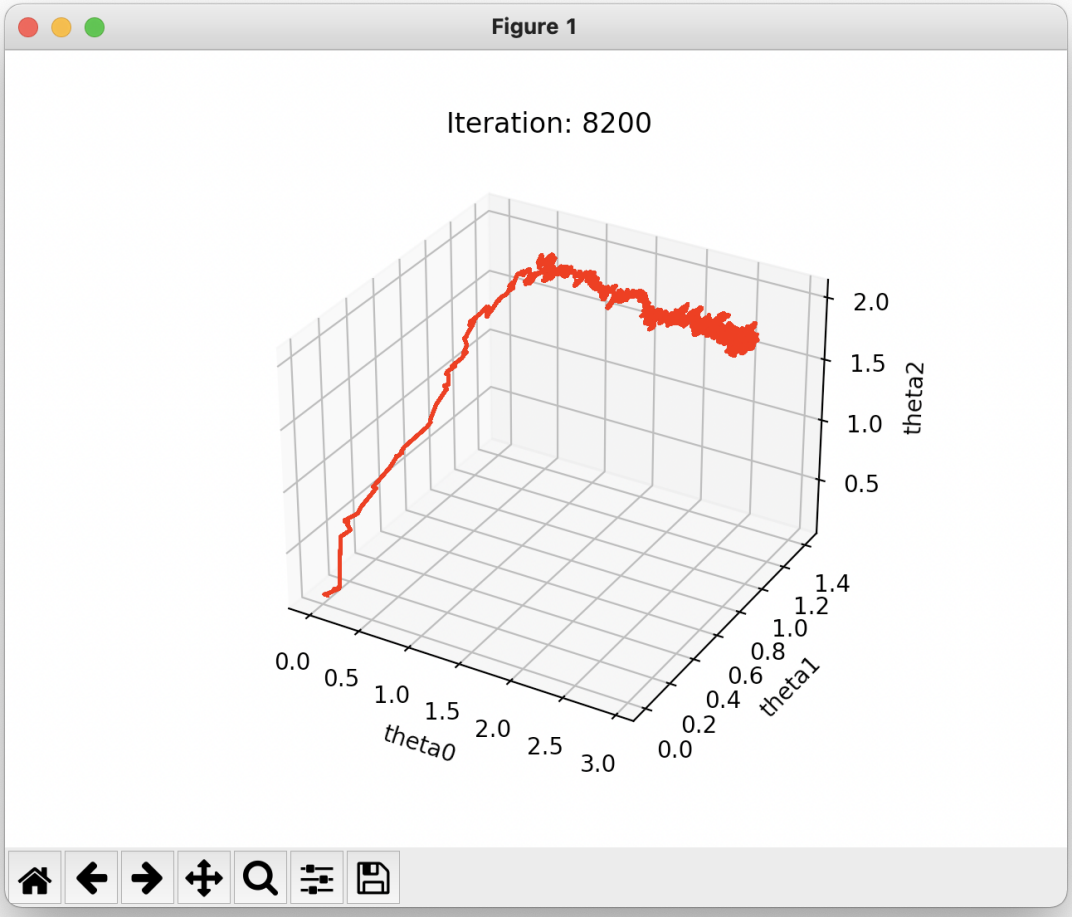
A. The points were sampled according to given normal distributions.
B. The value of parameters, and stopping criteria for each batch size is given below. The stopping criteria is defined as the average loss of k iterations and the average loss of previous k iterations are within 10^{-3} of each other. A good value of k depends on the batch size. As batch size decreases, the value of k should increase as the loss variation becomes noisy.
    a. Batch = 1, $k = 10000$, $\theta = [2.98274104,\ 0.99013716,\ 2.03490149]$, loss = 1.036, test loss = 1.047, iterations = 230000, time taken = 5.96s
    b. Batch = 100, $k = 1000$, $\theta = [2.99640015,\ 1.00124101,\ 1.99678686]$, loss = 0.998, test loss = 0.983, iterations = 10000, time taken = 0.28s

c. Batch = 10000, $k = 100$, $\theta = [2.84869305,\ 1.01826823,\ 1.99448665]$, loss = 1.008, test loss = 1.009, iterations = 4900, time taken = 1.45s
d. Batch = 1000000, $k = 1$, $\theta = [1.36560722,\ 1.19300853,\ 1.9353347\ ]$, loss = 1.824, test loss = 3.963, iterations = 954, time taken = 34.1s

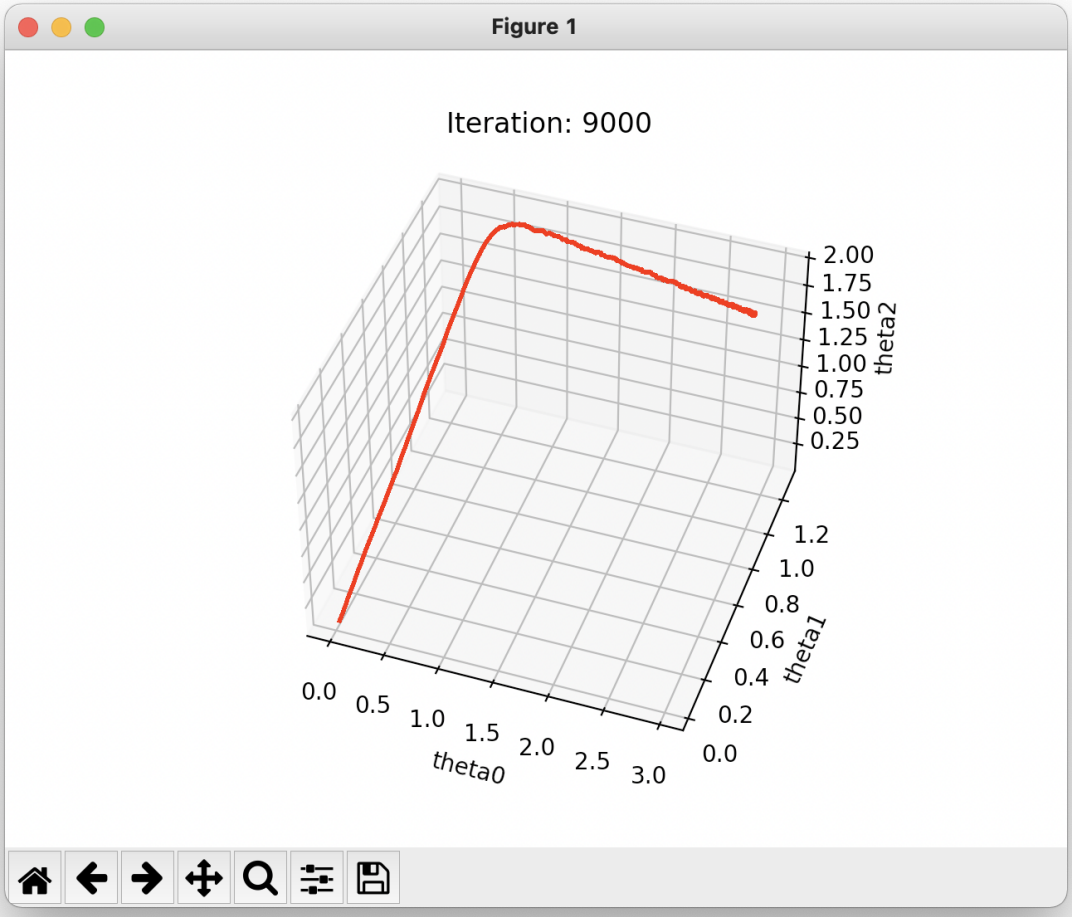
C. The value of parameters for reasonable batch size (1, 100, 10000) comes out to be close to the actual parameters. Both training loss and test loss comes out to be close to 1, which is due to the inherent noise of the data (std = 2). Hence all the models (a,b,c) are able to fit the underlying linear function quite well. The number of iterations decreases with increase in batch size, but the total time does not follow this pattern. As the cost of each iteration increases, the total time taken increases after a certain batch size. Batch size 100 converges to a reasonable hypothesis the quickest. For the extreme case of batch size 1000000, the final hypothesis is far off. This is because the updation per iteration is so slow that the algorithm falsely thinks that SGD has converged. Fot hypothesis a,b,c the test loss is close to training loss, which indicates that the model generalises well.

D. The movement of the parameters for each batch size are given below. For smaller batch sizes, the path is more noisy. This is because at each iteration the algorithm only optimises based on a small chink of data and does not move towards the actual minima. However the average movement is towards the actual minima.

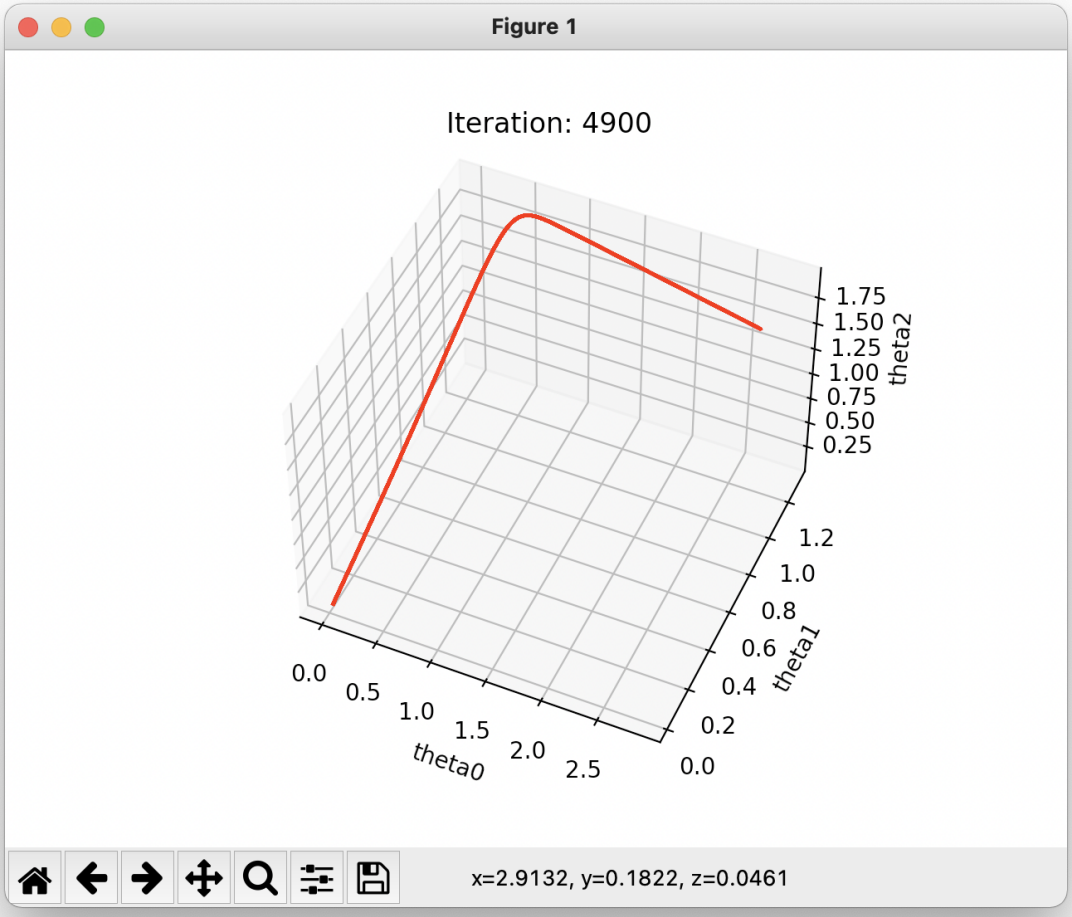
a. Batch size = 1



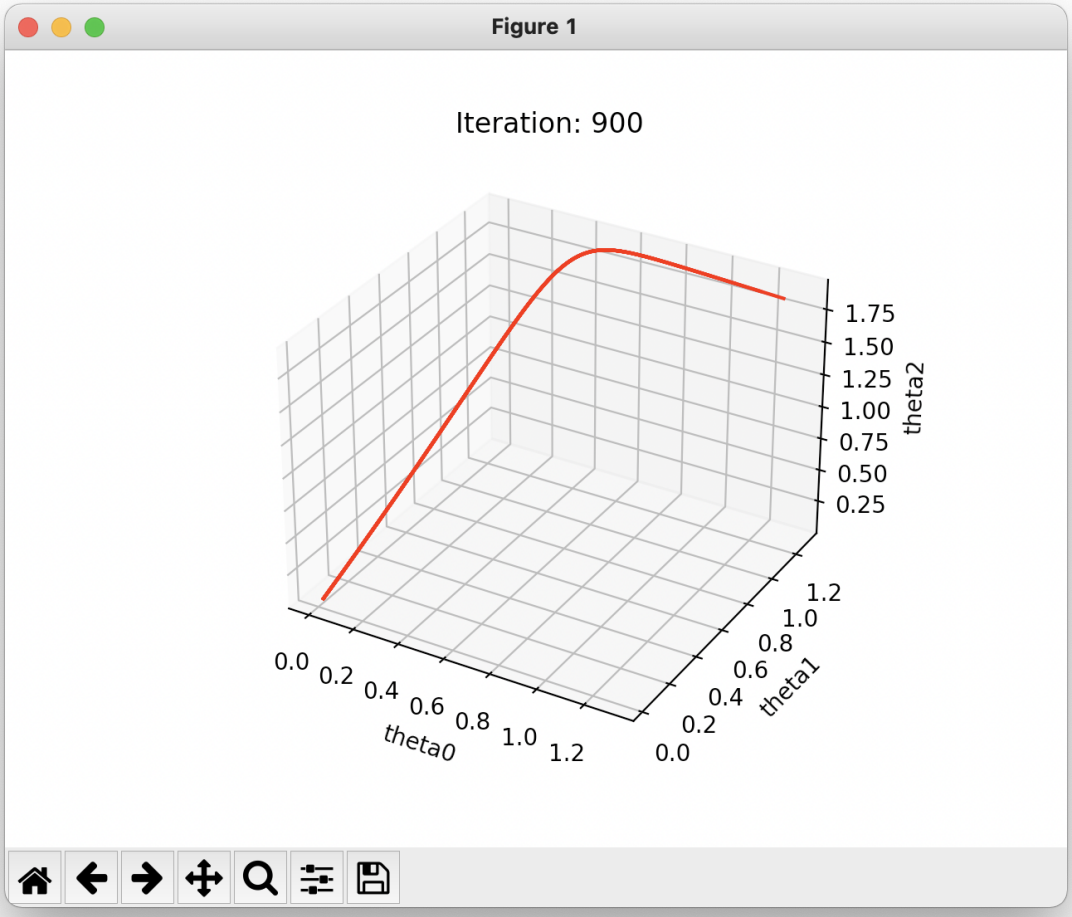
b. Batch size = 100



c. Batch size = 10000



d. Batch size = 1000000

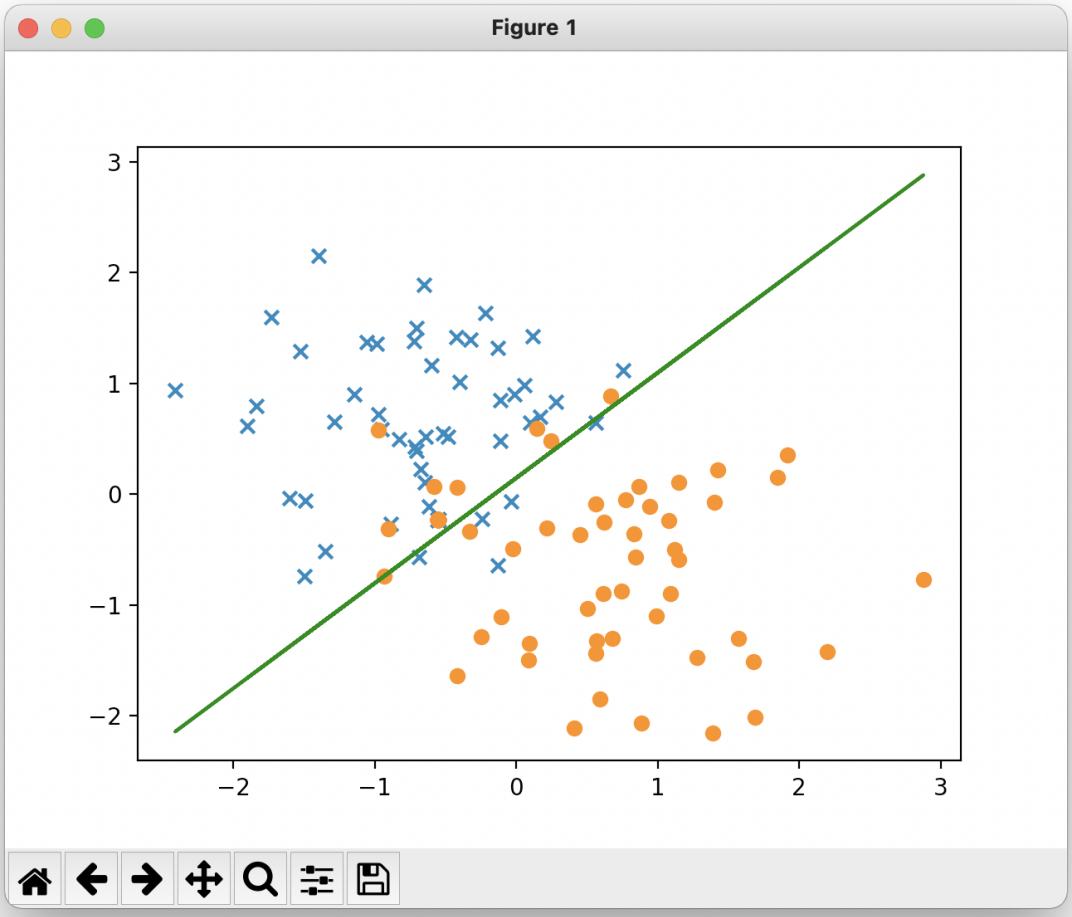


3. Logistic Regression

A. After applying logistic regression using Newton's Method on the given binary dataset, the final parameters where: $\theta_0 = 0.40$, $\theta_1 = 2.58$, $\theta_2 = -2.72$. Here the hypothesis is assumed to be $h_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2$. Nine iterations of Newton's method were done to reach convergence. The convergence criteria is that the absolute difference between current loss and previous iteration loss should be less than 10^{-9}.

B. The decision boundary is given in the following plot. The crosses are 0s and circles are 1s. The region above the line denotes $h_\theta(x) > 0.5$ and the region below denotes

$h_\theta(x) < 0.5$.



4. Gaussian Discriminant Analysis

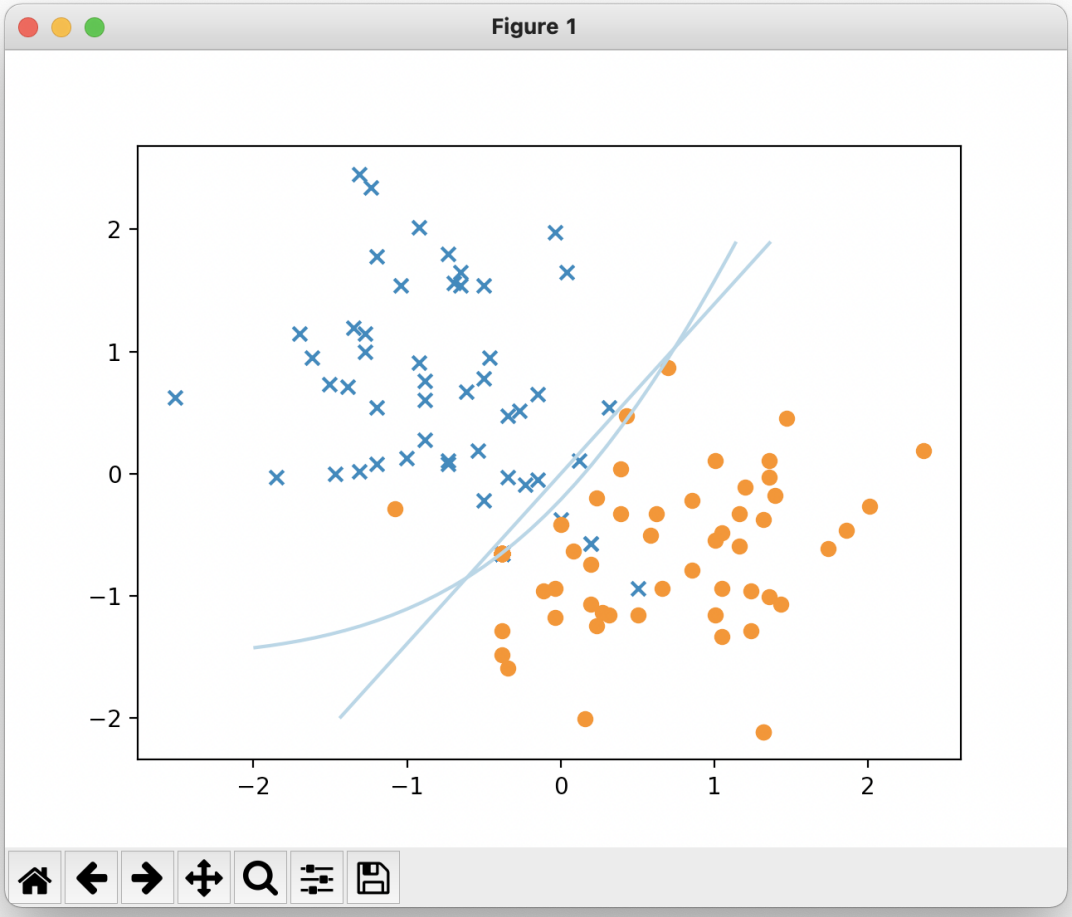
A. The value of parameters are:
$\mu_0 = [-0.75, 0.68]$, $\mu_1 = [0.75, -0.68]$, $\Sigma = [[0.42, -0.02], [-0.02, 0.53]]$, $\phi = 0.5$

B. See part E

C. The equation of decision boundary for the case of identical variance matrix is:
$-(\mu_0 - \mu_1)^T \Sigma^{-1} x + (1/2)(\mu_1^T \Sigma^{-1} \mu_1 - \mu_2^T \Sigma^{-1} \mu_2) + C = 0$, where
$C = log((1 - \phi)/\phi)$. See part E for plot.

D. The value of μ_0, μ_1 *and* ϕremains the same.
$\Sigma_0 = [0.38158978 \ -0.15486516] [-0.15486516 \ 0.64773717]]$and
$\Sigma_1 = [[0.47747117 \ 0.1099206] [0.1099206 \ 0.41355441]]$

E. The equation of decision boundary becomes:
$(1/2)x^T(\Sigma_1^{-1} - \Sigma_0^{-1})x - (\mu_0\Sigma_1^{-1} - \mu_1\Sigma_0^{-1})^T x + (1/2)(\mu_1^T\Sigma^{-1}\mu_1 - \mu_2^T\Sigma^{-1}\mu_2) + C = 0$
, where $C = log(((1 - \phi)/\phi)(|\Sigma_1|^{1/2}/|\Sigma_2|^{1/2}))$. The following plot shows all the sample points (crosses for Alaska and circles for Canada). The linear curve is the hypothesis for part B while the parabolic curve is for part E.



F. Both linear and quadratic boundaries seem to fit the data quite well. Almost all training points are classified the same by both the boundaries. One important difference here is that the quadratic boundary is curving towards the Alaska salmon. This means the quadratic hypothesis assigns a larger region to the Canadian salmon. Linear boundary makes no such distinction.

Instructions for Executing Code

Put the data folder inside the root directory. Execute python script by running `python3 qi.py <sub_part>` where i is the question number and <sub_part> is the sub part of question to run (a, b, c). Ex: `python3 q1.py a`