REPORT

Machine Learning Assignment 1

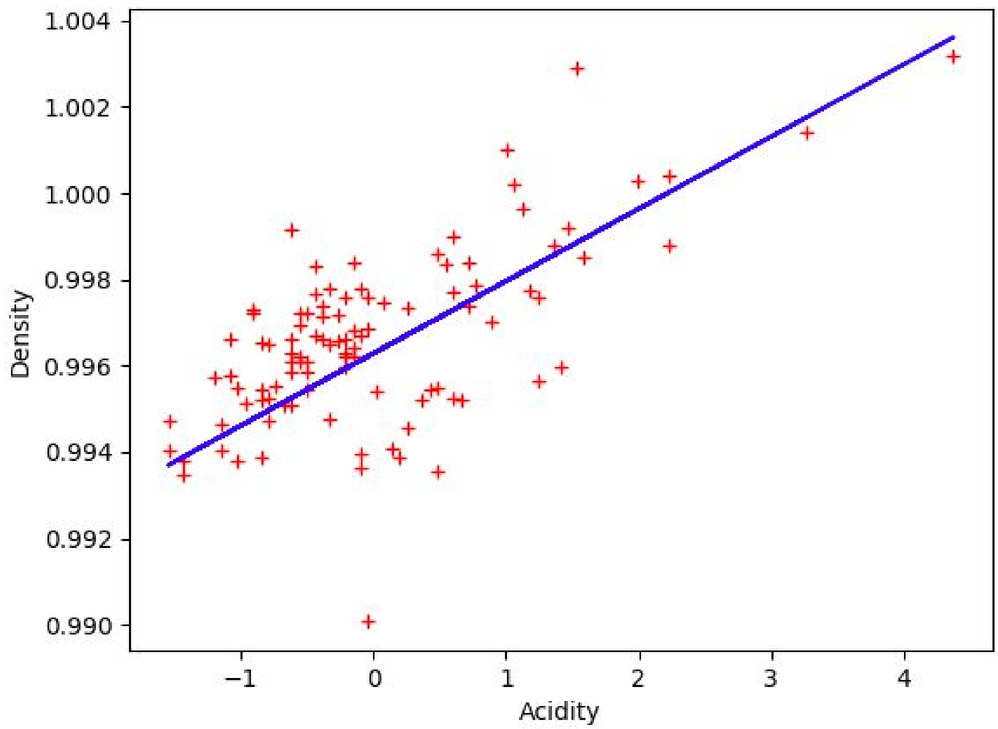
Sudeep Agarwal
2015CS50295

1. a)
   Implemented linear regression using batch gradient descent method. The input data is normalized before performing any operation. The file **linearRegression.py** takes alpha and epsilon as parameters where

   α = Learning Rate
   ϵ = Stopping Condition($cost^{(t)} - cost^{(t+1)} <=$ epsilon)

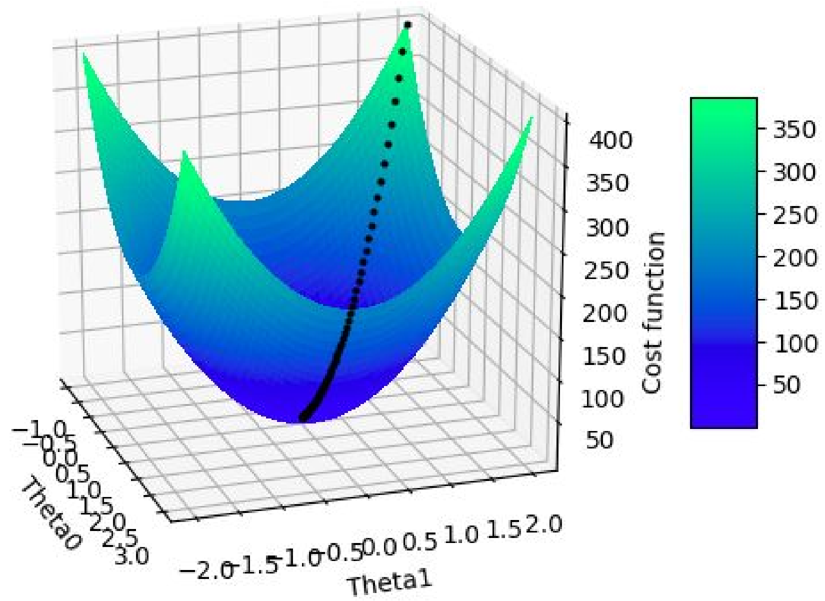
   $\alpha = 0.0004$
   $\epsilon = 0.000001$
   $\Theta_{init} = [-1\ 2]$

   b)



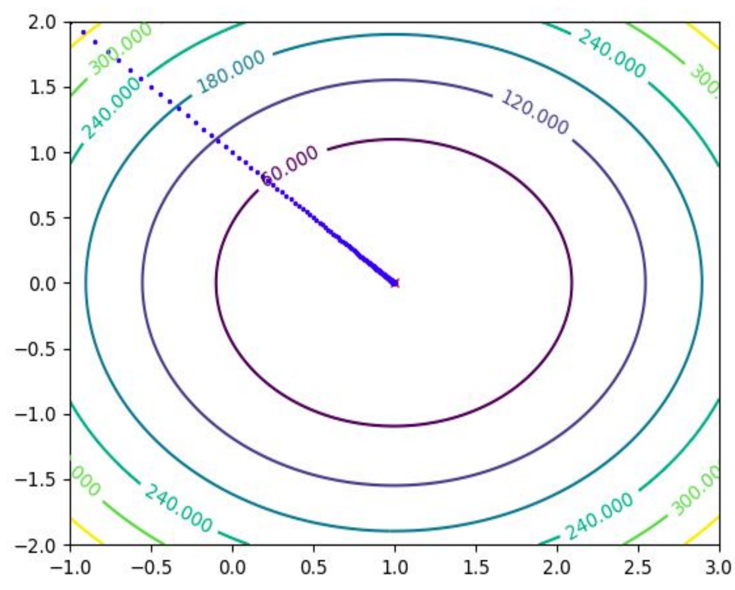
c)

The black dots in the figure below shows the successive values of theta in gradient descent.



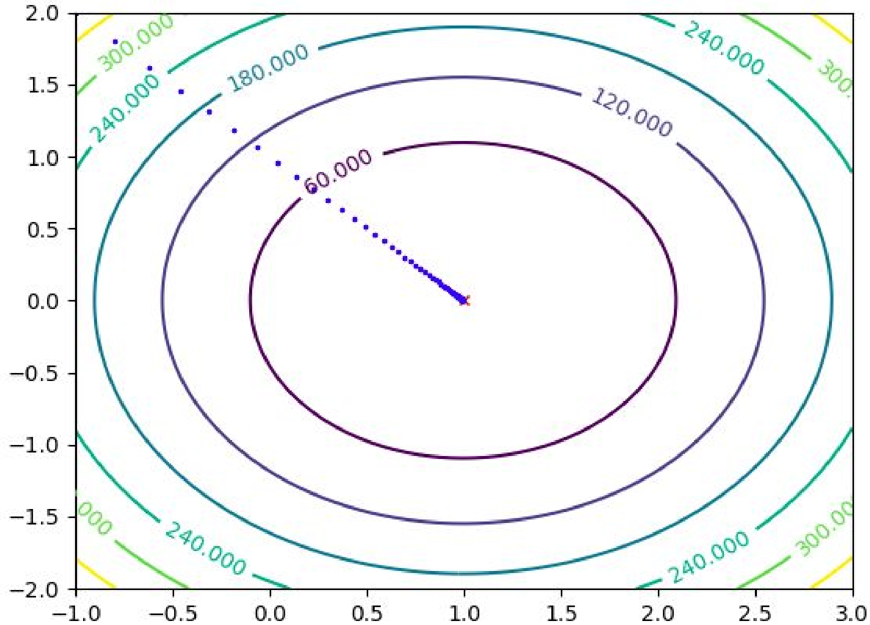
α = 0.0004 ϵ = 0.000001

d)

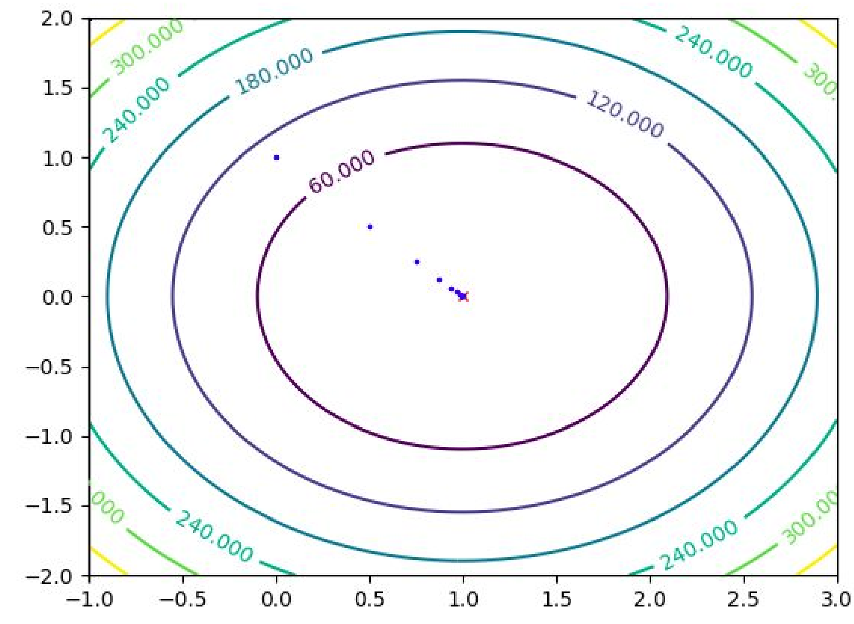


α = 0.0004 ϵ = 0.000001

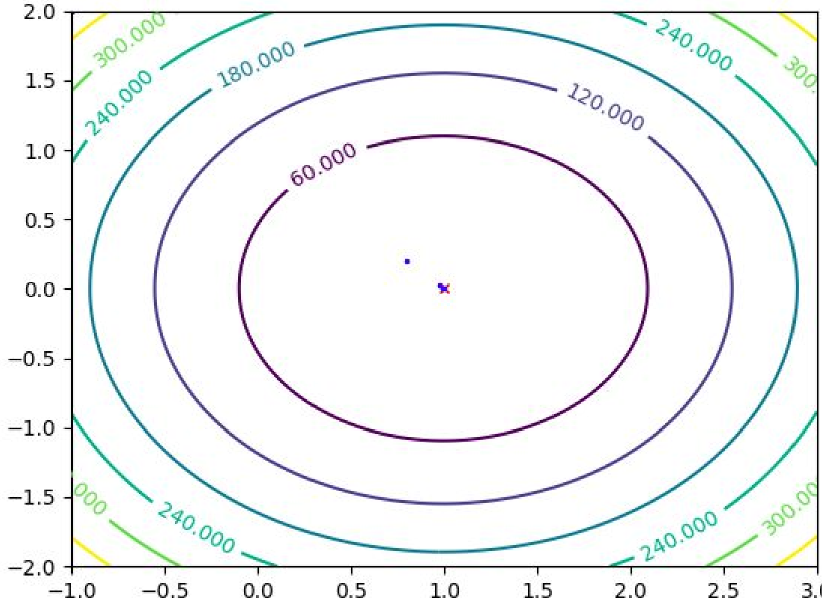
e)



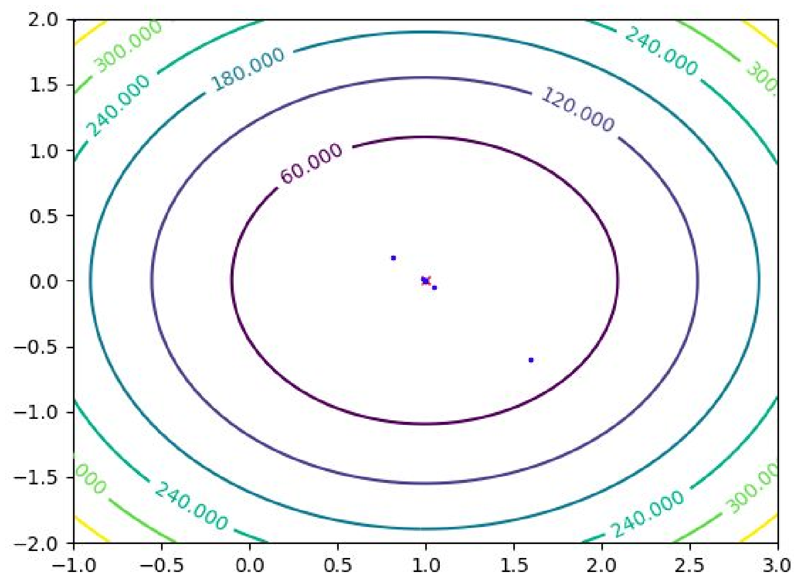
α *= 0.001 ϵ = 0.00001*



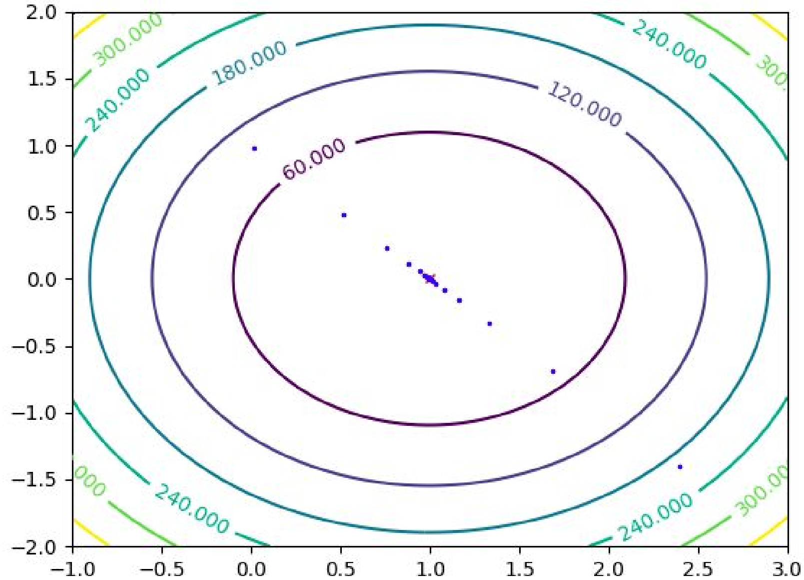
α *= 0.005 ϵ = 0.00001*



α = 0.009 ϵ = 0.00001



α = 0.013 ϵ = 0.00001

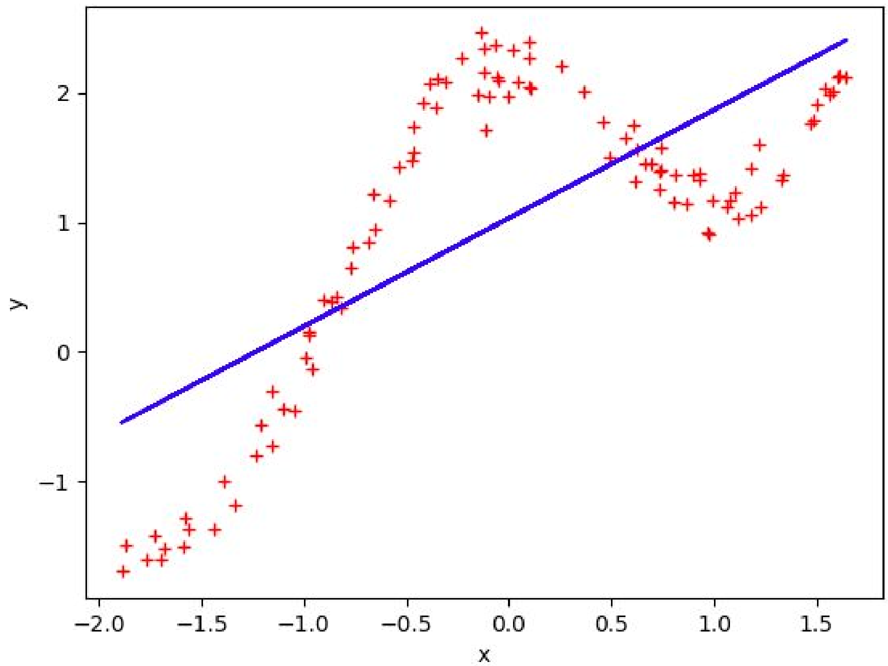


***α* = 0.017 *ϵ* = 0.00001**

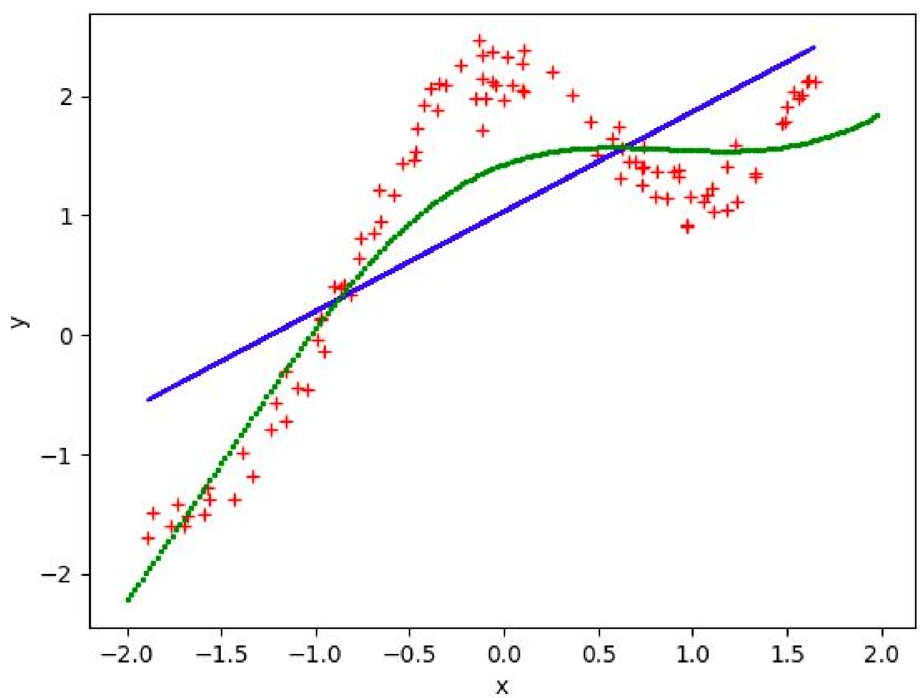
As seen from the above figures, the value of **alpha = 0.013 and above** is too large for epsilon value of 0.00001. Hence gradient descent doesn't converge.

2.

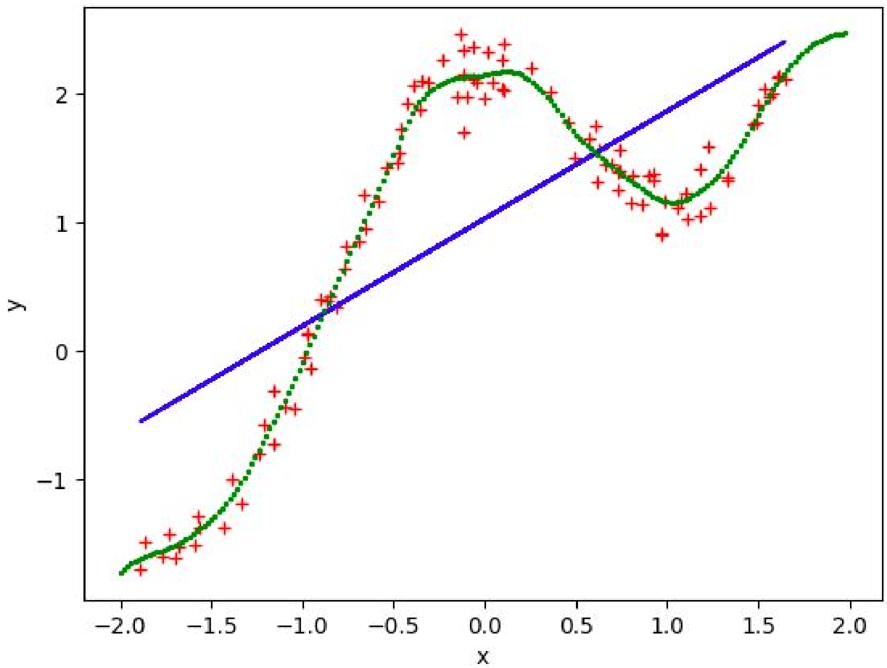
Implemented weighted linear regression using normal equations. The input data is normalized before performing any operation. The file **weightedRegression.py** takes $\boldsymbol{\tau}$ as parameter.



Uniform weights plot.

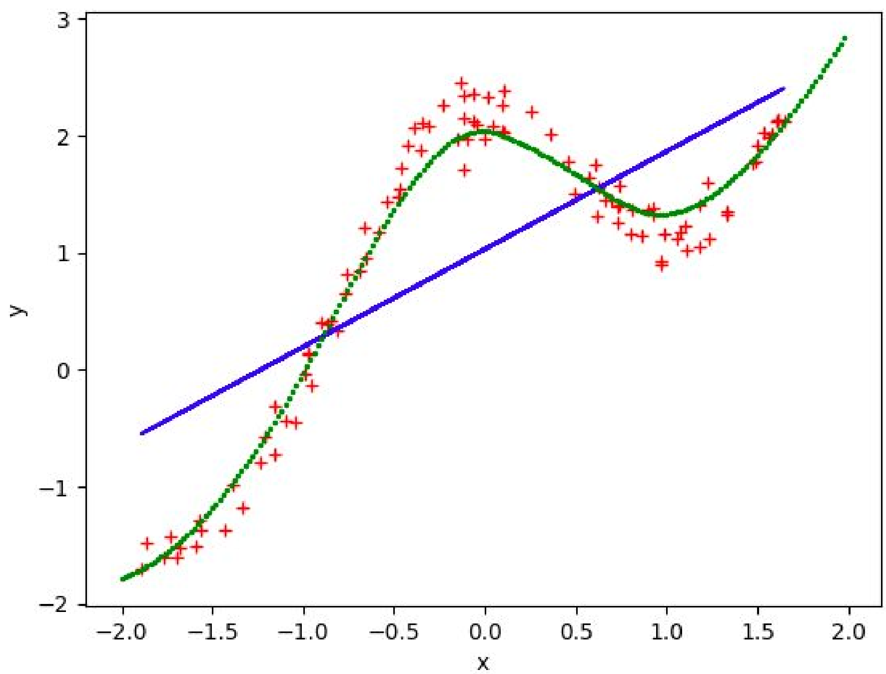


τ = *0.8*



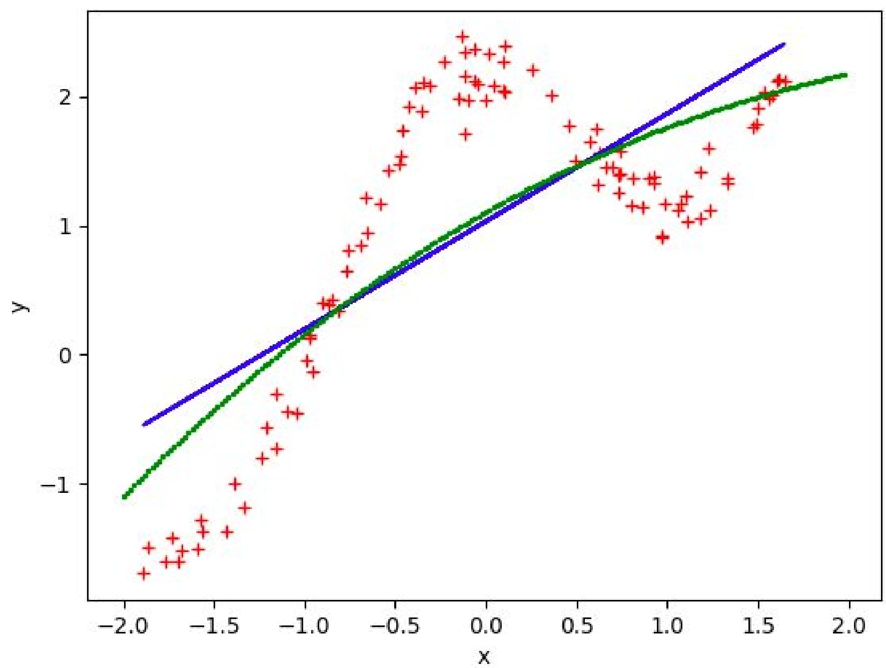
y
x
2
1
0
−1
−2.0
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
2.0

***τ* = 0.1**

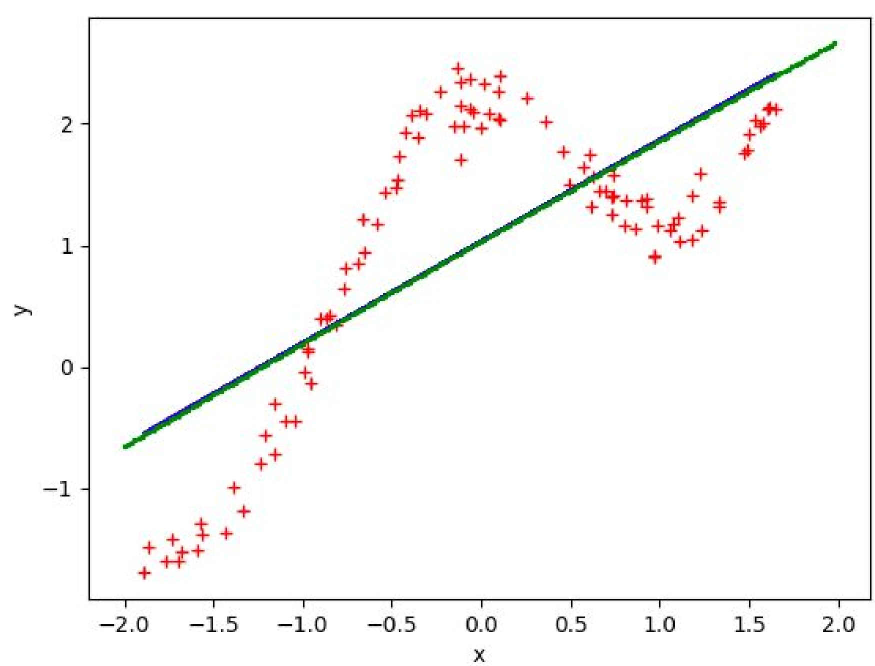


y
x
3
2
1
0
−1
−2
−2.0
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
2.0

***τ* = 0.3**



$\tau = 2$



$\tau = 10$

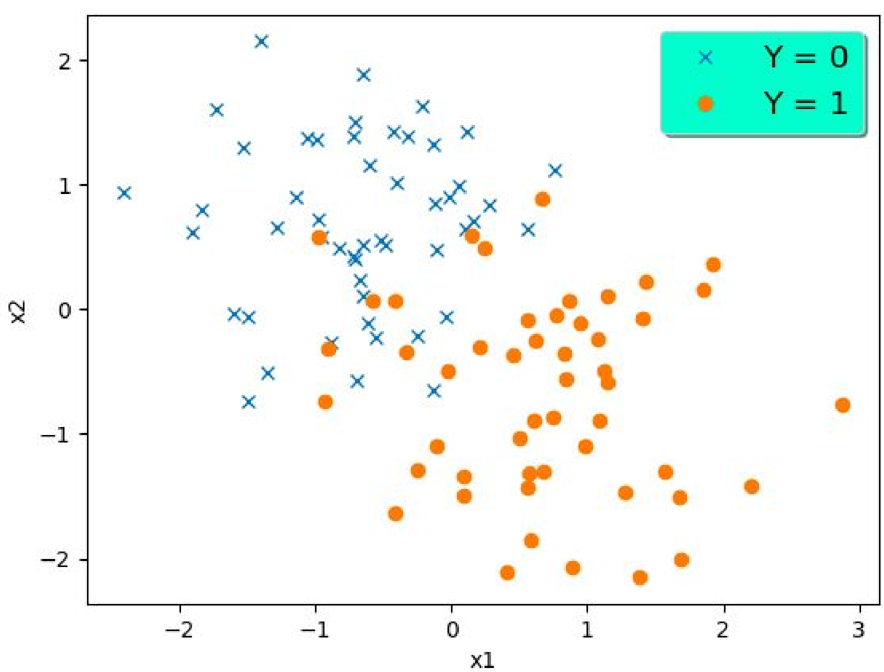
The value of $\boldsymbol{\tau}$ = 0.3 is best as the curve is smooth and fits data quite accurately. Problem with $\boldsymbol{\tau}$ = 0.1 is it overfits the data. For higher values of $\boldsymbol{\tau}$, it underfits data.

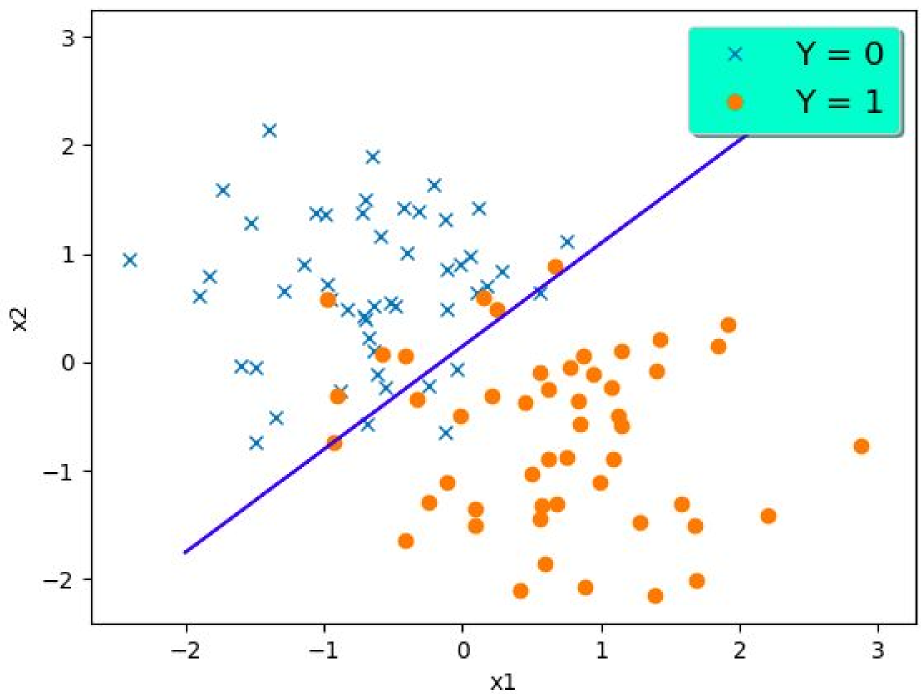
3.

Implemented logistic regression using Newton's method. The input data is normalized before performing any operation. The file **logisticRegression.py** takes epsilon as parameter which is used in stopping condition -

Stopping Condition =(norm($\Theta^{(t)}$ - $\Theta^{(t+1)}$) <= epsilon)

Θ_{init} = [0 0 0]
$\boldsymbol{\epsilon}$ = 0.000001
Θ obtained using Newton's method = [0.40125316 2.5885477 -2.72558849]
No. of iterations = 8





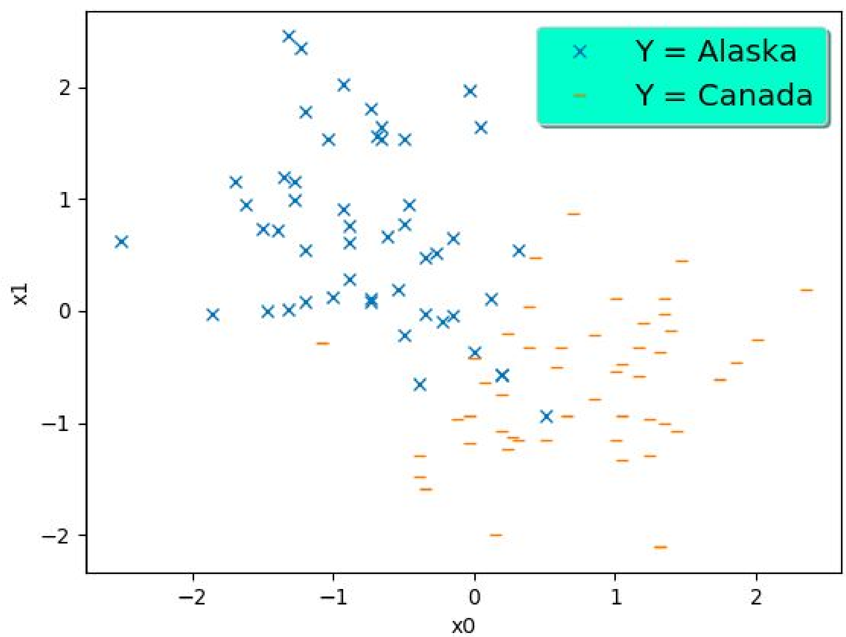
4) a)

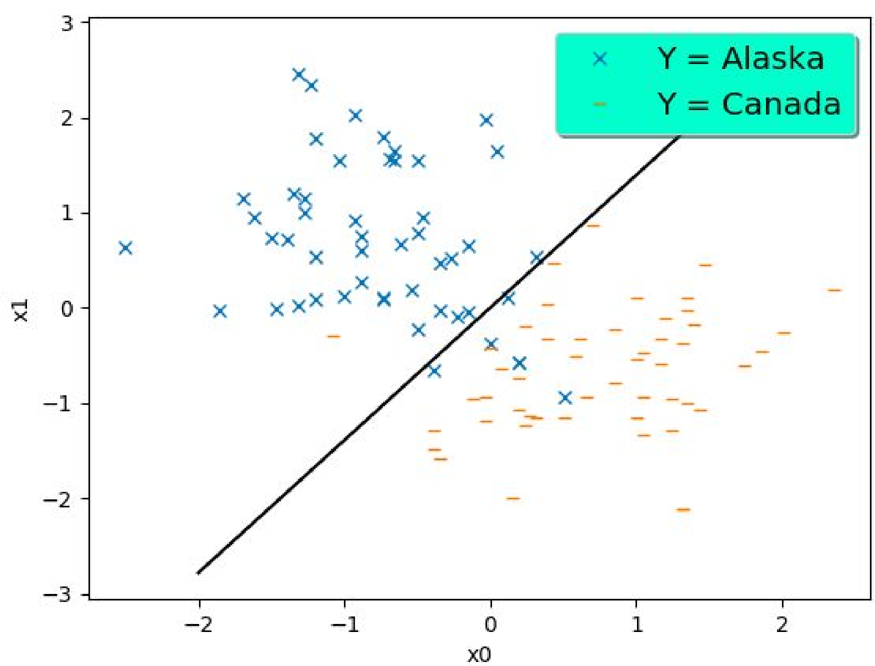
Implemented logistic regression using Newton's method. The input data is normalized before performing any operation. Obtained values using same covariance matrix =

$\boldsymbol{\mu}_0$ = [-0.75529433 0.68509431]
$\boldsymbol{\mu}_1$ = [0.75529433 -0.68509431]
ϕ = 0.5
$\boldsymbol{\Sigma}$ = [[0.42953048 -0.02247228]
[-0.02247228 0.53064579]]

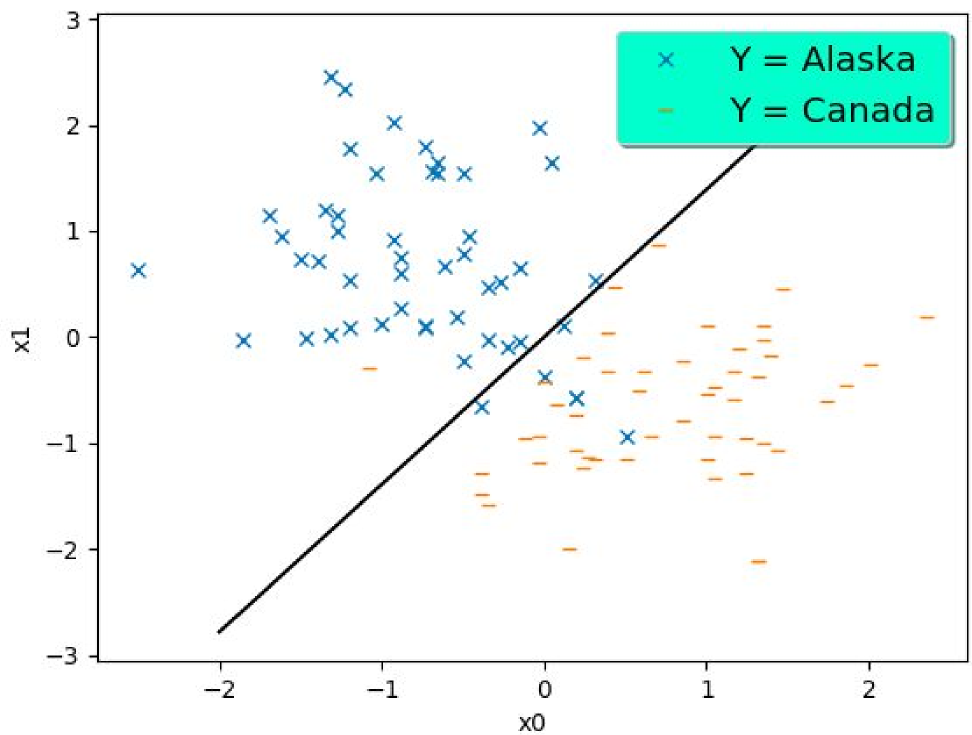
b)

Corresponding plots are -


Y = Alaska
Y = Canada
x1
x0
2
1
0
−1
−2
−2
−1
0
1
2


Y = Alaska
Y = Canada
x1
x0
3
2
1
0
−1
−2
−3
−2
−1
0
1
2

c) $(\mu_0^T\Sigma^{-1} - \mu_1^T\Sigma^{-1})\,X + X^T(\Sigma^{-1}\mu_0 - \Sigma^{-1}\mu_1) = \mu_0^T\Sigma^{-1}\mu_0 - \mu_1^T\Sigma^{-1}\mu_1 + 2ln(\phi\ /\ 1 - \phi)$



d)

$\boldsymbol{\mu}_0$ = [-0.75529433 0.68509431]

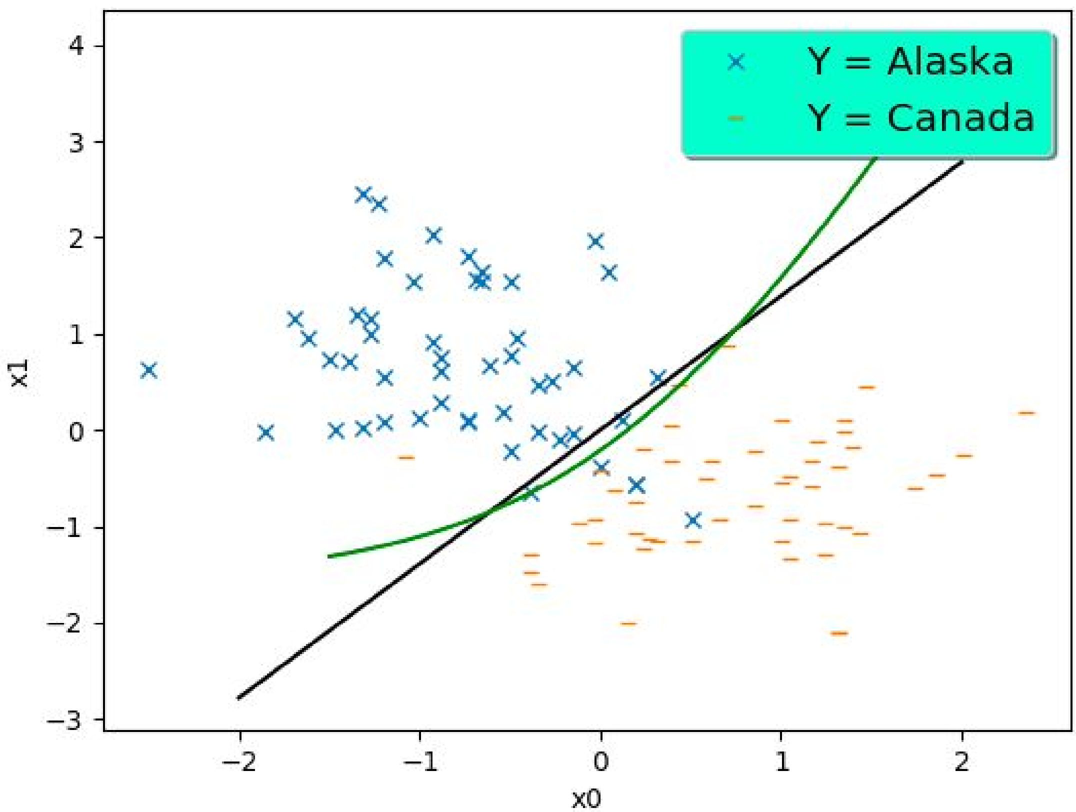
$\boldsymbol{\mu}_1$ = [0.75529433 -0.68509431]

ϕ = 0.5

$\boldsymbol{\Sigma}_0$ = [[0.38158978 -0.15486516]
[-0.15486516 0.64773717]]

$\boldsymbol{\Sigma}_1$ = [[0.47747117 0.1099206]
[0.1099206 0.41355441]]

e) $X^T\Sigma_1^{-1}X - X^T\Sigma_0^{-1}X + (\mu_0^T\Sigma_0^{-1}X - \mu_1^T\Sigma_1^{-1}X) + (X^T\Sigma_0^{-1}\mu_0 - X^T\Sigma_1^{-1}\mu_1) =$
$2ln(\phi / 1 - \phi) + ln(|\Sigma_0| / |\Sigma_1|) + (\mu_0^T\Sigma_0^{-1}\mu_0 - \mu_1^T\Sigma_1^{-1}\mu_1)$



f) As shown in figure below the curve turns out to be hyperbolic for the general setting. The curve gives a better fit than the linear and can be used to separate non linear data. If general setting gives a more accurate curve than it can be inferred that data is distributed with different covariances and can't be separated using linear boundary by considering same covariance.