Machine Learning Assignment 1

(Gulshan Jangid, 2014CS50285)

Question 1:

(A)

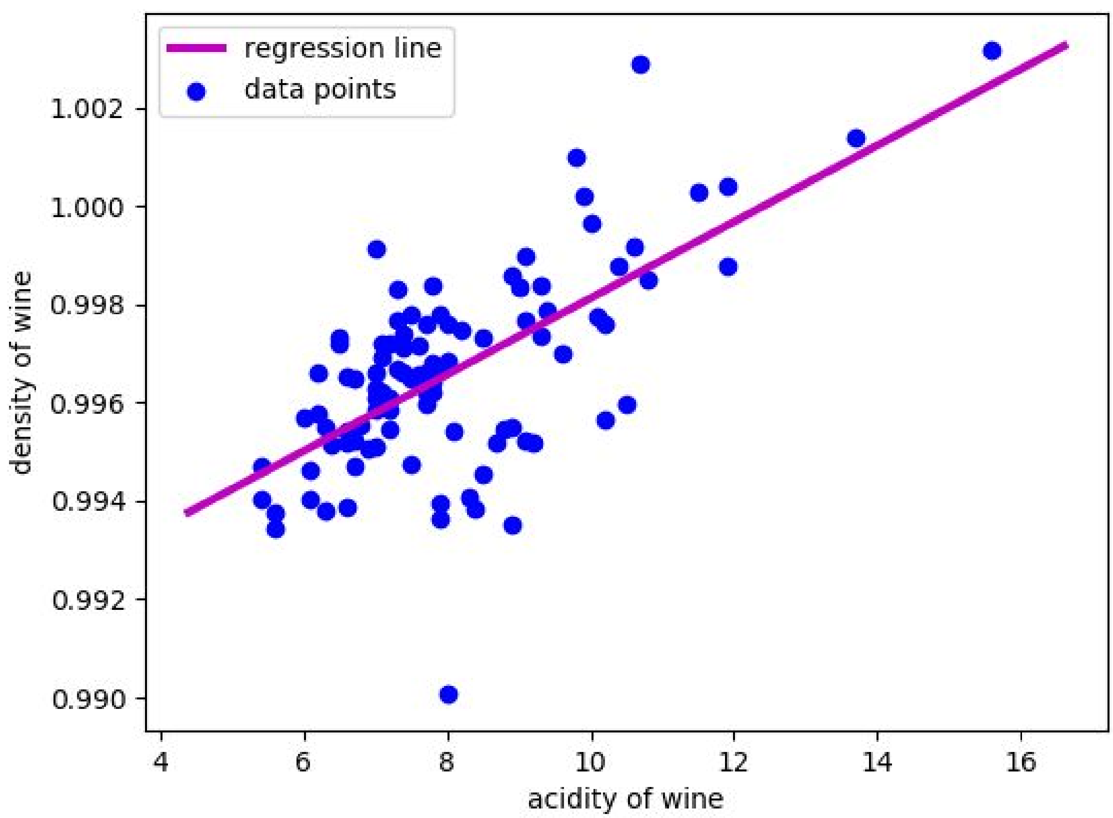
learning rate = 0.0295

Stopping criteria = |max attribute of (current theta-previous theta)| <= 10^{-10}

Runtime $\sim$ 6 seconds

Final θ = [0.9903495, 0.0007778]^T

(B)



(E)

When η is is small then regression takes long time to reach the minima, when η is large then overshooting problem happens and θ goes on increasing and increasing until overflow. In my case overshooting problem happens when $\eta > 0.296$ approx

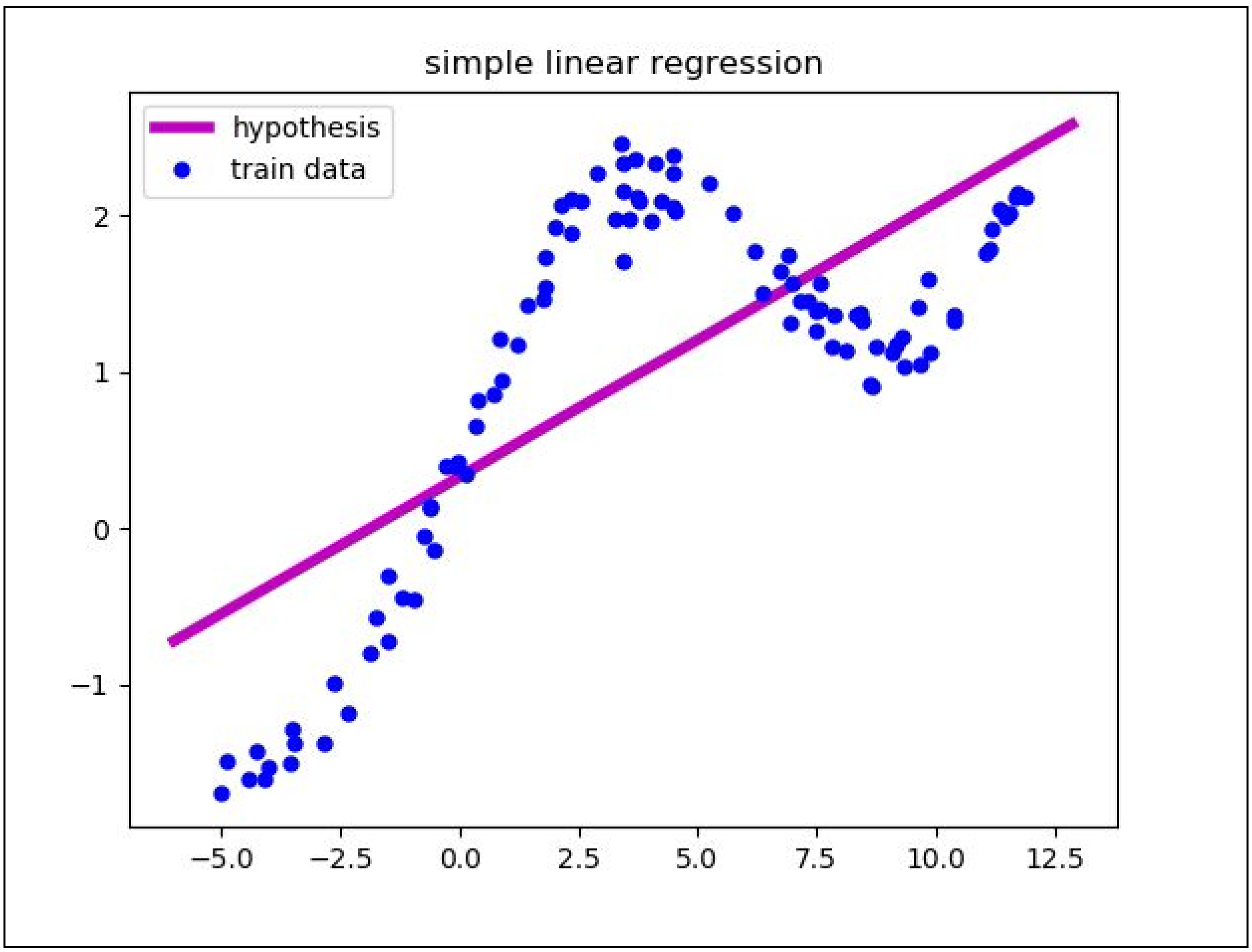
Question 2:

New value of θ that minimizes $J(\theta)$:

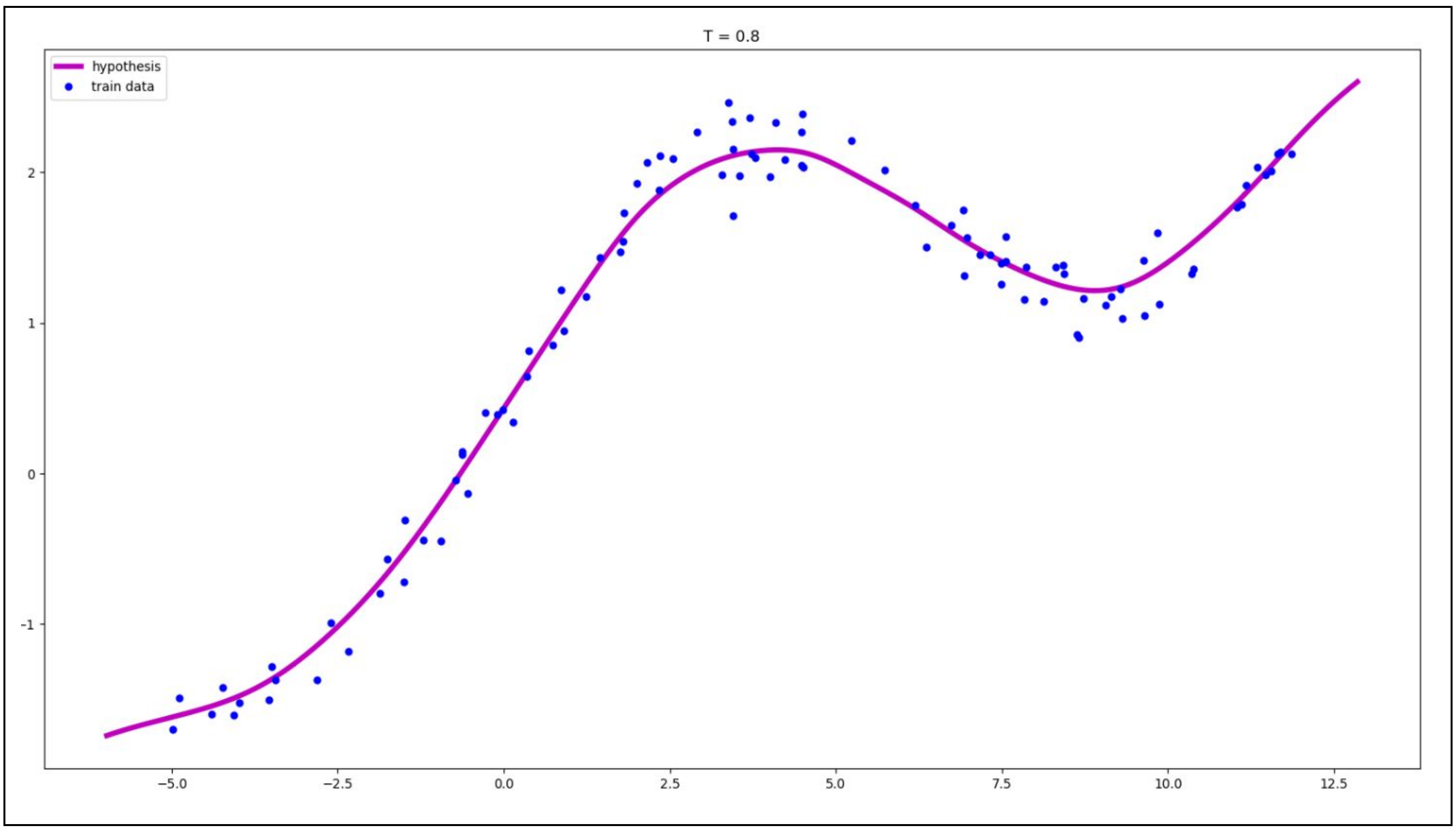
$$\boldsymbol{\theta = (X^T W X)^{-1} X^T W Y}$$

(A)

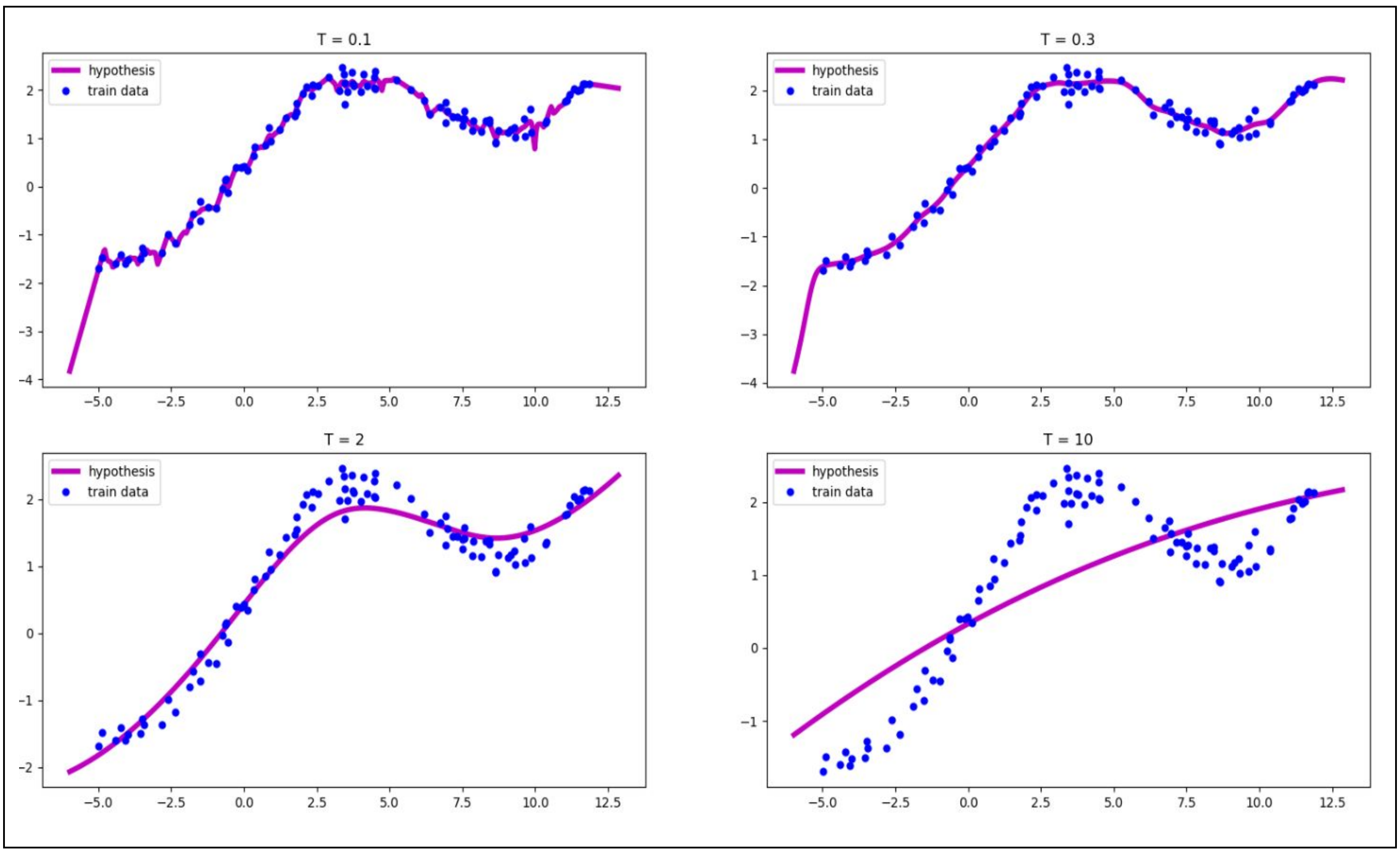
$\theta = (X^T X)^{-1} X^T Y$, here every row of X is $(x^i)^T$



(B)



(C)



$$\omega^{(i)} = \exp\left(-\left(x-x^{(i)}\right)^2 / 2\tau^2\right)$$

When τ approaches a large value, $\omega^{(i)}$ essentially becomes 1 which implies that we are assigning equal weights to all the values, i.e. including points which are not useful, which is what we did in simple linear regression case. So the plot tends to move towards like normal linear regression
When τ is too small then the curve overfits,when evaluating W matrix for x very close to x(i) ,only significant term will be W[i][i] all other terms will be close to zero so best line for this case will be one which passes exactly through point x(i). So overfitting happens when τ is too small
In the above given 5 values for τ , 0.8 fits the best followed by 0.3 and then 2

Question 3:

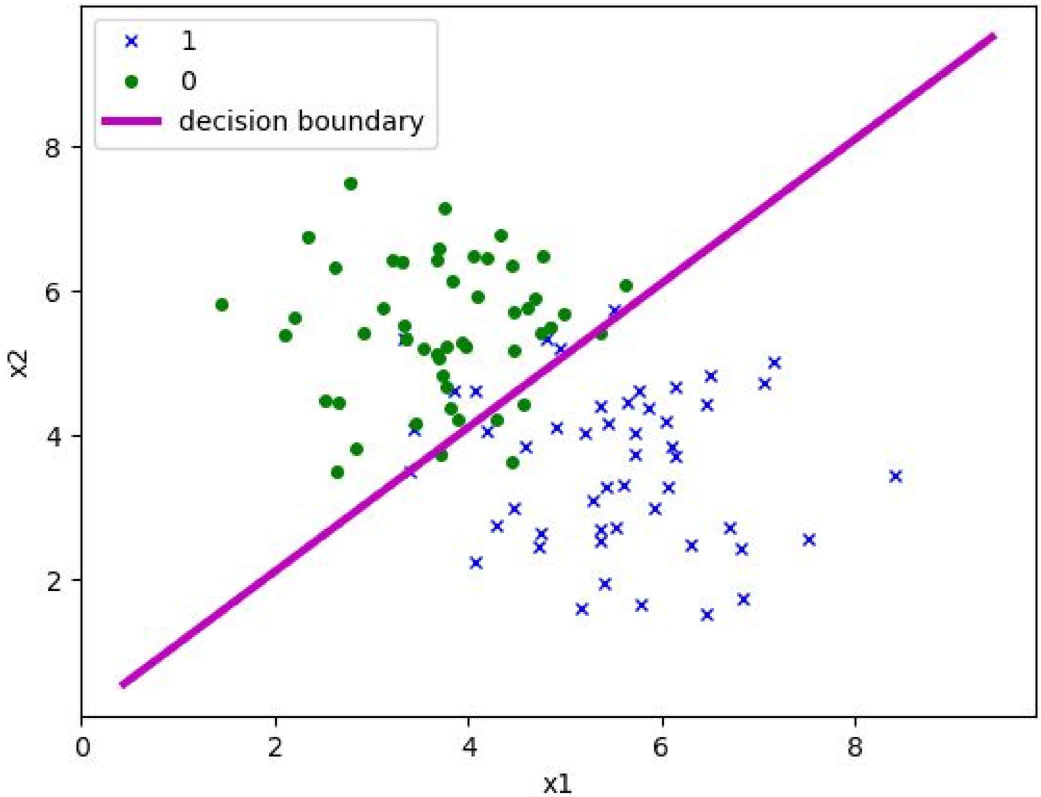
$$\ell(\theta) = \sum_{i=1}^{m} y^{(i)} \log h_\theta(x^{(i)}) + (1 - y^{(i)}) \log(1 - h_\theta(x^{(i)}))$$

$$\begin{aligned}
\frac{\partial}{\partial \theta_j} \ell(\theta) &= \left(y \frac{1}{g(\theta^T x)} - (1-y) \frac{1}{1 - g(\theta^T x)} \right) \frac{\partial}{\partial \theta_j} g(\theta^T x) \\
&= \left(y \frac{1}{g(\theta^T x)} - (1-y) \frac{1}{1 - g(\theta^T x)} \right) g(\theta^T x)(1 - g(\theta^T x) \frac{\partial}{\partial \theta_j} \theta^T x \\
&= \left(y(1 - g(\theta^T x)) - (1-y) g(\theta^T x) \right) x_j \\
&= (y - h_\theta(x)) x_j
\end{aligned}$$

Gradient matrix is calculated from the above formula

θ resulting from the fit is:

[[0.22329537]
 [1.96261552]
 [-1.9648612]]



Question 4:

(A) μ_0 = [[137.46],
[366.62]]

μ_1 = [[98.38],
[429.66]]

Σ = [[287.482, -26.748],
[-26.748, 1123.25]]

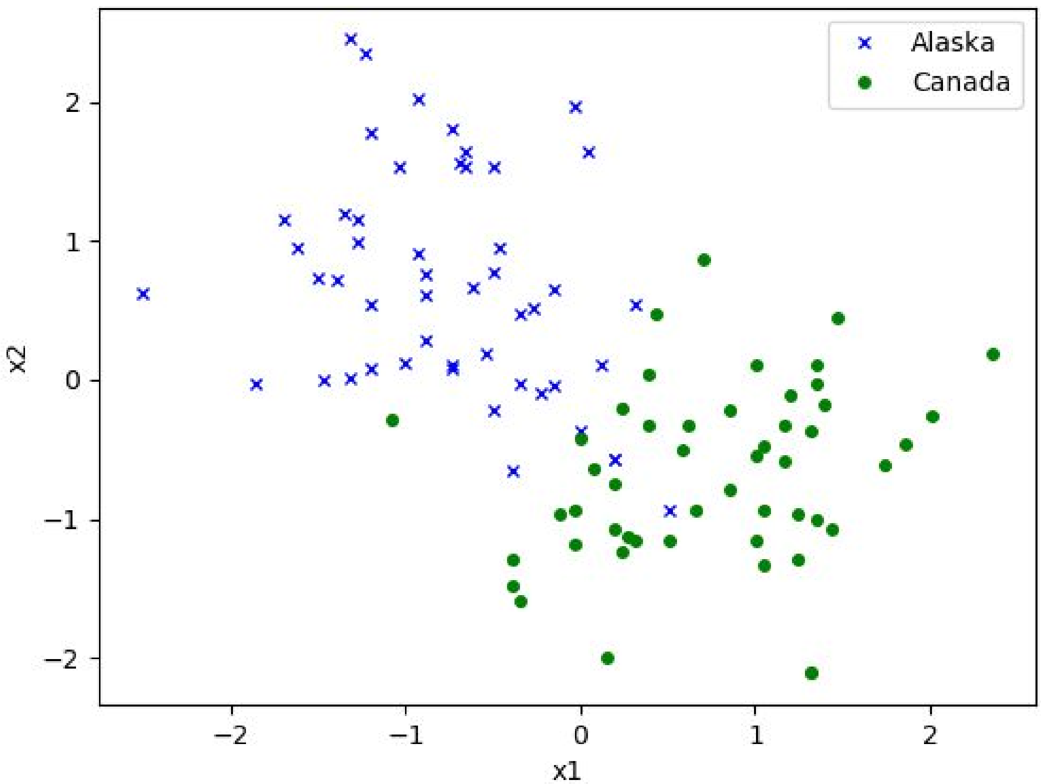
For normalized data:

μ_0 = [[0.75529433],
[-0.68509431]]

μ_1 = [[-0.75529433],
[0.68509431]]

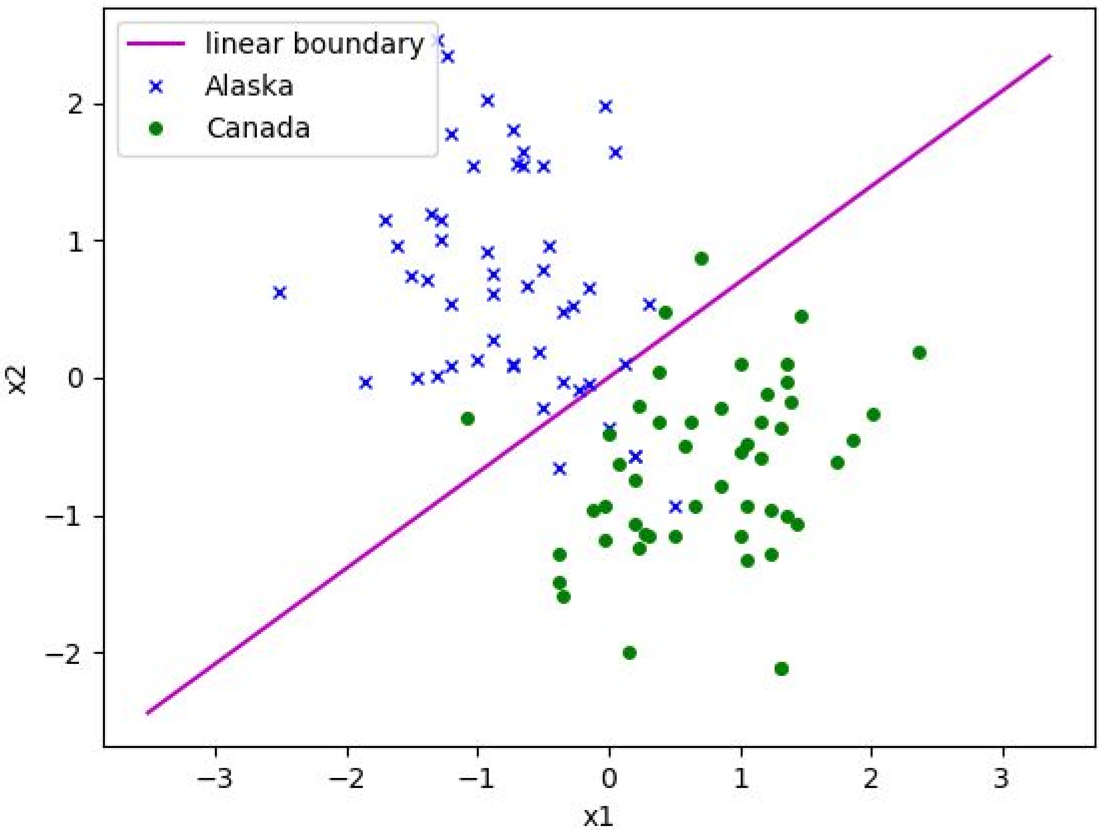
Σ = [[0.42953048, -0.02247228],
[-0.02247228, 0.53064579]]

(B)



(C)

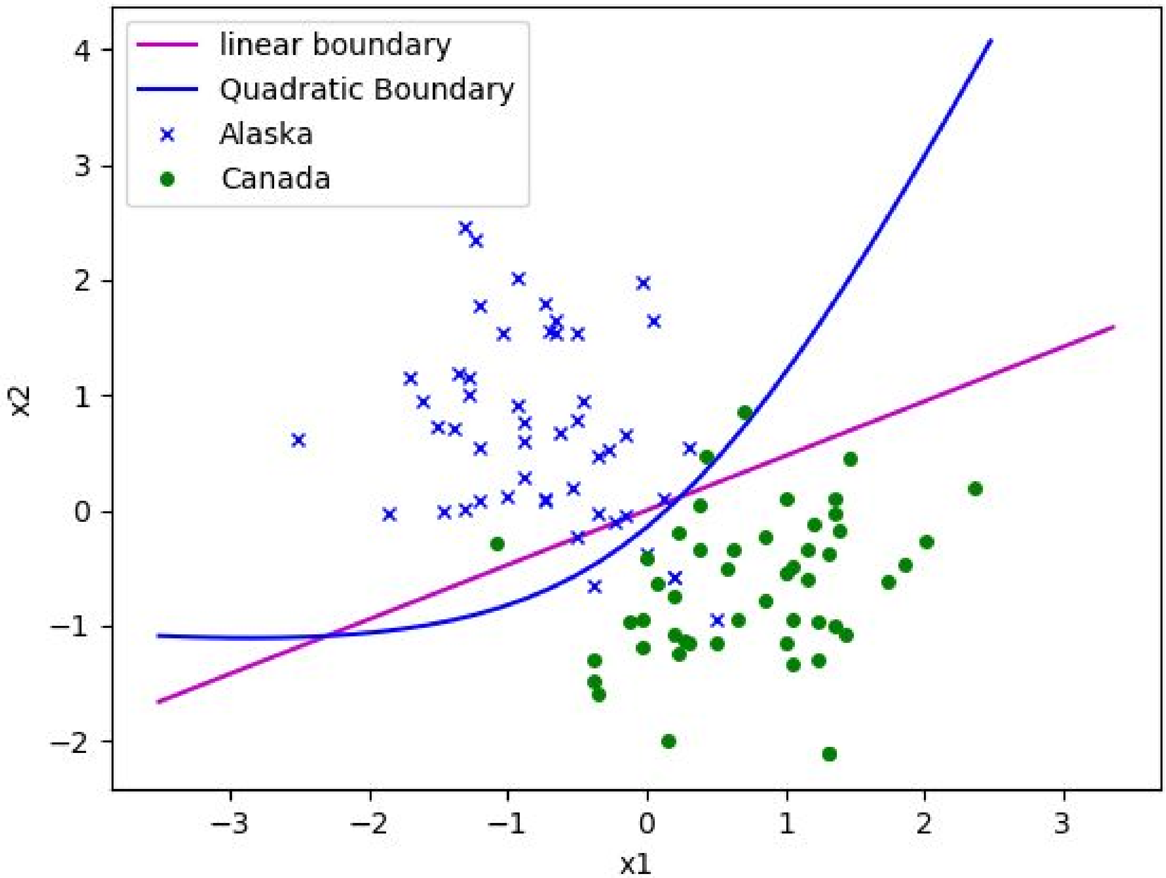
(D)



Equation of the linear separator:

$$x_2 = (½)\,(2.664+3.389*x_1)/(2.438)$$

(D)



Quadratic boundary equation:

$$-\tfrac{1}{2}\log|\Sigma_0| - \tfrac{1}{2}(x-\mu_0)^T \Sigma_0^{-1}(x-\mu_0)+\log(p_0) = -\tfrac{1}{2}\log|\Sigma_1| - \tfrac{1}{2}(x-\mu_1)^T \Sigma_1^{-1}(x-\mu_1)+\log(p_1)$$

Where $p_0 = \phi$ and $p_1=1-p_0$

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

(F)

Ones ('Alaska') in this dataset are negatively correlated.

Linear boundary assumes that Σ is same for both while its not so in this case, so quadratic boundary gives good result compared to linear boundary