CS 229 Autumn 2018 Problem Set #1

Ruining Li
University of Oxford

Problem 1

(a) Recall that the average empirical loss for logistic regression is

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

where $x^{(1)} \in \mathbb{R}^2$ is the input vector of a single training data point.

Differentiate $J(\theta)$ to get the gradient of the loss function:

$$\begin{aligned}\nabla_\theta J(\theta) &= -\frac{1}{m}\sum_{i=1}^{m} y^{(i)}(1-h_\theta(x^{(i)}))x^{(i)} - (1-y^{(i)})h_\theta(x^{(i)})x^{(i)} \quad \text{by the chain rule} \\ &= -\frac{1}{m}\sum_{i=1}^{m}(y^{(i)} - h_\theta(x^{(i)}))x^{(i)}.\end{aligned}$$

Hence, the Hessian H is given by

$$H = \frac{1}{m}\sum_{i=1}^{m} h_\theta(x^{(i)})(1-h_\theta(x^{(i)}))x^{(i)}(x^{(i)})^T.$$

For any vector z,

$$\begin{aligned}z^T H z &= \frac{1}{m}\sum_{i=1}^{m} h_\theta(x^{(i)})(1-h_\theta(x^{(i)}))\left\{z^T x^{(i)}(x^{(i)})^T z\right\} \\ &= \frac{1}{m}\sum_{i=1}^{m} h_\theta(x^{(i)})(1-h_\theta(x^{(i)}))\left\{\left(z^T x^{(i)}\right)^2\right\} \\ &\geq 0.\end{aligned}$$

It follows that H is positive semidefinite and J is convex.

(b) Please see the code for a detailed implementation.

(c) The posterior distribution (we drop the parameters on the left-hand side to keep the notation uncluttered)

$$\begin{aligned}p(y=1|x) &= \frac{p(x|y=1;\mu_1,\Sigma)p(y=1;\phi)}{p(x|y=1;\mu_1,\Sigma)p(y=1;\phi) + p(x|y=0;\mu_0,\Sigma)p(y=0;\phi)} \\ &= \frac{\exp\left(-\frac{1}{2}(x-\mu_1)^T\Sigma^{-1}(x-\mu_1)\right)\phi}{\exp\left(-\frac{1}{2}(x-\mu_1)^T\Sigma^{-1}(x-\mu_1)\right)\phi + \exp\left(-\frac{1}{2}(x-\mu_0)^T\Sigma^{-1}(x-\mu_0)\right)(1-\phi)} \\ &= \frac{1}{1+\exp\left(-(\theta^T x + \theta_0)\right)}\end{aligned}$$

where

$$\theta = \Sigma^{-1}(\mu_1 - \mu_0)$$
$$\theta_0 = \frac{1}{2}(\mu_0^T \Sigma^{-1} \mu_0 - \mu_1^T \Sigma^{-1} \mu_1) - \log \frac{1-\phi}{\phi}.$$

This indicates that GDA results in a classifier that has a linear decision boundary.

(d) The log-likelihood of the data is

$$\begin{aligned}
\ell(\phi, \mu_0, \mu_1, \Sigma) &= \log \prod_{i=1}^{m} p(x^{(i)}|y^{(i)}; \mu_0, \mu_1, \Sigma) p(y^{(i)}; \phi) \\
&= \sum_{i=1}^{m} \log p(x^{(i)}|y^{(i)}; \mu_0, \mu_1, \Sigma) + \log p(y^{(i)}; \phi) \\
&= \sum_{i=1}^{m} -y^{(i)} \left(\frac{1}{2} \log(2\pi\Sigma) + \frac{(x^{(i)} - \mu_1)^2}{2\Sigma} \right) - (1 - y^{(i)}) \left(\frac{1}{2} \log(2\pi\Sigma) + \frac{(x^{(i)} - \mu_0)^2}{2\Sigma} \right) \\
&\quad + y^{(i)} \log \phi + (1 - y^{(i)}) \log(1 - \phi).
\end{aligned}$$

To maximize ℓ, we differentiate ℓ w.r.t ϕ and set the partial derivative to 0:

$$\frac{\partial \ell}{\partial \phi} = \sum_{i=1}^{m} \left\{ \frac{y^{(i)}}{\phi} - \frac{1 - y^{(i)}}{1 - \phi} \right\} = 0.$$

After some algebraic transformation, we obtain that ϕ is indeed as given in the formula above.

Next, we differentiate ℓ w.r.t μ_0 and set the partial derivative to 0:

$$\frac{\partial \ell}{\partial \mu_0} = \sum_{i=1}^{m} \left\{ (1 - y^{(i)}) \frac{x^{(i)} - \mu_0}{\Sigma} \right\} = 0.$$

After some algebraic transformation, we obtain that μ_0 is indeed as given in the formula above. Similar results can be obtained for μ_1.

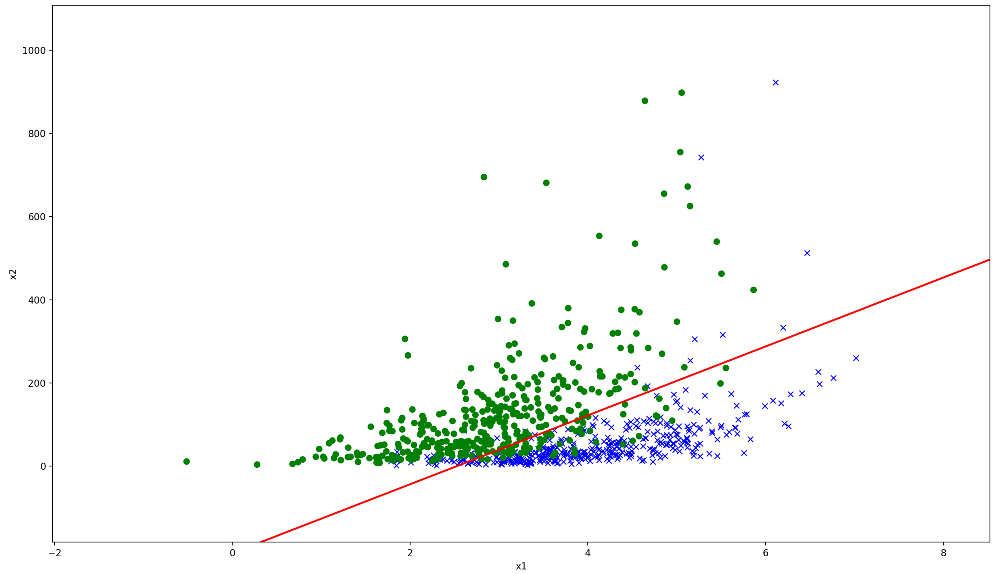
Last, we differentiate ℓ w.r.t Σ and set the partial derivative to 0: (Note here we assume $n = 1$ so that $\Sigma = [\sigma^2]$)

$$\frac{\partial \ell}{\partial \Sigma} = \sum_{i=1}^{m} \left\{ -y^{(i)} \left(\frac{1}{2\Sigma} - \frac{(x^{(i)} - \mu_1)^2}{2\Sigma^2} \right) - (1 - y^{(i)}) \left(\frac{1}{2\Sigma} - \frac{(x^{(i)} - \mu_0)^2}{2\Sigma^2} \right) \right\} = 0.$$

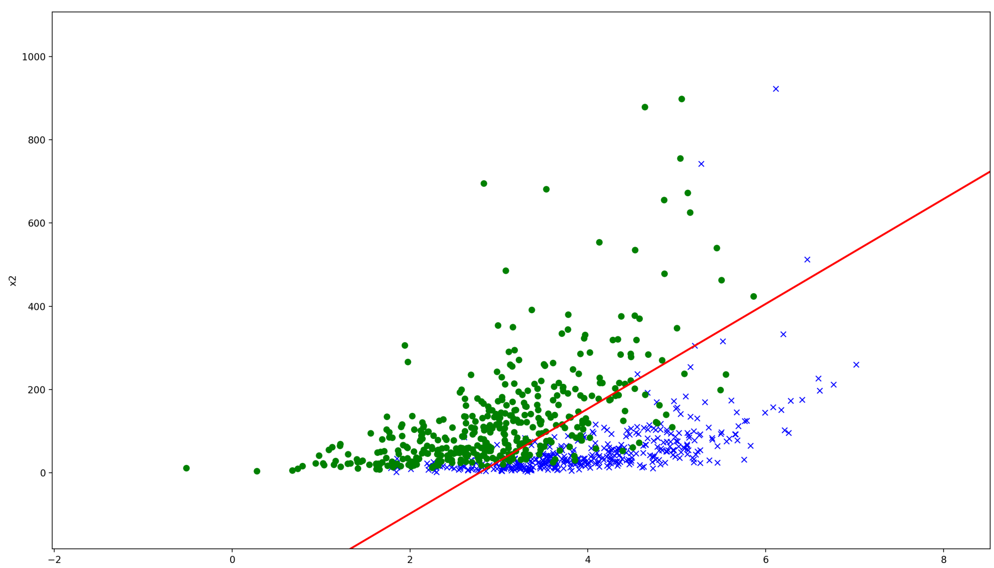
After solving Σ, we observe that the solution of Σ is consistent with the formula above for the special case $n = 1$.

(e) Please see the code for a detailed implementation.

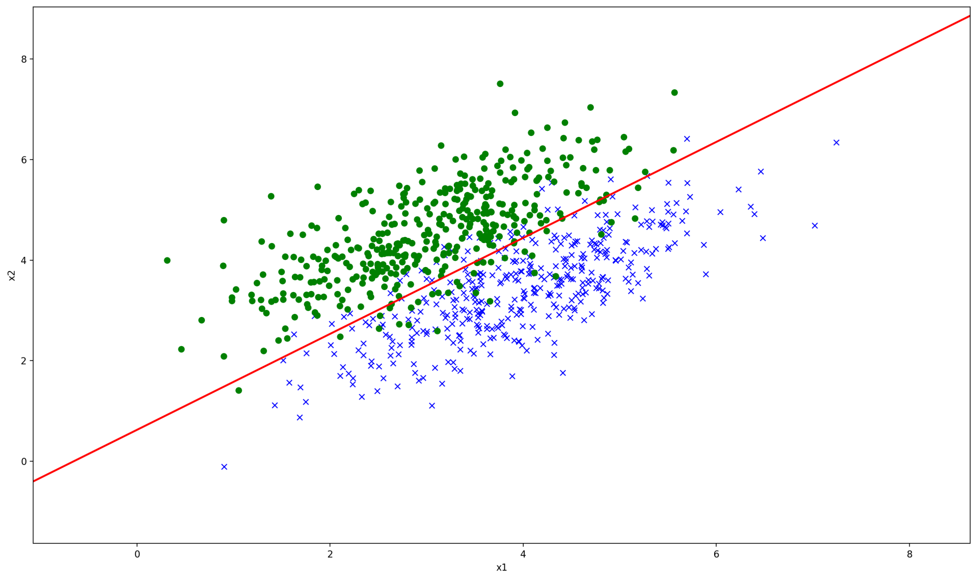
(f) The training data and the decision boundary of Dataset 1 found by logistic regression:



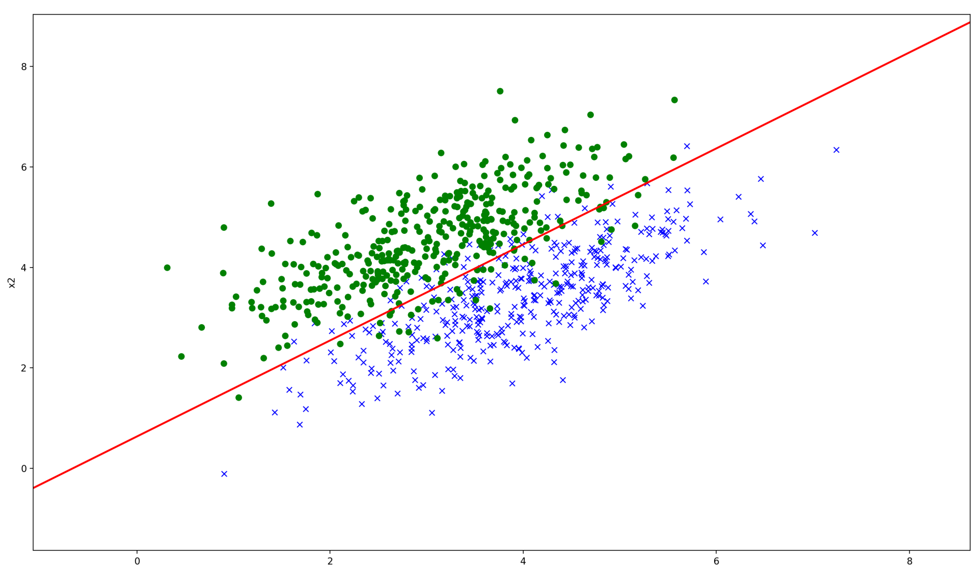
The training data and the decision boundary of Dataset 1 found by GDA:



(g) The training data and the decision boundary of Dataset 2 found by logistic regression:



The training data and the decision boundary of Dataset 2 found by GDA:



GDA seems to perform worse than logistic regression on Dataset 1. The reason might be that the data points of each class in Dataset 1 are not Gaussian-distributed. Therefore, the assumption of GDA is very far from the reality, resulting in worse classification performance.

Problem 2

(a) By the definition of conditional probability, we have:

$$p(y^{(i)} = 1|x^{(i)}) = \frac{p(y^{(i)} = 1|t^{(i)} = 1, x^{(i)})p(t^{(i)} = 1|x^{(i)})}{p(t^{(i)} = 1|y^{(i)} = 1, x^{(i)})}$$

where the denominator is 1 and the first term in the numerator is equal to $p(y^{(i)} = 1|t^{(i)} = 1)$, which is independent of $x^{(i)}$. $\{$In other words, $\alpha = p(y^{(i)} = 1|t^{(i)} = 1)\}$

(b) By the partition theorem, we have

$$p(y^{(i)} = 1|x^{(i)}) = p(y^{(i)} = 1|t^{(i)} = 1, x^{(i)})p(t^{(i)} = 1|x^{(i)}) + p(y^{(i)} = 1|t^{(i)} = 0, x^{(i)})p(t^{(i)} = 0|x^{(i)})$$

where the second term is 0 because when $t^{(i)} = 0$, $y^{(i)}$ must also be 0.

It follows that

$$h(x^{(i)}) \approx p(y^{(i)} = 1|x^{(i)}) = p(y^{(i)} = 1|t^{(i)} = 1)p(t^{(i)}|x^{(i)}) \approx \alpha$$

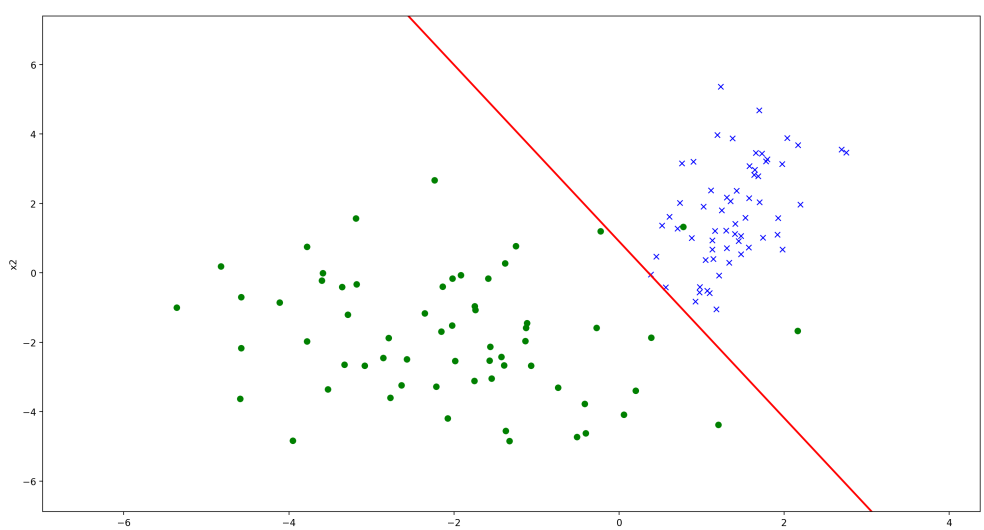
as required.

(c) Please see the code for a detailed implementation.

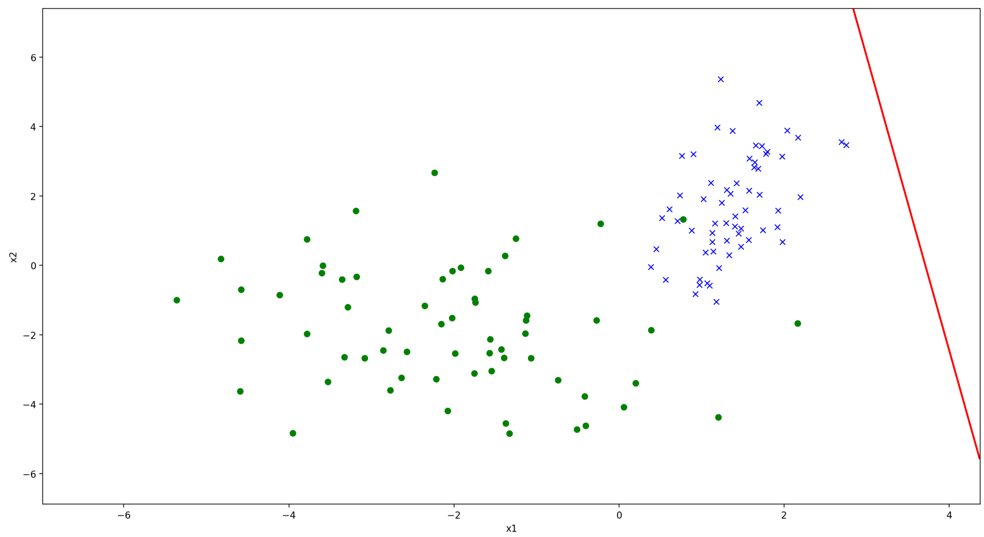
(d) Please see the code for a detailed implementation.

(e) Please see the code for a detailed implementation.

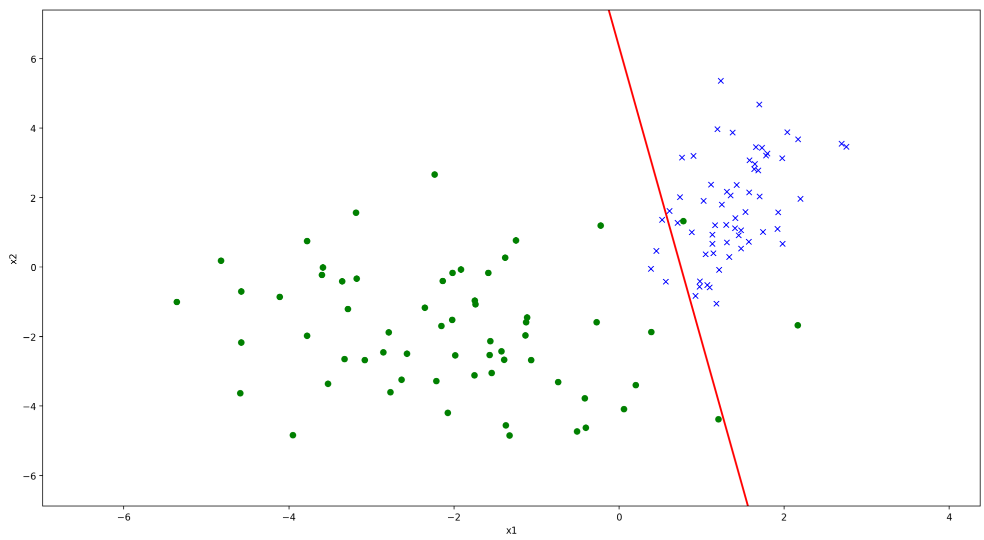
The test data and the decision boundary from part (c):



The test data and the decision boundary from part (d):



The test data and the decision boundary from part (e):



Note that the decision boundaries from part (d) and part (e) are parallel to each other.

Problem 3

(a)

$$p(y;\lambda) = \frac{e^{-\lambda}\lambda^y}{y!} = \exp\{-\lambda + y\log\lambda - \log y!\} = b(y)\exp\left\{\eta^T T(y) - a(\eta)\right\}$$

where

$$\begin{aligned} b(y) &= 1/y! \\ \eta &= \log\lambda \\ T(y) &= y \\ a(\eta) &= \lambda = \exp\eta. \end{aligned}$$

As a result, the Poisson distribution is indeed in the exponential family.

(b) The canonical response function is given by

$$g(\eta) = E[T(y);\eta] = E[y;\eta] = \exp\eta$$

where in the last equation we used the fact that a Poisson random variable with parameter λ has mean λ.

(c) Recall that one of the assumptions of GLM models is $\eta = \theta^T x$.

For the Poisson regression, the log-likelihood (for a single training example) is given by

$$\begin{aligned} \ell &= \log p(y^{(i)}|x^{(i)};\theta) \\ &= \log \frac{e^{-\exp(\theta^T x^{(i)})}\left\{\exp(\theta^T x^{(i)})\right\}^{y^{(i)}}}{y^{(i)}!} \\ &= -\exp(\theta^T x^{(i)}) + y^{(i)}(\theta^T x^{(i)}) - \log y^{(i)}!. \end{aligned}$$

Now we are ready to take the derivative of the log-likelihood with respect to θ:

$$\frac{\partial\ell}{\partial\theta} = -\exp(\theta^T x^{(i)})x^{(i)} + y^{(i)}x^{(i)} = \left\{y^{(i)} - \exp(\theta^T x^{(i)})\right\}x^{(i)} = (y^{(i)} - h_\theta(x^{(i)}))x^{(i)}$$

where $h_\theta(x^{(i)}) = E[y|x;\theta] = \lambda = \exp\eta = \exp(\theta^T x^{(i)})$.

This therefore gives us the stochastic gradient ascent rule

$$\theta := \theta + \alpha(y^{(i)} - h_\theta(x^{(i)}))x^{(i)}$$

(d) Please see the code for a detailed implementation.

Problem 4

(a) We observe that

$$\frac{\partial}{\partial \eta} p(y;\eta) = b(y) \exp\left\{\eta y - a(\eta)\right\} (y - \frac{\partial}{\partial \eta} a(\eta)).$$

Therefore, the mean of the distribution is given by

$$\begin{aligned}
E[y;\eta] &= \int b(y) \exp\left\{\eta y - a(\eta)\right\} y \, dy \\
&= \int \frac{\partial}{\partial \eta} p(y;\eta) \, dy + \int p(y;\eta) \frac{\partial}{\partial \eta} a(\eta) \, dy \\
&= \frac{\partial}{\partial \eta} \int p(y;\eta) \, dy + \frac{\partial}{\partial \eta} a(\eta) \int p(y;\eta) \, dy \\
&= \frac{\partial}{\partial \eta} a(\eta)
\end{aligned}$$

as the probability density function is normalized and integrated to constant 1.

(b) Similar to part (a), we observe that

$$\frac{\partial^2}{\partial \eta^2} p(y;\eta) = b(y) \exp\left\{\eta y - a(\eta)\right\} (y - \frac{\partial}{\partial \eta} a(\eta))^2 - b(y) \exp\left\{\eta y - a(\eta)\right\} \frac{\partial^2}{\partial \eta^2} a(\eta).$$

Therefore, the variance of the distribution is given by

$$\begin{aligned}
E[y;\eta] &= \int b(y) \exp\left\{\eta y - a(\eta)\right\} (y - \frac{\partial}{\partial \eta} a(\eta))^2 \, dy \\
&= \int \frac{\partial^2}{\partial \eta^2} p(y;\eta) \, dy + \int p(y;\eta) \frac{\partial^2}{\partial \eta^2} a(\eta) \, dy \\
&= \frac{\partial^2}{\partial \eta^2} \int p(y;\eta) \, dy + \frac{\partial^2}{\partial \eta^2} a(\eta) \int p(y;\eta) \, dy \\
&= \frac{\partial^2}{\partial \eta^2} a(\eta)
\end{aligned}$$

as required.

(c) The loss function is given by

$$\ell(\theta) = -\log \prod_{(x,y)\in \text{data}} p(y|x;\theta) = -\sum_{(x,y)\in \text{data}} \theta^T x y - a(\theta^T x) + \log b(y).$$

Therefore, the first derivative of the log-likelihood w.r.t. θ is givem by

$$-\sum_{(x,y)\in \text{data}} (y - a'(\theta^T x)) x$$

and the Hessian of the loss is thus

$$\sum_{(x,y)\in\text{data}} a''(\theta^T x)xx^T$$

which is PSD by a similar argument to part (a) of Problem 1.

We conclude that the NLL loss of GLM is convex.

Problem 5

(a) (i) Let W be the diagonal matrix whose i-th entry on the diagonal is $w^{(i)}$. Then, it's easy to show, by the definition of matrix-vector multiplication, that

$$J(\theta) = (X\theta - y)^T W (X\theta - y).$$

(ii) Differentiate $J(\theta)$ w.r.t. θ and set the derivative to 0, we obtain:

$$\frac{\partial J}{\partial \theta} = 2X^T W(X\theta - y) = 0.$$

Note that the 0 on the right-hand side of the equation represents a column vector whose entries are all 0.

This gives us the closed form of θ that minimizes $J(\theta)$:

$$\theta = (X^T W X)^{-1} X^T W y.$$

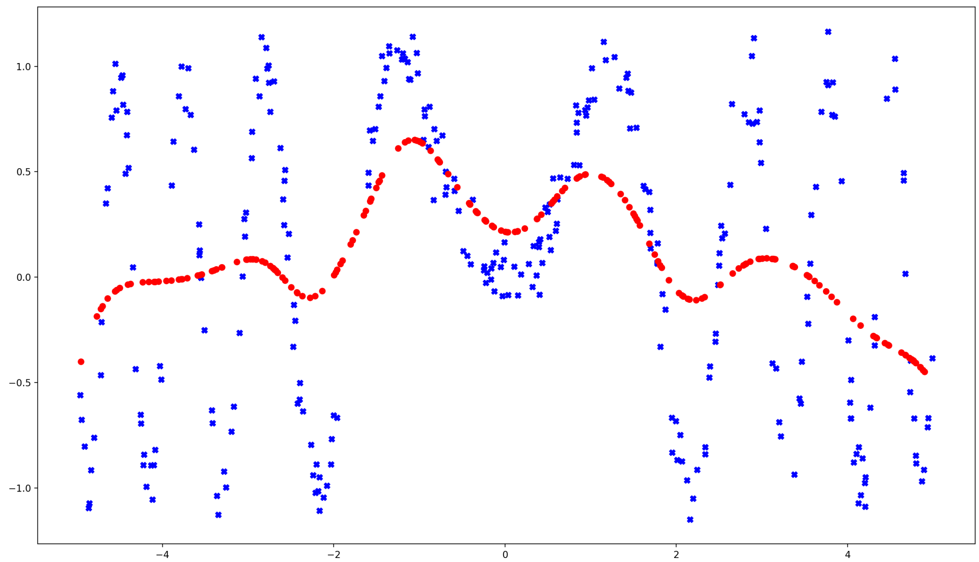
(iii) Maximizing the likelihood function is equivalent to minimizing $\ell(\theta)$, the negative log-likelihood.

$$\begin{aligned}\ell(\theta) &= -\log \prod_{i=1}^{m} \frac{1}{\sqrt{2\pi}\sigma^{(i)}} \exp\left(-\frac{(y^{(i)} - \theta^T x^{(i)})^2}{2(\sigma^{(i)})^2}\right) \\ &= \sum_{i=1}^{m} \frac{(y^{(i)} - \theta^T x^{(i)})^2}{2(\sigma^{(i)})^2} + \log(\sqrt{2\pi}\sigma^{(i)}).\end{aligned}$$

Minimizing $\ell(\theta)$ w.r.t. θ is equivalent to minimizing $J(\theta)$ with $w^{(i)} = \frac{1}{(\sigma^{(i)})^2}$.

(b) Please see the code for a detailed implementation.

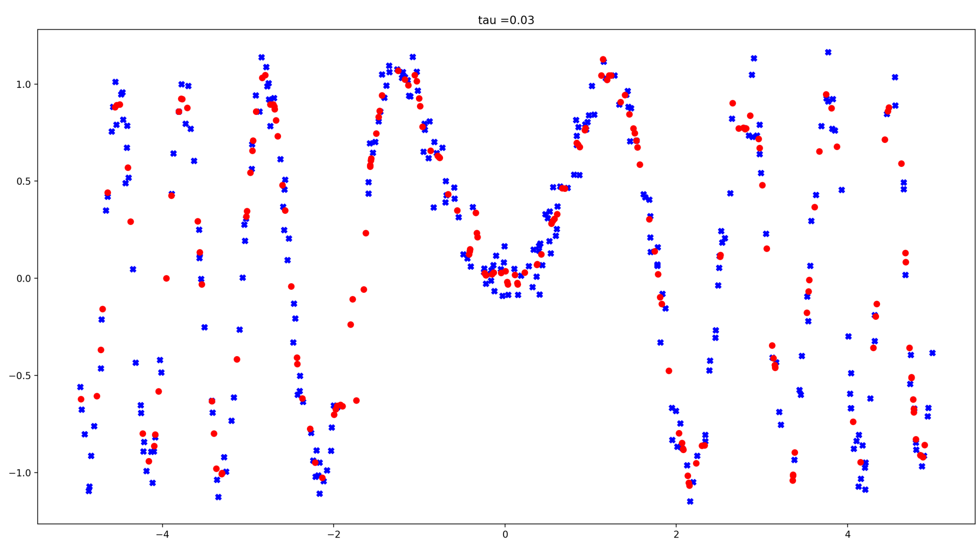
The plot of the data is shown below (blue cross represents training example and red dot represents validation example):

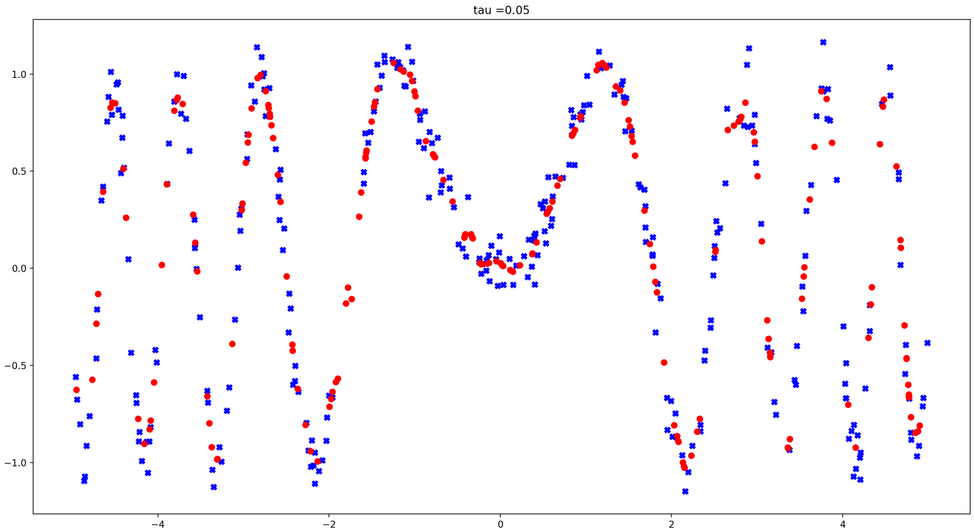


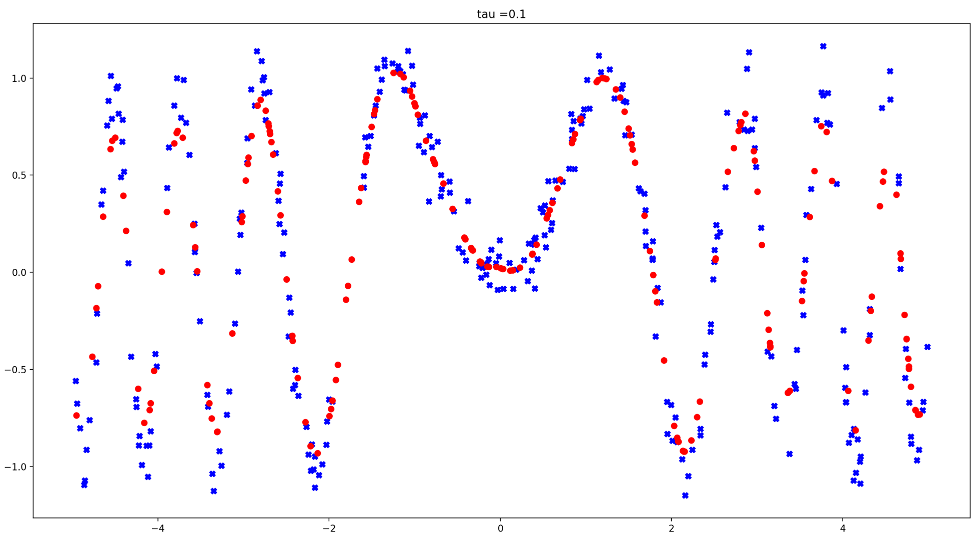
The model seems to be underfitting.

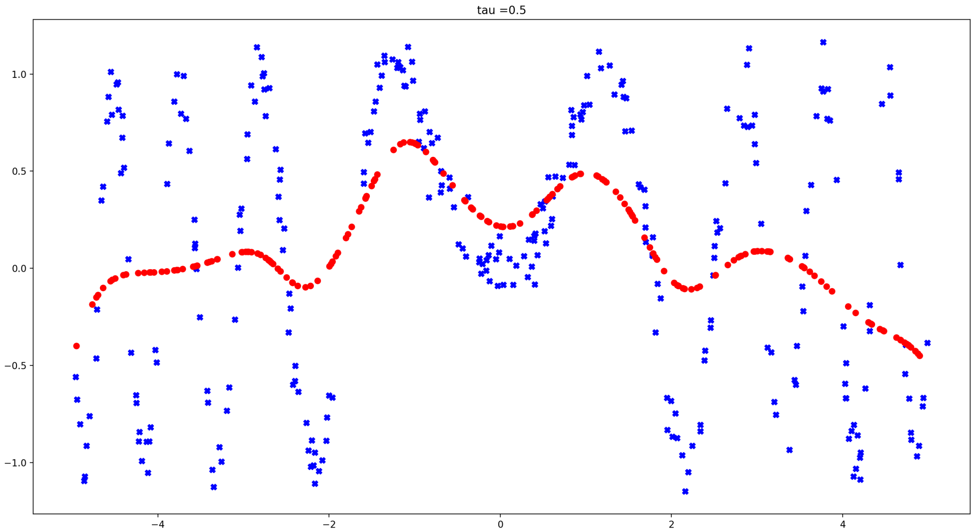
(c) Please see the code for a detailed implementation.

The plots of the data for various values of τ are shown below:

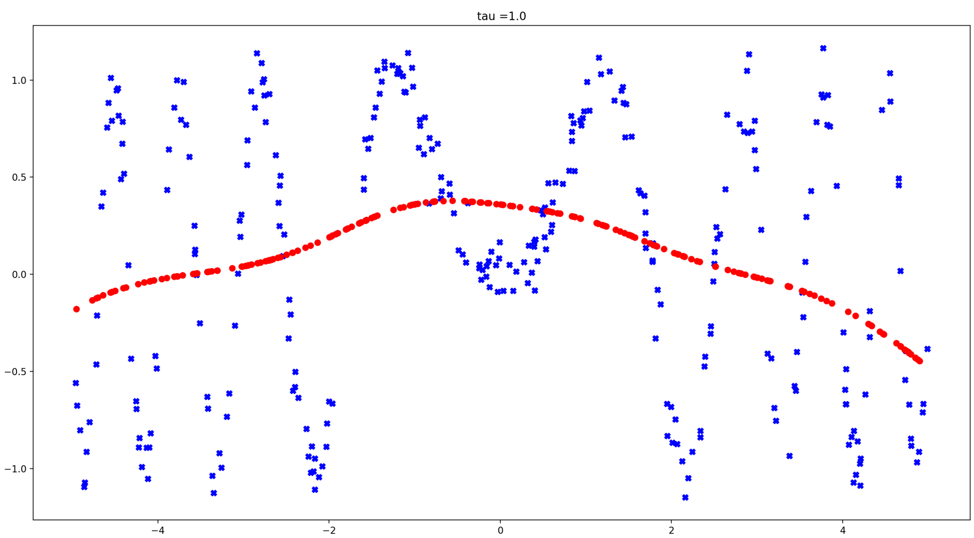



tau =0.05
1.0
0.5
0.0
−0.5
−1.0
−4
−2
0
2
4

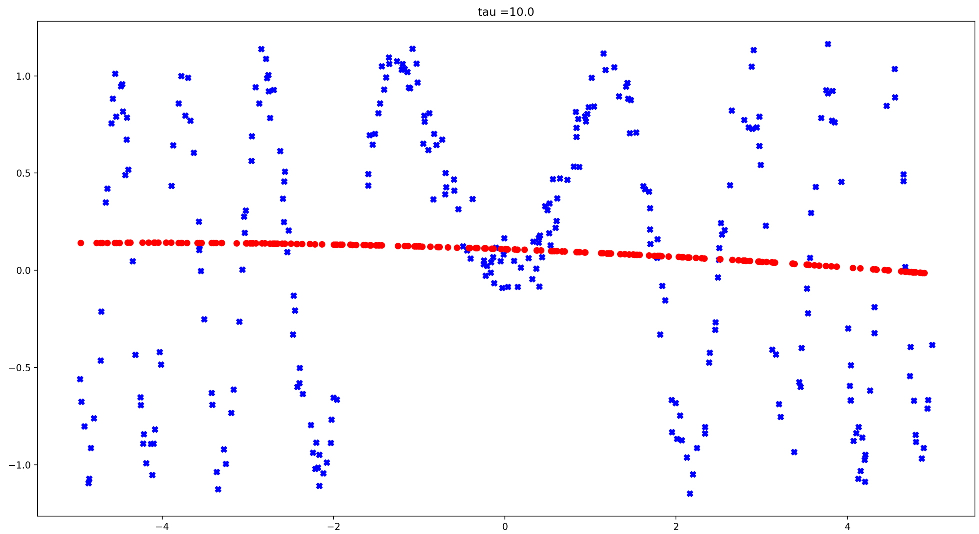

tau =0.1
1.0
0.5
0.0
−0.5
−1.0
−4
−2
0
2
4



tau =0.5
1.0
0.5
0.0
−0.5
−1.0
−4
−2
0
2
4



tau =1.0
1.0
0.5
0.0
−0.5
−1.0
−4
−2
0
2
4



We see that for a smaller τ, the weight is more localized, and therefore the predictions follow more closely with the local patterns of the training data.

The value of τ which achieves the lowest MSE on the `valid` split is 0.05, which produces an MSE of 0.01699 on the `test` split.