# CS282 Machine Learning 2021 Fall
# Quiz 1

## Problem 1

You want to perform a regression task with the following dataset: $x^{(i)} \in \mathbb{R}$ and $y^{(i)} \in \mathbb{R}, i = 1, \ldots, m$ are the $i$ th example and output in the dataset, respectively. Denote the prediction for example $i$ by $f\left(x^{(i)}\right)$ . Remember that for a given loss $\mathcal{L}$, we minimize the following cost function

$$\mathcal{J} = \frac{1}{m} \sum_{i=1}^{m} \mathcal{L}\left(f\left(x^{(i)}\right), y^{(i)}\right)$$

In this part we are deciding between using loss 1 and loss 2 , given by:

$$\mathcal{L}_1\left(f\left(x^{(i)}\right), y^{(i)}\right) = \left|y^{(i)} - f\left(x^{(i)}\right)\right|$$
$$\mathcal{L}_2\left(f\left(x^{(i)}\right), y^{(i)}\right) = \left(y^{(i)} - f\left(x^{(i)}\right)\right)^2$$

(a) An outlier is a data point that differs significantly from other observations. Which method do you think works better when there are some outliers in your dataset?

(b) "Using $\mathcal{L}_1$ loss enforces sparsity on the weights of the network." Do you agree with this statement? Why/Why not?

(c) "Using $\mathcal{L}_2$ loss forces the weights of the network to end up small." Do you agree with this statement? Why/Why not?

**Solution:**

(a) $L_1$ . The reason is it penalizes less for outliers. We would like to ignore outliers if possible. When it isn't possible, using a loss function which penalises outliers less, is more robust.

(b) This is false because in this case, the residual will be forced to be sparse not the weights.

(c) This is false because in this case, the residual will be forced to be small not the weights.

## Problem 2

Consider the dataset $\mathcal{D}$ given in (1) consisting of $n$ independent identically distributed observations. The features of $\mathcal{D}$ consist of pairs $\left(x_1^i, x_2^i\right) \in \mathbb{R}^2$ and the observations $y^i \in \mathbb{R}$ are continuous valued.

$$\mathcal{D} = \left\{ \left( \left(x_1^1, x_2^1\right), y^1 \right), \left( \left(x_1^2, x_2^2\right), y^2 \right), \ldots, \left( \left(x_1^n, x_2^n\right), y^n \right) \right\} \tag{1}$$

Consider the abstract model given in (2) . The function $f_{\theta_1,\theta_2}$ is a mapping from the features in $\mathbb{R}^2$ to an observation in $\mathbb{R}^1$ which depends on two parameters $\theta_1$ and $\theta_2$ . The $\epsilon^i$ correspond to the noise. Here we will assume that the $\epsilon^i \sim N\left(0, \sigma^2\right)$ are independent Gaussians with zero mean and variance $\sigma^2$

$$y^i = f_{\theta_1,\theta_2}\left(x_1^i, x_2^i\right) + \epsilon^i \tag{2}$$

(a) Show that the log likelihood of the data given the parameters is equal to.

$$l\left(D; \theta_1, \theta_2\right) = -\frac{1}{2\sigma^2} \sum_{i=1}^{n} \left(y^i - f_{\theta_1,\theta_2}\left(x_1^i, x_2^i\right)\right)^2 - n \log(\sqrt{2\pi}\sigma)$$

Recall the probability density function of the $N\left(\mu, \sigma^2\right)$ Gaussian distribution is given by

$$p(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$$

(b) Many common techniques used to find the maximum likelihood estimates of $\theta_1$ and $\theta_2$ rely on our ability to compute the gradient of the log-likelihood. Compute the gradient of the log likelihood with respect to $\theta_1$ and $\theta_2$ . Express your answer in terms of:

$$y^i, \quad f_{\theta_1,\theta_2}\left(x_1^i, x_2^i\right), \quad \frac{\partial}{\partial\theta_1} f_{\theta_1,\theta_2}\left(x_1^i, x_2^i\right), \quad \frac{\partial}{\partial\theta_2} f_{\theta_1,\theta_2}\left(x_1^i, x_2^i\right)$$

**Solution:**

(a) The first thing one should think about is the probability of a single data point under this model. We can of course write this as:

$$y^i - f_{\theta_1,\theta_2}\left(x_1^i, x_2^i\right) = \epsilon^i$$

Because we know the distribution of $\epsilon_i$ we know that:

$$y^i - f_{\theta_1,\theta_2}\left(x_1^i, x_2^i\right) \sim N\left(0, \sigma^2\right)$$

We can therefore write the likelihood of the data as:

$$\mathcal{L}\left(D; \theta_1, \theta_2\right) = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{\left(y^i - f_{\theta_1,\theta_2}\left(x_1^i, x_2^i\right)\right)^2}{2\sigma^2}\right)$$

To compute the log likelihood we take the log of the likelihood from above to obtain:

$$\begin{aligned}
l\left(D; \theta_1, \theta_2\right) &= \log\left(\mathcal{L}\left(D; \theta_1\theta_2\right)\right) \\
&= \log\left(\prod_{i=1}^{n} \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{\left(y^i - f_{\theta_1,\theta_2}\left(x_1^i, x_2^i\right)\right)^2}{2\sigma^2}\right)\right) \\
&= \sum_{i=1}^{n} \log\left(\frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{\left(y^i - f_{\theta_1,\theta_2}\left(x_1^i, x_2^i\right)\right)^2}{2\sigma^2}\right)\right) \\
&= \sum_{i=1}^{n} \left(-\log(\sqrt{2\pi}\sigma) - \frac{\left(y^i - f_{\theta_1,\theta_2}\left(x_1^i, x_2^i\right)\right)^2}{2\sigma^2}\right) \\
&= -n\log(\sqrt{2\pi}\sigma) - \frac{1}{2\sigma^2} \sum_{i=1}^{n} \left(y^i - f_{\theta_1,\theta_2}\left(x_1^i, x_2^i\right)\right)^2
\end{aligned}$$

(b) The important technique we use here is the chain rule. To save space I will take the gradient with respect to $\theta_j$.

$$\begin{aligned}
\frac{\partial}{\partial \theta_j} l\left(D; \theta_1, \theta_2\right) &= -\frac{1}{2\sigma^2} \sum_{i=1}^{n} \frac{\partial}{\partial \theta_j} \left(y^i - f_{\theta_1,\theta_2}\left(x_1^i, x_2^i\right)\right)^2 \\
&= -\frac{1}{2\sigma^2} \sum_{i=1}^{n} 2\left(y^i - f_{\theta_1,\theta_2}\left(x_1^i, x_2^i\right)\right) \frac{\partial}{\partial \theta_j} \left(-f_{\theta_1,\theta_2}\left(x_1^i, x_2^i\right)\right) \\
&= \frac{1}{\sigma^2} \sum_{i=1}^{n} \left(y^i - f_{\theta_1,\theta_2}\left(x_1^i, x_2^i\right)\right) \frac{\partial}{\partial \theta_j} f_{\theta_1,\theta_2}\left(x_1^i, x_2^i\right)
\end{aligned}$$

## Problem 3

Assume that you are given the following observations $(x, y) \in \mathbb{R}^2 \times \{\pm 1\}$:

| Instance | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| Data $(x_1, x_2)$ | (-1,-3) | (-2,1) | (0,0) | (2,4) | (3,1) |
| Label $y$ | +1 | -1 | -1 | -1 | +1 |

Show the steps of the perceptron algorithm for the above observations. We start with an initial set of weights $w = (1, 1)$ and bias $b = 0$.

(Hint: Update the weights and bias by selecting the misclassified instance with the smallest index)

**Solution:**

Let us define $g\left(y^{(i)}, w, x^{(i)}, b\right) = y^{(i)}\left(\left\langle w, x^{(i)}\right\rangle + b\right)$ and simplify notation by $g_i = g\left(y^{(i)}, w, x^{(i)}, b\right)$. Rewriting the perceptron algorithm, there is update $w^{(j)} \longleftarrow w^{(j-1)} + y^{(i)}x^{(i)}, b^{(j)} \longleftarrow b^{(j-1)} + y^{(i)}$ if $g_i \leq 0$ and no the update otherwise. Following this algorithm with the starting point $w^{(0)} = (1, 1), b^{(0)} = 0$,

(1) Instance 1 is misclassified $g_1 = +1 \cdot (-1 - 3 + 0) \leq 0$:

$$w^{(1)} = (1, 1) + (-1, -3) = (0, -2)$$
$$b^{(1)} = 0 + 1 = 1$$

(2) Instance 3 is misclassified $g_3 = -1 \cdot (0 + 0 + 1) \leq 0$:

$$w^{(2)} = (0, -2) + (0, 0) = (0, -2)$$
$$b^{(2)} = 1 - 1 = 0$$

(3) Instance 3 is misclassified $g_3 = -1 \cdot (0 + 0 + 0) \leq 0$:

$$w^{(3)} = (0, -2) + (0, 0) = (0, -2)$$
$$b^{(3)} = 0 - 1 = -1$$

(4) Instance 5 is misclassified $g_5 = +1 \cdot (0 - 2 - 1) \leq 0$:

$$w^{(4)} = (0, -2) + (3, 1) = (3, -1)$$
$$b^{(4)} = -1 + 1 = 0$$

(5) Instance 1 is misclassified $g_1 = +1 \cdot (-3 + 3 + 0) \leq 0$:

$$w^{(5)} = (3, -1) + (-1, -3) = (2, -4)$$
$$b^{(5)} = 0 + 1 = 1$$

(6) Instance 3 is misclassified $g_3 = -1 \cdot (0 + 0 + 1) \leq 0$:

$$w^{(6)} = (2, -4) + (0, 0) = (2, -4)$$
$$b^{(6)} = 1 - 1 = 0$$

(7) Instance 3 is misclassified $g_3 = -1 \cdot (0 + 0 + 2) \leq 0$:

$$w^{(7)} = (2, -4) + (0, 0) = (2, -4)$$
$$b^{(7)} = 0 - 1 = -1$$

## Problem 4

Mike loves ShanghaiTech so much that he wants to design an 'Email Filter' that only accepts emails that are related to ShanghaiTech. He decides to use Naive Bayes as the classification model and use the occurrence (whether the word appears in the email or not) of the following words as features. 'Project', 'Vacation', 'Deadline', 'Relaxing'. We denote the class variable as $Y$,

$$Y = \begin{cases} 1, & \text{email is related to ShanghaiTech,} \\ 0, & \text{otherwise} \end{cases}$$

We denote the feature variables as $X_w$, where $w \in \{$ 'Project', 'Vacation', 'Deadline', 'Relaxing' $\}$ and

$$X_w = \begin{cases} 1, & \text{email contains word } w \\ 0, & \text{otherwise} \end{cases}$$

Answer the following questions about supervised Naive Bayes.
(a)Joint Distribution
Write down the expression of joint distribution $P\left(Y, X_{\text{Project}}, X_{\text{Vacation}}, X_{\text{Deadline}}, X_{\text{Relaxing}}\right)$ using $P(Y)$ and $P(X_w|Y)$.(Hint: Naive Bayes)
(b)Supervised Learning of Naive Bayes
Mike annotated (assigning a label for each email) four training samples as shown in the following table.

| $Y$ | $X_{\text{Project}}$ | $X_{\text{Vacation}}$ | $X_{\text{Deadline}}$ | $X_{\text{Relaxing}}$ |
|---|---|---|---|---|
| 1 | 1 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 | 0 |
| 1 | 1 | 1 | 1 | 0 |
| 0 | 0 | 1 | 1 | 1 |

You need to estimate the probability distribution $P(Y)$ and $P\left(X_w \mid Y\right)$, and use it to build a Naive Bayes classifier. Please fill the following table, which represents the parameters of a Naive Bayes classifier. Each entry in the left part of the table represents a conditional probability, for example, the top-left entry represents $P\left(X_{\text{Project}} = 1 \mid Y = 1\right)$. Do not use any smoothing to regularize the probabilities.

| | $X_{\text{Project}} = 1$ | $X_{\text{Vacation}} = 1$ | $X_{\text{Deadline}} = 1$ | $X_{\text{Relaxing}} = 1$ | $P(Y)$ |
|---|---|---|---|---|---|
| $Y = 1$ | | | | | |
| $Y = 0$ | | | | | |

(c)After you estimate the parameters of the Naive Bayes classifier, use it to classify the following test sample:

| $X_{\text{Project}}$ | $X_{\text{Vacation}}$ | $X_{\text{Deadline}}$ | $X_{\text{Relaxing}}$ |
|---|---|---|---|
| 1 | 1 | 1 | 1 |

**Solution:**
(a):

$$\begin{gathered} P\left(Y, X_{\text{Project}}, X_{\text{Vacation}}, X_{\text{Deadline}}, X_{\text{Relaxing}}\right) = P(Y)\Pi_w P\left(X_w \mid Y\right) \\ w \in \{ \text{ 'Project', 'Vacation', 'Deadline, 'Relaxing' } \}. \end{gathered} \tag{3}$$

(b):

$$\begin{gathered} P(Y=1) = 0.5, \quad P(Y=0) = 0.5 \\ P\left(X_{\text{Project}} = 1 \mid Y = 1\right) = 1, \quad P\left(X_{\text{Vacation}} = 1 \mid Y = 1\right) = 0.5 \\ P\left(X_{\text{Deadline}} = 1 \mid Y = 1\right) = 1, \quad P\left(X_{\text{Relaxing}} = 1 \mid Y = 1\right) = 0 \\ P\left(X_{\text{Project}} = 1 \mid Y = 0\right) = 0.5, \quad P\left(X_{\text{Vacation}} = 1 \mid Y = 0\right) = 1 \\ P\left(X_{\text{Deadline}} = 1 \mid Y = 0\right) = 0.5, \quad P\left(X_{\text{Relaxing}} = 1 \mid Y = 0\right) = 0.5 \end{gathered} \tag{4}$$

(c):
For $Y = 1$,

$$\begin{aligned} & P\left(Y = 1, X_{\text{Project}} = 1, X_{\text{Vacation}} = 1, X_{\text{Deadline}} = 1, X_{\text{Relaxing}} = 1\right) \\ =& P(Y=1) \cdot P\left(X_{\text{Project}} = 1 \mid Y = 1\right) \cdot P\left(X_{\text{ProVacationject}} = 1 \mid Y = 1\right) \\ & \cdot P\left(X_{\text{Deadline}} = 1 \mid Y = 1\right) \cdot P\left(X_{\text{Relaxing}} = 1 \mid Y = 1\right) = 0 \end{aligned}$$

For $Y = 0$

$$\begin{aligned} &P\left(Y=0, X_{\text{Project}}=1, X_{\text{Vacation}}=1, X_{\text{Deadline}}=1, X_{\text{Relaxing}}=1\right) \\ =\ &P(Y=0) \cdot P\left(X_{\text{Project}}=1 \mid Y=0\right) \cdot P\left(X_{\text{ProVacationject}}=1 \mid Y=0\right) \\ &\cdot P\left(X_{\text{Deadline}}=1 \mid Y=0\right) \cdot P\left(X_{\text{Relaxing}}=1 \mid Y=0\right)=\frac{1}{16} \end{aligned}$$

Thus

$$Y=\underset{y}{\operatorname{argmax}} P\left(Y=y, X_{\text{Project}}=1, X_{\text{Vacation}}=1, X_{\text{Deadline}}=1, X_{\text{Relaxing}}=1\right)=0$$

## Problem 5

Suppose $f : R^n \to R$ differentiable, the lower bound of $f$ is $f_*$, we want to use Gradient Descent to minimize $f$.
(1) If $f$ is L-smooth with $L > 0$, will the algorithm converge if we start from any point in the domain? If not, what assumptions should $f$ satisfies?
(2) Now suppose $f$ is L-smooth, $w_0$ is our start point, we fix the stepsize $\alpha$, give a proof of the convergence and also give the number of iterations in big-O for finding an $\epsilon$-critical point (i.e., $a$ point where $\|\nabla f(w)\|_2 \leq \epsilon$).
(Hint: Stepsize $\alpha$ is a constant, so the number of iterations in big-O does not depend on the accurate value of stepsize, in other words, you can determine $\alpha$ to make sure the objective always descent.)

**Solution:**
(1):
Yes, it will converge, no other assumptions needed.
(2):
By L-smooth of $f$:

$$\left|f\left(w_{k+1}\right)-\left(f\left(w_k\right)+\alpha\left\|\nabla f\left(w_k\right)\right\|_2^2\right)\right| \leq \frac{\alpha^2 L}{2}\left\|\nabla f\left(w_k\right)\right\|_2^2$$

Let $\alpha=\frac{1}{L}$, then $f\left(w_{k+1}\right) \leq f\left(w_k\right)-\frac{1}{2L}\left\|\nabla f\left(w_k\right)\right\|_2^2$.
So, in $k$ gradient descent steps, we have an iterate $w_k$ such that

$$f_*-f\left(w_0\right) \leq f\left(w_k\right)-f\left(w_0\right) \leq-\frac{1}{2L}\sum_{i=0}^{k-1}\left\|\nabla f\left(w_i\right)\right\|_2^2$$

Rearranging, we get

$$\frac{1}{k}\sum_{i=0}^{k-1}\left\|\nabla f\left(w_i\right)\right\|_2^2 \leq \frac{2L\left(f\left(w_0\right)-f_*\right)}{k}$$

So for one of the $w_i's$, $\left\|\nabla f\left(w_i\right)\right\|_2^2 \leq \frac{2L(f(w_0)-f_*)}{k}$. So, if we want to find an $\epsilon$-critical point i.e., $a$ point where $\|\nabla f(w)\|_2 \leq \epsilon$, we need $O(\frac{2L(f(w_0)-f_*)}{\epsilon^2})$ iterations.($O(\frac{1}{\epsilon^2})$ or other equivalent form is also correct)

## Problem 6

Suppose $f:\ R^n \to R$ twice continuously differentiable with lower bound, domain of $f$ is $R^n$, we want to minimize $f$ starting at $x_0$ for as less iterations as possible.
(1)Give the iteration formula of $x$ in Newton's Method. Is it a descent direction?
(2)Give an example and the corresponding start point that the Newton's Method doesn't converge, here we let $n = 1$.

**Solution:**
(1):

$$\boldsymbol{x}_{k+1}=\boldsymbol{x}_k-\left[\nabla^2 f\left(\boldsymbol{x}_k\right)\right]^{-1} \nabla f\left(\boldsymbol{x}_k\right)$$

It is not necessarily a descent direction.
(2):
Consider $f(x) = \sqrt{1+x^2}$ defined over $R$. The minimizer of $f$ over $\mathbb{R}$ is $x_* = 0$. The first and second derivatives of $f$ are

$$f'(x) = \frac{x}{\sqrt{1+x^2}}, \quad f''(x) = \frac{1}{(1+x^2)^{\frac{3}{2}}}$$

The update of Newton's method hast the form

$$x_{k+1} = x_k - \frac{f'(x_k)}{f''(x_k)} = x_k - x_k\left(1+x_k^2\right) = -x_k^3$$

We therefore see that for $|x_0| \geq 1$ the method diverges and that for $|x_0| < 1$ the method converges very rapidly to the solution $x_* = 0$.