Machine Learning Foundation Homework 3

B04705003 Tzu-Chuan, Lin

Problem 1. See Figure 1.

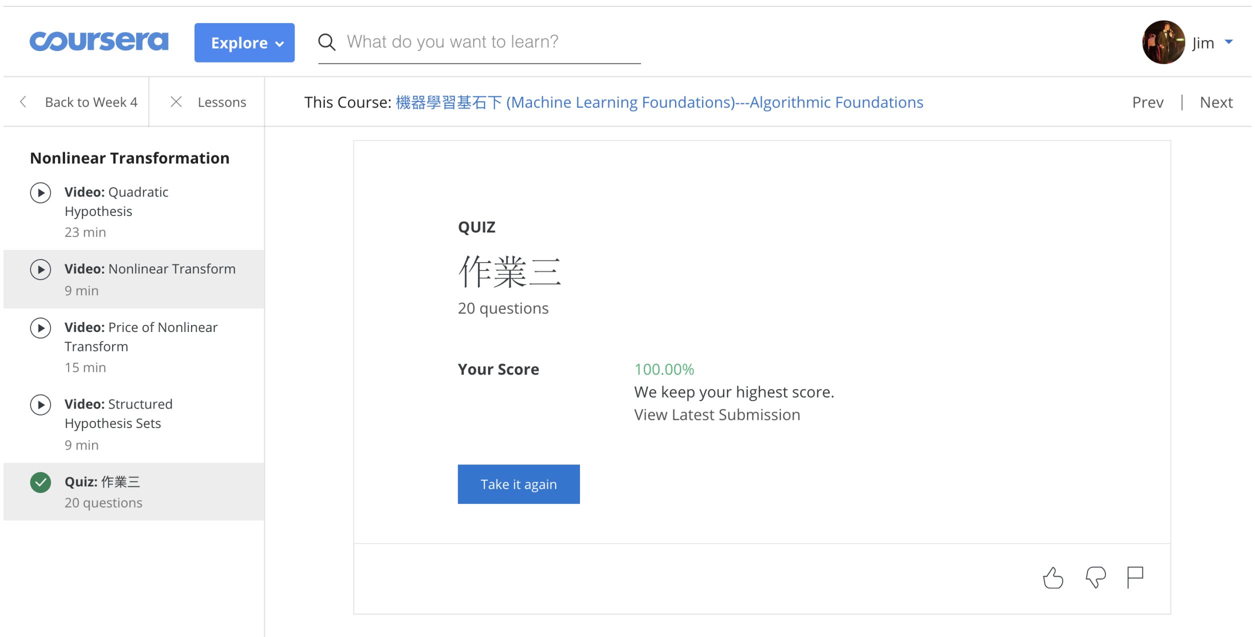


Figure 1: Problem 1

Problem 2. *Proof.* $err(\mathbf{w}) = \max(0, -y\mathbf{w}^T\mathbf{x})$ results in PLA.

$$
\begin{aligned}
\nabla_{\mathbf{w}} err(\mathbf{w}) &= \begin{cases} \frac{\partial -y\mathbf{w}^T\mathbf{x}}{\partial \mathbf{w}}, & \text{if } -y\mathbf{w}^T\mathbf{x} > 0 \\ 0, & \text{if } -y\mathbf{w}^T\mathbf{x} < 0 \end{cases} \\
&= \begin{cases} -y\mathbf{x}, & \text{if } y\mathbf{w}^T\mathbf{x} < 0 \\ 0, & \text{if } y\mathbf{w}^T\mathbf{x} > 0 \end{cases} \\
&= -[\![y \neq sign(\mathbf{w}^T\mathbf{x})]\!] y\mathbf{x}
\end{aligned}
$$

We can easily see that the update on PLA algorithm(on Lecture 11 slide 11):

$$\mathbf{w}_{t+1} \leftarrow \mathbf{w}_t + 1 \cdot [\![y \neq sign(\mathbf{w}^T\mathbf{x})]\!] y\mathbf{x}$$

is same as:

$$\mathbf{w}_{t+1} \leftarrow \mathbf{w}_t + \eta(-\nabla_{\mathbf{w}} err(\mathbf{w}))$$

when $\eta = 1$. $\square$

Problem 3.

$$E_{in} = \frac{1}{N}\sum_{n=1}^{N}(\ln(\sum_{k=1}^{K}\exp(\mathbf{w}_k^T\mathbf{x}_n)) - \mathbf{w}_{y_n}^T\mathbf{x}_n)$$

Compute the gradient:

$$\begin{aligned}
\frac{\partial E_{in}}{\partial \mathbf{w}_i} &= \frac{1}{N}\sum_{n=1}^{N}\frac{\partial}{\partial \mathbf{w}_i}(\ln(\sum_{k=1}^{K}\exp(\mathbf{w}_k^T\mathbf{x}_n)) - \mathbf{w}_{y_n}^T\mathbf{x}_n)) \\
&= \frac{1}{N}\sum_{n=1}^{N}\frac{\partial}{\partial \mathbf{w}_i}(\ln(\sum_{k=1}^{K}\exp(\mathbf{w}_k^T\mathbf{x}_n))) - \frac{\partial}{\partial \mathbf{w}_i}(\mathbf{w}_{y_n}^T\mathbf{x}_n) \\
&= \frac{1}{N}\sum_{n=1}^{N}\frac{\exp(\mathbf{w}_i^T\mathbf{x}_n)}{\sum_{k=1}^{K}\exp(\mathbf{w}_k\mathbf{x}_n)}\cdot\mathbf{x}_n - [\![y_n = i]\!]\mathbf{x}_n \\
&= \frac{1}{N}\sum_{n=1}^{N}h_i(\mathbf{x}_n)\mathbf{x}_n - [\![y_n = i]\!]\mathbf{x}_n \\
&= \frac{1}{N}\sum_{n=1}^{N}(h_i(\mathbf{x}_n) - [\![y_n = i]\!])\mathbf{x}_n
\end{aligned}$$

Problem 4. See Figure 2 and 3. Note that because the spec doesn't specify which err function should I use for E_{in}, I choose to use $err_{1/0}$ as E_{in}.

We can easily find out that SGD(Stochastic Gradient Descent($\eta = 0.001$)) E_{in} decrease very slow compared with GD($\eta = 0.01$).

Problem 5. See Figure 2 and 3.

I find out that E_{out} behaves almost as E_{in}. It implies that training data and testing data are from similar distribution.

Problem 6(Bonus). *Proof.* First, we claim that:

$$\underset{w_1,w_2,...,w_K}{\arg\min}\ RMSE(H) = \underset{w_1,w_2,...,w_K}{\arg\min}\ RMSE(H)^2$$

because square function is an increasing function on $[0, +\infty)$. Then, we define a special feature transformation which act like:

$$\phi(\mathbf{x}) = [h_1(\mathbf{x}), h_2(\mathbf{x}), ..., h_K(\mathbf{x})]^T$$

Also, we abuse the notation ϕ so that it can be applied to the whole dataset $X \in \mathcal{R}^{N\times(d+1)}$:

$$\phi(X) = \begin{bmatrix} \phi(\mathbf{x}_1)^T \\ \phi(\mathbf{x}_2)^T \\ ... \\ \phi(\mathbf{x}_K)^T \end{bmatrix}$$

Moreover, define:

$$\mathbf{w} = [w_1, w_2, ..., w_K]^T$$

After this special feature transformation, our minimization problem becomes:

$$\begin{aligned}
\underset{w_1,w_2,...,w_K}{\arg\min}\ RMSE(H)^2 &= \underset{w_1,w_2,...,w_K}{\arg\min}\ \frac{1}{N}\sum_{n=1}^{N}(y_n - H(\mathbf{x}_n))^2 \\
&= \underset{w_1,w_2,...,w_K}{\arg\min}\ \frac{1}{N}\sum_{n=1}^{N}(y_n - \mathbf{w}^T\phi(\mathbf{x}_n))^2 \\
&= \underset{w_1,w_2,...,w_K}{\arg\min}\ \frac{1}{N}\left\|\phi(X)\mathbf{w} - y\right\|^2
\end{aligned}$$

Because it is a convex problem, we can solve it by taking $\nabla_{\mathbf{w}} RMSE(H)^2 = 0$.

Compute the gradient:

$$\nabla_{\mathbf{w}} RMSE(H)^2 = \frac{2}{N}(\phi(X)^T\phi(X)\mathbf{w} - \phi(X)^T y)$$

The remain problem is whether we can compute $\phi(X)^T y$ and the answer is yes, by putting all remaining information together.

Consider:

$$RMSE(h_0)^2 = e_0^2 = \frac{1}{N}\sum_{n=1}^{N} y_n^2$$

$$\Rightarrow \sum_{n=1}^{N} y_n^2 = Ne_0^2$$

$$RMSE(h_k)^2 = e_k^2 = \frac{1}{N}\sum_{n=1}^{N}(y_n - h_k(\mathbf{x}_n))^2$$

$$\Rightarrow \sum_{n=1}^{N} y_n^2 - 2\sum_{n=1}^{N} y_n h_k(\mathbf{x}_n) + \sum_{n=1}^{N} h_k(\mathbf{x}_n)^2 = Ne_k^2$$

$$\Rightarrow \sum_{n=1}^{N} y_n h_k(\mathbf{x}_n) = \frac{1}{2}\sum_{n=1}^{N} h_k(\mathbf{x}_n)^2 + \frac{N}{2}(e_0^2 - e_k^2)$$

$$\Rightarrow \phi(X)^T y = \frac{1}{2}\begin{bmatrix} \left\|\phi(X)_{:,1}\right\|^2 \\ \left\|\phi(X)_{:,2}\right\|^2 \\ ... \\ \left\|\phi(X)_{:,K}\right\|^2 \end{bmatrix} + \frac{N}{2}(\mathbf{e}_0^2 - \mathbf{e}^2)$$

where:

$$\mathbf{e}_0 = \begin{bmatrix} e_0 \\ e_0 \\ ... \\ e_0 \end{bmatrix} \in \mathcal{R}^{k\times 1}$$

$$\mathbf{e} = \begin{bmatrix} e_1 \\ e_2 \\ ... \\ e_K \end{bmatrix} \in \mathcal{R}^{k\times 1}$$

and if we take square on a column vector simply means element-wise square function.

Therefore, the gradient equal to zero point is :

$$\begin{aligned}
\mathbf{w}^* &= (\phi(X)^T\phi(X))^\dagger(\phi(X)^T y) \\
&= (\phi(X)^T\phi(X))^\dagger(\frac{1}{2}\begin{bmatrix} \|\phi(X)_{:,1}\|^2 \\ \|\phi(X)_{:,2}\|^2 \\ ... \\ \|\phi(X)_{:,K}\|^2 \end{bmatrix} + \frac{N}{2}(\mathbf{e}_0^2 - \mathbf{e}^2))
\end{aligned}$$

which is a minimizer of:

$$\min_{w_1,w_2,...,w_K} RMSE(H)$$

$\square$

Therefore, optimal $RMSE(H)$ becomes:

$$RMSE(H) = \sqrt{\frac{1}{N}\|\phi(X)\mathbf{w}^* - y\|^2}$$

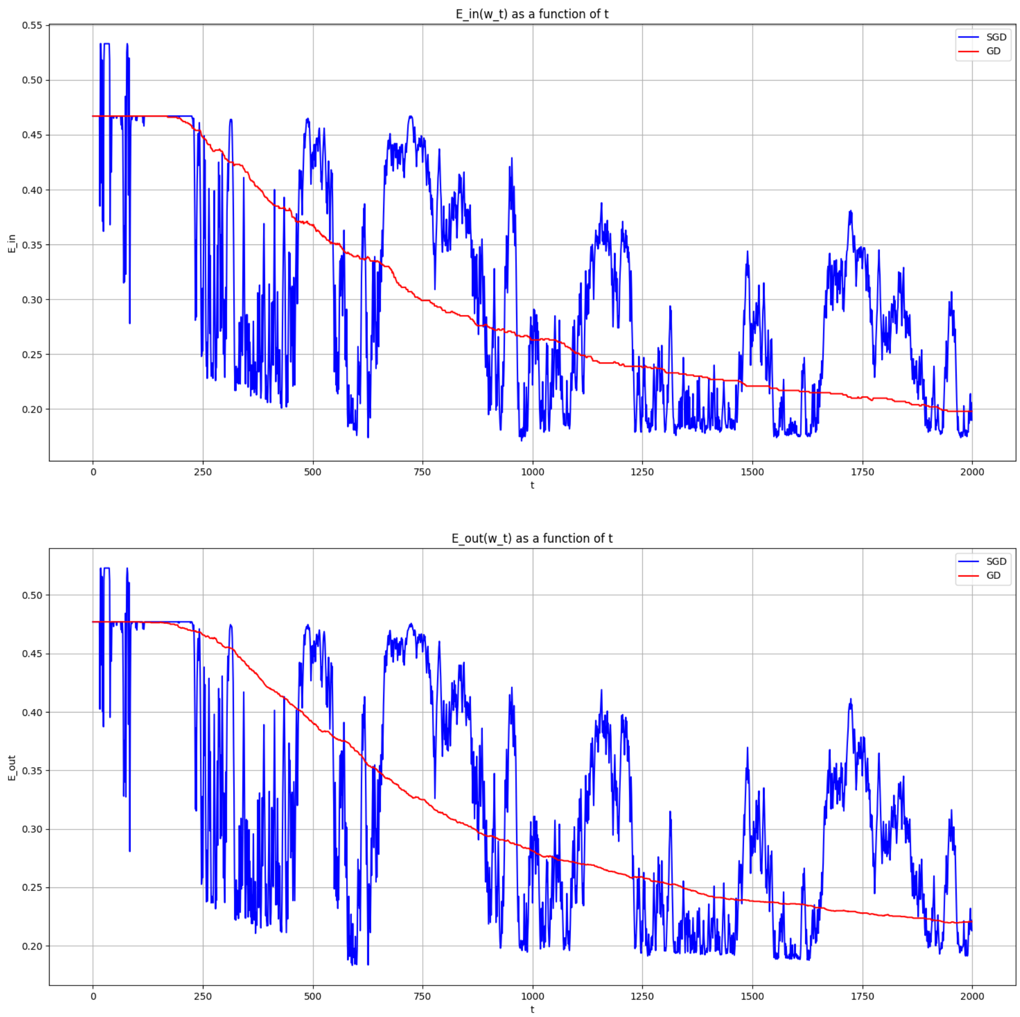
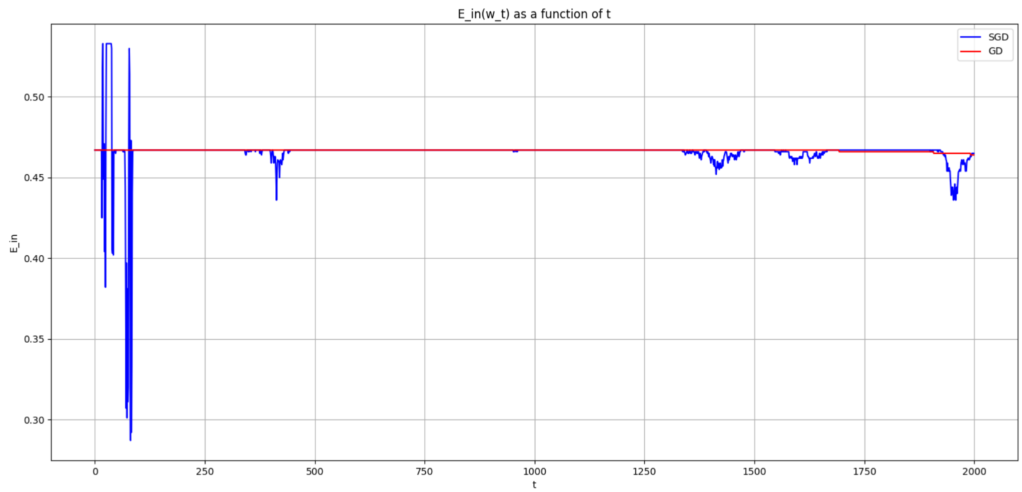


Figure 2: $\eta = 0.01$



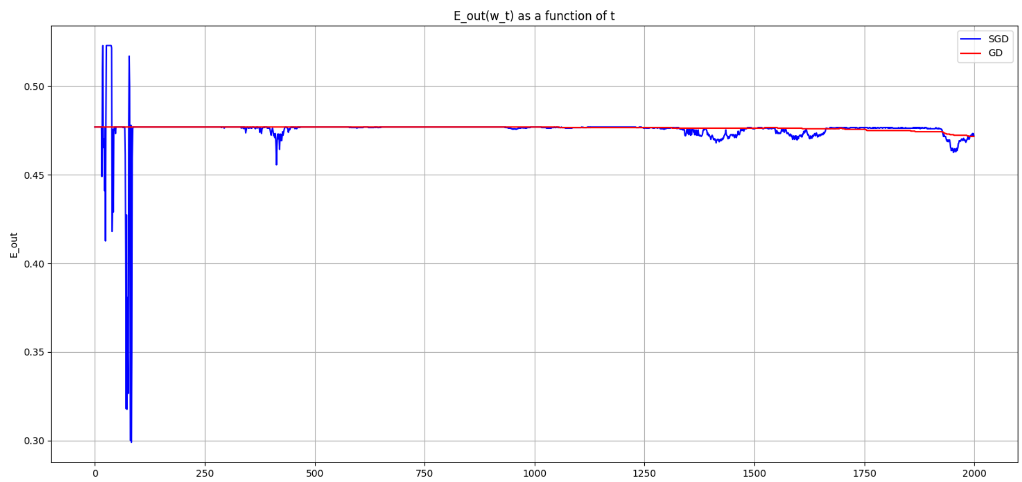


Figure 3: $\eta = 0.001$