Homework #3

You will use Gradescope to upload your choices and your scanned/printed solutions. For problems marked with (*), please follow the guidelines on the course website and upload your source code to Gradescope as well. Any programming language/platform is allowed.

Any form of cheating, lying, or plagiarism will not be tolerated. Students can get zero scores and/or fail the class and/or be kicked out of school and/or receive other punishments for those kinds of misconducts.

Discussions on course materials and homework solutions are encouraged. But you should write the final solutions alone and understand them fully. Books, notes, and Internet resources can be consulted, but not copied from.

Since everyone needs to write the final solutions alone, there is absolutely no need to lend your homework solutions and/or source codes to your classmates at any time. In order to maximize the level of fairness in this class, lending and borrowing homework solutions are both regarded as dishonest behaviors and will be punished according to the honesty policy.

You should write your solutions in English or Chinese with the common math notations introduced in class or in the problems. We do not accept solutions written in any other languages.

This homework set comes with 16 problems and a total of 400 points. For each problem, there is one correct choice. If you choose the correct answer, you get 20 points; if you choose an incorrect answer, you get 0 points. For four of the secretly-selected problems, the TAs will grade your detailed solution in terms of the written explanations and/or code based on how logical/clear your solution is. Each of the four problems graded by the TAs counts as additional 20 points (in addition to the correct/incorrect choices you made). In general, each homework (except homework 0) is of a total of 400 points.

Linear Regression

1. Consider a noisy target $y = \mathbf{w}_f^T \mathbf{x} + \epsilon$, where $\mathbf{x} \in \mathbb{R}^{d+1}$ (including the added coordinate $x_0 = 1$), $y \in \mathbb{R}$, $\mathbf{w}_f \in \mathbb{R}^{d+1}$ is an unknown vector, and ϵ is an i.i.d. noise term with zero mean and σ^2 variance. Assume that we run linear regression on a training data set $\mathcal{D} = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)\}$ generated i.i.d. from some $P(\mathbf{x})$ and the noise process above, and obtain the weight vector $\mathbf{w}_{\text{lin}}$. As briefly discussed in Lecture 5, it can be shown that the expected in-sample error $E_{\text{in}}(\mathbf{w}_{\text{lin}})$ with respect to $\mathcal{D}$ is given by:

$$\mathbb{E}_{\mathcal{D}}\left[E_{\text{in}}(\mathbf{w}_{\text{lin}})\right] = \sigma^2 \left(1 - \frac{d+1}{N}\right).$$

For $\sigma = 0.1$ and $d = 19$, what is the smallest number of examples N such that $\mathbb{E}_{\mathcal{D}}\left[E_{\text{in}}(\mathbf{w}_{\text{lin}})\right]$ is no less than 0.005? Choose the correct answer; explain your answer.

[a] 25
[b] 30
[c] 35
[d] 40
[e] 45

C

2. Consider the target function $f(x) = x^2$. Sample x uniformly from $[0,1]$, and use all linear hypotheses $h(x) = w_0 + w_1 \cdot x$ to approximate the target function with respect to the squared error. What are the weights (w_0^*, w_1^*) of the optimal hypothesis? Choose the correct answer; explain your answer.

[a] $(0,1)$

[b] $(\frac{1}{2}, \frac{1}{2})$

[c] $(-\frac{1}{6}, 1)$

[d] $(-\frac{1}{4}, \frac{1}{4})$

[e] $(\frac{1}{3}, 0)$

(*Hint: The optimal hypothesis g^* must reach the minimum $E_{\text{out}}(g^*)$.*)

e

3. Following the previous problem, assume that we sample two examples x_1 and x_2 uniformly from $[0,1]$ to form the training set $\mathcal{D} = \{(x_1, f(x_1)), (x_2, f(x_2))\}$, and use linear regression to get g for approximating the target function with respect to the squared error. <u>You can neglect the degenerate cases where x_1 and x_2 are the same</u>. What is $\mathbb{E}_{\mathcal{D}}(|E_{\text{in}}(g) - E_{\text{out}}(g)|)$? Choose the correct answer; explain your answer.

↓

Need to consider this case during doing integral ?

[a] $\frac{1}{60}$

[b] $\frac{4}{15}$

[c] $\frac{3}{20}$

[d] $\frac{1}{25}$

[e] $\frac{1}{30}$

Cross-Entropy Error   *Deeplearning book - MIT Press*

b

4. In class, we introduced our version of the cross-entropy error function

$$E_{\text{in}}(\mathbf{w}) = \frac{1}{N} \sum_{n=1}^{N} -\ln \theta(y_n \mathbf{w}^T \mathbf{x}_n).$$

$(x_1, 1)$ $(+3, 1)$
$(x_2, 0)$

based on the definition of $y_n \in \{-1, +1\}$. If we transform y_n to $y_n' \in \{0,1\}$ by $y_n' = \frac{y_n+1}{2}$, which of the following error function is equivalent to E_{in} above? Choose the correct answer; explain your answer.

[a] $\frac{1}{N} \sum_{n=1}^{N} \left(y_n' \ln \theta(\mathbf{w}^T \mathbf{x}_n) + (1 - y_n') \ln(\theta(-\mathbf{w}^T \mathbf{x}_n)) \right)$ $-\ln \theta(y_n \mathbf{w}^T x_n)$ ——①

[b] $\frac{1}{N} \sum_{n=1}^{N} \left(y_n' \ln \theta(-\mathbf{w}^T \mathbf{x}_n) + (1 - y_n') \ln(\theta(\mathbf{w}^T \mathbf{x}_n)) \right)$ $\Leftarrow \ln \theta(-y_n \mathbf{w}^T x_n)$ ——②

[c] $\frac{1}{N} \sum_{n=1}^{N} \left(y_n' \ln \theta(\mathbf{w}^T \mathbf{x}_n) - (1 - y_n') \ln(\theta(-\mathbf{w}^T \mathbf{x}_n)) \right)$ $\sum -\ln \theta(y_n \mathbf{w}^T x_n)$ ——③

[d] $\frac{1}{N} \sum_{n=1}^{N} \left(y_n' \ln \theta(-\mathbf{w}^T \mathbf{x}_n) - (1 - y_n') \ln(\theta(\mathbf{w}^T \mathbf{x}_n)) \right)$ $\Leftarrow \sum \ln \theta(-y_n \mathbf{w}^T x_n)$ ——④

[e] none of the other choices

??

$-\ln \theta(\mathbf{w}^T x_n)$

$= \ln (1 + e^{-\mathbf{w}^T x_n})$

$= \ln \frac{1}{e^{\mathbf{w}^T x_n}} (1 + e^{\mathbf{w}^T x_n})$

$=$

$\ln \theta(-\mathbf{w}^T x_n)$

$= \ln (1 - \theta(\mathbf{w}^T x_n))$

$= \ln \left(1 - \frac{1}{1 + e^{-\mathbf{w}^T x_n}} \right)$

$= \ln \left(\frac{e^{-\mathbf{w}^T x_n}}{1 + e^{-\mathbf{w}^T x_n}} \right)$

5. Consider a coin with an unknown head probability μ. Independently flip this coin N times to get $y_1, y_2, \ldots, y_N$, where $y_n = 1$ if the n-th flipping results in head, and 0 otherwise. Define $\nu = \frac{1}{N}\sum_{n=1}^{N} y_n$. How many of the following statements about ν are true? Choose the correct answer; explain your answer by briefly illustrating why those statements are true.

- With probability more than $1 - \delta$,

$$\mu \leq \nu + \sqrt{\frac{1}{2N}\ln\frac{2}{\delta}}$$

$\sqrt{\frac{1}{2N}\ell_n \frac{2M}{\delta}}$

for all $N \in \mathbb{N}$ and $0 < \delta < 1$.

$M = 1$?

- ν maximizes likelihood$(\hat{\mu})$ over all $\hat{\mu} \in [0, 1]$. [ ]
- ν minimizes the squared error

$$E^{\text{sqr}}(\hat{y}) = \frac{1}{N}\sum_{n=1}^{N}(\hat{y} - y_n)^2$$

over all $\hat{y} \in \mathbb{R}$. $(0, 1)$
- When $0 < \nu < 1$, it minimizes the cross-entropy error (which is similar to the cross-entropy error for logistic regression)

$$E^{\text{ce}}(\hat{y}) = -\frac{1}{N}\sum_{n=1}^{N}\Big(y_n \ln\hat{y} + (1 - y_n)\ln(1 - \hat{y})\Big)$$

$\sum_{n=1}^{N} y_n = N \times \frac{y_n \cdots y_n}{N}$

$= N\nu$

over all $\hat{y} \in (0, 1)$.

(*Note: μ is similar to the role of the "target function" and $\hat{\mu}$ is similar to the role of the "hypothesis" in our machine learning framework.*)

[a] 0

[b] 1

[c] 2

[d] 3

[e] 4

Stochastic Gradient Descent

6. In the perceptron learning algorithm, we find one example $(\mathbf{x}_{n(t)}, y_{n(t)})$ that the current weight vector $\mathbf{w}_t$ mis-classifies, and then update $\mathbf{w}_t$ by

PLA $\rightarrow$ $\qquad \mathbf{w}_{t+1} \leftarrow \mathbf{w}_t + y_{n(t)}\mathbf{x}_{n(t)}.$

The algorithm can be viewed as optimizing some $E_{\text{in}}(\mathbf{w})$ that is composed of one of the following point-wise error functions with stochastic gradient descent (neglecting any non-differentiable points of the error function). What is the error function? Choose the correct answer; explain your answer.

[a] $\text{err}(\mathbf{w}, \mathbf{x}, y) = \max(0, -y\mathbf{w}^T\mathbf{x})$

[b] $\text{err}(\mathbf{w}, \mathbf{x}, y) = -\max(0, -y\mathbf{w}^T\mathbf{x})$

[c] $\text{err}(\mathbf{w}, \mathbf{x}, y) = \max(y\mathbf{w}^T\mathbf{x}, -y\mathbf{w}^T\mathbf{x})$

[d] $\text{err}(\mathbf{w}, \mathbf{x}, y) = -\max(y\mathbf{w}^T\mathbf{x}, -y\mathbf{w}^T\mathbf{x})$

[e] none of the other choices

對正確的 error function

取 gradient，會得到

$+ y_{n(t)} X_{n(t)}$

Multinomial Logistic Regression

7. In Lecture 6, we solve multiclass classification by OVA or OVO decompositions. One alternative to deal with multiclass classification is to extend the original logistic regression model to Multinomial Logistic Regression (MLR). For a K-class classification problem, we will denote the output space $\mathcal{Y} = \{1, 2, \cdots, K\}$. The hypotheses considered by MLR can be indexed by a matrix

$$W = \begin{bmatrix} | & | & \cdots & | & \cdots & | \\ \mathbf{w}_1 & \mathbf{w}_2 & \cdots & \mathbf{w}_k & \cdots & \mathbf{w}_K \\ | & | & \cdots & | & \cdots & | \end{bmatrix}_{(d+1) \times K},$$

that contains weight vectors $(\mathbf{w}_1, \cdots, \mathbf{w}_K)$, each of length $d+1$. The matrix represents a hypothesis

$$h_y(\mathbf{x}) = \frac{\exp(\mathbf{w}_y^T \mathbf{x})}{\sum_{i=1}^{K} \exp(\mathbf{w}_i^T \mathbf{x})}$$

that can be used to approximate the target distribution $P(y|\mathbf{x})$ for any $(\mathbf{x}, y)$. MLR then seeks for the maximum likelihood solution over all such hypotheses. For a given data set $\{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)\}$ generated i.i.d. from some $P(\mathbf{x})$ and target distribution $P(y|\mathbf{x})$, the likelihood of $h_y(\mathbf{x})$ is proportional to $\prod_{n=1}^{N} h_{y_n}(\mathbf{x}_n)$. That is, minimizing the negative log likelihood is equivalent to minimizing an $E_{\text{in}}(W)$ that is composed of the following error function

$$\text{err}(W, \mathbf{x}, y) = -\ln h_y(\mathbf{x}) = -\sum_{k=1}^{K} [\![y = k]\!] \ln h_k(\mathbf{x}).$$

When minimizing $E_{\text{in}}(W)$ with SGD, we update the $W^{(t)}$ at the t-th iteration to $W^{(t+1)}$ by

$$W^{(t+1)} \leftarrow W^{(t)} + \eta \cdot V,$$

where V is a $(d+1) \times K$ matrix whose k-th column is an update direction for the k-th weight vector. Assume that an example $(\mathbf{x}_n, y_n)$ is used for the SGD update above. What is the y_n-th column of V? Choose the correct answer; explain your answer.

[a] $(1 - h_{y_n}(\mathbf{x}_n))\mathbf{x}_n$

[b] $(h_{y_n}(\mathbf{x}_n) - 1)\mathbf{x}_n$

[c] $(-h_{y_n}(\mathbf{x}_n))\mathbf{x}_n$

[d] $(h_{y_n}(\mathbf{x}_n))\mathbf{x}_n$

[e] none of the other choices

Nonlinear Transformation

8. Given the following training data set:

$$\mathbf{x}_1 = (0, 1), y_1 = -1 \quad \mathbf{x}_2 = (0, -1), y_2 = -1 \quad \mathbf{x}_3 = (-1, 0), y_3 = +1 \quad \mathbf{x}_4 = (1, 0), y_4 = +1$$

Use the quadratic transform $\mathbf{\Phi}_2(\mathbf{x}) = (1, x_1, x_2, x_1^2, x_1 x_2, x_2^2)$ and take $\text{sign}(0) = 1$. Which of the following weight vector $\tilde{\mathbf{w}}$ represents a linear classifier in the $\mathcal{Z}$-space that can separate all the transformed examples perfectly? Choose the correct answer; explain your answer.
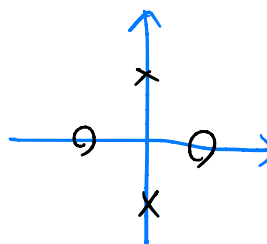
[a] $(0, -1, 0, 0, 0, 0)$

[b] $(0, 0, -1, 0, 0, 0)$

[c] $(0, 0, 0, -1, 0, 0)$

[d] $(0, 0, 0, 0, -1, 0)$

[e] $(0, 0, 0, 0, 0, -1)$

9. Consider a feature transform $\mathbf{\Phi}(\mathbf{x}) = \Gamma\mathbf{x}$ where Γ is a $(d+1)$ by $(d+1)$ invertible matrix. For a training data set $\{(\mathbf{x}_n, y_n)\}_{n=1}^N$, run linear regression on the original data set, and get $\mathbf{w}_{\text{lin}}$. Then, run linear regression on the $\mathbf{\Phi}$-transformed data, and get $\tilde{\mathbf{w}}$. For simplicity, assume that the matrix X (with every row being $\mathbf{x}_n^T$) satisfies that $X^T X$ is invertible. What is the relationship between $\mathbf{w}_{\text{lin}}$ and $\tilde{\mathbf{w}}$? Choose the correct answer; explain your answer.

[a] $\mathbf{w}_{\text{lin}} = \Gamma\tilde{\mathbf{w}}$

[b] $\mathbf{w}_{\text{lin}} = \Gamma^T\tilde{\mathbf{w}}$

[c] $\mathbf{w}_{\text{lin}} = (\Gamma^{-1})^T\tilde{\mathbf{w}}$

[d] $\mathbf{w}_{\text{lin}} = \Gamma^{-1}\tilde{\mathbf{w}}$

[e] none of the other choices

10. After "visualizing" the data and noticing that all $\mathbf{x}_1,, \mathbf{x}_2, \ldots, \mathbf{x}_n$ are distinct, Dr. Trans magically decides the following transform

$$\mathbf{\Phi}(\mathbf{x}) = (⟦\mathbf{x} = \mathbf{x}_1⟧, ⟦\mathbf{x} = \mathbf{x}_2⟧, \ldots, ⟦\mathbf{x} = \mathbf{x}_N⟧).$$

That is, $\mathbf{\Phi}(\mathbf{x})$ is a N-dimensional vector whose n-th component is 1 if and only if $\mathbf{x} = \mathbf{x}_n$. If we run linear regression after applying this transform, what is the optimal $\tilde{\mathbf{w}}$? Choose the correct answer; explain your answer.

optimal meaning ?

[a] $\mathbf{1}$, the vector of all 1s.

[b] $\mathbf{0}$, the vector of all 0s.

$\Phi(x) = (0, 1, 0, 0, \ldots, 0)$

[c] $\mathbf{y}$

[d] $-\mathbf{y}$

$I \rightarrow \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$

$y \begin{bmatrix} 0 & 1 \\ 0 & \vdots \\ 0 & \vdots \end{bmatrix}$

[e] none of the other choices

(Note: Be sure to also check what $E_{\text{in}}(\tilde{\mathbf{w}})$ is!)

11. Assume that we couple linear regression with one-versus-all decomposition for multi-class classification, and get K weight vectors $\mathbf{w}_{[k]}^*$. Assume that the squared error $E_{\text{in}}^{\text{sqr}}(\mathbf{w}_{[k]}^*)$ for the k-th binary classification problem is e_k. What is the tightest upper bound of $E_{\text{in}}^{0/1}(g)$, where g is the multi-class classifier formed by the one-versus-all decomposition? Choose the correct answer; explain your answer.

[a] $2\sum_{k=1}^{K} e_k$

[b] $\sum_{k=1}^{K} e_k$

[c] $\frac{1}{2}\sum_{k=1}^{K} e_k$

$E_{\text{in}}^{\text{sqr}}(\mathbf{w}_{[k]}^*) = e^k$

[d] $\frac{1}{K}\sum_{k=1}^{K} e_k$

[e] $\frac{1}{2K}\sum_{k=1}^{K} e_k$

$E_{\text{in}}^{\%}(g)$

Experiments with Linear and Nonlinear Models

Next, we will play with transform + linear regression for binary classification. Please use the following set for training:

$$\texttt{https://www.csie.ntu.edu.tw/~htlin/course/ml21fall/hw3/hw3\_train.dat}$$

and the following set for testing (estimating E_{out}):

$$\texttt{https://www.csie.ntu.edu.tw/~htlin/course/ml21fall/hw3/hw3\_test.dat}$$

Each line of the data set contains one $(\mathbf{x}_n, y_n)$ with $\mathbf{x}_n \in \mathbb{R}^{10}$. The first 10 numbers of the line contains the components of $\mathbf{x}_n$ orderly, the last number is y_n which belongs to $\{-1, +1\} \subseteq \mathbb{R}$. That is, we can use those y_n for either binary classification or regression.

12. (*) Consider the following *homogeneous* order-Q polynomial transform

$$\mathbf{\Phi}(\mathbf{x}) = (1, x_1, x_2, ..., x_{10}, x_1^2, x_2^2, ..., x_{10}^2, ..., x_1^Q, x_2^Q, ..., x_{10}^Q).$$

Transform the training and testing data according to $\mathbf{\Phi}(\mathbf{x})$ with $Q = 2$, and implement the linear regression algorithm on the transformed data. What is $\left| E_{\text{in}}^{0/1}(g) - E_{\text{out}}^{0/1}(g) \right|$, where g is the hypothesis returned by the transform + linear regression procedure? Choose the closest answer; provide your code.

[a] 0.28

[b] 0.32

[c] 0.36

[d] 0.40

[e] 0.44

0.3263

13. (*) Repeat the previous problem, but with $Q = 8$ instead. What is $\left| E_{\text{in}}^{0/1}(g) - E_{\text{out}}^{0/1}(g) \right|$, where g is the hypothesis returned by the transform + linear regression procedure? Choose the closest answer; provide your code.

[a] 0.30

[b] 0.35

[c] 0.40

[d] 0.45

[e] 0.50

0.4576

14. (*) Repeat the previous problem, but with $\mathbf{\Phi}_2$ (the full order-2 polynomial transform introduced in the lecture, which is of $1 + 10 + 45 + 10$ dimensions) instead. What is $\left| E_{\text{in}}^{0/1}(g) - E_{\text{out}}^{0/1}(g) \right|$, where g is the hypothesis returned by the transform + linear regression procedure? Choose the closest answer; provide your code.

[a] 0.33

[b] 0.41

[c] 0.49

[d] 0.57

[e] 0.65

0.3386

$d = 10, \quad Q = 2$

$$\binom{10 - 1 + 2}{2}$$

$$= C_2^{11} = \frac{11 \times 10}{2} = 55$$

$$\binom{10 + 2}{2} = C_3^{12} = \frac{12 \times 11}{2} = 66$$

15. (*) Instead of transforming to a higher dimensional space, we can also transform to a lower dimensional space. Consider the following 10 transforms:

$$\boldsymbol{\Phi}^{(1)}(\mathbf{x}) = (x_0, x_1)$$
$$\boldsymbol{\Phi}^{(2)}(\mathbf{x}) = (x_0, x_1, x_2)$$
$$\cdots$$
$$\boldsymbol{\Phi}^{(10)}(\mathbf{x}) = (x_0, x_1, x_2, \ldots, x_{10})$$

Run $\boldsymbol{\Phi}^{(i)}$ + linear regression to get a hypothesis g_i. Which i leads to a $\boldsymbol{\Phi}^{(i)}$ that reaches the minimum $\left| E_{\text{in}}^{0/1}(g_i) - E_{\text{out}}^{0/1}(g_i) \right|$? Choose the closest answer; provide your code.

should be precise answer ?

[a] 1

[b] 2

[c] 3

[d] 5

[e] 8

16. (*) Consider a transform that randomly chooses 5 out of 10 dimensions. That is, $\boldsymbol{\Phi}(\mathbf{x}) = (x_0, x_{i_1}, x_{i_2}, x_{i_3}, x_{i_4}, x_{i_5})$, where i_1 to i_5 are distinct random integers uniformly and independently generated within $\{1, 2, \ldots, 10\}$. Run $\boldsymbol{\Phi}$ + linear regression to get a hypothesis g. What is the average $\left| E_{\text{in}}^{0/1}(g_i) - E_{\text{out}}^{0/1}(g_i) \right|$ over 200 experiments, each generating $\boldsymbol{\Phi}$ with a different random seed? Choose the closest answer; provide your code.

[a] 0.06

[b] 0.11

[c] 0.16

[d] 0.21

[e] 0.26

☆ 如果是 distinct, 就不會是 independent ?

⇒ Oscar : without replacement

run 1e4 times ≈ 0.208