Homework #3

RED CORRECTION: 11/09/2021 10:15

RELEASE DATE: 11/08/2021

DUE DATE: 11/25/2021, BEFORE 13:00 on Gradescope

QUESTIONS ARE WELCOMED ON THE NTU COOL FORUM.

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

D

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:

$$\mathbb{E}_{\mathcal{D}}\left[E_{\text{in}}(\mathbf{w}_{\text{lin}})\right] = \underset{0.1}{\sigma^2}\left(1 - \frac{\overset{19}{d}+1}{N}\right). \text{ make } \geq 0.005$$

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

$$(0.01)\cdot\left(1 - \frac{19+1}{N}\right)$$

$$= 0.01 - \frac{20}{100N} \geq 0.005$$

$$= \frac{-1}{5N} \geq -0.005$$

$$= \frac{1}{N} \leq 0.025$$

$$= \frac{1}{N} \leq \frac{25}{1000} = \frac{1}{40}$$

$$N \geq 40$$

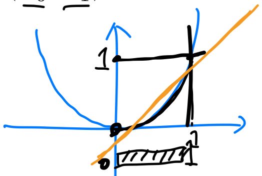
$y = -\frac{1}{6}x + 1$ $\quad -\left(\frac{\frac{1}{6}\times\frac{1}{6}}{2}\right) + \left(\frac{\frac{5}{6}\times\frac{5}{6}}{2}\right)$

$= -2\times\frac{1}{36} + 2\times\frac{25}{36} = -\frac{1}{72} + \frac{25}{72} = \frac{24}{72} = \frac{1}{3}$ #

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. $u_1 = +$

$h(x) = w_0 + w_1 x$

- [a] $(0, 1)$
- [b] $(\frac{1}{2}, \frac{1}{2})$
- [c] $(-\frac{1}{6}, 1)$ V
- [d] $(-\frac{1}{4}, \frac{1}{4})$
- [e] $(\frac{1}{3}, 0)$



But why

$X = \begin{bmatrix} - & N & - \\ - & N & - \\ - & N & - \end{bmatrix} N \quad [\ \ ]$

$\frac{1}{6}x =$ $\quad \frac{1}{180}$

$\frac{1}{N}\|h(x) - y\|^2$

$w_1 - 2x = 0$

$w_1 = 2x$

$\frac{1}{N}(2w_1 - 4x) = 0$

$\frac{1}{N} \times 2 \times [w_1 - 2x]$

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

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. You can neglect the degenerate cases where x_1 and x_2 are the same. What is $\mathbb{E}_{\mathcal{D}}(|E_{\text{in}}(g) - E_{\text{out}}(g)|)$? Choose the correct answer; explain your answer.

E

$dx \cdot dx_1 \cdot dx_2$ 的三重積分

- [a] $\frac{1}{60}$
- [b] $\frac{4}{15}$
- [c] $\frac{3}{20}$
- [d] $\frac{1}{25}$
- [e] $\frac{1}{30}$

$g(x_1, x_2)$ random variable.

pdf: $\frac{1}{b-a}$

$\underset{x\sim p}{\mathbb{E}}\left[\left((g(x_1,x_2))(x) - x^2\right)^2\right]$

$= \int_0^1 (g(x_1,x_2)(x) - x^2)^2 \cdot 1 \cdot dx$

$\underset{x_1,x_2\sim D}{\mathbb{E}}\left[E_{out}(g(x_1,x_2))\right]$

$\int_0^1\int_0^1 \square \, dx_1 dx_2$

期望值可以寫双重積分

Cross-Entropy Error

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

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

-1 or 1

B

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.

② ? Loss 值不重要

- [a] $\frac{1}{N}\sum_{n=1}^{N}\left(y_n' \ln\theta(\mathbf{w}^T\mathbf{x}_n) + (1-y_n')\ln(\theta(-\mathbf{w}^T\mathbf{x}_n))\right)$
- [b] $\frac{1}{N}\sum_{n=1}^{N}\left(y_n' \ln\theta(-\mathbf{w}^T\mathbf{x}_n) + (1-y_n')\ln(\theta(\mathbf{w}^T\mathbf{x}_n))\right)$
- [c] $\frac{1}{N}\sum_{n=1}^{N}\left(y_n' \ln\theta(\mathbf{w}^T\mathbf{x}_n) - (1-y_n')\ln(\theta(-\mathbf{w}^T\mathbf{x}_n))\right)$
- [d] $\frac{1}{N}\sum_{n=1}^{N}\left(y_n' \ln\theta(-\mathbf{w}^T\mathbf{x}_n) - (1-y_n')\ln(\theta(\mathbf{w}^T\mathbf{x}_n))\right)$
- [e] none of the other choices

$y_n = 0$

$y_n = 1 \quad y_n = 2$

求出來的 w^* 都一樣

← 差一个常數項 ⇒ 但做 minimize，所以沒有差

$\min -\ln(\theta(y_n w x))$

$\Leftrightarrow \min \ln(-\theta(y w x))$

$\Leftrightarrow \min \ln(\theta(-y w x))$

$y_n \{0, 1\}$ 代進來

$\frac{1}{1+2} = \frac{1}{2}$

← 23. p15

y_n的均值(在sample中head的機率)

E

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 $1-\delta$,

α: # head $= N\nu$

likelihood.

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

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

不管μ是多少, ν 都使 error最小,

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

what 可使它最小, 偏微

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

$\hat{y}^* \overset{?}{=} \nu$

over all $\hat{y} \in \mathbb{R}$.

- When $0 < \nu < 1$, it minimizes the cross-entropy error (which is similar to the cross-entropy error for logistic regression)

一樣

cross entropy

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

$\frac{\partial E^{ce}(\hat{y})}{\partial \hat{y}} = 0$ $\hat{y}^* \overset{?}{=} \nu$

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

A

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

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.

比較合理

[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

if y = good, $[\![y \neq \text{sign}(w^T x)]\!]$ or

$y w^T x < 0$: 代表 mistake

> 0 代表 OK!

8, 9,

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

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

第k的class的update方向

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

正確的class?

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

x属於 class y 的机率

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 $E_{\text{in}}(\mathrm{W})$ that is composed of the following error function

对偏微. ② 对W偏微分求極值

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

代入 x_n y_n correct label

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

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

this is positive. ③ 找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

[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

$\text{err}(W, x_n y_n) = -\sum_{k=1}^{K} [\![y_n = k]\!] \ln h_k(x_n)$

$= -\ln h_{y_n}(x_n) \Rightarrow \dfrac{\partial \text{err}}{\partial w_{y_n}} = -\dfrac{h_{y_n}'(x_n)}{h_{y_n}(x_n)} = -\dfrac{\frac{\partial}{\partial w_y}\left(\frac{\exp(w_y^T x)}{\sum \exp(w_i^T x)}\right)}{h_{y_n}(x_n)}$

$\dfrac{\partial \text{err}}{\partial w_{y_n}} = 0.$

Nonlinear Transformation

8. Given the following training data set:

-1, -1, 0, 0

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

0 0 1 -1

E

Use the quadratic transform $\mathbf{\Phi}_2(\mathbf{x}) = (1, x_1, x_2, x_1^2, x_1x_2, x_2^2)$ 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] $(0, 0, 0, 0, 0, -1)$

$\Gamma[\,x\,]$ $\Gamma[x_1\, x_2\, x_3\, x_4]$

9. Consider a feature transform $\Phi(\mathbf{x}) = \Gamma\mathbf{x}$ 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 ~~column~~ row being $\mathbf{x}_n$) satisfies that $\mathrm{X}^T\mathrm{X}$ is invertible. What is the relationship between $\mathbf{w}_{\text{lin}}$ and $\tilde{\mathbf{w}}$? Choose the correct answer; explain your answer.

C

By 25 P.14 $W_{lin} = X^\dagger y = (X^TX)^{-1}X^Ty$ $= (XX^T)^{-1}Xy$

$\frac{\partial}{\partial w}(X^TXw - X^Ty)$ $X_{dd} = (\Gamma X)^T$

$\Gamma X = [\;]$

$\frac{\partial}{\partial w}((\Gamma X)(\Gamma X)^T\tilde{w} - \Gamma X y) = 0$

$\Rightarrow \Gamma X X^T \Gamma^T \tilde{w} - \Gamma X y = 0 \Rightarrow \tilde{w} = (\Gamma X X^T \Gamma^T)^{-1}\Gamma X y$

$\Rightarrow \Gamma^{-T}(XX^T)^{-1}\Gamma^{-1}\Gamma X y$

[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 notice that all $x_1, x_2, \cdots x_n$ are distinct Dr. Trans magically decides the following transform

$$\Phi(\mathbf{x}) = ([\![\mathbf{x} = \mathbf{x}_1]\!], [\![\mathbf{x} = \mathbf{x}_2]\!], \ldots, [\![\mathbf{x} = \mathbf{x}_N]\!]).$$

That is, $\Phi(\mathbf{x})$ is a N-dimentional 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.

C

只提醒等於 x_1 whats $x_1 \sim x_N$

$\Phi(x_1) = (1, 0, 0, 0 \cdots 0)$

$\Phi(x_2) = (0, 1, 0 \cdots 0)$

$\Phi(x_N) = (0, 0, \cdots 1)$

[a] **1**, the vector of all 1s. 永遠output=1

[b] **0**, the vector of all 0s. =0

[c] $\mathbf{y}$

[d] $-\mathbf{y}$

[e] none of the other choices

$\tilde{w}^T\Phi(x)$ y_1

$[y_1, y_2, y_3 \cdots y_N]$

(Note: Be sure to also check what $E_{\text{in}}(\tilde{\mathbf{w}})$ is!) =0 吧!

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

OVA

C

OVA 選argmax 不一樣就加一

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

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

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

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

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

$[w_1\; w_2 \cdots w_K]$ e^k: squared error

最糟的情況? E^{squr} 全是0.5

$E_{in}^{0/1}(g) \leq ?$

一個mistake, 只少有2個classifier有犯錯.

① 一共有多少mistake. $N \cdot E_{in}^{0/1}(g)$ ⇒ total number of mistake

$\leq N\left(\sum_{k=1}^{K} E_{in}^{0/1}(w_{[k]}^*)\right) \times \frac{1}{2}$ 如果是等號的話要乘×1/2 tightest upper bound.

小分類器的犯錯次數加在一起

如何變成 E_{in}^{sqa}

$E_{in}^{0/1}(g) \leq E_{in}^{sqa}(g) = e^k$

e.g. w_1, w_2, w_3

$(x, 1)$ ← label

$g(x) = 2$

$w_2^Tx > w_1^Tx$ ← w_2^T, w_1^T 都錯 (至

Experiments with Linear and Nonlinear Models

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

`https://www.csie.ntu.edu.tw/~htlin/course/ml21fall/hw3/hw3_train.dat`

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

`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

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

B

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

$X \xrightarrow{\Phi} \Phi(x)$

$= \underbrace{(1, x_1, \cdots x_2, x_1^2, \cdots\cdots x_{10}^2)}_{\text{當做線}}$

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.

D

[a] 0.30

[b] 0.35

[c] 0.40

[d] 0.45

[e] 0.50

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

[a] 0.33

[b] 0.41

[c] 0.49

[d] 0.57

[e] 0.65

$[1,$ $\quad x_1 \sim x_{10}$ $\quad x_1^2 \sim x_{10}^2$

C_2^{10} $\quad x_1x_2, x_1x_3, x_1x_4 \cdots\cdots$

$x_2x_3, x_2x_4 \cdots\cdots$

$\vdots$

x_9x_{10}

21
45
66

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

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

$x_0 = 1$

C

Which i lead to a $\Phi^{(i)}$ that reaches the minimum

Run $\mathbf{\Phi}^{(i)}$ + linear regression to get a hypothesis g_i. ~~What is the minimum~~ $\left|E_{\text{in}}^{0/1}(g_i) - E_{\text{out}}^{0/1}(g)\right|$ ~~over i~~? Choose the closest answer; provide your code.

[a] 1

[b] 2

[c] 3

[d] 5

[e] 8

1	2	3	4	5	6	7	8	9	10
0.136	0.134	0.132	0.144	0.25	0.32	0.26	0.26	0.24	0.32

16. (*) Consider a transform that randomly chooses 5 out of 10 dimensions. That is, $\mathbf{\Phi}(\mathbf{x}) = (x_0, x_{i_1}, x_{i_2}, x_{i_3}, x_{i_4}, x_{i_5})$, where i_1 to i_5 are distinct random integers uniformly and independently generated within $\{1, 2, \ldots, 10\}$. Run $\mathbf{\Phi}$ + 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 $\mathbf{\Phi}$ with a different random seed? Choose the closest answer; provide your code.

D

[a] 0.06

[b] 0.11

[c] 0.16

[d] 0.21

[e] 0.26