

# 作业一

测验, 20 个问题

10  
points

10  
points

1.

Which of the following problems are best suited for machine learning?

质数

(i) Classifying numbers into primes and non-primes → 可用公式解决.

检测 潜在 作弊 信用

(ii) Detecting potential fraud in credit card charges

(iii) Determining the time it would take a falling object to hit the ground → 可用公式解决

最佳

交通 路口

(iv) Determining the optimal cycle for traffic lights in a busy intersection

路口

特定 医疗

(v) Determining the age at which a particular medical test is recommended

推荐

(i) and (ii)

(ii), (iv), and (v)

(i), (ii), (iii), and (iv)

none of the other choices

10  
points

2.

# 作業一

测验, 20 个问题

For Questions 2-5, identify the best type of learning that can be used to solve each task below.

西洋棋

任務

賽駒

策略

接受

Play chess better by practicing different strategies and receive outcome as feedback.

unsupervised learning

reinforcement learning

none of other choices

supervised learning

active learning

10  
points

3. 分類

Categorize books into groups without pre-defined topics.

主題

supervised learning

reinforcement learning

none of other choices

active learning

unsupervised learning

10  
points

4.

Recognize whether there is a face in the picture by a thousand face pictures and ten thousand nonface pictures.

active learning

## 作業一

测验, 20 个问题

supervised learning

reinforcement learning

none of other choices

unsupervised learning

10  
points

5. 開擇 排程 實驗 老鼠 評估

Selectively schedule experiments on mice to quickly evaluate the potential of cancer medicines.

潛在

none of other choices

active learning

unsupervised learning

supervised learning

reinforcement learning

10  
points

6.

# 作業一

Question 6-8 are about Off-Training-Set error.

测验, 20 个问题

Let  $\mathcal{X} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N, \mathbf{x}_{N+1}, \dots, \mathbf{x}_{N+L}\}$  and  $\mathcal{Y} = \{-1, +1\}$  (binary classification). Here the set of training examples is

$\mathcal{D} = \left\{ (\mathbf{x}_n, y_n) \right\}_{n=1}^N$ , where  $y_n \in \mathcal{Y}$ , and the set of test inputs is  $\left\{ \mathbf{x}_{N+\ell} \right\}_{\ell=1}^L$ . The Off-Training-Set error ( $OTS$ ) with respect to an underlying target  $f$  and a hypothesis  $g$  is

$$E_{OTS}(g, f) = \frac{1}{L} \sum_{\ell=1}^L [[g(\mathbf{x}_{N+\ell}) \neq f(\mathbf{x}_{N+\ell})]].$$

Consider  $f(\mathbf{x}) = +1$  for all  $\mathbf{x}$  and

$$g(\mathbf{x}) = \begin{cases} +1, & \text{for } \mathbf{x} = \mathbf{x}_k \text{ and } k \text{ is odd and } 1 \leq k \leq N+L \\ -1, & \text{otherwise} \end{cases}$$

$E_{OTS}(g, f) = ?$  (Please note the difference between floor and ceiling functions in the choices)

$\frac{1}{L} \times (\lfloor \frac{N+L}{2} \rfloor - \lceil \frac{N}{2} \rceil)$

$\frac{1}{L} \times (\lceil \frac{N+L}{2} \rceil - \lceil \frac{N}{2} \rceil)$

$\frac{1}{L} \times (\lceil \frac{N+L}{2} \rceil - \lfloor \frac{N}{2} \rfloor)$

$\frac{1}{L} \times (\lfloor \frac{N+L}{2} \rfloor - \lceil \frac{N}{2} \rceil)$

none of the other choices

10  
points

7。

# 作業一

测验, 20 个问题

We say that a target function  $f$  can "generate"  $\mathcal{D}$  in a noiseless setting if  $f(\mathbf{x}_n) = y_n$  for all  $(\mathbf{x}_n, y_n) \in \mathcal{D}$ .

For all possible  $f: \mathcal{X} \rightarrow \mathcal{Y}$ , how many of them can generate  $\mathcal{D}$  in a noiseless setting?

Note that we call two functions  $f_1$  and  $f_2$  the same if  $f_1(\mathbf{x}) = f_2(\mathbf{x})$  for all  $\mathbf{x} \in \mathcal{X}$ .

)  $2^{N+L}$

$2^L$

$2^N$

1

none of the other choices

10  
points

8.

A deterministic algorithm  $\mathcal{A}$  is defined as a procedure that takes  $\mathcal{D}$  as an input, and outputs a hypothesis  $g$ . For any two deterministic algorithms  $\mathcal{A}_1$  and  $\mathcal{A}_2$ , if all those  $f$  that can "generate"  $\mathcal{D}$  in a noiseless setting are equally likely in probability,

$$\mathbb{E}_f \left\{ E_{OTS}(\mathcal{A}_1(\mathcal{D}), f) \right\} = \mathbb{E}_f \left\{ E_{OTS}(\mathcal{A}_2(\mathcal{D}), f) \right\}.$$

For any given  $f$  that "generates"  $\mathcal{D}$ ,

$$E_{OTS}(\mathcal{A}_1(\mathcal{D}), f) = E_{OTS}(\mathcal{A}_2(\mathcal{D}), f).$$

$$\mathbb{E}_f \left\{ E_{OTS}(\mathcal{A}_1(\mathcal{D}), f) \right\} = \mathbb{E}_f \left\{ E_{OTS}(f, f) \right\}.$$

# 作业一

测验, 20 个问题

none of the other choices

- ( ) For any given  $f'$  that does not "generate"  $\mathcal{D}$ ,

$$\left\{ E_{OTS}(\mathcal{A}_1(\mathcal{D}), f') \right\} = \left\{ E_{OTS}(\mathcal{A}_2(\mathcal{D}), f') \right\}.$$

10  
points

9.

For Questions 9-12, consider the bin model introduced in class. Consider a bin with infinitely many marbles, and let  $\mu$  be the fraction of orange marbles in the bin, and  $\nu$  is the fraction of orange marbles in a sample of 10 marbles. If  $\mu = 0.5$ , what is the probability of  $\nu = \mu$ ? Please choose the closest number.

) 0.39

( ) 0.90

( ) 0.24.

( ) 0.56

( ) 0.12

$$\begin{aligned} & C_5^{10} \cdot (0.5)^5 (0.5)^5 \\ &= \frac{10 \cdot 9 \cdot 8 \cdot 7 \cdot 6}{5 \cdot 4 \cdot 3 \cdot 2} \cdot \frac{1}{2^5} = \frac{63}{2^8} = 0.24609 \dots \end{aligned}$$

10  
points

10.

If  $\mu = 0.9$ , what is the probability of  $\nu = \mu$ ? Please choose the closest number.

0.12

0.39

$$C_9^{10} (0.9)^9 (0.1)^1 = 0.39$$

作業一

0.56

测验, 20 个问题

( 0.90

0.24

10  
points

11.

If  $\mu = 0.9$ , what is the actual probability of  $\nu \leq 0.1$ ?

(  $8.5 \times 10^{-9}$

(  $9.1 \times 10^{-9}$

(  $1.0 \times 10^{-9}$

(  $0.1 \times 10^{-9}$

(  $4.8 \times 10^{-9}$

$$\begin{aligned} & \underbrace{\binom{10}{0} (0.9)^0 (0.1)^0}_{V=0} = 10^{-10} \\ & + \underbrace{\binom{10}{1} (0.9)^1 (0.1)^9}_{V=0.1} = 9 \cdot 10^{-9} \\ & 9.1 \cdot 10^{-9} \end{aligned}$$

10  
points

12.

If  $\mu = 0.9$ , what is the bound given by Hoeffding's Inequality for the probability of  $\nu \leq 0.1$ ?

(  $5.52 \times 10^{-12}$

(  $5.52 \times 10^{-6}$

(  $5.52 \times 10^{-4}$

(  $5.52 \times 10^{-8}$

(  $5.52 \times 10^{-10}$

# 作业一

测验, 20 个问题

10  
points

13.

Questions 13-14 illustrate what happens with multiple bins using dice to indicate 6 bins. Please note that the dice is not meant to be thrown for random experiments in this problem. They are just used to bind the six faces together. The probability below only refers to drawing from the bag.

Consider four kinds of dice in a bag, with the same (super large) quantity for each kind.

A: all even numbers are colored orange, all odd numbers are colored green

B: all even numbers are colored green, all odd numbers are colored orange

C: all small (1~3) are colored orange, all large numbers (4~6) are colored green

D: all small (1~3) are colored green, all large numbers (4~6) are colored orange

If we pick 5 dice from the bag, what is the probability that we get 5 orange 1's?

$\frac{1}{256}$

$\frac{8}{256}$

$\frac{31}{256}$

$\frac{46}{256}$

none of the other choices

Orange 1's  $\Rightarrow$  B or C.

$$(0.5)^5 = \frac{1}{32}$$

10  
points

14.

# 作業一

测验, 20 个问题

If we pick 5 dice from the bag, what is the probability that we get "some number" that is purely orange?

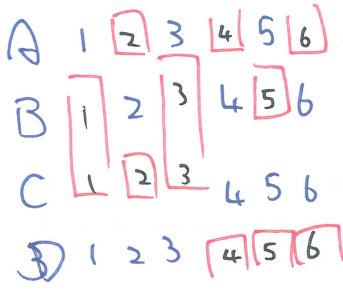
$\frac{1}{256}$

$\frac{8}{256}$

$\frac{31}{256}$

$\frac{46}{256}$

none of the other choices



$$\frac{5}{2} * 4 - 4 = \frac{5}{4}$$

10  
points

Code.

15.

For Questions 15-20, you will play with PLA and pocket algorithm. First, we use an artificial data set to study PLA. The data set is in

<http://ppt.cc/CDqBU>

Each line of the data set contains one  $(\mathbf{x}_n, y_n)$  with  $\mathbf{x}_n \in \mathbb{R}^4$ . The first 4 numbers of the line contains the components of  $\mathbf{x}_n$  orderly, the last number is  $y_n$ .

Please initialize your algorithm with  $\mathbf{w} = 0$  and take  $\text{sign}(0)$  as  $-1$

Implement a version of PLA by visiting examples in the naive cycle using the order of examples in the data set. Run the algorithm on the data set. What is the number of updates before the algorithm halts?

< 10 updates

11 - 30 updates

31 - 50 updates

$\geq 201$  updates

51 - 200 updates

# 作业一

测验, 20 个问题

10  
points

16.

Implement a version of PLA by visiting examples in fixed, pre-determined random cycles throughout the algorithm. Run the algorithm on the data set. Please repeat your experiment for 2000 times, each with a different random seed. What is the average number of updates before the algorithm halts?

- < 10 updates
- 11 - 30 updates
- 31 - 50 updates
- $\geq 201$  updates
- 51 - 200 updates

10  
points

17.

Implement a version of PLA by visiting examples in fixed, pre-determined random cycles throughout the algorithm, while changing the update rule to be

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

with  $\eta = 0.5$ . Note that your PLA in the previous Question corresponds to  $\eta = 1$ . Please repeat your experiment for 2000 times, each with a different random seed. What is the average number of updates before the algorithm halts?

- < 10 updates
-

## 作業一

测验, 20 个问题

- 11 - 30 updates
- 31 - 50 updates
- $\geq 201$  updates
- 51 - 200 updates

10  
points

18.

Next, we play with the pocket algorithm. Modify your PLA in Question 16 to visit examples purely randomly, and then add the "pocket" steps to the algorithm. We will use

<http://ppt.cc/lk83i>

as the training data set  $\mathcal{D}$ , and

<http://ppt.cc/EwKVZ>

as the test set for "verifying" the  $g$  returned by your algorithm (see lecture 4 about verifying). The sets are of the same format as the previous one. Run the pocket algorithm with a total of 50 updates on  $\mathcal{D}$ , and verify the performance of  $\mathbf{w}_{POCKET}$  using the test set. Please repeat your experiment for 2000 times, each with a different random seed. What is the average error rate on the test set?

- $< 0.2$
- 0.2 - 0.4
- 0.4 - 0.6
- $\geq 0.8$
- 0.6 - 0.8

## 作業一

测验, 20 个问题

10  
points

19.

Modify your algorithm in Question 18 to return  $\mathbf{w}_{50}$  (the PLA vector after 50 updates) instead of  $\hat{\mathbf{w}}$  (the pocket vector) after 50 updates.

Run the modified algorithm on  $\mathcal{D}$ , and verify the performance using the test set.

Please repeat your experiment for 2000 times, each with a different random seed. What is the average error rate on the test set?

- < 0.2
- 0.2 - 0.4
- 0.4 - 0.6
- $\geq 0.8$
- 0.6 - 0.8

10  
points

20.

Modify your algorithm in Question 18 to run for 100 updates instead of 50, and verify the performance of  $\mathbf{w}_{POCKET}$  using the test set. Please repeat your experiment for 2000 times, each with a different random seed. What is the average error rate on the test set?

- < 0.2
- 0.2 - 0.4
- 0.4 - 0.6
- $\geq 0.8$
-

0.6 - 0.8

## 作业一

测验, 20 个问题



我 (伟臣沈) 了解提交不是我自己完成的作业 将永远不会通过此课程或导致我的 Coursera 帐号被关闭。

[了解荣誉准则的更多信息](#)

[提交测试](#)



