

程序代写代做 CS编程辅导



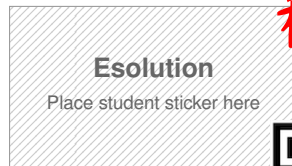
WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

<https://tutorcs.com>



程序代写代做 CS编程辅导

Note:

- During the attendance check a sticker containing a unique code will be put on this exam.
- This code contains a unique number that associates this exam with your registration number.
- This number is printed both next to the code and to the signature field in the attendance check



Machine Learning

Graded Exercise: IN2064 / Retake

Date: Thursday 1st April, 2021

Examiner: Prof. Dr. Stephan Günnemann

Time: 16:30 – 18:30

WeChat: cstutorcs

Working instructions

- This graded exercise consists of **pages with a total of 31 problems**. Please make sure now that you received a complete copy of the answer sheet.
- The total amount of achievable credits in this graded exercise is 108 credits.
- Allowed resources:
 - all materials that you will use on your own (lecture slides, calculator etc.)
 - **not allowed are any forms of collaboration between examinees and plagiarism**
- You have to sign the code of conduct. (Typing your name is fine)
- You have to either print this document and scan your solutions or paste scans/pictures of your handwritten solutions into the solution boxes in this PDF. **Editing the PDF digitally is prohibited except for signing the code of conduct and answering multiple choice questions.**
- Make sure that the **QR codes are visible** on every uploaded page. Otherwise, we cannot grade your submission.
- **You must solve the specified version of the problem.** Different problems may have different version: e.g. Problem 1 (Version A), Problem 5 (Version C), etc. If you solve the wrong version you get **zero** points.
- Only write on the provided sheets, **submitting your own additional sheets is not possible.**
- Last two pages can be used as scratch paper.
- All sheets (including scratch paper) have to be submitted to the upload queue. Missing pages will be considered empty.
- **Only use a black or blue color (no red or green)! Pencils are allowed.**
- Write your answers only in the provided solution boxes or the scratch paper.
- **For problems that say "Justify your answer" you only get points if you provide a valid explanation.**
- **For problems that say "Prove" you only get points if you provide a valid mathematical proof.**
- If a problem does not say "Justify your answer" or "Prove" it's sufficient to only provide the correct answer.
- Instructor announcements and clarifications will be posted **on Piazza** with email notifications.
- Exercise duration - 120 minutes.

Left room from _____ to _____ / Early submission at _____

Problem 1 (Version A) (4 credits)

a)

程序代写代做 CS编程辅导

Yes, it is possible, however the likelihood includes the unobserved parameter θ . So to maximize the likelihood, either θ needs to be marginalized out first

$$p(\theta | a, b) d\theta = \frac{\mathbb{E}}{p(\theta | a, b)} [p(\mathcal{D} | \theta)]$$

or one has to use an approximation such as Expectation Maximization.



WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

b)

QQ: 749389476

No, it is not immediately possible because for MAP estimation you need a prior on the variables you want to infer. So you would need to introduce a hyper-prior on the parameters a and b , for example an $\text{Exponential}(\lambda)$ distribution. Then you can compute an MAP estimate by maximizing $p(\mathcal{D} | a, b) p(a, b)$ where the data likelihood is computed as in a).

<https://tutorcs.com>

Problem 1 (Version B) (4 credits)

0
1
2

a)

程序代写代做 CS编程辅导

Yes, it is possible, however the likelihood includes the unobserved parameter θ . So to maximize the likelihood, either θ needs to be marginalized out first



$$\int p(\mathcal{D} | \theta) p(\theta | a, b) d\theta = \mathbb{E}_{p(\theta | a, b)} [p(\mathcal{D} | \theta)]$$

or one has to use an algorithm such as Expectation Maximization.

WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

0
1
2

b)

QQ: 749389476

No, it is not immediately possible because for MAP estimation you need a prior on the variables you want to infer. So you would need to introduce a hyper-prior on the parameters a and b , for example an Exponential(λ) distribution. Then you can compute an MAP estimate by maximizing $p(\mathcal{D} | a, b) p(a, b)$ where the data likelihood is computed as in a).

<https://tutorcs.com>

Problem 2 (Version A) (5 credits)

a)

Bagging at feature level.

程序代写代做 CS编程辅导



b)

There are many valid solutions. One solution is $\mathcal{D} = \{([a, b], 0), ([a, b], 1)\}$, where we have two instances with exactly the same features $\mathbf{x}_1 = \mathbf{x}_2 = [a, b] \in \mathbb{R}^2$ for any constants a, b , but different labels, $y_1 \neq y_2$. No matter what the split is, both instances will end up in the same leaf and the purity will not change.

Another solution is to write down the XOR dataset.

WeChat: cstutorcs

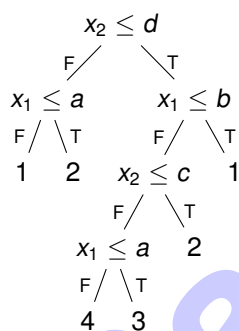
Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

c)

The decision tree looks as follows:



https://tutorcs.com

Problem 2 (Version B) (5 credits)

0
1

a)

程序代写代做 CS编程辅导

Bagging at feature level.



0
1
2

b)

There are many valid solutions. One solution is $\mathcal{D} = \{([a, b], 0), ([a, b], 1)\}$, where we have two instances with exactly the same features $\mathbf{x}_1 = \mathbf{x}_2 = [a, b] \in \mathbb{R}^2$ for any constants a, b , but different labels, $y_1 \neq y_2$. No matter what the split is, both instances will end up in the same leaf and the purity will not change.

Another solution is to write down the XOR dataset.

WeChat: estutorcs

Assignment Project Exam Help

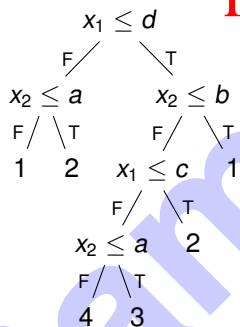
Email: tutorcs@163.com

QQ: 749389476

0
1
2

c)

The decision tree looks as follows:



<https://tutorcs.com>

Problem 2 (Version C) (5 credits)

a)

Bagging at feature level.

程序代写代做 CS编程辅导



b)

There are many valid solutions. One solution is $\mathcal{D} = \{([a, b], 0), ([a, b], 1)\}$, where we have two instances with exactly the same features $\mathbf{x}_1 = \mathbf{x}_2 = [a, b] \in \mathbb{R}^2$ for any constants a, b , but different labels, $y_1 \neq y_2$. No matter what the split is, both instances will end up in the same leaf and the purity will not change.

Another solution is to write down the XOR dataset.

WeChat: cstutorcs

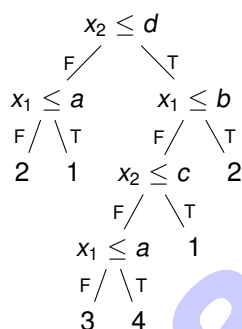
Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

c)

The decision tree looks as follows:



<https://tutorcs.com>

Problem 2 (Version D) (5 credits)

0
1

a)

程序代写代做 CS编程辅导

Bagging at feature level.



0
1
2

b)

There are many valid solutions. One solution is $\mathcal{D} = \{([a, b], 0), ([a, b], 1)\}$, where we have two instances with exactly the same features $\mathbf{x}_1 = \mathbf{x}_2 = [a, b] \in \mathbb{R}^2$ for any constants a, b , but different labels, $y_1 \neq y_2$. No matter what the split is, both instances will end up in the same leaf and the purity will not change.

Another solution is to write down the XOR dataset.

WeChat: estutorcs

Assignment Project Exam Help

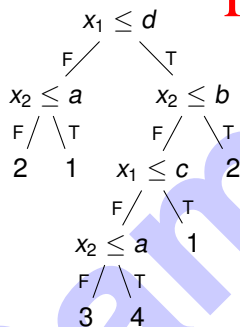
Email: tutorcs@163.com

QQ: 749389476

0
1
2

c)

The decision tree looks as follows:



<https://tutorcs.com>

Problem 3 (Version A) (2 credits)

程序代写代做 CS编程辅导

f is linear in the parameters $\mathbf{w} = (a \ b \ c)^T$ which becomes more apparent if we rewrite it as


$$\mathbf{v}^T \begin{pmatrix} \sin(\mathbf{x}_2) \\ \frac{1}{2} \|\mathbf{x}\|_1 \\ -\mathbf{x}_1^2 \mathbf{x}_2 \end{pmatrix}.$$

In this form, the problem is linear

We define a feature transformation

$$\begin{pmatrix} \sin(\mathbf{x}_2) \\ \frac{1}{2} \|\mathbf{x}\|_1 \\ -\mathbf{x}_1^2 \mathbf{x}_2 \end{pmatrix}.$$

and can now write the problem in the usual linear regression form

WeChat: cstutorcs

where $\mathbf{y}_i = y_i$ and $\Phi_{i,:} = \phi(\mathbf{x}_i)^T$. Then we can apply the closed form for ordinary least squares and get

Assignment Project Exam Help

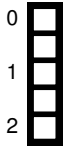
Email: tutorcs@163.com

QQ: 749389476

<https://tutorcs.com>



Problem 3 (Version B) (2 credits)



程序代写代做 CS编程辅导

f is linear in the parameters $\mathbf{w} = (a \ b \ c)^T$ which becomes more apparent if we rewrite it as



$$f(\mathbf{x}, \mathbf{w}) = \mathbf{w}^T \begin{pmatrix} \|\mathbf{x}\|_2 \\ -\frac{1}{2}\mathbf{x}_1^2\mathbf{x}_2 \\ \cos(\mathbf{x}_1) \end{pmatrix}.$$

In this form, the problem is linear in the parameters. We define a feature transformation

$$\phi(\mathbf{x}) = \begin{pmatrix} \|\mathbf{x}\|_2 \\ -\frac{1}{2}\mathbf{x}_1^2\mathbf{x}_2 \\ \cos(\mathbf{x}_1) \end{pmatrix}.$$

and can now write the problem in the usual linear regression form

WeChat: cstutorcs

where $\mathbf{y}_i = y_i$ and $\Phi_{i,:} = \phi(\mathbf{x}_i)^T$. Then we can apply the closed form for ordinary least squares and get

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

<https://tutorcs.com>

Problem 3 (Version C) (2 credits)

程序代写代做 CS编程辅导

f is linear in the parameters $\mathbf{w} = (a \ b \ c)^T$ which becomes more apparent if we rewrite it as


$$y^T \begin{pmatrix} -x_1^2 x_2 \\ \tan(x_2) \\ \frac{1}{2} \|\mathbf{x}\|_\infty \end{pmatrix}.$$

In this form, the problem is linear

We define a feature transformation

$$\begin{pmatrix} -x_1^2 x_2 \\ \tan(x_2) \\ \frac{1}{2} \|\mathbf{x}\|_\infty \end{pmatrix}.$$

and can now write the problem in the usual linear regression form

WeChat: cstutorcs

where $\mathbf{y}_i = y_i$ and $\Phi_{i,:} = \phi(\mathbf{x}_i)^T$. Then we can apply the closed form for ordinary least squares and get

Assignment, Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

<https://tutorcs.com>



Problem 4 (Version A) (6 credits)

0
1
2

a)

程序代写代做 CS编程辅导

A weight \mathbf{w}^* on \mathcal{D} is equivalent to a weight $\mathbf{w}^{*,\alpha} = \mathbf{w}^*/\alpha$ on \mathcal{D}_α and vice versa. The norm of \mathbf{w}^* will stay finite because the dataset is not linearly separable. Therefore, the two classifiers will reach the same log-likelihood and for the optimal $\mathbf{w}^{*,\alpha} = \mathbf{w}^*/\alpha$. So for any \mathbf{x}_{test} with $\mathbf{w}^* \cdot \mathbf{x}_{\text{test}} = 0$ (i.e. on the decision boundary for a class 5), we will have $s = t$.

A logistic regression



$$f(\mathbf{x}; \mathbf{w}) = \sigma(\mathbf{x}^T \mathbf{w})$$

and the sigmoid function, meaning that $\mathbf{x}^T \mathbf{w} < \mathbf{x}^T \mathbf{w}' \Rightarrow \sigma(\mathbf{x}^T \mathbf{w}) < \sigma(\mathbf{x}^T \mathbf{w}')$. Because of $\alpha > 1$,

$$|\mathbf{x}_{\text{test}}^T \mathbf{w}^{*,\alpha}| = \alpha^{-1} |\mathbf{w}^* \cdot \mathbf{x}_{\text{test}}| < |\mathbf{w}^* \cdot \mathbf{x}_{\text{test}}|.$$

So we can have both $s < t$ and $s > t$ depending on the sign of $\mathbf{w}^* \cdot \mathbf{x}_{\text{test}}$.

WeChat: cstutorcs

0
1
2

b)

Assignment Project Exam Help

In the unregularized setting, the MLE will have infinite norm. But with a weight vector \mathbf{w} of infinite norm $\|\mathbf{w}\| = \infty$, the model $f(\mathbf{x}; \mathbf{w})$ will only take on three values: 0 or 1 if \mathbf{x} is on either side of the hyperplane defined by \mathbf{w} or $\frac{1}{2}$ if \mathbf{x} is exactly on the hyperplane. Therefore any \mathbf{w} such that the mapping $\mathbf{w} \cdot \mathbf{x}$ linearly separates the data are equivalent. Because in this setup class 1 is truly contained in the first quadrant and class 0 in the third, any hyperplane that separates the two achieves the same log-likelihood, so the optimal \mathbf{w} is not unique.

Email: tutores@163.com

QQ: 749389476

https://tutorcs.com

0
1
2

c)

$$\mathbf{w}^{*,a} = \begin{pmatrix} \infty \\ 0 \end{pmatrix} \quad \text{and} \quad \mathbf{w}^{*,b} = \begin{pmatrix} 0 \\ \infty \end{pmatrix}.$$

As explained in the previous problem, any hyperplane that separates the two quadrants will work and so we can choose the y and x axes. A differently classified test point is

$$\mathbf{x}_{\text{test}} = \begin{pmatrix} -1 \\ 1 \end{pmatrix}$$

because

$$f(\mathbf{x}_{\text{test}}; \mathbf{w}^{*,a}) = \sigma(-\infty) = 0 < \frac{1}{2} < 1 = \sigma(\infty) = f(\mathbf{x}_{\text{test}}; \mathbf{w}^{*,b}).$$

Problem 4 (Version B) (6 credits)

a)

程序代写代做 CS编程辅导

A weight \mathbf{w}^* on \mathcal{D} is equivalent to a weight $\mathbf{w}^{*,\alpha} = \mathbf{w}^*/\alpha$ on \mathcal{D}_α and vice versa. The norm of \mathbf{w}^* will stay finite because the dataset is not linearly separable. Therefore, the two classifiers will reach the same log-likelihood and for the optimal weights it holds $\alpha \cdot \mathbf{w}^* \cdot \mathbf{x}_{\text{test}} = 0$ (i.e. on the decision boundary for a classifier with weights \mathbf{w}^*) have $s = t$.

A logistic regression model is



$$= \sigma(\mathbf{x}^T \mathbf{w})$$

and the sigmoid function is $\sigma(x) = \frac{1}{1 + e^{-x}}$

Because of $\alpha > 1$, we have

$$\text{Since } \mathbf{x}^T \mathbf{w} < \mathbf{x}^T \mathbf{w}' \Rightarrow \sigma(\mathbf{x}^T \mathbf{w}) < \sigma(\mathbf{x}^T \mathbf{w}').$$

$$|\mathbf{w}^* \cdot \mathbf{x}_{\text{test}}| < |\mathbf{w}^* \cdot \mathbf{x}_{\text{test}}|.$$

So we can have both $s < t$ and $s > t$ depending on the sign of $\mathbf{w}^* \cdot \mathbf{x}_{\text{test}}$.

WeChat: cstutorcs

b)

Assignment Project Exam Help

In the unregularized setting, the MLE will have infinite norm. But with a weight vector \mathbf{w} of infinite norm $\|\mathbf{w}\| = \infty$, the model $f(\mathbf{x}; \mathbf{w})$ will only take on three values: 0 or 1 if \mathbf{x} is on either side of the hyperplane defined by \mathbf{w} or $\frac{1}{2}$ if \mathbf{x} is exactly on the hyper plane. Therefore any \mathbf{w} such that the mapping $\mathbf{w} \cdot \mathbf{x}$ linearly separates the data are equivalent. Because in this setup class 1 is truly contained in the first quadrant and class 0 in the third, any hyperplane that separates the two achieves the same log-likelihood, so the optimal \mathbf{w} is not unique.

Email: tutores@163.com

QQ: 749389476

<https://tutorcs.com>

c)

$$\mathbf{w}^{*,a} = \begin{pmatrix} \infty \\ 0 \end{pmatrix} \quad \text{and} \quad \mathbf{w}^{*,b} = \begin{pmatrix} 0 \\ \infty \end{pmatrix}.$$

As explained in the previous problem, any hyperplane that separates the two quadrants will work and so we can choose the y and x axes. A differently classified test point is

$$\mathbf{x}_{\text{test}} = \begin{pmatrix} -1 \\ 1 \end{pmatrix}$$

because

$$f(\mathbf{x}_{\text{test}}; \mathbf{w}^{*,a}) = \sigma(-\infty) = 0 < \frac{1}{2} < 1 = \sigma(\infty) = f(\mathbf{x}_{\text{test}}; \mathbf{w}^{*,b}).$$

Problem 5 (Version A) (3 credits)

0
1

a)

程序代写代做 CS编程辅导

Yes, since the line search procedure is guaranteed to find a point where the objective function is at least as low as at the previous iteration.

Specifically, if the gradient where the function is differentiable. If the



ne search is guaranteed to find a point in the direction of the lower. Such point always exists since the function f is continuously differentiable. The search will produce $\theta_{t+1} = \theta_t$, which means that $f(\theta_{t+1}) = f(\theta_t)$.

WeChat: cstutorcs

0
1

b)

Assignment Project Exam Help

No. If the step size is too large, we may overshoot and land at a point that has a higher value of the objective function.

Email: tutorcs@163.com

QQ: 749389476

<https://tutorcs.com>

0
1

c)

No. The adaptive step size can still be too large, and we still can overshoot, similar to the fixed size case.

Problem 5 (Version B) (3 credits)

a)

程序代写代做 CS编程辅导

No. If the step size is too large, we may overshoot and land at a point that has a higher value of the objective function.



WeChat: cstutorcs



b)

Assignment Project Exam Help

No. The adaptive step size can still be too large, and we still can overshoot, similar to the fixed size case.

Email: tutorcs@163.com

QQ: 749389476

<https://tutorcs.com>



c)

Yes, since the line search procedure is guaranteed to find a point where the objective function is at least as low as at the previous iteration.

Specifically, if the gradient is nonzero, line search is guaranteed to find a point in the direction of the gradient where the objective function is lower. Such point always exists since the function f is continuously differentiable. If the gradient is zero, line search will produce $\theta_{t+1} = \theta_t$, which means that $f(\theta_{t+1}) = f(\theta_t)$.



Problem 5 (Version C) (3 credits)

0
1

a)

程序代写代做 CS编程辅导

Yes, since the line search procedure is guaranteed to find a point where the objective function is at least as low as at the previous iteration.

Specifically, if the gradient where the function is differentiable. If the



ne search is guaranteed to find a point in the direction of the lower. Such point always exists since the function f is continuously differentiable. The search will produce $\theta_{t+1} = \theta_t$, which means that $f(\theta_{t+1}) = f(\theta_t)$.

WeChat: cstutorcs

0
1

b)

Assignment Project Exam Help

No. The adaptive step size can still be too large, and we still can overshoot, similar to the fixed size case.

Email: tutorcs@163.com

QQ: 749389476

<https://tutorcs.com>

0
1

c)

No. If the step size is too large, we may overshoot and land at a point that has a higher value of the objective function.

Problem 5 (Version D) (3 credits)

a)

程序代写代做 CS编程辅导

No. The adaptive step size can still be too large, and we still can overshoot, similar to the fixed size case.



WeChat: cstutorcs



b)

Assignment Project Exam Help

No. If the step size is too large, we may overshoot and land at a point that has a higher value of the objective function.

Email: tutorcs@163.com

QQ: 749389476

<https://tutorcs.com>



c)

Yes, since the line search procedure is guaranteed to find a point where the objective function is at least as low as at the previous iteration.

Specifically, if the gradient is nonzero, line search is guaranteed to find a point in the direction of the gradient where the objective function is lower. Such point always exists since the function f is continuously differentiable. If the gradient is zero, line search will produce $\theta_{t+1} = \theta_t$, which means that $f(\theta_{t+1}) = f(\theta_t)$.



Problem 6 (Version A) (3 credits)

0 ☐
1 ☐

a)

程序代写代做 CS编程辅导

Option 1:

```
out = np.log(1 + np.exp(x @ y))
```

Option 2:

```
out = np.log1p(np
```

We can also replac



y) or x.dot(y).

WeChat: cstutorcs

Assignment Project Exam Help

0 ☐
1 ☐
2 ☐

b)

Email: tutorcs@163.com

QQ: 749389476

The gradients w.r.t. x and y are

$$\frac{\partial}{\partial x} \log(1 + \exp(\mathbf{x}^T \mathbf{y})) = \frac{1}{1 + \exp(\mathbf{x}^T \mathbf{y})} \exp(\mathbf{x}^T \mathbf{y}) \mathbf{y}^T$$
$$\frac{\partial}{\partial y} \log(1 + \exp(\mathbf{x}^T \mathbf{y})) = \frac{1}{1 + \exp(\mathbf{x}^T \mathbf{y})} \exp(\mathbf{x}^T \mathbf{y}) \mathbf{x}^T$$

We can implement these computations in Numpy as

```
s = np.exp(x @ y) / (1 + np.exp(x @ y))
```

```
d_x = d_out * s * y
```

```
d_y = d_out * s * x
```

We can also replace

```
s = np.exp(x @ y) / (1 + np.exp(x @ y))
```

with

```
s = 1 / (1 + np.exp(-x @ y))
```

since these are two equivalent ways to compute the sigmoid function.

<https://tutorcs.com>

Problem 6 (Version B) (3 credits)

a)

程序代写代做 CS编程辅导

Option 1:

```
out = np.log(np.exp(x @ y) - 1)
```

Option 2:

```
out = np.log(np.expm1(x @
```

We can also replace $x @ y$ w



dot(y).

WeChat: cstutorcs

Assignment Project Exam Help

b)

The gradients w.r.t. \mathbf{x} and \mathbf{y} are

$$\frac{\partial}{\partial \mathbf{x}} \log(\exp(\mathbf{x}^T \mathbf{y}) - 1) = \frac{1}{\exp(\mathbf{x}^T \mathbf{y}) - 1} \exp(\mathbf{x}^T \mathbf{y}) \mathbf{y}^T$$
$$\frac{\partial}{\partial \mathbf{y}} \log(\exp(\mathbf{x}^T \mathbf{y}) - 1) = \frac{1}{\exp(\mathbf{x}^T \mathbf{y}) - 1} \exp(\mathbf{x}^T \mathbf{y}) \mathbf{x}^T$$

We can implement these computations in Numpy as

```
s = np.exp(x @ y) / (np.exp(x @ y) - 1)
```

```
d_x = d_out * s * y
```

```
d_y = d_out * s * x
```

Email: tutorcs@163.com

QQ: 749389476

<https://tutorcs.com>

Problem 6 (Version C) (3 credits)

0
1

a)

程序代写代做 CS编程辅导

Option 1:

```
out = np.log(1 + np.exp(x @ y))
```

Option 2:

```
out = np.log1p(np
```

We can also replac



y) or x.dot(y).

WeChat: cstutorcs

Assignment Project Exam Help

0
1
2

b)

The gradients w.r.t. x and y are

Email: tutorcs@163.com

$$\frac{\partial}{\partial x} \log(1 + \exp(\mathbf{x}^T \mathbf{y})) = \frac{1}{1 + \exp(\mathbf{x}^T \mathbf{y})} \exp(\mathbf{x}^T \mathbf{y}) \mathbf{y}^T$$
$$\frac{\partial}{\partial y} \log(1 + \exp(\mathbf{x}^T \mathbf{y})) = \frac{1}{1 + \exp(\mathbf{x}^T \mathbf{y})} \exp(\mathbf{x}^T \mathbf{y}) \mathbf{x}^T$$

QQ: 749389476

We can implement these computations in Numpy as

```
s = np.exp(x @ y) / (1 + np.exp(x @ y))
```

```
d_x = d_out * s * y
```

```
d_y = d_out * s * x
```

We can also replace

```
s = np.exp(x @ y) / (1 + np.exp(x @ y))
```

with

```
s = 1 / (1 + np.exp(-x @ y))
```

since these are two equivalent ways to compute the sigmoid function.

<https://tutorcs.com>

Problem 6 (Version D) (3 credits)

a)

程序代写代做 CS编程辅导

Option 1:

```
out = np.log(np.exp(x @ y) - 1)
```

Option 2:

```
out = np.log(np.expm1(x @
```

We can also replace $x @ y$ w



dot(y).

WeChat: cstutorcs

Assignment Project Exam Help

b)

The gradients w.r.t. \mathbf{x} and \mathbf{y} are

Email: tutorcs@163.com

$$\frac{\partial}{\partial \mathbf{x}} \log(\exp(\mathbf{x}^T \mathbf{y}) - 1) = \frac{1}{\exp(\mathbf{x}^T \mathbf{y}) - 1} \exp(\mathbf{x}^T \mathbf{y}) \mathbf{y}^T$$
$$\frac{\partial}{\partial \mathbf{y}} \log(\exp(\mathbf{x}^T \mathbf{y}) - 1) = \frac{1}{\exp(\mathbf{x}^T \mathbf{y}) - 1} \exp(\mathbf{x}^T \mathbf{y}) \mathbf{x}^T$$

QQ: 749389476

We can implement these computations in Numpy as

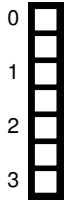
```
s = np.exp(x @ y) / (np.exp(x @ y) - 1)
```

```
d_x = d_out * s * y
```

```
d_y = d_out * s * x
```

<https://tutorcs.com>

Problem 7 (Version A) (3 credits)



程序代写代做 CS编程辅导

The rules we know are:

1. $k(\mathbf{x}_1, \mathbf{x}_2) = k_1(\dots)$
2. $k(\mathbf{x}_1, \mathbf{x}_2) = c \cdot \dots$
3. $k(\mathbf{x}_1, \mathbf{x}_2) = k_1(\dots)$
4. $k(\mathbf{x}_1, \mathbf{x}_2) = k_3(\dots)$ is a kernel k_3 on $\mathcal{X}' \subseteq \mathbb{R}^M$ and $\phi : \mathcal{X} \rightarrow \mathcal{X}'$
5. $k(\mathbf{x}_1, \mathbf{x}_2) = \mathbf{x}_1^\top \dots$ symmetric and positive semidefinite
6. $k(\mathbf{x}_1, \mathbf{x}_2) = \exp(k_6(\mathbf{x}_1, \mathbf{x}_2))$



$k(\mathbf{x}_1, \mathbf{x}_2) = \sigma^2 \left(-\frac{1}{2}(\mathbf{x}_1 - \mathbf{x}_2)^\top \Sigma^{-1}(\mathbf{x}_1 - \mathbf{x}_2) \right)$ is a kernel by rule (2), iff $\exp \left(-\frac{1}{2}(\mathbf{x}_1 - \mathbf{x}_2)^\top \Sigma^{-1}(\mathbf{x}_1 - \mathbf{x}_2) \right)$ is. Now let us rearrange the given equation:

$$-\frac{1}{2}(\mathbf{x}_1 - \mathbf{x}_2)^\top \Sigma^{-1}(\mathbf{x}_1 - \mathbf{x}_2) = -\frac{1}{2}[\mathbf{x}_1^\top \Sigma^{-1} \mathbf{x}_1 - \mathbf{x}_1^\top \Sigma^{-1} \mathbf{x}_2 - \mathbf{x}_2^\top \Sigma^{-1} \mathbf{x}_1 + \mathbf{x}_2^\top \Sigma^{-1} \mathbf{x}_2]$$

Assignment Project Exam Help

$$\Rightarrow \exp \left(-\frac{1}{2}(\mathbf{x}_1 - \mathbf{x}_2)^\top \Sigma^{-1}(\mathbf{x}_1 - \mathbf{x}_2) \right) = \exp \left(-\frac{1}{2} \mathbf{x}_1^\top \Sigma^{-1} \mathbf{x}_1 \right) \exp \left(\mathbf{x}_1^\top \Sigma^{-1} \mathbf{x}_2 \right) \exp \left(-\frac{1}{2} \mathbf{x}_2^\top \Sigma^{-1} \mathbf{x}_2 \right)$$

Email: tutorcs@163.com

$\exp \left(-\frac{1}{2} \mathbf{x}_1^\top \Sigma^{-1} \mathbf{x}_1 \right) \exp \left(-\frac{1}{2} \mathbf{x}_2^\top \Sigma^{-1} \mathbf{x}_2 \right)$ is a kernel by rule (4) with $\phi(\mathbf{x}) = \exp \left(-\frac{1}{2} \mathbf{x}^\top \Sigma^{-1} \mathbf{x} \right)$.

$\exp(\mathbf{x}_1^\top \Sigma^{-1} \mathbf{x}_2)$ is a kernel by rules (5) and rule (6) if Σ^{-1} is PSD.

We know that $\Sigma \in \mathbb{R}^{D \times D}$ is invertible and positive semi-definite. We define $\mathbf{a} = \Sigma \mathbf{b}$ and write:

$$\mathbf{a}^\top \Sigma^{-1} \mathbf{a} = \mathbf{b}^\top \Sigma^{-1} \Sigma \mathbf{b} = \mathbf{b}^\top \mathbf{b}$$

QQ: 749389476

Since Σ is PSD (i.e. $\mathbf{b}^\top \Sigma \mathbf{b} > 0, \forall \mathbf{b} \in \mathbb{R}^d \setminus \{0\}$), then so is Σ^{-1} (i.e. $\mathbf{a}^\top \Sigma^{-1} \mathbf{a} > 0, \forall \mathbf{a} \in \mathbb{R}^d \setminus \{0\}$).

<https://tutorcs.com>

Problem 7 (Version B) (3 credits)

程序代写代做 CS编程辅导

	0
	1
	2
	3

The rules we know are:

1. $k(\mathbf{x}_1, \mathbf{x}_2) = k_1(\mathbf{x}_1, \mathbf{x}_2) + k_2(\mathbf{x}_1, \mathbf{x}_2)$
2. $k(\mathbf{x}_1, \mathbf{x}_2) = c \cdot k_1(\mathbf{x}_1, \mathbf{x}_2)$, $c > 0$
3. $k(\mathbf{x}_1, \mathbf{x}_2) = k_1(\mathbf{x}_1, \mathbf{x}_2) \cdot k_2(\mathbf{x}_1, \mathbf{x}_2)$
4. $k(\mathbf{x}_1, \mathbf{x}_2) = k_3(\phi(\mathbf{x}_1), \phi(\mathbf{x}_2))$ where $\mathcal{X}' \subseteq \mathbb{R}^M$ and $\phi: \mathcal{X} \rightarrow \mathcal{X}'$
5. $k(\mathbf{x}_1, \mathbf{x}_2) = \mathbf{x}_1^T \mathbf{A} \mathbf{x}_2$, with \mathbf{A} symmetric and positive semidefinite
6. $k(\mathbf{x}_1, \mathbf{x}_2) = \exp(k_6(\mathbf{x}_1, \mathbf{x}_2))$



$k(\mathbf{x}_1, \mathbf{x}_2) = \sigma^2 \left(-\frac{1}{2}(\mathbf{x}_1 - \mathbf{x}_2)^T \Sigma^{-1}(\mathbf{x}_1 - \mathbf{x}_2) \right)$ is a kernel by rule (2), iff $\exp \left(-\frac{1}{2}(\mathbf{x}_1 - \mathbf{x}_2)^T \Sigma^{-1}(\mathbf{x}_1 - \mathbf{x}_2) \right)$ is. Now let us rearrange the given equation:

$$-\frac{1}{2}(\mathbf{x}_1 - \mathbf{x}_2)^T \Sigma^{-1}(\mathbf{x}_1 - \mathbf{x}_2) = -\frac{1}{2}[\mathbf{x}_1^T \Sigma^{-1} \mathbf{x}_1 - \mathbf{x}_1^T \Sigma^{-1} \mathbf{x}_2 - \mathbf{x}_2^T \Sigma^{-1} \mathbf{x}_1 + \mathbf{x}_2^T \Sigma^{-1} \mathbf{x}_2]$$

Assignment Project Exam Help

$$\Rightarrow \exp \left(-\frac{1}{2}(\mathbf{x}_1 - \mathbf{x}_2)^T \Sigma^{-1}(\mathbf{x}_1 - \mathbf{x}_2) \right) = \exp \left(-\frac{1}{2} \mathbf{x}_1^T \Sigma^{-1} \mathbf{x}_1 \right) \exp \left(\mathbf{x}_1^T \Sigma^{-1} \mathbf{x}_2 \right) \exp \left(-\frac{1}{2} \mathbf{x}_2^T \Sigma^{-1} \mathbf{x}_2 \right)$$

Email: tutorcs@163.com

$\exp \left(-\frac{1}{2} \mathbf{x}_1^T \Sigma^{-1} \mathbf{x}_1 \right) \exp \left(-\frac{1}{2} \mathbf{x}_2^T \Sigma^{-1} \mathbf{x}_2 \right)$ is a kernel by rule (4) with $\phi(\mathbf{x}) = \exp \left(-\frac{1}{2} \mathbf{x}^T \Sigma^{-1} \mathbf{x} \right)$.

$\exp(\mathbf{x}_1^T \Sigma^{-1} \mathbf{x}_2)$ is a kernel by rules (5) and rule (6) if Σ^{-1} is PSD.

We know that $\Sigma \in \mathbb{R}^{D \times D}$ is invertible and positive semi-definite. We define $\mathbf{a} = \Sigma \mathbf{b}$ and write:

$$\mathbf{a}^T \Sigma^{-1} \mathbf{a} = \mathbf{b}^T \Sigma^T \Sigma^{-1} \Sigma \mathbf{b} = \mathbf{b}^T \Sigma \mathbf{b}$$

QQ: 749389476

Since Σ is PSD (i.e. $\mathbf{b}^T \Sigma \mathbf{b} > 0, \forall \mathbf{b} \in \mathbb{R}^d \setminus \{0\}$), then so is Σ^{-1} (i.e. $\mathbf{a}^T \Sigma^{-1} \mathbf{a} > 0, \forall \mathbf{a} \in \mathbb{R}^d \setminus \{0\}$).

https://tutorcs.com

Problem 8 (Version A) (4 credits)

0
1
2

a)

程序代写代做 CS编程辅导

The negative log-likelihood (NLL) is



$$\begin{aligned} \ell(\mathbf{a}, \mathbf{b}) &= - \sum_{i=1}^N \sum_{j=1}^D \log p(X_{ij} | \mathbf{a}, \mathbf{b}) \\ &= \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^D (X_{ij} - a_i b_j)^2 \\ &= \frac{1}{2} \|\mathbf{X} - \mathbf{a}\mathbf{b}^T\|_F^2 \end{aligned}$$

We see that minimizing the NLL is equivalent to minimizing the Frobenius norm of the difference between \mathbf{X} and the rank-1 matrix $\mathbf{a}\mathbf{b}^T$.

In other words, we are looking for the optimal rank-1 approximation of the matrix \mathbf{X} . We know from the lecture that this can be done using the left and right singular vectors of \mathbf{X} corresponding to the largest singular value of \mathbf{X} . That is, we can set $\mathbf{a} = \sqrt{\sigma_1} \mathbf{u}_1$ and $\mathbf{b} = \sqrt{\sigma_1} \mathbf{v}_1$.

WeChat: cstutores

Assignment Project Exam Help

Email: tutorcs@163.com

0
1
2

b)

QQ: 749389476

No, the solution is not unique. If \mathbf{a}^* and \mathbf{b}^* minimize the negative log-likelihood, then so do $\frac{1}{c} \mathbf{a}^*$ and $c \mathbf{b}^*$ for any scalar $c \in \mathbb{R}$.

https://tutorcs.com

Problem 9 (Version A) (2 credits)

程序代写代做 CS编程辅导

(1) and (5) are correct for $\sigma = 2$ and $\sigma = 5$, respectively.

1. First column is correct.
2. Second column shows similarity.
3. Third column misses the lower left cluster and it is located at the center instead (2.75, 3.5).
4. Fourth column shows a
5. Upper row $\sigma = 2$, lower row $\sigma = 5$.

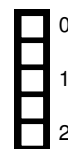
WeChat: cstutorcs

Assignment Project Exam Help

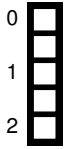
Email: tutorcs@163.com

QQ: 749389476

<https://tutorcs.com>



Problem 9 (Version B) (2 credits)



程序代写代做 CS编程辅导

(3) and (7) are correct for $\sigma = 5$ and $\sigma = 2$, respectively.

1. First column: ...ad of similarity.
2. Second column: ...ce in the lower left cluster and it is located at the center instead (2.75, 3.5).
3. Third column: ...
4. Fourth column: ...cal matrix.
5. Upper row $\sigma = 5$, lower row $\sigma = 2$.

WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

<https://tutorcs.com>

Problem 9 (Version C) (2 credits)

程序代写代做 CS编程辅导

(4) and (8) are correct for $\sigma = 5$ and $\sigma = 2$, respectively.

1. First column misses one point from the upper left cluster and it is located at the center instead of (2.75, 3.5).
2. Second column shows two clusters.
3. Third column shows a cluster with high similarity.
4. Fourth column is correct.
5. Upper row $\sigma = 5$, lower row $\sigma = 2$.

WeChat: cstutorcs

Assignment Project Exam Help

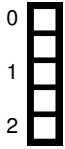
Email: tutorcs@163.com

QQ: 749389476

<https://tutorcs.com>



Problem 9 (Version D) (2 credits)



程序代写代做 CS编程辅导

(2) and (6) are correct for $\sigma = 2$ and $\sigma = 5$, respectively.

1. First column in the lower left cluster and it is located at the center instead (2.75, 3.5).
2. Second column
3. Third column al matrix.
4. Fourth column lead of similarity.
5. Upper row $\sigma = 5$, lower row $\sigma = 2$.

WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476


<https://tutorcs.com>

Problem 10 (All versions) (4 credits)

a)

程序代写代做 CS编程辅导

At the decision boundary we have



$$\begin{aligned}
 \|\mathbf{x} - \mu_1\|_2 &= \|\mathbf{x} - \mu_2\|_2 \\
 \|\mathbf{x} - \mu_1\|_2^2 &= \|\mathbf{x} - \mu_2\|_2^2 \\
 (\mathbf{x} - \mu_1)^T(\mathbf{x} - \mu_1) &= (\mathbf{x} - \mu_2)^T(\mathbf{x} - \mu_2) \\
 \mathbf{x}^T + \mu_1^T &= \mathbf{x}^T - 2\mu_2^T\mathbf{x} + \mu_2^T \\
 2\mu_1^T\mathbf{x} &= 0.
 \end{aligned} \tag{27.1}$$

We can thus define the decision plane $\mathbf{x}_0 + \mathbf{w}^T\mathbf{x} = 0$ with $\mathbf{x}_0 = \mu_1^2 - \mu_2^2$ and $\mathbf{w} = 2\mu_2 - 2\mu_1$.

WeChat: cstutorcs

Assignment Project Exam Help

b)

[This is a full proof, not just the justification students need to give.]
At the decision boundary we have

Email: tutormcs@163.com

QQ: 749389476

$$\Leftrightarrow \log \pi_1 - \frac{1}{2} \log((2\pi)^d |\Sigma_1|) - \frac{1}{2}(\mathbf{x}_i - \mu_1)^T \Sigma_1^{-1}(\mathbf{x}_i - \mu_1) = \log \pi_2 - \frac{1}{2} \log((2\pi)^d |\Sigma_2|) - \frac{1}{2}(\mathbf{x}_i - \mu_2)^T \Sigma_2^{-1}(\mathbf{x}_i - \mu_2), \tag{27.2}$$

<https://tutorcs.com>

where we have used monotonicity of the log function. To obtain a linear decision boundary the quadratic term must drop out, i.e.

$$\mathbf{x}^T \Sigma_1^{-1} \mathbf{x} = \mathbf{x}^T \Sigma_2^{-1} \mathbf{x} \quad \forall \mathbf{x}. \tag{27.3}$$

Since Σ is a covariance matrix and invertible, it must be positive definite. The same holds for its inverse, which must thus have a decomposition $\Sigma^{-1} = \mathbf{V}^T \mathbf{V}$, where \mathbf{V} is invertible as well. Using this, we have

$$\begin{aligned}
 \mathbf{x}^T \mathbf{V}_1^T \mathbf{V}_1 \mathbf{x} &= \mathbf{x}^T \mathbf{V}_2^T \mathbf{V}_2 \mathbf{x} \quad \forall \mathbf{x} \\
 \Leftrightarrow \|\mathbf{V}_1 \mathbf{x}\|_2 &= \|\mathbf{V}_2 \mathbf{x}\|_2 \quad \forall \mathbf{x} \\
 \Leftrightarrow \|\mathbf{y}\|_2 &= \|\mathbf{V}_2 \mathbf{V}_1^{-1} \mathbf{y}\|_2 \quad \forall \mathbf{y},
 \end{aligned} \tag{27.4}$$

where we have substituted $\mathbf{y} = \mathbf{V}_1 \mathbf{x}$. We can do this due to the full rank (invertibility) of \mathbf{V}_1 . $\mathbf{U} = \mathbf{V}_2 \mathbf{V}_1^{-1}$ must therefore be an isometry with respect to the L_2 norm, i.e. a unitary matrix with $\mathbf{U}^T \mathbf{U} = \mathbf{I}$. Due to invertibility $\mathbf{U} \mathbf{V}_1 = \mathbf{V}_2$ and, finally,

$$\Sigma_2^{-1} = \mathbf{V}_2^T \mathbf{V}_2 = \mathbf{V}_1^T \mathbf{U}^T \mathbf{U} \mathbf{V}_1 = \mathbf{V}_1^T \mathbf{V}_1 = \Sigma_1^{-1}. \tag{27.5}$$

The decision boundary is therefore linear if and only if $\Sigma_1 = \Sigma_2$, which is precisely linear discriminant analysis (LDA).

Problem 11 (Version A) (4 credits)

0
1

a)

程序代写代做 CS编程辅导

No. We cannot conclude anything because "Fairness through Unawareness" does not work. There can be many highly correlated features that are proxies of the sensitive attribute.



0
1
2

b)

WeChat: cstutorcs

First we compute the predictions R to obtain:

ID	1	2	3	4	5	6	7
X	0.5	-1.0	-0.5	2.0	0.5	1.5	0
A	a	b	b	a	b	a	b
R	1	1	1	0	0	0	1
Y	1	1	0	0	0	0	1

Assignment Project Exam Help

Email: tutorcs@163.com

We see that 1/3 instances in group a have $R = 1$ vs. 3/4 instances in group b. *Independence* is not satisfied.

For group a we have $TP=1/1$ and $FP=0/2$.

For group b we have $TP=2/2$ and $FP=0/2$.

QQ: 749389476

Since only TP matches for both groups, *Equality of Opportunity* is satisfied and *Separation* is not satisfied.

<https://tutorcs.com>

0
1

c)

We modify the instance with ID 1, changing the non-sensitive feature from $X = 0.5$ to $X = 1.5$. Now the prediction changes from $R = 1$ to $R = 0$.

Now we have 0/3 instances with $R = 1$ within its group, vs. 3/4 in the other group so *Independence* is still not satisfied. The TP rate has changed from 1/1 to 0/1 compared to 2/2 in the other group, which means that neither *Equality of Opportunity* nor *Separation* are satisfied.

Problem 11 (Version B) (4 credits)

a)

程序代写代做 CS编程辅导

No. We cannot conclude anything because "Fairness through Unawareness" does not work. There can be many highly correlated features that are proxies of the sensitive attribute.



0
1

b)

WeChat: cstutorcs

First we compute the predictions R to obtain:

ID	1	2	3	4	5	6	7
X	0.5	-1.0	-0.5	2.0	0.5	-0.5	0.1
A	b	a	a	b	a	b	a
R	1	1	1	0	0	0	1
Y	1	1	0	0	0	0	1

Assignment Project Exam Help

Email: tutorcs@163.com

We see that 3/4 instances in group a have $R = 1$ vs. 1/3 instances in group b . *Independence* is not satisfied.

For group a we have $TP=2/2$ and $FP=1/2$.

For group b we have $TP=1/1$ and $FP=0/2$.

QQ: 749389476

Since only TP matches for both groups, *Equality of Opportunity* is satisfied and *Separation* is not satisfied.

<https://tutorcs.com>

0
1
2

c)

We modify the instance with ID 1, changing the non-sensitive feature from $X = 0.5$ to $X = 1.5$. Now the prediction changes from $R = 1$ to $R = 0$.

Now we have 0/3 instances with $R = 1$ within its group, vs. 3/4 in the other group so *Independence* is still not satisfied. The TP rate has changed from 1/1 to 0/1 compared to 2/2 in the other group, which means that neither *Equality of Opportunity* nor *Separation* are satisfied.

0
1

Problem 11 (Version C) (4 credits)

0
1

a)

程序代写代做 CS编程辅导

No. We cannot conclude anything because "Fairness through Unawareness" does not work. There can be many highly correlated features that are proxies of the sensitive attribute.



0
1
2

b)

WeChat: cstutorcs

First we compute the predictions R to obtain:

ID	1	2	3	4	5	6	7
X	0.5	-1.0	-0.5	2.0	0.5	1.5	0
A	a	b	b	a	b	a	b
R	1	1	1	0	0	0	1
Y	1	1	0	0	0	0	1

Assignment Project Exam Help

Email: tutorcs@163.com

We see that 1/3 instances in group a have $R = 1$ vs. 3/4 instances in group b. *Independence* is not satisfied.

For group a we have $TP=1/1$ and $FP=0/2$.

For group b we have $TP=2/2$ and $FP=0/2$.

QQ: 749389476

Since only TP matches for both groups, *Equality of Opportunity* is satisfied and *Separation* is not satisfied.

<https://tutorcs.com>

0
1

c)

We modify the instance with ID 1, changing the non-sensitive feature from $X = 0.5$ to $X = 1.5$. Now the prediction changes from $R = 1$ to $R = 0$.

Now we have 0/3 instances with $R = 1$ within its group, vs. 3/4 in the other group so *Independence* is still not satisfied. The TP rate has changed from 1/1 to 0/1 compared to 2/2 in the other group, which means that neither *Equality of Opportunity* nor *Separation* are satisfied.

Problem 11 (Version D) (4 credits)

a)

程序代写代做 CS编程辅导

No. We cannot conclude anything because "Fairness through Unawareness" does not work. There can be many highly correlated features that are proxies of the sensitive attribute.



0
1

b)

WeChat: cstutorcs

First we compute the predictions R to obtain:

ID	1	2	3	4	5	6	7
X	0.5	-1.0	-0.5	2.0	0.5	-0.5	0.1
A	b	a	a	b	a	b	a
R	1	1	1	0	0	0	1
Y	1	1	0	0	0	0	1

Assignment Project Exam Help

Email: tutorcs@163.com

We see that 3/4 instances in group a have $R = 1$ vs. 1/3 instances in group b . *Independence* is not satisfied.

For group a we have $TP=2/2$ and $FP=1/2$.

For group b we have $TP=1/1$ and $FP=0/2$.

QQ: 749389476

Since only TP matches for both groups, *Equality of Opportunity* is satisfied and *Separation* is not satisfied.

<https://tutorcs.com>

0
1
2

c)

We modify the instance with ID 1, changing the non-sensitive feature from $X = 0.5$ to $X = 1.5$. Now the prediction changes from $R = 1$ to $R = 0$.

Now we have 0/3 instances with $R = 1$ within its group, vs. 3/4 in the other group so *Independence* is still not satisfied. The TP rate has changed from 1/1 to 0/1 compared to 2/2 in the other group, which means that neither *Equality of Opportunity* nor *Separation* are satisfied.

0
1

Additional space for solutions—clearly mark the (sub)problem your answers are related to and strike out invalid solutions.

程序代写代做 CS编程辅导



WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

<https://tutorcs.com>