

Passed (42/52)

Submitted at 16 Aug 2024 10:23

Contest
AIO2024 Contest

Exam
AIO Module 2 Examination (720 mins)



Description

Biến ngẫu nhiên liên tục là gì?

- A. Một hàm ánh xạ từ không gian mẫu đến tập số thực với miền giá trị hữu hạn
- B. Một hàm ánh xạ từ không gian mẫu đến tập số thực với miền giá trị không đếm được
- C. Một hàm ánh xạ từ tập số thực đến không gian mẫu
- D. Một hàm ánh xạ từ tập số nguyên đến tập số thực

✓ M2AIO2 ^

Description

Hàm phân phối xác suất (Probability Distribution Function) của một biến ngẫu nhiên X được định nghĩa là gì?

A.
$$P(X=x)$$

B.
$$P(X \leq x)$$

$$\mathsf{C}.\,E[X]$$

D.
$$Var(X)$$

Α

В

С

D



Description

Công thức tính giá trị kỳ vọng (Expected value) của một biến ngẫu nhiên rời rạc X là gì?

A.
$$E[X]=\Sigma_i^n x \times P(X=x)$$
 B. $E[X]=\Sigma_i^n (x-\mu)^2 \times P(X=x)$ C. $E[X]=\sqrt{Var(X)}$ D. $E[X]=\Sigma_i^n P(X=x)$

A 🔽

В

С

D

✓ M2AIO4 ^

Description

Công thức tính phương sai (Variance) của một biến ngẫu nhiên X là gì?

A.
$$Var(X) = E[X] - \mu^2$$
 B. $Var(X) = E[X^2] - E[X]^2$ C. $Var(X) = E[X]^2 - E[X^2]$ D. $Var(X) = \sqrt{E[X^2] - E[X]^2}$

Α

В

С

D

✓ M2AIO5 ^

Description

Công thức tính độ lệch chuẩn (Standard Deviation) của một biến ngẫu nhiên X là gì?

A.
$$\sigma(X)=\sqrt{Var(X)^2}$$
 B. $\sigma(X)=Var(X)$ C. $\sigma(X)=\sqrt{Var(X)}$ D. $\sigma(X)=E[(X-\mu)^2]$

Α

В

C 🔽

D

✓ M2AIO6 ^

Description

Công thức tính hiệp phương sai (Covariance) giữa hai biến ngẫu nhiên X và Y là gì?

A.
$$\operatorname{Cov}(X,Y) = E[XY] - E[X]E[Y]$$

B.
$$\operatorname{Cov}(X,Y) = E[(X-\mu Y)(Y-\mu Y)]$$

$$\operatorname{C.Cov}(X,Y) = E[X]E[Y] - E[XY]$$

$$\operatorname{D.Cov}(X,Y) = \sigma X * \sigma Y$$

A 🔽

В

С

D

✓ M2AIO7 ^

Description

Trong một thí nghiệm tung xúc xắc công bằng, X là số chấm xuất hiện. Tính ${\cal E}[X^2]$.

A. 15.17

B. 3.50

C. 12.25

D. 2.92

A 🔽			
В			
С			
D			

✓ M2AIO8 ^

Description

Cho hai biến ngẫu nhiên $X=\{1,2,3,4\}$ và $Y=\{2,3,4,3\}$. Tính hiệp phương sai Cov(X,Y). A. 0.5 B. 0.6 C. 0.7 D. 0.8

A ☑
B
C

D

× M2AIO9 ^

Description

Trong một trò chơi, bạn sẽ mất 1\$ nếu sinh viên kế tiếp tham gia vào lớp AlO2025 thích toán, ngược lại bạn nhận được 2\$ nếu sinh viên ấy không thích toán. Xác suất một sinh viên thích toán là 0.3. Tính giá trị kỳ vọng

- A. -1.1\$
- B. 1,1\$ X (Select a wrong answer)
- C. 2\$ X (Don't select the correct answer)
- D. 4\$



Description

Hệ số tương quan Pearson có phạm vi giá trị nào?

- A. [0, 1]
- B. [-1, 1] 🔽
- C. $[-\infty, +\infty]$
- D. [0, +∞]



Description

Trong thống kê, hãy tính giá trị "Degrees of Freedom" (bậc tự do) cho một mẫu dữ liệu có số lượng mẫu N=30?

- A. 29 🛂
- B. 30
- C. 31
- D. 1



Cho hai biến ngẫu nhiên X và Y có hệ số tương quan là 0.6. Biết rằng Var(X) = 4 và Var(Y) = 9. Tính Cov(X, Y) và giải thích ý nghĩa của kết quả này trong việc mô tả mối quan hệ giữa X và Y

- A. Cov(X, Y) = 0.36, X và Y có mối quan hệ tuyến tính khá yếu và cùng chiều.
- B. Cov(X, Y) = 0.36, X và Y có mối quan hệ tuyến tính khá yếu và ngược chiều.
- C. Cov(X, Y) = 3.6, X và Y có mối quan hệ tuyến tính khá mạnh và ngược chiều.
- D. Cov(X, Y) = 3.6, X và Y có mối quan hệ tuyến tính khá mạnh và cùng chiều.

 ✓

✓ M2AIO14 ^

Description

Bạn đang phân tích mối quan hệ giữa nhiệt độ hàng ngày và doanh số bán kem trong một thành phố. Bạn có dữ liệu về nhiệt độ (°C) và doanh số bán kem (\$) trong 30 ngày. Hệ số tương quan Pearson giữa hai biến này là 0.75. Nếu bạn chuyển đổi nhiệt độ từ °C sang °F và doanh số bán kem từ \$ sang $\mathfrak E$, hệ số tương quan sẽ thay đổi như thế nào? Biết °F= ° $C\times9/5+32$ và $1\$=0.91\epsilon$.

- A. Hệ số không đổi ☑
- B. Hệ số tăng lên
- C. Hệ số giảm xuống
- D. Không thể tính hệ số



A.
$$P(A)=rac{ ext{Số kết quả thuận lợi}}{ ext{Tổng số kết quả có thể}}$$
B. $P(A)=rac{ ext{Số kết quả không thuận lợi}}{ ext{Tổng số kết quả có thể}}$
C. $P(A)=rac{ ext{Tổng số kết quả có thể}}{ ext{Số kết quả thuận lợi}}$
D. $P(A)=1-rac{ ext{Số kết quả thuận lợi}}{ ext{Tổng số kết quả thuận lợi}}$

A 🔽

В

C

D



Công thức tính xác suất có điều kiện P(A | B) là gì?

A.
$$P(A \mid B) = \frac{P(A \cap B)}{P(A)}$$

B.
$$P(A \mid B) = \frac{P(A \cap B)}{P(B)}$$

C.
$$P(A \mid B) = \frac{P(A)}{P(B)}$$

A.
$$P(A \mid B) = \frac{P(A \cap B)}{P(A)}$$

B. $P(A \mid B) = \frac{P(A \cap B)}{P(B)}$
C. $P(A \mid B) = \frac{P(A)}{P(B)}$
D. $P(A \mid B) = \frac{P(B)}{P(A \cap B)}$

Α

В

C

D



Description

Công thức của quy tắc nhân xác suất (Multiplication Rule) cho hai sự kiện phụ thuộc A và B là gì?

A.
$$P(A\cap B)=P(A)+P(B)$$

B. $P(A\cap B)=P(A)\times P(B)$
C. $P(A\cap B)=P(A)\times P(B\mid A)$
D. $P(A\cap B)=\frac{P(A)}{P(B)}$

Α

В

C 🔽

D

✓ M2AIO18 ^

Description

Công thức nào sau đây biểu diễn định lý xác suất toàn phần?

$$\begin{aligned} &\text{A. } P(H) = \sum P(A_i) \times P(H \mid A_i) \\ &\text{B. } P(H) = P(A_1) \times P(H | A_1) + P(A_2) \times P(H | A_2) \\ &\text{C. } P(H) = \frac{P(A_1 | H) \times P(H)}{P(A_1)} \\ &\text{D. } P(H) = P(A_1) + P(A_2) + \dots + P(A_n) \end{aligned}$$

A 🛂

В

С

D

✓ M2AIO19 ^

Description

Trong một hộp có 10 quả bóng, trong đó có 4 quả màu đỏ và 6 quả màu xanh. Nếu lấy ngẫu nhiên lần lượt 2 quả bóng (mỗi lần 1 quả) không hoàn lại, xác suất để cả hai quả đều màu đỏ là bao nhiêu?

A. 2/15 🛂

B. 1/2

C. 3/25

D. 4/24

✓ M2AIO20 ^

Một công ty sản xuất 1000 sản phẩm mỗi ngày. Biết rằng 2% sản phẩm bị lỗi. Nếu chọn ngẫu nhiên một sản phẩm, xác suất để sản phẩm đó không bị lỗi là bao nhiêu?

- A. 0.02
- B. 0.98
- C. 0.2
- D. 0.8

✓ M2AIO21 ^

Description

Trong một cuộc khảo sát, 30% người được hỏi thích pizza, 40% thích burger, và 15% thích cả hai. Xác suất để một người được chọn ngẫu nhiên không thích cả pizza lẫn burger là bao nhiêu?

- A. 0.85
- B. 0.30
- C. 0.58
- D. 0.45

✓ M2AIO22 ^

Description

Một hệ thống máy tính gồm 3 bộ phận độc lập. Xác suất hoạt động của mỗi bộ phận lần lượt là 0.9, 0.95 và 0.85. Hệ thống hoạt động khi ít nhất 2 trong 3 bộ phận hoạt động. Tính xác suất để hệ thống hoạt động.

- A. 0.892
- B. 0.974 <
- C. 0.247
- D. 0.727

Description

Công thức Bayes được viết như thế nào để hợp lý cho bài toán với X là các đặc trưng và Y là lớp?

- A. P(Y|X) = P(X|Y) * P(Y) / P(X)
- B. P(Y|X) = P(X|Y) * P(X) / P(Y)
- C. P(Y|X) = P(X|Y) / (P(X) * P(Y))
- D. P(Y|X) = P(X|Y) * P(Y) * P(X)

✓ M2AIO24 ^

Description

Trong phân loại Naive Bayes, giả định nào được đưa ra về các đặc trưng (features)?

- A. Các đặc trưng phải phụ thuộc lẫn nhau
- B. Các đặc trưng phải độc lập có điều kiện với nhau khi biết lớp
- C. Các đặc trưng phải tuân theo phân phối chuẩn
- D. Các đặc trưng phải là biến liên tục

✓ M2AIO25 ^

Description

Trong phân phối chuẩn (Gaussian distribution), tham số μ và σ lần lượt đại diện cho:

- A. Trung bình và phương sai
- B. Trung vị và độ lệch chuẩn
- C. Trung bình và độ lệch chuẩn
- D. Mode và phương sai

✓ M2AIO26 ^

Cho một tập dữ liệu với 2 lớp và 2 đặc trưng liên tục. Nếu sử dụng Gaussian Naive Bayes để phân loại, cần ước lượng bao nhiêu tham số cho các phân phối trong mô hình?

- A. 4
- B. 6
- C. 8
- D. 10

✓ M2AIO27 ^

Description

Khi áp dụng mô hình Gaussian Naive Bayes cho dữ liệu liên tục, tại sao chúng ta lại sử dụng hàm mật độ xác suất thay vì xác suất trực tiếp cho các giá trị của biến?

- A. Vì dữ liệu liên tục có vô số giá trị có thể ☑
- B. Vì dữ liệu liên tục luôn tuân theo phân phối chuẩn
- C. Vì dễ tính toán hơn
- D. Vì cho kết quả chính xác hơn trong mọi trường hợp

× M2AIO28 ^

Description

Cho một tập dữ liệu phân loại hoa Iris với 2 lớp và 2 đặc trưng: chiều dài (Length) và chiều rộng (Width) của cánh hoa. Các thông số thống kê như sau:

Classes	μ_Length	σ_Length	μ_Width	σ_Width	P(Class)
Class 0	4.8	0.2	3.3	0.3	0.5
Class 1	6.2	0.4	2.9	0.3	0.5

Với một mẫu mới có Length = 5.5 và Width = 3.0, hãy tính xác suất P(Class 1 | Length=5.5, Width=3.0) sử dụng Gaussian Naive Bayes.

- A. 0.1573 X (Select a wrong answer)
- B. 0.1753
- C. 0.1537
- D. 0.1357 X (Don't select the correct answer)

✓ M2AIO29 ^

Description

Một bệnh hiếm gặp có tỷ lệ mắc trong dân số là 1%. Một xét nghiệm chẩn đoán bệnh này có độ nhạy (sensitivity) là 95% (xác suất xét nghiệm dương tính khi người đó thực sự mắc bệnh) và độ đặc hiệu (specificity) là 90% (xác suất xét nghiệm âm tính khi người đó không mắc bệnh).

Nếu một người có kết quả xét nghiệm dương tính, tính xác suất người đó thực sự mắc bệnh? (Sử dụng Naive Bayes)

- A. 0.0856
- B. 0.0866
- C. 0.0876
- D. 0.0886

✓ M2AIO30 ^

Description

Một công ty bảo hiểm muốn dự đoán liệu một khách hàng có khả năng mua bảo hiểm xe hơi hay không dựa trên hai đặc điểm: Tuổi (Trẻ, Trung niên, Già) và Thu nhập (Thấp, Trung bình, Cao). Dữ liệu huấn luyện như sau:

Dựa vào dữ liệu huấn luyện, xác suất tiên nghiệm (prior) của việc "Có" mua bảo hiểm là bao nhiêu?

Tuổi	Thu nhập	Mua bảo hiểm
Trẻ	Thấp	Không
Trẻ	Trung bình	Có
Trung niên	Thấp	Có
Già	Trung bình	Có
Già	Cao	Không
Trung niên	Cao	Có
Trẻ	Cao	Không
Trung niên	Trung bình	Có
Già	Thấp	Có
Trẻ	Trung bình	Không

- A) 0.50
- B) 0.40
- C) 0.60 Z
- D) 0.80

✓ M2AIO31 ^

Description

Một công ty bảo hiểm muốn dự đoán liệu một khách hàng có khả năng mua bảo hiểm xe hơi hay không dựa trên hai đặc điểm: Tuổi (Trẻ, Trung niên, Già) và Thu nhập (Thấp, Trung bình, Cao). Dữ liệu huấn luyện như sau:

Dựa vào dữ liệu huấn luyện, Xác suất có điều kiện P(Tuổi = Trung niên | Mua bảo hiểm = Có) là

Tuổi	Thu nhập	Mua bảo hiểm
Trẻ	Thấp	Không
Trẻ	Trung bình	Có
Trung niên	Thấp	Có
Già	Trung bình	Có
Già	Cao	Không
Trung niên	Cao	Có
Trẻ	Cao	Không
Trung niên	Trung bình	Có
Già	Thấp	Có
Trẻ	Trung bình	Không

- A) 0.20
- B) 0.50 🛂
- C) 0.80
- D) 1.00

X M2AIO32 ^

Description

Tuổi	Thu nhập	Mua bảo hiểm
Trẻ	Thấp	Không
Trẻ	Trung bình	Có
Trung niên	Thấp	Có
Già	Trung bình	Có
Già	Cao	Không
Trung niên	Cao	Có
Trẻ	Cao	Không
Trung niên	Trung bình	Có
Già	Thấp	Có
Trẻ	Trung bình	Không

Xác suất có điều kiện P(Thu nhập = Cao | Mua bảo hiểm = Có) là

A) 0.167 X (Don't select the correct answer)

B) 0.20

C) 0.33 × (Select a wrong answer)

D) 0.83

X M2AIO33 ^

Description

Tuổi	Thu nhập	Mua bảo hiểm
Trẻ	Thấp	Không
Trẻ	Trung bình	Có
Trung niên	Thấp	Có
Già	Trung bình	Có
Già	Cao	Không
Trung niên	Cao	Có
Trẻ	Cao	Không
Trung niên	Trung bình	Có
Già	Thấp	Có
Trẻ	Trung bình	Không

Đối với một khách hàng "Trung niên" với Thu nhập "Cao", xác suất mua bảo hiểm là:

- A) P(Có|Trung niên, Cao) = 0.70, P(Không|Trung niên, Cao) = 0.30
- C) P(Có|Trung niên, Cao) = 0.42, P(Không|Trung niên, Cao) = 0.58
- D) P(Có|Trung niên, Cao) = 1.00, P(Không|Trung niên, Cao) = 0.00 ★ (Select a wrong answer)

✓ M2AIO34 ^

Description

Tuổi	Thu nhập	Mua bảo hiểm
Trẻ	Thấp	Không
Trẻ	Trung bình	Có
Trung niên	Thấp	Có
Già	Trung bình	Có
Già	Cao	Không
Trung niên	Cao	Có
Trẻ	Cao	Không
Trung niên	Trung bình	Có
Già	Thấp	Có
Trẻ	Trung bình	Không

Trong trường hợp này, giả định "naive" của Naive Bayes có thể dẫn đến sai lệch nào?

- A) Bỏ qua mối quan hệ tiềm ẩn giữa tuổi và thu nhập
- B) Đánh giá không chính xác tầm quan trọng của tuổi
- C) Bỏ qua mối quan hệ tiềm ẩn giữa mua bảo hiểm và thu nhập, tuổi
- D) Giả định sai về sự phân phối của các đặc trưng

× M2AIO35 ^

Description

Tuổi	Thu nhập	Mua bảo hiểm
Trẻ	Thấp	Không
Trẻ	Trung bình	Có
Trung niên	Thấp	Có
Già	Trung bình	Có
Già	Cao	Không
Trung niên	Cao	Có
Trẻ	Cao	Không
Trung niên	Trung bình	Có
Già	Thấp	Có
Trẻ	Trung bình	Không

Đối với một khách hàng "Già" với Thu nhập "Cao", người ta có mua bảo hiểm hay không:

- A) Có
- B) Không 🛂
- C) 50/50 X (Don't select the correct answer)
- D) Không biết

✓ M2AIO36 ^

Description

Naive Bayes Classifier được gọi là "Naive" (Ngây thơ) vì:

- A) Nó giả định rằng các đặc trưng (features) độc lập có điều kiện 💆
- B) Nó giả định rằng các đặc trưng (features) hoàn toàn độc lập
- C) Nó chỉ xem xét xác suất có điều kiện đơn giản
- D) Nó bỏ qua mối quan hệ phức tạp giữa các đặc trưng

✓ M2AIO37 ^

Một nhà máy sản xuất bóng đèn LED muốn phân loại sản phẩm thành "Đạt" hoặc "Không đạt" dựa trên hai đặc điểm liên tục: Cường độ ánh sáng (lumen) và Tuổi thọ (giờ). Dưới đây là dữ liệu của 10 bóng đèn LED đã được kiểm tra:

Số thứ tự	Cường độ ánh sáng (lumen)	Tuổi thọ (giờ)	Phân loại
1	805	51000	Đạt
2	780	49500	Đạt
3	795	52000	Đạt
4	820	50500	Đạt
5	810	48000	Đạt
6	785	53000	Đạt
7	815	49000	Đạt
8	800	50000	Đạt
9	610	31000	Không đạt
10	590	28000	Không đạt
11	620	33000	Không đạt
12	580	29000	Không đạt
13	600	32000	Không đạt
14	595	27000	Không đạt
15	605	34000	Không đạt
16	585	31500	Không đạt

Dựa vào dữ liệu huấn luyện, xác suất tiên nghiệm (prior) của việc bóng đèn "Đạt" là bao nhiêu?

- A) 0.48
- B) 0.49
- C) 0.50 🛂
- D) 0.51

✓ M2AIO38 ^

Description

Một nhà máy sản xuất bóng đèn LED muốn phân loại sản phẩm thành "Đạt" hoặc "Không đạt" dựa trên hai đặc điểm liên tục: Cường độ ánh sáng (lumen) và Tuổi thọ (giờ). Dưới đây là dữ liệu của 10 bóng đèn LED đã được kiểm tra:

Lưu ý: Tính phương sai theo công thức Sample Variance (N-1)

$$s^2 = rac{1}{N-1} \sum_{i=1}^N (x_i - ar{x})^2$$

Số thứ tự	Cường độ ánh sáng (lumen)	Tuổi thọ (giờ)	Phân loại
1	805	51000	Đạt
2	780	49500	Đạt
3	795	52000	Đạt
4	820	50500	Đạt
5	810	48000	Đạt
6	785	53000	Đạt
7	815	49000	Đạt
8	800	50000	Đạt
9	610	31000	Không đạt
10	590	28000	Không đạt
11	620	33000	Không đạt
12	580	29000	Không đạt
13	600	32000	Không đạt
14	595	27000	Không đạt
15	605	34000	Không đạt
16	585	31500	Không đạt

Trung bình (µ) của cường độ ánh sáng cho bóng đèn "Đạt" là:

- A) 801.25 lumen <
- B) 802.50 lumen
- C) 803.75 lumen
- D) 805.00 lumen

X M2AIO39 ^

Một nhà máy sản xuất bóng đèn LED muốn phân loại sản phẩm thành "Đạt" hoặc "Không đạt" dựa trên hai đặc điểm liên tục: Cường độ ánh sáng (lumen) và Tuổi thọ (giờ). Dưới đây là dữ liệu của 10 bóng đèn LED đã được kiểm tra:

Lưu ý: Tính phương sai theo công thức Sample Variance (N-1)

$$s^2 = rac{1}{N-1} \sum_{i=1}^N (x_i - ar{x})^2$$

Số thứ tự	Cường độ ánh sáng (lumen)	Tuổi thọ (giờ)	Phân loại
1	805	51000	Đạt
2	780	49500	Đạt
3	795	52000	Đạt
4	820	50500	Đạt
5	810	48000	Đạt
6	785	53000	Đạt
7	815	49000	Đạt
8	800	50000	Đạt
9	610	31000	Không đạt
10	590	28000	Không đạt
11	620	33000	Không đạt
12	580	29000	Không đạt
13	600	32000	Không đạt
14	595	27000	Không đạt
15	605	34000	Không đạt
16	585	31500	Không đạt

Độ lệch chuẩn (σ) của tuổi thọ cho bóng đèn "Không đạt" là :

A) 2404.14 giờ X (Select a wrong answer)

B) 2463.12 giờ X (Don't select the correct answer)

C) 2304.42 giờ

D) 2043.04 giờ

X M2AIO40 ^

Description

Chú ý: Đây là câu thứ tự 39 (website), với mã câu hỏi là M2AIO40 tức thứ tự 40 trong file pdf.

Một nhà máy sản xuất bóng đèn LED muốn phân loại sản phẩm thành "Đạt" hoặc "Không đạt" dựa trên hai đặc điểm liên tục: Cường độ ánh sáng (lumen) và Tuổi thọ (giờ). Dưới đây là dữ liệu của 10 bóng đèn LED đã được kiểm tra:

Lưu ý: Tính phương sai theo công thức Sample Variance (N-1)

$$s^2 = rac{1}{N-1} \sum_{i=1}^N (x_i - ar{x})^2$$

Số thứ tự	Cường độ ánh sáng (lumen)	Tuổi thọ (giờ)	Phân loại
1	805	51000	Đạt
2	780	49500	Đạt
3	795	52000	Đạt
4	820	50500	Đạt
5	810	48000	Đạt
6	785	53000	Đạt
7	815	49000	Đạt
8	800	50000	Đạt
9	610	31000	Không đạt
10	590	28000	Không đạt
11	620	33000	Không đạt
12	580	29000	Không đạt
13	600	32000	Không đạt
14	595	27000	Không đạt
15	605	34000	Không đạt
16	585	31500	Không đạt

Xác suất không chuẩn hóa của một bóng đèn có cường độ ánh sáng 750 lumen và tuổi thọ 45000 giờ được phân loại là:

```
A) P(\text{Đạt}|750,45000) \approx 3.75 \times 10^{-11}, P(\text{Không đạt}|750,45000) \approx 1.71 \times 10^{-41} B) P(\text{Đạt}|750,45000) \approx 6.5 \times 10^{-7}, P(\text{Không đạt}|750,45000) \approx 5.1 \times 10^{-11} C) P(\text{Đạt}|750,45000) \approx 8.2 \times 10^{-10}, P(\text{Không đạt}|750,45000) \approx 2.6 \times 10^{-6} D) P(\text{Đạt}|750,45000) \approx 9.25 \times 10^{-11}, P(\text{Không đạt}|750,45000) \approx 7.81 \times 10^{-5}
```

A X (Don't select the correct answer)

B X (Select a wrong answer)

С

D



Description

Thuật toán Naive Bayes trong bài toán phân loại văn bản, giả định "Naive" là gì?

 $P("brown"|\ Ham).P("sugar"|\ Ham).P("pearl"|\ Ham).P("milk"|\ Ham).P("tea"|\ Ham)$

- A. Tất cả văn bản đều có xác suất như nhau để thuộc vào 1 lớp
- B. Độ dài của văn bản quyết định nó thuộc lớp nào
- C. Sự xuất hiện của một từ không ảnh hưởng đến sự xuất hiện của các từ khác
- D. Các văn bản của một lớp luôn chứa các từ khóa cụ thể

✓ M2AIO42 ^

Description

Trong phân loại văn bản sử dụng Naive Bayes, tại sao chúng ta thường sử dụng log likelihood thay vì xác suất thông thường?

- A. Để tăng độ chính xác của mô hình
- B. Để giảm thời gian tính toán
- C. Để tránh underflow khi nhân nhiều số nhỏ <a>
- D. Để đơn giản hóa quá trình huấn luyện mô hình

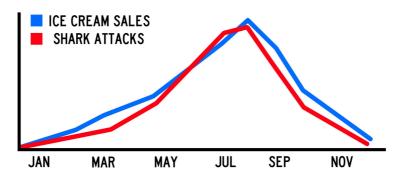
✓ M2AIO43 ^

- A. Để tăng độ chính xác của mô hình
- B. Để giảm thời gian tính toán
- C. Để chuẩn hóa dữ liệu
- D. Để tránh xác suất bằng 0 cho các từ chưa xuất hiện



Description

CORRELATION IS NOT CAUSATION!



Both ice cream sales and shark attacks increase when the weather is hot and sunny, but they are not caused by each other (they are caused by good weather, with lots of people at the beach, both eating ice cream and having a swim in the sea)

Câu phát biểu nào đúng nhất cho hình bên trên?

- A. Hai biến có tương quan luôn có quan hệ nhân quả
- B. Hai biến có tương quan không bao giờ có quan hệ nhân quả
- C. Tương quan và nhân quả là hai khái niệm hoàn toàn độc lập
- D. Tương quan không nhất thiết có quan hệ nhân quả <a>



Chọn đáp án đúng.

Một hàm Python thực hiện Laplacian Smoothing cho một từ điển tần suất từ. Hàm này nhận vào hai tham số:

- Một từ điển word_freq có dạng {word: frequency}, trong đó frequency là số lần xuất hiện của từ word trong dữ liệu.
- Tổng số từ total_words trong dữ liệu.

Hàm cần trả về một từ điển mới với xác suất của các từ đã được điều chỉnh bằng phương pháp Laplacian smoothing.

```
def laplacian_smoothing(word_freq, total_words):
    return {word: (freq + 1) / (total_words + 1) for word, freq in word_freq.items()}

B.

def laplacian_smoothing(word_freq, total_words):
    return {word: (freq + 1) / (total_words + len(word_freq)) for word, freq in word_freq.items()}

C.

def laplacian_smoothing(word_freq, total_words):
    return {word: freq / total_words for word, freq in word_freq.items()}

D.

def laplacian_smoothing(word_freq, total_words):
    vocab_size = len(word_freq)
    return {word: freq / (total_words + vocab_size) for word, freq in word_freq.items()}
```

Α

В

C

D



Description

Lưu ý:

- Sử dụng numpy để xử lý vấn đề
- Đọc kỹ input, output của đề bài để thông qua các test cases

Cơ sở lý thuyết:

Trung bình (hay giá trị trung bình) của một tập dữ liệu $X=\{x_1,x_2,\dots,x_N\}$ được tính bằng công thức:

$$\mu = rac{1}{N} \sum_{i=1}^N x_i$$

Trong đó, μ là giá trị trung bình, N là số lượng phần tử trong tập dữ liệu, và x_i là các giá trị dữ liệu riêng lẻ.

Câu hỏi coding:

Cho một tập dữ liệu $X=\{-2,7,3,0,9,1,-3,6,5,-3\}$, tính toán giá trị trung bình μ của tập dữ liệu nàv.

Giải thích chi tiết:

- Input: Một danh sách các số nguyên, ví dụ: `[-2, 7, 3, 0, 9, 1, -3, 6, 5, -3]`.
- Output: Giá trị trung bình của danh sách các số nguyên, kiểu dữ liệu là số thực (`float`). Ví dụ, đối với input trên, giá trị trung bình được tính toán sẽ là `1.9`.

Code

```
class Solution:
def calculate_mean(self, X):
return sum(X) / len(X)
pass
```

Solution Code

```
class Solution:
def calculate_mean(self, X):
    X = np.array(X)
mean_value = np.mean(X)
return float(mean_value)
```

Test cases

Test case 1 <

Input	Output	Expected output
X = [-2, 7, 3, 0, 9, 1, -3, 6, 5, -3]	2.3	2.3

Test case 2

Input	Output	Expected output
X = [10, 20, 30, 40, 50]	30.0	30.0



Description

Lưu ý:

- Có thể numpy để giải quyết vấn đề hoặc code hoàn toàn bằng python.
- Lưu ý input, output của bài toàn để vượt qua toàn bộ test cases.

Cơ sở lý thuyết:

Trung vị (hay median) của một tập dữ liệu $X=\{x_1,x_2,\ldots,x_N\}$ được tính bằng cách:

- 1. Sắp xếp dữ liệu: Đầu tiên, sắp xếp tập dữ liệu X thành một dãy tăng dần S.
- 2. Tính trung vị:
 - Nếu số lượng phần tử N là lẻ, trung vị m là phần tử ở vị trí giữa của dãy S: $m=S_{\frac{N+1}{2}}.$
 - Nếu số lượng phần tử N là chẵn, trung vị m là trung bình cộng của hai phần tử giữa của dãy S: $m=rac{1}{2}(S_{rac{N}{2}}+S_{rac{N}{2}+1})$.

Câu hỏi coding:

Cho một tập dữ liệu $X=\{x_1,x_2,\ldots,x_N\}$, tính toán giá trị trung vị m của tập dữ liệu này.

Giải thích chi tiết

- Input: Một danh sách các số nguyên, ví dụ: `[7, 0, 6, 2, 5, 7]`.
- Output: Giá trị trung vị của danh sách các số nguyên, kiểu dữ liệu là số thực (`float`). Ví dụ, đối với input trên, ta sắp xếp dữ liệu thành `[ø, 2, 5, 6, 7, 7]`. Vì số lượng phần tử là chẵn, giá trị trung vị là trung bình cộng của hai phần tử giữa: $\frac{1}{2}(5+6)=5.5$.

Code

Solution Code

```
class Solution:
def calculate_median(self, X):
    X = np.array(X)
median_value = np.median(X)
return float(median_value)
```

Test cases

Test case 1 ✓

Test case 2 ✓

Input	Output	Expected output
X = [1, 3, 5, 7, 9]	5.0	5.0

✓ M2AIO48 ^

Description

Lưu ý:

- Dùng numpy để giải quyết vấn đề
- Chú ý input, output của bài toán để pass qua tất cả test cases
- Kết quả làm tròn đến 2 chữ số thập phân

Cơ sở lý thuyết:

• Trung bình: Được tính bằng công thức:

$$\mu = rac{1}{N} \sum_{i=1}^N x_i$$

• Phương sai: Được tính bằng công thức:

$$\mathrm{var}(X) = \frac{1}{N} \sum_{i=1}^N (x_i - \mu)^2$$

• Độ lệch chuẩn: Là căn bậc hai của phương sai:

$$\sigma = \sqrt{\mathrm{var}(X)}$$

Câu hỏi coding:

Cho một tập dữ liệu $X=\{6,3,4,9,2,1\}$, tính toán phương sai $\mathrm{var}(X)$ và độ lệch chuẩn σ của tập dữ liệu này.

Giải thích chi tiết:

- Input: Một danh sách các số nguyên, ví dụ: `[6, 3, 4, 9, 2, 1]`.
- Output: Phương sai và độ lệch chuẩn của danh sách các số nguyên, cả hai đều có kiểu dữ liệu là số thực ('float'). Ví dụ, đối với input trên, phương sai là '7.139' và độ lệch chuẩn là '2.671'.

Code

```
import numpy as np

class Solution:
def calculate_variance_std(self, data):

# Your code here
variance = round(float(np.var(data)), 2)

std_dev = round(float(np.std(data)), 2)

return variance, std dev
```

Solution Code

```
data = np.array(data)
variance = np.var(data, ddof=0)

std_dev = np.std(data, ddof=0)
```

Test case 1

Input Output Expected output

data = [6, 3, 4, 9, 2, 1] (7.14, 2.67) (7.14, 2.67)

Test case 2

Input Output Expected output

data = [10, 20, 30, 40, 50] (200.0, 14.14) (200.00, 14.14)

X M2AIO49 ^

Description

Lưu ý:

- Dùng numpy để xử lý vấn đề
- Đọc kĩ input, output của bài toàn để vượt qua tất cả test cases.
- Kết quả làm tròn đến 2 chữ số thập phân

Cơ sở lý thuyết:

- Hệ số tương quan (Correlation Coefficient) giữa hai biến ngẫu nhiên X và Y được tính bằng công thức:

$$p_{xy} = \frac{E[(X-\mu_X)(Y-\mu_Y)]}{\sigma_X\sigma_Y} = \frac{n(\sum_i x_i y_i) - (\sum_i x_i)(\sum_i y_i)}{\sqrt{n\sum_i x_i^2 - (\sum_i x_i)^2} \sqrt{n\sum_i y_i^2 - (\sum_i y_i)^2}}$$

Trong đó:

- ullet μ_X và μ_Y là giá trị trung bình của X và Y.
- ullet σ_X và σ_Y là độ lệch chuẩn của X và Y.
- ullet n là số lượng phần tử trong tập dữ liệu.

Câu hỏi coding:

Cho dữ liệu X và Y, tính toán hệ số tương quan giữa X và Y.

Giải thích chi tiết:

- Input: Hai danh sách các số nguyên, ví dụ: `X = [7, 15, 10, 12, 8, 6, 15]` và `Y = [18, 6, 16, 10, 11, 12, 17]`.
- Output: Hệ số tương quan giữa hai danh sách số nguyên, kiểu dữ liệu là số thực ('float'). Ví dụ, đối với input trên, hệ số tương quan được tính toán sẽ là 'e.93' (đã làm tròn đến 2 chữ số thập phân).

Code

Solution Code

7

Test cases

Test case 1 ☑

Input	Output	Expected output
X = [7, 15, 10, 12, 8, 6, 15] Y = [18, 6, 16, 10, 11, 12, 17]	-0.29	-0.29
Test case 2 ✓		
Input	Output	Expected output
X = [1, 2, 3, 4, 5] Y = [2, 4, 6, 8, 10]	1.0	1.0



Description

Lưu ý:

- Sử dụng numpy để giải quyết vấn đề.
- Chú ý input, output của bài toán đề vượt qua các test cases.

Cơ sở lý thuyết:

- Tích vô hướng (Dot Product) giữa hai vector ${\bf v}$ và ${\bf u}$ được tính bằng công thức:

$$\mathbf{v} \cdot \mathbf{u} = v_1 \cdot u_1 + v_2 \cdot u_2 + \dots + v_n \cdot u_n$$

Trong đó,
$$\mathbf{v}=egin{bmatrix} v_1\\v_2\\\vdots\\v_n \end{bmatrix}$$
 và $\mathbf{u}=egin{bmatrix} u_1\\u_2\\\vdots\\u_n \end{bmatrix}$ là các vector.

Câu hỏi coding:

Cho hai vector ${\bf v}$ và ${\bf u}$, tính toán tích vô hướng ${\bf v}\cdot{\bf u}$.

Giải thích chi tiết:

- Input: Hai vector, mỗi vector là một danh sách các số nguyên. Ví dụ: `[1, 2, 3, 4]` và `[2, 3, 4, 5]`. Vector đầu vào có cùng số lượng phần tử.
- Output: Tích vô hướng của hai vector, kiểu dữ liệu là số nguyên (`int`). Kết quả là tổng của các tích của các phần tử tương ứng trong hai vector.

Code

```
import numpy as np

class Solution:
def calculate_dot_product(self, v, u):
    # Your code here
dot_product = np.dot(v, u)
return int(dot_product)
pass
```

Solution Code

```
6 return dot_product
7
```

Test cases

Test case 1 <

Input	Output	Expected output
v = [1, 2, 3, 4] u = [2, 3, 4, 5]	40	40

Test case 2

Input	Output	Expected output
v = [5, 6, 7]	38	38
u = [1, 2, 3]		

✓ M2AIO51 ^

Description

Lưu ý:

- Kết quả trả về ở dạng numpy.float64, tức không cần ép kiểu về dạng float.
- Nếu A, B 2 vector không có shape phù hợp, trả về -1

Cơ sở lý thuyết:

- Nhân Ma Trận (Matrix Multiplication) giữa hai ma trận ${f A}$ và ${f B}$ được thực hiện như sau:

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1k} \\ b_{21} & b_{22} & \cdots & b_{2k} \\ \vdots & \vdots & \ddots & \vdots \\ b_{n1} & b_{n2} & \cdots & b_{nk} \end{bmatrix}$$

Ma trận kết quả ${f C}$ sẽ có kích thước m imes k và mỗi phần từ c_{ij} của ma trận ${f C}$ được tính bằng:

$$c_{ij} = \sum_{l=1}^n a_{il} \cdot b_{lj}$$

Trong đó, $\mathbf{A} \in \mathbb{R}^{m imes n}$ và $\mathbf{B} \in \mathbb{R}^{n imes k}$.

Coding Problem:

Câu hỏi coding:

Cho hai ma trận ${f A}$ và ${f B}$, tính toán ma trận ${f C}$ bằng cách nhân ${f A}$ với ${f B}$.

Giải thích chi tiết:

• Input: Hai ma trận, ví dụ:

$$\mathbf{A} = \begin{bmatrix} 8 & 8 & 1 \\ 2 & 7 & 0 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 8 & 1 \\ 7 & 1 \\ 2 & 8 \end{bmatrix}$$

Ma trận ${f A}$ có kích thước 2 imes 3 và ma trận ${f B}$ có kích thước 3 imes 2.

• Output: Ma trận kết quả ${f C}$, kiểu dữ liệu là một ma trận với kích thước 2×2 . Ví dụ, đối với input trên, ma trận kết quả sẽ là:

$$\mathbf{C} = \begin{bmatrix} 72 & 16 \\ 33 & 15 \end{bmatrix}$$

```
class Solution:
def calculate_matrix_product(self, A, B):
    # Your code here
try:
result = np.dot(A, B)
return result
```

Solution Code

10 return C 11

Test cases

Test case 1 <a>Z

Input	Output	Expected output	
A = [[8, 8, 1], [2, 7, 0]] array([[122, 24], B = [[8, 1], [7, 1], [2, 8]] [65, 9]])		np.array([[122, 24], [65, 9]	
Test case 2 ☑			
Input	Output	Expected output	
A = [[1, 2], [3, 4]]	array([[19, 22],	np.array([[19, 22], [43, 50]])	
B = [[5, 6], [7, 8]]	[43, 50]])		

X M2AIO52 ^

Description

Lưu ý:

- Sử dụng python thuần, không sử dụng thư viện numpy
- Ma trận A luôn có shape là 2×2

Cơ sở lý thuyết:

• Eigenvalue (Giá trị riêng) và Eigenvector (Vector riêng): Để tính giá trị riêng và vector riêng của ma trận \mathbf{A} , ta sử dụng các công thức sau:

Eigenvalue
$$(\lambda)$$
: $det(\mathbf{A} - \lambda \mathbf{I}) = 0$

Trong đó, ${f I}$ là ma trận đơn vị cùng kích thước với ${f A}$.

Eigenvector
$$(\mathbf{v})$$
: $\mathbf{A}\mathbf{v} = \lambda \mathbf{v} \iff \mathbf{A}\mathbf{v} - \lambda \mathbf{I}\mathbf{v} = 0$

• Vector chuẩn hóa: Vector riêng ${\bf v}$ có thể được chuẩn hóa để có độ dài bằng 1. Nếu ${\bf v}$ là vector riêng, vector chuẩn hóa của nó được tính bằng:



Câu hỏi coding:

Cho ma trận ${\bf A}$, tính toán các giá trị riêng (eigenvalues) và các vector riêng (eigenvectors) đã được chuẩn hóa.

Giải thích chi tiết:

• Input: Ma trận vuông **A**. Ví dụ:

$$\mathbf{A} = egin{bmatrix} 1 & 4 \ 3 & 2 \end{bmatrix}$$

- Output: Các giá trị riêng (eigenvalues) và các vector riêng (eigenvectors) đã được chuẩn hóa của ma trận ${\bf A}$. Ví dụ, với ma trận trên:
 - Giá trị riêng: $\lambda_1=5, \lambda_2=-2$
 - Vector riêng chuẩn hóa: $\mathbf{v_1} = [-0.8, 0.6]^T$ và $\mathbf{v_2} = [-0.7071, -0.7071]^T$

Code

```
import math

class Solution:
def calculate_eigenvalues_eigenvectors(self, A):
def get_eigenvalues(matrix):
    # Your code here
return eigenvalue1, eigenvalue2
```

Solution Code

```
1 import math
2
3 class Solution:
```



Figure and Comments of the Com

```
def calculate_eigenvalues_eigenvectors(self, A):
    def get_eigenvalues(matrix):
        a, b, c, d = matrix[0][0], matrix[0][1], matrix[1][0], matrix[1][1]
        trace = a + d
        determinant = a * d - b * c
```

Error

Test cases

Test case 1 X

Input

A = [[1, 4], [3, 2]]	NameError: name 'eigenvalue1' is not defined	{5.0: [-0.7071067811865476, -0.7071067811865476], -2.0: [-0.6, 0.8]}
Test case 2 X		
Input	Error	Expected output
A = [[2, 1], [1, 2]]	NameError: name 'eigenvalue1' is not defined	{3.0: [-0.7071067811865475, -0.7071067811865475], 1.0: [-0.7071067811865475,

Expected output

0.7071067811865475]}

× M2AIO53 (Bonus) ^

Description

Câu này không có trong file PDF (Optional)

Yêu cầu:

Cho hai vector x và y. Hãy tính giá trị cosine similarity của chúng (không sử dụng thư viện numpy)

Input:

• Vector x và vector y (chứa các số nguyên)

Output:

Giá trị cosine similarity

Lưu ý:

- Nếu hai vector không có cùng độ dài: return -1
- Có thể sử dụng thư viện math

Cosine Similarity

- Data (vector \mathbf{x}, \mathbf{y}): $\mathbf{X} = \{x_1, ..., x_N\} \mathbf{y} = \{y_1, ..., y_N\}$
- Cosine Similarity: $cs(\mathbf{x}, \mathbf{y}) = \frac{\mathbf{x} \cdot \mathbf{y}}{\parallel \mathbf{x} \parallel \parallel \mathbf{y} \parallel} = \frac{\sum_{1}^{n} x_{i} y_{i}}{\sqrt{\sum_{1}^{n} x_{i}^{2}} \sqrt{\sum_{1}^{n} y_{i}^{2}}}$

PROPERTY OF THE PROPERTY OF T

March Street Comments of the C

Code

```
class Solution:
def calculate_cosine_similarity(self, x, y):
    # Your code here
if len(x) != len(y):
    return -1

dot_product = sum(xi * yi for xi, yi in zip(x, y))
```

Solution Code

```
import math

class Solution:
def calculate_cosine_similarity(self, x, y):
    if len(x) != len(y):
        return -1
    # Calculate dot product
dot product = sum(x i * y i for x i, y i in zip(x, y))
```

Test cases

Test case 1 <

Input	Output	Expected output
x = [1, 3, 4, 3] y = [0, 3, 4, 0]	0.8451542547285166	0.8451542547285166

Test case 2

Input	Output	Expected output
x = [1, 1, 1] y = [1, 1, 1]	1.000000000000002	1.0000000000000000000002