KHAI PHÁ DỮ LIỆU – CHƯƠNG 4
PHÂN LỚP DỮ LIỆU (DATA CLASSIFICATION)

Tổng hợp kiến thức, câu hỏi tự luận và trắc nghiệm

1. TỔNG QUAN VỀ PHÂN LỚP

1.1. Định nghĩa và quy trình

Phân lớp (Classification) là phương pháp phân tích dữ liệu nhằm trích xuất mô hình mô tả các lớp dữ liệu. Đây là quy trình **hai bước**:

1. **Huấn luyện (Training):** Xây dựng bộ phân lớp bằng cách học (analyzing) tập huấn luyện có nhãn $\langle X, y \rangle$.
2. **Phân lớp (Classification):** Phân lớp các mẫu/đối tượng mới. Nếu độ chính xác của bộ phân lớp chấp nhận được, dùng nó để phân lớp dữ liệu mới.

Đặc điểm: Phân lớp là học **có giám sát (supervised learning)** – có nhãn lớp trong dữ liệu huấn luyện.

Ví dụ tình huống:

- Email: "spam" hay "not spam".
- Giao dịch trực tuyến: "gian lận" hay "bình thường".
- Y tế: u bướu "lành tính" hay "ác tính"; bệnh nhân "mắc bệnh" hay "không mắc bệnh".

Các thuật toán phân lớp phổ biến:

- Logistic Regression, Decision Tree, Bayesian method, ANN, K-NN, Case-based reasoning, Genetic algorithms, Rough/Fuzzy sets.

2. HỒI QUY LOGISTIC (LOGISTIC REGRESSION)

2.1. Hàm Sigmoid

Hồi quy tuyến tính $h_\theta(X) = \theta^T X$ có thể cho kết quả > 1 hoặc < 0, không phù hợp cho phân lớp. Hồi quy Logistic đảm bảo $0 \leq h_\theta(X) \leq 1$ bằng hàm Sigmoid (Logistic function):

$$h_\theta(X) = g(\theta^T X) = \frac{1}{1 + e^{-\theta^T X}}$$

$$g(z) = \frac{1}{1 + e^{-z}}, \quad g(z) \in (0, 1)$$

Ý nghĩa: $h_\theta(x) = P(y = 1 \mid x; \theta)$ – xác suất nhãn là 1 khi đầu vào là x.

2.2. Decision Boundary (Ranh giới quyết định)

- Dự đoán $y = 1$ khi $h_\theta(X) \geq 0.5 \Leftrightarrow \theta^T X \geq 0$.
- Dự đoán $y = 0$ khi $h_\theta(X) < 0.5 \Leftrightarrow \theta^T X < 0$.

Decision boundary là tập $\{\theta^T X = 0\}$ – có thể là đường thẳng (tuyến tính) hoặc đường cong (phi tuyến) tùy thuộc vào đặc trưng.

2.3. Hàm chi phí và Gradient Descent

Hàm chi phí logistic:

$$\text{cost}(h_\theta(x), y) = \begin{cases} -\log(h_\theta(x)) & \text{nếu } y = 1 \\ -\log(1 - h_\theta(x)) & \text{nếu } y = 0 \end{cases}$$

Rút gọn: $\text{cost}(h_\theta(x), y) = -y \log(h_\theta(x)) - (1 - y) \log(1 - h_\theta(x))$

$$J(\theta) = \frac{1}{N} \sum_{i=1}^{N} \left[-y^{(i)} \log(h_\theta(x^{(i)})) - (1 - y^{(i)}) \log(1 - h_\theta(x^{(i)})) \right]$$

Tối thiểu hóa $J(\theta)$ bằng gradient descent (dạng công thức giống hồi quy tuyến tính sau khi lấy đạo hàm).

2.4. Phân lớp đa lớp – One-vs-Rest

Với k lớp, huấn luyện k bộ phân lớp $h_\theta^{(i)}(x)$ ($i = 1, \ldots, k$), mỗi bộ phân biệt lớp i với tất cả lớp còn lại. Dự đoán lớp i có $h_\theta^{(i)}(x)$ lớn nhất.

3. CÂY QUYẾT ĐỊNH (DECISION TREE)

3.1. Cấu trúc và ý nghĩa

- **Internal node (Nút trong):** Kiểm tra một thuộc tính cụ thể.
- **Branch (Nhánh):** Kết quả của phép kiểm tra.
- **Leaf node (Nút lá):** Nhãn lớp.

3.2. Thuật toán xây dựng cây quyết định

Đặc điểm: Greedy, divide-and-conquer, đệ quy, top-down. Độ phức tạp: $O(n \times |D| \times \log|D|)$.

Các thuật toán: ID3, C4.5, CART (Classification and Regression Trees – cây nhị phân).

3.3. Tiêu chí chọn thuộc tính phân tách

(a) Information Gain (ID3):

$$\text{Info}(D) = -\sum_{i=1}^{m} p_i \log_2(p_i), \quad p_i = |C_{i,D}|/|D|$$

$$\text{Info}_A(D) = \sum_{j=1}^{v} \frac{|D_j|}{|D|} \text{Info}(D_j)$$

$$\text{Gain}(A) = \text{Info}(D) - \text{Info}_A(D)$$

Chọn thuộc tính có **Gain lớn nhất**. Hạn chế: ưu tiên thuộc tính có nhiều giá trị (tạo nhiều partition nhỏ).

(b) Gain Ratio (C4.5):

$$\text{SplitInfo}_A(D) = -\sum_{j=1}^{v} \frac{|D_j|}{|D|} \log_2 \frac{|D_j|}{|D|}$$

$$\text{GainRatio}(A) = \frac{\text{Gain}(A)}{\text{SplitInfo}_A(D)}$$

Chuẩn hóa Information Gain theo độ phân tách; giảm thiên vị với thuộc tính nhiều giá trị.

(c) Gini Index (CART):

$$\text{Gini}(D) = 1 - \sum_{i=1}^{m} p_i^2$$

$$\text{Gini}_A(D) = \frac{|D_1|}{|D|}\text{Gini}(D_1) + \frac{|D_2|}{|D|}\text{Gini}(D_2)$$

Chia nhị phân; chọn thuộc tính và ngưỡng phân tách có **Gini nhỏ nhất**.

4. PHÂN LỚP BAYESIAN

4.1. Định lý Bayes

$$P(H \mid X) = \frac{P(X \mid H) \cdot P(H)}{P(X)}$$

- $P(H \mid X)$: xác suất hậu nghiệm (posterior) – xác suất X thuộc lớp H.
- $P(X \mid H)$: likelihood – xác suất quan sát X nếu lớp là H.
- $P(H)$: xác suất tiên nghiệm của lớp (prior).
- $P(X)$: hằng số chuẩn hóa.

4.2. Naive Bayesian Classification

Giả định **class conditional independence**: các thuộc tính độc lập với nhau trong cùng một lớp:

$$P(X \mid C_i) = \prod_{k=1}^{n} P(x_k \mid C_i)$$

Phân lớp: Chọn lớp C_i có $P(C_i \mid X) \propto P(X \mid C_i) \cdot P(C_i)$ lớn nhất.

Tính $P(x_k \mid C_i)$:

- Thuộc tính danh mục: $P(x_k \mid C_i) = \frac{|\{X' | x'_k = x_k \wedge X' \in C_i\}|}{|C_{i,D}|}$
- Thuộc tính liên tục (Gauss): $P(x_k \mid C_i) = \frac{1}{\sqrt{2\pi}\sigma_i} \exp\left(-\frac{(x_k-\mu_i)^2}{2\sigma_i^2}\right)$

Vấn đề xác suất bằng 0: Nếu $P(x_k \mid C_i) = 0$ thì $P(X \mid C_i) = 0$, gây ra "zero-frequency problem".

Giải pháp Laplace smoothing:

$$P(x_k \mid C_i) = \frac{|\{X' \mid x'_k = x_k \wedge X' \in C_i\}| + 1}{|C_{i,D}| + m}$$

trong đó m là số giá trị khác nhau của thuộc tính A_k.

Ưu/nhược điểm của Naive Bayes:

- Ưu: Dễ cài đặt, học nhanh, dễ hiểu kết quả, hiệu quả trong nhiều trường hợp.
- Nhược: Giả định độc lập lớp điều kiện có thể không thỏa trong thực tế.

5. MẠNG NƠRON NHÂN TẠO (ANN)

5.1. Mô hình Neuron – Logistic Unit

Mỗi neuron nhân tạo là một logistic unit với hàm kích hoạt Sigmoid $g(z)$:

$$a_j^{(l)} = g\left(\sum_i \Theta_{ji}^{(l-1)} a_i^{(l-1)}\right)$$

5.2. Kiến trúc ANN

- **Layer 1:** Input layer (lớp đầu vào).
- **Layer 2, ..., L-1:** Hidden layers (lớp ẩn).
- **Layer L:** Output layer (lớp đầu ra).

Ma trận trọng số $\Theta^{(j)}$ có kích thước $s_{j+1} \times (s_j + 1)$ (với s_j là số nút tại lớp j).

5.3. Feedforward (Forward Propagation)

Tính giá trị kích hoạt từ lớp đầu vào đến lớp đầu ra:

$$z^{(l+1)} = \Theta^{(l)} a^{(l)}, \quad a^{(l+1)} = g(z^{(l+1)})$$

(thêm bias node $a_0^{(l)} = 1$ tại mỗi lớp)

5.4. Hàm chi phí ANN

Phân lớp nhị phân ($K = 1$):

$$J(\Theta) = -\frac{1}{N} \sum_{i=1}^{N} \left[y^{(i)} \log h_\Theta(x^{(i)}) + (1 - y^{(i)}) \log(1 - h_\Theta(x^{(i)})) \right]$$

5.5. Backpropagation

Thuật toán tính gradient để tối ưu hóa $J(\Theta)$:

1. **Feedforward:** Tính $a^{(l)}$ cho tất cả các lớp.
2. **Tính lỗi:** $\delta^{(L)} = a^{(L)} - y$.
3. **Lan truyền ngược:** $\delta^{(l)} = (\Theta^{(l)})^T \delta^{(l+1)} \cdot g'(z^{(l)})$.

4. **Tính gradient:** $\frac{\partial J}{\partial \Theta_{ij}^{(l)}} = a_j^{(l)} \cdot \delta_i^{(l+1)}$.

5. Cập nhật $\Theta^{(l)}$ bằng gradient descent.

6. K-NEAREST NEIGHBOR (K-NN)

Ý tưởng: Phân lớp đối tượng X bằng cách bỏ phiếu đa số (majority vote) từ k đối tượng gần nhất trong tập huấn luyện.

Độ đo khoảng cách: Thường dùng Euclidean: $d(p,q) = \sqrt{\sum_i (p_i - q_i)^2}$

Chọn k:

- k quá nhỏ: nhạy cảm với nhiễu.
- k quá lớn: có thể chọn đối tượng từ lớp khác.
- Gợi ý: $k \leq \sqrt{|D|}$.

7. ĐÁNH GIÁ VÀ CHỌN MÔ HÌNH PHÂN LỚP

7.1. Tiêu chí đánh giá

- **Accuracy:** Khả năng nhận dạng đúng các đối tượng.
- **Speed:** Chi phí tính toán khi huấn luyện và sử dụng.
- **Robustness:** Hoạt động tốt với dữ liệu nhiễu hoặc thiếu.
- **Scalability:** Xây dựng và cập nhật bộ phân lớp với tập dữ liệu rất lớn.
- **Interpretability:** Khả năng hiểu cách bộ phân lớp hoạt động.

7.2. Precision, Recall và F-score

	Thực tế: X	**Thực tế: !X**
Dự đoán: X	TP	FP
Dự đoán: !X	FN	TN

$$\text{Precision} = \frac{TP}{TP+FP}, \quad \text{Recall} = \frac{TP}{TP+FN}$$

$$\text{F-score} = \frac{2 \times P \times R}{P+R}$$

7.3. Phương pháp đánh giá

- **Holdout method:** Chia ngẫu nhiên D thành tập train (2/3) và test (1/3).
- **K-fold cross-validation:** Chia D thành k phần bằng nhau; lặp k lần (dùng phần i để test, phần còn lại để train); tính trung bình.

8. TÓM TẮT

- **Phân lớp:** Học có giám sát; 2 bước: training + classification.
- **Logistic Regression:** Sigmoid function; decision boundary; gradient descent; one-vs-rest đa lớp.
- **Decision Tree (ID3/C4.5/CART):** Tiêu chí phân tách: Information Gain, Gain Ratio, Gini Index.
- **Naive Bayes:** Định lý Bayes + giả định độc lập; Laplace smoothing cho zero-frequency.
- **ANN:** Feedforward + Backpropagation; xử lý phi tuyến phức tạp.
- **K-NN:** Lazy learning, dựa trên khoảng cách.
- **Đánh giá:** Accuracy, Precision, Recall, F-score; Holdout, K-fold cross-validation.

9. CÂU HỎI TỰ LUẬN

Câu 1. Phân lớp dữ liệu (Classification) là gì? Mô tả quy trình 2 bước của phân lớp và giải thích tại sao phân lớp là học có giám sát. So sánh với phân cụm (Clustering).

Câu 2. Giải thích tại sao hồi quy tuyến tính $h_\theta(X) = \theta^T X$ không phù hợp cho bài toán phân lớp. Hàm Sigmoid giải quyết vấn đề này như thế nào?

Câu 3. Trình bày hàm Sigmoid $g(z)$ và ý nghĩa của $h_\theta(x) = P(y = 1 \mid x; \theta)$ trong hồi quy Logistic. Quy tắc quyết định phân lớp dựa trên $h_\theta(x)$ và $\theta^T X$ như thế nào?

Câu 4. Giải thích khái niệm **Decision Boundary** trong hồi quy Logistic. Khi nào thì ranh giới quyết định là đường thẳng? Khi nào là đường cong? Cho ví dụ cụ thể.

Câu 5. Trình bày hàm chi phí của hồi quy Logistic. Tại sao không dùng MSE như hồi quy tuyến tính? Giải thích ý nghĩa của từng trường hợp $y = 1$ và $y = 0$.

Câu 6. Giải thích phương pháp **One-vs-Rest** để phân lớp đa lớp bằng Logistic Regression. Mô tả quy trình huấn luyện và dự đoán. Hạn chế của phương pháp này?

Câu 7. Mô tả cấu trúc của **Decision Tree**. Giải thích các khái niệm: internal node, branch, leaf node. Thuật toán xây dựng cây quyết định có đặc điểm gì (greedy, divide-and-conquer)?

Câu 8. Trình bày và so sánh 3 tiêu chí chọn thuộc tính phân tách trong cây quyết định: **Information Gain**, **Gain Ratio** và **Gini Index**. Khi nào Gain Ratio tốt hơn Information Gain?

Câu 9. Tính Information Gain của thuộc tính "age" trong tập dữ liệu AllElectronics: $\text{Info}(D) = 0.940$, $\text{Info}_{\text{age}}(D) = 0.694$. Kết quả Gain(age) = 0.246. Giải thích ý nghĩa của con số này.

Câu 10. Trình bày **Định lý Bayes**. Giải thích các khái niệm: posterior probability, prior probability, likelihood, evidence. Tại sao phân lớp Bayesian dựa trên "tối đa hóa posterior"?

Câu 11. Giải thích giả định **class conditional independence** trong Naive Bayes. Khi nào giả định này có thể không thỏa? Hệ quả khi giả định bị vi phạm là gì?

Câu 12. Trình bày **Laplace smoothing** trong Naive Bayes. Tại sao cần Laplace smoothing? Công thức điều chỉnh như thế nào? Có phương pháp thay thế nào không?

Câu 13. Mô tả kiến trúc **Mạng Nơron Nhân Tạo (ANN)**. Giải thích: input layer, hidden layers, output layer, weights matrix $\Theta^{(j)}$, kích thước của $\Theta^{(j)}$.

Câu 14. Trình bày thuật toán **Feedforward (Forward Propagation)** trong ANN. Vì sao cần thêm bias node $a_0^{(l)} = 1$ tại mỗi lớp?

Câu 15. Giải thích thuật toán **Backpropagation**. Mục đích của backpropagation là gì? Mô tả cách tính $\delta^{(L)}$, $\delta^{(L-1)}$ và gradient $\frac{\partial J}{\partial \Theta_{ij}^{(l)}}$.

Câu 16. Tại sao **ANN** được sử dụng cho bài toán phân lớp phi tuyến phức tạp mà Logistic Regression không xử lý được? Minh họa bằng ví dụ bài toán XOR.

Câu 17. Giải thích thuật toán **K-Nearest Neighbor (K-NN)**. Trình bày cách chọn k, độ đo khoảng cách và quy tắc bỏ phiếu đa số. Ưu và nhược điểm của K-NN?

Câu 18. Phân biệt **Precision** và **Recall**. Trong tình huống phát hiện gian lận thẻ tín dụng, tiêu chí nào quan trọng hơn? Tại sao cần F-score?

Câu 19. So sánh **Holdout method** và **K-fold cross-validation** để đánh giá bộ phân lớp. Phương pháp nào đáng tin cậy hơn và tại sao?

Câu 20. So sánh toàn diện 5 phương pháp phân lớp: Logistic Regression, Decision Tree, Naive Bayes, ANN và K-NN theo các tiêu chí: accuracy, speed, robustness, scalability và interpretability.

10. CÂU HỎI TRẮC NGHIỆM

Câu 1. Phân lớp (Classification) là loại học nào?

A. Học không giám sát.

B. Học có giám sát.

C. Học tăng cường.

D. Học bán giám sát.

Câu 2. Hàm Sigmoid $g(z) = \frac{1}{1+e^{-z}}$ có miền giá trị:

A. $(-\infty, +\infty)$

B. $[0,1]$ (nhưng không đạt đúng 0 hoặc 1, ngoại trừ giới hạn).

C. $[-1,1]$

D. $[0.5,1]$

Câu 3. Trong Logistic Regression, dự đoán $y = 1$ khi:

A. $h_\theta(x) \geq 0.5 \Leftrightarrow \theta^T x \geq 0$.

B. $h_\theta(x) < 0.5$.

C. $\theta^T x < 0$.

D. $h_\theta(x) = 1$.

Câu 4. $h_\theta(x) = P(y = 1 \mid x; \theta) = 0.7$ có nghĩa là:

A. 70% đối tượng thuộc lớp 0.

B. Xác suất 70% đối tượng thuộc lớp 1 với đầu vào x.

C. Mô hình chính xác 70%.

D. $\theta^T x = 0.7$.

Câu 5. Hàm chi phí Logistic Regression khi $y = 1$ và $h_\theta(x) \to 0$ sẽ:

A. Tiến về 0 (cost nhỏ).

B. Tiến về vô cực (cost rất lớn – phạt mạnh dự đoán sai).

C. Bằng 0.5.

D. Bằng $-\log(0.5)$.

Câu 6. Phương pháp **One-vs-Rest** trong phân lớp đa lớp với k lớp cần huấn luyện:

A. 1 bộ phân lớp.

B. $k-1$ bộ phân lớp.

C. k bộ phân lớp.

D. $k(k-1)/2$ bộ phân lớp.

Câu 7. Trong Decision Tree, **Leaf node** chứa:

A. Điều kiện kiểm tra thuộc tính.

B. Nhãn lớp (class label).

C. Tập con của dữ liệu huấn luyện.

D. Hệ số Gini Index.

Câu 8. Information Gain $\text{Gain}(A) = \text{Info}(D) - \text{Info}_A(D)$ là:

A. Lượng thông tin cần thiết để phân lớp sau khi dùng thuộc tính A.

B. Phần giảm entropy khi biết thuộc tính A (lớn hơn là tốt hơn).

C. Độ đo Gini của phân hoạch.

D. Xác suất thuộc tính A được chọn làm nút gốc.

Câu 9. Thuật toán cây quyết định nào sử dụng **Gain Ratio**?

A. ID3.

B. C4.5.

C. CART.

D. K-NN.

Câu 10. Gini Index **nhỏ hơn** tương ứng với:

A. Partition có độ hỗn loạn cao hơn (impure).

B. Partition thuần hơn (các đối tượng chủ yếu thuộc một lớp).

C. Thuộc tính phân tách kém hơn.

D. Entropy lớn hơn.

Câu 11. CART (Classification and Regression Trees) đặc điểm là:

A. Cây đa phân (multiple branches per node).

B. Cây nhị phân (binary split); sử dụng Gini Index.

C. Cây nhị phân; sử dụng Information Gain.

D. Cây đa phân; sử dụng Gain Ratio.

Câu 12. Trong Naive Bayes, giả định **class conditional independence** nghĩa là:

A. Tất cả các lớp có xác suất bằng nhau.

B. Các thuộc tính độc lập với nhau trong cùng một lớp.

C. Các lớp độc lập với nhau.

D. Thuộc tính liên tục và danh mục được xử lý giống nhau.

Câu 13. **Laplace smoothing** trong Naive Bayes giải quyết:

A. Vấn đề dữ liệu thiếu (missing data).

B. Vấn đề xác suất bằng 0 (zero-frequency problem).

C. Vấn đề giả định độc lập bị vi phạm.

D. Vấn đề chiều cao (high dimensionality).

Câu 14. Công thức Laplace smoothing: $P(x_k \mid C_i) = \frac{\text{count}+1}{|C_{i,D}|+m}$, trong đó m là:

A. Số lớp trong tập dữ liệu.

B. Số giá trị khác nhau của thuộc tính A_k.

C. Số mẫu trong tập huấn luyện.

D. Số thuộc tính của đối tượng.

Câu 15. Trong ANN, ma trận trọng số $\Theta^{(j)}$ ánh xạ:

A. Từ lớp $j+1$ về lớp j.

B. Từ lớp j sang lớp $j+1$.

C. Từ lớp đầu vào trực tiếp đến lớp đầu ra.

D. Từ lớp ẩn sang lớp đầu vào.

Câu 16. Kích thước của $\Theta^{(j)}$ trong ANN với s_j nút tại lớp j và s_{j+1} nút tại lớp $j+1$ là:

A. $s_j \times s_{j+1}$

B. $s_{j+1} \times (s_j + 1)$ (bao gồm bias)

C. $(s_j + 1) \times s_{j+1}$

D. $s_j \times (s_{j+1} + 1)$

Câu 17. Trong Backpropagation, $\delta_j^{(l)}$ biểu diễn:

A. Giá trị kích hoạt của neuron j tại lớp l.

B. Sai số (error) tạo ra bởi neuron j tại lớp l.

C. Trọng số kết nối của neuron j tại lớp l.

D. Giá trị gradient của J theo $\Theta_{ij}^{(l)}$.

Câu 18. K-NN phân lớp đối tượng mới bằng cách:

A. Tính xác suất thuộc mỗi lớp theo Bayes.

B. Bỏ phiếu đa số từ k đối tượng gần nhất trong tập huấn luyện.

C. Xây dựng cây quyết định từ tập huấn luyện.

D. Tính sigmoid của $\theta^T x$.

Câu 19. Khi k trong K-NN quá nhỏ, bộ phân lớp sẽ:

A. Ổn định hơn với nhiễu.

B. Nhạy cảm hơn với nhiễu (overfitting).

C. Có xu hướng phân lớp sai do chọn quá nhiều lớp.

D. Không bị ảnh hưởng bởi k.

Câu 20. Giá trị k được gợi ý trong slide C4 là:

A. $k = 1$

B. $k \leq \sqrt{|D|}$

C. $k = 10$

D. $k = |D|/2$

Câu 21. Precision được tính bằng:

A. $TP/(TP+FN)$

B. $TP/(TP+FP)$

C. $TN/(TN+FP)$

D. $(TP+TN)/(TP+FP+FN+TN)$

Câu 22. Recall được tính bằng:

A. $TP/(TP+FP)$

B. $TP/(TP+FN)$

C. $FP/(FP+TN)$

D. $TN/(TN+FN)$

Câu 23. F-score được tính bằng:

A. $(P+R)/2$

B. $2PR/(P+R)$

C. $\sqrt{P\times R}$

D. $P\times R$

Câu 24. Với dataset 13 flows (9 BG và 4 FG), bộ phân lớp xác định 7 flows là BG (4 BG đúng và 3 FG sai). Precision và Recall của lớp BG lần lượt là:

A. P = 4/9, R = 4/7.

B. P = 4/7, R = 4/9.

C. P = 4/13, R = 4/7.

D. P = 4/7, R = 4/13.

Câu 25. Thuật toán cây quyết định nào sử dụng **Information Gain** và phù hợp với thuộc tính danh mục?

A. C4.5.

B. CART.

C. ID3.

D. Naive Bayes.

Câu 26. Trong ANN, lớp đầu vào $L=1$ chứa:

A. Kết quả dự đoán.

B. Các đặc trưng đầu vào $x_1,\ldots,x_n$ (và bias $x_0=1$).

C. Các trọng số Θ.

D. Giá trị delta (lỗi) của mỗi neuron.

Câu 27. $\text{Info}(D)=-\sum_{i=1}^{m} p_i \log_2 p_i$ bằng **0** khi:

A. Tất cả đối tượng phân bổ đều giữa các lớp.

B. Tất cả đối tượng thuộc cùng một lớp (partition thuần nhất).

C. $m=2$ lớp.

D. $|D|=1$.

Câu 28. Bộ phân lớp nào **dễ giải thích nhất** (highest interpretability)?

A. ANN với nhiều hidden layers.

B. K-NN.

C. Decision Tree.

D. Logistic Regression với nhiều biến.

Câu 29. Ưu điểm chính của ANN so với Decision Tree là:

A. ANN dễ hiểu và giải thích hơn.

B. ANN xử lý tốt hơn các bài toán phân lớp phi tuyến phức tạp với dữ liệu chiều cao.

C. ANN cần ít dữ liệu huấn luyện hơn.

D. ANN huấn luyện nhanh hơn với mọi loại dữ liệu.

Câu 30. Tiêu chí **Scalability** trong đánh giá bộ phân lớp quan tâm đến:

A. Khả năng bộ phân lớp hiểu được kết quả.

B. Khả năng xây dựng và cập nhật bộ phân lớp với tập dữ liệu rất lớn.

C. Khả năng phân lớp đúng các đối tượng nhiễu.

D. Tốc độ dự đoán trên tập test.

Câu 31. Trong quy trình phân lớp 2 bước, bước **Evaluation** được thực hiện ở:

A. Chỉ ở bước Training.

B. Ở đầu bước Classification (dùng testing dataset để đánh giá trước khi áp dụng).

C. Sau khi đã phân lớp tất cả dữ liệu mới.

D. Không cần thực hiện evaluation nếu training dataset đủ lớn.

Câu 32. Phương trình $h_\theta(x) = g(\theta_0 + \theta_1 x_1^2 + \theta_2 x_2^2)$ tạo ra decision boundary dạng:

A. Đường thẳng.

B. Đường tròn (hoặc elipse).

C. Đường hyperbola.

D. Không thể xác định.

Câu 33. Trong K-fold cross-validation, tổng số lần một mẫu dữ liệu được dùng để test là:

A. k lần.

B. 1 lần.

C. $k-1$ lần.

D. $1/k$ lần.

Câu 34. Trong Naive Bayes, $P(C_i) = |C_{i,D}|/|D|$ biểu diễn:

A. Likelihood (xác suất quan sát X nếu lớp là C_i).

B. Prior probability (xác suất tiên nghiệm của lớp C_i).

C. Posterior probability (xác suất hậu nghiệm).

D. Evidence (hằng số chuẩn hóa).

Câu 35. ANN được lấy cảm hứng từ:

A. Hệ thống điều khiển robot.

B. Cấu trúc và hoạt động của não người.

C. Cấu trúc cây quyết định phân cấp.

D. Lý thuyết tập mờ (Fuzzy sets).

Câu 36. Khi Naive Bayes gặp vấn đề $P(x_k \mid C_i) = 0$ mà không dùng Laplace smoothing thì:

A. Chỉ ảnh hưởng đến thuộc tính đó.

B. Toàn bộ $P(X \mid C_i) = 0$, loại bỏ hoàn toàn lớp C_i khỏi xét.

C. Tăng xác suất của các lớp khác.

D. Không ảnh hưởng nếu có đủ thuộc tính khác.

Câu 37. Trong đánh giá mô hình phân lớp, độ đo **Accuracy** có hạn chế khi:

A. Tập test quá nhỏ.

B. Dữ liệu mất cân bằng lớp (class imbalance), ví dụ 99% lớp A và 1% lớp B.

C. Số lớp lớn hơn 2.

D. Bộ phân lớp sử dụng gradient descent.

ĐÁP ÁN

Câu hỏi tự luận – Hướng dẫn trả lời

Câu	Nội dung cần trình bày
1	Classification: xây dựng hàm $f(X) = y$ từ dữ liệu có nhãn. Bước 1: Training (học mô hình từ $\langle X, y \rangle$). Bước 2: Classification (phân lớp dữ liệu mới). Supervised vì có nhãn. Khác clustering: clustering không có nhãn, tự tìm nhóm.
2	Hồi quy tuyến tính: $h_\theta(X)$ có thể > 1 hoặc $< 0 \Rightarrow$ không phải xác suất. Sigmoid: $g(z) = 1/(1+e^{-z}) \in (0,1) \Rightarrow$ đảm bảo $0 \leq h_\theta \leq 1$, có thể diễn giải là xác suất.
3	$h_\theta(x) = g(\theta^T x) \in (0,1)$. Ý nghĩa: $P(y=1|x;\theta)$. Quy tắc: dự đoán $y=1$ khi $h_\theta(x) \geq 0.5 \Leftrightarrow \theta^T x \geq 0$; dự đoán $y=0$ khi ngược lại.
4	Decision boundary là tập $\{\theta^T x = 0\}$. Đường thẳng khi đặc trưng tuyến tính $\theta_0 + \theta_1 x_1 + \theta_2 x_2 = 0$. Đường cong khi thêm đặc trưng bậc cao: $\theta_0 + \theta_1 x_1^2 + \theta_2 x_2^2 = 0 \Rightarrow$ đường tròn.
5	Dùng log-loss thay MSE vì: MSE với sigmoid cho J không lồi (nhiều cực tiểu cục bộ). Log-loss: $y=1, h \to 0$: cost $\to \infty$ (phạt mạnh); $y=1, h=1$: cost $=0$. Tương tự $y=0$.
6	One-vs-Rest: với k lớp, huấn luyện k bộ phân lớp $h_\theta^{(i)}$, mỗi cái phân biệt lớp i vs. tất cả còn lại. Dự đoán: lớp $i = \arg\max_i h_\theta^{(i)}(x)$. Hạn chế: không xét đồng thời tất cả lớp.
7	Internal node: test thuộc tính (ví dụ: age ≤ 30). Branch: kết quả test. Leaf: nhãn lớp. Thuật toán: greedy (chọn thuộc tính tốt nhất tại mỗi nút, không quay lại), divide-and-conquer, đệ quy, top-down.
8	Information Gain: đo giảm entropy; có thể ưu tiên thuộc tính nhiều giá trị (bad). Gain Ratio: chuẩn hóa bằng SplitInfo $\Rightarrow$ khắc phục thiên vị của IG; dùng khi thuộc tính có nhiều giá trị. Gini: phân tách nhị phân, đơn giản hơn.
9	Gain(age) = 0.940 - 0.694 = 0.246 bits. Nghĩa: Khi biết thuộc tính "age", lượng thông tin cần thêm để phân lớp giảm 0.246 bits. Để so sánh với Gain(income), Gain(student), Gain(credit_rating) để chọn thuộc tính phân tách.
10	Bayes: $P(H|X) = P(X|H)P(H)/P(X)$. Posterior: xác suất thuộc lớp sau khi quan sát X. Prior: xác suất lớp trước khi quan sát. Likelihood: xác suất quan sát X nếu lớp là H. Phân lớp tối đa hóa posterior để chọn lớp có khả năng cao nhất.

Câu	Nội dung cần trình bày
11	Giả định: $P(X\|C_i) = \prod_k P(x_k\|C_i)$ – các thuộc tính độc lập trong mỗi lớp. Không thỏa khi: thuộc tính có tương quan (ví dụ: chiều cao và cân nặng). Hệ quả: xác suất tính toán không phản ánh đúng thực tế, có thể cho kết quả sai.
12	Zero-frequency: $P(x_k\|C_i) = 0$ khi giá trị x_k chưa gặp trong lớp C_i trong training $\Rightarrow P(X\|C_i) = 0$, loại bỏ lớp sai. Laplace: thêm 1 vào tử, m vào mẫu. Thay thế: z-estimate: $P = (\text{count} + zP(x_k))/(\|C_i\| + z)$.
13	Layer 1: input $(x_1, \ldots, x_n)$. Hidden layers: xử lý phi tuyến. Layer L: output. $\Theta^{(j)}$: ma trận trọng số từ lớp j sang $j+1$; kích thước $s_{j+1} \times (s_j + 1)$ (bias node thêm 1 vào s_j).
14	Feedforward: $z^{(l+1)} = \Theta^{(l)} a^{(l)}$; $a^{(l+1)} = g(z^{(l+1)})$; thêm $a_0^{(l)} = 1$ tại mỗi lớp. Bias node cần thiết để cho phép dịch chuyển (shift) decision boundary, tương tự intercept θ_0 trong hồi quy.
15	Backprop tính gradient $\partial J/\partial\Theta^{(l)}$. $\delta^{(L)} = a^{(L)} - y$ (lỗi output). $\delta^{(l)} = (\Theta^{(l)})^T \delta^{(l+1)} \cdot g'(z^{(l)})$ (lan truyền ngược). $\frac{\partial J}{\partial \Theta_{ij}^{(l)}} = a_j^{(l)} \delta_i^{(l+1)}$. Cập nhật Θ bằng gradient descent.
16	XOR không phân tách tuyến tính được. ANN: có thể biểu diễn NAND $\rightarrow$ tổ hợp $\rightarrow$ XOR. ANN học các đặc trưng phi tuyến qua hidden layers, tạo ra biên phân lớp phức tạp. Logistic Regression chỉ tạo ra biên tuyến tính.
17	K-NN: tính khoảng cách từ X đến tất cả điểm trong D; chọn k điểm gần nhất; dự đoán lớp có nhiều phiếu nhất. Chọn k: $k \leq \sqrt{\|D\|}$. Ưu: đơn giản. Nhược: tốn bộ nhớ, tính khoảng cách chậm khi $\|D\|$ lớn (lazy learner).
18	Precision: trong số dự đoán là X, bao nhiêu phần đúng. Recall: trong số thực sự là X, bao nhiêu phần được phát hiện. Phát hiện gian lận: Recall quan trọng hơn (không bỏ sót gian lận thật). F-score cân bằng cả hai.
19	Holdout: chia 1 lần, kết quả phụ thuộc cách chia. K-fold: chia k lần, tính trung bình $\Rightarrow$ đáng tin cậy hơn vì ít phụ thuộc cách chia dữ liệu; tận dụng toàn bộ dữ liệu vừa train vừa test.
20	Logistic Reg: accuracy trung bình, nhanh, ổn định, interpretable. Decision Tree: tốt với dữ liệu danh mục, nhanh, interpretable tốt. Naive Bayes: nhanh, ít dữ liệu, giả định độc lập. ANN: accuracy cao nhất (phi tuyến), chậm, kém interpretable. K-NN: accuracy tốt, chậm với data lớn, tốn bộ nhớ.

Câu hỏi trắc nghiệm – Đáp án

Câu	ĐA	Câu	ĐA	Câu	ĐA	Câu	ĐA
1	B	11	B	21	B	31	C
2	B	12	B	22	B	32	B
3	A	13	B	23	B	33	B
4	B	14	B	24	B	34	B
5	B	15	B	25	C	35	B
6	C	16	B	26	B	36	B
7	B	17	B	27	B	37	B
8	B	18	B	28	C	38	B
9	B	19	B	29	C	39	B
10	B	20	B	30	B	40	B