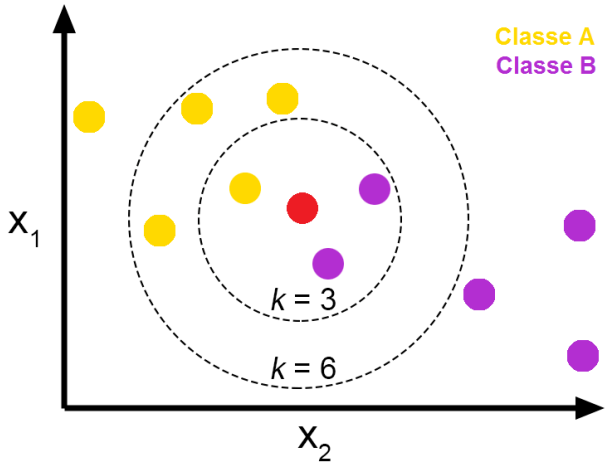
KNN Quizz

Hoàng-Nguyên Vũ

1. **Mô tả:**

   - **K-Nearest Neighbor** là một trong những thuật toán supervised-learning đơn giản nhất trong Machine Learning. Khi training, thuật toán này gần như không học gì từ dữ liệu training (hay còn được biết tới tên gọi là lazy learning), mọi tính toán được thực hiện khi mô hình cần dự đoán kết quả của dữ liệu mới cần dự đoán. K-nearest neighbor có thể áp dụng được vào cả hai loại của bài toán Supervised learning là Classification và Regression.



Hình 1: KNN distance

2. **Bài tập: Lưu ý:** Một số câu có trên 2 đáp án

Câu 1. Hai khoảng cách sau đây (Khoảng cách Euclide và Khoảng cách Manhattan) mà chúng ta thường sử dụng trong thuật toán K-NN. Những khoảng cách này nằm giữa hai điểm A(x_1,y_1) và B(x_2,y_2). Đáp án nào sau đây đúng với mô tả về biểu đồ bên dưới?

Hình 2: KNN distance

A) Bên trái là Khoảng cách Manhattan và bên phải là Khoảng cách Euclide
B) Bên trái là Khoảng cách Euclide và bên phải là Khoảng cách Manhattan
C) Cả bên trái và bên phải đều không phải là Khoảng cách Manhattan
D) Cả bên trái và bên phải đều không phải là Khoảng cách Euclide

Câu 2. Quan sát đoạn code sau:

```
# Paragraph A
#Create dataset
X, y = make_blobs(n_samples = 500, n_features = 2, centers = 4,
    cluster_std = 1.5, random_state = 4)
# Paragraph B
#Import libraries
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.datasets import make_blobs
from sklearn.neighbors import KNeighborsClassifier
from sklearn.model_selection import train_test_split
# Paragraph C
# Modeling
knn_with_k5 = KNeighborsClassifier(n_neighbors = 5)
knn_with_k1 = KNeighborsClassifier(n_neighbors=1)
knn_with_k5.fit(X_train, y_train)
knn_with_k1.fit(X_train, y_train)
y_pred_5 = knn_with_k5.predict(X_test)
y_pred_1 = knn_with_k1.predict(X_test)
# Paragraph D
#Split data
X_train, X_test, y_train, y_test = train_test_split(X, y,
    random_state = 0)

```

Thứ tự đúng của các đoạn trên là:
A) D - C - B - A
B) A - D - B - C
C) C - B - A - D
D) B - A - D - C

Câu 3. What is K-Nearest Neighbors (KNN) algorithm used for ?
A) Classification
B) Regression
C) Both a and b
D) None of the above

Câu 4. What happens if K is too small in KNN ?
A) Overfitting
B) Underfitting
C) No effect
D) Both a and b

Câu 5. What is the curse of dimensionality and how does it affect KNN ?

A) The curse of dimensionality refers to the increased computational complexity with higher dimensions, which can degrade KNN's performance.
B) The curse of dimensionality refers to the increased ease of classification with higher dimensions, which can improve KNN's performance.
C) The curse of dimensionality refers to the reduced computational complexity with higher dimensions, which can improve KNN's performance.
D) The curse of dimensionality does not affect KNN.

Câu 6. What is the computational complexity of KNN during prediction for each new query instance ?

A) O(n)
B) O(log n)
C) O(n log n)
D) O(n^2)

Câu 7. Which of the following is not a hyperparameter of the KNN algorithm ?

A) K
B) Distance metric
C) Learning rate
D) Weight function

Câu 8. Hoàn thiện đoạn code dưới đây để cài đặt hàm tính toán khoảng cách theo Euclidean:

```python
import math

def compute_distance(data_point1, data_point2):
    result = 0
    """
        Enter your code here
    """
    return math.sqrt(result)

```

A)

```python
for i in range(n):
    result += abs(data_point1[i] - data_point2[i])

```

B)

```python
for i in range(n):
    result += max(data_point1[i], data_point2[i])

```

C)

```python
for i in range(n):
    result += (data_point1[i] - data_point2[i])**2

```

D)

```
for i in range(n):
    result += (data_point1[i] - data_point2[i])**n

```

Câu 9. Đâu là nguyên do **chính** khiến chúng ta lại phải tính Confusion Matrix thay vì cách tính Accuracy truyền thống ?

A. Tại vì trong nghiên cứu khoa học, chúng ta cần phân tích sâu về các thông số của thuật toán, vì vậy confusion matrix sẽ giúp chúng ta đưa ra các quyết định chính xác hơn.

B. Tại vì khi tính Confusion Matrix bao gồm luôn cả Accuracy.

C. Tại vì trong data trong thực tế sẽ rất khó tính được Accuracy.

D. Tại vì không thể đánh giá chính xác độ hiệu quả của thuật toán bằng Accuracy với những tập dữ liệu bị biased (thiên vị).

Câu 10. Như bạn đã biết, KNN là một thuật toán dựa theo lazy learning, rằng thuật toán sẽ không có quy trình training data. Nhưng trong mã code của một thuật toán KNN lại có đoạn code dưới đây:

```
classifier = KNeighborsClassifier(n_neighbors = 4, p = 2, weights
    = customizeWeight)
classifier.fit(X_train, y_train)
y_pred = classifier.predict(X_test)

```

Cho biết rằng hàm fit() thường được dùng cho quá trình training. Vậy theo bạn, đâu là lý do chính xác nhất cho việc sử dụng câu lệnh này của tác giả ?

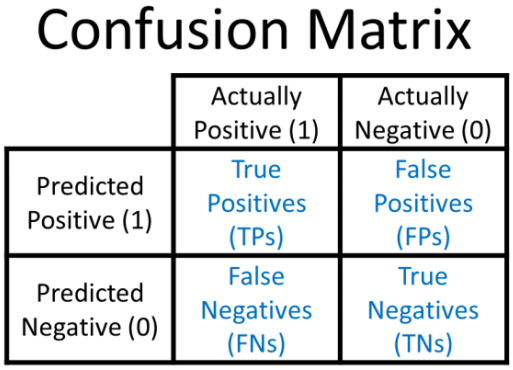
A. Bởi họ đã có sai lầm về mặt kiến thức, đoạn code không cần thiết có lệnh này mà vẫn chạy được.

B. Mặc dù không cần train, nhưng quá trình tính toán khoảng cách của thuật toán vẫn cần câu lệnh này hỗ trợ.

C. Bởi hàm này giúp chúng ta xây dựng cây K-D Tree hoặc Ball Tree hỗ trợ thuật toán tính khoảng cách.

D. Câu lệnh trên là để hỗ trợ chúng ta lựa chọn K phù hợp nhất cho thuật toán.

Câu 11. Có bài toán sau: "Vừa gà vừa chó, Bó lại cho tròn, Ba mươi sáu con, Một trăm chân chẵn". Đem tất cả ba mươi sáu con ở trên đem đi chụp hình, mỗi con 10 tấm, tổng là 360 hình cho một thuật toán Machine Learning training. Sau khi train, chúng ta đưa vào câu lệnh: “Hãy cho tôi ảnh con chó”, thì trả lại được 100 tấm hình, trong đó có 80 hình là chó và 20 hình là gà. **Hãy tính** các giá trị: **Precision, F1 Score, Recall và Accuracy** của mô hình.
Gợi ý: Hãy xem lại nội dung Confusion Matrix trước khi giải.



Hình 3: Nội dung Confusion Matrix

A) Recall: 0.667 - Precision: 0.571 - F1 Score: 0.666 - Accuracy: 0.777
B) Recall: 0.571 - Precision: 0.8 - F1 Score: 0.777 - Accuracy: 0.666
C) Recall: 0.8 - Precision: 0.571 - F1 Score: 0.777 - Accuracy: 0.666
D) Recall: 0.571 - Precision: 0.8 - F1 Score: 0.666 - Accuracy: 0.777

Câu 12. Những ý nào dưới đây là một vài mẹo cơ bản trong việc lựa chọn K trong thuật toán KNN mà không phải xét thêm trọng số ?
A) Lựa chọn k là số lẻ.
B) Lựa chọn căn bậc hai của k.
C) Lựa chọn k là số chẵn.
D) Lên Google tìm kiếm số k tối ưu cho mọi thuật toán.
E) Thử nghiệm nhiều lần để chọn k tối ưu.
F) Đánh giá dựa trên độ phức tạp của thuật toán.

Câu 13. How does the choice of K affect the bias-variance tradeoff in KNN ?
A) Increasing K increases bias and decreases variance
B) Increasing K decreases bias and increases variance
C) Increasing K increases both bias and variance
D) Increasing K decreases both bias and variance

Câu 14. Nhược điểm của KNN là gì ?
A) Nhạy cảm với nhiễu trong dữ liệu.
B) Tốn thời gian tính toán khi K lớn.
C) Cần lưu trữ tất cả các điểm dữ liệu.
D) Tất cả các nhược điểm trên.

Câu 15. KNN có thể được sử dụng với các thư viện machine learning nào ?
A) Scikit-learn
B) TensorFlow
C) PyTorch
D) Tất cả các thư viện trên

Câu 16. Tại sao thuật toán K-Nearest Neighbors (KNN) được coi là không hiệu quả đối với outliers ?

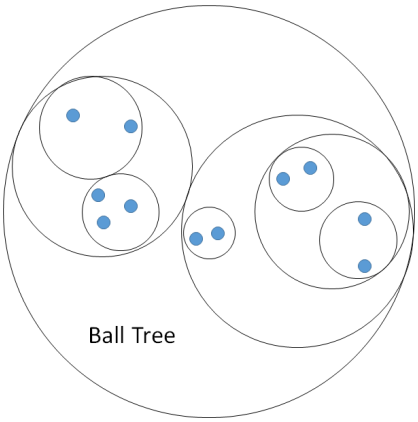
A. Bởi vì nó dựa vào một số lượng cố định các láng giềng để dự đoán, bất kể khoảng cách của chúng.

B. Bởi vì các ngoại lệ có thể ảnh hưởng đến dự đoán một cách không tương xứng nếu chúng nằm trong số những người lân cận gần nhất với điểm truy vấn.

C. Bởi vì nó sử dụng thước đo khoảng cách làm tăng trọng số của các ngoại lệ trong tập dữ liệu.

D. Bởi vì nó đòi hỏi một lượng lớn bộ nhớ, khiến nó nhạy cảm với việc bổ sung các giá trị ngoại lệ.

Câu 17. Hãy sắp xếp các bước dưới đây theo thứ tự phù hợp ý tưởng của cách xây dựng Ball-Tree:



Hình 4: Ball Tree

1. Lựa chọn 1 điểm bất kỳ trong training data
2. Tìm điểm xa nhất đối với điểm tìm được ở bước 2
3. Tìm điểm xa nhất đối với điểm vừa xác định
4. Tìm các điểm centroid đơn giản bằng cách tính trung bình giữa các điểm data trong từng nhóm data sau khi phân nhỏ
5. Xác định giá trị trung vị, lúc này chúng ta có thể dễ dàng chia tập data thành các phần nhỏ hơn
6. Lặp lại cho đến khi kết thúc tập dữ liệu train
7. Nối 2 đường thẳng giữa 2 điểm ở bước 2 và 3
8. Chiếu tất cả các điểm data còn lại lên đường ở bước 4

A) $1 \rightarrow 3 \rightarrow 2 \rightarrow 7 \rightarrow 8 \rightarrow 5 \rightarrow 6 \rightarrow 4$
B) $1 \rightarrow 3 \rightarrow 2 \rightarrow 7 \rightarrow 8 \rightarrow 6 \rightarrow 5 \rightarrow 4$
C) $1 \rightarrow 3 \rightarrow 2 \rightarrow 7 \rightarrow 6 \rightarrow 8 \rightarrow 5 \rightarrow 4$
D) $1 \rightarrow 3 \rightarrow 2 \rightarrow 7 \rightarrow 8 \rightarrow 5 \rightarrow 4 \rightarrow 6$

- Hết -