[DA_Project TIMA] Task 13_Xây dựng mô hình phân cụm

Xây dựng mô hình phân cụm (clustering) là một quá trình quan trọng trong học máy không giám sát (unsupervised learning)

Quy trình xây dựng mô hình phân cụm

Bước 1: Xác định vấn đề và chọn thuật toán phân cụm phù hợp.

Bước 2: Thu thập và chuẩn bị dữ liệu.

Bước 3: Xác định số lượng cụm (nếu cần).

Bước 4: Xây dựng mô hình phân cụm.

Bước 5: Đánh giá và phân tích kết quả phân cụm.

Bước 6: Tinh chỉnh mô hình.

Bước 7: Triển khai mô hình vào thực tế.

Quy trình chi tiết

Bước 1: Xác định vấn đề và lựa chọn thuật toán phân cụm

- **Xác định mục tiêu**: cần biết mục tiêu cụ thể của phân cụm là gì, ví dụ: phân nhóm khách hàng, phân loại tài liệu, phân tích dữ liệu khảo sát, v.v.
- **Chọn thuật toán phân cụm phù hợp**: Có nhiều thuật toán phân cụm khác nhau, bao gồm:
    - **K-Means**: Dễ hiểu và phổ biến, thích hợp cho dữ liệu dạng số và khi bạn biết số lượng cụm.

```
Bash
from sklearn.cluster import KMeans

# K-Means Clustering
kmeans = KMeans(n_clusters=2, random_state=0)
df['Cluster'] = kmeans.fit_predict(df)

print("\nK-Means Clustering:")
print(df)
```

- **DBSCAN**: Phù hợp cho các cụm có hình dạng bất kỳ và không cần biết số lượng cụm.

```
Bash
from sklearn.cluster import DBSCAN

# DBSCAN
dbscan = DBSCAN(eps=2, min_samples=2)
df['Cluster'] = dbscan.fit_predict(df)

print("\nDBSCAN Clustering:")
print(df)
```

- **Hierarchical Clustering**: Xây dựng cây phân cụm, phù hợp khi bạn muốn tìm kiếm cấu trúc phân cụm theo thứ bậc.

```
Bash
from scipy.cluster.hierarchy import dendrogram, linkage
import matplotlib.pyplot as plt

# Linkage cho Hierarchical Clustering
linked = linkage(df, method='ward')

# Dendrogram
plt.figure(figsize=(8, 6))
dendrogram(linked, orientation='top', distance_sort='descending', show_leaf_counts=True)
plt.title("Hierarchical Clustering Dendrogram")
plt.show()
```

- **Gaussian Mixture Models (GMM)**: Phân cụm dựa trên phân phối xác suất.

```
Bash
from sklearn.mixture import GaussianMixture

# Gaussian Mixture Models
gmm = GaussianMixture(n_components=2, random_state=0)
df['Cluster'] = gmm.fit_predict(df)

print("\nGaussian Mixture Models Clustering:")
print(df)
```

Bước 2: Thu thập và chuẩn bị dữ liệu

- **Thu thập dữ liệu**: Đảm bảo dữ liệu bạn sử dụng là đầy đủ và chất lượng. Có thể là dữ liệu số, văn bản, hình ảnh, v.v.
- **Tiền xử lý dữ liệu**:
  - **Xử lý giá trị thiếu**: Xử lý các giá trị bị thiếu bằng cách loại bỏ, thay thế, hoặc ước tính lại.
  - **Chuẩn hóa dữ liệu**: Nếu sử dụng thuật toán như K-Means, cần chuẩn hóa dữ liệu (ví dụ: sử dụng Min-Max hoặc Standard Scaler) để các đặc trưng có cùng thang đo.
  - **Chuyển đổi dữ liệu dạng văn bản (nếu có)**: Chuyển đổi văn bản thành các vector số (sử dụng TF-IDF, Word2Vec, v.v.) nếu dữ liệu của bạn là văn bản.

Bước 3: Xác định số lượng cụm (nếu cần)

- **K-Means** yêu cầu bạn phải biết trước số lượng cụm (k). Để xác định giá trị tốt cho k, bạn có thể sử dụng các kỹ thuật như:
  - **Phương pháp Elbow**: Vẽ đồ thị giữa số lượng cụm và tổng bình phương sai số (inertia). Tìm điểm "khuỷu tay" để chọn số lượng cụm tối ưu.

```
Bash
# Phương pháp Elbow
inertia = []
K = range(1, 10)  # Số cụm thử nghiệm

for k in K:
    kmeans = KMeans(n_clusters=k, random_state=0)
    kmeans.fit(df)
    inertia.append(kmeans.inertia_)

# Vẽ đồ thị Elbow
```

```
plt.figure(figsize=(8, 6))
plt.plot(K, inertia, 'bx-')
plt.xlabel('Số lượng cụm (k)')
plt.ylabel('Tổng bình phương sai số (Inertia)')
plt.title('Phương pháp Elbow')
plt.show()
```

- **Silhouette Score**: Đo lường mức độ tương đồng của mỗi điểm với cụm của nó và cụm gần nhất, giúp đánh giá chất lượng phân cụm.

```
Bash
# Silhouette Score
silhouette_scores = []

for k in range(2, 10):  # Silhouette cần ít nhất 2 cụm
    kmeans = KMeans(n_clusters=k, random_state=0)
    labels = kmeans.fit_predict(df)
    score = silhouette_score(df, labels)
    silhouette_scores.append(score)

# Vẽ đồ thị Silhouette Score
plt.figure(figsize=(8, 6))
plt.plot(range(2, 10), silhouette_scores, 'bx-')
plt.xlabel('Số lượng cụm (k)')
plt.ylabel('Silhouette Score')
plt.title('Silhouette Score để chọn số lượng cụm tối ưu')
plt.show()
```

Bước 4: Xây dựng mô hình phân cụm

- **Áp dụng thuật toán phân cụm**: Sau khi chuẩn bị dữ liệu và xác định số lượng cụm, bạn có thể sử dụng thuật toán phân cụm để phân chia dữ liệu thành các nhóm. Ví dụ, với K-Means:
  - Chọn số lượng cụm (k).
  - Khởi tạo các centroid (tâm của các cụm).
  - Lặp lại quá trình phân nhóm và cập nhật centroid cho đến khi hội tụ.

Bước 5: Đánh giá và phân tích kết quả

- **Đánh giá chất lượng phân cụm**: Có thể sử dụng các chỉ số như:
  - **Silhouette Score**: Đánh giá mức độ phân nhóm chính xác.

    Giá trị dao động từ -1 đến 1.

    1: Các cụm phân biệt rõ ràng.

    0: Các cụm chồng lấn.

    -1: Điểm bị phân cụm sai.

    Giá trị cao cho thấy phân cụm tốt.
  - **Davies-Bouldin Index**: Đo lường độ phân tán của các cụm.

    Giá trị càng nhỏ, cụm càng gọn và phân biệt tốt.

    Chỉ số đo độ phân tán trong mỗi cụm và khoảng cách giữa các cụm.

- **Kiểm tra phân cụm**: Quan sát các nhóm được phân chia để xem các nhóm có ý nghĩa không. Nếu dữ liệu có nhãn, có thể so sánh kết quả phân cụm với nhãn thực tế (nếu có).

```
Bash
# Vẽ phân cụm
plt.figure(figsize=(8, 6))
plt.scatter(df['Feature1'], df['Feature2'], c=labels, cmap='viridis', s=100)
plt.scatter(kmeans.cluster_centers_[:, 0], kmeans.cluster_centers_[:, 1],
        c='red', marker='X', s=200, label='Centroids')
plt.title("Kết quả phân cụm")
plt.xlabel("Feature1")
plt.ylabel("Feature2")
plt.legend()
plt.show()
```

Bước 6: Tinh chỉnh và cải thiện mô hình

- **Thử các thuật toán khác**: Nếu kết quả phân cụm không tốt, thử nghiệm với các thuật toán phân cụm khác như DBSCAN, Hierarchical Clustering hoặc Gaussian Mixture Models.
- **Điều chỉnh siêu tham số**: Thử thay đổi các tham số của thuật toán, ví dụ: số cụm (k), khoảng cách, v.v.
- **Xử lý lại dữ liệu**: Cân nhắc lại tiền xử lý dữ liệu, có thể thêm hoặc loại bỏ các đặc trưng.

Bước 7: Triển khai và sử dụng mô hình

- **Ứng dụng vào thực tế**: Sau khi xây dựng và đánh giá mô hình phân cụm, bạn có thể sử dụng kết quả phân cụm để ra quyết định trong các bài toán thực tế, chẳng hạn như phân tích khách hàng, đề xuất sản phẩm, phân loại tài liệu, v.v.

Áp dụng kiến thực xây dựng các mô hình sau

1. **Phân cụm theo số tiền đăng ký vay ban đầu (`SoTienDKVayBanDau`)**
   - **Ý nghĩa**: Phân nhóm khách hàng dựa trên mức độ đăng ký vay ban đầu để ngân hàng có thể đưa ra các chương trình khuyến mãi hoặc hỗ trợ tài chính phù hợp với từng nhóm.
2. **Phân cụm theo số tiền giải ngân (`TienGiaiNgan`)**
   - **Ý nghĩa**: Phân nhóm khách hàng theo số tiền thực tế được giải ngân để ngân hàng có thể điều chỉnh chiến lược giải ngân, tăng cường hoặc giảm bớt các khoản vay.
3. **Phân cụm theo số tiền còn lại (`SoTienConLai`)**
   - **Ý nghĩa**: Giúp xác định các nhóm khách hàng dựa trên số tiền còn lại của khoản vay để từ đó đưa ra chiến lược thu hồi nợ hoặc hỗ trợ tái cấp vốn.
4. **Phân cụm theo điểm tín dụng (`TS_CREDIT_SCORE_V2`)**
   - **Ý nghĩa**: Nhóm khách hàng theo điểm tín dụng để xác định khách hàng có khả năng trả nợ tốt, từ đó đưa ra các quyết định cho vay phù hợp.
5. **Phân cụm theo trạng thái khoản vay (`Trạng thái`)**
   - **Ý nghĩa**: Phân nhóm khách hàng theo trạng thái khoản vay (đang vay, đã trả, quá hạn) để có các biện pháp thu hồi nợ hoặc tiếp tục hỗ trợ cho các khoản vay chưa trả.
6. **Phân cụm theo giới tính (`Gender`)**
   - **Ý nghĩa**: Phân nhóm khách hàng theo giới tính để xây dựng các chiến lược tiếp thị và sản phẩm dịch vụ phù hợp với từng đối tượng khách hàng.

7. **Phân cụm theo độ tuổi (`Birthday`)**
   - **Ý nghĩa**: Phân nhóm khách hàng theo độ tuổi để cung cấp các dịch vụ ngân hàng phù hợp với từng độ tuổi, ví dụ, các sản phẩm vay dành cho người trẻ hoặc các khoản vay hưu trí cho người cao tuổi.

8. **Phân cụm theo thành phố hoặc khu vực sinh sống (`CityName`, `DistrictName`, `WardName`)**
   - **Ý nghĩa**: Phân nhóm khách hàng theo khu vực sinh sống để cung cấp các chương trình tín dụng phù hợp với đặc điểm kinh tế, dân cư của từng khu vực.

9. **Phân cụm theo hình thức cư trú (`Hình thức cư trú`)**
   - **Ý nghĩa**: Phân nhóm khách hàng theo hình thức cư trú (sở hữu nhà, thuê nhà) để ngân hàng có thể đưa ra các sản phẩm vay thế chấp hoặc các dịch vụ tín dụng khác.

10. **Phân cụm theo thời gian đã sống ở khu vực (`Thời gian đã sống`)**
    - **Ý nghĩa**: Phân nhóm khách hàng theo thời gian cư trú tại khu vực để đánh giá độ ổn định trong sinh sống và có chiến lược tín dụng phù hợp.

11. **Phân cụm theo nghề nghiệp (`JobName`)**
    - **Ý nghĩa**: Phân nhóm khách hàng theo nghề nghiệp để ngân hàng cung cấp các sản phẩm vay hoặc bảo hiểm phù hợp với từng nhóm khách hàng theo ngành nghề.

12. **Phân cụm theo mức thu nhập (`Salary`)**
    - **Ý nghĩa**: Phân nhóm khách hàng theo mức thu nhập để thiết kế các gói vay phù hợp với khả năng chi trả của từng nhóm khách hàng.

13. **Phân cụm theo thành phố hoặc khu vực làm việc (`CityCompany`, `DistrictNameCompany`)**
    - **Ý nghĩa**: Phân nhóm khách hàng theo nơi làm việc để xác định các nhóm có thu nhập ổn định, giúp cung cấp các sản phẩm tín dụng phù hợp.

14. **Phân cụm theo thu nhập thực nhận (`ReceiveYourIncomeSalary`)**
    - **Ý nghĩa**: Phân nhóm khách hàng theo thu nhập thực tế để xác định khả năng chi trả nợ của từng nhóm khách hàng.

15. **Phân cụm theo vị trí công việc (`DescriptionPositionJob`)**
    - **Ý nghĩa**: Phân nhóm khách hàng theo chức danh công việc để xây dựng các sản phẩm tín dụng hoặc lãi suất phù hợp với từng vị trí công việc (nhân viên, quản lý, giám đốc, v.v.).

16. **Phân cụm theo người thân (`RelativeFamilyName`)**
    - **Ý nghĩa**: Phân nhóm khách hàng theo mối quan hệ gia đình để đánh giá mức độ ổn định tài chính và khả năng hỗ trợ từ người thân trong trường hợp khách hàng gặp khó khăn tài chính.

17. **Phân cụm theo loại sản phẩm tín dụng (`ProductCreditName`)**
    - **Ý nghĩa**: Phân nhóm khách hàng theo loại sản phẩm tín dụng mà họ đăng ký để đưa ra các chiến lược quản lý nợ và phát triển sản phẩm phù hợp.

18. **Phân cụm theo phương thức thanh toán lãi suất (`InterestPaymentType`)**
    - **Ý nghĩa**: Phân nhóm khách hàng theo phương thức thanh toán lãi suất để ngân hàng có thể điều chỉnh chính sách tín dụng hoặc lãi suất cho từng nhóm.

19. **Phân cụm theo lịch sử nợ xấu (`HasBadDebt`)**
    - **Ý nghĩa**: Phân nhóm khách hàng theo việc có nợ xấu hay không để ngân hàng đưa ra các chiến lược thu hồi nợ hoặc cung cấp các gói vay tín chấp có yêu cầu cao hơn đối với khách hàng có nợ xấu.

20. **Phân cụm theo lịch sử thanh toán trễ (`HasLatePayment`)**

- **Ý nghĩa**: Phân nhóm khách hàng theo lịch sử thanh toán trễ để giúp ngân hàng xác định các khách hàng có nguy cơ không trả nợ đúng hạn, từ đó có biện pháp giám sát và quản lý nợ hiệu quả.