BÀI THỰC HÀNH

IT6075

KHAI THÁC DỮ LIỆU VÀ ỨNG DỤNG

BÀI 1. KHAI PHÁ LUẬT KẾT HỢP

1. **Dữ liệu**
   - DataSet1.txt
   - DataSet2.txt
   - DataSet3.txt
   - DataSet4.txt
   - DataSet5.txt
   - GroceryStoreDataSet.csv
   - store_data.csv
   - Dữ liệu sinh viên tự thu thập: ít nhất 01 file dạng txt hoặc csv hoặc cả hai.
2. **Nhiệm vụ**

Phần 1. Cài đặt môi trường: Sinh viên khởi tạo một thư mục (sẽ dùng chứa toàn bộ các bài thực hành của học phần), tạo pycharm project trong thư mục đó, cài đặt các gói cần thiết (có thể tham khảo thư mục bài thực hành mẫu).

Phần 2: Kiểm tra dữ liệu: Sinh viên thu thập, kiểm tra nội dung từng file dữ liệu.

Phần 3. Cài đặt thuật toán Apriori bằng gói apriori_python

- Đọc dữ liệu từ file
- Gọi thuật toán để tính các tập phổ biến và các luật trong tập dữ liệu.
- In kết quả.

Phần 4. Cài đặt thuật toán Apriori bằng gói apriori_python với dữ liệu bất kỳ

- Sinh viên tự thu thập dữ liệu tùy ý.
- Tiền xử lý dữ liệu (nếu cần).
- Thực hiện chạy thuật toán Apriori trên dữ liệu vừa thu thập và báo cáo kết quả

Phần 5. Cài đặt thuật toán APRIORI

- Xem code mẫu
- Thực hiện cài đặt theo code mẫu, báo cáo kết quả.

BÀI 2. PHÂN LỚP DỮ LIỆU VỚI CÂY QUYẾT ĐỊNH VÀ NAÏVE BAYES

1. Dữ liệu

- diabetes.csv (Cây quyết định)
- adult.csv (Cây quyết định)
- Loandata.csv (Naïve Bayes)
- Loan Payments data.csv (Naive Bayes)
- Dữ liệu sinh viên tự thu thập: ít nhất 01 file dạng txt hoặc csv hoặc cả hai cho mỗi thuật toán.

2. Nhiệm vụ

Phần 1: Kiểm tra dữ liệu: Sinh viên thu thập, kiểm tra nội dung từng file dữ liệu.

Phần 2. Cài đặt thuật toán phân lớp dữ liệu bằng cây quyết định bằng DecisionTreeClassifier của sklearn.tree:

- Đọc và xử lý, hiển thị dữ liệu từ file
- Tạo các tệp Train và Test
- Tạo mô hình phân lớp dựa trên cây quyết định.
- In kết quả: gồm các độ đo và cây quyết định.

Phần 3. Cài đặt thuật toán phân lớp bằng mô hình Naïve Bayes từ gói sklearn.naive_bayes trên dữ liệu tự sinh

- Sinh và hiển thị bộ dữ liệu
- Tạo các tệp Train và Test
- Tạo mô hình phân lớp bằng Naïve Bayes.
- In kết quả: gồm các độ đo và confusion matrix.

Phần 4. Cài đặt thuật toán phân lớp bằng mô hình Naïve Bayes từ gói sklearn.naive_bayes trên dữ liệu từ file (Loandata.csv,…)

- Đọc và hiển thị dữ liệu
- Tạo các tệp Train và Test
- Tạo mô hình phân lớp bằng Naïve Bayes.
- In kết quả: gồm các độ đo và confusion matrix.

Phần 5. Cài đặt thuật toán phân lớp bằng mô hình Naïve Bayes từ gói sklearn.naive_bayes trên dữ liệu từ file do sinh viên thu thập:

- Đọc và hiển thị dữ liệu
- Tạo các tệp Train và Test
- Tạo mô hình phân lớp bằng Naïve Bayes.
- In kết quả: gồm các độ đo và confusion matrix.

BÀI 3. PHÂN LỚP DỮ LIỆU VỚI KNN VÀ SVM

1. Dữ liệu

- teleCust1000t.csv (KNN)
- iris.csv (SVM)
- iris_svm_light.txt (SVM)
- Dữ liệu sinh viên tự thu thập: ít nhất 01 file dạng txt hoặc csv hoặc cả hai cho mỗi thuật toán.

2. Nhiệm vụ

Phần 1: Kiểm tra dữ liệu: Sinh viên thu thập, kiểm tra nội dung từng file dữ liệu.

Phần 2. Cài đặt thuật toán phân lớp dữ liệu k-láng giềng gần nhất bằng gói KneighborsClassifier của sklearn.neighbors:

- Đọc và xử lý, hiển thị dữ liệu từ file
- Chuẩn hóa dữ liệu bằng z-score.
- Tạo các tệp Train và Test
- Tạo mô hình phân lớp bằng KNN.
- In kết quả: gồm các độ đo.

Phần 3. Cài đặt thuật toán phân lớp bằng mô hình SVM từ gói svm của sklearn trên dữ liệu lấy từ datasets của sklearn (file mẫu SVM.py)

- Tải về và hiển thị bộ dữ liệu Breast_cancer
- Tạo các tệp Train và Test
- Tạo mô hình phân lớp bằng SVM.
- In kết quả: gồm các độ đo acc, precision, recall, f1,...

Phần 4. Cài đặt thuật toán phân lớp bằng mô hình SVM từ gói svm của sklearn trên dữ liệu lấy từ file, định dạng dữ liệu là file for svm light (xem file mẫu SVM_data_from_file.py)

- Đọc và hiển thị dữ liệu
- Tạo các tệp Train và Test
- Tạo mô hình phân lớp bằng SVM.
- In kết quả: gồm các độ đo acc, precision, recall, f1,...

Phần 5. Cài đặt thuật toán phân lớp bằng mô hình SVM từ gói svm của sklearn trên dữ liệu lấy từ file, dữ liệu đọc lên được lưu dưới dạng numpy array (xem file mẫu SVM_numpy.py)

- Đọc và hiển thị dữ liệu
- Tạo các tệp Train và Test
- Tạo mô hình phân lớp bằng SVM.
- In kết quả: gồm các độ đo acc, precision, recall, f1,...

Phần 6. Cài đặt thuật toán phân lớp bằng mô hình SVM từ gói svm của sklearn trên dữ liệu lấy từ file, dữ liệu đọc lên được lưu dưới dạng pandas frame (xem file mẫu SVM_pandas.py)

- Đọc và hiển thị dữ liệu
- Tạo các tệp Train và Test
- Tạo mô hình phân lớp bằng SVM.
- In kết quả: gồm các độ đo acc, precision, recall, f1,...

Phần 7. Cài đặt thuật toán phân lớp bằng mô hình SVM từ gói svm của sklearn trên dữ liệu lấy từ file do sinh viên thu thập (file dữ liệu có định dạng bất kỳ) và dữ liệu đọc lên được lưu dưới định dạng tùy ý (numpy, pandas,…).

- Đọc và hiển thị dữ liệu
- Tạo các tệp Train và Test
- Tạo mô hình phân lớp bằng SVM.
- In kết quả: gồm các độ đo acc, precision, recall, f1,...

BÀI 4. PHÂN CỤM DỮ LIỆU VỚI KMEANS

1. Dữ liệu

- iris.csv (kmeans)
- Dữ liệu sinh viên tự thu thập: ít nhất 01 file dạng txt hoặc csv hoặc cả hai cho mỗi thuật toán.

2. Nhiệm vụ

Phần 1: Kiểm tra dữ liệu: Sinh viên thu thập, kiểm tra nội dung từng file dữ liệu.

Phần 2. Cài đặt thuật toán phân nhóm dữ liệu k-means bằng gói KMeans của sklearn.cluster với dữ liệu hai chiều, được lưu trọng hai mảng numpy hard-code.

- Sinh dữ liệu và hiển thị dữ liệu
- Tạo mô hình phân nhóm kmeans.
- Hiển thị kết quả phân cụm trên một đồ thị scatter.

Phần 3. Cài đặt thuật toán phân nhóm dữ liệu k-means bằng gói KMeans của sklearn.cluster với dữ liệu đọc từ file:

- Đọc dữ liệu từ file và lưu trữ trong các mảng numpy.
- Xử lý dữ liệu và hiển thị dữ liệu.
- Tạo mô hình phân nhóm bằng k-means.
- In kết quả: gồm các độ đo phù hợp với mô hình kmeans như: Rand index, silhouette_score,…

Phần 4. Cài đặt thuật toán phân nhóm k-means từ gói KMeans của sklearn.cluster trên dữ liệu lấy từ file do sinh viên thu thập (file dữ liệu có định dạng bất kỳ) và dữ liệu đọc lên được lưu dưới định dạng tùy ý (numpy, pandas,…).

- Đọc, xử lý và hiển thị dữ liệu
- Tạo mô hình phân nhóm k-means.
- In kết quả: gồm các độ đo phù hợp.