KHAI THÁC DỮ LIỆU

Nguyễn Xuân Việt Đức - 22280012

Bài tập lý thuyết - Lần 2

Phân biệt K-Mean, K-Medoids và Kernel K-Mean

1. Điểm giống nhau

- **Mục tiêu chung:** Cả ba thuật toán đều nhằm mục đích phân cụm dữ liệu thành các nhóm (clusters) dựa trên sự tương đồng giữa các điểm dữ liệu.
- **Sử dụng số cụm k:** Cả ba thuật toán đều yêu cầu người dùng xác định trước số lượng cụm k.
- **Dựa trên tối ưu hóa:** Tất cả các thuật toán đều cố gắng giảm thiểu một hàm mất mát (loss function) để tối ưu hóa sự phân cụm.
- **Thuật toán lặp:** Các thuật toán đều sử dụng quy trình lặp để cải thiện kết quả phân cụm qua từng bước.

2. Điểm khác nhau

- **Trung tâm cụm (Centroid):**
  - **K-Mean:** Sử dụng trung bình (mean) của các điểm trong cụm làm trung tâm cụm.
  - **K-Medoids:** Sử dụng một điểm dữ liệu thực tế trong cụm (medoid) làm trung tâm cụm.
  - **Kernel K-Mean:** Sử dụng trung bình trong không gian đặc trưng kernel.
- **Hàm mất mát (Loss):**
  - **K-Mean:** Cực tiểu hóa tổng bình phương khoảng cách Euclidean từ các điểm đến centroid.
  - **K-Medoids:** Cực tiểu hóa tổng khoảng cách (thường là Manhattan) từ các điểm đến medoid.
  - **Kernel K-Mean:** Dựa trên hàm kernel để tính khoảng cách phi tuyến.

- **Xử lý ngoại lệ (Outliers):**
  - **K-Mean:** Nhạy cảm với ngoại lệ, vì trung bình bị ảnh hưởng bởi các giá trị lệch.
  - **K-Medoids:** Ít nhạy cảm với ngoại lệ, vì medoid là một điểm thực tế và không bị ảnh hưởng bởi giá trị lệch.
  - **Kernel K-Mean:** Có thể giảm ảnh hưởng của ngoại lệ nhờ không gian kernel.
- **Không gian dữ liệu:**
  - **K-Mean:** Hoạt động trong không gian Euclidean tuyến tính.
  - **K-Medoids:** Hoạt động trong không gian Euclidean, nhưng không yêu cầu tuyến tính.
  - **Kernel K-Mean:** Hoạt động trong không gian phi tuyến nhờ sử dụng kernel.
- **Độ phức tạp tính toán:**
  - **K-Mean:** Thấp, vì chỉ cần tính trung bình và khoảng cách Euclidean.
  - **K-Medoids:** Cao hơn K-Mean, vì phải tính khoảng cách cho từng cặp điểm và tìm medoid tối ưu.
  - **Kernel K-Mean:** Cao nhất, vì phải tính toán ma trận kernel và các phép biến đổi trong không gian đặc trưng.
- **Ứng dụng:**
  - **K-Mean:** Tốt cho các bài toán có dữ liệu tuyến tính, phân cụm đơn giản.
  - **K-Medoids:** Tốt cho dữ liệu có nhiễu hoặc ngoại lệ, khi cần cụm trung tâm đại diện thực tế.
  - **Kernel K-Mean:** Tốt cho dữ liệu phức tạp, phi tuyến như hình dạng phi chuẩn.

3. Tóm tắt

- **K-Mean:** Đơn giản, hiệu quả, nhưng nhạy cảm với ngoại lệ và chỉ phù hợp với dữ liệu tuyến tính.
- **K-Medoids:** Phức tạp hơn, nhưng ít nhạy cảm với ngoại lệ và đảm bảo cụm trung tâm là một điểm thực tế.
- **Kernel K-Mean:** Linh hoạt nhất nhờ sử dụng kernel, phù hợp với dữ liệu phi tuyến, nhưng đòi hỏi chi phí tính toán cao.