Lưu đồ thuật toán phân cụm K-means:

1. Khởi tạo

Gán biến covergedCentroids = 0

Iterations = 0

maxIterations = 100

Tạo 2 mảng chưa centroids và giá trị wcss

centroids =[]

new\_centroids = []

wcss\_values = []

1. Khởi tạo các centroids:

Sử dụng takeSample để chọn ngẫu nhiên **k** điểm dữ liệu làm điểm trung tâm ban đầu. Mà các các điểm dữ liệu **k** không giống nhau. Và lưu vào dánh sách “tmp”

Dùng vòng lặp **for** để lặp qua từng centroids. Chuyển centroids thành số thực để đảm bảo có thể sử dụng tính toán.

Ở mỗi trọng tâm được biểu diễn ở dạng danh sách chứ chỉ số và danh sách giá trị của trọng tâm và thêm vào danh sách “centroids”.

1. Thuật toán K-Means:

Bắt đầu một vòng lặp While nội bộ cho đến khi hội tụ hoặc quá số lần lặp (**maxIterations**)

Sử dụng **map** để gán mỗi điểm trong RDD vào trung tâm gần nhất và lưu vào bộ nhớ cache.

Sử dụng **Reduce**  tính tổng các điểm dữ liệu trong mỗi trọng tâm

Trong mỗi lần lặp, dữ liệu được ánh xạ vào các trọng tâm và sau đó được giảm tổng hợp để tính toán các trọng tâm mới.

1. Kiểm tra sự hội tụ:

Kiểm tra sự hội tụ bằng cách sử dụng khoảng cách Euclidean. Tính khoảng cách giữa trọng tâm cũ và trọng tâm mới

Nếu khoảng cách Euclidean giữa các trung tâm cũ và mới tương ứng nhỏ hơn ngưỡng threshold 0.0001 đối với tất cả các cụm thì coi như là hội tụ

1. Cập nhật Trung tâm:

Cập nhật trung tâm với new\_centroids. Và kiểm tra điều kiện dừng

1. Tính toán Chi phí:

Tính toán chi phí của phân cụm hiện tại bằng cách sử dụng hệ số WCSS.

Nếu chi phí hiện tại nhỏ hơn chi phí trước đó, cập nhật cost và result\_centers.

1. Dọn dẹp bộ nhớ cache:

Hủy bỏ bộ nhớ cache của các cụm để giải phóng tài nguyên.

1. Trả về kết quả:

Trả về giá trị WCSS, danh sách các cụm, và danh sách các trọng tâm