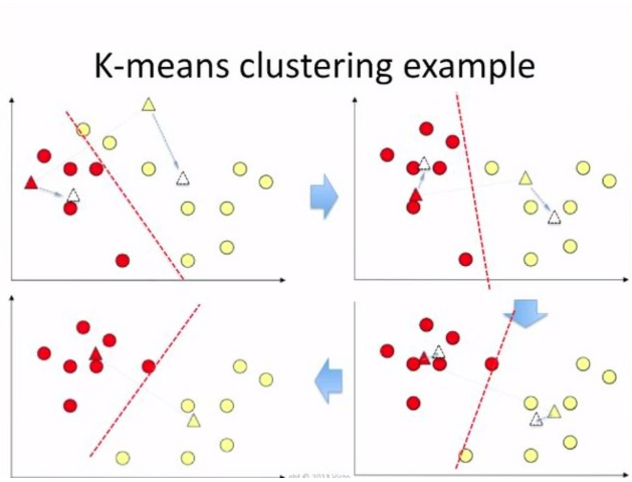
Kmeans talk – Unsupervised learning- you do not know the number of clusters or cluster membership

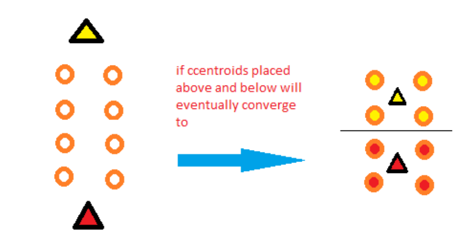
First: Kmeans is not KNN
K-means is a clustering algorithm that tries to partition a set of points into K sets (clusters) such that the points in each cluster tend to near each other. It is unsupervised because the points have no external classification.
K-nearest neighbors is a supervised classification algorithm that classifies a point, by combining the the known classes of the K nearest points.

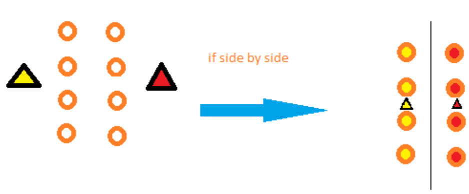
Problems
K-means is vulnerable to outliers
K-means works best with spherical clusters
K-means requires you to pick the number of clusters you have before you run the algorithm (how to know?)
K-means is not guaranteed to produce the same result every run (it depends on initial cluster centers)

Algorithm
1. Randomly pick *k* centroids from sample points as initial cluster centers
2. Assign each sample to the nearest centroid
3. Move the centroids to the center of the samples assigned to it
4. Repeat steps 2 and 3 until cluster membership stops changing



It is not guaranteed to converge to same solution every time





sklearn

use kmeans++ to initialize (picks the k points in the data furthest from each other)

Inertia (For every sample, add the Sum of squared distances to it's cluster center). Proxy for choosing correct number of clusters

How to choose the ideal number of clusters

the elbow method

calculate the inertia for every value of K, plot these inertias, look for point where K starts to increase rapidly (the elbow)

the Silhouette score

Plots how tightly grouped the samples in the clusters are.

How to calculate the silhouette coefficient of a single sample in a dataset. Do this for all samples then plot per cluster.

1. ai – cluster cohesion- average distance between a sample and all other points in the cluster

2. bi – seperation- average distance between sample and all other points in nearest cluster

Silhouette score (Si) for point= (bi-ai)/max(bi,ai)

$-1<Si<1$
=0 if ai=bi then point is right on edge of being misclassified
=1 ideal $bi>>ai$ very far away from nearest clustering
<0 then point probably misclassified

Do this for every point, then plot the scores per point and per cluster