k-means clustering explained (StatQuest)

<https://youtu.be/4b5d3muPQmA>

1. Set k = the number of clusters you want to id
2. Randomly select k distinct data points to correspond with a cluster
3. Measure the distance from the first point in the data set to each randomly selected data point from step 2
4. Assign the point to the cluster closest to it
5. Keep doing this for each point in the data set
6. Now you’ve got every point assigned to a cluster, calculate the mean of each cluster
7. Now do it again with the means you just determined, and repeat until the current cluster = previous cluster
8. You can cheat by eyeballing the data, picking a value for k which you think lines up with what kind of clusters are actually there, and manually setting the initially selected data points.

k-means clustering explained (Computerphile)

<https://youtu.be/yR7k19YBqiw>

agglomerative clustering lecture

<https://youtu.be/XJ3194AmH40>

hierarchical clustering explained (StatQuest)

<https://youtu.be/7xHsRkOdVwo>

Determine which two observations in the data set are most similar, cluster them

Do this iteratively, treating each cluster as a single observation once generated, allowing observations to be added to existing clusters or form new ones

Keep doing this until you end up with one cluster

dendrogram explained

<https://youtu.be/ijUMKMC4f9I>

The dendrogram will illustrate the results of a hierarchical clustering process by showing links between observations whose lengths differ according to how late in the algorithm they were formed

DBSCAN (StatQuest)

<https://youtu.be/RDZUdRSDOok>

Handles nested clusters

1. Draw a circle around each point
2. Count the number of other points overlapped by this circle
3. Core point = point close to at least n other points, n can be chosen
4. If you’re not a core point you’re a non-core point
5. Do the circle drawing thing again on the core points to form clusters
6. Once you’ve formed the clusters, do the circle thing to incorporate non-core points into a cluster
7. Non-core points cannot extend a cluster
8. If you don’t get picked up by any cluster, you’re an outlier and are considered irrelevant (probably)

Linear Discriminant Analysis (StatQuest)

<https://youtu.be/azXCzI57Yfc>

like PCA, but focuses on maximizing the separability among known categories

axis generation criteria = maximize distance between means between categories, minimize variation within each category ((mu - mu)^2) / (s^2 + s^2)

ideally (mu - mu)^2 would be very large and (s^2 – s^2) would be very small

important to optimize both scatter and mean distance into to get good separation

PCA in Python (StatQuest)

<https://www.youtube.com/watch?v=Lsue2gEM9D0>

Guide for PCA in Python

<https://towardsdatascience.com/pca-using-python-scikit-learn-e653f8989e60>

Guide for MDS in Python

<https://stackabuse.com/guide-to-multidimensional-scaling-in-python-with-scikit-learn/>

Guide for T-SNE in Python

<https://towardsdatascience.com/visualising-high-dimensional-datasets-using-pca-and-t-sne-in-python-8ef87e7915b>