Unsupervised: PCA and Clustering

Today we will have some math

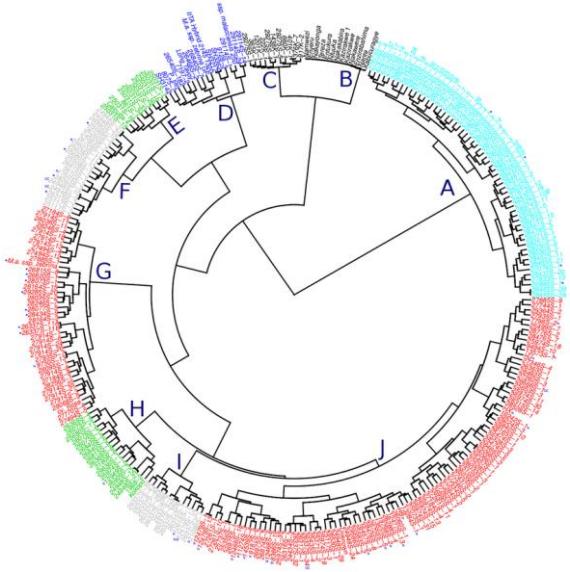
Sarid Research Services

Materials by Adi Sarid https://adisarid.github.io and http://www.sarid-ins.com

What is clustering?

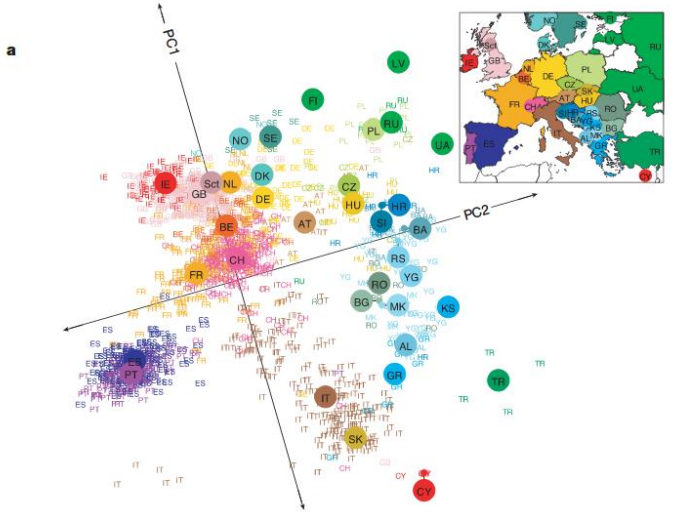
- In data science, problems can be classified into
  - Supervised learning (use x_1, x_2,… to predict y)
  - Unsupervised learning (don't have a y target, need to understand the data)
- Clustering is an unsupervised learning method with which we can learn:
  - What groups exist in the data?
  - What are "similar" observations?

Example - population genetics



Dendrogram showing how different species relate to one another

https://www.researchgate.net/figure/Dendrogram-showing-the-genetic-diversity-of-the-genomic-selection-training-population_fig2_317632929



PCA on DNA markers separates European populations

https://stats.stackexchange.com/questions/8777/in-genome-wide-association-studies-what-are-principal-components

KMeans Clustering

- KMeans divides observations in an n-dimensional space by distance from one another
  - Minimize the within variance
  - Maximize the between-group variance
- For example, find partition (C_1,…, C_k) to reach: $\min_{C_1,\ldots,C_k} \sum_{k=1}^{K} W(C_k)$
- With $W(C_k)$ defined as:

$$W(C_k) = \frac{1}{|C_k|} \sum_{i,i' \in C_k} \sum_{j=1}^{p} (x_{ij} - x_{i'j})^2$$

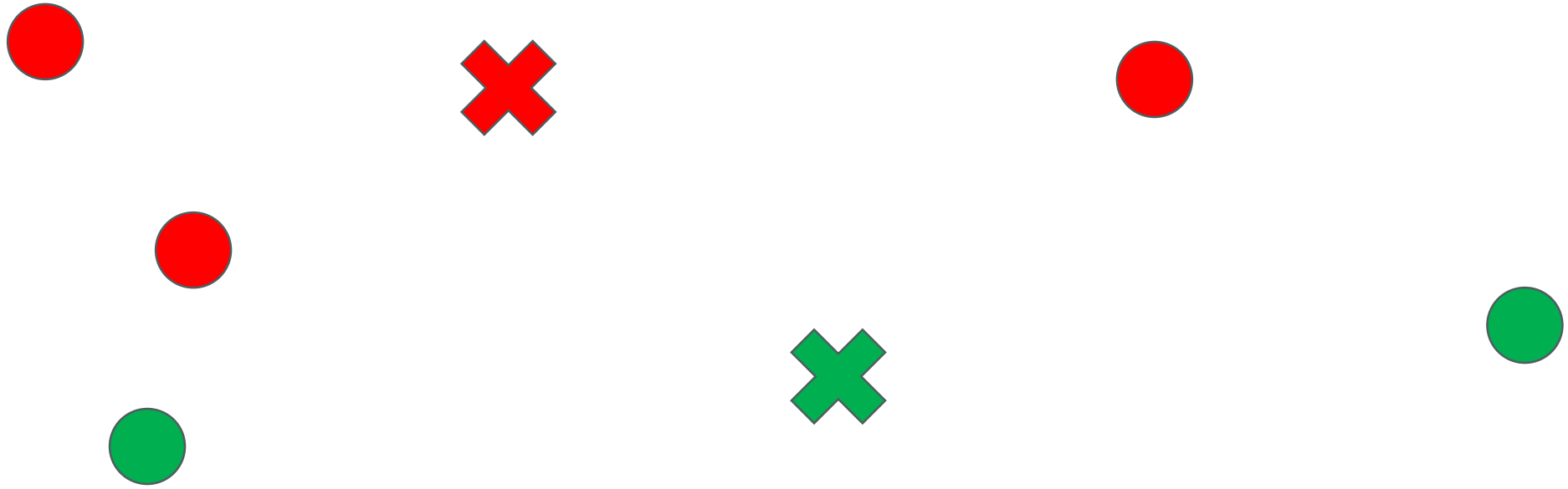
What is this distance?

KMeans Clustering - explained

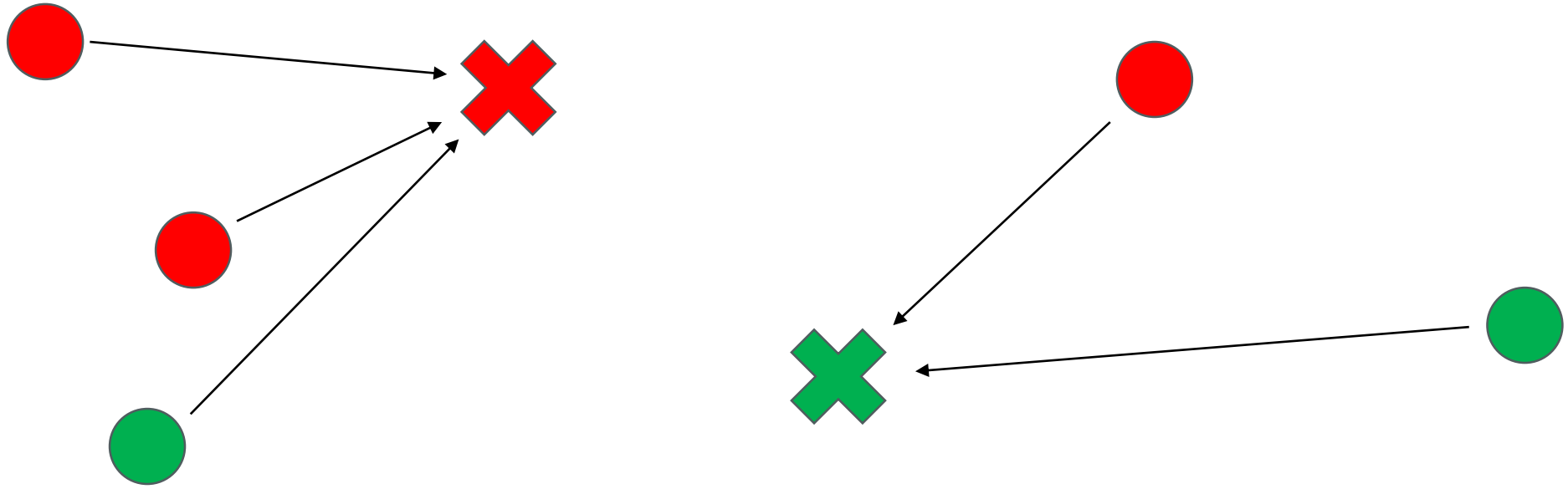
- We need to find a partition to K clusters, but how do we determine K?
  - Sometimes it's in the work's goals, sometimes trial and error
- How does the algorithm work?
  - Randomly assign a cluster to each point 1,…, n
  - Repeat the following until no re-assignments are made
    - Calculate each cluster's **centroid** (central mass of the cluster, e.g., average position)
    - Change the observation's classification according to centroid
    - Update centroids
    - Return to previous step

K Means - illustration
(initial classification, randomly)

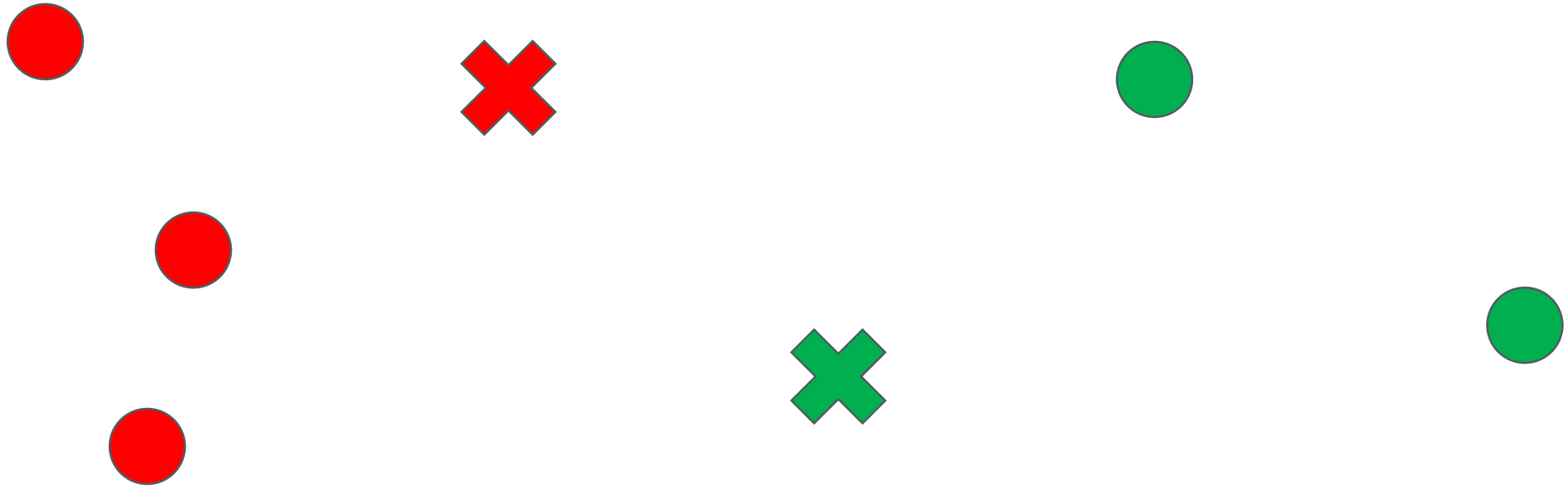
K Means - illustration
(set centroids)



K Means - illustration
(find closest centroid)



K Means - illustration
(update classification, update centroids)



K Means - illustration
(no more updates - algorithm concludes)

Example with R

- K Means clustering in R is performed using *kmeans*
- In the simplest form:

```
kmeans_result <- kmeans(dataset, centers = k)
```

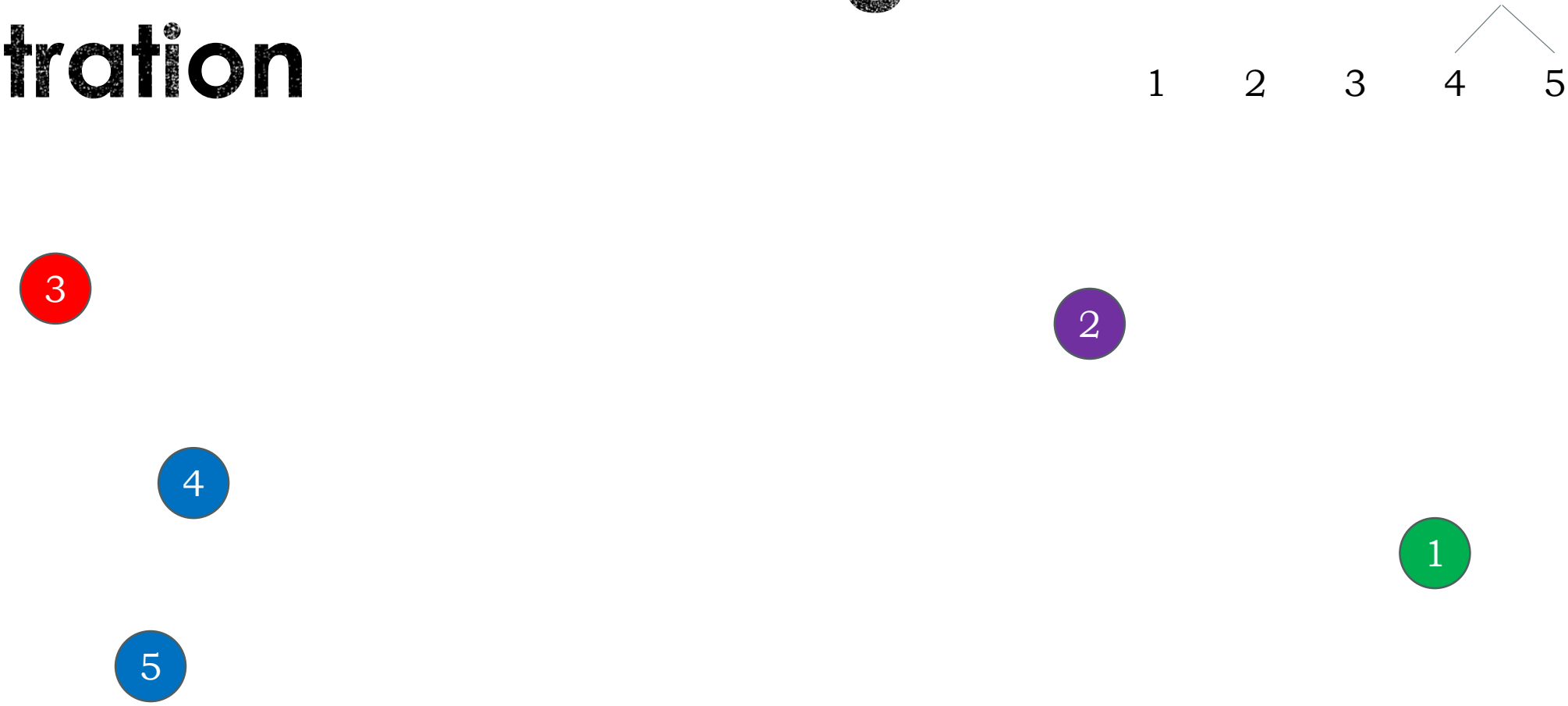
- Additional options to control the algorithm method
- Open 05-Clustering.Rmd

Hierarchical clustering

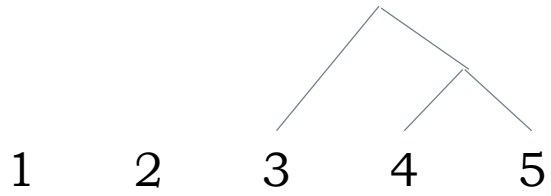
- What happens when you don't know *k* or want a better "mapping"?
- Instead of **assign->update-re-assign** look for **the "next merge"**
- Algorithm
  - Start with *n* observations and a distance function between each observation
  - In total there are *n(n-1)/2* such pairs, every pair is now a "cluster"
  - While the number of current clusters > 1:
    - Check all distances from all current clusters to each other
    - Choose the two clusters which are the closest and merge them
    - The number of clusters is decreased by 1
    - Re-compute the cluster distances
- Bottom-up approach

Hierarchical clustering - illustration

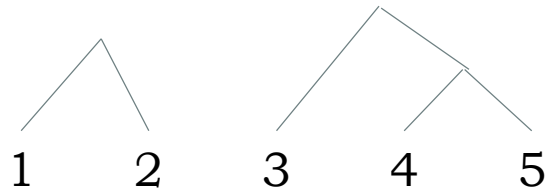
Hierarchical clustering - illustration



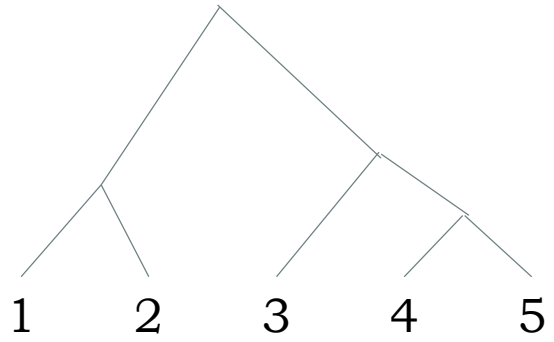
Hierarchical clustering - illustration



Hierarchical clustering - illustration



Hierarchical clustering - illustration



tree height is determined by distance function

What is the "linkage" (distance) function?

- To determine the "closest" cluster we can either use:
  - Central mass between clusters (centroid as in the K Means)
  - The two closest points (single linkage)
  - The average distance (average linkage)
  - The maximum distance (complete linkage)
- The function is specified in the command argument *method*

```
hclust_result <- hclust(dataset, method = "euclidean")
```

PCA (Principle component analysis)

- PCA is a computation which utilizes linear algebra to reduce the dimension of the data
- Question: why would we want to reduce the dimension of the data? (the number of features)