LIVE ONLINE TRAINING

Business Analytics With Python Bootcamp

Week 4: Diagnostic Business Analytics with Python

Agenda

1. **Recap & Intro**
2. **Introduction to Diagnostic Analytics (30 minutes)**
3. **Segmentation (60 minutes)**
   - Interactive lab: Calculate RFM Values
   - Break
4. **Clustering (60 minutes)**
   - Interactive labs: Develop and Interpret a Hierarchical & k-Means Clustering
   - Break
5. **Rule mining (60 minutes)**
   - Interactive lab: Perform a Market Basket Analysis
   - Outlook for next week

Recap

Bootcamp overview

Learning goals:

- Derive actionable insights from data
- Perform exploratory data analysis and create meaningful visualizations
- Use value-based analysis techniques and create association rules for effective decision support
- Apply clustering techniques to discover segments in your data, e.g. different customer groups
- Build predictive models for regression and classification tasks
- Understand the key criteria for evaluating the performance of a predictive model
- Suggest specific business actions that will lead to better results

Bootcamp overview

Last week

Week 1 Jan 31	Week 2 Feb 7	Week 3 Feb 14	Week 4 Feb 21	Week 5 Feb 28	Week 6 Mar 7
Business Analytics Fundamentals	**Data Analysis With Python Fundamentals**	**Descriptive Business Analytics With Python**	**Diagnostic Business Analytics With Python**	**Predictive Business Analytics With Python**	**Prescriptive Business Analytics With Python**
Goal: Get the theoretical foundations for doing business analytics	**Goal:** Get the practical foundations for using Python for data analysis	**Goal:** Learn how to explore & describe your data (What happened?)	**Goal**: Learn how to get insights from your data (Why did it happen?)	**Goal:** Learn how to predict the future, given historical data (What will happen?)	**Goal:** Suggest better actions based on data (What to do?)

NOTE: With today's registration, you'll be signed up for all six sessions. Although you can attend any of the sessions individually, it's recommended participating in all six weeks.

Quiz time!

As a business analyst, why should I care about SQL?

A) SQL is an outdated technology that is no longer in use.

B) SQL lets you access data from almost any database, even via Python.

C) SQL is a programming language used only for cloud data warehouses.

D) Learning SQL is only necessary for database administrators.

What’s considered best practice for fetching data from a database?

A) I fetch all data of interest using the SELECT * FROM <TABLE> command and do the rest in Python.

B) I extract only a few columns of interest using the SELECT <COLUMN1>, <COLUMN2> FROM <TABLE> command and do the rest in Python.

C) I write a Python script to extract the data from the database and do all the data processing in Python.

D) I try to do as much data processing as possible on the database using SQL and do the rest in Python.

Which of these answers is true for a left join?

A) It will return the same number of rows for the left table

B) It will return the same number of rows for the right table

C) It will return only instances which were in the left and right table

D) It will return all instances from both tables

What should I do with outliers in my dataset?

A) Keep them

B) Remove them

C) Replace them

D) It depends

When should you do an Exploratory Data Analysis (EDA)

A) Every time you work with a new data set

B) Only when you do predictive modeling

C) When you have enough time

D) EDA is only done in research

Bootcamp overview

This Week

Week 1 Jan 31	Week 2 Feb 7	Week 3 Feb 14	Week 4 Feb 21	Week 5 Feb 28	Week 6 Mar 7
Business Analytics Fundamentals	**Data Analysis With Python Fundamentals**	**Descriptive Business Analytics With Python**	**Diagnostic Business Analytics With Python**	**Predictive Business Analytics With Python**	**Prescriptive Business Analytics With Python**
Goal: Get the theoretical foundations for doing business analytics	**Goal:** Get the practical foundations for using Python for data analysis	**Goal:** Learn how to explore & describe your data (What happened?)	**Goal**: Learn how to get insights from your data (Why did it happen?)	**Goal:** Learn how to predict the future, given historical data (What will happen?)	**Goal:** Suggest better actions based on data (What to do?)

NOTE: With today's registration, you'll be signed up for all six sessions. Although you can attend any of the sessions individually, it's recommended participating in all six weeks.

Bootcamp overview

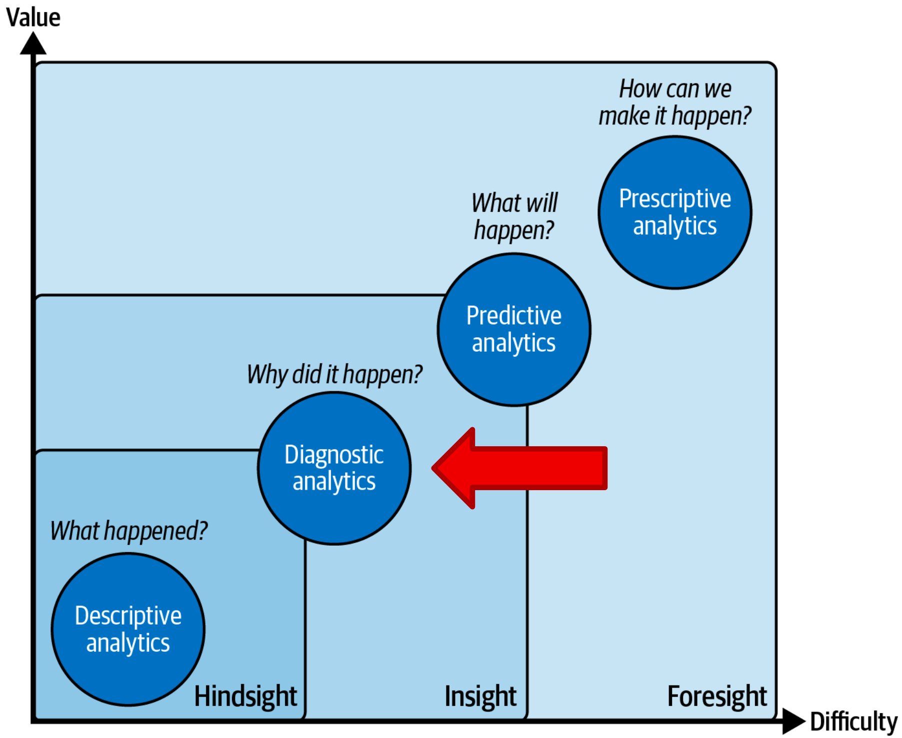
Learning goals Week 4:

- Understand the fundamentals of diagnostic analytics
- Apply diagnostic analytics techniques
- Differentiate correlation vs. causation
- Understand the 5-Whys method and conduct a root-cause analysis
- Learn the essentials about rule mining and association rules
- Understand the concepts of support, confidence, and lift
- Understand (customer) segmentation techniques
- Conduct a RFM analysis
- Use clustering techniques

Introduction to Diagnostic Analytics

Diagnostic Analytics - Introduction

- Diagnostic analytics focuses on identifying the root cause of a problem or **key factors driving a variable of interest**.
- It is a crucial part of business analytics and can help organizations make informed decisions based on data (turn data into insights)
- Diagnostic analytics involves the use of various tools and frameworks to identify the underlying cause of a problem.

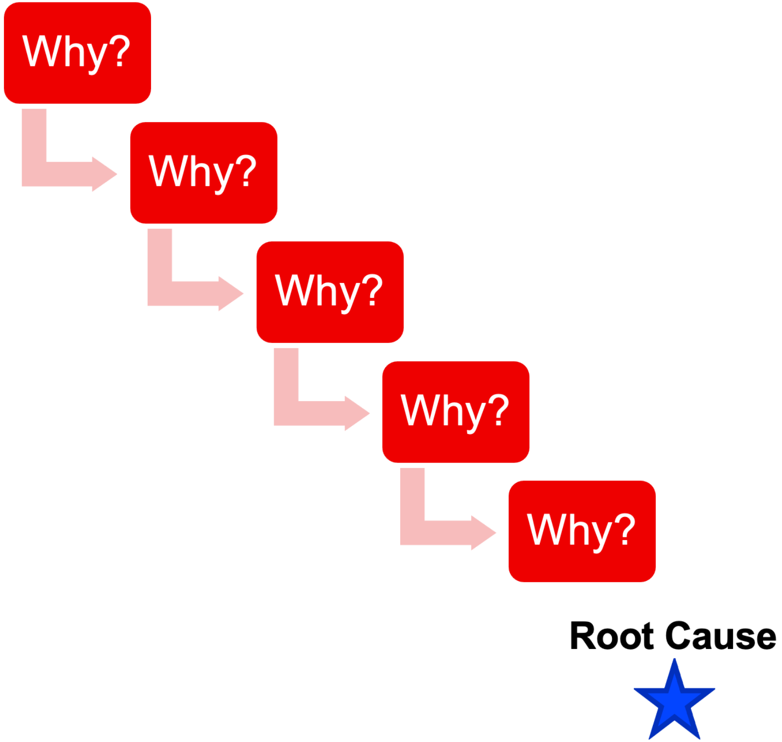


Diagnostic Analytics – Essential techniques

- **Root-cause analysis (RCA):** Identify the underlying cause of a problem or issue by systematically analyzing the symptoms of a problem to identify the underlying cause. (“drill-down”)
- **Segmentation:** Divide a large population or dataset into smaller groups or segments based on specific criteria (“slice-and-dice”)
- **Clustering**: Group similar objects or data points together based on their characteristics or attributes.

RCA: 5 Why method – Introduction

- Introduced by Sakichi Toyoda and used within Toyota Motor Corporation in the 1930s
- Problem-solving technique that involves asking "why" questions to identify the root cause of a problem
- Simple yet powerful method that can be applied to almost any type of problem
- **Goal**: dig deeper into the issue until you reach the underlying cause, typically after 5 steps
- Received various criticism (more on that later), but still useful



RCA: The 5 Why method - Example

Why are our sales declining?

- Because fewer people are visiting the site

Why are fewer people visiting the site?

- Because our SEO is not effective

Why is our SEO not effective?

- Because we don't use the right keywords

Why don't we use the right keywords?

- Because we don't have access to data needed to identify them

Why don't we have the data?

Root Cause: We don't have the right tools and knowledge to analyze data

Example: A company has noticed a decline in sales over the past few months, and they want to use the 5-Why method to identify the root cause of the problem.

The 5 Why method – Criticism

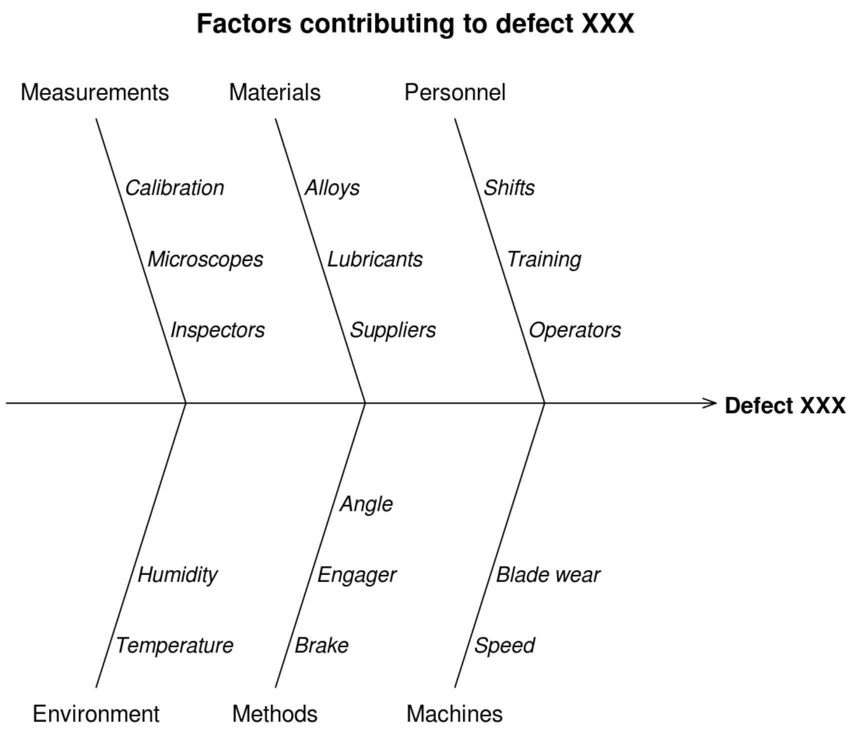
- The 5-Why method is not without criticisms, including oversimplification, biased questions, lack of data, limited scope, and lack of depth.
- In some cases, the root cause may be identified in fewer than 5 steps, while in other cases it may take more than 5 steps to get to the root cause.

→ **The key is to continue asking "why" questions until the underlying cause has been identified.**

- To mitigate some of the criticisms, it's important to use the method in conjunction with other problem-solving techniques (e.g. Issue Trees, Fishbone Diagram) and to ensure that the process is objective and data-driven.

The Fishbone diagram

- The fishbone diagram (Ishikawa diagram) is a visual tool used to identify all possible causes of a problem
- It helps to understand the relationships between the different causes and the effect or problem
- Can be used to guide data analysis and identify the data needed to solve the problem
- More complex than the 5-Why method – can be combined with it.



- Source: Wikipedia

More frameworks for root-cause analyses

- There are many other frameworks for conducting a root-cause analysis including:
  - Failure Modes and Effects Analysis (FMEA)
  - Fault Tree Analysis (FTA)
  - Root Cause Tree (RCT)
  - 8 Disciplines (8D)

- Each framework has its strengths and weaknesses, and the choice of framework will depend on the nature and complexity of the problem, as well as the resources and expertise available.
- It's important to choose the appropriate framework based on the problem at hand and to use it in conjunction with other problem-solving techniques.

- Source: Wikipedia

Things that can go wrong

Sampling bias

- **Selection bias**: Sample is not representative of the population, example: if a study on the effectiveness of a new drug only recruits participants from a certain age group, the results may not be generalizable to other age groups.

- **Survivorship bias:** This occurs when an analysis only includes observations that have “survived” a particular event or condition, leading to an unrepresentative sample. Examples: People who have survived a disease, customers who have bought a product, startup founders who build unicorns, …

Review > Psychol Bull. 1991 Jan;109(1):90-106. doi: 10.1037/0033-2909.109.1.90.

Left-handedness: a marker for decreased survival fitness

S Coren [1], D F Halpern

Affiliations + expand

PMID: 2006231 DOI: 10.1037/0033-2909.109.1.90

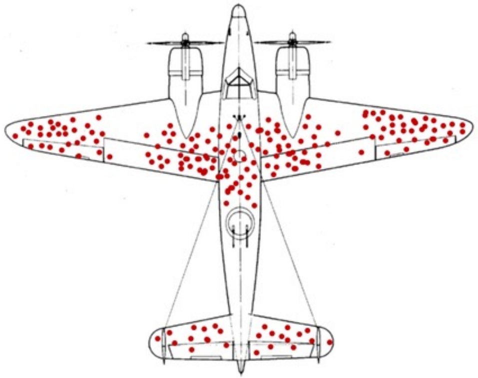
Abstract

Life span studies have shown that the population percentage of left-handers diminishes steadily, so that they are drastically underrepresented in the oldest age groups. Data are reviewed that indicate that this population trend is due to the reduced longevity of left-handers. Some of the elevated risk for sinistrals is apparently due to environmental factors that elevate their accident susceptibility. Further evidence suggests that left-handedness may be a marker for birth stress related neuropathy, developmental delays and irregularities, and deficiencies in the immune system due to the intrauterine hormonal environment. Some statistical and physiological factors that may cause

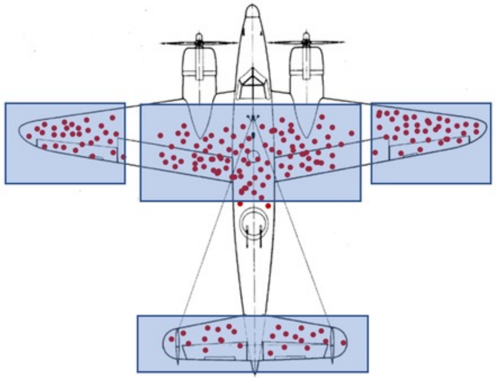
Failure to find statistical significance in left-handedness and pathology studies: A forgotten consideration

handedness and various risk factors reported by other investigators. This paper demonstrates that many of these studies have simply lacked the statistical power to do so. As demonstrated here, the problem usually consists of inadequate sample sizes for the conditions. For example, given a sample of 1,000 subjects and a true rate of pathology of 10% in right-handers, even if left-handers have a 50% higher risk (true rate of pathology, 15%) the statistical power is only.3. This means that this difference would not reach statistical significance at $p < .05$ in 70% of the studies with a sample of this size! A figure

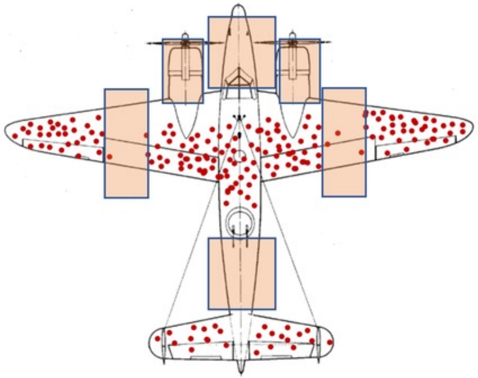
Survivorship bias



Our data if only from returning flights. Here we is a visualization of the places that bullet holes were observed.



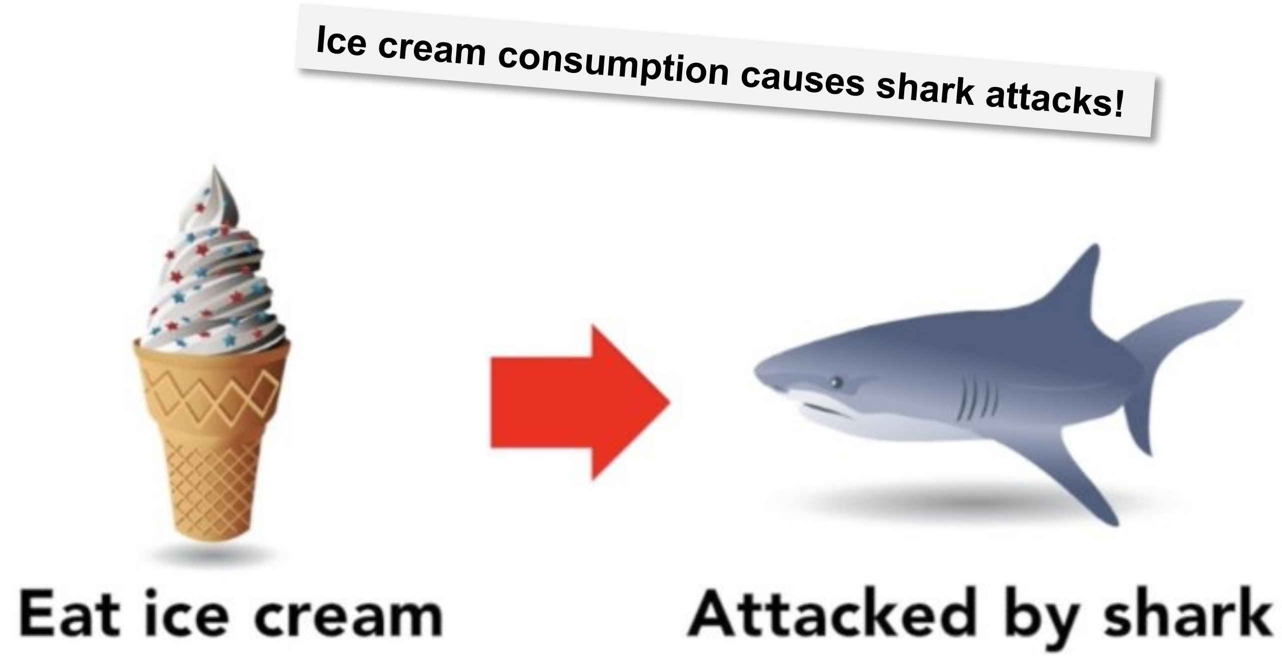
And initial guess at how to fix this might be to apply additional armor platting to the parts of the plane with the most holes...



.... However this is where planes that *returned* had bullet holes. The planes we want to protect are the ones that did *not* return, so we should place armor there.

- Source: https://www.countbayesie.com/blog/2020/11/5/survivorship-bias-in-house-hunting-a-practical-modeling-example-using-jax

Correlation vs. Causation



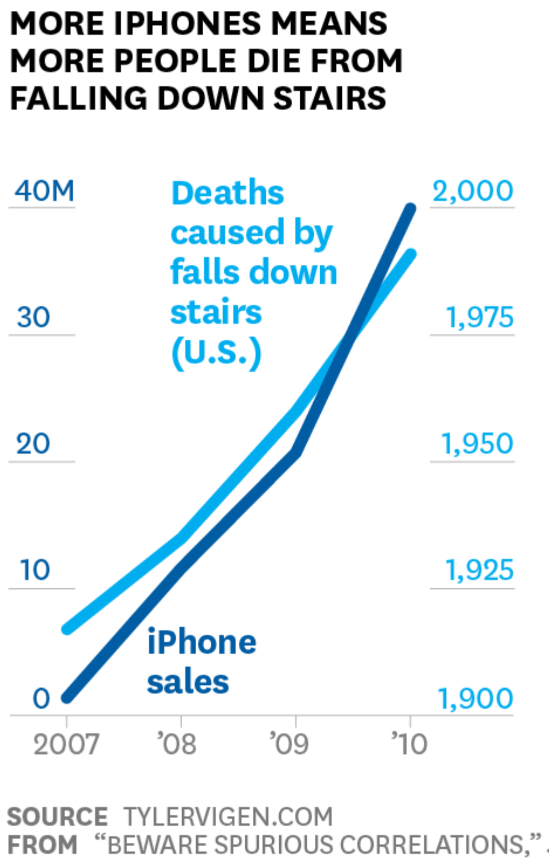
- Source: https://www.kdnuggets.com/2019/01/dr-data-ice-cream-linked-shark-attacks.html

Correlation vs. Causation

- **Correlation**: describes a relationship between objects.
- **Correlation does not imply causation!** Correlation cannot be used to infer a causal relationship.
- **Causation**: A relationship between at least two objects in which one is caused by the other. An event E is caused by a cause C, if the occurrence of C implies the occurrence of E. **Causation implies correlation**.

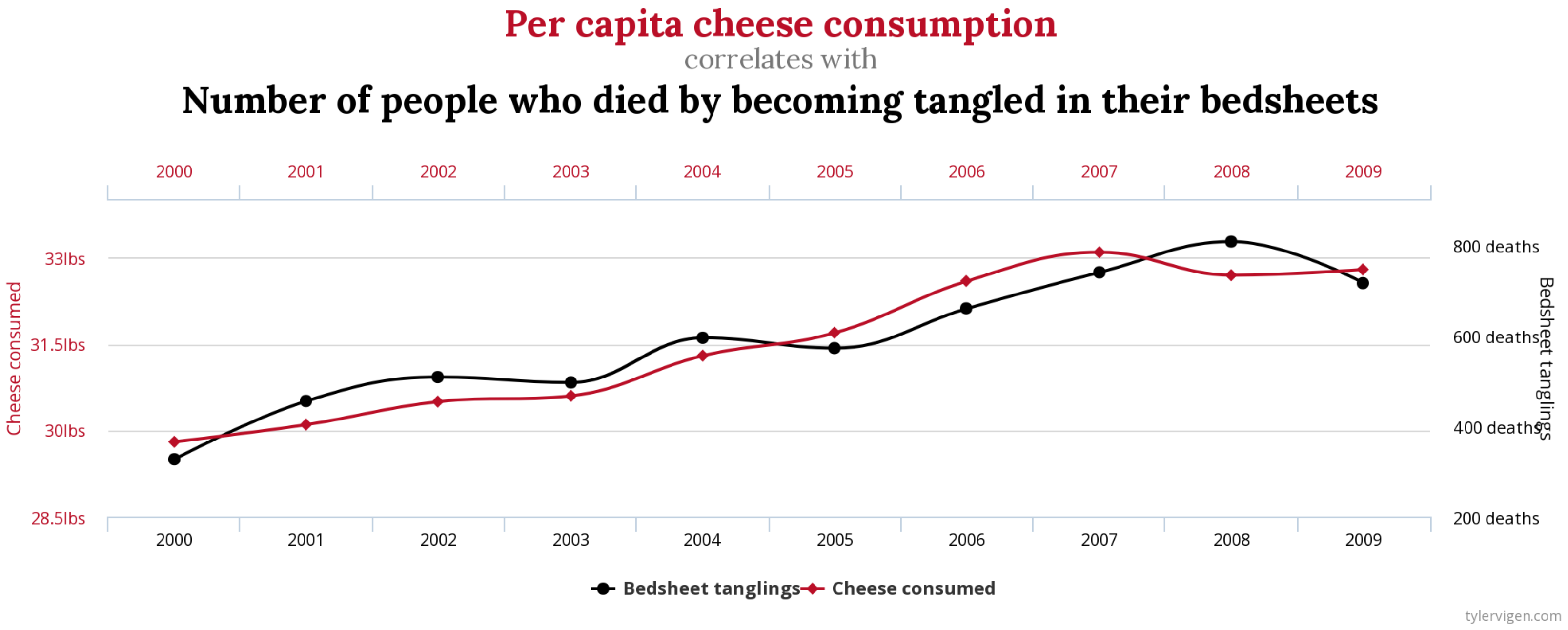
→ Data Analytics finds correlations!

→ Correlations can happen by coincidence in large data sets (“If you torture the data long enough, it will confess”)



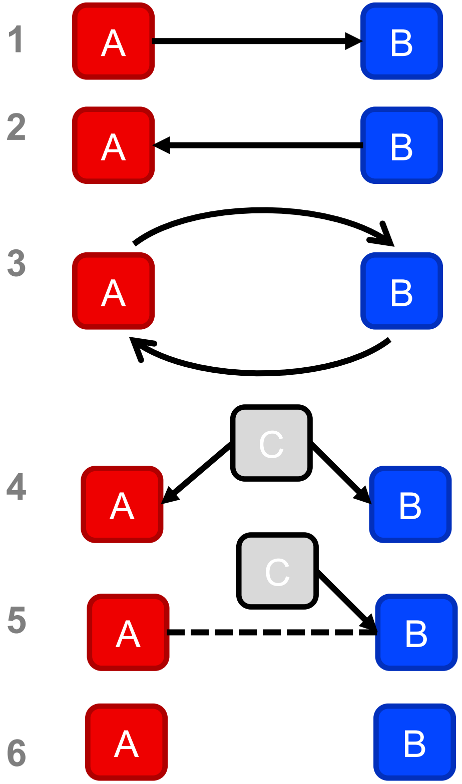
- https://hbr.org/2015/06/beware-spurious-correlations

Correlation vs. Causation



- Source: https://www.tylervigen.com/spurious-correlations

Correlation vs. Causation



- Source: Wikipedia

Correlation between A and B is interpreted as A causes B
6 options are possible:

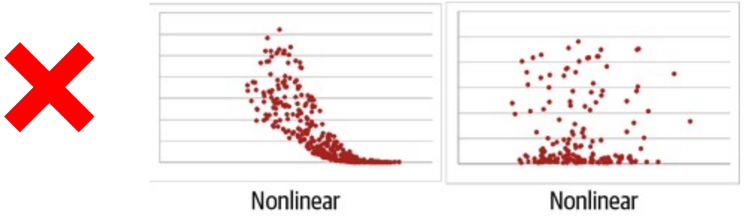
1. A causes B
2. B causes A
3. A and B both partly cause each other
4. A and B are both caused by a third factor C
5. B is caused by C, which is correlated to A
6. There is no connection between A and B; the correlation happened by chance.

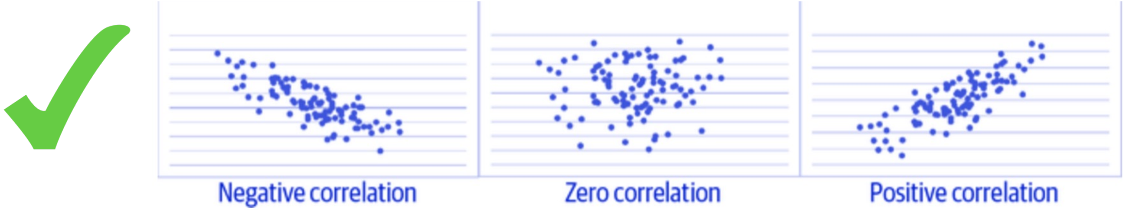
Examples for confounding factor C:

- The number of newly planted trees (A) correlates with the growth rate of grass (B)
- Both events are affected by the season of the year (C), specifically by temperature, humidity and sunshine conditions.

Pearson-Correlation

- Measures the strength of the **linear relationship** between two variables.
- The value lies between -1 (perfect negative linear relationship) and +1 (perfect positive **linear** relationship).





- Quelle: Mount, G., Advancing into Analytics (2021), O'Reilly

Q&A

Segmentation

Introduction to Segmentation Techniques

- **Segmentation** = dividing a group objects or entities into smaller, more manageable groups based on shared characteristics.
- **Various techniques**: demographic segmentation, behavioral segmentation, psychographic segmentation, …
- Clustering: A distinct set of methods and techniques to group data points together based on similarities in their attributes.

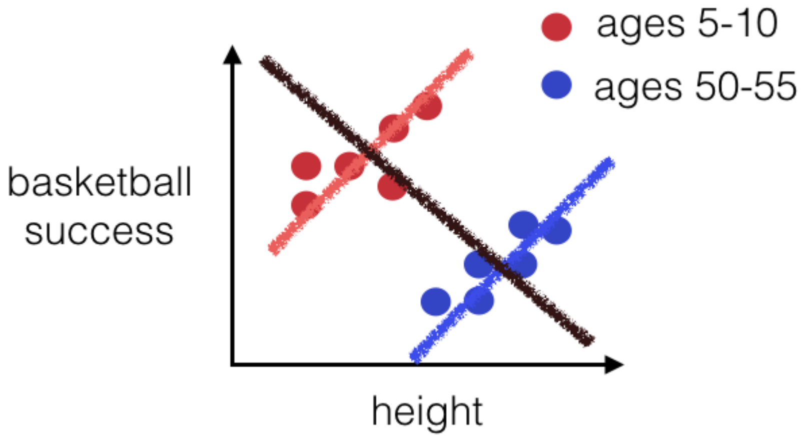
- Image source: https://www.baker-richards.com/segmentation-what-how-and-why/

Segmentation: Goals

→ Identify patterns and trends that can help us better understand the data we are analyzing.

→ Gain insights into patterns and trends that might not be apparent when looking at the data as a whole.

- **Simpson’s paradox**: A phenomenon where a trend appears in several groups of data but disappears or reverses when the groups are combined



- Image source: https://towardsdatascience.com/simpsons-paradox-d2f4d8f08d42

Segmentation: Use Cases

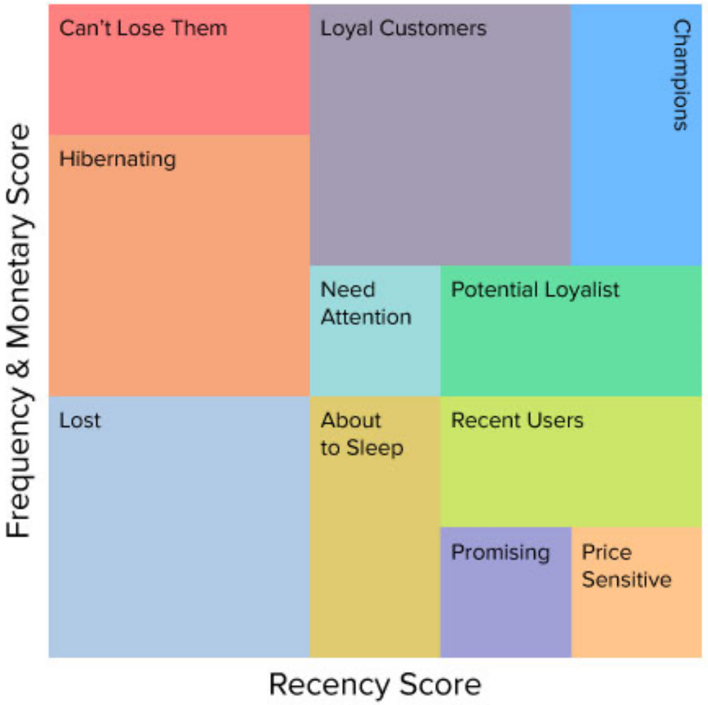
- **Behavioral segmentation:** Divide customers based on their behavior, such as purchase history, website activity, or engagement with marketing campaigns to identify customers who are more likely to make a purchase or respond to a marketing campaign.
- **Demographic segmentation**: Divide customers based on demographic factors such as age, gender, income, or education level to identify groups of customers who may have different needs, preferences, or buying habits.
- **Geographic segmentation:** Divide customers based on location, such as their country, region, or city to identify groups of customers who may have different regional preferences
- **Customer lifecycle segmentation**: Divide customers based on where they are in their relationship with a brand, such as first-time buyers, loyal customers, or inactive customers to adapt messaging or offers.

RFM Analysis

RFM-Analysis is one of the most popular behavioral segmentation techniques used in business analytics.
(**RFM** = **R**ecency, **F**requency, **M**onetary Attributes)

- **Recency**: how recently a customer has made a purchase
- **Frequency:** how often a customer makes a purchase
- **Monetary:** how much a customer spends on each purchase

Goal: Segment customers based on their purchasing behavior and identify different segments of customers, such as high-spending, frequent purchasers or low-spending, infrequent purchasers.



- Image Source: https://www.notifyvisitors.com/blog/rfm-analysis/

Interactive Lab:
Perform a RFM analysis

Clustering

Clustering

- → What is clustering?
- → How does clustering work?

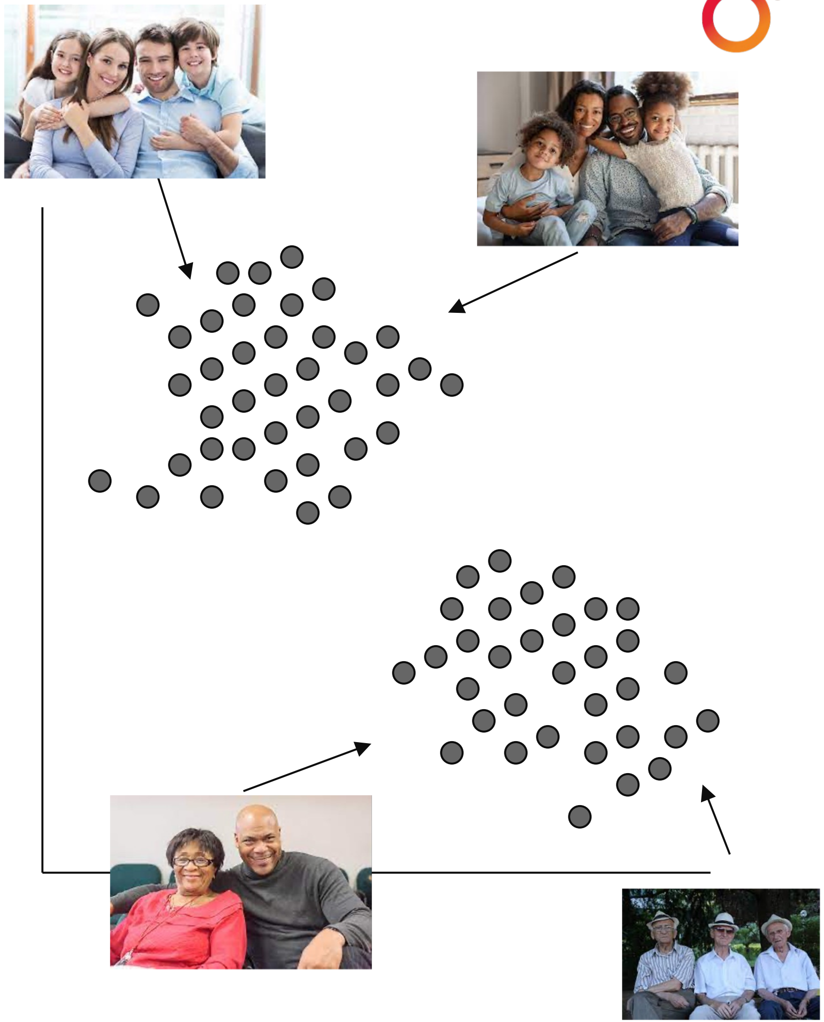
Clustering

Task

- Given a set of objects, find interesting structures in the data
- Cluster = set of objects which are similar to each other and different from other clusters

- Examples:
    - Customer Profiling
    - Advertisement

→ Typically highly dimensional data!



How does clustering work?

→ Optimization problem

→ But: No loss function (missing label)

Two important concepts:

- Variability
- Dissimilarity

- Image source: https://www.ml-science.com/k-means-clustering

Clustering – Variability

- Measures the **variability** of data points **inside** a single cluster

- Why not variance?
    - Normalization is problematic for clustering
    - The amount of examples (cluster size) matters!
    - Big “bad” clusters are worse than small “bad” clusters

$$variability(c) = \sum_{e \in c} distance(mean(c), e)^2$$

c… single cluster
e… example

Clustering – Dissimilarity

- Measures the **total variability** of **multiple clusters**

- What are we optimizing?
  - Value for C that minimizes dissimilarity(C)?
  - → No! Each data point would be its own cluster!
  - → **Constraint needed!**
  - Examples: Total number of clusters, minimum distance between clusters, etc.

$$dissimilarity(C) = \sum_{c \in C} variability(c)$$

C… group of clusters
c… single cluster

Constraints in Clustering

How to choose a constraint (number of clusters) in clustering:

Option A: Apply constraint from problem / business logic
- Example: Can’t handle more than 5 separate marketing campaigns

Option B: Derive constraint from data itself
- Silhouette value
- Elbow method
- …

Quick recap

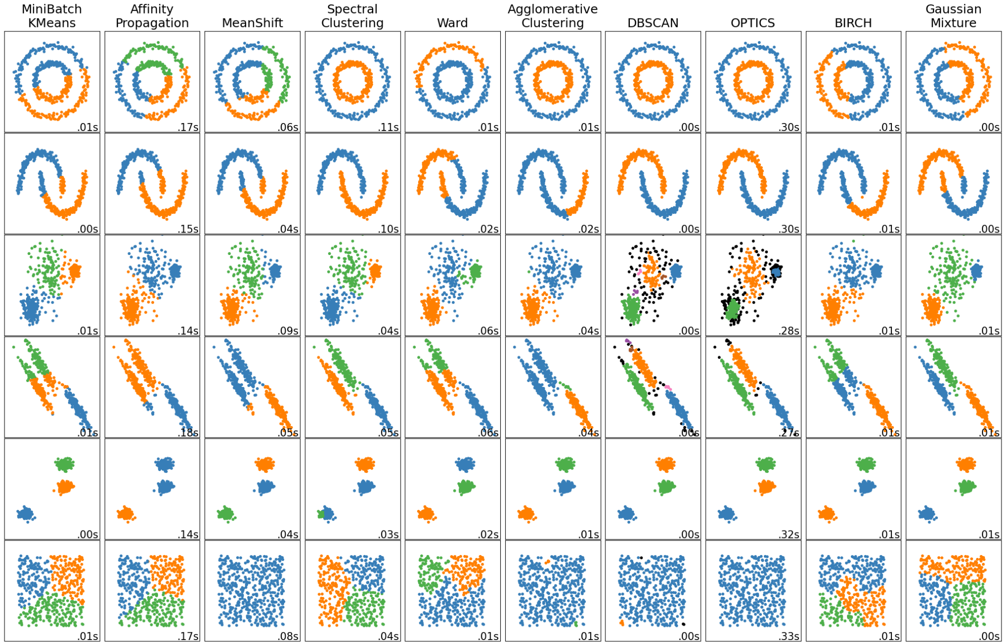
- Clustering = Process to organize objects into separate groups where items within the groups are similar to each other and different to other groups (clusters)
- Clustering is an optimization problem
- Optimize **dissimilarity** of all clusters while simultaneously minimize **variability** within clusters given a constraint (e.g. total cluster number, minimal distance)
- Different **constraints** lead to different solutions

Types of Clustering

- Too many methods to cover them all
- No universal framework
- Concepts are overlapping

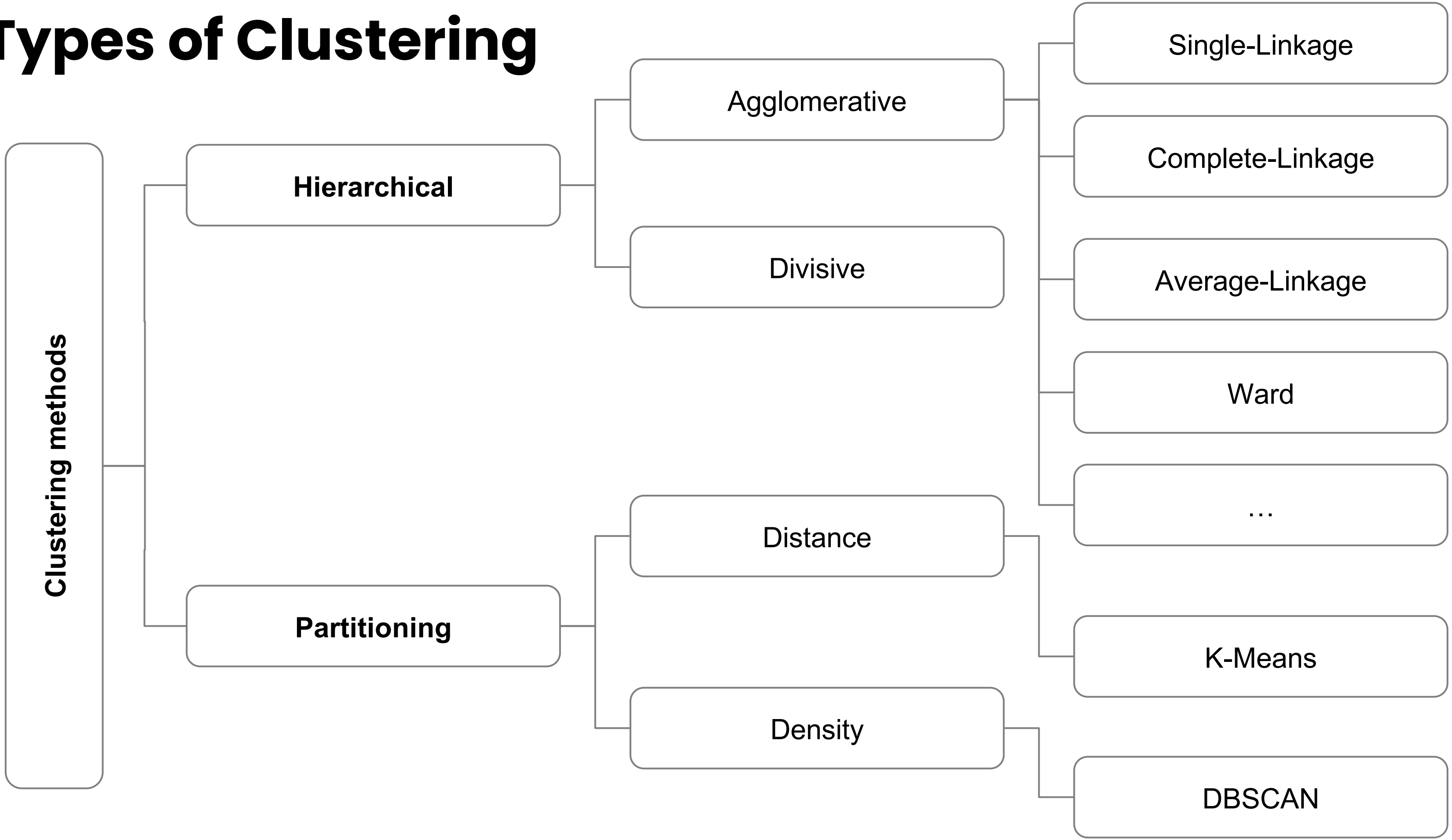
Two major categories:

- **Hierarchical**
- **Partitioning**

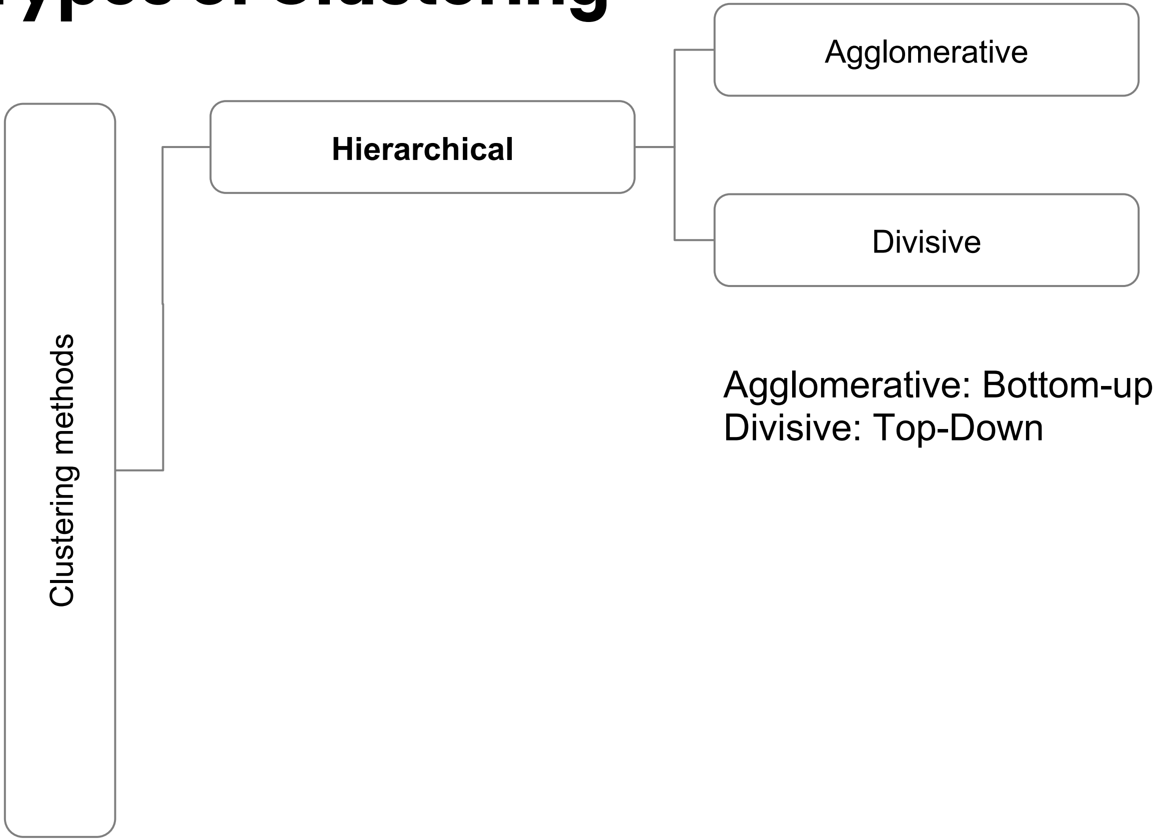


- Image Source: https://scikit-learn.org/stable/modules/clustering.html

Types of Clustering

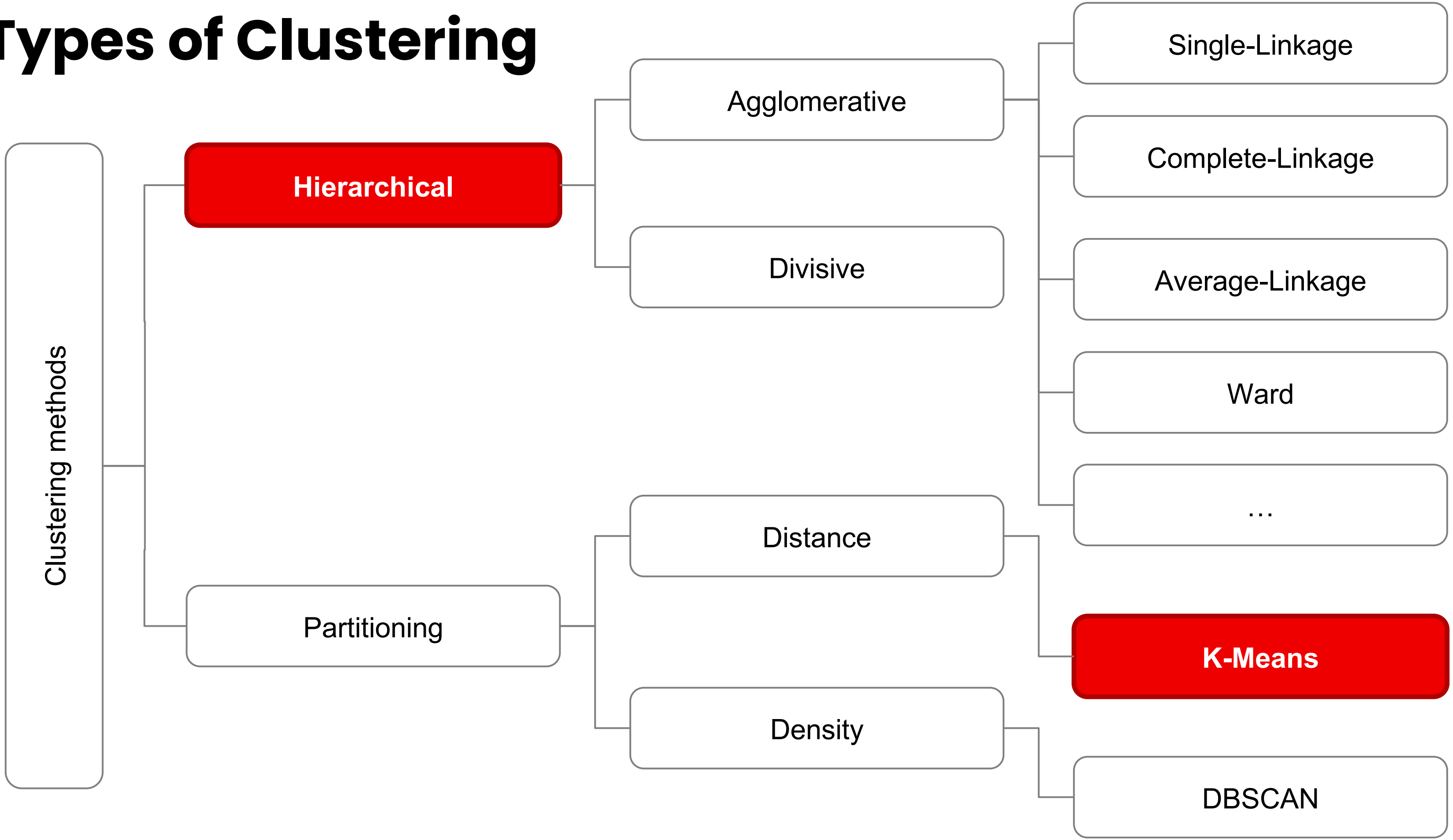


Types of Clustering



Agglomerative: Bottom-up
Divisive: Top-Down

Types of Clustering



Before we start…

Prerequisite for clustering:

Distance measures

→ Measure the **difference** (amount of space) between two objects
→ Tool to make a statement about proximity

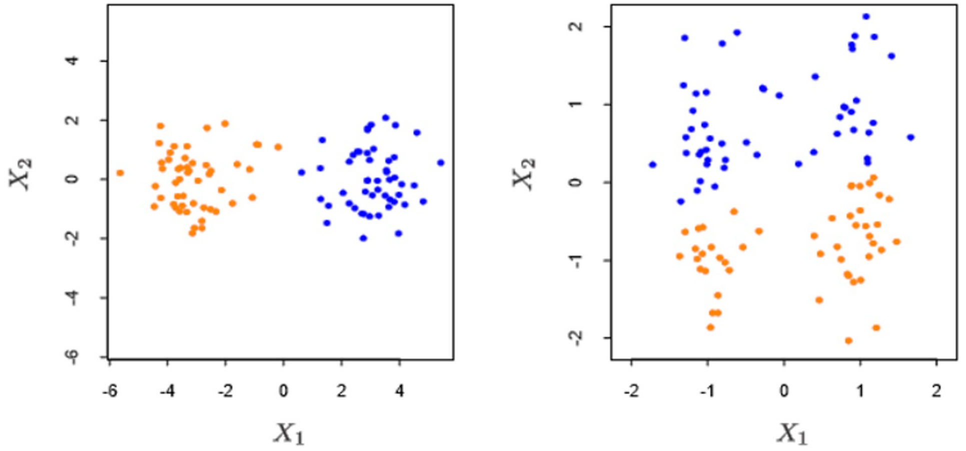
Examples:

- Regular ‘distance’ such as miles, km, etc.
- In data analysis: unitless scales (normalized / standardized measures)

Why normalize scales?

- Original units can strengthen or weaken the effect of the clustering

→ This should be avoided!

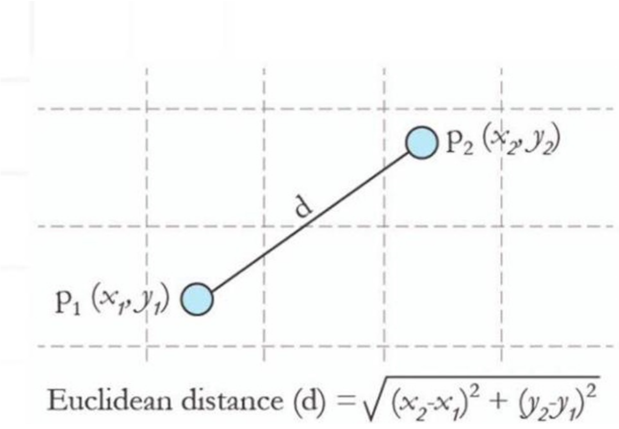


- Source: Hastie, T. et al. (2009): p. 506

Distance Measures for numerical data

Euclidean distance

- widely used distance measure
- multidimensional data
- “Straight-line distance”, “as the crow flies”
- Only valid in Euclidean space: Pythagorean theorem is valid ($a^2 + b^2 = c^2$)



Example: Two points A (x1, y1) and B (x2,y2)

x_diff = x1 - x2   x_diff_sq = x_diff^2
y_diff = y1 - y2   y_diff_sq = y_diff^2

dist_sq = (x_diff_sq + y_diff_sq)
d = sqrt(dist_sq)

Distance Measures for numerical data

Manhattan distance

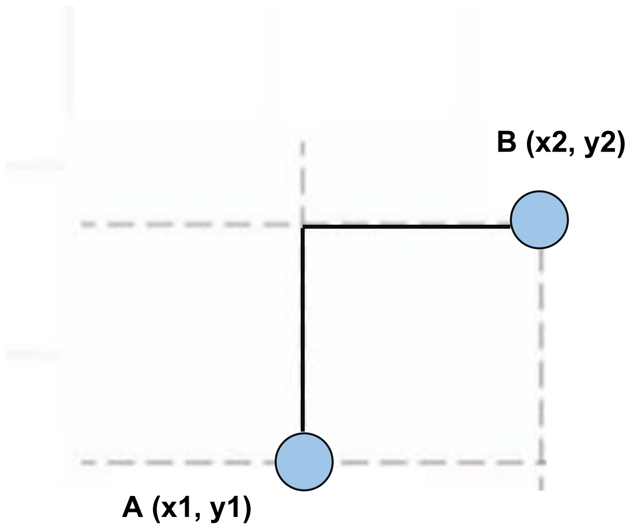
- “Taxicab” or “Cityblock” distance
- Measures the distance between two points in a grid (no diagonal, only horizontal and vertical paths)
- Important when data has grid-based layout (e.g. image pixels)

Example: Two points A (x1, y1) and B (x2,y2)

d = |x1-y1| + |x2-y2|

N-dimensional Space:

d = |x1-y1| + |x2-y2| + ... + |xn-n|



Distance Measures for categorical data

Hamming distance

- Measures the “mismatches” between two equal length sequences of data

Example:

dog != cat => 1
black == black => 0
chipped != not chipped => 1

d = 2

ID	Type	Color	Chipped?
1	dog	black	chipped
2	cat	black	Not chipped

$$D_H = \sum_{i=1}^{k} |x_i - y_i|$$

$$x = y \Rightarrow D = 0$$

$$x \neq y \Rightarrow D = 1$$

K-Means Algorithm

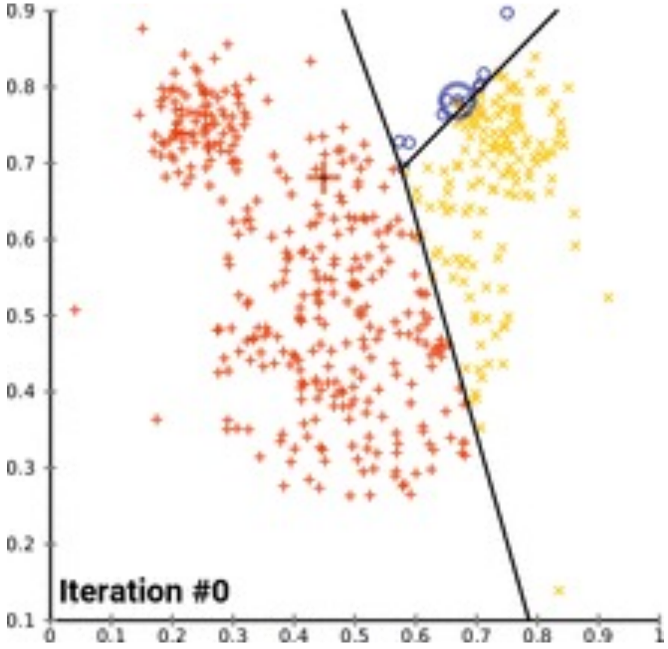
- What is the k-Means Clustering algorithm?
- How does it work?
- Advantages & disadvantages

K-Means Algorithm

- One of the popular clustering technique

  → General purpose

- Based on Centroids
- Examples belong to the closest centroid
- Needs parameter *k*: number of clusters
- Execution is very fast
- Resulting clusters are convex
- Partitioning algorithm

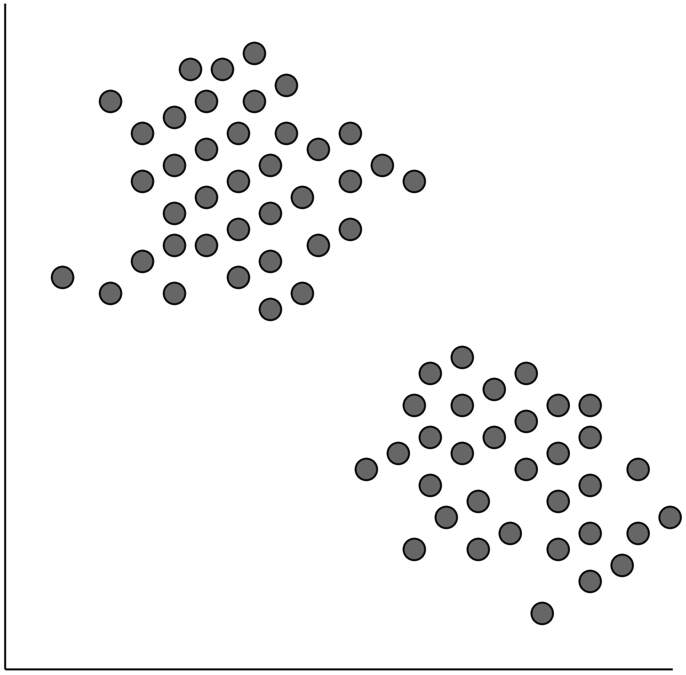


- Source: https://en.wikipedia.org/wiki/K-means_clustering

K-Means Algorithm

Start

$k = 2$



K-Means Algorithm

Start

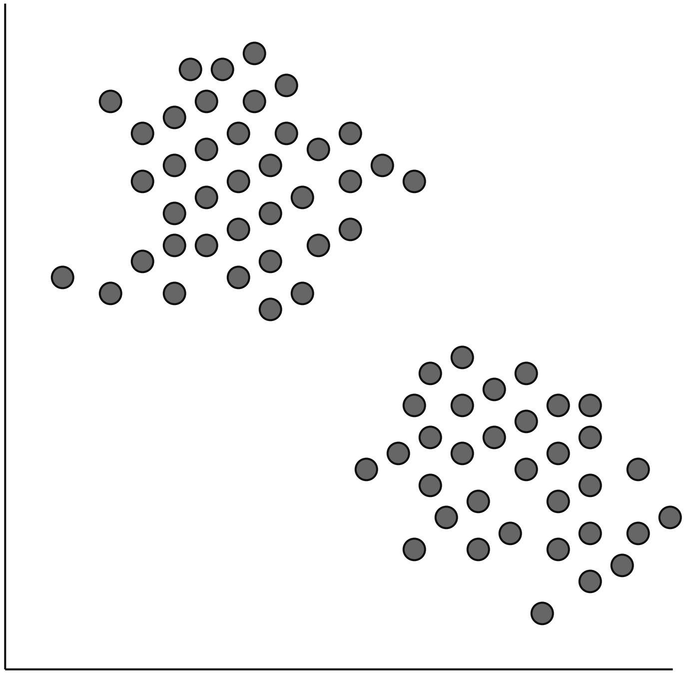
↓

Initialize centroids

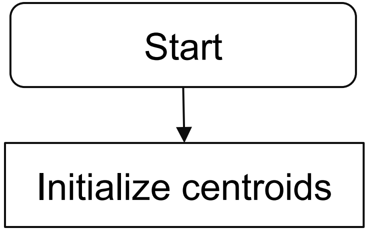
Cluster Centroids

Cluster 1

Cluster 2



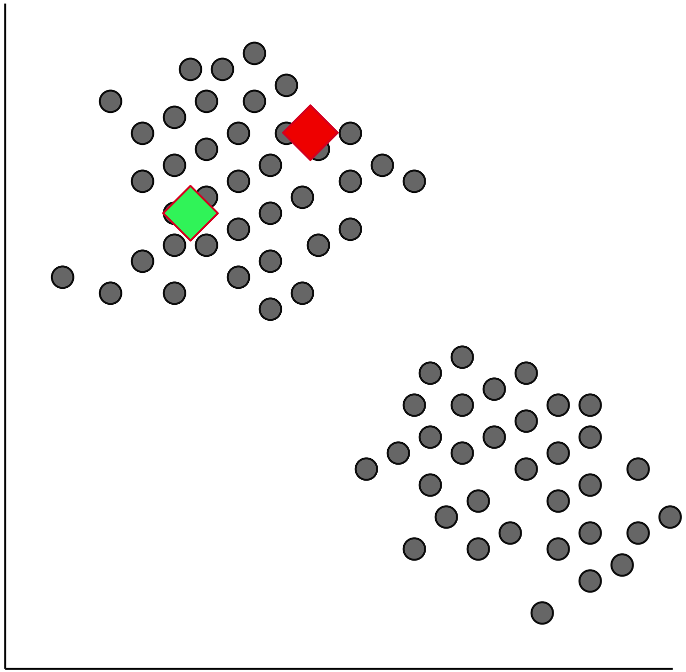
K-Means Algorithm



Cluster Centroids

Cluster 1

Cluster 2



Random partitioning:

- Randomly assign each data point to one of the k clusters.
- Compute the mean of the points in each cluster to obtain the initial positions of the centroids.

Forgy method:

- selects k random points from the data directly as the initial positions of the centroids.

K-Means Algorithm – Iteration 1

Start

↓

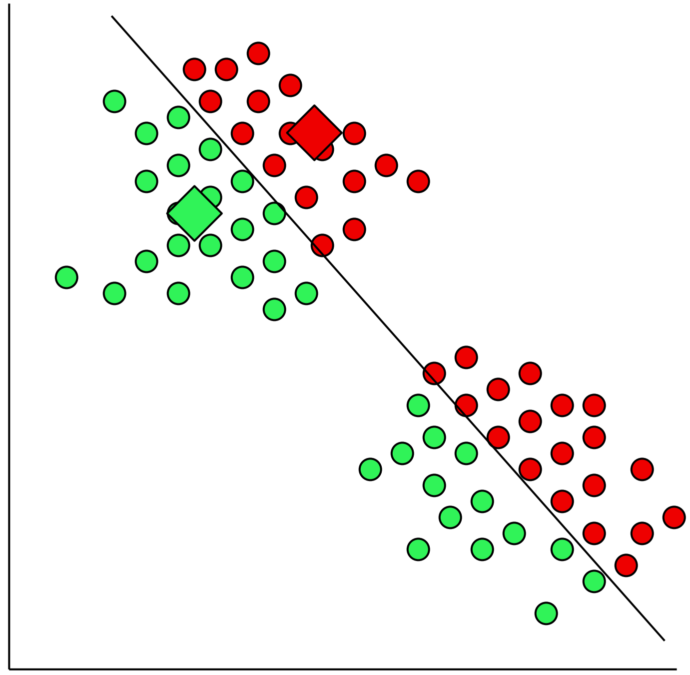
Initialize centroids

↓

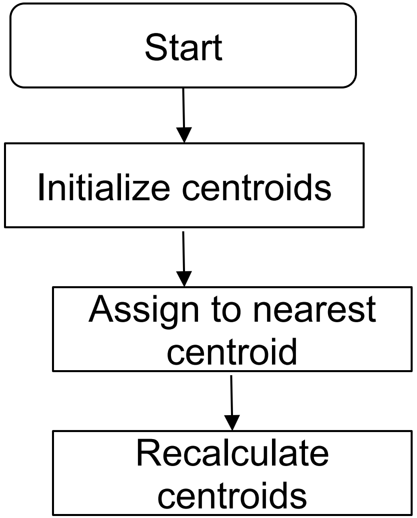
Assign to nearest centroid

→ Assign each data point to the cluster with the nearest centroid.

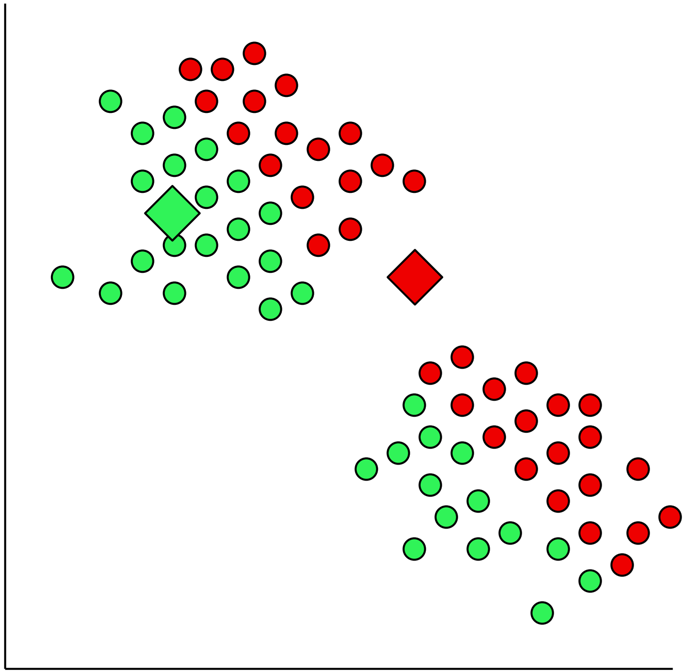
(e.g. Euclidean distance)



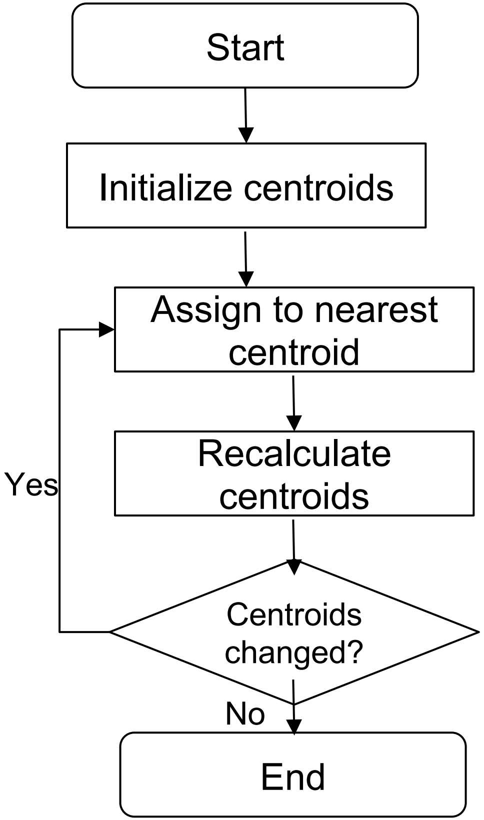
K-Means Algorithm – Iteration 1



→ Recalculate the centroids using all data points in each cluster



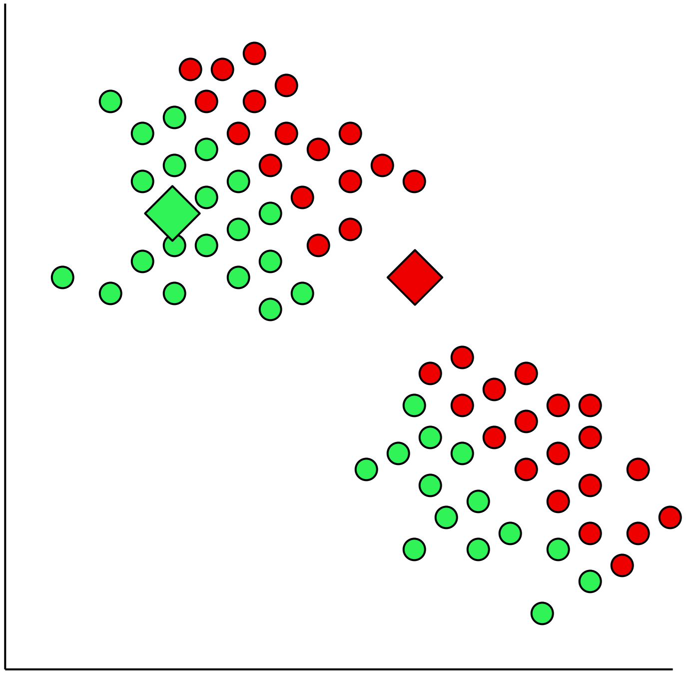
K-Means Algorithm – Iteration 1



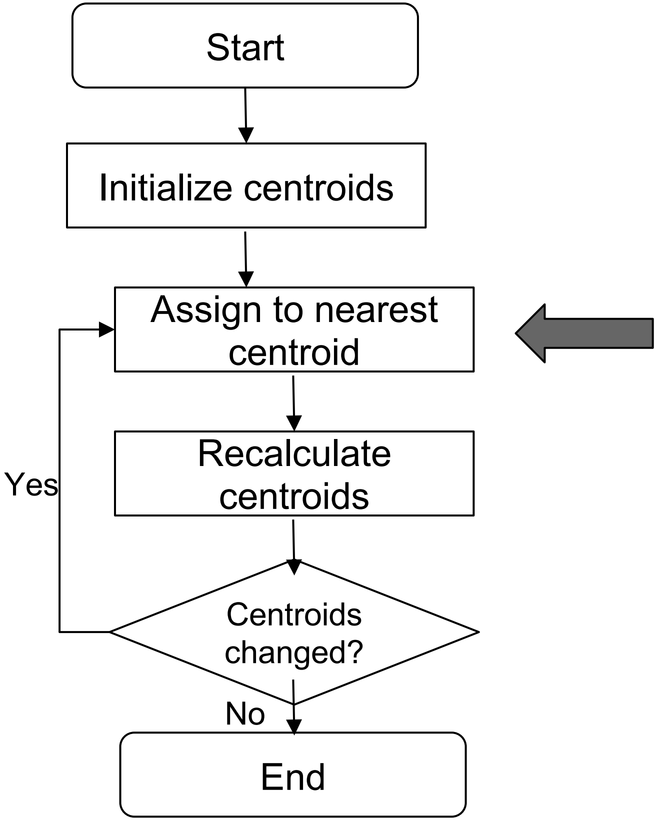
→ Check if the cluster centroids have changed significantly.

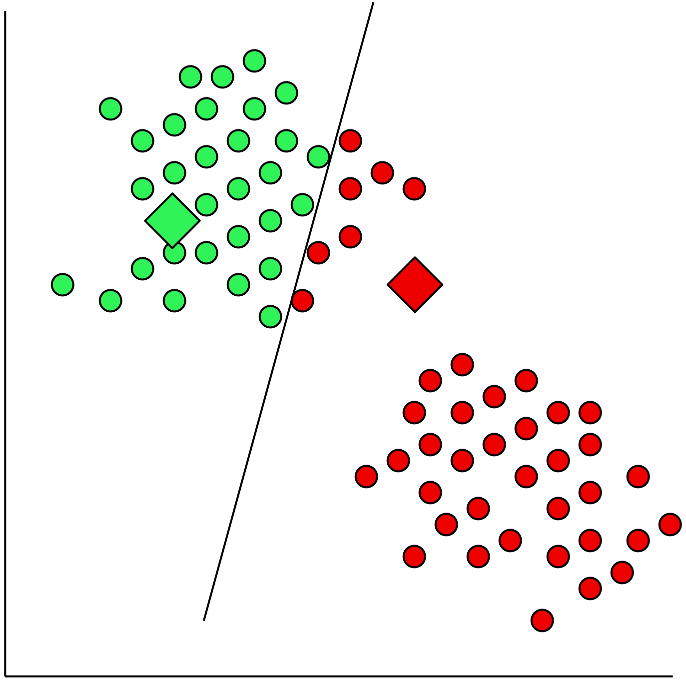
If yes:
→ do next iteration and repeat the process.

If no change or max. number of iterations reached:
→ algorithm terminates

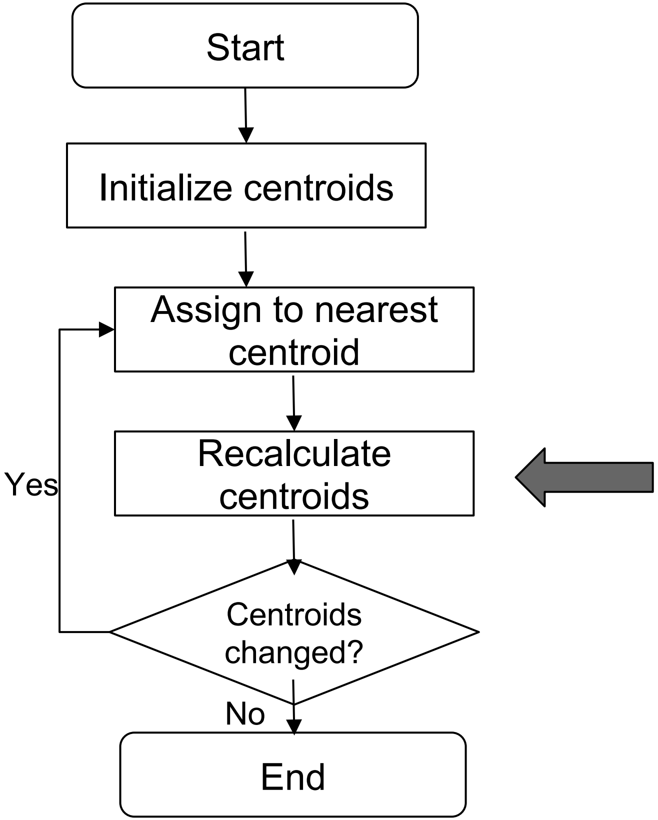


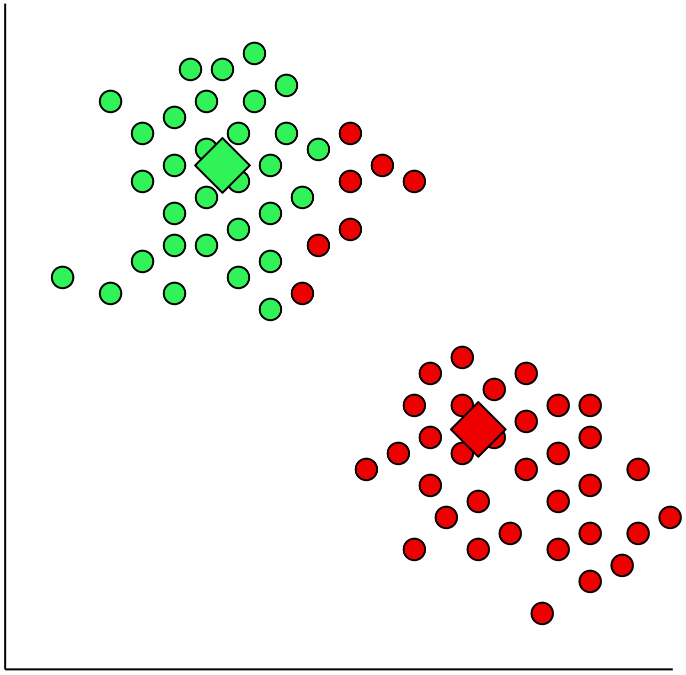
K-Means Algorithm – Iteration 2



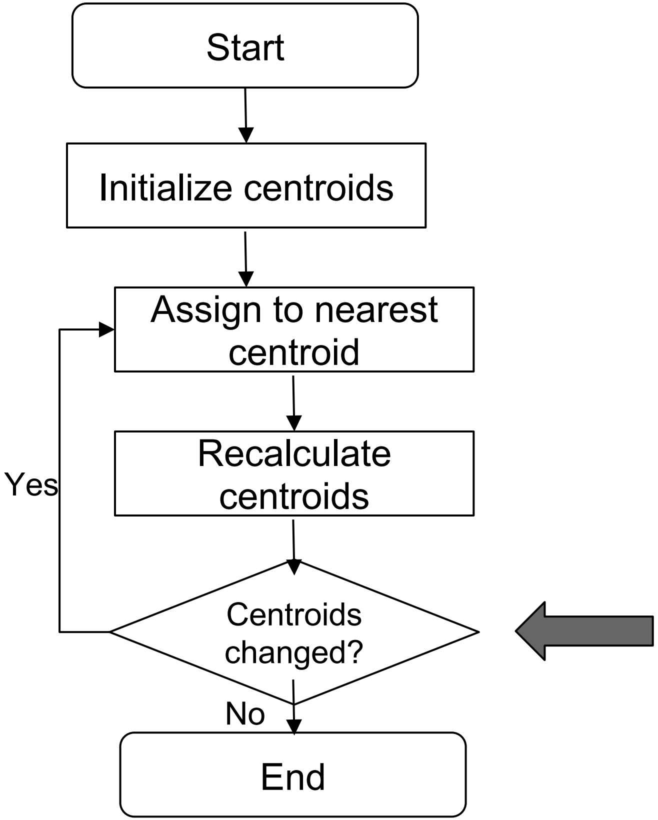


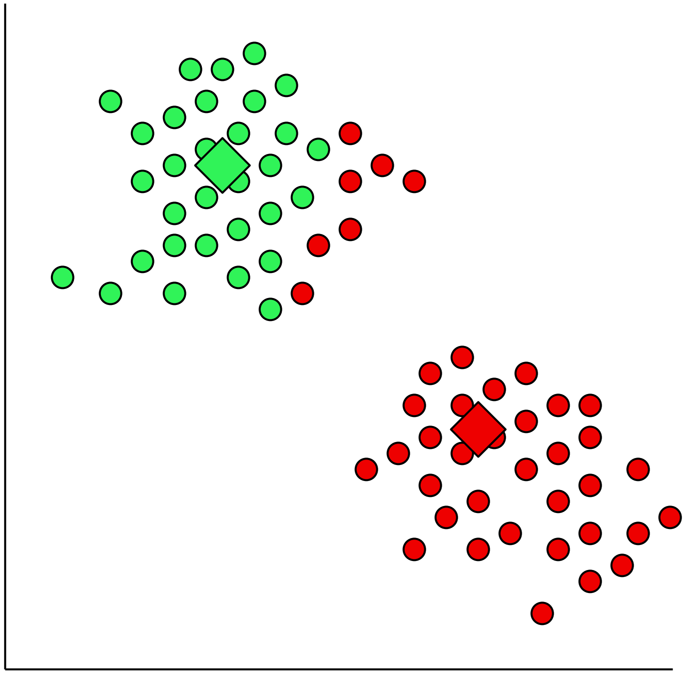
K-Means Algorithm – Iteration 2



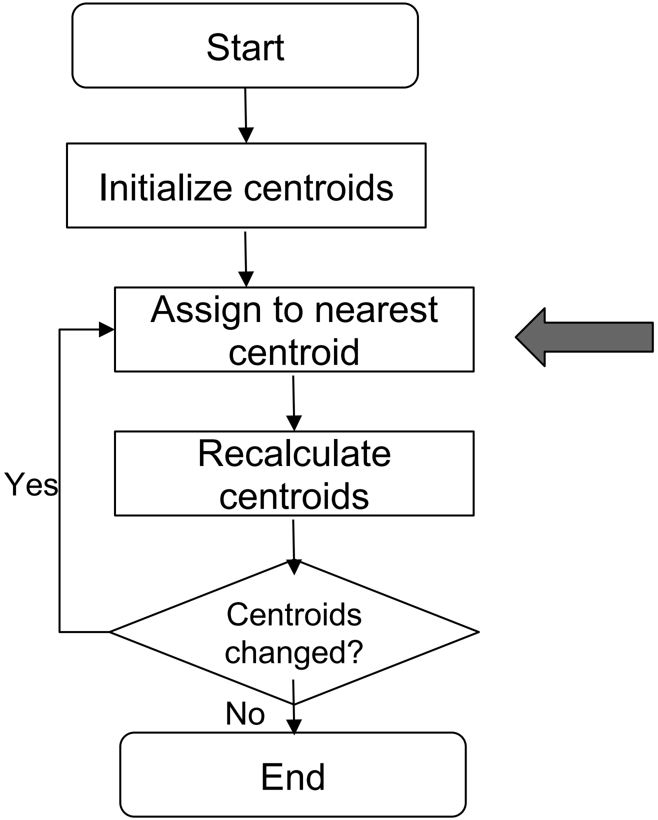


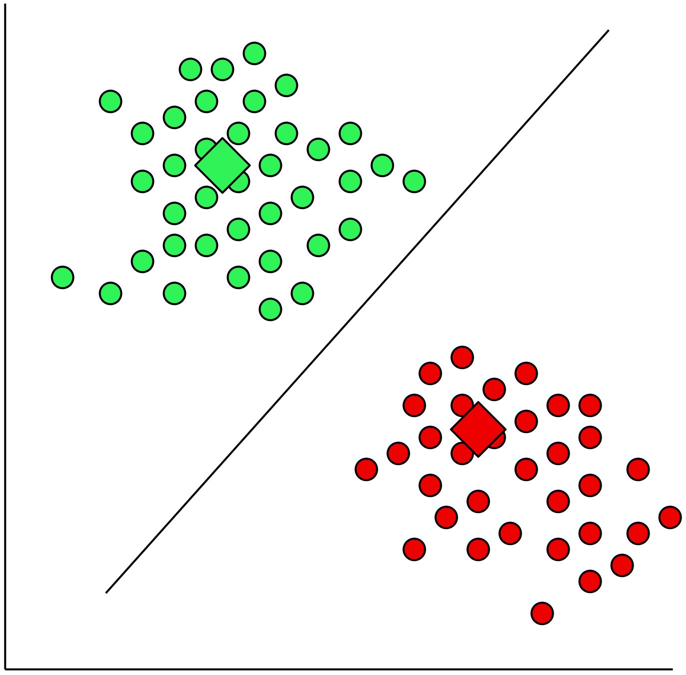
K-Means Algorithm – Iteration 2



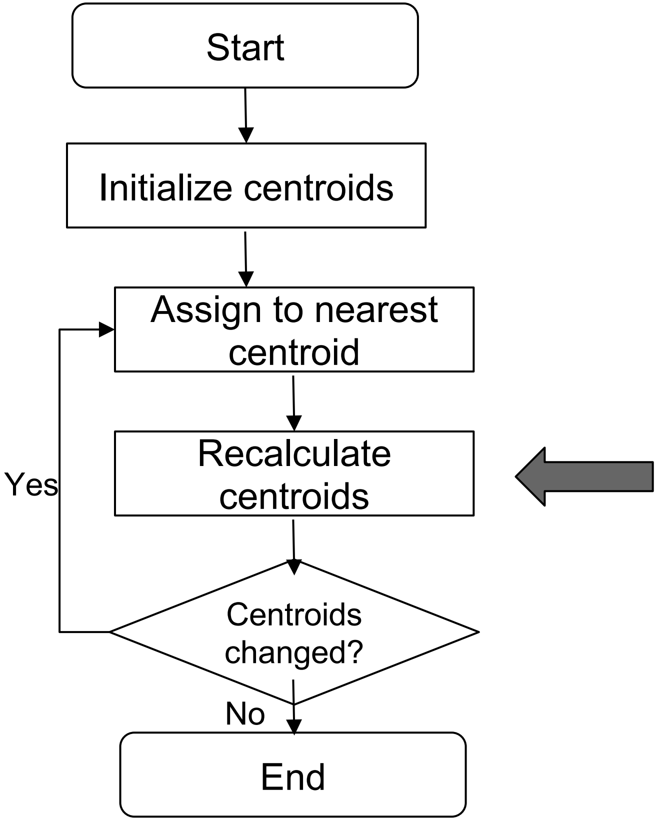


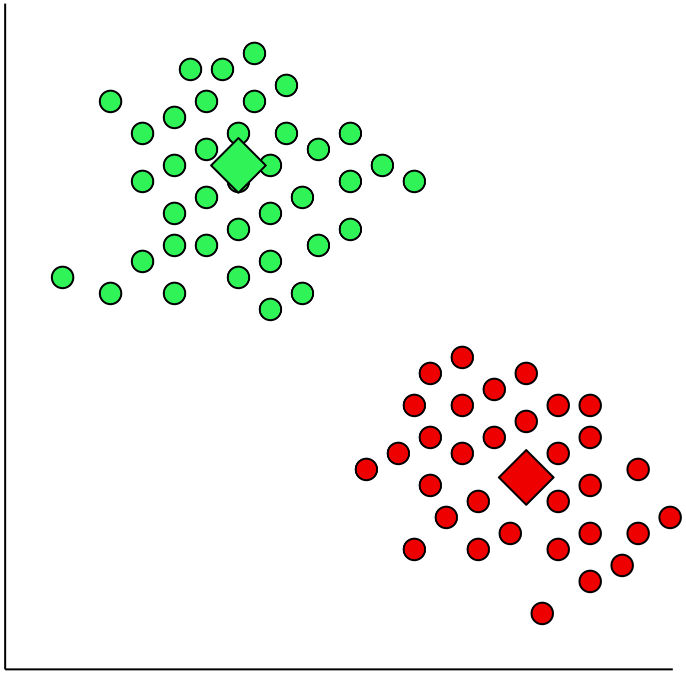
K-Means Algorithm – Iteration 3



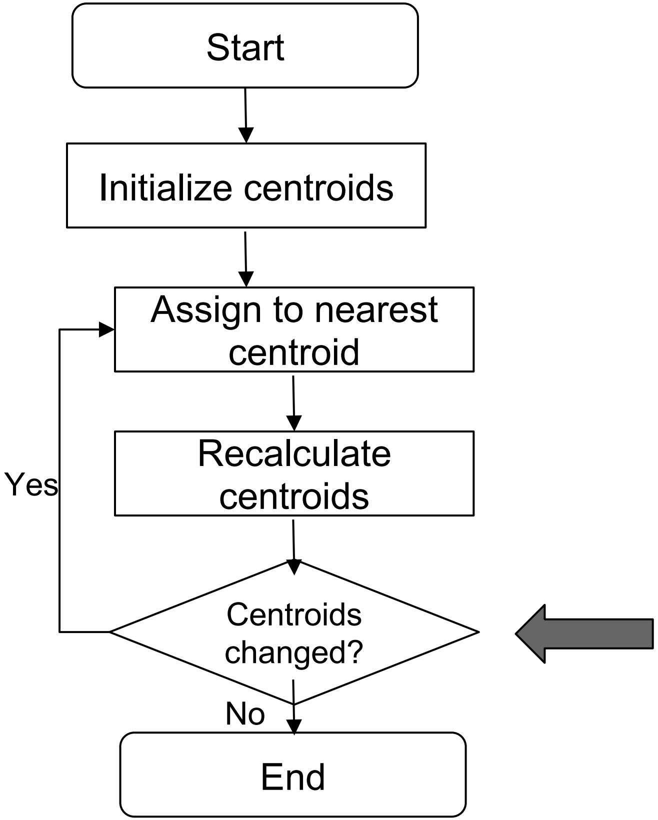


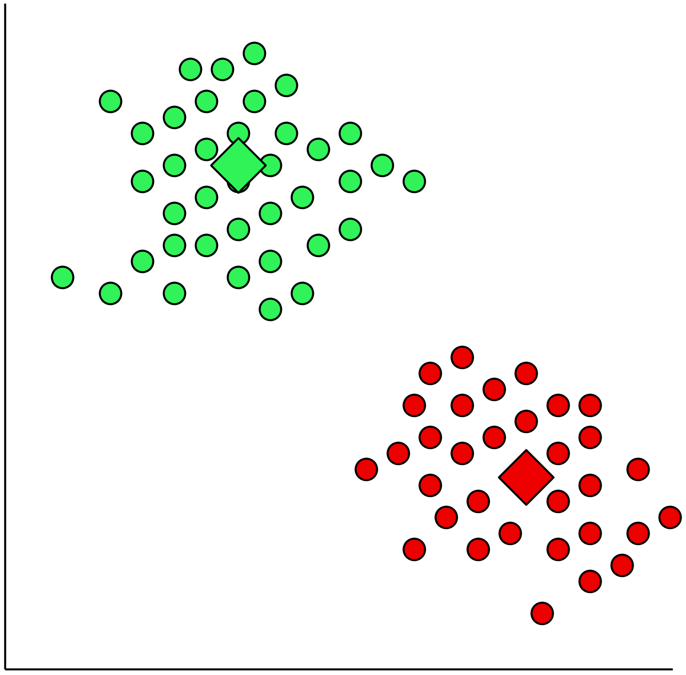
K-Means Algorithm – Iteration 3



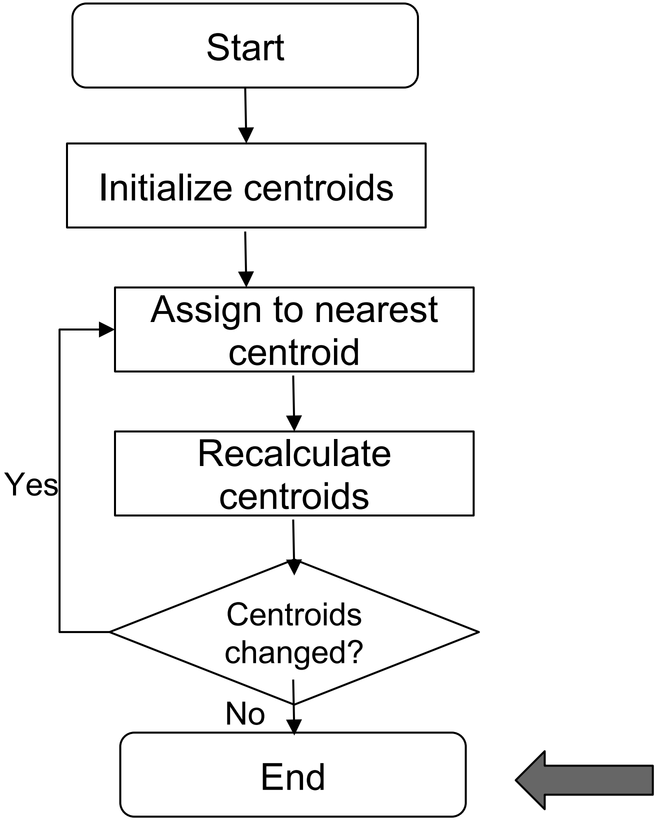


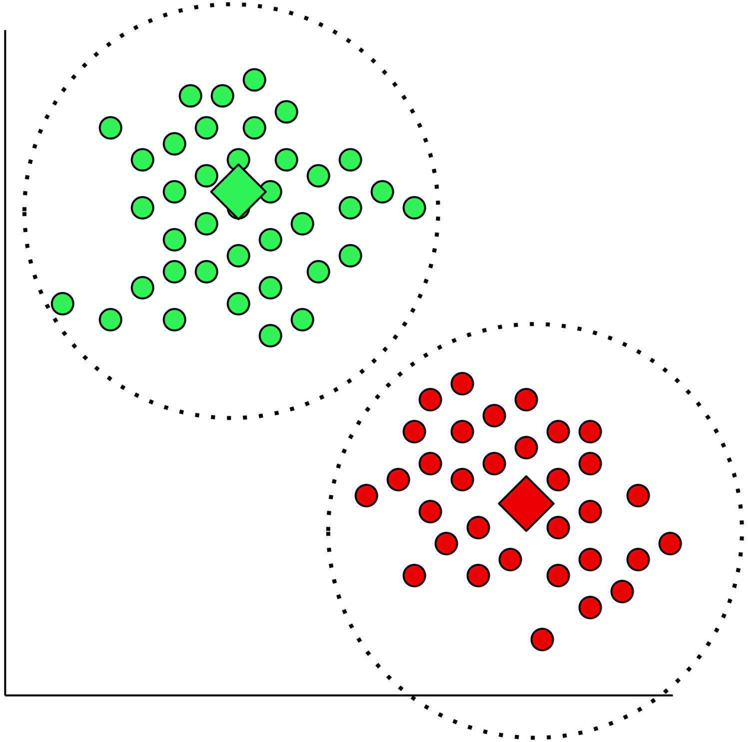
K-Means Algorithm – Iteration 3





K-Means Algorithm – Termination





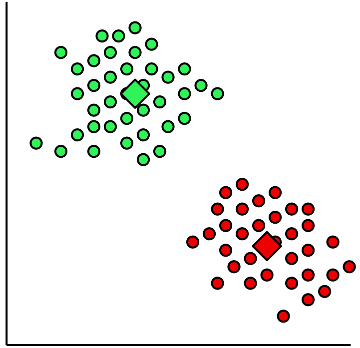
Finding a good value for k

- For the K-Means, algorithm, we have to specify k ourselves
- k is the total amount of clusters we get

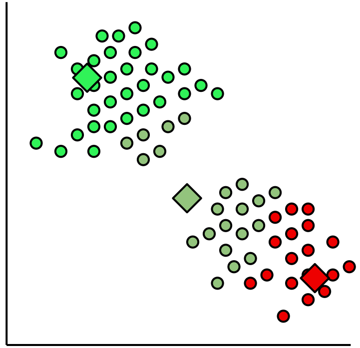
How to get a good value for k?

→ More clusters isn’t always better!

k = 2



k = 3



Silhouette Method

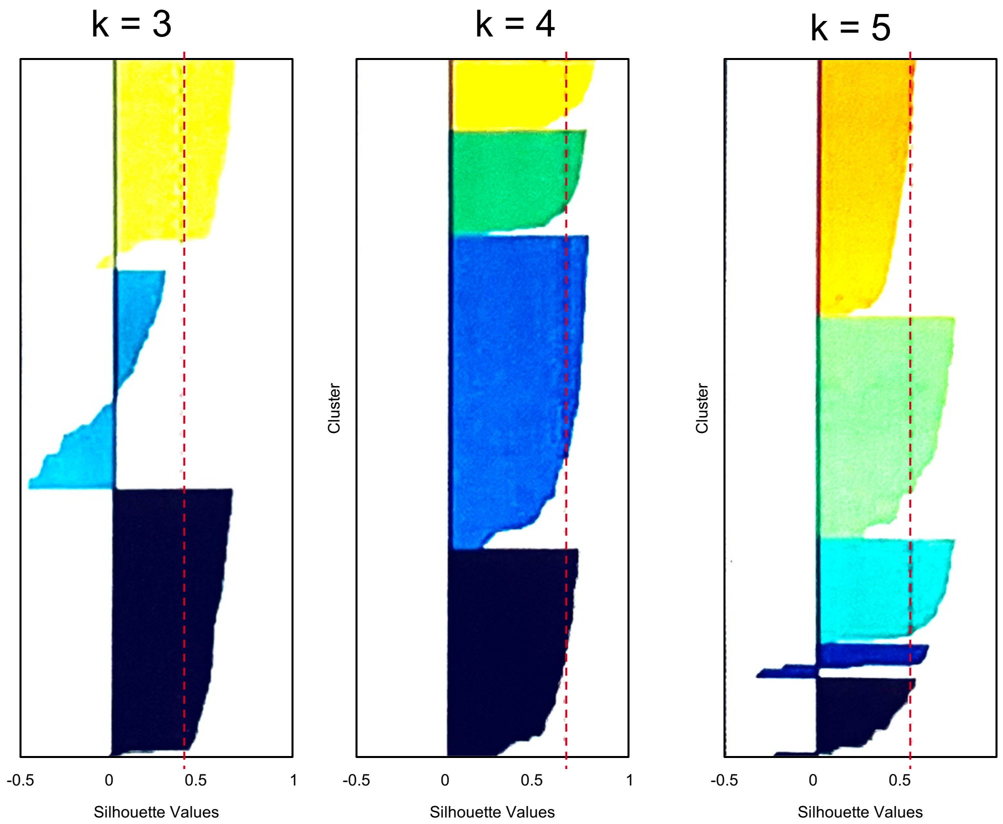
- Measures how close each point is in one cluster compared to points in neighbouring clusters
- Ranges from -1 to 1, higher values indicate good separation from other clusters
- Calculate silhouette values for all data points

Nearest-cluster distance **Intra-cluster distance**

$$s(i) = \frac{b(i) - a(i)}{\max\left(a(i), b(i)\right)}$$

Silhouette method

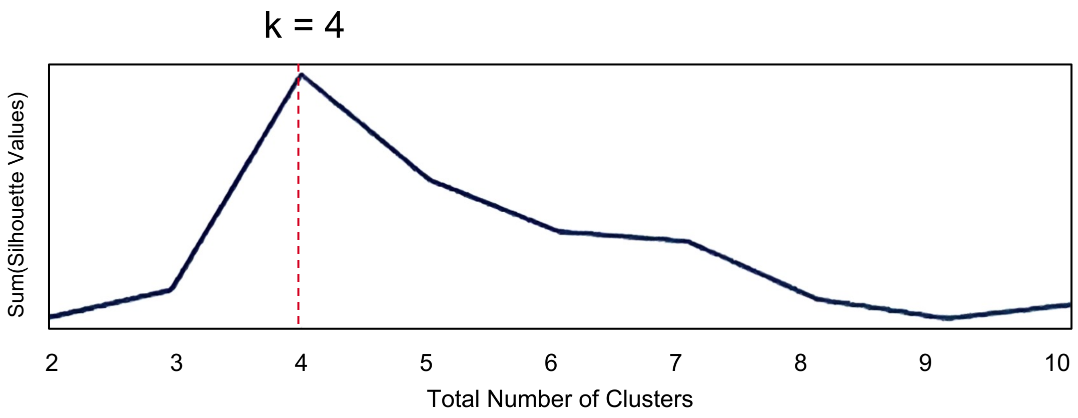
- Silhouette value plots show quality of clustering process
- Negative values indicate that points should belong to a different cluster
- The higher the overall (average) silhouette value, the better



Silhouette method

- Plot total (or average) silhouette value for values of k to find the best value for k

→ The higher the total (average) silhouette value, the better the clustering.



K-Means Algorithm – Pros & Cons

Pros:

- Very fast
- Scales well
- New points can be assigned to a cluster without training a new model
- General purpose
- Easy to understand

Cons:

- Sensitive to outliers
- Number of clusters must be known in advance
- Limited cluster shapes (all clusters must be spherical with same radius)
- Result and performance depends on the random start points
- Tends to produce clusters of the same size

K-Means: Quick recap

- K-Means is a very powerful and fast general-purpose clustering algorithm for partitioning datasets into clusters.
- Assigns observations to randomly chosen centroids and then re-assigns them based on the minimal distance to each cluster centroid until recalculation of centroids does not change their position.
- Can only detect simple spherical / convex shapes
- Sensitive to outliers
- Widely used, often the first clustering algorithm to start with.

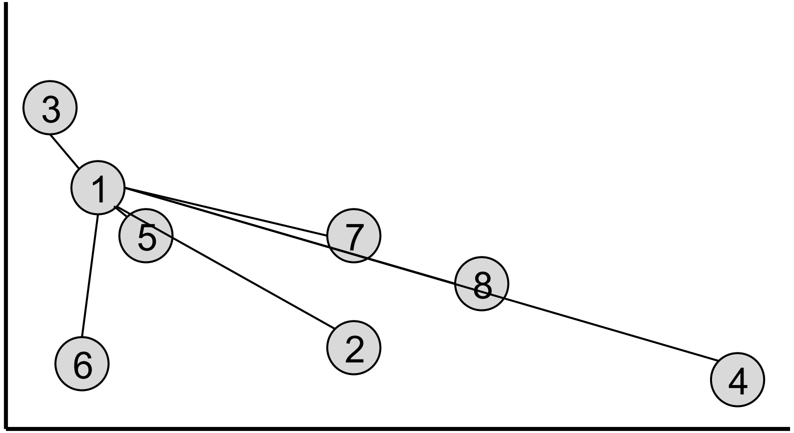
Hierarchical Clustering

Introduction

- Understand key concepts and terminology
- Understand clustering algorithms work
- How algorithms create clusters step by step
- Pros and cons of hierarchical clustering

Distance Matrix

- Gives us the distance between each pair of data points
- Different ways to measure distance *d*



	1	2	3	4	5	6	7	8
1	0	0.8	0.2	1.9	0.1	0.6	0.7	0.9
2		0	…	…	…	…	…	…
3			0	…	…	…	…	…
4				0	…	…	…	…
5					0	…	…	…
6						0	…	…
7							0	…
8								0

Distance Matrix - 1-dimensional data

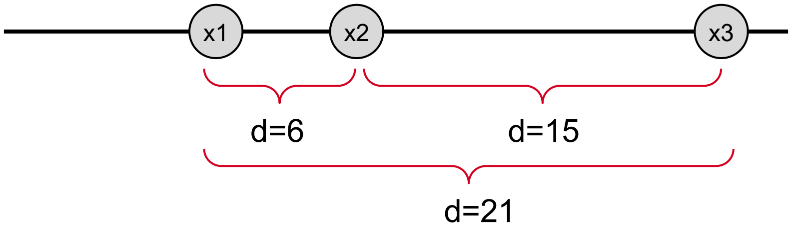
- The distance matrix will always be 2-dimensional, no matter the dimensionality of the underlying data!

Example: 1-dimensional data

x1 = 14

x2 = 20

x3 = 35



	x1	x2	x3
x1	0	6	21
x2	6	0	15
x3	21	15	0

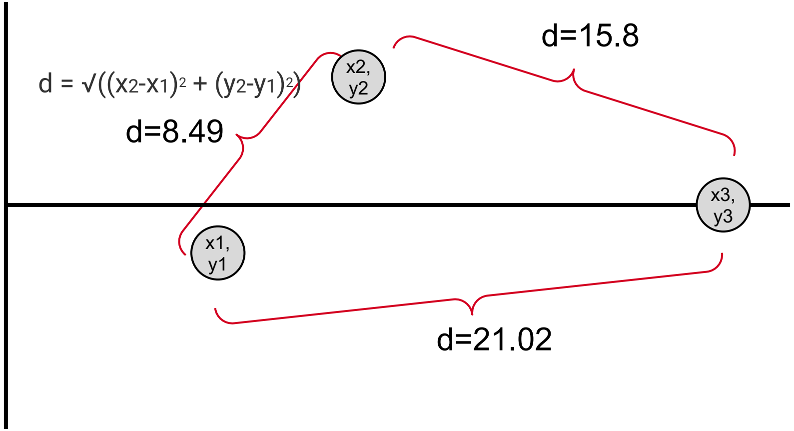
Distance Matrix - 2-dimensional data

Example: Euclidean distance for 2-dimensional data

(x1, y1) = (x1, y1) = (14, -1)

(x2, y2) = (x2, y2) = (20,5)

(x3, y3) = (x3, y3) = (35, 0)



	(x1, y1)	(x2, y2)	(x3, y3)
(x1, y1)	0	8.49	21.02
(x2, y2)	8.49	0	15.8
(x3, y3)	21.02	15.8	0

Cluster Distance

How do we measure the difference between clusters that contain multiple points?

→ **Linkage criteria!**

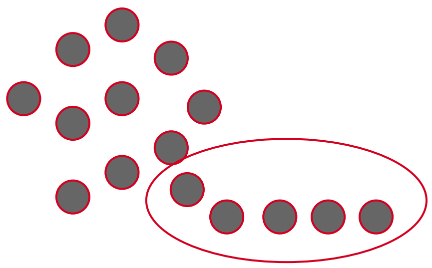
Linkage criteria

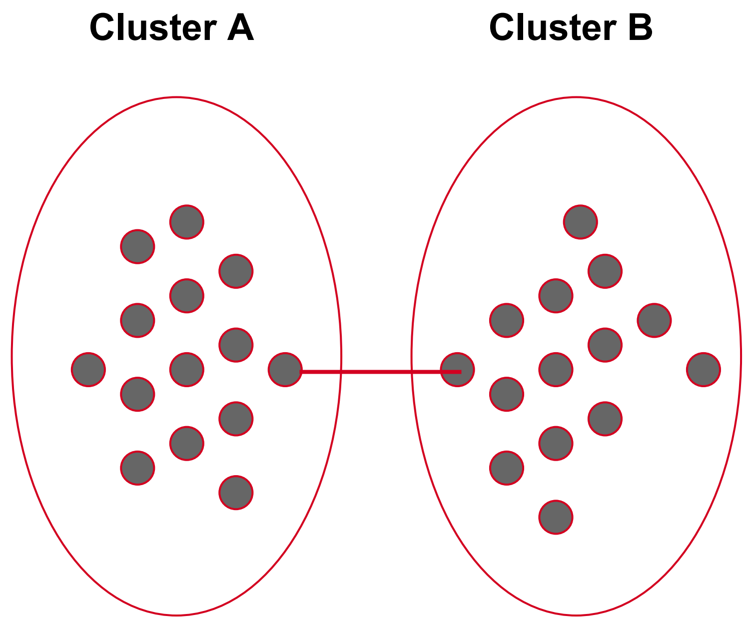
Single Linkage:

Minimal distance between the points of each cluster

$$d(C_{(ij)}, C_k) = \min\{d(C_i, C_k), d(C_j, C_k)\}$$

Problem: Chaining effect





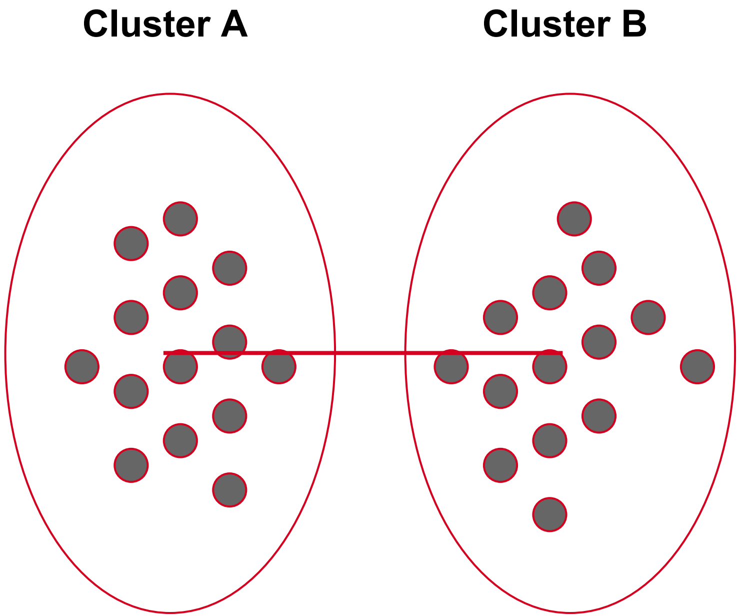
Linkage criteria

Average Linkage:

Average distance between the points of each cluster

$$d(C_{(ij)}, C_k) = \frac{d(c_i, c_k)|c_i| + d(c_j, c_k)|c_j|}{|c_i| + |c_j|}$$

Different methods to calculate the average (weighted, unweighted, centroid, …)



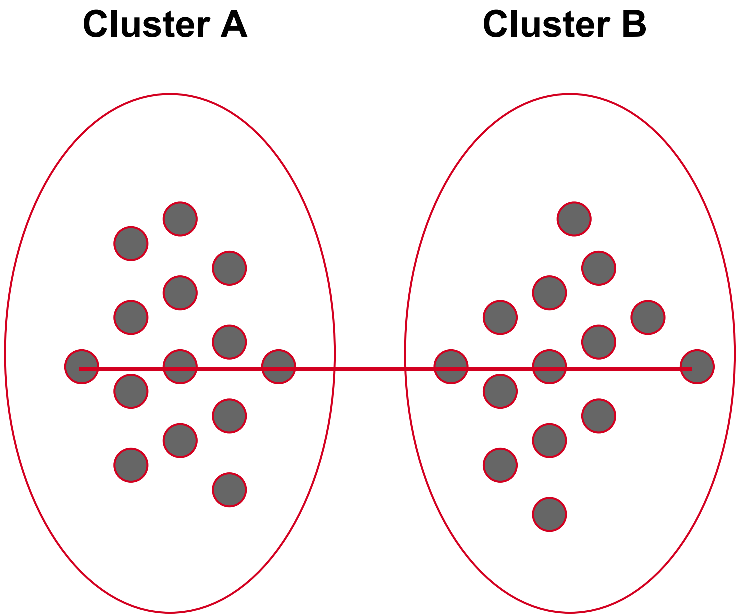
Linkage criteria

Complete Linkage:

Maximum distance between the points of each cluster

$$d(C_{(ij)}, C_k) = \max\{d(C_i, C_k), d(C_j, C_k)\}$$

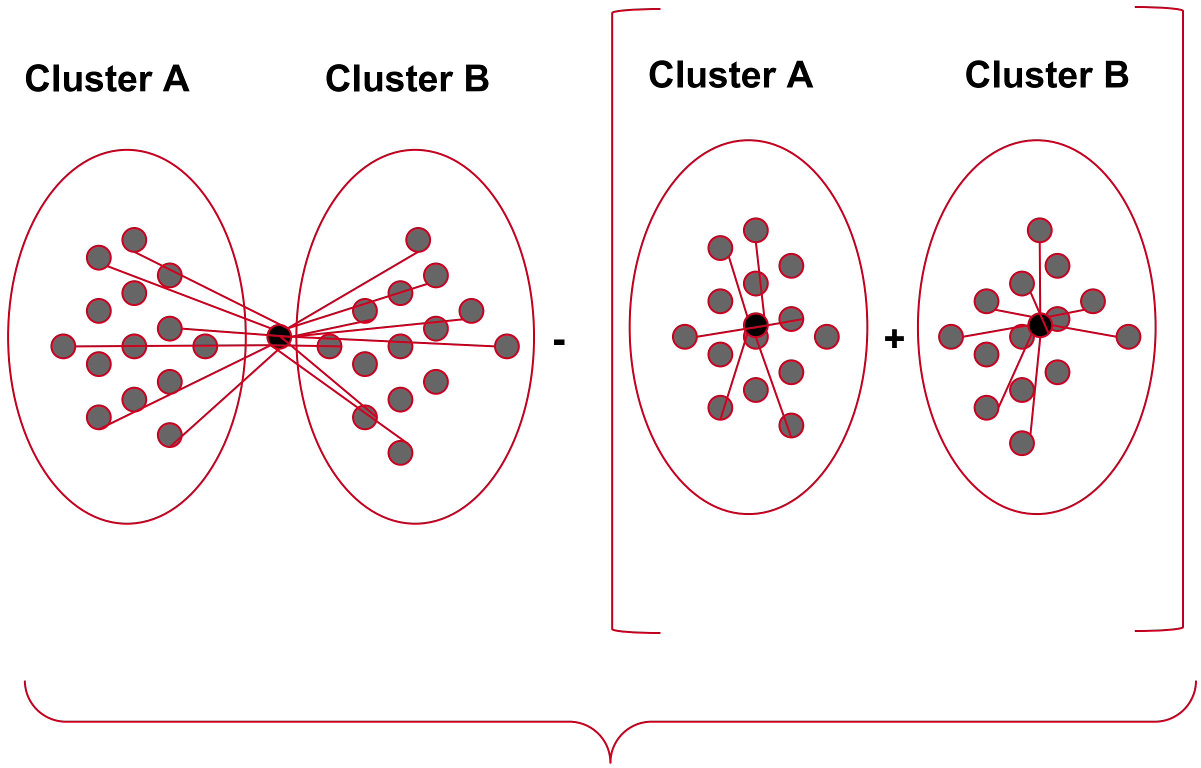
→ Tends to produce compact clusters



Linkage criteria

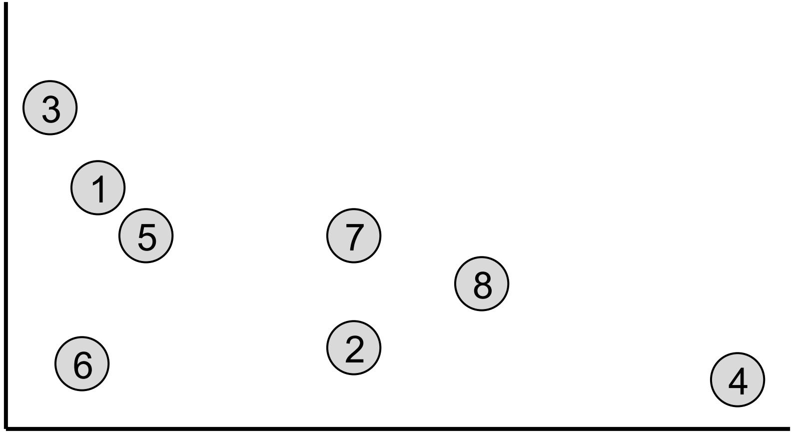
Ward method:

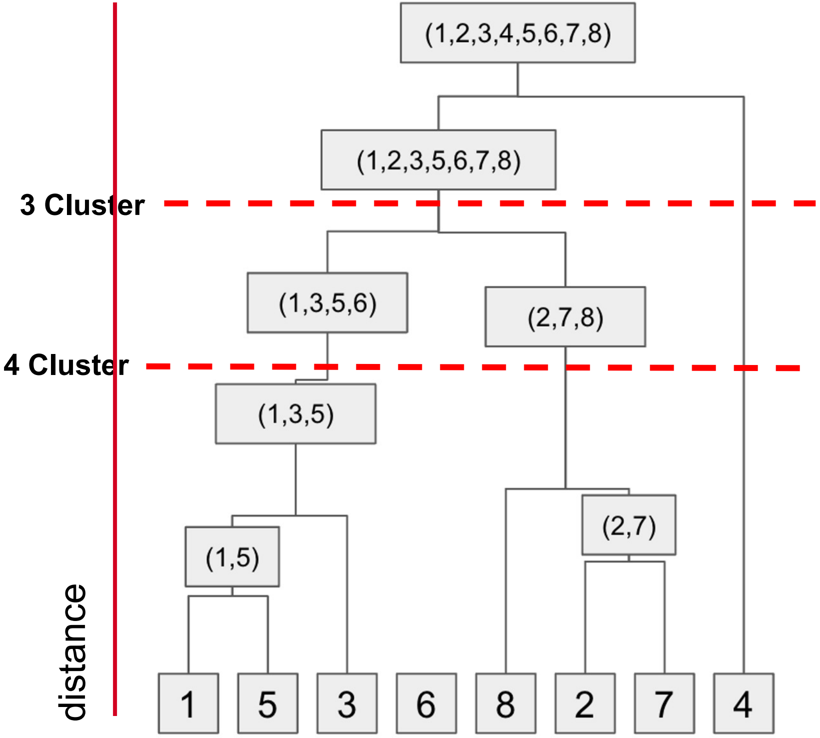
- Recursive linkage method
- Fuse clusters with only a few observations
- Relatively sensitive to outliers
- Attempts to minimize the within-cluster variance.
- More suited to detect clusters with non-spherical form or different diameters.



Dendrogram

- Returns a hierarchy of clusters
- Nodes are (sets of) examples
- Clusters can be obtained by cutting tree at a selected height d
- Select the number of clusters where either
  (A) the vertical distance between two merge clusters is the highest, or
  (B) the number of clusters you want to have is reached.





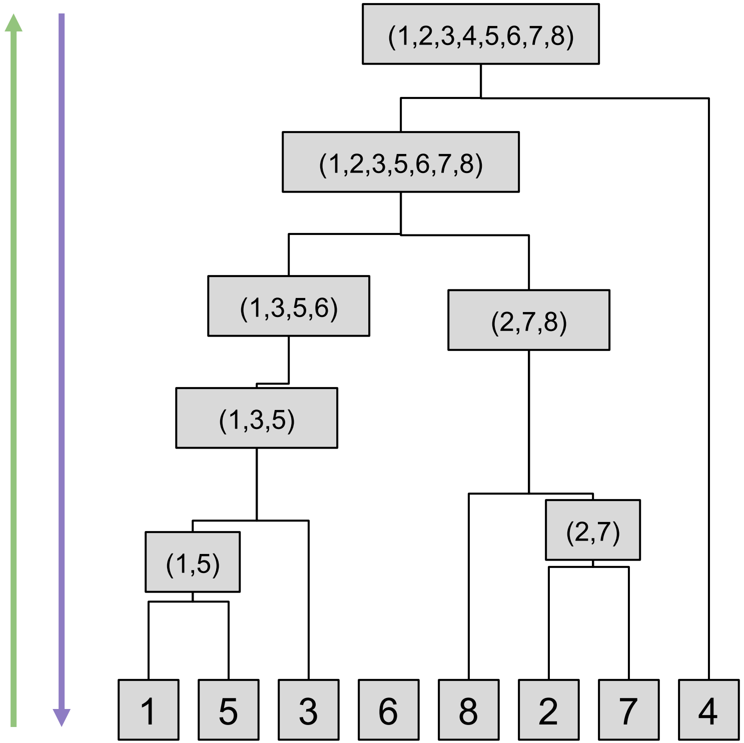
Agglomerative vs. Divisive Clustering

Agglomerative Clustering

- Bottom-up
- n_points cluster at start
- Merges cluster
- Computationally expensive

Divisive Clustering

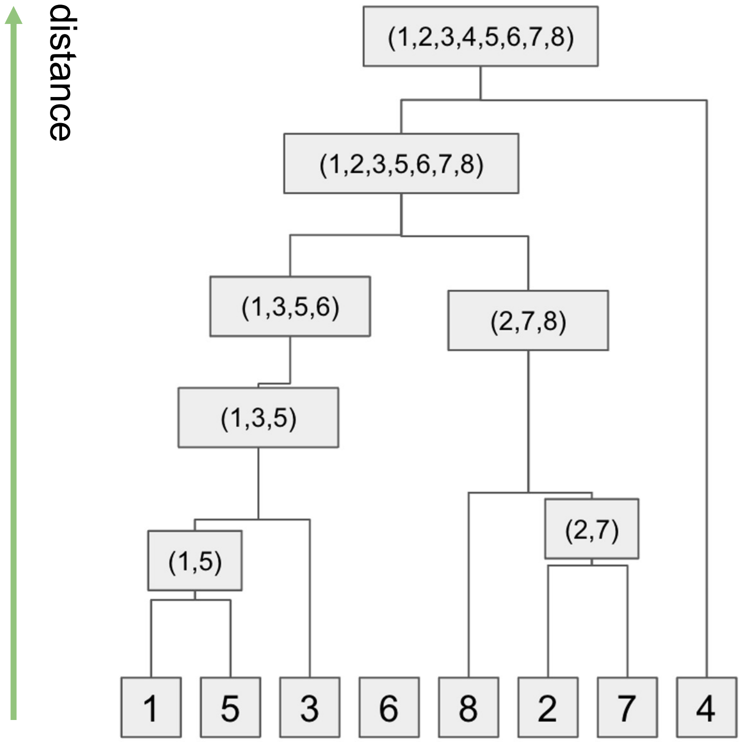
- Top-down
- One cluster at start
- Splits cluster
- even more expensive



Agglomerative Clustering

Learning process:

- Put every example in a single cluster
- While n_clusters < 1:
  - Select two clusters with minimum distance
  - Join these two clusters



Agglomerative Clustering Example

Cluster Cities by Distance

Step 1: Get Distances

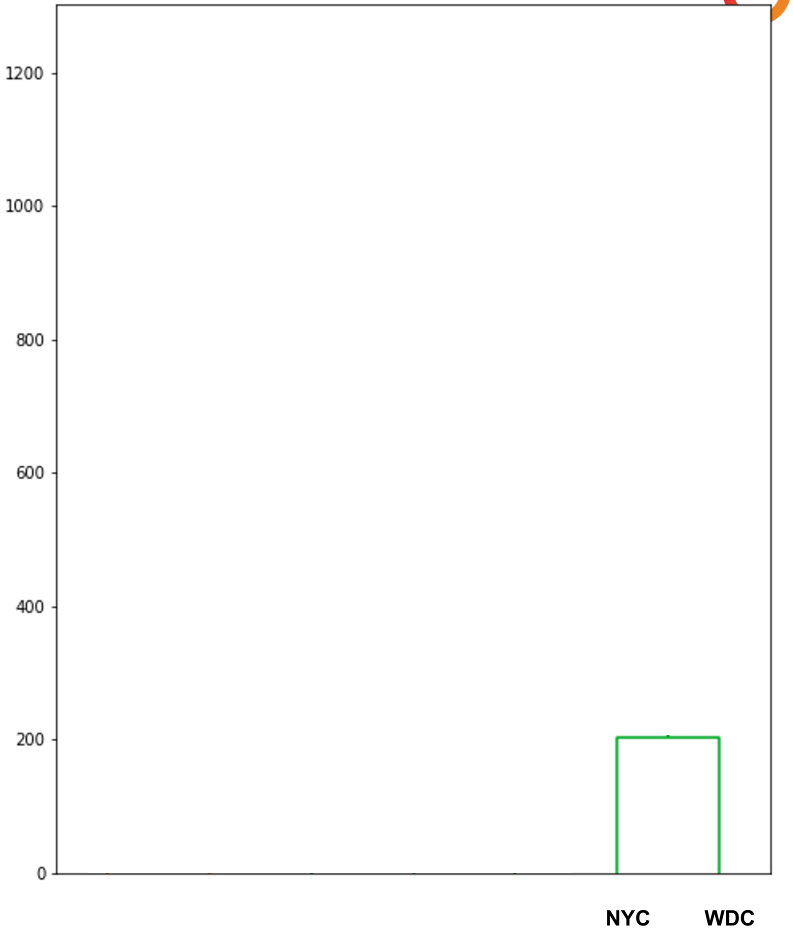
City	Coordinates	Country
New York	40.7128° N, 74.0060° W	USA
Washington DC	38.9072° N, 77.0369° W	USA
Chicago	41.8781° N, 87.6298° W	USA
San Francisco	37.7749° N, 122.4194° W	USA
Los Angeles	34.0522° N, 118.2437° W	USA
Dallas	32.7767° N, 96.7970° W	USA
Houston	29.7604° N, 95.3698° W	USA
Vancouver	49.2827° N, 123.1207° W	Canada
Montréal	45.5019° N, 73.5674° W	Canada
Toronto	43.6532° N, 79.3832° W	Canada
Mexico City	19.4326° N, 99.1332° W	Mexico

Distance Matrix

	New York	Washington DC	Chicago	San Francisco	Los Angeles	Dallas	Houston
New York	0	204	715	2,565	2,455	1,370	1,415
Washington		0	595	2,435	2,300	1,170	1215
Chicago			0	1,850	1,745	800	940
San Francisco				0	350	1,460	1,635
Los Angeles					0	1,230	1,370
Dallas						0	225
Houston							0

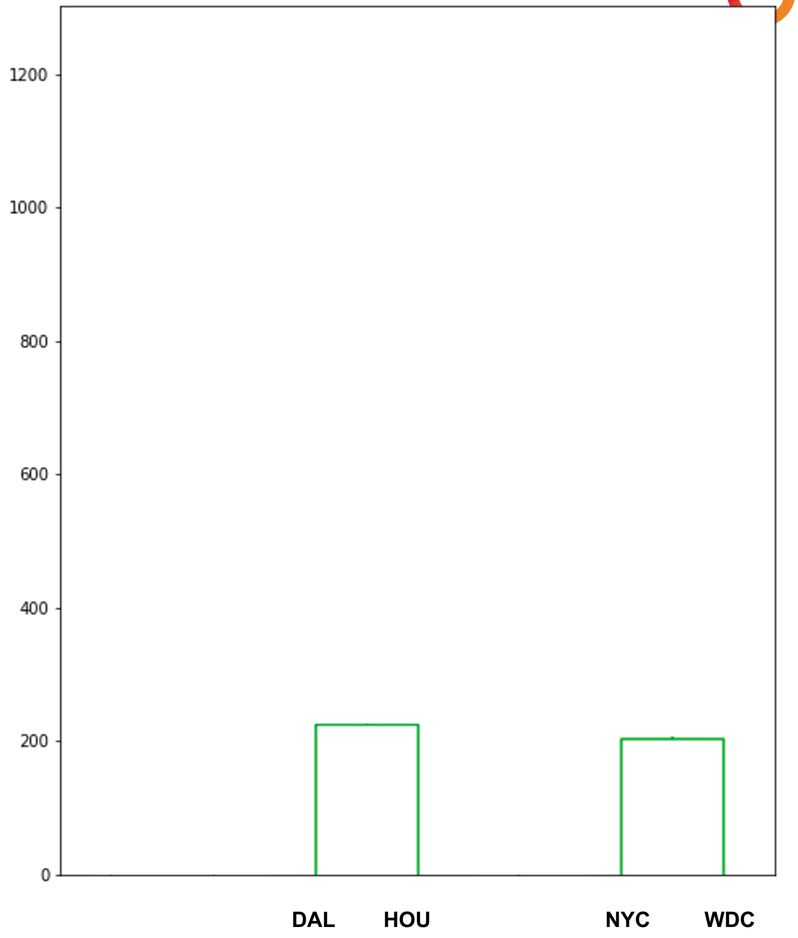
Hierarchical Clustering

	NYC	WDC	CHI	SFO	LOS	DAL	HOU
NYC	0	**204**	713	2,572	2,451	1,373	1,419
WDC		0	595	2,441	2,299	1,184	1220
CHI			0	1,859	1,746	804	941
SFO				0	347	1,484	1,645
LOS					0	1,240	1,373
DAL						0	224
HOU							0



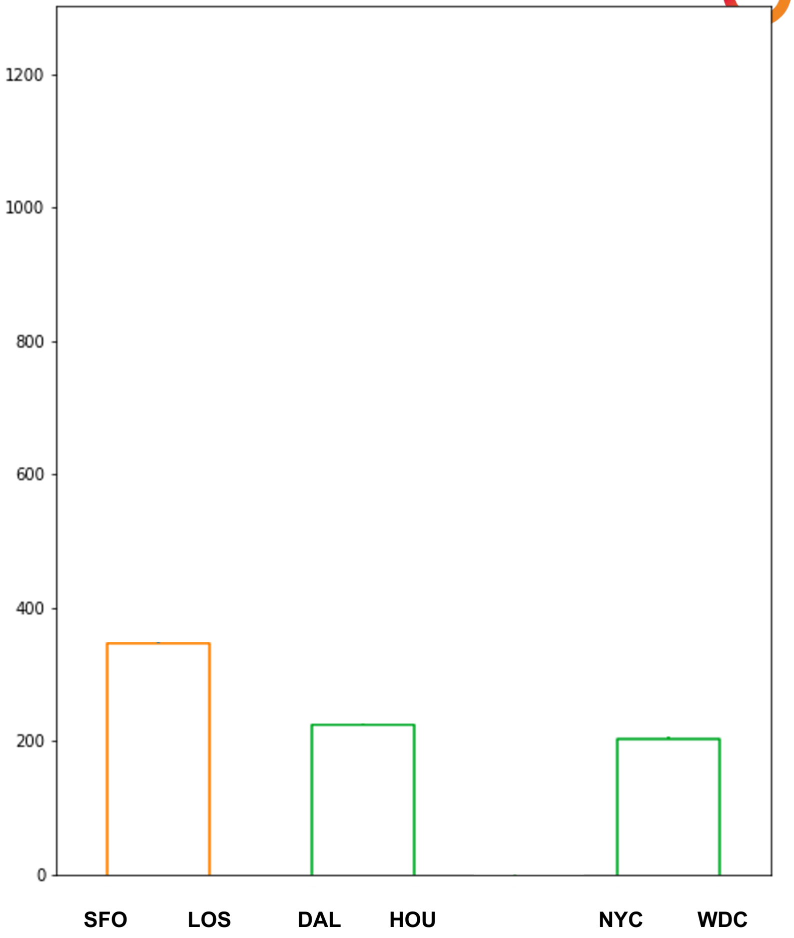
Hierarchical Clustering

	NYC	WDC	CHI	SFO	LOS	DAL	HOU
NYC	0	204	713	2,572	2,451	1,373	1,419
WDC		0	595	2,441	2,299	1,184	1220
CHI			0	1,859	1,746	804	941
SFO				0	347	1,484	1,645
LOS					0	1,240	1,373
DAL						0	**224**
HOU							0



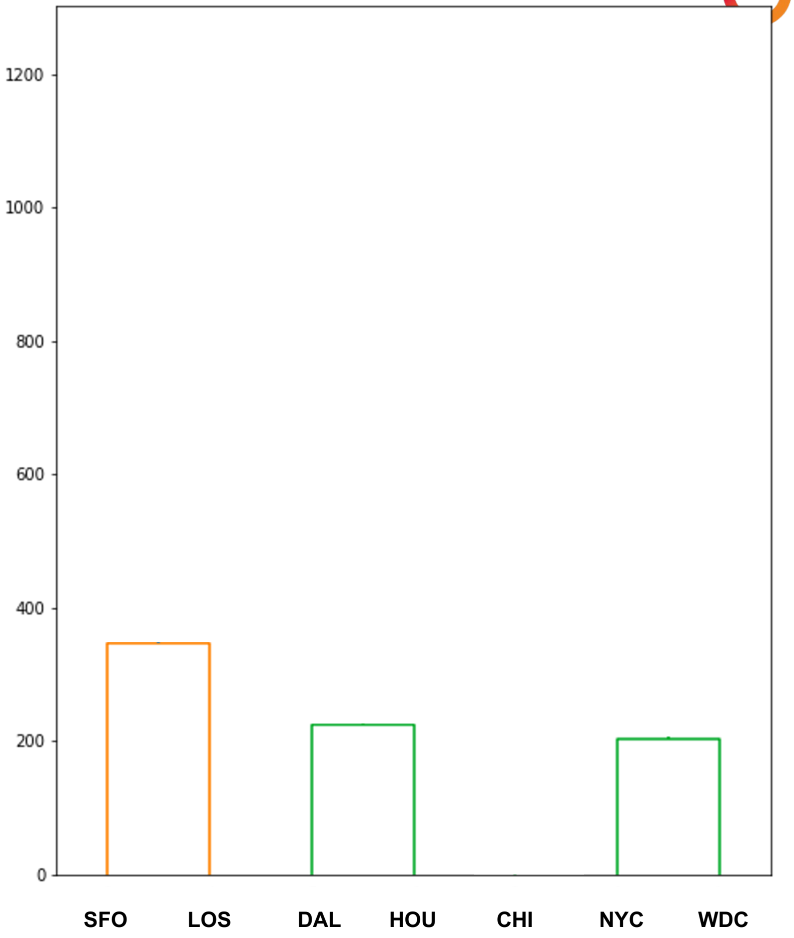
Hierarchical Clustering

	NYC	WDC	CHI	SFO	LOS	DAL	HOU
NYC	0	204	713	2,572	2,451	1,373	1,419
WDC		0	595	2,441	2,299	1,184	1220
CHI			0	1,859	1,746	804	941
SFO				0	**347**	1,484	1,645
LOS					0	1,240	1,373
DAL						0	224
HOU							0



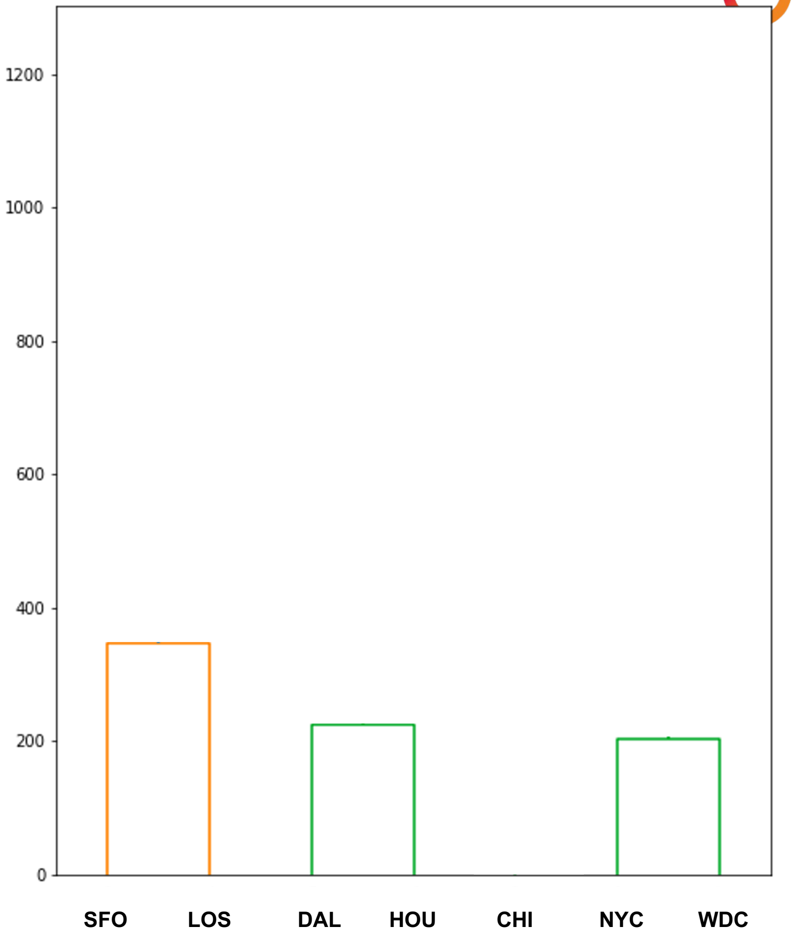
Hierarchical Clustering

	NYC	WDC	CHI	SFO	LOS	DAL	HOU
NYC	0	204	713	2,572	2,451	1,373	1,419
WDC		0	595	2,441	2,299	1,184	1220
CHI			0	1,859	1,746	804	941
SFO				0	347	1,484	1,645
LOS					0	1,240	1,373
DAL						0	224
HOU							0



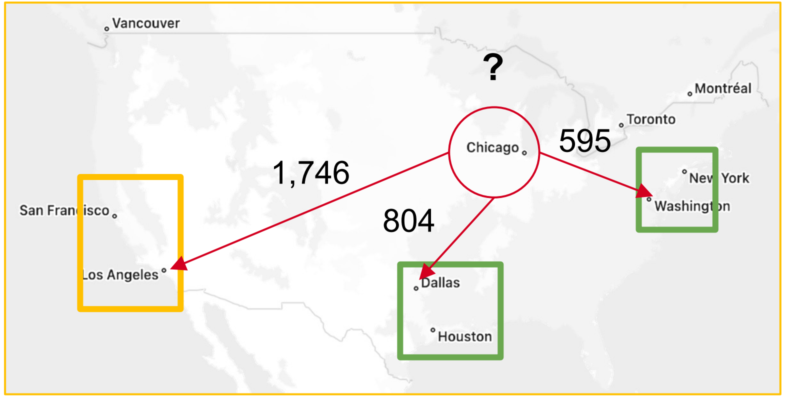
Hierarchical Clustering

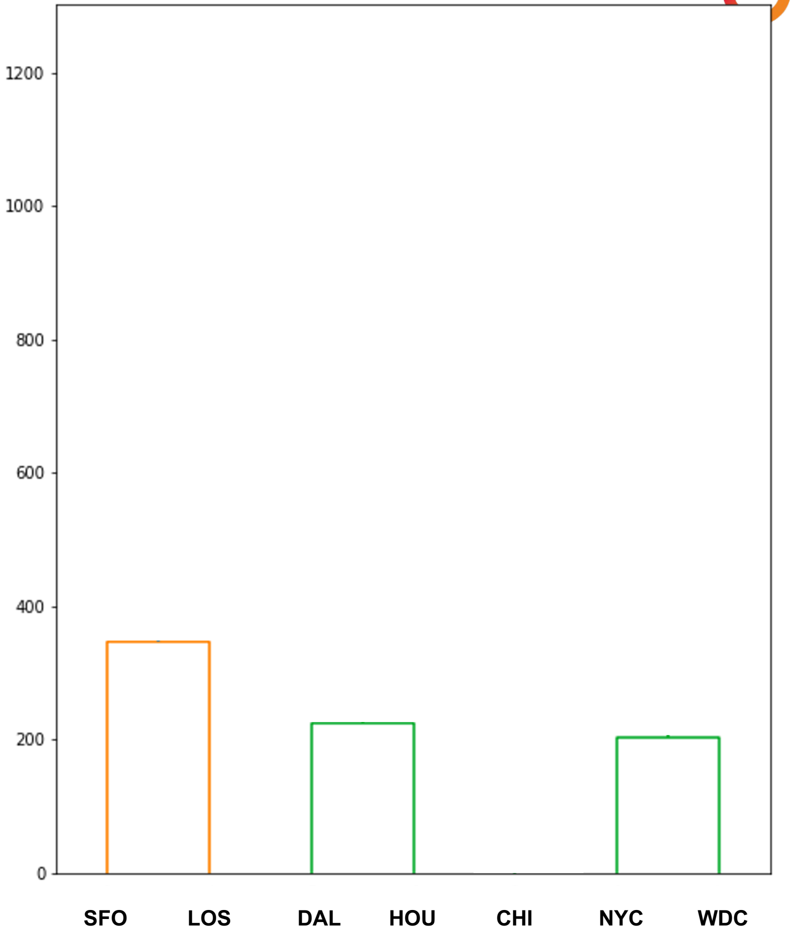
	NYC	WDC	CHI	SFO	LOS	DAL	HOU
NYC	0	204	713	2,572	2,451	1,373	1,419
WDC		0	**595**	2,441	2,299	1,184	1220
CHI			0	1,859	**1,746**	**804**	941
SFO				0	347	1,484	1,645
LOS					0	1,240	1,373
DAL						0	224
HOU							0



Hierarchical Clustering

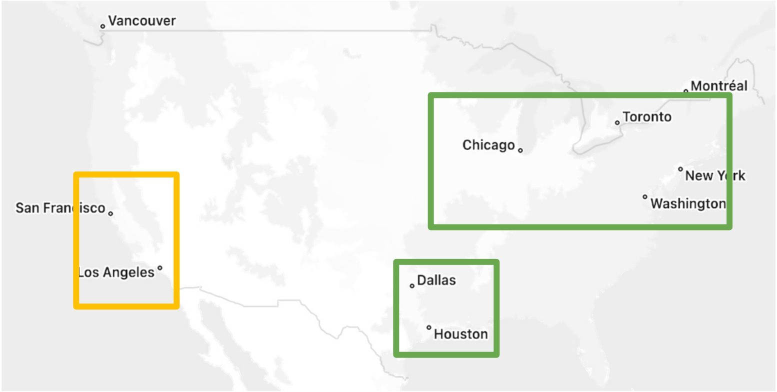
	NYC	WDC	CHI	SFO	LOS	DAL	HOU
NYC	0	204	713	2,572	2,451	1,373	1,419
WDC		0	**595**	2,441	2,299	1,184	1220
CHI			0	1,859	**1,746**	**804**	941
SFO				0	347	1,484	1,645
LOS					0	1,240	1,373
DAL						0	224
HOU							0

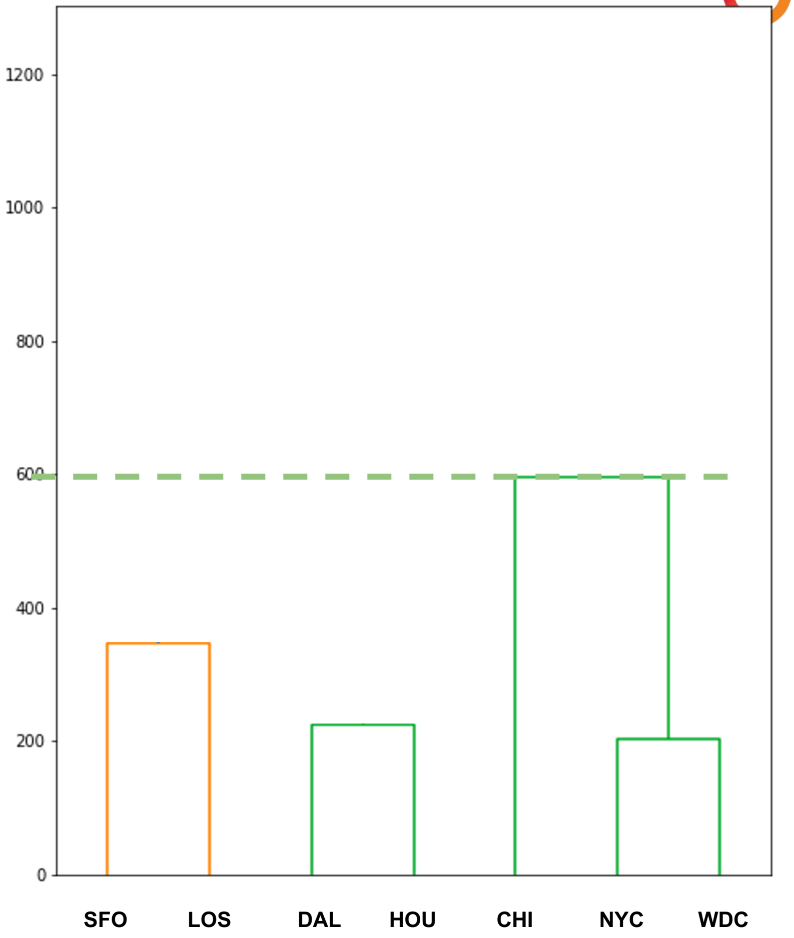




Hierarchical Clustering

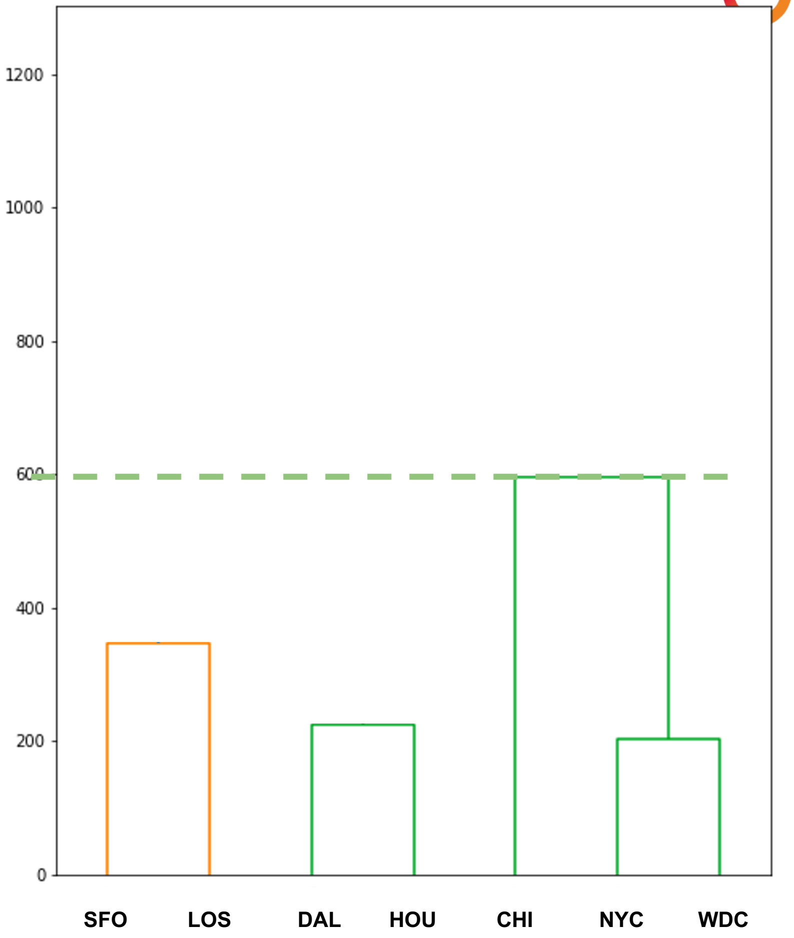
	NYC	WDC	CHI	SFO	LOS	DAL	HOU
NYC	0	204	713	2,572	2,451	1,373	1,419
WDC		0	595	2,441	2,299	1,184	1220
CHI			0	1,859	1,746	804	941
SFO				0	347	1,484	1,645
LOS					0	1,240	1,373
DAL						0	224
HOU							0





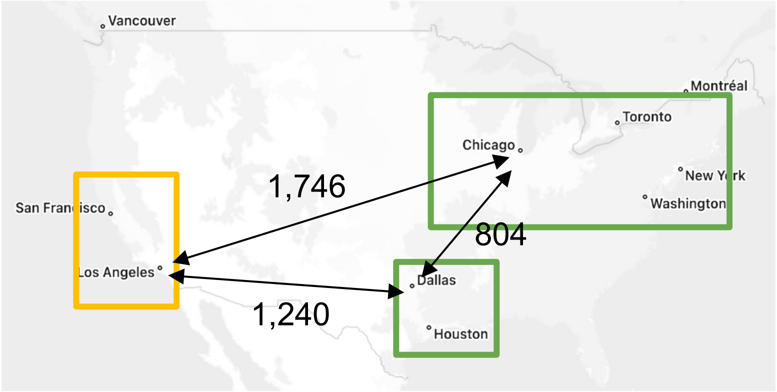
Hierarchical Clustering

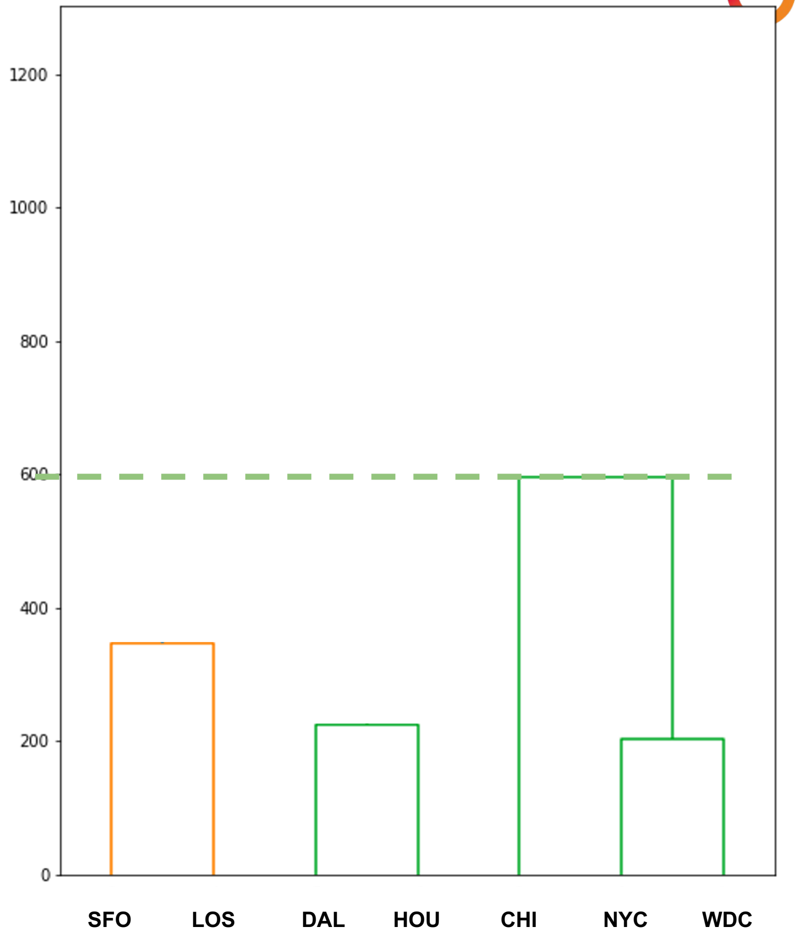
	NYC	WDC	CHI	SFO	LOS	DAL	HOU
NYC	0	204	713	2,572	2,451	1,373	1,419
WDC		0	595	2,441	2,299	1,184	1220
CHI			0	1,859	1,746	804	941
SFO				0	347	1,484	1,645
LOS					0	1,240	1,373
DAL						0	224
HOU							0



Hierarchical Clustering

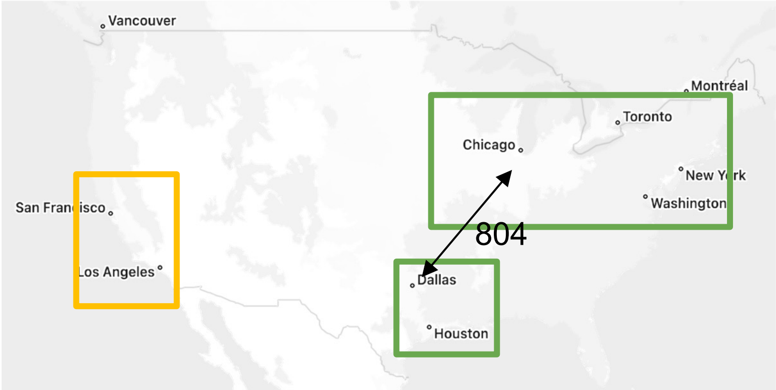
	NYC	WDC	CHI	SFO	LOS	DAL	HOU
NYC	0	204	713	2,572	2,451	1,373	1,419
WDC		0	595	2,441	2,299	1,184	1220
CHI			0	1,859	**1,746**	**804**	941
SFO				0	347	1,484	1,645
LOS					0	**1,240**	1,373
DAL						0	224
HOU							0

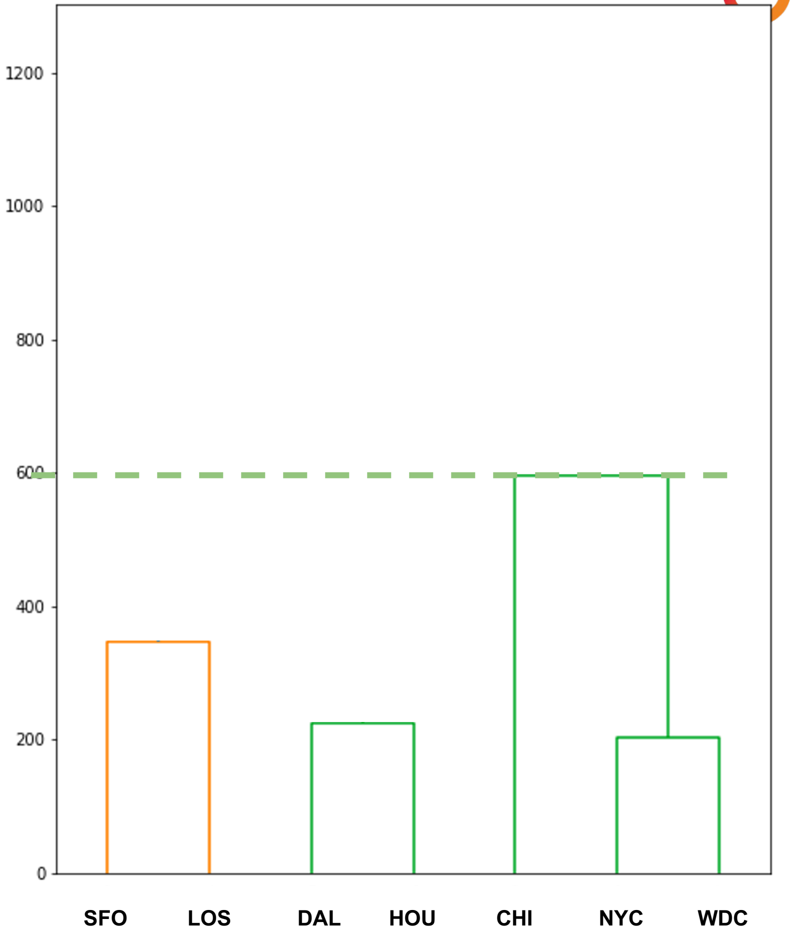




Hierarchical Clustering

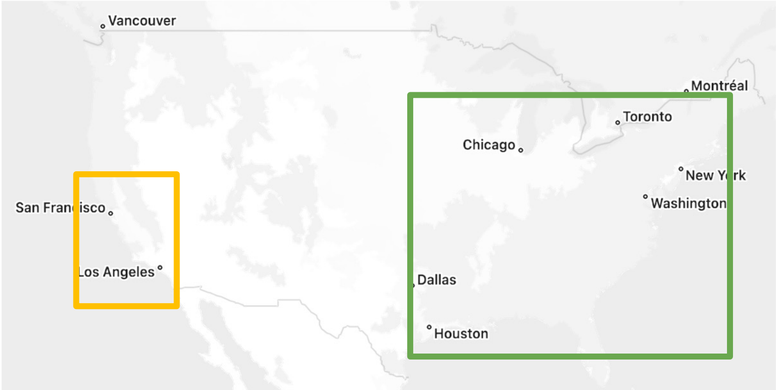
	NYC	WDC	CHI	SFO	LOS	DAL	HOU
NYC	0	204	713	2,572	2,451	1,373	1,419
WDC		0	595	2,441	2,299	1,184	1220
CHI			0	1,859	1,746	**804**	941
SFO				0	347	1,484	1,645
LOS					0	1,240	1,373
DAL						0	224
HOU							0

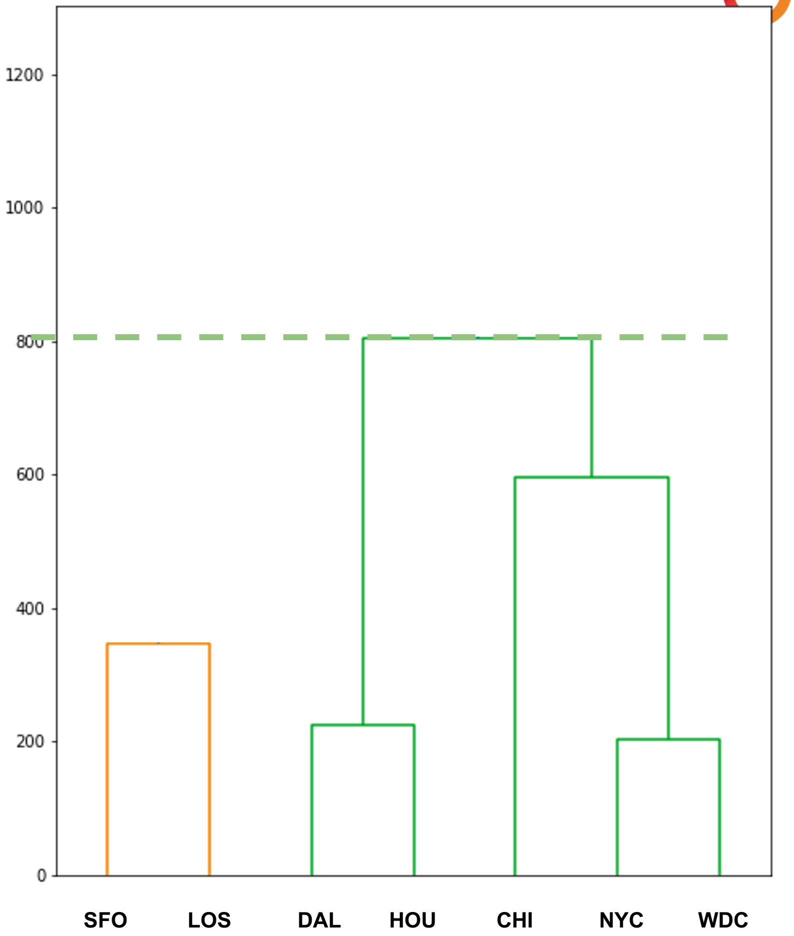




Hierarchical Clustering

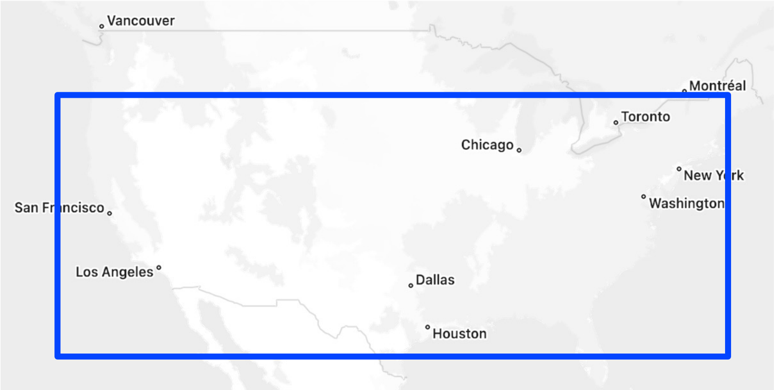
	NYC	WDC	CHI	SFO	LOS	DAL	HOU
NYC	0	204	713	2,572	2,451	1,373	1,419
WDC		0	595	2,441	2,299	1,184	1220
CHI			0	1,859	1,746	**804**	941
SFO				0	347	1,484	1,645
LOS					0	1,240	1,373
DAL						0	224
HOU							0

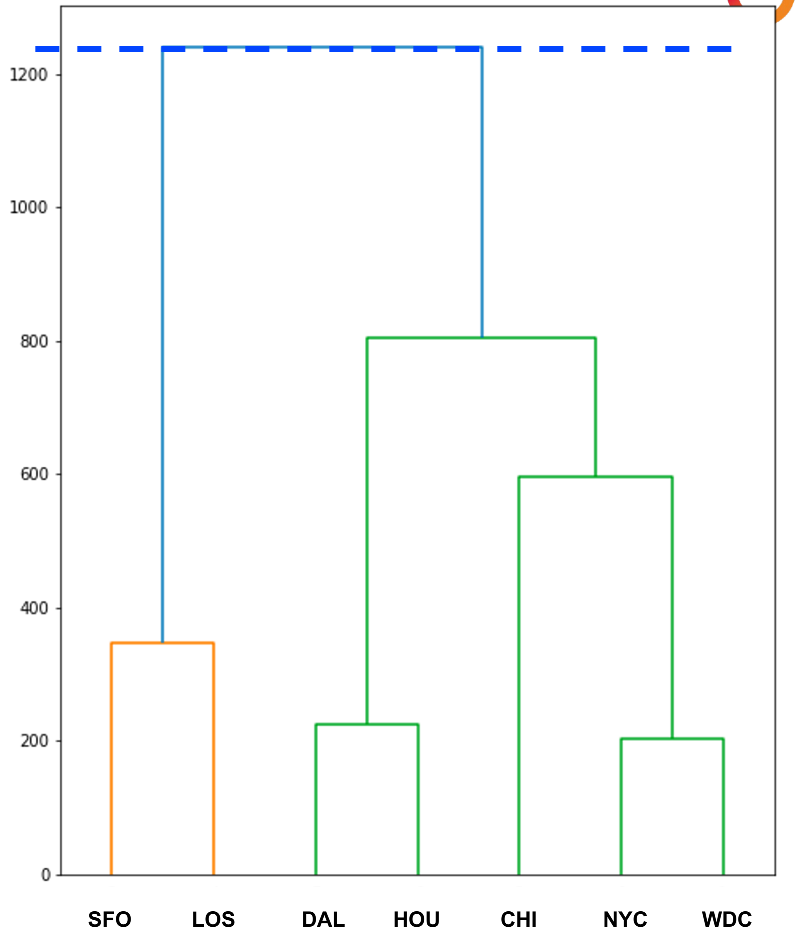




Hierarchical Clustering

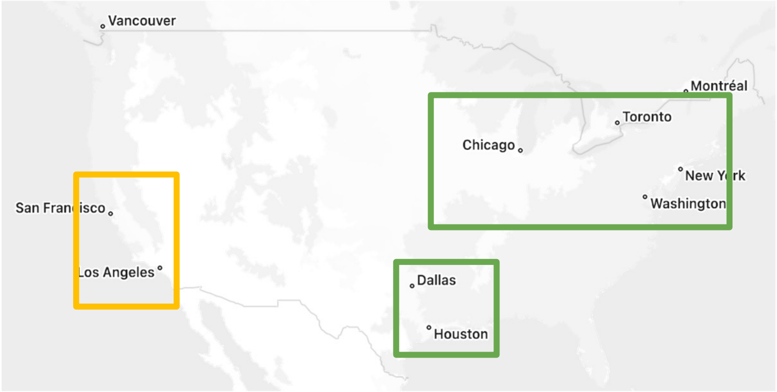
	NYC	WDC	CHI	SFO	LOS	DAL	HOU
NYC	0	204	713	2,572	2,451	1,373	1,419
WDC		0	595	2,441	2,299	1,184	1220
CHI			0	1,859	1,746	804	941
SFO				0	347	1,484	1,645
LOS					0	1,240	1,373
DAL						0	224
HOU							0

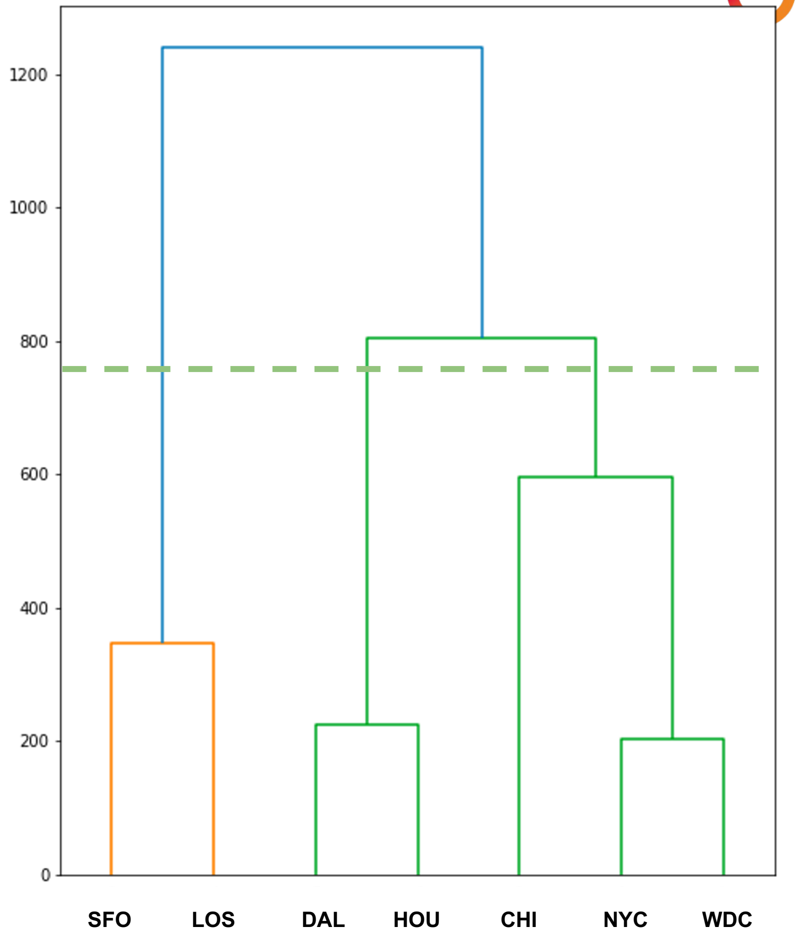




Hierarchical Clustering

	NYC	WDC	CHI	SFO	LOS	DAL	HOU
NYC	0	204	713	2,572	2,451	1,373	1,419
WDC		0	595	2,441	2,299	1,184	1220
CHI			0	1,859	1,746	804	941
SFO				0	347	1,484	1,645
LOS					0	1,240	1,373
DAL						0	224
HOU							0





Hierarchical Clustering – Pros & Cons

Pros

- Can handle many clusters well
- Flexible in choice of cluster metric
- Works well with uneven cluster sizes
- Detects outliers
- Possible connectivity constraints
- Good overview of the resulting clusters based on the threshold *distance*

Cons

- Computationally expensive, not suited for big data
- Doesn’t work well with complex shapes
- New points can’t be assigned to a cluster without recalculating the whole model
- Number of clusters must be defined manually
- Complex analysis of the result

Hierarchical Clustering – Recap

- Difference between divisive and agglomerative algorithms.
- How agglomerative clustering works step by step
- Major advantages and disadvantages of hierarchical clustering.

Interactive labs: Develop and Interpret a Hierarchical & k-Means Clustering

Q&A

How do you feel?

Rule mining

“Beer and Diaper” example

- Probably more a modern myth of data analysis, but nevertheless drives home to main point:
- Supermarket wanted to analyze customer shopping behavior and looked at shopping carts transactional data
- Retailer found out that customers who bought diapers were likely to buy beer as well.

→ They put beer bottles next to the diaper stand and could significantly increase beer sales

→ Association Rule mining helped getting these insights

- Image source: https://jborden.com/2018/12/07/beer-and-diapers-the-perfect-couple/

Other Examples

→ More than a correlation analysis: Provides “if-then” rules

Example application areas:

- **Recommendation systems**: Identify items that are frequently purchased or consumed together to give better recommendations
- **Finance**: Identify patterns in financial transactions, e.g. relationship between different stocks to make better investment decisions
- **Web analytics**: Find out pages that are often visited together and improve site navigation
- **Bioinformatics**: Identify DNA patterns that might indicate certain medical conditions
- Intrusion detection: Detect unusual patterns of network activity that might indicate a security breach
- …

Association Rule Learning

→ **Goal**: Discover interesting patterns and relationships between variables in large (transactional) datasets

- Basic idea: Find rules that show relationships between different items in dataset (e.g. when item A appears, then item B is likely to appear as well)

- Strong background in retail, but not limited to that.

Definitions

I… set of *n* binary **items *i*** *(no quantities!)*

D… set of transactions *t* (**database**)

Each transaction in *D* contains a subset of the items in *I*.

Rule: $X \Rightarrow Y$ where $X, Y \subseteq I$

(Read as "If X then Y" (E.g. Diapers => Beer)

Typically: *X* multiple items and *Y* single item

X… antecedent (LHS)
Y… consequent (RHS)

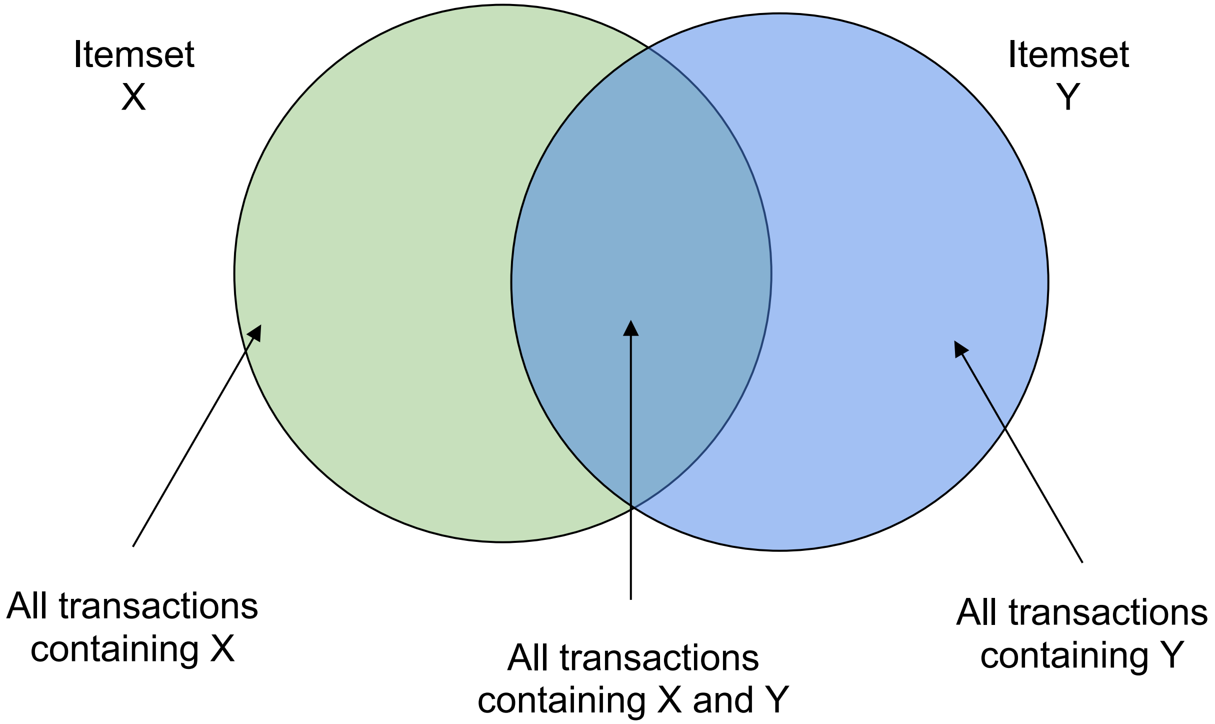
$$I = \{i_1, i_2, \ldots, i_n\}$$

$$D = \{t_1, t_2, \ldots, t_m\}$$

How to find rules in a dataset

Algorithms:
- Apriori
- FP Growth

Important concepts:
- Support
- Confidence
- Lift



Quick recap:

- Association rule mining is a method used to identify relationships between different items in a dataset.

- Can be applied to many different fields such as recommendation systems, web usage mining, bioinformatics and intrusion detection.

- Main concepts used in association rule mining are support, confidence and lift.

Support, Confidence & Lift

- How do we calculate support, confidence and lift?
- What’s their interpretation?

Sample Data

Transaction ID	🍉	🍎	🍏	🍒	🍓	🥕	🍍
1	1	1	0	0	0	0	1
2	0	0	1	0	0	1	1
3	0	0	0	1	1	0	0
4	1	1	1	0	0	1	1
5	0	1	0	0	0	0	0

I = {🍉, 🍎, 🍏, 🍒, 🍓, 🥕, 🍍}

Sample Data

Transaction ID	🍉	🍎	🍏	🍒	🍓	🥕	🍍
1	1	1	0	0	0	0	1
2	0	0	1	0	0	1	1
3	0	0	0	1	1	0	0
4	1	1	1	0	0	1	1
5	0	1	0	0	0	0	0

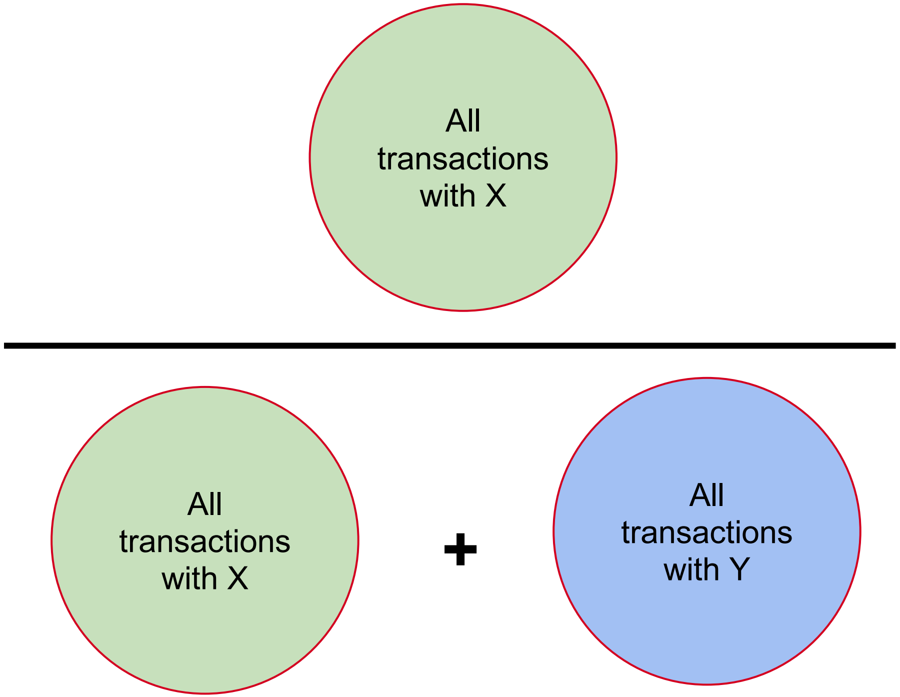
Reading example: Transaction 5 contains only 1 item (item 🍎)

Example Rule: {🍎, 🍏} => {🍉}

Intuition: Support

$$= \frac{\text{Transactions containing X}}{\text{All transactions}}$$

→ Measures the **frequency** of a particular combination of **items** in the dataset



Support

→ Let's consider an example…

ID							
1	1	1	0	0	0	0	1
2	0	0	1	0	0	1	1
3	0	0	0	1	1	0	0
4	1	1	1	0	0	1	1
5	0	1	0	0	0	0	0

$$\text{Support}\,(X) = \frac{\text{Number of transactions with } X}{\text{Total number of transactions}}$$

Support

→ Measures the **frequency** of a particular combination of **items** in the dataset

$X = \{$🍒, 🍓$\}$

ID	🍉	🍎	🍏	🍒	🍓	🥕	🍍
1	1	1	0	0	0	0	1
2	0	0	1	0	0	1	1
3	0	0	0	1	1	0	0
4	1	1	1	0	0	1	1
5	0	1	0	0	0	0	0

$$\text{Support}\,(X) = \frac{\text{Number of transactions with } X}{\text{Total number of transactions}}$$

$$\text{Support}(X) = \frac{1}{5} = 0.2$$

Support

→ Measures the **frequency** of a particular **rule** in the dataset

$X = \{$🍉$\}$
$Y = \{$🍎$\}$

ID	🍉	🍎	🍏	🍒	🍓	🥕	🍍
1	1	1	0	0	0	0	1
2	0	0	1	0	0	1	1
3	0	0	0	1	1	0	0
4	1	1	1	0	0	1	1
5	0	1	0	0	0	0	0

$$\text{Support}\,(X \Rightarrow Y) = \frac{\text{Number of transactions with X and Y}}{\text{Total number of transactions}}$$

$$\text{Support}(X \Rightarrow Y) = \frac{2}{5} = 0.4 = 40\%$$

Support

Support gives us a way to **filter** the most frequent items using a minimum threshold.

e.g., Minimum Support = 40%

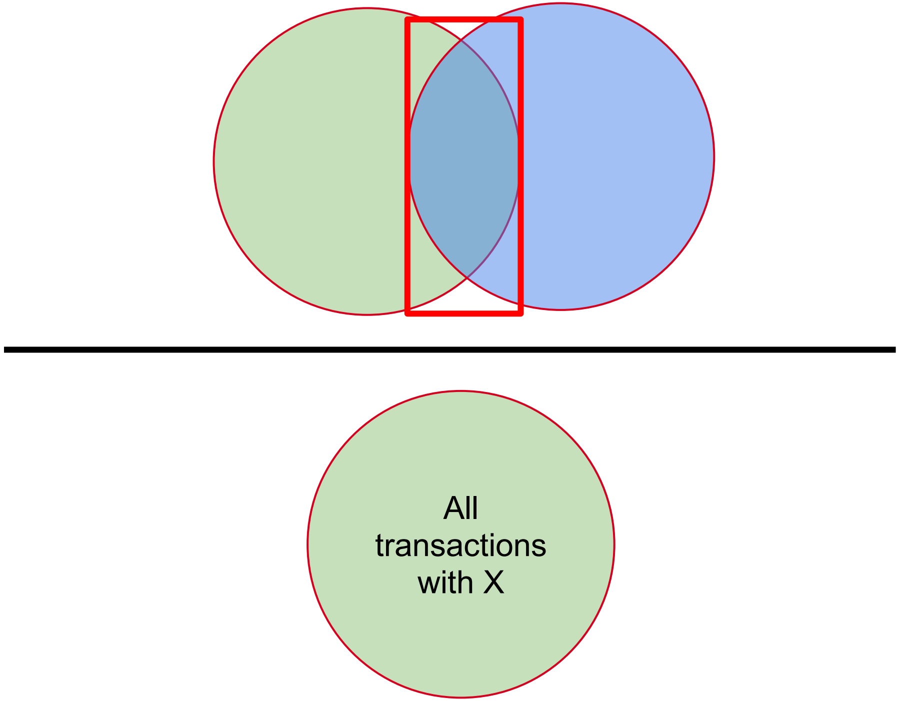
Item	Frequency	Support
🍉	2	0.4
🍎	3	0.6
🍏	2	0.4
🍒	1	0.2
🍓	1	0.2
🥕	2	0.4
🍍	3	0.6

ID	🍉	🍎	🍏	🍒	🍓	🥕	🍍
1	1	1	0	0	0	0	1
2	0	0	1	0	0	1	1
3	0	0	0	1	1	0	0
4	1	1	1	0	0	1	1
5	0	1	0	0	0	0	0

Confidence Rule X => Y

$$= \frac{\text{Transactions containing X and Y}}{\text{Transactions containing X}}$$

Measures the **ratio** of one itemset to another itemset**.**



Confidence

Let's consider an example…

X = {🍎, 🍏}
Y = {🍉}

ID	🍉	🍎	🍏	🍒	🍓	🥕	🍍
1	1	1	0	0	0	0	1
2	0	0	1	0	0	1	1
3	0	0	0	1	1	0	0
4	1	1	1	0	0	1	1
5	0	1	0	0	0	0	0

$$\text{Confidence}\,(X \Rightarrow Y) = \frac{\text{Number of transactions with X and Y}}{\text{Number of transactions with X}}$$

$$\text{Confidence}(X \Rightarrow Y) = \frac{1}{1} = 1 = 100\%$$

Confidence

Let's consider an example…

X = {🍎, 🍏}
Y = {🍉}

ID	🍉	🍎	🍏	🍒	🍓	🥕	🍍
1	1	1	0	0	0	0	1
2	0	0	1	0	0	1	1
3	0	0	0	1	1	0	0
4	1	1	1	0	0	1	1
5	0	1	0	0	0	0	0

$$\text{Confidence }(X \Rightarrow Y) = \frac{\text{Number of transactions with X and Y}}{\text{Number of transactions with X}}$$

Good rule?

$$\text{Confidence}(X \Rightarrow Y) = \frac{1}{1} = 1 = 100\%$$

Confidence + Support

Example:

Minimum support = 40%

→ Itemset {🍎, 🍏} has only 20% support

→ Itemset would be discarded, no rule would be created.

$$\boxed{\text{Support}(X) = \frac{1}{5} = 0.2 = 20\%}$$

ID	🍉	🍎	🍏	🍒	🍓	🥕	🍍
1	1	1	0	0	0	0	1
2	0	0	1	0	0	1	1
3	0	0	0	1	1	0	0
4	1	1	1	0	0	1	1
5	0	1	0	0	0	0	0

$$\text{Confidence}\,(X \Rightarrow Y) = \frac{\text{Number of transactions with X and Y}}{\text{Number of transactions with X}}$$

X = {🍎, 🍏}
Y = {🍉}

$$\text{Confidence}(X \Rightarrow Y) = \frac{1}{1} = 1 = 100\%$$

Confidence + Support

ID	🍉	🍎	🍏	🍒	🍓	🥕	🍍
1	1	1	0	0	0	0	1
2	0	0	1	0	0	1	1
3	0	0	0	1	1	0	0
4	1	1	1	0	0	1	1
5	0	1	0	0	0	0	0

- **Use Support and Confidence together to find strong rules**

- But how exactly and in which order?
  → **A priori Algorithm**

$$\text{Confidence}(X \Rightarrow Y) = \frac{\text{Number of transactions with X and Y}}{\text{Number of transactions with X}}$$

$$\text{Support}(X) = \frac{\text{Number of transactions with X}}{\text{Total number of transactions}}$$

Lift

→ **Ratio of the observed support to the expected support if X and Y were independent.**

Lift = 1… occurrence of X and Y are independent (no rule)

Lift > 1… degree of which two occurrences are dependent on another

Lift < 1… Items substitute each other

ID	🍉	🍎	🍏	🍒	🍓	🥕	🍍
1	1	1	0	0	0	0	1
2	0	0	1	0	0	1	1
3	0	0	0	1	1	0	0
4	1	1	1	0	0	1	1
5	0	1	0	0	0	0	0

$$\text{Lift}(X \Rightarrow Y) = \frac{\text{Support}(X \Rightarrow Y)}{\text{Support}(X) \times \text{Support}(Y)}$$

Lift

→ **Ratio of the observed support to the expected support if X and Y were independent.**

Lift = 1… occurrence of X and Y are independent (no rule)

Lift > 1… degree of which two occurrences are dependent on another

Lift < 1… Items substitute each other

Example: $X = \{$, $\}$
$Y = \{$$\}$

ID							
1	1	1	0	0	0	0	1
2	0	0	1	0	0	1	1
3	0	0	0	1	1	0	0
4	1	1	1	0	0	1	1
5	0	1	0	0	0	0	0

$$\text{Lift}\,(X \Rightarrow Y) = \frac{\text{Support}(X \Rightarrow Y)}{\text{Support}(X) \times \text{Support}(Y)}$$

$$\text{Support}\,(X \Rightarrow Y) = \frac{\text{Number of transactions with X and Y}}{\text{Total number of transactions}}$$

$$\text{Lift}(X \Rightarrow Y) = \frac{\frac{2}{5}}{\text{Support}(X) \times \text{Support}(Y)} =$$

Lift

→ **Ratio of the observed support to the expected support if X and Y were independent.**

Lift = 1… occurrence of X and Y are independent (no rule)

Lift > 1… degree of which two occurrences are dependent on another

Lift < 1… Items substitute each other

Example: X = {🍉, 🍎}
Y = {🍍}

ID	🍉	🍎	🍏	🍒	🍓	🥕	🍍
1	1	1	0	0	0	0	1
2	0	0	1	0	0	1	1
3	0	0	0	1	1	0	0
4	1	1	1	0	0	1	1
5	0	1	0	0	0	0	0

$$\text{Lift}(X \Rightarrow Y) = \frac{\text{Support}(X \Rightarrow Y)}{\text{Support}(X) \times \text{Support}(Y)}$$

$$\text{Support}(X) = \frac{\text{Number of transactions with } X}{\text{Total number of transactions}}$$

$$\text{Lift}(X \Rightarrow Y) = \frac{\frac{2}{5}}{\frac{2}{5} \times \text{Support}(Y)} =$$

Lift

→ **Ratio of the observed support to the expected support if X and Y were independent.**

Lift = 1… occurrence of X and Y are independent (no rule)

Lift > 1… degree of which two occurrences are dependent on another

Lift < 1… Items substitute each other

Example: X = {🍉, 🍎}
Y = {🍍}

ID	🍉	🍎	🍏	🍒	🍓	🥕	🍍
1	1	1	0	0	0	0	1
2	0	0	1	0	0	1	1
3	0	0	0	1	1	0	0
4	1	1	1	0	0	1	1
5	0	1	0	0	0	0	0

$$\text{Lift}\,(X \Rightarrow Y) = \frac{\text{Support}(X \Rightarrow Y)}{\text{Support}(X) \times \text{Support}(Y)}$$

$$\text{Support}\,(X) = \frac{\text{Number of transactions with X}}{\text{Total number of transactions}}$$

$$\text{Lift}(X \Rightarrow Y) = \frac{\frac{2}{5}}{\frac{2}{5} \times \frac{3}{5}} = \frac{0.4}{0.4 \times 0.6}$$

Lift

→ **Ratio of the observed support to the expected support if X and Y were independent.**

Lift = 1… occurrence of X and Y are independent (no rule)

Lift > 1… degree of which two occurrences are dependent on another

Lift < 1… Items substitute each other

→ "Gain" of seeing item Y when itemset X happened.

Example: $X = \{$🍉, 🍎$\}$

$Y = \{$🍍$\}$

ID	🍉	🍎	🍏	🍒	🍓	🥕	🍍
1	1	1	0	0	0	0	1
2	0	0	1	0	0	1	1
3	0	0	0	1	1	0	0
4	1	1	1	0	0	1	1
5	0	1	0	0	0	0	0

$$\text{Lift}\,(X \Rightarrow Y) = \frac{\text{Support}(X \Rightarrow Y)}{\text{Support}(X) \times \text{Support}(Y)}$$

$$\text{Support}\,(X) = \frac{\text{Number of transactions with X}}{\text{Total number of transactions}}$$

$$\text{Lift}(X \Rightarrow Y) = \frac{\frac{2}{5}}{\frac{2}{5} \times \frac{3}{5}} = \frac{0.4}{0.4 \times 0.6} = 1.667$$

Quick recap

- **Rule**: X => Y *("if X then Y")*
- **Support** → "Frequency"
- **Confidence** → "Probability that Y happens given X"
- **Lift** → "Gains we can expect from seeing Y in our transactions when X happened"

Apriori Algorithm

- What is the Apriori algorithm and why do we need it?
- How does it work conceptually?

Apriori Algorithm

Why do we need it?

- Previously we identified patterns by looking at data
- Not possible for large datasets
- Possible Algortihms:
  - Apriori
  - FP-Growth
  - …

ID	🍉	🍎	🍏	🍒	🍓	🥕	🍍
1	1	1	0	0	0	0	1
2	0	0	1	0	0	1	1
3	0	0	0	1	1	0	0
4	1	1	1	0	0	1	1
5	0	1	0	0	0	0	0

ID	🍉	🍎	🍏	🍒	🍓	🥕	🍍
1	1	1	0	0	0	0	1
2	0	0	1	0	0	1	1
3	0	0	0	1	1	0	0
…	…	…	…	…	…	…	…
986,431	1	0	0	1	1	0	0

Apriori Algorithm

- Designed to work on **transactional data** (orders, website visits, …)
  - Items in a transaction happened at the same time or have same identifier
  - Transaction = Set of items (Itemset)
- **Bottom-up** method:
  - Counts frequency of individual items first
  - Creates new item sets (pairs, triplets, etc.) by adding items one by one given a threshold (constraint) value C (e.g., support)
  - Stops when no further extensions are found given C
  - Rules are then filtered from the frequent itemsets using e.g., Confidence or Lift

Apriori Algorithm – Example

Apriori Assumptions:

- Subsets of a frequent itemset are frequent
- Subsets of an infrequent itemset are infrequent
- Itemset meets a constraint C (e.g, Minimum confidence or lift on the transaction dataset)

ID	🍉	🍎	🍏	🍒	🍓	🥕	🍍
1	1	1	0	0	0	0	1
2	0	0	1	0	0	1	1
3	0	0	0	1	1	0	0
4	1	1	1	0	0	1	1
5	0	1	0	0	0	0	0

Transactions

ID	Items
1	{🍉, 🍎, 🍍}
2	{🍏, 🥕, 🍍}
3	{🍒, 🍓}
4	{🍉, 🍎, 🍏, 🥕, 🍍}
5	{🍎}

Apriori Algorithm – Limitations

- Algorithm scans database multiple times
- Dataset must be kept in memory
- High complexity $O(2^d)$ d… number of unique items in dataset
- Returns many rules → manual filtering needed:
  - Frequent itemset {A,B,C} returns candidate rules
    - {A,B} => {C}
    - {A,C} => {B}
    - {B,C} => {A}

Market Basket Analysis

- Special application of Association Rule Mining in the context of retail
- Analyzing items that are frequently bought together to improve overall sales or pricing strategy (bundling)
  - Optimize position of items in store
  - Offer discounted bundles
  - Suggest items to customers (ecommerce)
  - …

Market Basket Analysis

- Requires data in transactional form
- Every transaction is a shopping cart (“market basket”)
- Either by same or different customers - no relevance here!
- The amount of items purchased does not matter!

T1: {‘Chocolate’, 'Water', ‘Red Wine’}

T2: {‘Chocolate’, 'Water'}

T3: {‘Chocolate’, ‘Red Wine’}

T4: {'Water', 'Fruits', 'Tea', ‘Red Wine’}

T5: {'Red Wine’', ‘Chocolate’}

T6: {‘Chocolate’}

Interactive Lab: Perform a Market Basket Analysis

Q&A
How do you feel?

Wrap-up

What did we learn today?

- Understand the fundamentals of diagnostic analytics
- Apply diagnostic analytics techniques
- Differentiate correlation vs. causation
- Understand the 5-Whys method and conduct a root-cause analysis
- Learn the essentials about rule mining and association rules
- Understand the concepts of support, confidence, and lift
- Understand (customer) segmentation techniques
- Conduct a RFM analysis
- Use clustering techniques

Outlook for next week

Week 5: Predictive Business Analytics with Python

- Learn to work with data under uncertainty
- Introduction to inferential statistics incl. regression modeling
- Do inferential statistics in Python
- Understand the fundamentals of Machine Learning and the difference to inferential statistics
- Learn different types of ML incl. decision trees
- Evaluate machine learning models
- Run predictive modeling in Python for regression and classification tasks
- Introduction to time series forecasting
- Understand the concepts of trends, seasonality, and randomness
- Run a time series forecast using ARIMA Models in Python

Thank you!

Keep in touch:

linkedin.com/in/tobias-zwingmann

tobias@rapyd.ai