Acquiring and Understanding Real Data

https://tinyurl.com/cis5190-10-26-2022

Osbert Bastani and Zachary G. Ives

CIS 4190/5190 – Fall 2022

Reminders

- Homework 4 due Wednesday November 2, 8pm
- Quiz 7 due tomorrow, Oct 27, 8pm

© 2019-22 D. Jayaraman, O. Bastani, Z. Ives

Our Machine Learning Toolkit in Practice

So far, we've assumed the data is ready for us to apply ML techniques

1. Available as an "**X** matrix": instances as rows, features as columns
2. We know the classes we want to predict
3. We are given training labels for the classes

https://www.base-4.com/open-apartments-faster-with-modular/

But this isn't always the case!

- Data may need to be wrangled and integrated
- We may need to understand our data and tasks
- We may need to understand what the data "tells us"

https://dreamsmeaning.site/house-under-construction/

For more depth, see CIS 5450 – but here we'll briefly discuss some of the major techniques and ideas

© 2019-22 D. Jayaraman, O. Bastani, Z. Ives

ML "Workflow"

Understanding the domain, framing the ML problem

Project

Today

ML Design (hypothesis class, loss function, optimizer, hyperparameters, features)

Train model

Validate / Evaluate

Main focus of this class

End of semester

© 2019-22 D. Jayaraman, O. Bastani, Z. Ives

Need to Do Work to Prepare Data for ML

Data is rarely "clean": Real data is messy, fragmented

80% + of human time is spent on data wrangling, according to practitioners

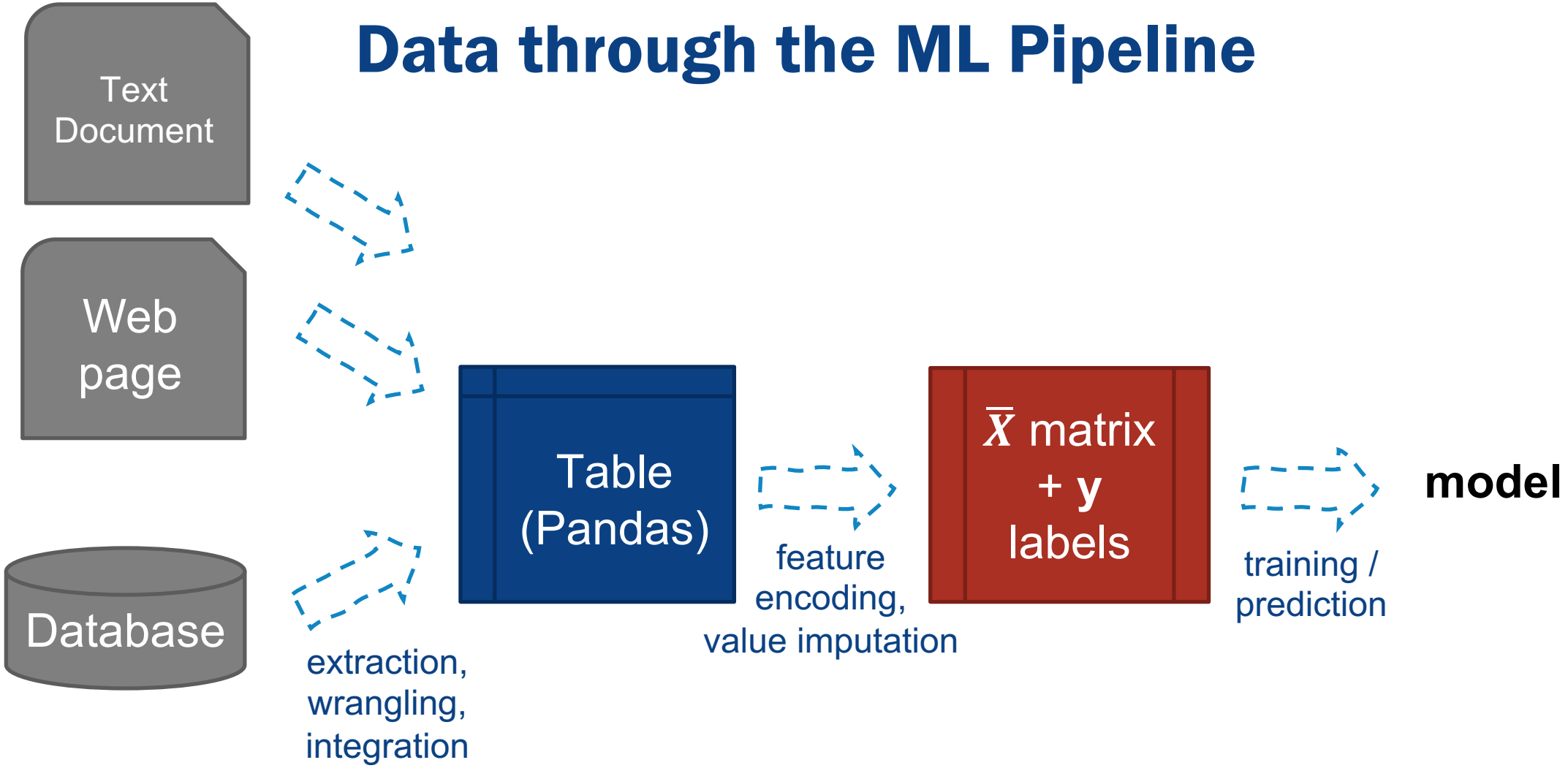
- Depends on who you ask, the application, the data source, etc...
- And "80%+" can be an underestimate!

Having good data with appropriate features is absolutely critical to success

© 2019-22 D. Jayaraman, O. Bastani, Z. Ives

From Raw to Usable Data

Data through the ML Pipeline



May need to iterate many times, and clean at each step!

© 2019-22 D. Jayaraman, O. Bastani, Z. Ives

Challenges of Acquiring Relevant Data

https://www.drivethrurpg.com/product/190343/Fiefdom-Manors-Ruling-manorial-fiefs

In many data science/ML applications: identifying and *getting access to data sources* is a challenge itself

- "Data fiefdoms"
- Lack of clarity on what is needed / need for data discovery
- Proprietary data
- Privacy constraints on access: HIPAA, FERPA, GDPR, …

Once you have permissions, may need to *wrangle* the data into Pandas tables

- From popular data sharing formats: CSV, JSON, XML
- From HTML tables
- From structured files: DICOM, HDF5, Excel, MatLab, …
- Text → information extraction and NLP
- Database connections: SQLAlchemy etc

© 2019-22 D. Jayaraman, O. Bastani, Z. Ives

Merging Data: Pandas or SQL merge / join

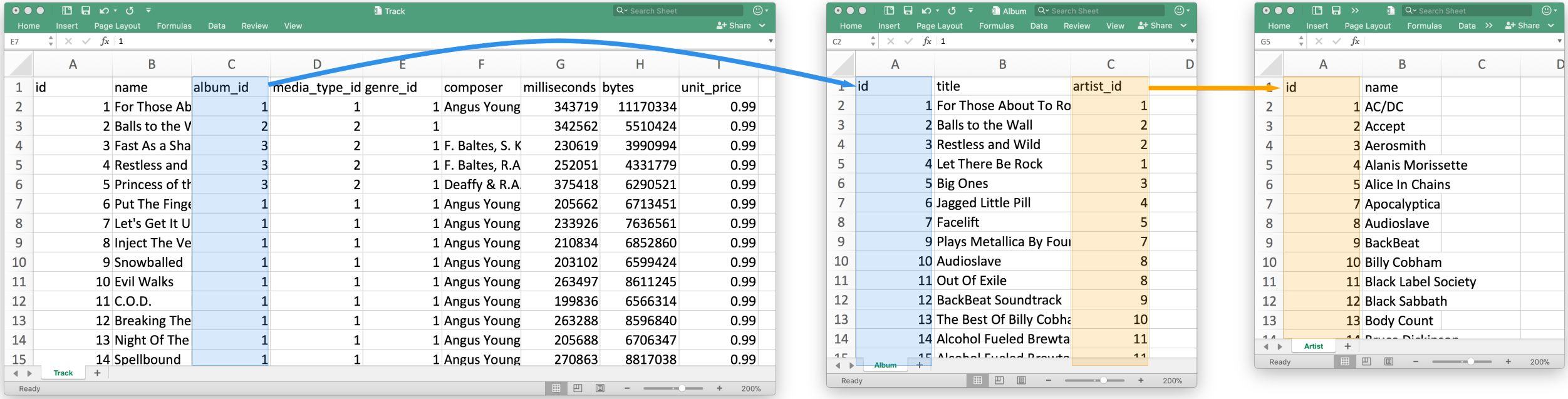
Data may be split across different files

Requires doing a join based on a key to combine data into one table

tracks

albums

artists



id	name	album_id	media_type_id	genre_id	composer	milliseconds	bytes	unit_price
1	For Those Ab	1	1	1	Angus Young	343719	11170334	0.99
2	Balls to the W	2	2	1		342562	5510424	0.99
3	Fast As a Sha	3	2	1	F. Baltes, S. K	230619	3990994	0.99
4	Restless and	3	2	1	F. Baltes, R.A	252051	4331779	0.99
5	Princess of th	3	2	1	Deaffy & R.A	375418	6290521	0.99
6	Put The Finge	1	1	1	Angus Young	205662	6713451	0.99
7	Let's Get It U	1	1	1	Angus Young	233926	7636561	0.99
8	Inject The Ve	1	1	1	Angus Young	210834	6852860	0.99
9	Snowballed	1	1	1	Angus Young	203102	6599424	0.99
10	Evil Walks	1	1	1	Angus Young	263497	8611245	0.99
11	C.O.D.	1	1	1	Angus Young	199836	6566314	0.99
12	Breaking The	1	1	1	Angus Young	263288	8596840	0.99
13	Night Of The	1	1	1	Angus Young	205688	6706347	0.99
14	Spellbound	1	1	1	Angus Young	270863	8817038	0.99

id	title	artist_id
1	For Those About To Ro	1
2	Balls to the Wall	2
3	Restless and Wild	2
4	Let There Be Rock	1
5	Big Ones	3
6	Jagged Little Pill	4
7	Facelift	5
9	Plays Metallica By Fou	7
10	Audioslave	8
11	Out Of Exile	8
12	BackBeat Soundtrack	9
13	The Best Of Billy Cobha	10
14	Alcohol Fueled Brewta	11

id	name
1	AC/DC
2	Accept
3	Aerosmith
4	Alanis Morissette
5	Alice In Chains
7	Apocalyptica
8	Audioslave
9	BackBeat
10	Billy Cobham
11	Black Label Society
12	Black Sabbath
13	Body Count

© 2019-22 D. Jayaraman, O. Bastani, Z. Ives

Image: https://dataschool.com/learn-sql/joins/

Integration May be Hard

Merged table may be too large for memory

- Incrementally load and join data, using SGD or mini-batches
- Use online learning techniques

Encoding issues

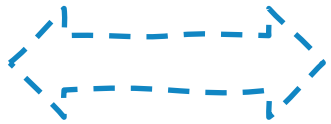
- Inconsistent data formats or terminology
- Key aspects mentioned in cell comments or auxiliary files

Record linking problem (next)

© 2019-22 D. Jayaraman, O. Bastani, Z. Ives

The Record Linking Problem

Ins ID	Name
203342	J Smith
123452	Mao Y



Student ID	Name
3432432	Jon Smithee
9734783	Jane Smyth
8273737	Ying Mao

Huge literature. Some popular ideas:

- String similarity above a threshold
  - String edit distance ("J Smith" → "Jon Smithee" with 4 edits)
  - String overlap (n-grams)
- May want to tokenize and compare tokens, not just strings
  - Or consider "soundex", common mis-substitutions, …
- Often combine similarities of multiple fields (e.g., addresses, employer)

© 2019-22 D. Jayaraman, O. Bastani, Z. Ives

Once Data Is Integrated: Need to Deal with Non-Numeric Feature Types

MSSubClass	MSZoning	LotFrontage	LotArea	Street	Alley	LotShape	...	MoSold	YrSold	SaleType	SaleCondition	SalePrice
20	RL	80.0	10400	Pave	NaN	Reg	...	5	2008	WD	Normal	174000
180	RM	35.0	3675	Pave	NaN	Reg	...	5	2006	WD	Normal	145000
60	FV	72.0	8640	Pave	NaN	Reg	...	6	2010	Con	Normal	215200
20	RL	84.0	11670	Pave	NaN	IR1	...	3	2007	WD	Normal	320000
60	RL	43.0	10667	Pave	NaN	IR2	...	4	2009	ConLw	Normal	212000
80	RL	82.0	9020	Pave	NaN	Reg	...	6	2008	WD	Normal	168500
60	RL	70.0	11218	Pave	NaN	Reg	...	5	2010	WD	Normal	189000
80	RL	85.0	13825	Pave	NaN	Reg	...	12	2008	WD	Normal	140000
60	RL	NaN	13031	Pave	NaN	IR2	...	7	2006	WD	Normal	187500

Categorical features

Ordinal features

Numeric features

Looks numeric, but is actually categorical

© 2019-22 D. Jayaraman, O. Bastani, Z. Ives

Data from: De Cock. Journal of Statistics Education 19(3), 2011

Encoding Features

Encode *categorical* features

- Use **one-hot encoding:** Expand $X_j \in \{1,2,3\}$ into $[1,0,0]$ or $[0,1,0]$ or $[0,0,1]$

Encode *ordinal* features

- Convert to a number, preserving the order (e.g. [low, medium, high] → [1, 2, 3])
- *Encoding may not capture relative differences, so may still want one-hot encoding*

HouseStyle	FullBath	RoofMatl	BsmtCond	KitchenQual
1Story	2	CompShg	TA	TA
SLvl	1	CompShg	TA	TA
2Story	2	CompShg	TA	Gd
1Story	2	CompShg	Gd	Ex
2Story	2	CompShg	TA	Gd
SLvl	1	WdShngl	TA	TA
2Story	2	CompShg	TA	Gd
SLvl	1	CompShg	TA	TA
2Story	2	CompShg	TA	TA
2Story	2	CompShg	TA	Gd

HouseStyle	FullBath	RoofMatl	BsmtCond	KitchenQual
1Story	2	CompShg	3	3
SLvl	1	CompShg	3	3
2Story	2	CompShg	3	4
1Story	2	CompShg	4	5
2Story	2	CompShg	3	4
SLvl	1	WdShngl	3	3
2Story	2	CompShg	3	4
SLvl	1	CompShg	3	3
2Story	2	CompShg	3	3
2Story	2	CompShg	3	4

© 2019-22 D. Jayaraman, O. Bastani, Z. Ives

Data Quality Issues

Missing feature values

- **Delete features** with mostly missing values
- **Delete instances** with (many) missing features, if rare

- **Impute** via mean (numeric) or mode (categorical or ordinal)
- Learn to **predict the missing values** (i.e., a kind of model stacking)

- Flag missing values using **binary variables**
  - Data might not be "missing at random"
  - It might be meaningful that instances have missing features!
    e.g., history might be missing from an unconscious trauma patient

© 2019-22 D. Jayaraman, O. Bastani, Z. Ives

Missing Values

ID	Last_Visit
1234	2018-03-05
4567	0
8910	2019-12-12

Rather than removing the case where Last_Visit is unknown – flag it with a separate feature!

We will learn a weight for the feature if it's there, and also a different weight if it isn't!

© 2019-22 D. Jayaraman, O. Bastani, Z. Ives

Data Quality Issues

Incorrect feature values

- Typos: e.g., color = {“bleu”, “green”, “gren”, “red”}
- Inconsistent spelling (e.g., “color”, “colour”)
- Inconsistent abbreviations (e.g., “Oak St.”, “Oak Street”)
- Garbage: e.g., color = “w┐r--śïį”
- Often: compare against a dictionary (~ spell-check)

Missing instance labels

- Delete instances if only a few are missing labels
- (Can also use “semi-supervised learning” techniques that can exploit unlabeled data samples alongside labeled data samples)

© 2019-22 D. Jayaraman, O. Bastani, Z. Ives

Example: Amazon Product Knowledge Graph

2021 Apple 12.9-inch iPad Pro (Wi-Fi, 256GB) - Space Gray

★★★★★ 5,339 ratings

New

$999.00

FREE delivery **Sunday, October 30**

Add to Cart

See more

5 other options
sorted by price + delivery: low to high

Filter

Used - Like New

$1,078.55

FREE delivery **October 27 - November 1**. Details

Add to Cart

Ships from: Expercom - Apple Premier Partner

Sold by: Expercom - Apple Premier Partner

★★★★☆ (1852 ratings)

88% positive over last 12 months

Used - Like New

$1,079.00

FREE delivery **October 27 - 28**. Details

Add to Cart

Condition: Open Box Item # 1586413 What's in the box: Apple iPad Pro 12.9" M1 Chip (Space Gray), 20W USB Type-C Po...More

Ships from: Adorama

Sold by: Adorama

★★★★☆ (575279 ratings)

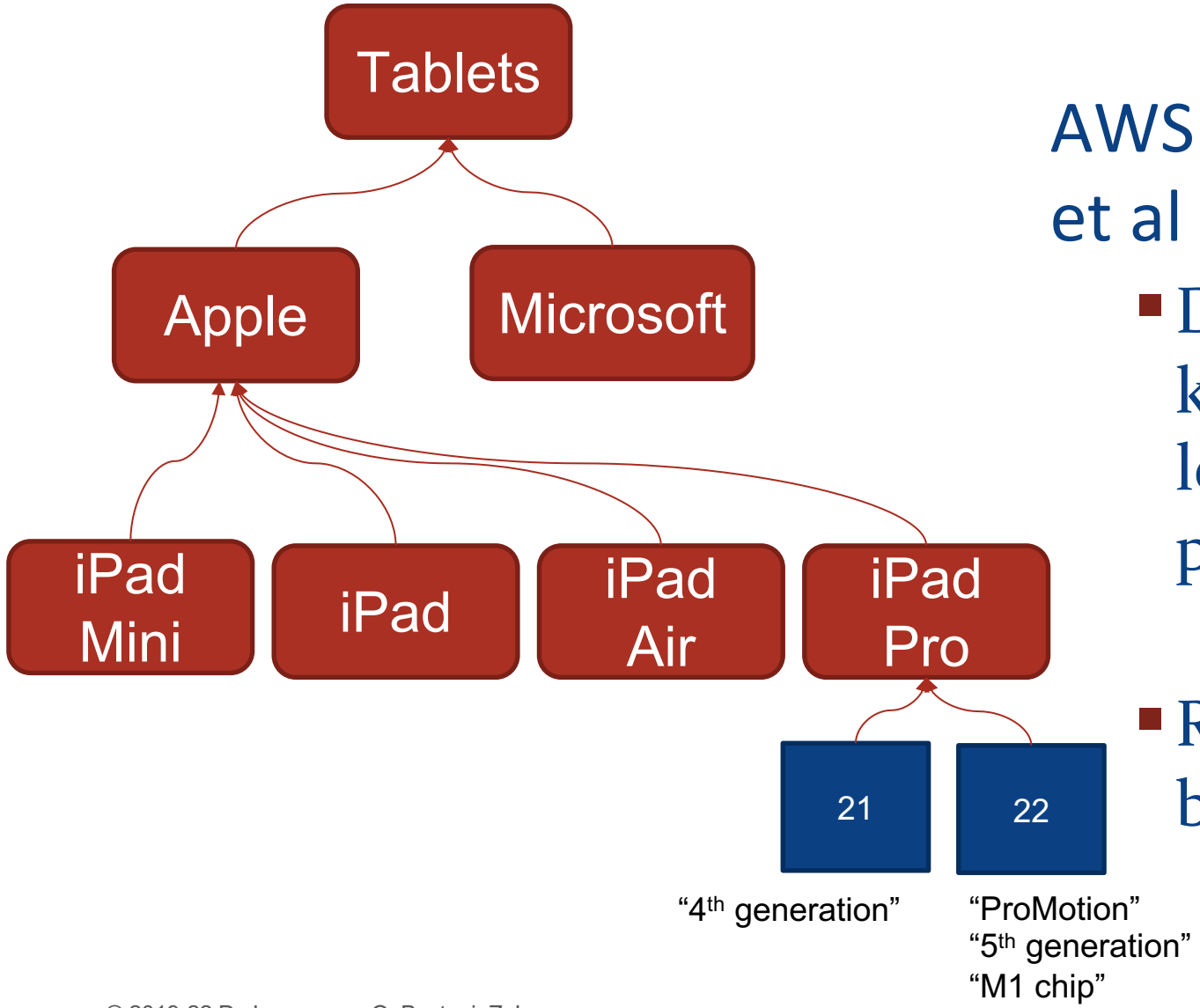
90% positive over last 12 months

Amazon wants to rank + recommend products by relevance – to search or on the front page

Q: How does it know whether a product in a vendor catalog is a 2021 iPad Pro vs a 2019 version? The right size?

A: Aligning with a *product knowledge graph*!

Amazon Product Knowledge Graph



AWS knowledge graph tools [Dong et al KDD 18]:

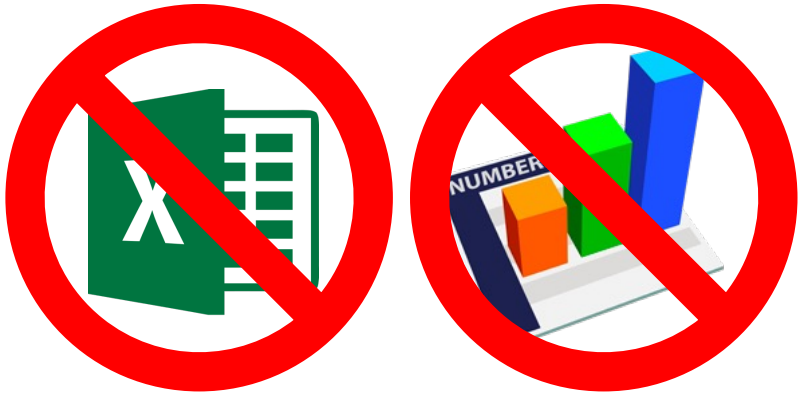
- Distant supervision to take knowledge graph, use NNs to learn patterns from open Web page product descriptions
  apply to model catalog entries
- Random forest for record linking based on terms, values

© 2019-22 D. Jayaraman, O. Bastani, Z. Ives

Script Your Data Preprocessing!

Don't manually edit via a spreadsheet program

- No history of changes
- Very easy to introduce mistakes
- Hard to correct earlier decisions



Instead, write a script that loads the raw data and does all preprocessing

- Documents all steps
- Incremental debugging
- Easy to make changes to earlier steps
- Repeatable

© 2019-22 D. Jayaraman, O. Bastani, Z. Ives

Summary: Data Integration

Key tasks as a data science consultant or a data scientist:

- Understand the data, important aspects of the domain
- Identify relevant data sources
- Import and wrangle the data
- Integrate and clean

Integration across sources may happen before or after cleaning – *why?*

- Record linking
- Value imputation

Data processing should often be done in a pipeline or script

© 2019-22 D. Jayaraman, O. Bastani, Z. Ives

Understanding Your Data

What's Interesting? Important? Representative?

Machine learning is algorithmic – but its use should be guided by an understanding of the data, underlying hypothesis space, etc.

> e.g., two features may be heavily correlated, or there may be odd outliers, or unexpected scaling, or …

Thus: we should always get a sense of the data before we start trying to build models!

© 2019-22 D. Jayaraman, O. Bastani, Z. Ives

Simplest Starting Point: Dataframe.describe()

Note the missing values

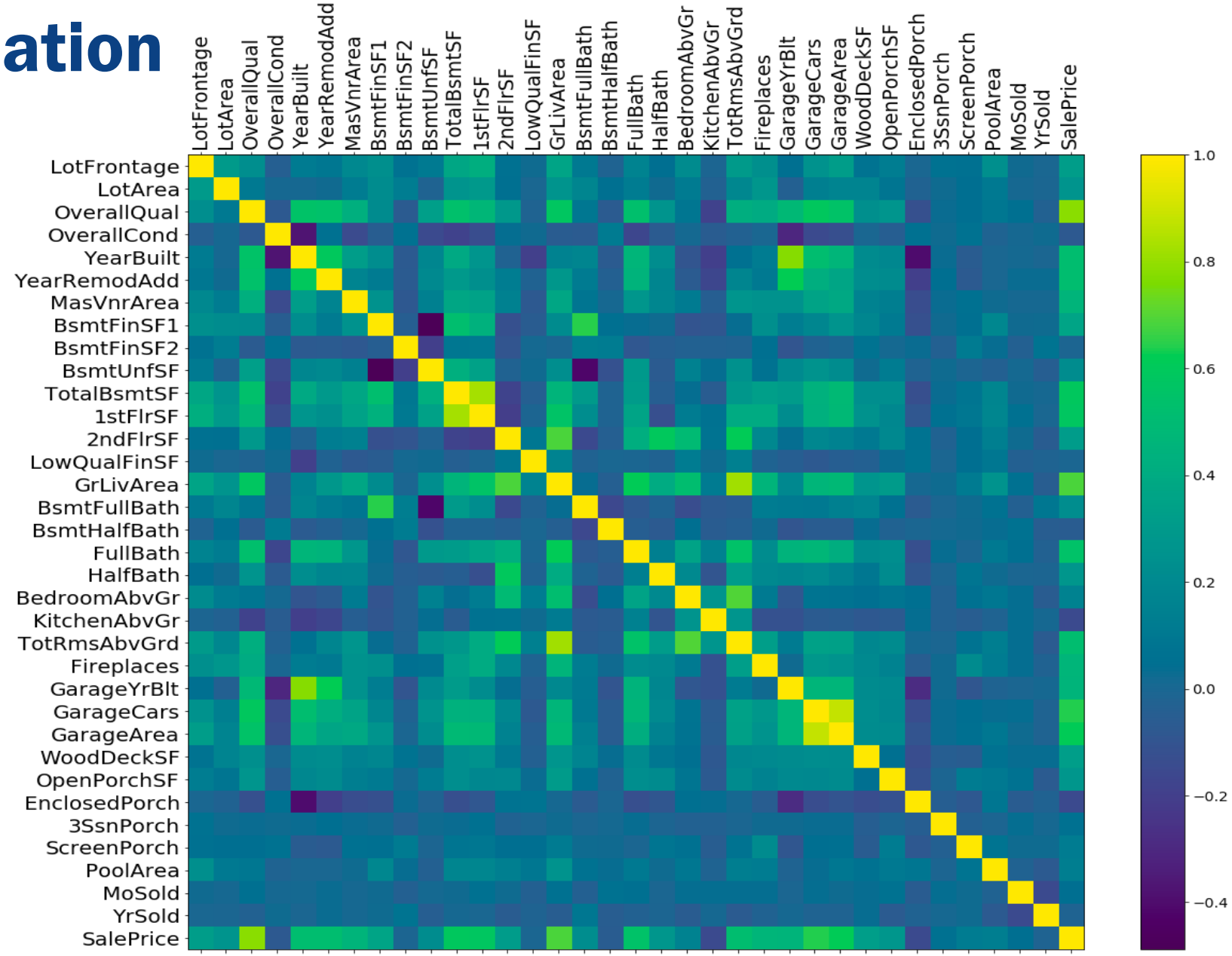
No missing target values

	Id	MSSubClass	LotFrontage	LotArea	OverallQual	OverallCond	YearBuilt	YearRemodAdd	MasVnrArea	...	SalePrice
count	1022.000000	1022.000000	832.000000	1022.000000	1022.000000	1022.000000	1022.000000	1022.000000	1019.000000		1022.000000
mean	732.338552	57.059687	70.375000	10745.437378	6.128180	5.564579	1970.995108	1984.757339	105.261040		181312.692759
std	425.860402	42.669715	25.533607	11329.753423	1.371391	1.110557	30.748816	20.747109	172.707705		77617.461005
min	1.000000	20.000000	21.000000	1300.000000	1.000000	1.000000	1872.000000	1950.000000	0.000000		34900.000000
25%	367.500000	20.000000	59.000000	7564.250000	5.000000	5.000000	1953.000000	1966.000000	0.000000		130000.000000
50%	735.500000	50.000000	70.000000	9600.000000	6.000000	5.000000	1972.000000	1994.000000	0.000000		165000.000000
75%	1100.500000	70.000000	80.000000	11692.500000	7.000000	6.000000	2001.000000	2004.000000	170.000000		215000.000000
max	1460.000000	190.000000	313.000000	215245.000000	10.000000	9.000000	2010.000000	2010.000000	1378.000000		745000.000000

Potential outliers

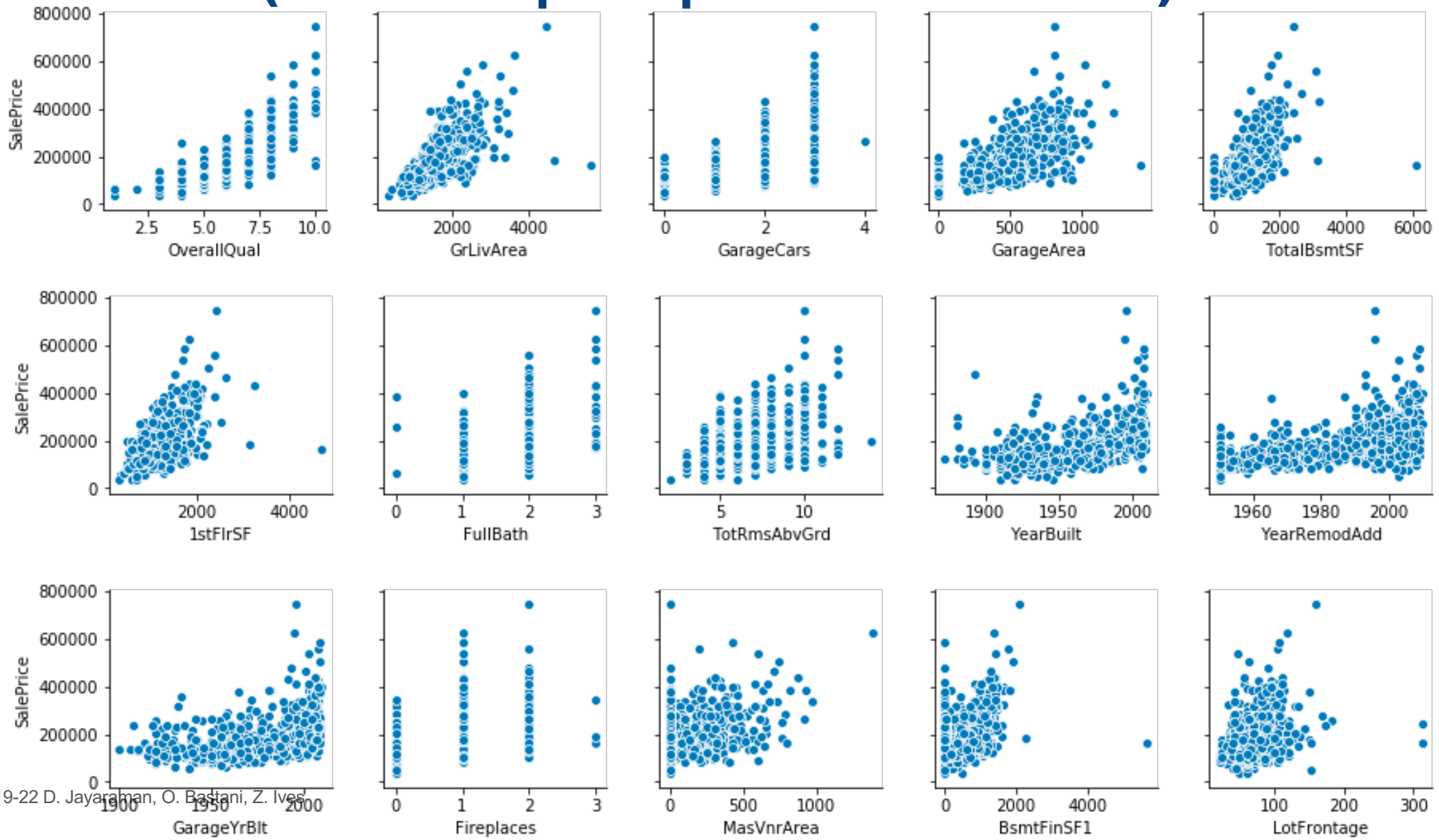
© 2019-22 D. Jayaraman, O. Bastani, Z. Ives

Feature Correlation Matrix



© 2019-22 D. Jayaraman, O. Bastani, Z. Ives

Plot Features Most Correlated with Target Variable (Seaborn `pairplot` and similar)



© 2019-22 D. Jayaraman, O. Bastani, Z. Ives

Handling Outliers: Causes of Outliers

Errors

- Human error in data collection or data entry
- Measurement/instrumentation errors
- Experimental errors
- Data merge errors
  - e.g., merging datasets with different scales
- Data preprocessing errors

Natural

- Novelties in the data – not mistakes!

© 2019-22 D. Jayaraman, O. Bastani, Z. Ives

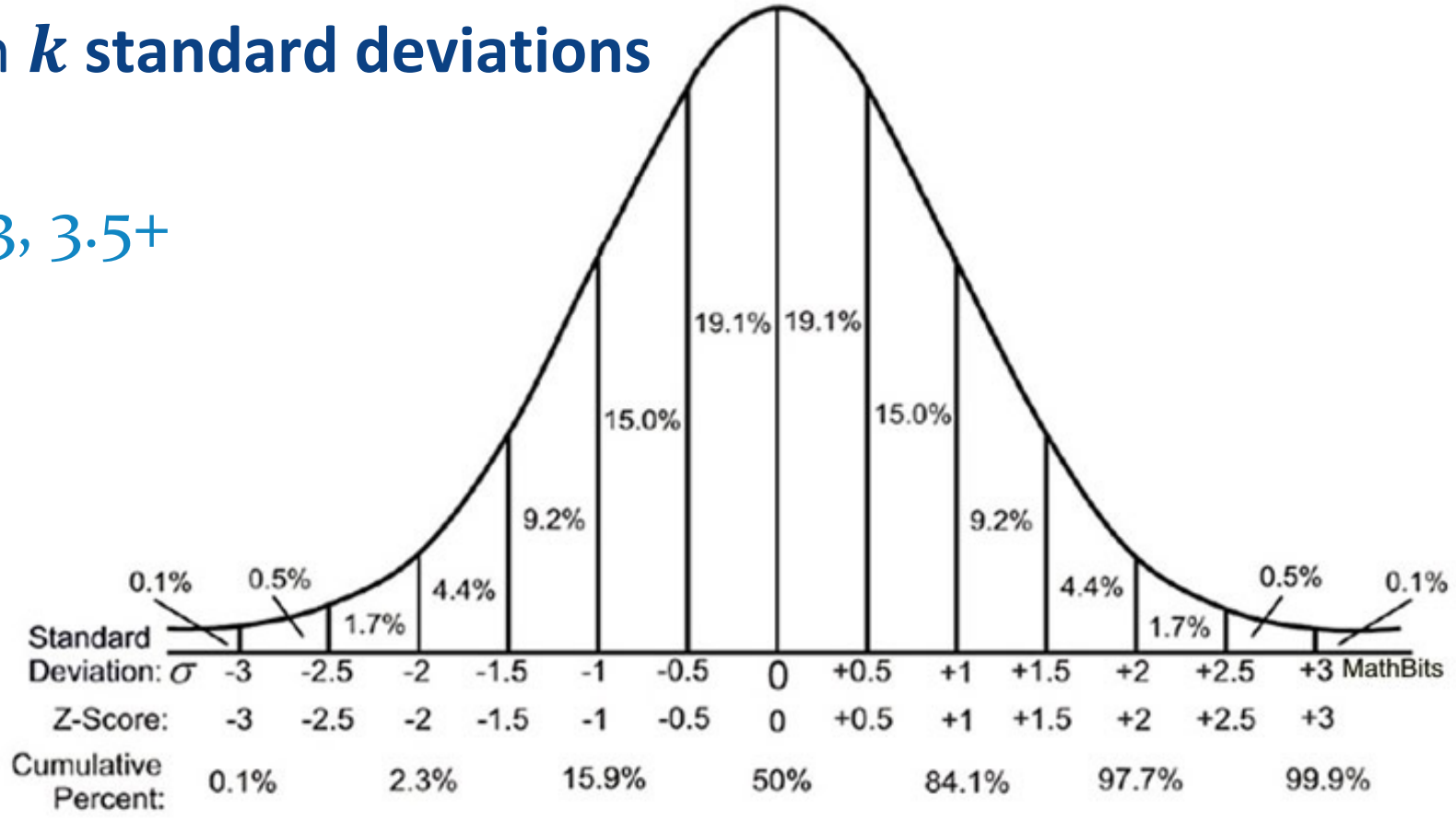
Outlier Detection: Z-score

- Assume feature values are **Gaussian-distributed**
- Discard points more than ***k* standard deviations** away from the mean

  Good values for k: 2.5, 3, 3.5+

Cautions:

- Mostly for low-*d* feature spaces on reasonably small-to-medium data sets
- Incorrect if parametric assumption doesn't hold



https://mathbitsnotebook.com/Algebra2/Statistics/STzScores.html

© 2019-22 D. Jayaraman, O. Bastani, Z. Ives

Feature Standardization

Rescales features to have zero mean and unit variance

- Let μ_j, σ_j be: $\mu_j = \frac{1}{N}\sum_{i=1}^{N} x_{ij}$, $\sigma_j^2 = \sum_{i=1}^{N}\left(x_{ij} - \mu_j\right)^2 / N$

- Replace each value with $x_{ij} \leftarrow \frac{x_{ij}-\mu_j}{\sigma_j}$ for $j = 1 \ldots D$ (not x_0)

Same as the "Z score"

Could also rescale features to lie in [0, 1] (for special cases where there's a clear scale, e.g., pixel values)

- Replace each value with $x_{ij} \leftarrow \frac{x_{ij}-\min(x_j)}{\max(x_j)-\min(x_j)}$

Must use the same transformation for both training and prediction
(μ_j and σ_j are computed on training data and also used on the test data)

Summary

- Important to look at data, identify missing values, correlations, and outliers

- Remove outliers
- Impute or drop missing values
- Use regularization or drop correlated features

© 2019-22 D. Jayaraman, O. Bastani, Z. Ives

Beyond describe():
Understanding *Structure* in Data Domains

- We can visualize data, and call describe(), to get a sense of univariate and bivariate distributions

- But: suppose the data naturally falls into different groupings – perhaps even suggesting which *classes* we would like to learn?

➢ This motivates a study of additional techniques for extracting structure present in the data

© 2019-22 D. Jayaraman, O. Bastani, Z. Ives

Unsupervised Machine Learning

Types of Learning

Supervised learning

- Given: training data + desired outputs (labels)

Unsupervised learning

- Given: training data (without labels)

Semi-supervised learning

- Given: training data + some labels

Reinforcement learning

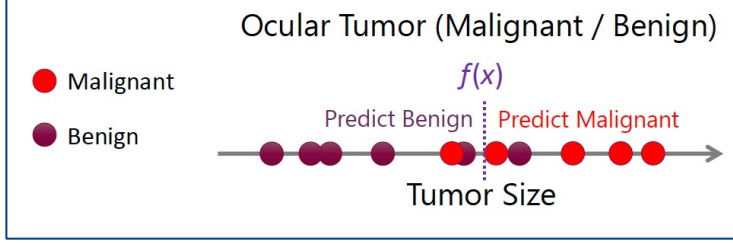
- Given: observations and occasional rewards as the agent performs sequential actions in an environment

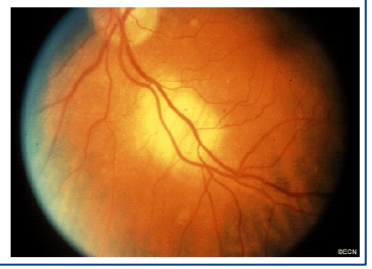
© 2019-22 D. Jayaraman, O. Bastani, Z. Ives

From Supervised to Unsupervised Learning

Supervised Learning: Classification

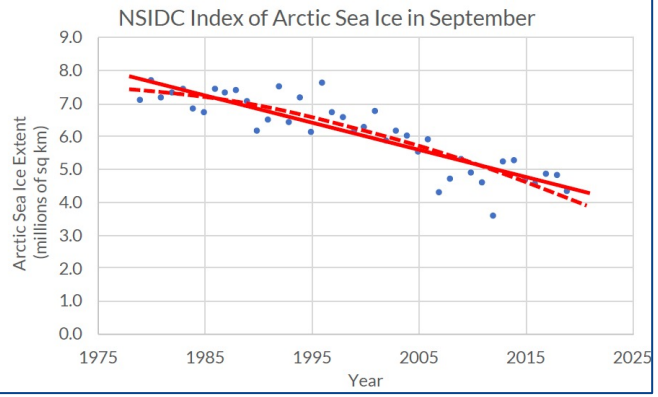
- Given (x_1, y_1), (x_2, y_2), …, (x_n, y_n)
- Learn a function $f(x)$ to predict y given x
  - y is categorical == classification





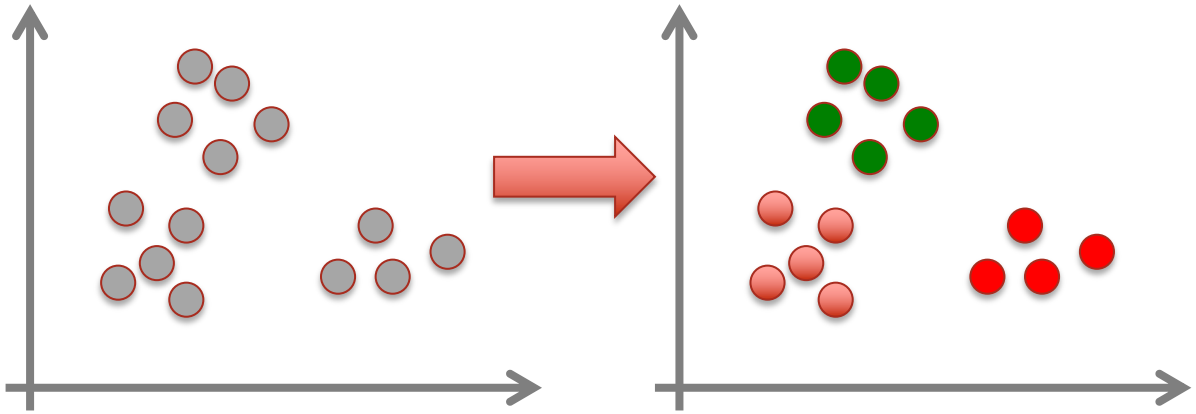
Supervised Learning: Regression

- Given (x_1, y_1), (x_2, y_2), …, (x_n, y_n)
- Learn a function $f(x)$ to predict y given x
  - y is numeric == regression



Given $x_1, x_2, \ldots, x_N$ (without labels)

Learn "hidden structure" in the data



Types and Uses of Unsupervised Learning

Dimensionality Reduction

Map samples $\boldsymbol{x}_i \in \mathbb{R}^D$ to $f(\boldsymbol{x}_i) \in \mathbb{R}^{D' \ll D}$

Special case: clustering, $f(\boldsymbol{x}_i) \in \mathbb{N}$

- **Feature Learning:** For preprocessing inputs to an ML algorithm, since lower-dimensional features permit smaller models and fewer data samples.
- **Compression (for storage):** e.g. JPEG standard for images is now adopting unsupervised ML approaches https://jpeg.org/items/20190327_press.html
- **Visualization:** Exploring a dataset, or an ML model's outputs

© 2019-22 D. Jayaraman, O. Bastani, Z. Ives

Unsupervised ML in the ML Workflow

Framing an ML problem

Data curation (sourcing, scraping, collection, labeling)

Data analysis / visualization

ML Design (hypothesis class, loss function, optimizer, hyperparameters, features)

Train model

Validate / Evaluate

Deploy (and generate new data)

Monitor performance on new data

© 2019-22 D. Jayaraman, O. Bastani, Z. Ives

"Here's some data, could you do something with it?"

Unsupervised learning helps identify uses for data:

- Visualize the data, find clusters
  e.g. "based on our polling data, there are three main voting blocs, based on age, race, education level, income, political beliefs, and home-ownership. Features like marital status and # children are irrelevant."
- Identify interesting supervised learning problems within your dataset
  e.g. "do our company's profits y_i actually vary systematically based on the weather $\boldsymbol{x}_i$?"
- Generate new data
  e.g. "given all of Bach's work, I could generate new music that would sound like Bach."

© 2019-22 D. Jayaraman, O. Bastani, Z. Ives

Popular Clustering Algorithms

1. One of the simplest, scalable techniques: k-Means clustering

2. Greedy: hierarchical clustering

© 2019-22 D. Jayaraman, O. Bastani, Z. Ives

K-Means Clustering

The Clustering Setting

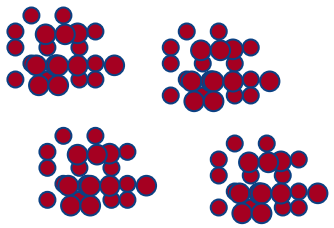
Task:

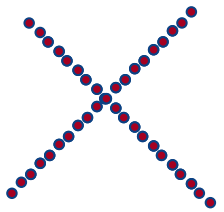
Input: $\mathcal{D} = \{\boldsymbol{x}_i\}_{i=1}^{N}$

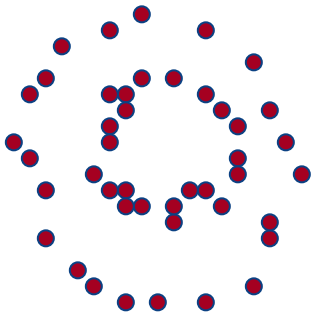
Want to discover a mapping $f(\boldsymbol{x}_i) \in \{1, 2, 3, \ldots, K\}$ that discovers natural groupings in the data.

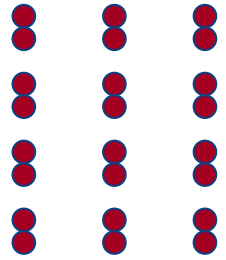
Performance Metric / Objective Function: What is a good mapping $f(.)$?

Somewhat loosely defined, and different clustering algorithms differ in their definition of a good clustering $f(.)$





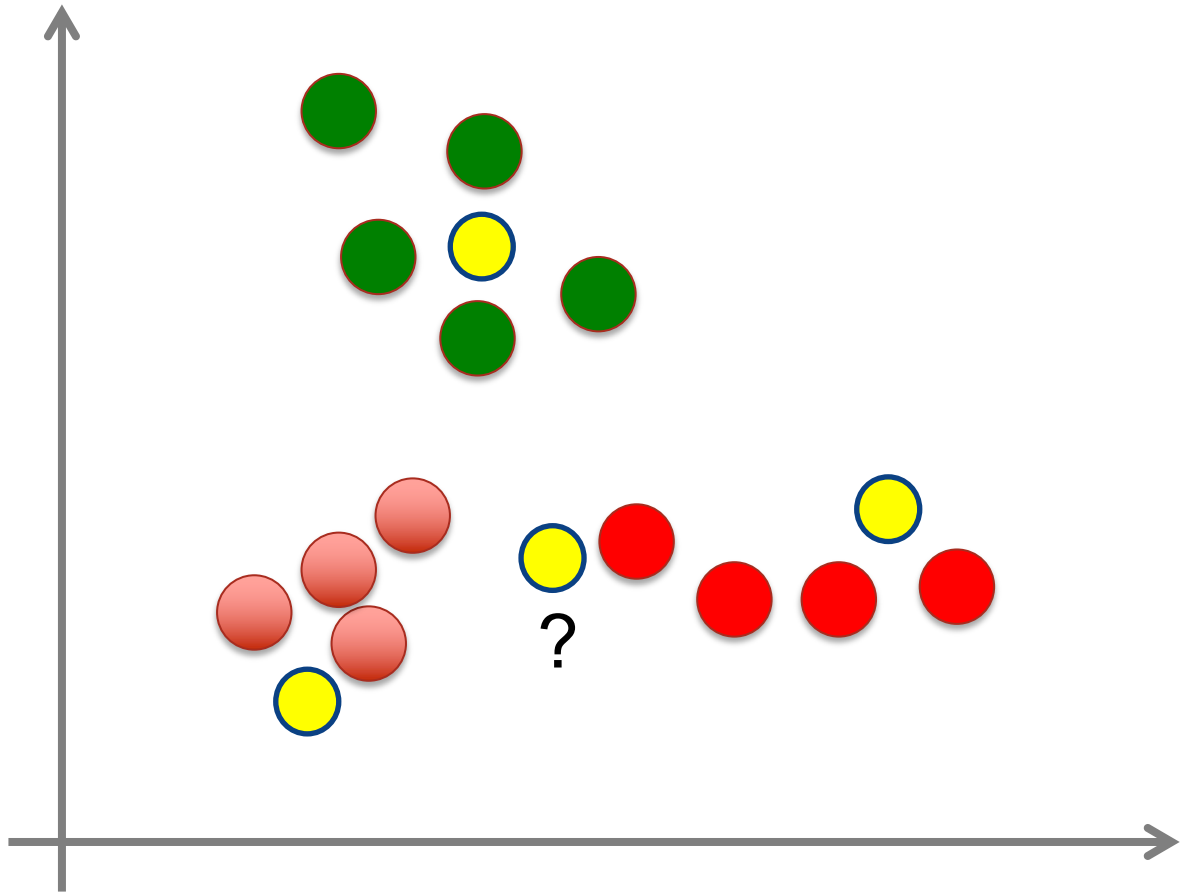




© 2019-22 D. Jayaraman, O. Bastani, Z. Ives

Figures: Dan Roth

Guess The Cluster Assignment



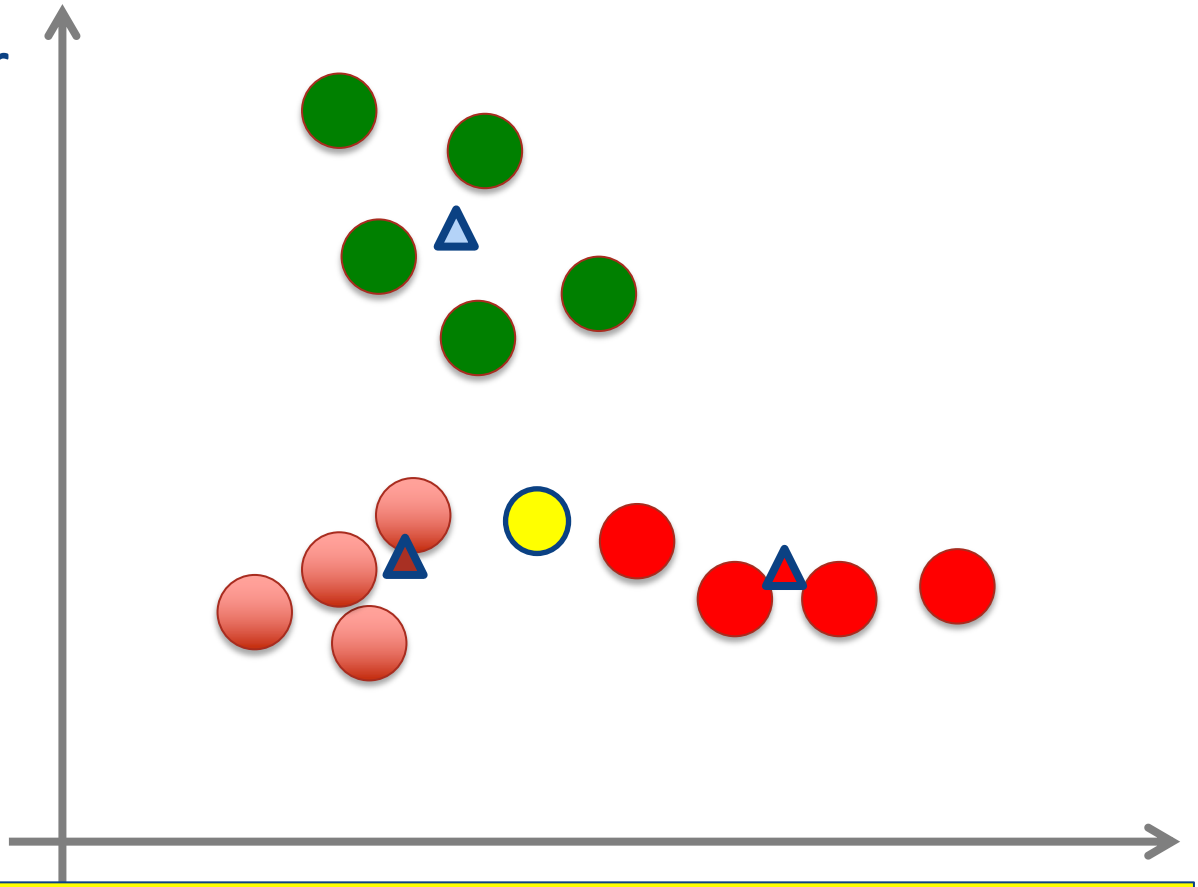
© 2019-22 D. Jayaraman, O. Bastani, Z. Ives

Cluster Assignment in K-Means

- Define a centroid μ_k for each cluster k
- For any new sample, assign the cluster whose centroid/mean is closest!

Equivalent to 1-nearest neighbor classification over the cluster means.

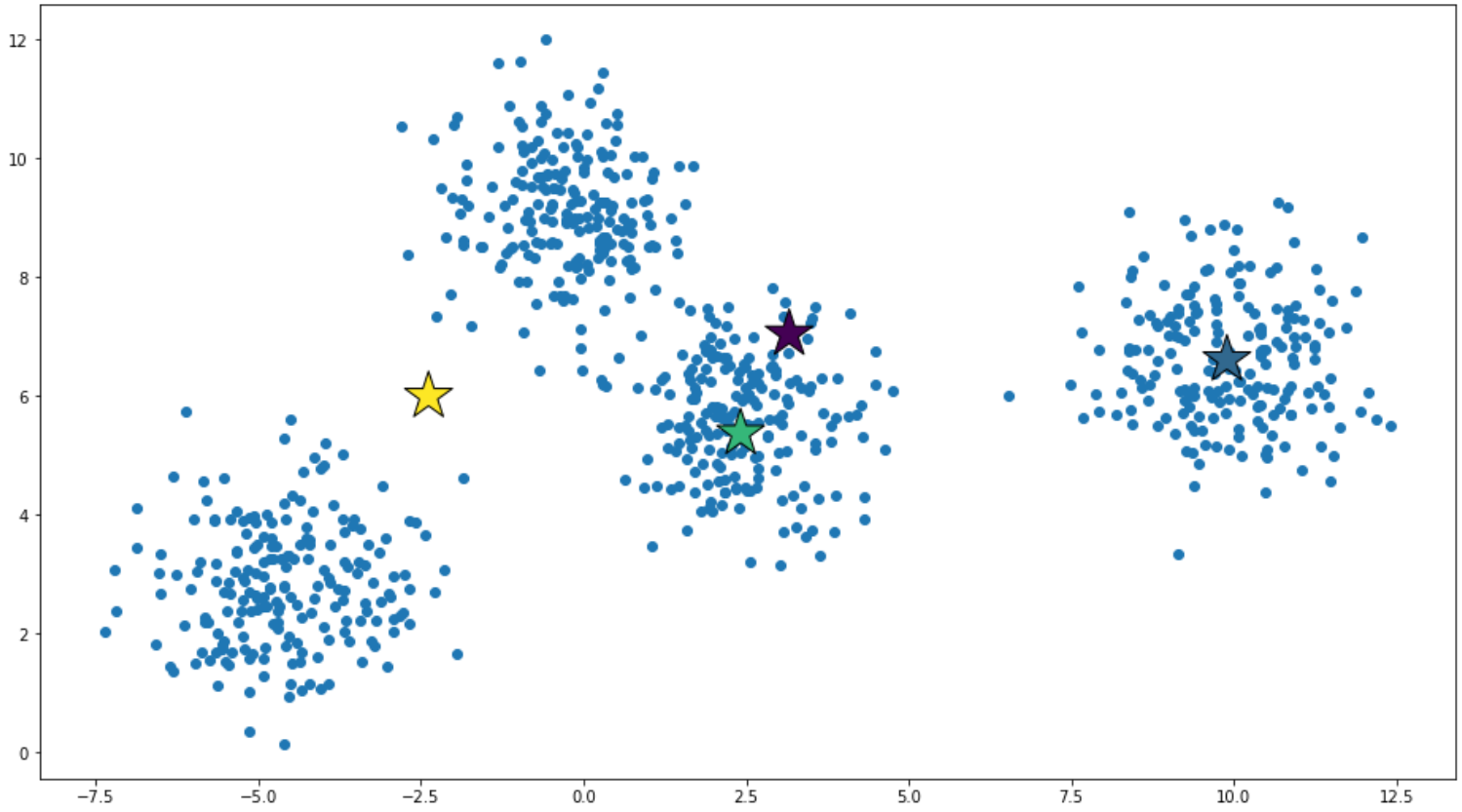
Note: Certainly not the only answer we could have come up with!



Can we expand this into a full, consistent clustering algorithm?

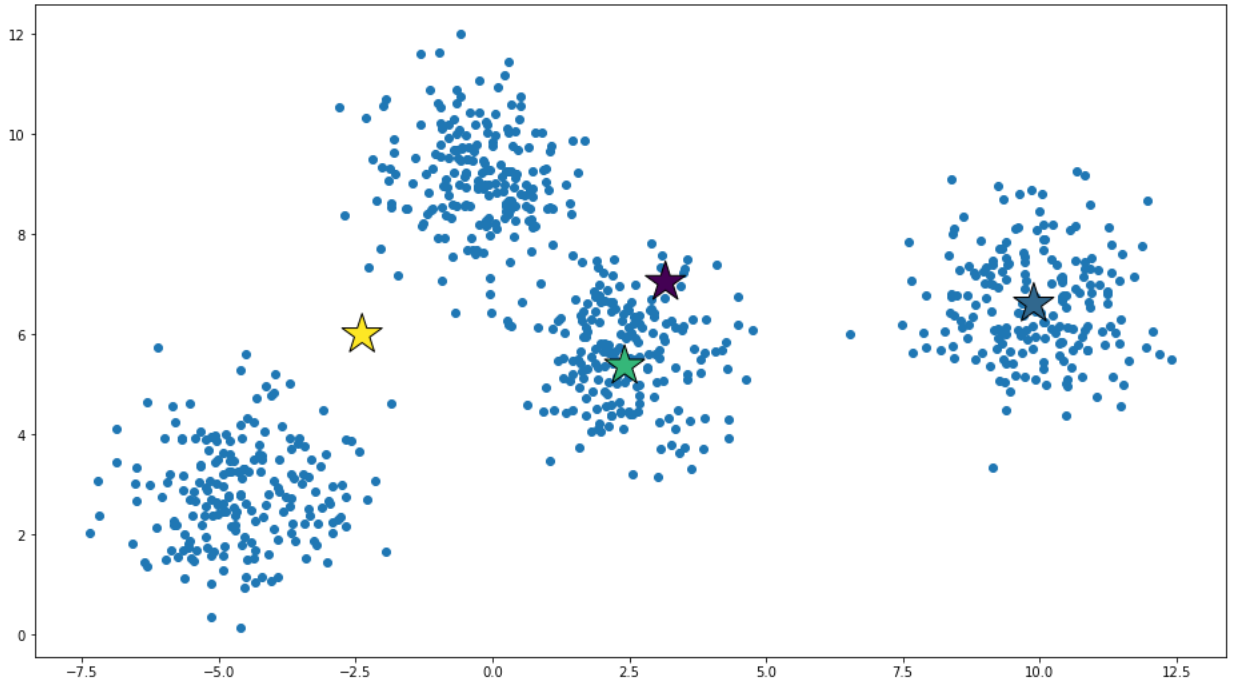
© 2019-22 D. Jayaraman, O. Bastani, Z. Ives

Clustering Data



© 2019-22 D. Jayaraman, O. Bastani, Z. Ives

Clustering Data

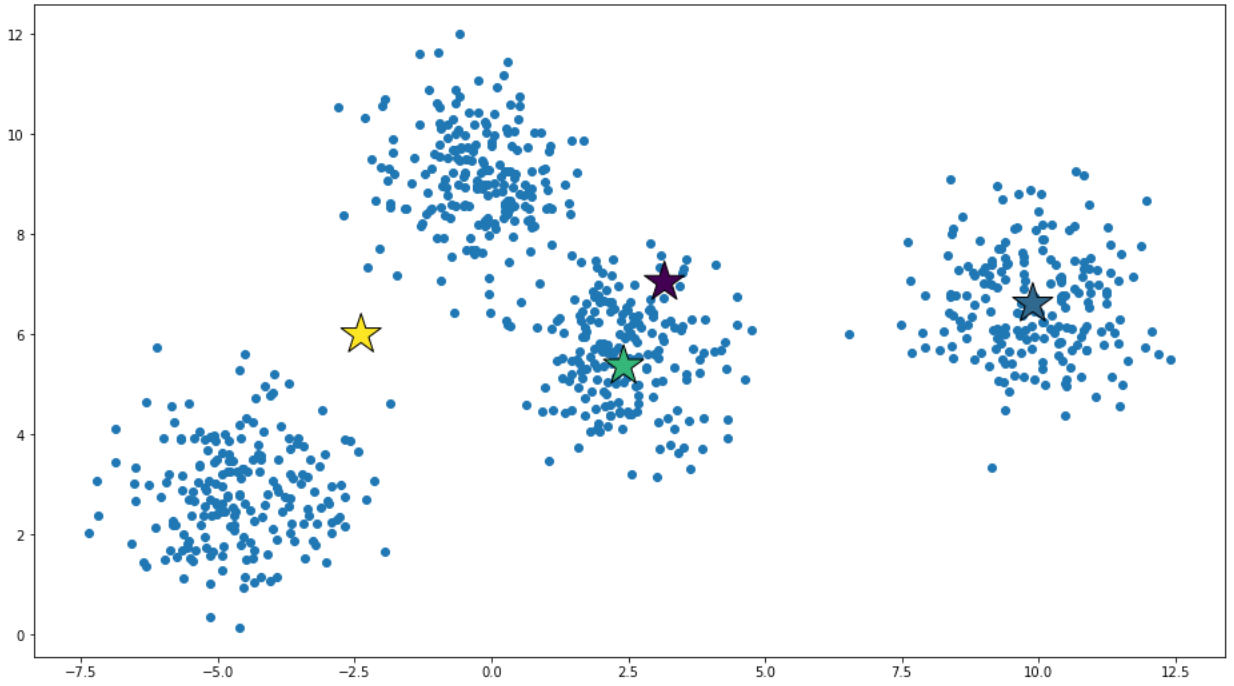


© 2019-22 D. Jayaraman, O. Bastani, Z. Ives

K-Means Clustering

K-Means (K, X)

- Randomly choose K cluster means
- Loop until convergence, do:
  - Assign each point to the cluster of the closest centroid
  - Re-estimate the cluster centroids based on the data assigned to each cluster

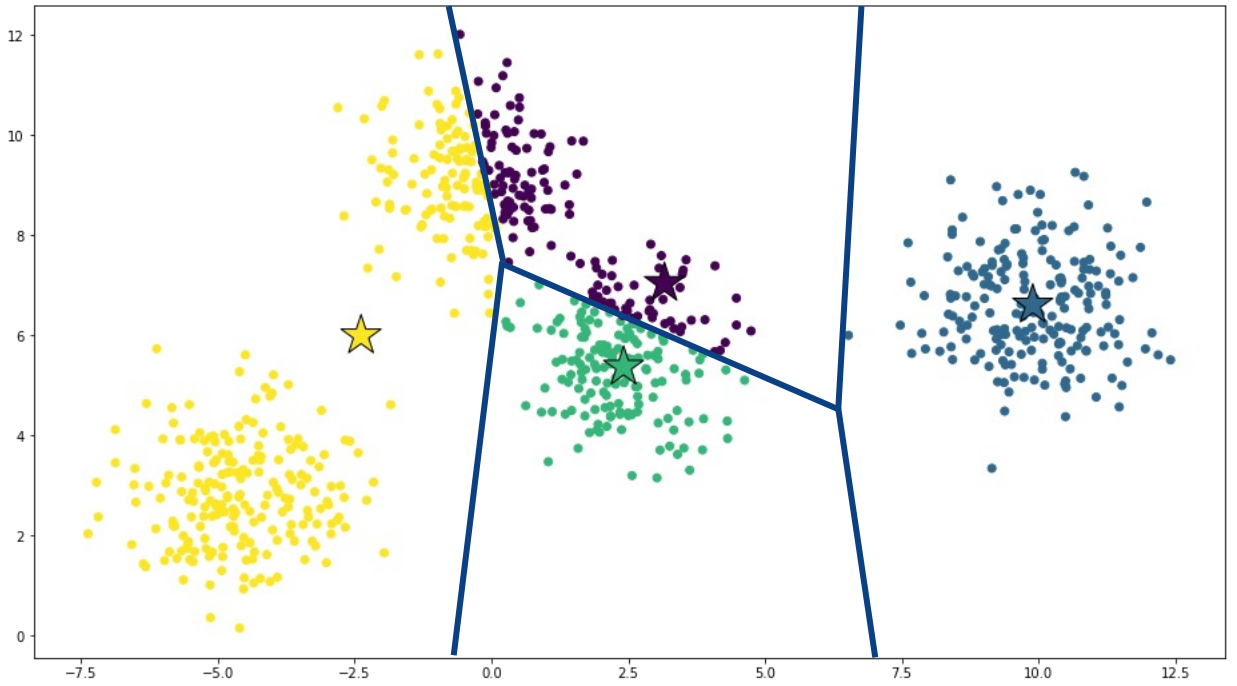


© 2019-22 D. Jayaraman, O. Bastani, Z. Ives

K-Means Clustering

K-Means (K, X)

- Randomly choose K cluster center locations (centroids)
- Loop until convergence, do:
  - Assign each point to the cluster of the closest centroid
  - Re-estimate the cluster centroids based on the data assigned to each cluster



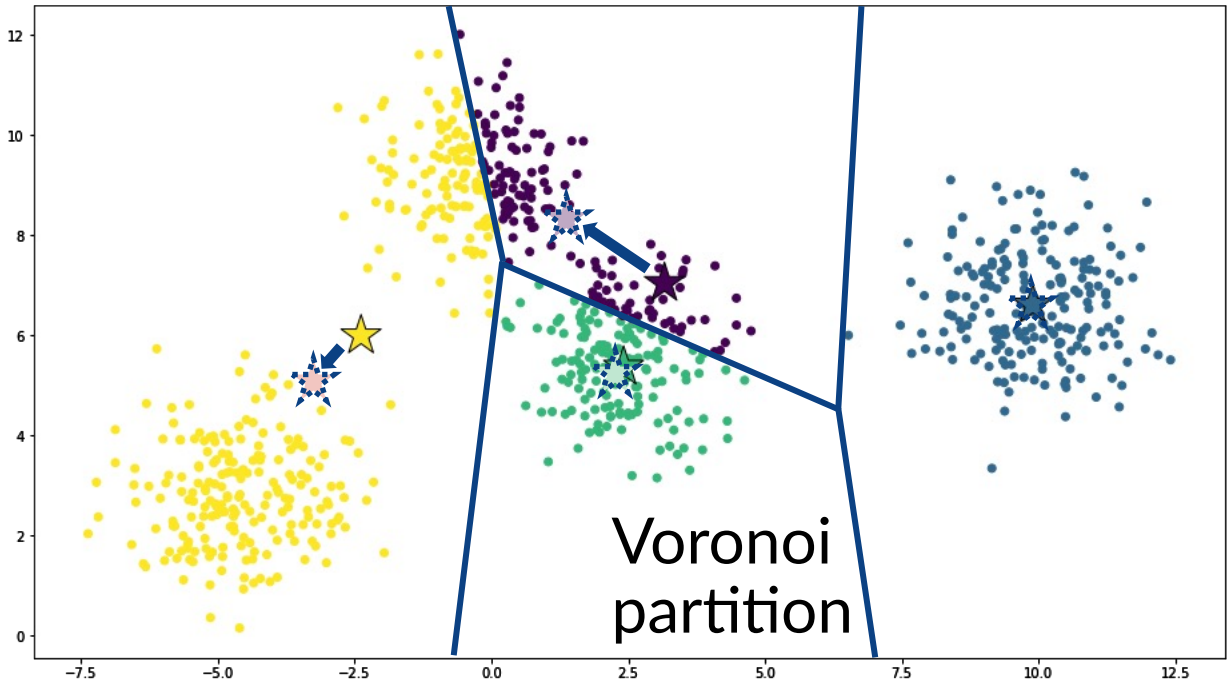
Voronoi partition

© 2019-22 D. Jayaraman, O. Bastani, Z. Ives

K-Means Clustering

K-Means (K, X)

- Randomly choose K cluster center locations (centroids)
- Loop until convergence, do:
  - Assign each point to the cluster of the closest centroid
  - Re-estimate the cluster centroids based on the data assigned to each cluster



© 2019-22 D. Jayaraman, O. Bastani, Z. Ives

K-Means Clustering

K-Means (K, X)

- Randomly choose K cluster center locations (centroids)
- Loop until convergence, do:
  - Assign each point to the cluster of the closest centroid
  - Re-estimate the cluster centroids based on the data assigned to each cluster

Optimizer: "Alternating Minimization"

K-means finds a local optimum of the following objective function:

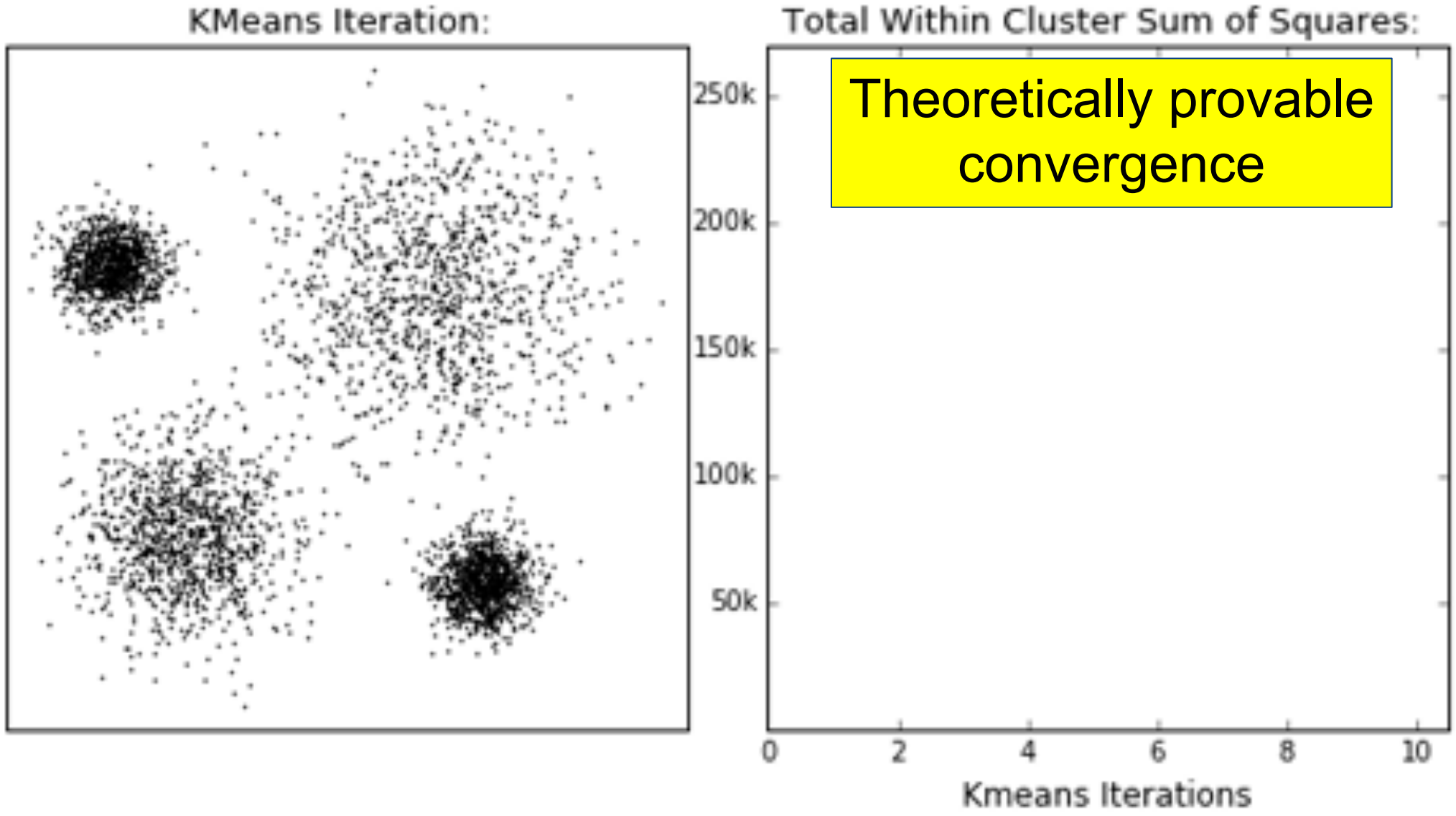
"Sum of squared distances" loss function

$$\arg\min_{\boldsymbol{S}} \sum_{k=1}^{K} \sum_{\boldsymbol{x} \in S_k} \|\boldsymbol{x} - \boldsymbol{\mu}_K\|_2^2$$

where $\boldsymbol{S} = \{S_1, \ldots, S_K\}$ are sets corresponding to disjoint clusters, and the clusters together include all samples.

© 2019-22 D. Jayaraman, O. Bastani, Z. Ives

K-Means Clustering Convergence



© 2019

https://dashee87.github.io/data%20science/general/Clustering-with-Scikit-with-GIFs/

Generalizing to Other Distance Functions

K-Means Objective Function:

$$\arg\min_{S} \sum_{k=1}^{K} \sum_{x \in S_k} \|x - \mu_K\|_2^2$$

But it is possible to define k-means with other notions of pairwise distance between samples too. For example:

$$\left(\sum_{d} |x_{1d} - x_{2d}|^1 \right)^{\frac{1}{1}}$$

ℓ_1 distance

$$\sum_{d} |x_{1d} - x_{2d}|$$

"K-medoids" clustering

Q: What would have to change in the algorithm?

K-Means (k, X)
- Randomly choose k cluster center locations (centroids)
- Loop until convergence, do:
  - Assign each point to the cluster of the closest centroid
  - Re-estimate the cluster centroids based on the data assigned to each cluster

© 2019-22 D. Jayaraman, O. Bastani, Z. Ives

K-Means is Too Sensitive to Initialization

Alternative strategies:

1. Do many runs of K-Means, each with different initial centroids, and pick the best
2. Pick initial centroids using a better method than random choice

K-means+ + initialization

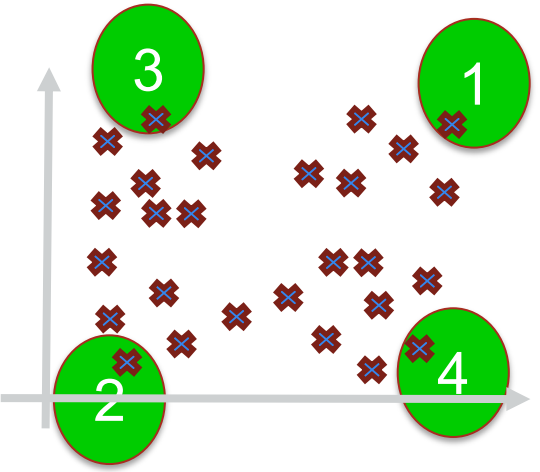
- Choose a data point uniformly at random as the first centroid
- Loop for $2{:}K$, do:
  - Let $D(\boldsymbol{x})$ be the distance from each point $\boldsymbol{x}$ to the closest centroid
  - Choose data point $\boldsymbol{x}$ randomly $\propto D(\boldsymbol{x})^2$ as the next centroid. Higher chance to pick points that are far from previous centroids.

© 2019-22 D. Jayaraman, O. Bastani, Z. Ives

K-means++ Illustrated

Place the initial centroids **far away from one another**:

1. Initialize an empty set M (for storing selected centroids); **Randomly select** the first centroid from the input sample and assign it to M
2. For each x_i that is not in M, find the distance $D(x_i)$ to the closest centroid in M
3. Choose one new data point at random as a new centroid **using probability distribution ~ $D(x)^2$**
4. Repeat (2) and (3) until K centroids have been chosen



Then do "classic" k-means

© 2019-22 D. Jayaraman, O. Bastani, Z. Ives

How Many Clusters?

Measuring the Performance of k-Means

How can we evaluate how good our clustering is?

Some options:

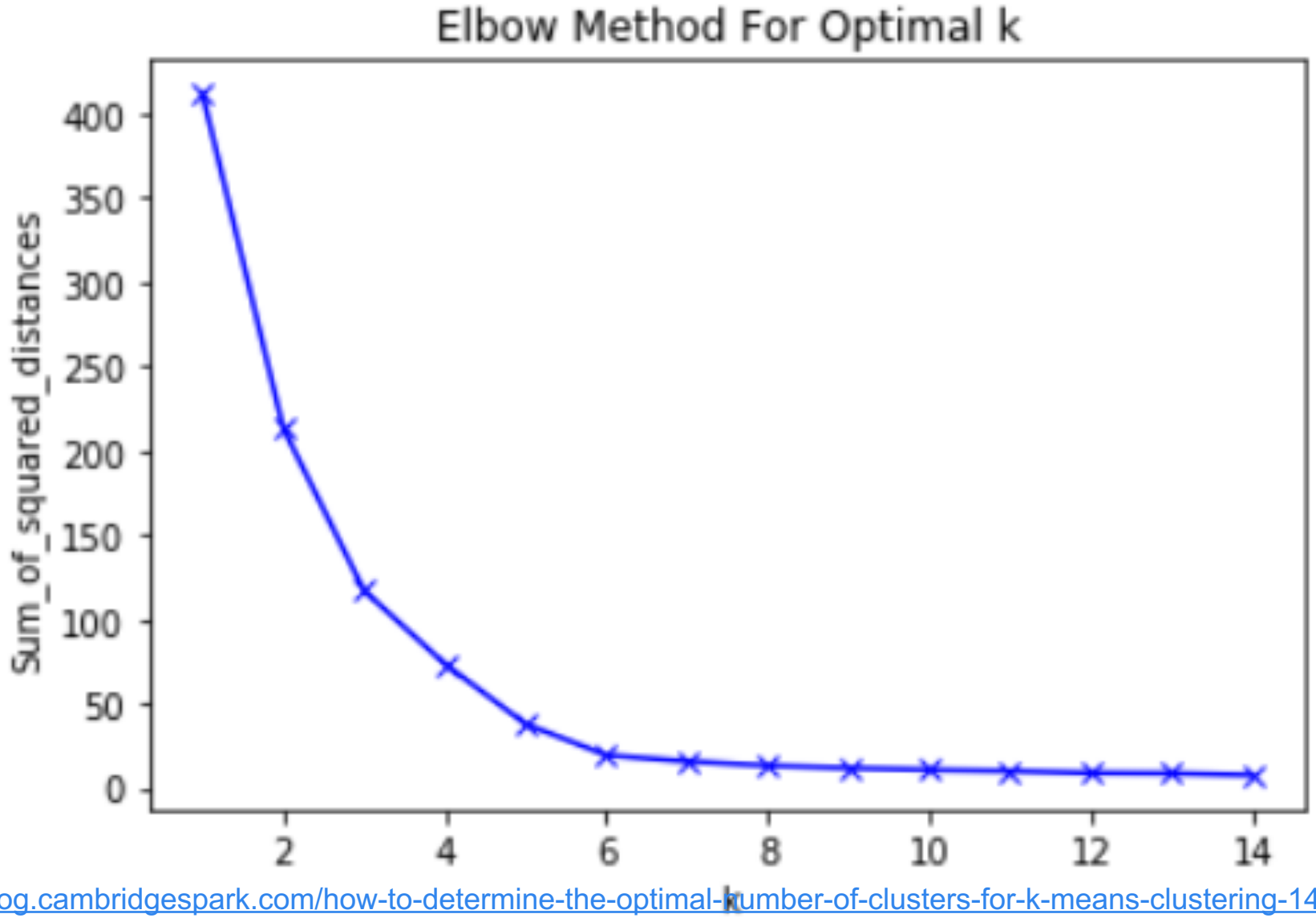
- Evaluation using the k-means objective itself
- Comparing to class labels (for a subset of data)

  Sometimes possible
- Subjective evaluation by a human domain expert

…

© 2019-22 D. Jayaraman, O. Bastani, Z. Ives

“Knee Point” For Selecting K



https://blog.cambridgespark.com/how-to-determine-the-optimal-number-of-clusters-for-k-means-clustering-14f27070048f

© 2019-22 D. Jayaraman, O. Bastani, Z. Ives

k-Means Clustering

- Non-deterministic (may get different output based on different start values) but guaranteed to converge
- Iterative algorithm with two sub-steps (after random cluster centroids chosen):
  1. Assign points to nearest cluster
  2. Recompute cluster centroid

- Select number of clusters by exploring error (distortion)

Are there other methods of clustering?

© 2019-22 D. Jayaraman, O. Bastani, Z. Ives

Hierarchical Clustering

Hierarchical Clustering

Instead of repeating until convergence based on global measures, like k-Means –

Let's consider a greedy *local* algorithm, that iteratively makes choices

1. Start with **single-item clusters**, then build successively bigger and bigger clusters: *agglomerative*
2. Or: **start with one cluster**, break into the most logical sub-clusters, repeat: *divisive*

These are called *hierarchical clustering* approaches…

© 2019-22 D. Jayaraman, O. Bastani, Z. Ives

Basic Intuition

- Agglomerative: In each iteration, we find the closest clusters and merge them
- Divisive: In each iteration, we find the two most distant sub-clusters and split there

But: we know how to compute differences in points – need to generalize this to computing distances among clusters

© 2019-22 D. Jayaraman, O. Bastani, Z. Ives

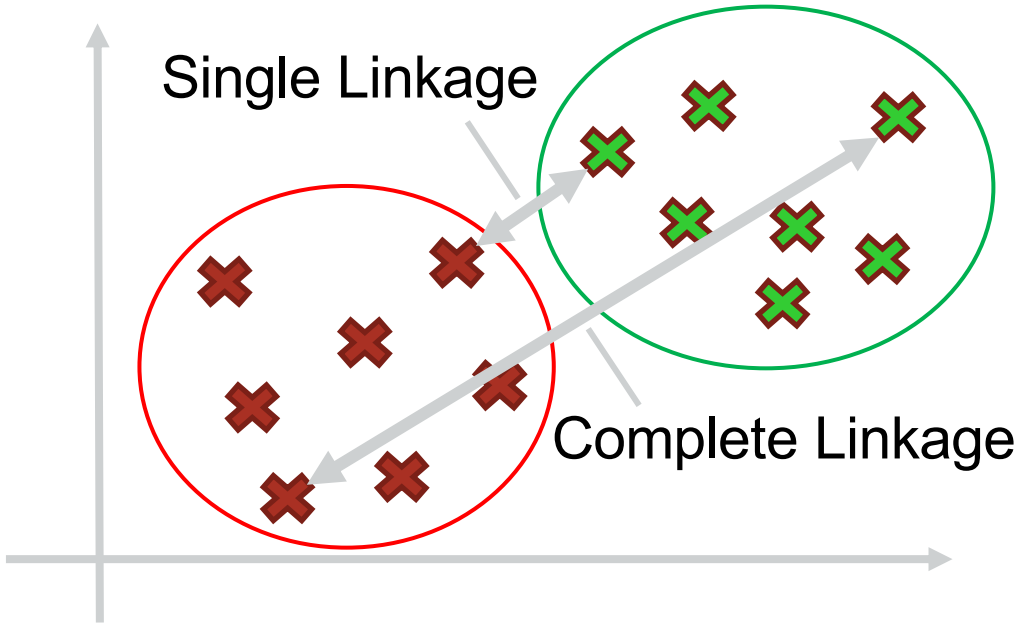
Potential Cluster Differences

Single Linkage: Compute distances between the **most similar** members for each pair of clusters

Merge the clusters with the smallest **min-distance**

Complete Linkage: Compute distance between the **most dissimilar** members for each pair of clusters

Merge the clusters with the smallest **max-distance**



© 2019-22 D. Jayaraman, O. Bastani, Z. Ives

How Do We Do this Efficiently?
Considering Agglomerative Case

Really inefficient to iterate over each point and compute its distance to every other point

- $O(n^2)$ computations, which is bad for big data!

Idea: **precompute and memorize**:

- Compute a **distance matrix** where **distance**[*i,j*] is the distance between nodes *i,j*
- We'll update this matrix every time we merge

© 2019-22 D. Jayaraman, O. Bastani, Z. Ives

Pseudocode: Agglomerative Clustering with Complete Linkage

single

Agglomerative Clustering:

1. Compute distance matrix **dist** between all pairs of points (a,b)
2. Repeat:

    Iterate over pairs of clusters A, B, compute their distance:

        Look at all pairwise distances **dist[a,b]** between $a \in A, b \in B$

    Merge the pair of clusters with *min* distance between most distant members

    least

    Update the distance matrix

    Until a single cluster remains

© 2019-22 D. Jayaraman, O. Bastani, Z. Ives

Example: Reconstructing Phylogenetic Trees

Q: Is a panda a bear?

Or is its closest relative the red panda, which is related to the raccoon?

?

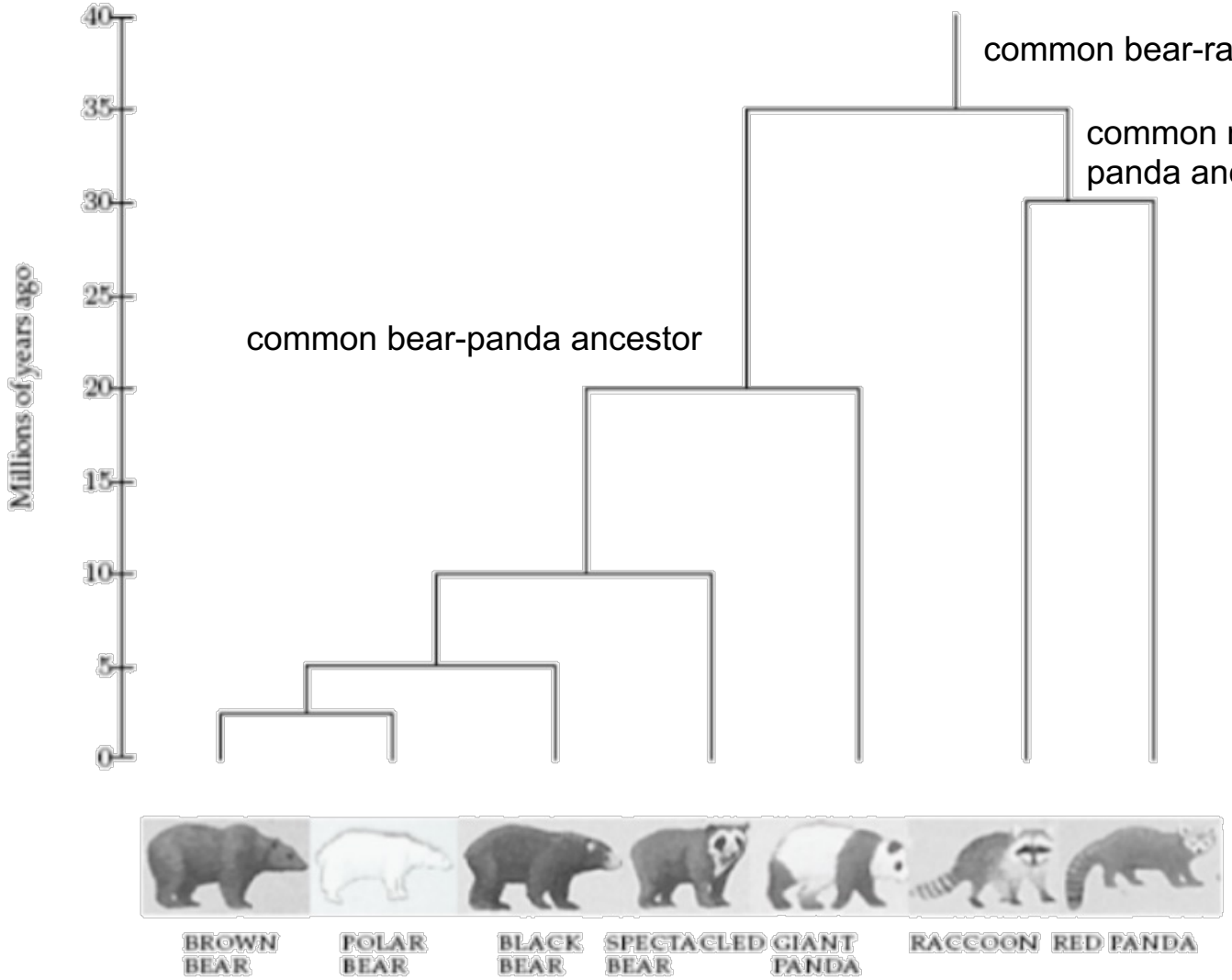
https://www.smithsonianmag.com/science-nature/eight-amazing-facts-about-red-pandas-180979708/

https://www.nwf.org/Educational-Resources/Wildlife-Gu

© 2019-22 D. Jayaraman, O. Bastani, Z. Ives

https://towardsdatascience.com/hierarchical-clustering-and-its-applications-41c1ad4441a6

Example: Reconstructing Phylogenetic Trees



Features: gene sequences

Clustering coefficient:
gene sequence
similarity via edit distances

Plot "dendrogram"

→ Panda is more closely related to modern bears, not red pandas!

© 2019-22 D. Jayaraman, O. Bastani, Z. Ives

https://towardsdatascience.com/hierarchical-clustering-and-its-applications-41c1ad4441a6

Summary of Hierarchical Clustering

Hierarchical clustering is often easier to visualize and interpret with a taxonomy

"Dendrogram" plots

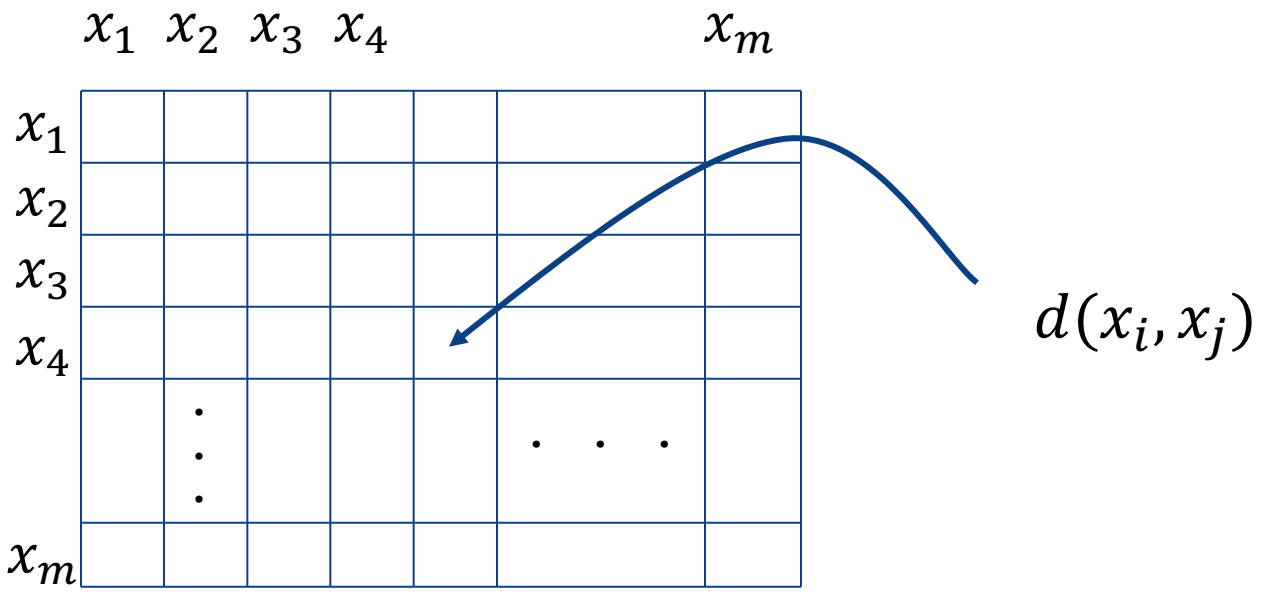
We don't need to specify the number of clusters up front!

Limitation: Doesn't scale well to big problems

© 2019-22 D. Jayaraman, O. Bastani, Z. Ives

Distance Measures

- In studying Clustering techniques, we assume that we are given a matrix of distances between all pairs of data points.
- We assume that the input to the problem is:



© 2019-22 D. Jayaraman, O. Bastani, Z. Ives

Distance Measures

- In studying Clustering techniques, we assume that we are given a matrix of distances between all pairs of data points.
- A distance measure (metric) is a function $d: \boldsymbol{R}^d \times \boldsymbol{R}^d \rightarrow \boldsymbol{R}$ that satisfies:

$$1.\ d(x,y) \geq 0, d(x,y) = 0 \Leftrightarrow x = y$$
$$2.\ d(x,y) + d(y,z) \geq d(x,z)$$
$$3.\ d(x,y) = d(y,x)$$

- For the purpose of clustering, sometimes the distance (similarity) is **not required to be a metric**
  - No Triangle Inequality
  - No Symmetry

© 2019-22 D. Jayaraman, O. Bastani, Z. Ives

Distance Measures

Examples:

- Euclidean Distance:

$$d(x,y) = \sqrt{(x-y)^2} = \sqrt{(x-y)^T\,(x-y)} = \sqrt{\sum_{i=1}^{d}(x_i - y_i\,)^2}$$

- Manhattan Distance:

$$d(x,y) = |x-y| \; = \sum_{i=1}^{d} |x_i - y_i\,|$$

- Infinity (Sup) Distance:

$$d(x,y) = \max_{1 \le i \le d} |x_i - y_i|$$

- Notice that if $d(x,y)$ is the Euclidean metric, $d^2(x,y)$ is not a metric
- But can be used as a measure (no triangle inequality)

© 2019-22 D. Jayaraman, O. Bastani, Z. Ives

Distance Measures

- Examples:
  - Euclidean Distance:

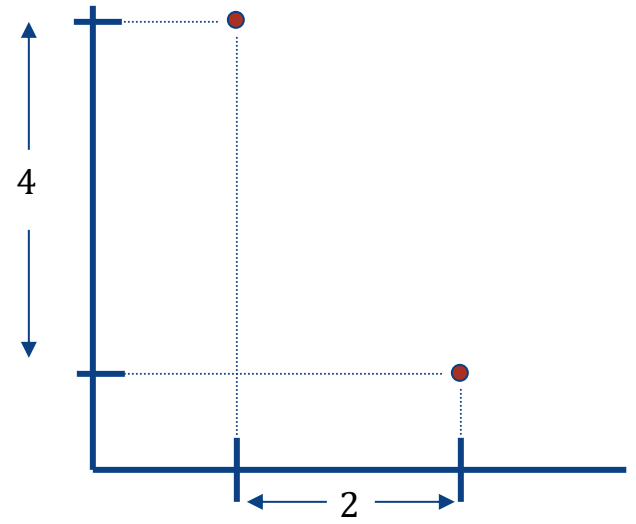
$$d(x,y) = \sqrt{(x-y)^2} = \sqrt{(x-y)^T\,(x-y)} = \sqrt{\sum_{i=1}^{d}(x_i - y_i\,)^2}$$

  - Manhattan Distance:

$$d(x,y) = |x-y| \; = \sum_{i=1}^{d} |x_i - y_i\,|$$

  - Infinity (Sup) Distance:

$$d(x,y) = \max_{1\le i\le d} |x_i - y_i|$$



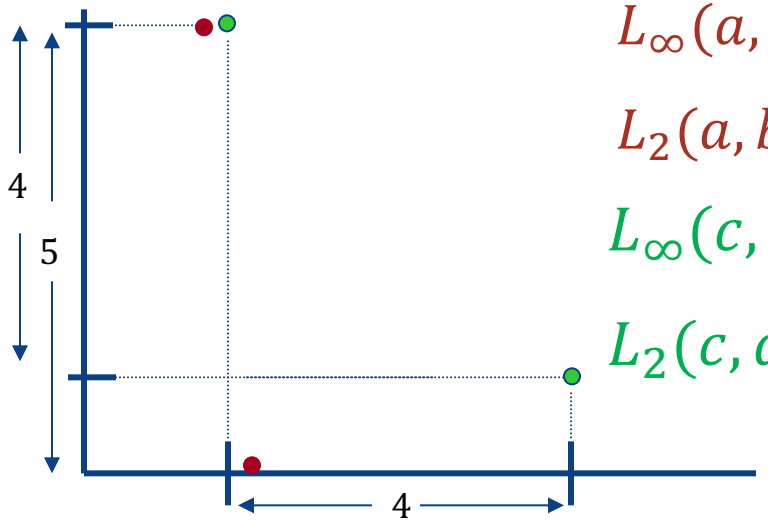
Euclidean: $(4^2 + 2^2)^{\frac{1}{2}} = 4.47$
Manhattan: $4 + 2 = 6$
Sup: $Max(4,2) = 4$

Distance Measures

- Notice that:
  - Infinity (Sup) Distance < Euclidean Distance < Manhattan Distance:

$$L_\infty = \max_{1 \le i \le d} |x_i - y_i| \quad L_2 = \sqrt{(x-y)^2} = \sqrt{\sum_{i=1}^{d} (x_i - y_i)^2} \quad L_1 = |x-y| = \sum_{i=1}^{d} |x_i - y_i|$$

- But different distances do not induce same order on pairs of points



$$L_\infty(a,b) = 5$$

$$L_2(a,b) = (5^2 + \varepsilon^2)^{\frac{1}{2}} = 5 + \varepsilon$$

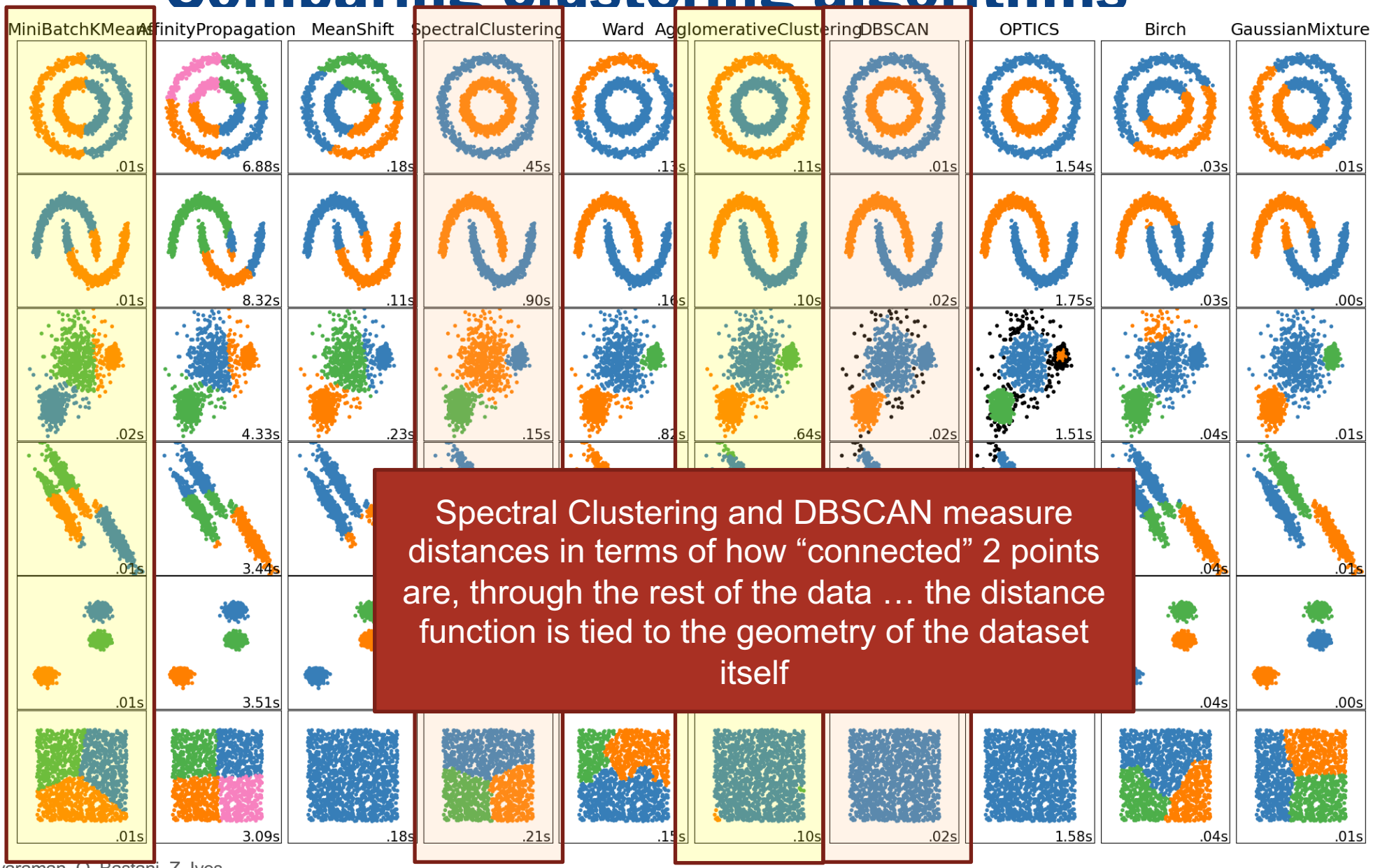
$$L_\infty(c,d) = 4$$

$$L_2(c,d) = (4^2 + 4^2)^{\frac{1}{2}} = 4\sqrt{2} = 5.66$$

$$L_\infty(c,d) < L_\infty(a,b)$$
$$L_2(c,d) > L_2(a,b)$$

© 2019-22 D. Jayaraman, O. Bastani, Z. Ives

Comparing clustering algorithms



Spectral Clustering and DBSCAN measure distances in terms of how "connected" 2 points are, through the rest of the data … the distance function is tied to the geometry of the dataset itself

© 2019-22 D. Jayaraman, O. Bastani, Z. Ives

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

Summary of Clustering

- Critical to understanding the structure of our data
- Often useful for creating high-level features useful for supervised learning
- We saw two approaches: k-Means vs hierarchical clustering

© 2019-22 D. Jayaraman, O. Bastani, Z. Ives