Today’s material available at:

https://github.com/juliennelachance/ai4all_clustering

Clustering

Princeton AI4ALL

By Julie LaChance

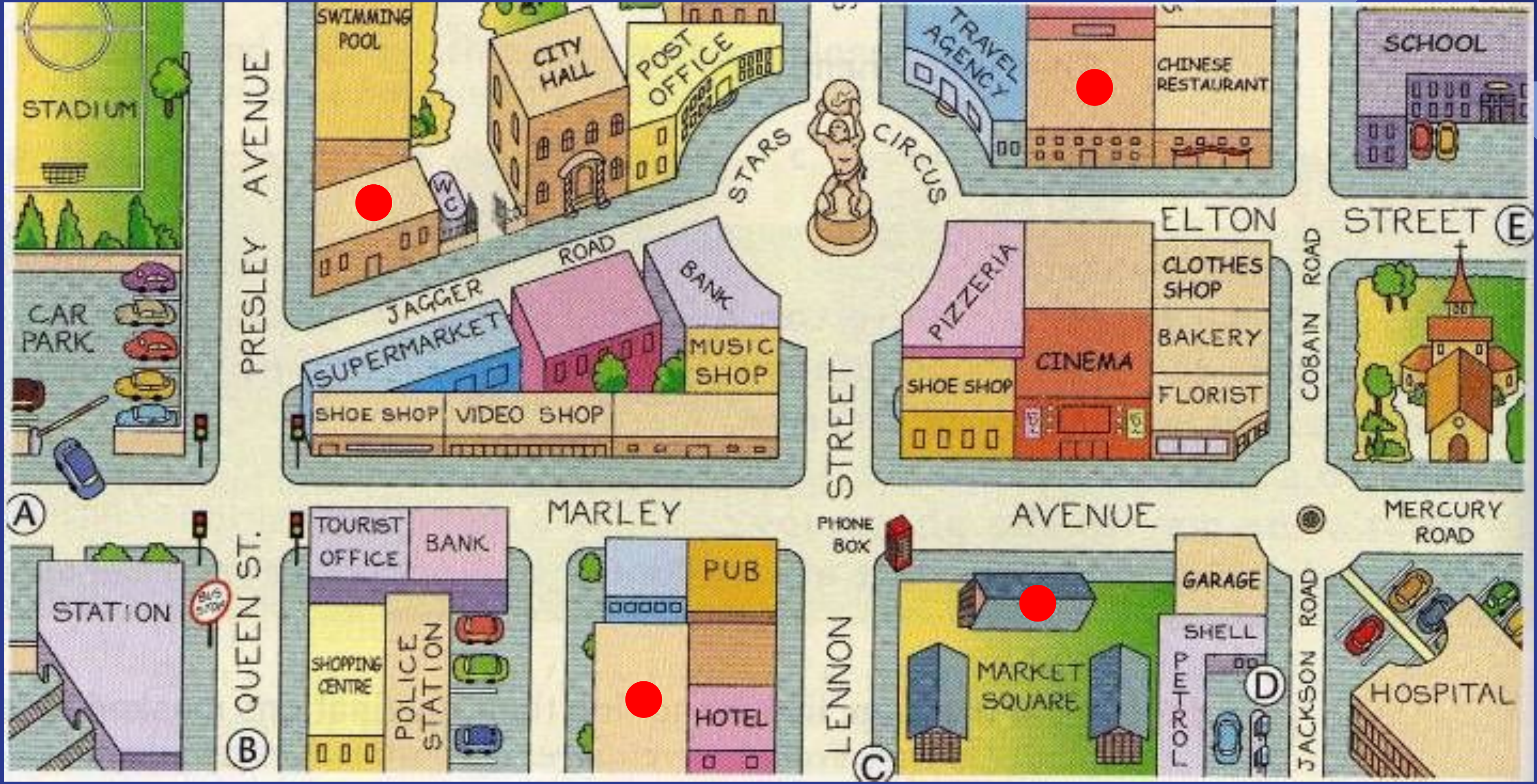
Icebreaker

- Everyone receives a slip of paper

- Your task:

  Using the information about your person, form dog-walking groups

- Be prepared to justify your answers!
  - What features are important and why?

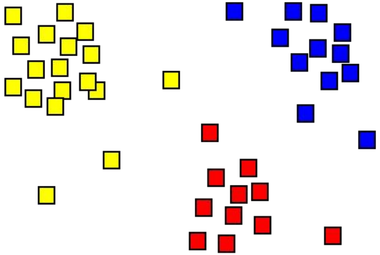

STADIUM
CAR PARK
SWIMMING POOL
WC
CITY HALL
POST OFFICE
STARS CIRCUS
TRAVEL AGENCY
CHINESE RESTAURANT
SCHOOL
ELTON STREET
E
PRESLEY AVENUE
JAGGER ROAD
SUPERMARKET
BANK
MUSIC SHOP
SHOE SHOP
VIDEO SHOP
PIZZERIA
CINEMA
CLOTHES SHOP
BAKERY
FLORIST
SHOE SHOP
COBAIN ROAD
LENNON STREET
A
MARLEY AVENUE
PHONE BOX
MERCURY ROAD
STATION
BUS STOP
QUEEN ST.
B
TOURIST OFFICE
BANK
SHOPPING CENTRE
POLICE STATION
PUB
HOTEL
C
MARKET SQUARE
GARAGE
SHELL
PETROL
D
JACKSON ROAD
HOSPITAL

Overview:

- What is clustering?
- Applications of clustering
- Types of clustering
- k-means algorithm
- Recommendation system example

What is clustering?

What is clustering?



- From Wikipedia:

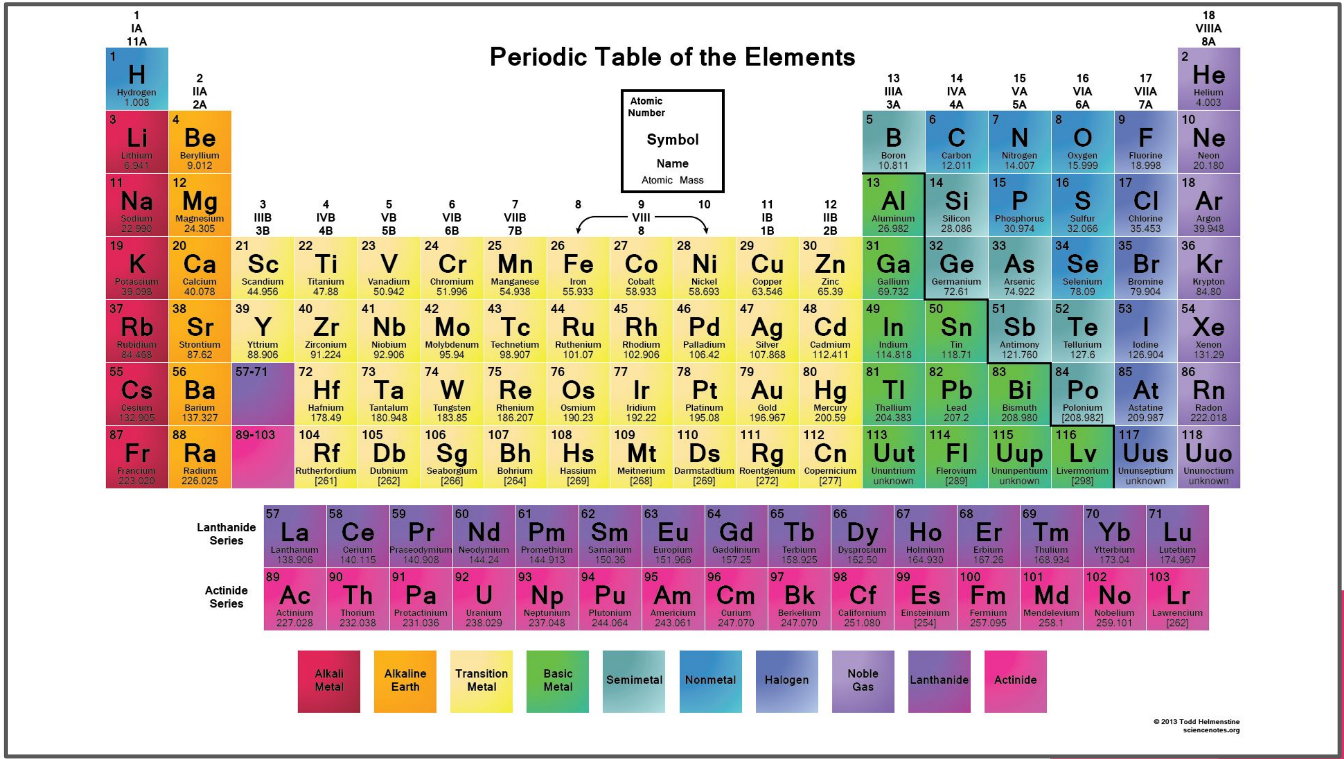
  “**Cluster analysis** or **clustering** is the task of grouping a set of objects in such a way that objects in the same group (called a **cluster**) are more similar (in some sense) to each other than to those in other groups (clusters).”

- Thinking back to our first lecture on machine learning:

  This is an important type of ***unsupervised learning***. Why?

Applications of clustering

Periodic Table of Elements



Periodic Table of the Elements

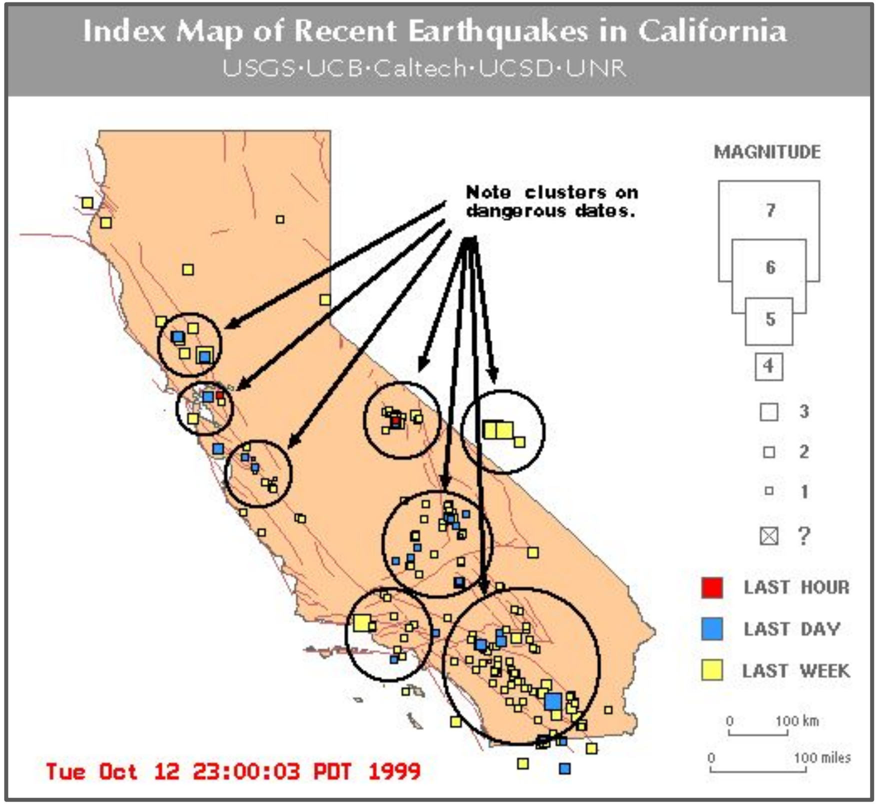
Atomic Number
Symbol
Name
Atomic Mass

1 IA 11A	2 IIA 2A	3 IIIB 3B	4 IVB 4B	5 VB 5B	6 VIB 6B	7 VIIB 7B	8	9 VIII 8	10	11 IB 1B	12 IIB 2B	13 IIIA 3A	14 IVA 4A	15 VA 5A	16 VIA 6A	17 VIIA 7A	18 VIIIA 8A
1 H Hydrogen 1.008																	2 He Helium 4.003
3 Li Lithium 6.941	4 Be Beryllium 9.012											5 B Boron 10.811	6 C Carbon 12.011	7 N Nitrogen 14.007	8 O Oxygen 15.999	9 F Fluorine 18.998	10 Ne Neon 20.180
11 Na Sodium 22.990	12 Mg Magnesium 24.305											13 Al Aluminum 26.982	14 Si Silicon 28.086	15 P Phosphorus 30.974	16 S Sulfur 32.066	17 Cl Chlorine 35.453	18 Ar Argon 39.948
19 K Potassium 39.098	20 Ca Calcium 40.078	21 Sc Scandium 44.956	22 Ti Titanium 47.88	23 V Vanadium 50.942	24 Cr Chromium 51.996	25 Mn Manganese 54.938	26 Fe Iron 55.933	27 Co Cobalt 58.933	28 Ni Nickel 58.693	29 Cu Copper 63.546	30 Zn Zinc 65.39	31 Ga Gallium 69.732	32 Ge Germanium 72.61	33 As Arsenic 74.922	34 Se Selenium 78.09	35 Br Bromine 79.904	36 Kr Krypton 84.80
37 Rb Rubidium 84.468	38 Sr Strontium 87.62	39 Y Yttrium 88.906	40 Zr Zirconium 91.224	41 Nb Niobium 92.906	42 Mo Molybdenum 95.94	43 Tc Technetium 98.907	44 Ru Ruthenium 101.07	45 Rh Rhodium 102.906	46 Pd Palladium 106.42	47 Ag Silver 107.868	48 Cd Cadmium 112.411	49 In Indium 114.818	50 Sn Tin 118.71	51 Sb Antimony 121.760	52 Te Tellurium 127.6	53 I Iodine 126.904	54 Xe Xenon 131.29
55 Cs Cesium 132.905	56 Ba Barium 137.327	57-71	72 Hf Hafnium 178.49	73 Ta Tantalum 180.948	74 W Tungsten 183.85	75 Re Rhenium 186.207	76 Os Osmium 190.23	77 Ir Iridium 192.22	78 Pt Platinum 195.08	79 Au Gold 196.967	80 Hg Mercury 200.59	81 Tl Thallium 204.383	82 Pb Lead 207.2	83 Bi Bismuth 208.980	84 Po Polonium [208.982]	85 At Astatine 209.987	86 Rn Radon 222.018
87 Fr Francium 223.020	88 Ra Radium 226.025	89-103	104 Rf Rutherfordium [261]	105 Db Dubnium [262]	106 Sg Seaborgium [266]	107 Bh Bohrium [264]	108 Hs Hassium [269]	109 Mt Meitnerium [268]	110 Ds Darmstadtium [269]	111 Rg Roentgenium [272]	112 Cn Copernicium [277]	113 Uut Ununtrium unknown	114 Fl Flerovium [289]	115 Uup Ununpentium unknown	116 Lv Livermorium [298]	117 Uus Ununseptium unknown	118 Uuo Ununoctium unknown

Lanthanide Series	57 La Lanthanum 138.906	58 Ce Cerium 140.115	59 Pr Praseodymium 140.908	60 Nd Neodymium 144.24	61 Pm Promethium 144.913	62 Sm Samarium 150.36	63 Eu Europium 151.966	64 Gd Gadolinium 157.25	65 Tb Terbium 158.925	66 Dy Dysprosium 162.50	67 Ho Holmium 164.930	68 Er Erbium 167.26	69 Tm Thulium 168.934	70 Yb Ytterbium 173.04	71 Lu Lutetium 174.967
Actinide Series	89 Ac Actinium 227.028	90 Th Thorium 232.038	91 Pa Protactinium 231.036	92 U Uranium 238.029	93 Np Neptunium 237.048	94 Pu Plutonium 244.064	95 Am Americium 243.061	96 Cm Curium 247.070	97 Bk Berkelium 247.070	98 Cf Californium 251.080	99 Es Einsteinium [254]	100 Fm Fermium 257.095	101 Md Mendelevium 258.1	102 No Nobelium 259.101	103 Lr Lawrencium [262]

Alkali Metal	Alkaline Earth	Transition Metal	Basic Metal	Semimetal	Nonmetal	Halogen	Noble Gas	Lanthanide	Actinide

Earthquakes and Seismology



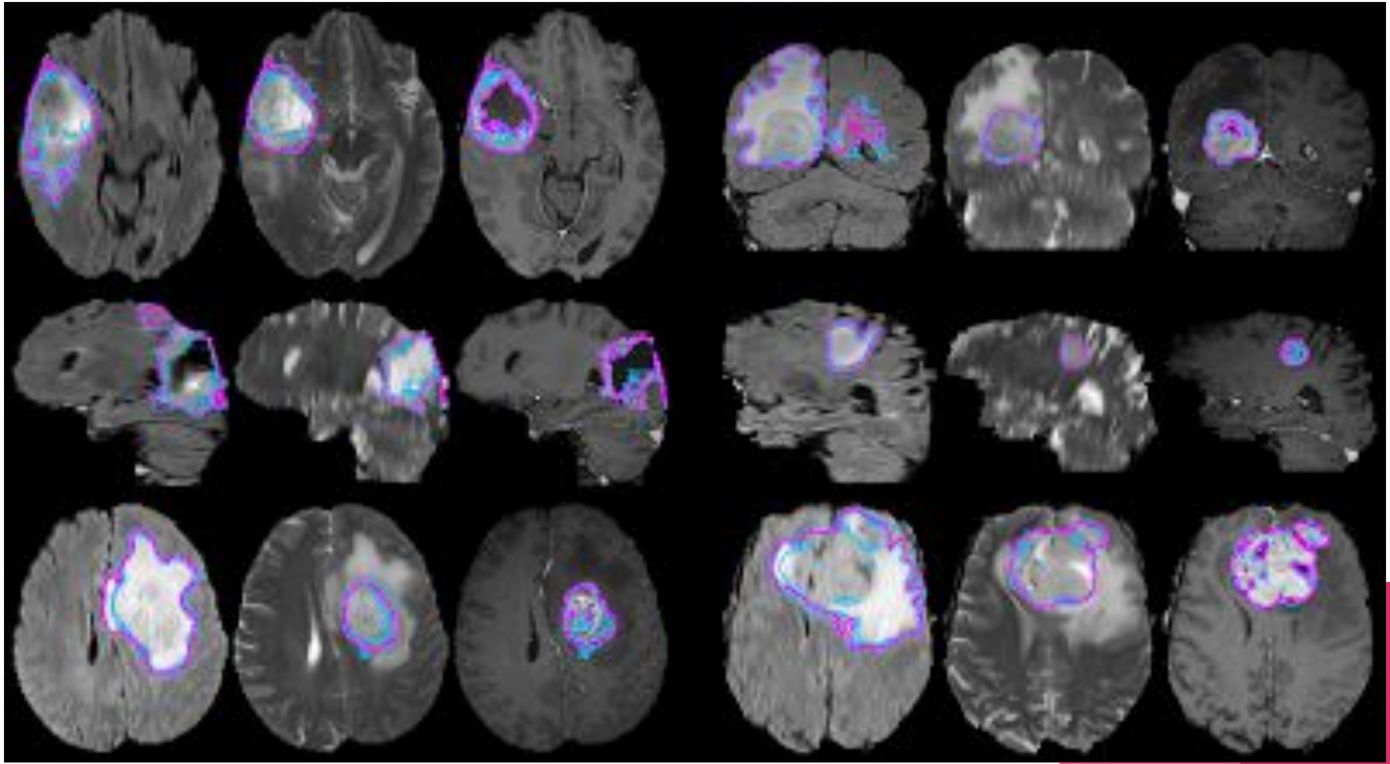
Index Map of Recent Earthquakes in California

USGS·UCB·Caltech·UCSD·UNR

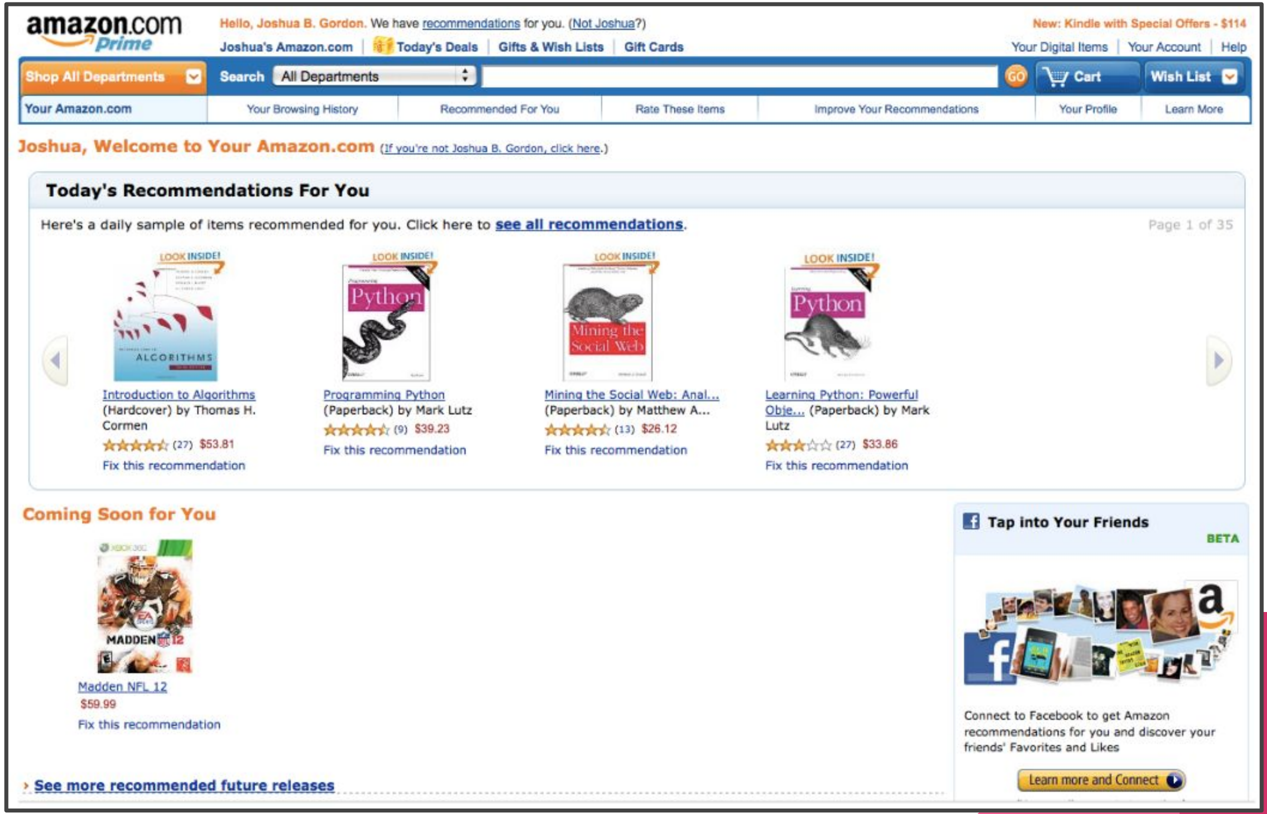
Note clusters on dangerous dates.

Tue Oct 12 23:00:03 PDT 1999

Medical Imaging



Recommendation Systems: Amazon



Hello, Joshua B. Gordon. We have recommendations for you. (Not Joshua?)

New: Kindle with Special Offers - $114

Joshua, Welcome to Your Amazon.com (If you're not Joshua B. Gordon, click here.)

Today's Recommendations For You

Here's a daily sample of items recommended for you. Click here to see all recommendations.

Page 1 of 35

Introduction to Algorithms (Hardcover) by Thomas H. Cormen
★★★★★ (27) $53.81
Fix this recommendation

Programming Python (Paperback) by Mark Lutz
★★★★★ (9) $39.23
Fix this recommendation

Mining the Social Web: Anal... (Paperback) by Matthew A...
★★★★★ (13) $26.12
Fix this recommendation

Learning Python: Powerful Obje... (Paperback) by Mark Lutz
★★★☆☆ (27) $33.86
Fix this recommendation

Coming Soon for You

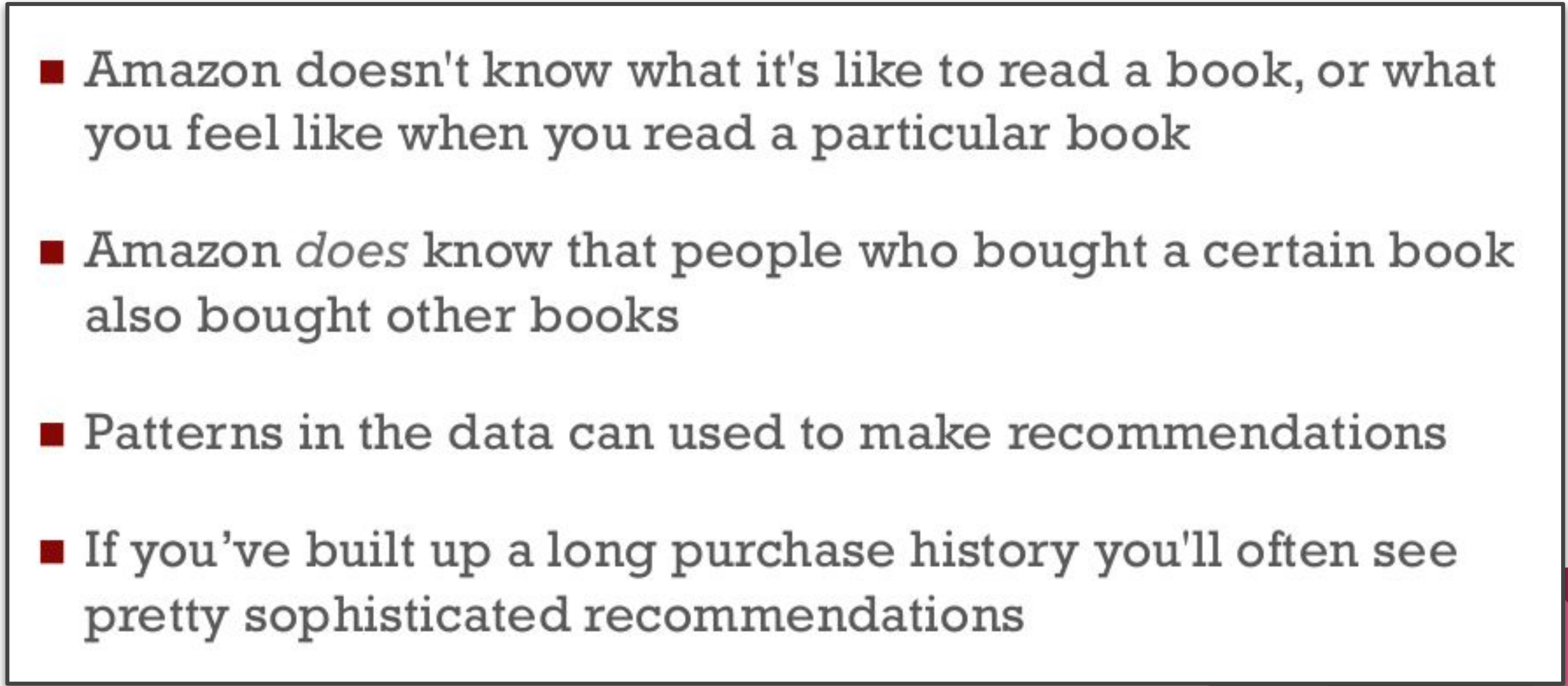
Madden NFL 12
$59.99
Fix this recommendation

› See more recommended future releases

Tap into Your Friends

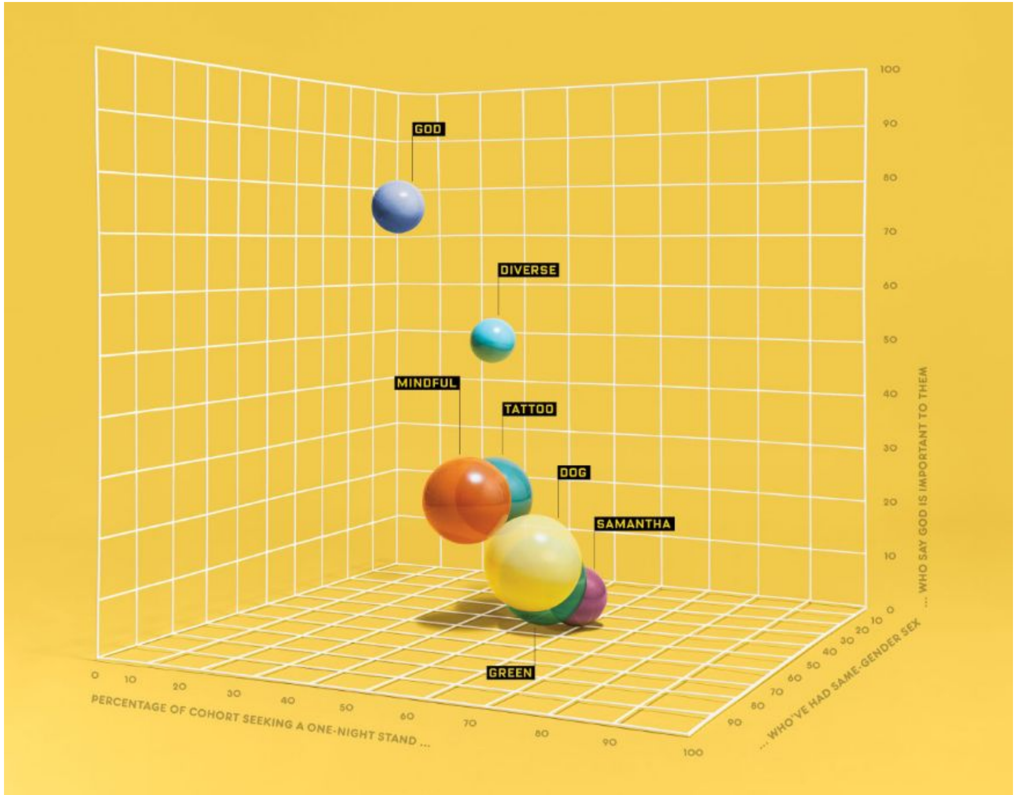
Connect to Facebook to get Amazon recommendations for you and discover your friends' Favorites and Likes

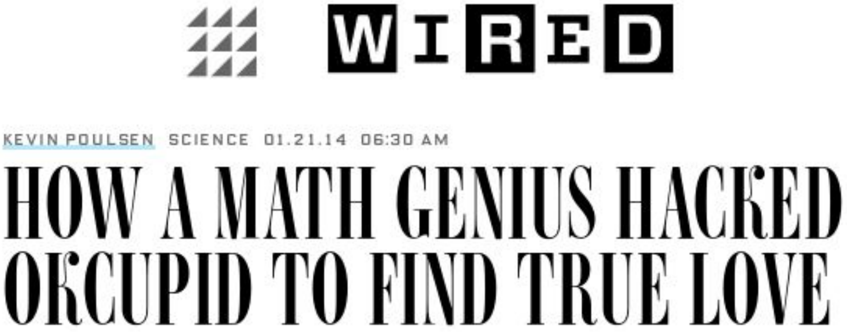
Recommendation Systems: Amazon



- Amazon doesn't know what it's like to read a book, or what you feel like when you read a particular book
- Amazon *does* know that people who bought a certain book also bought other books
- Patterns in the data can used to make recommendations
- If you’ve built up a long purchase history you'll often see pretty sophisticated recommendations

And even more nefarious purposes...




WIRED

KEVIN POULSEN SCIENCE 01.21.14 06:30 AM

HOW A MATH GENIUS HACKED OKCUPID TO FIND TRUE LOVE

https://www.wired.com/2014/01/how-to-hack-okcupid/

Discussion:

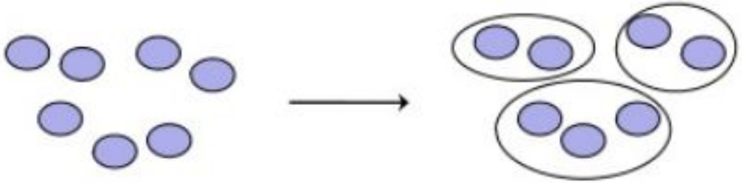
Can you come up with other applications of clustering?

Types of clustering

Two main types of clustering:

- **Hard clustering**:

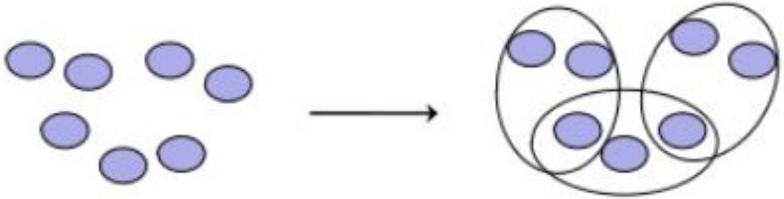
  Each object is in exactly one class.



- **Soft clustering**:

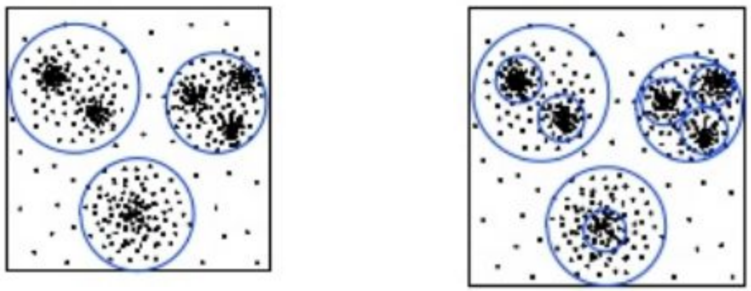
  Objects are assigned a degree to which they are in each class.

  - (Think of this as probability of being in a class)



Flat vs. Hierarchical clustering

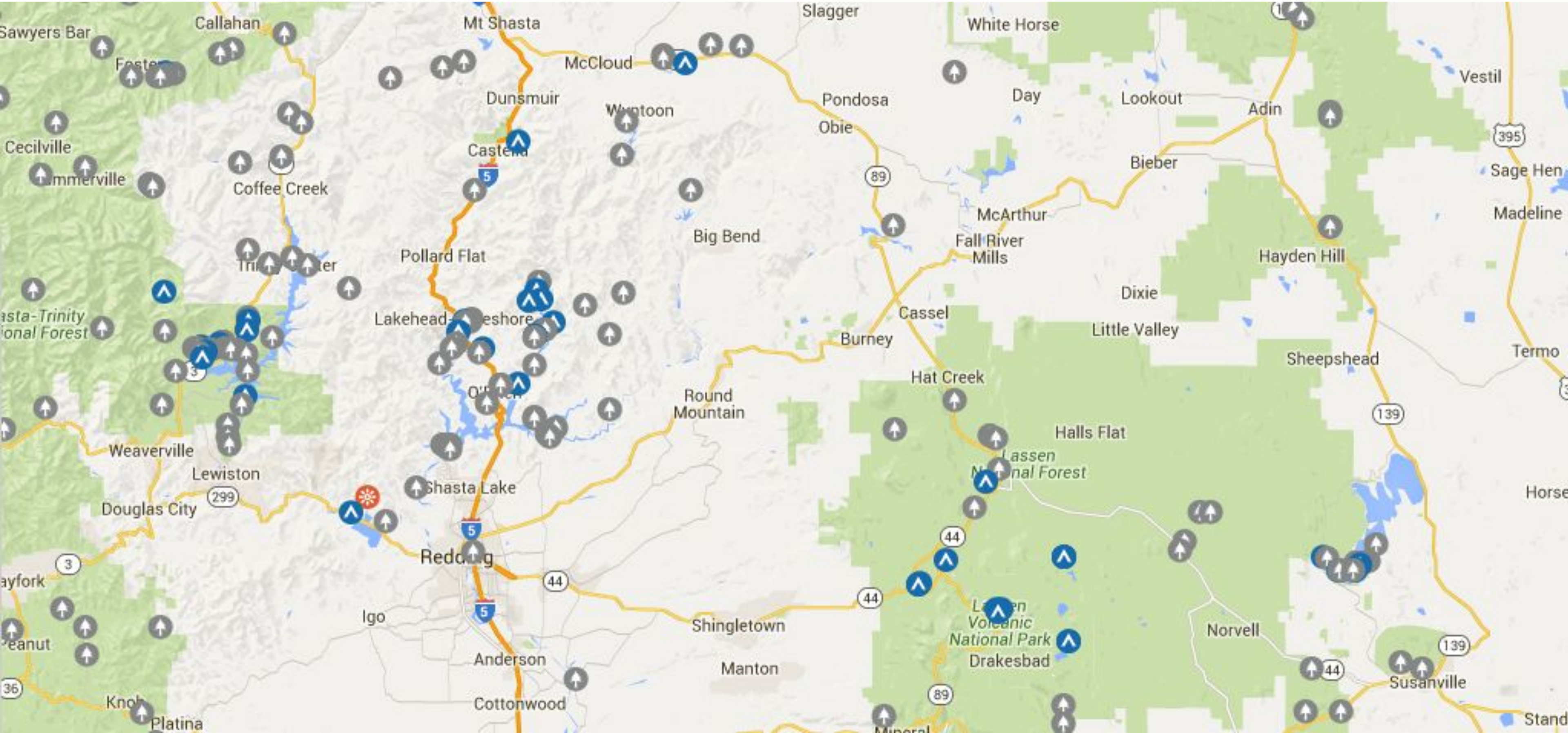
- Flat = distinct clusters
- Hierarchical = clusters within clusters



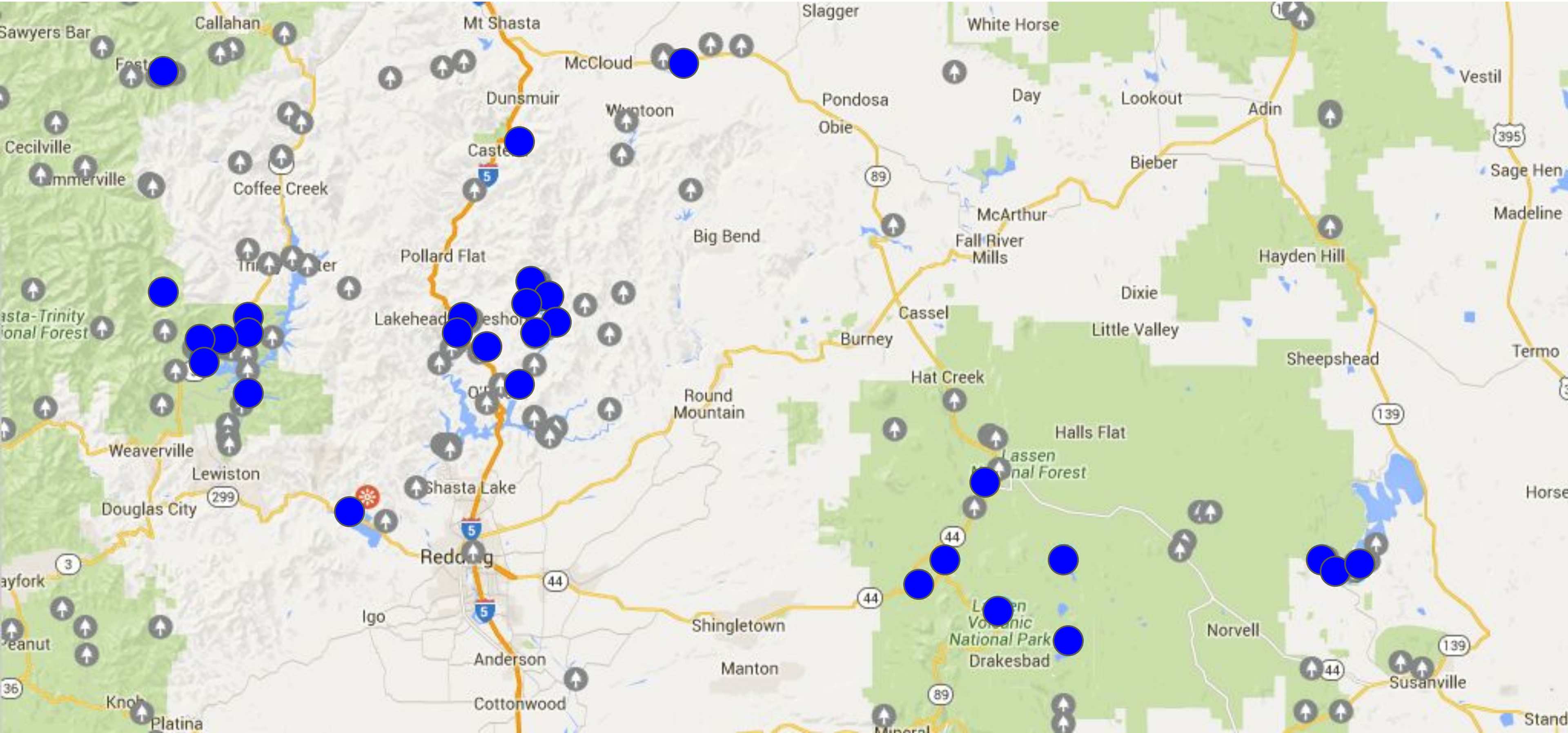
More later on types of clustering *algorithms*.

But first, an example...

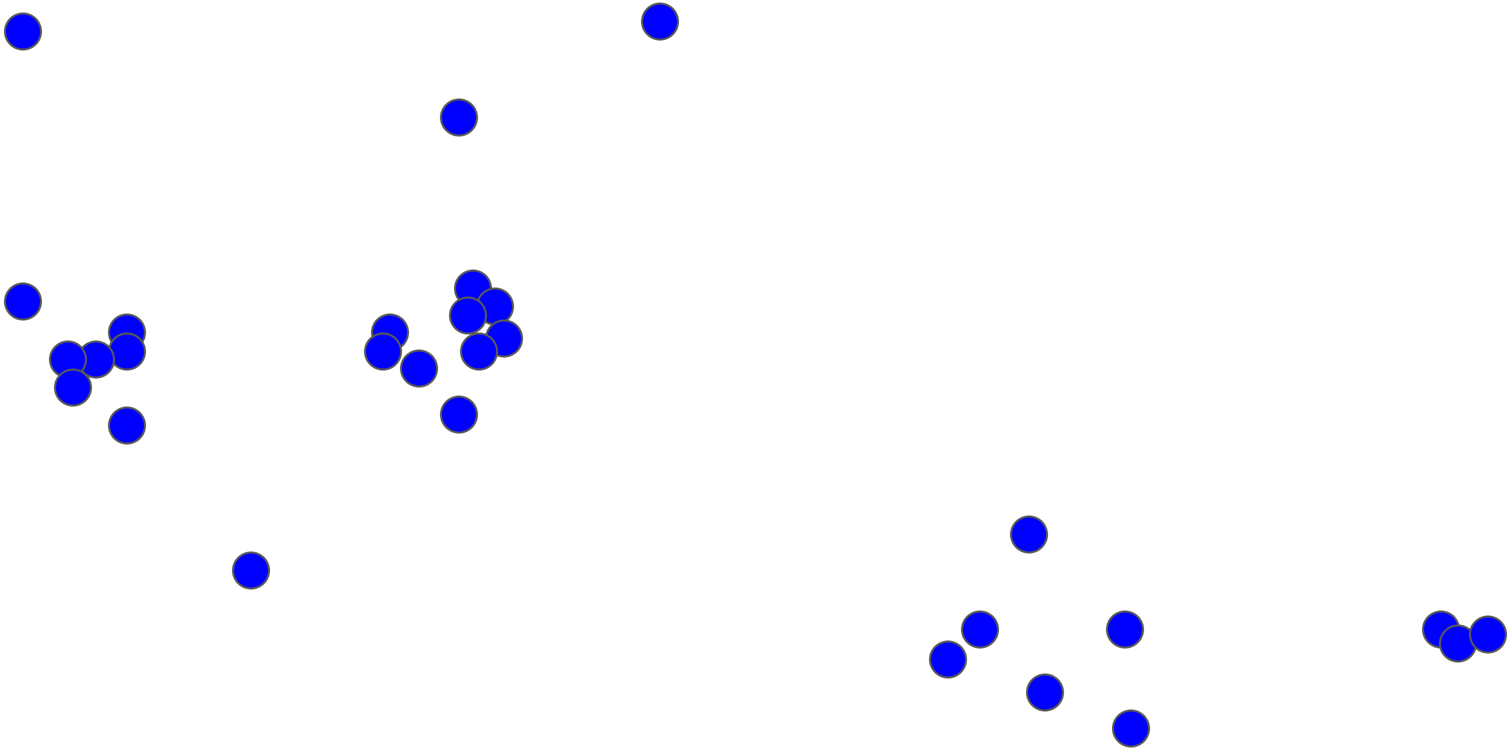
Campgrounds



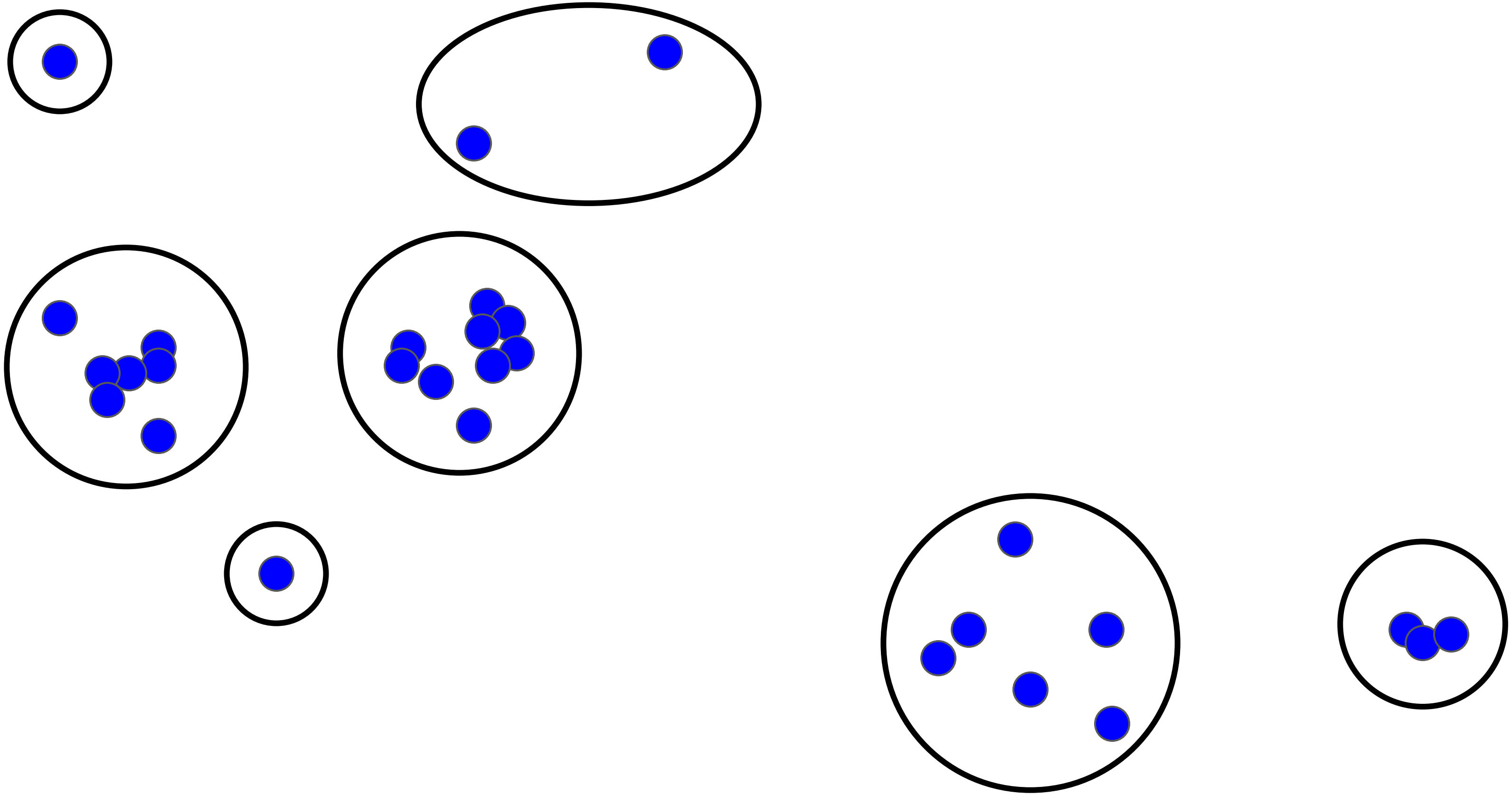
Campgrounds



Campgrounds



Campgrounds



Breakout Session:

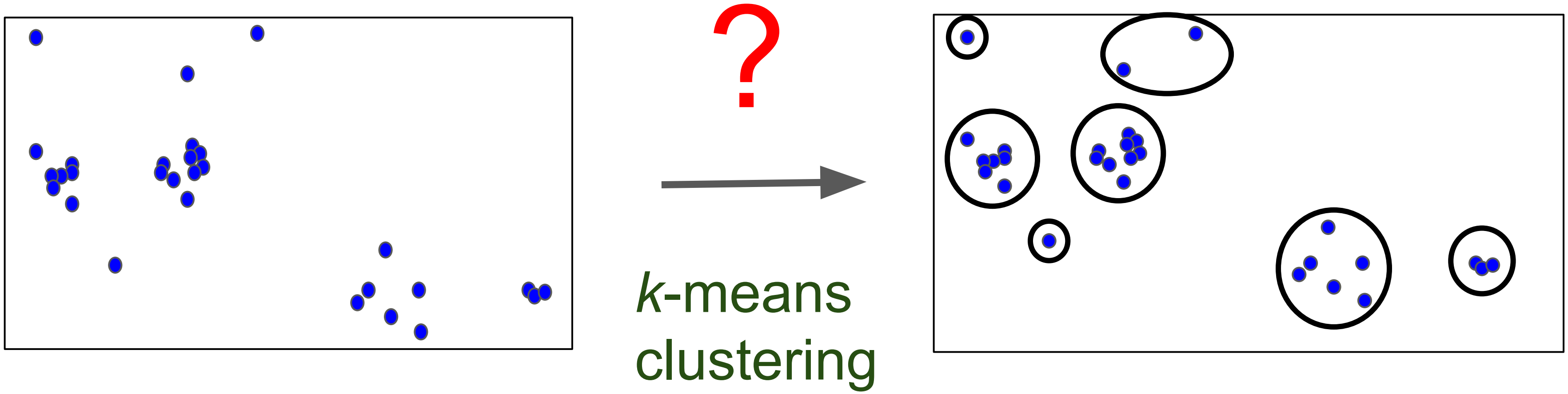
Try out the demo at http://mlehman.github.io/kmeans-javascript/

Discuss with your table: how does the algorithm work?

(Note: this demo has a bug! Can’t use more than 6 clusters)

k-means clustering

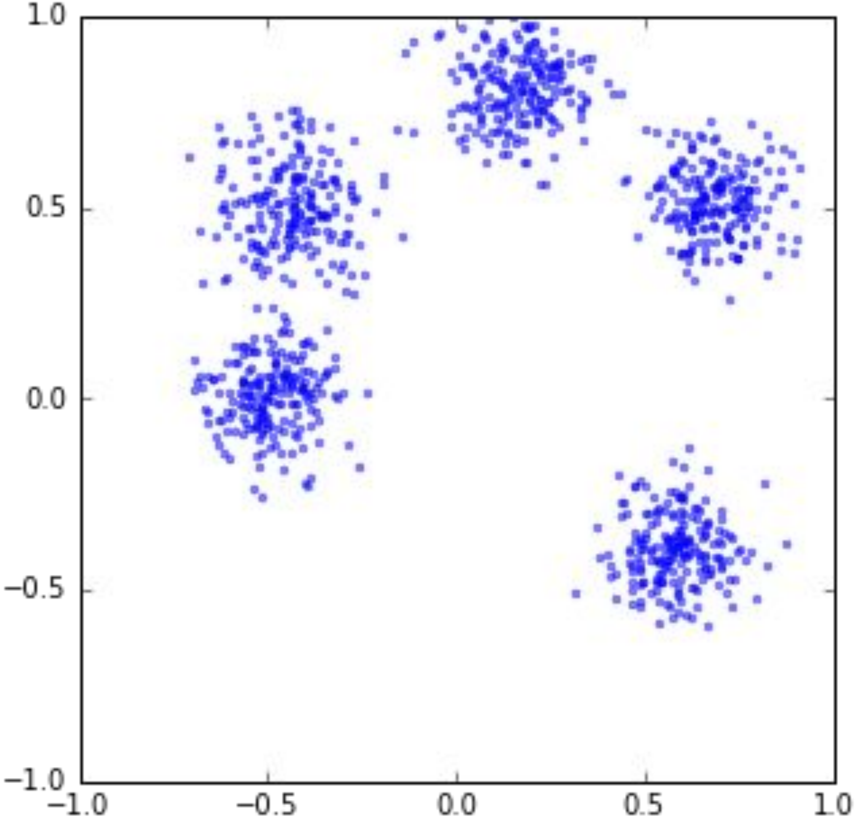
How can we automatically cluster campgrounds?



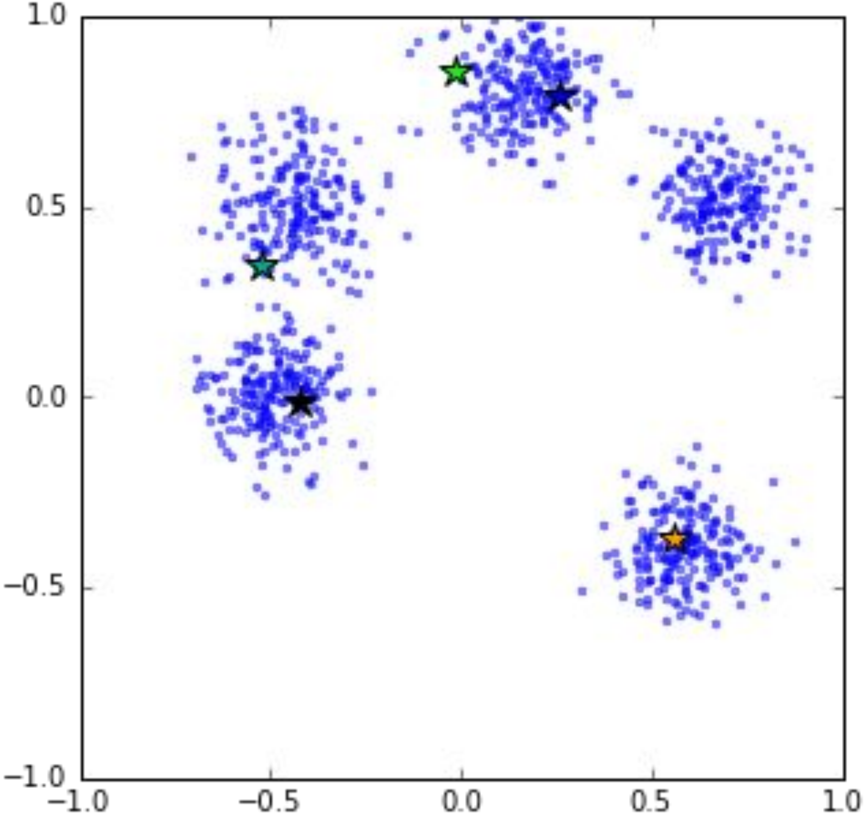
k-means clustering

Goal: Assign each of the *n* points to one out of *k* clusters (defined by a cluster centroid).

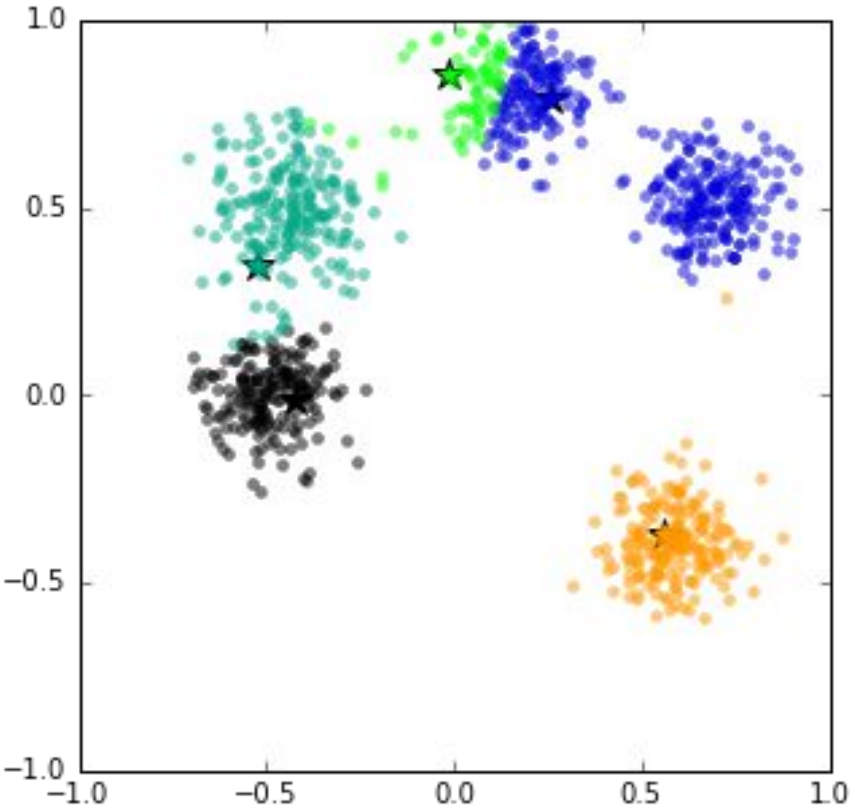
Find 5 clusters for the 1000 points below!



Step 0: Pick k random points as the cluster centroids

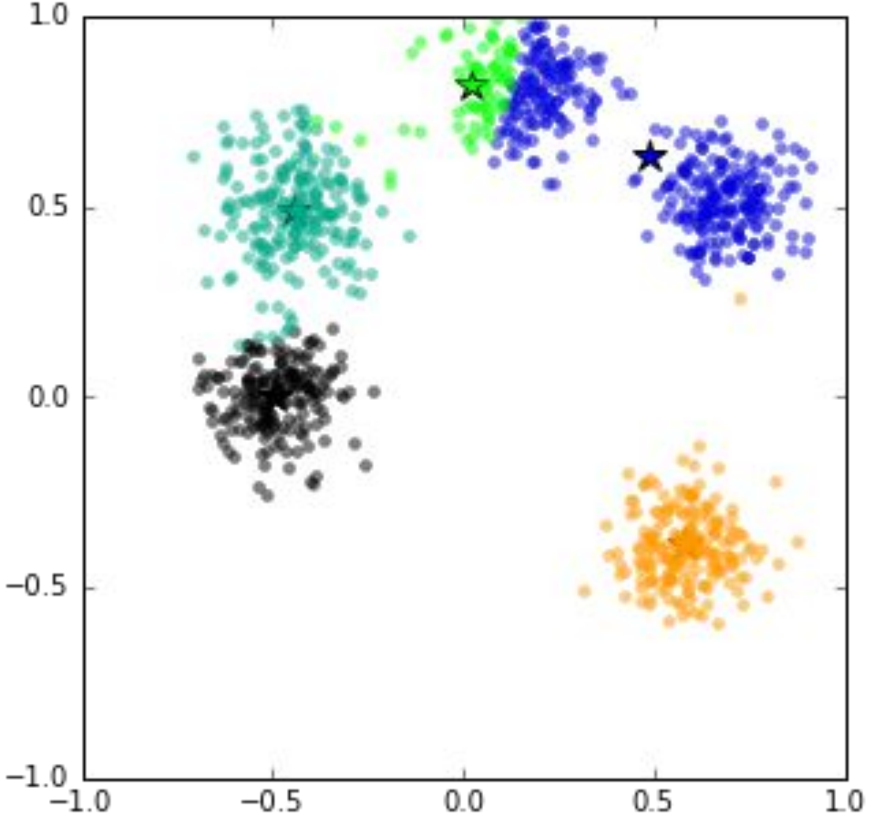


Iteration 1



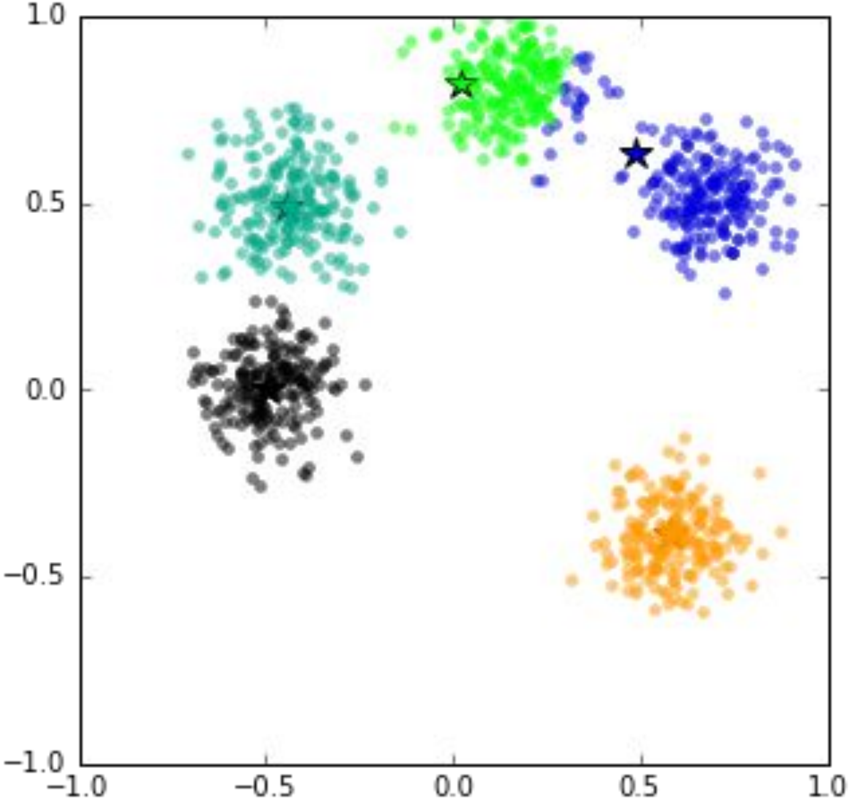
Step 1: Assign each point to closest cluster centroid

Iteration 1 continued



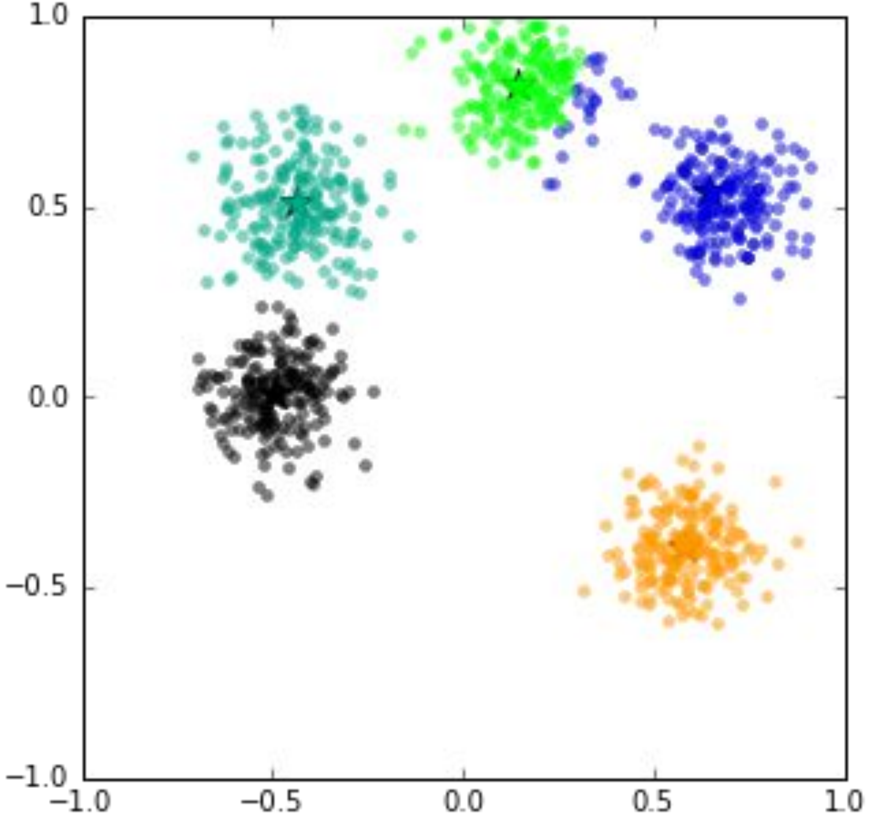
Step 2: Find cluster centroids as center of mass of elements

Iteration 2



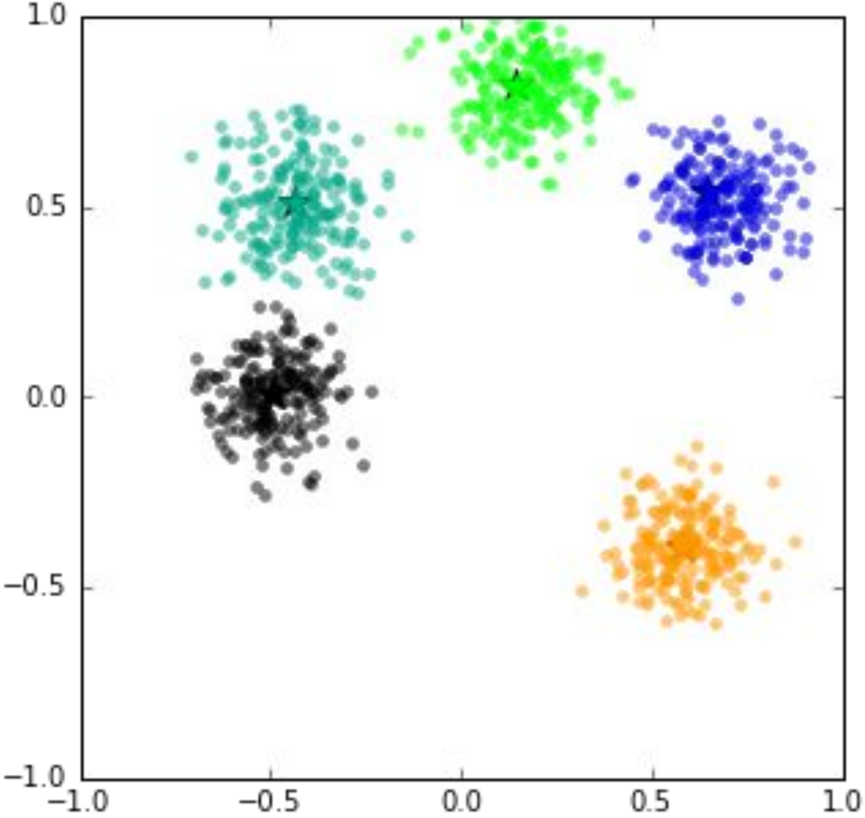
Step 1: Assign each point to closest cluster centroid

Iteration 2 continued



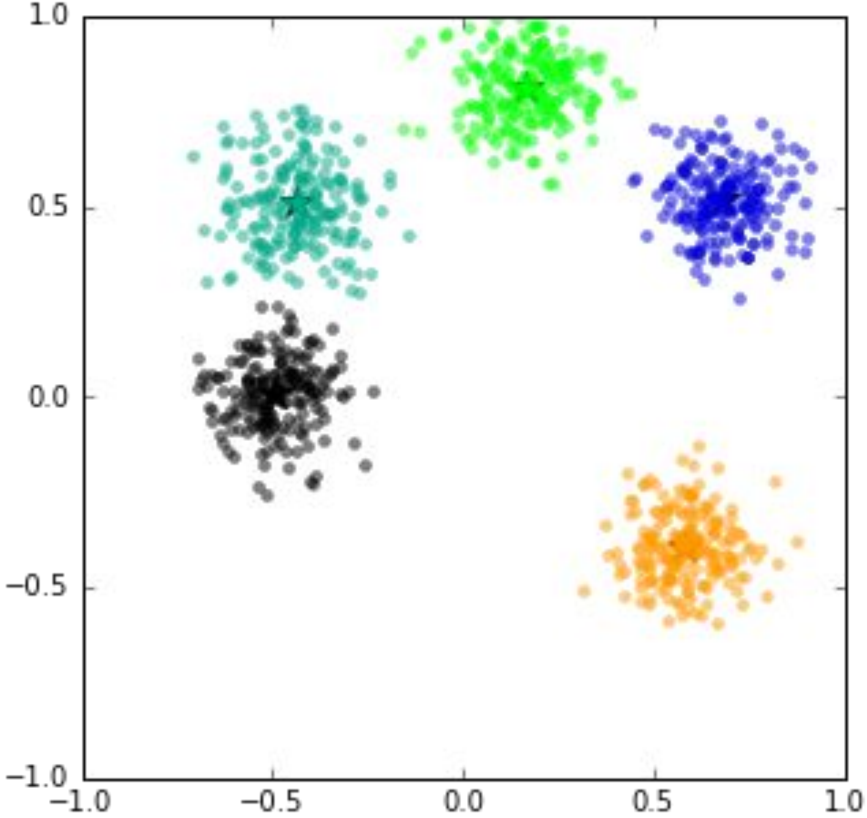
Step 2: Find cluster centroids as center of mass of elements

Iteration 3



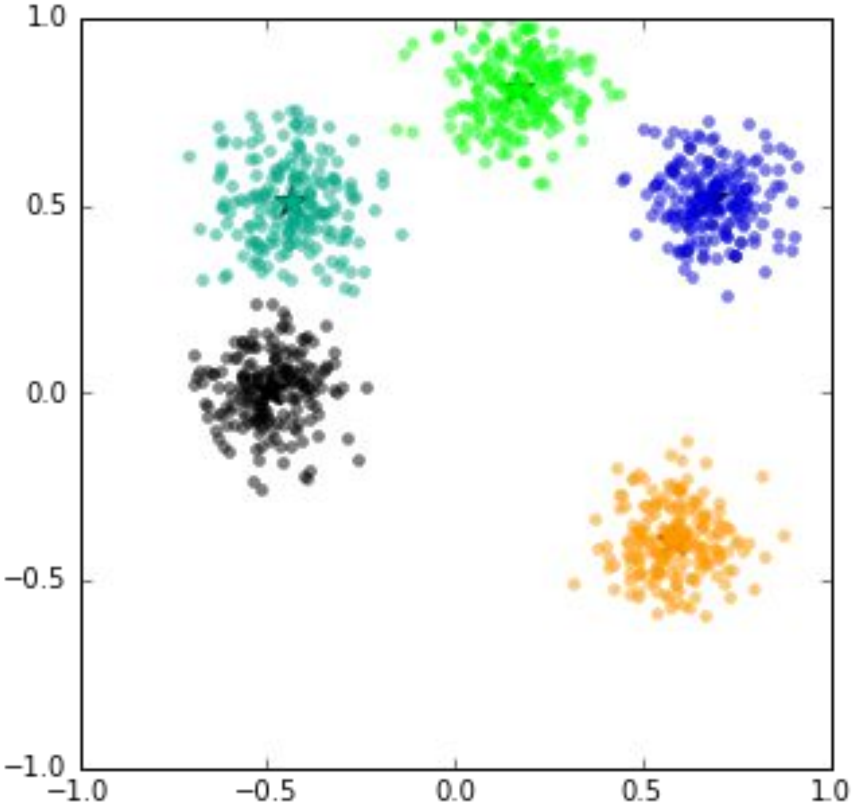
Step 1: Assign each point to closest cluster centroid

Iteration 3 continued



Step 2: Find cluster centroids as center of mass of elements

Iteration 4



Step 1: Assign each point to closest cluster centroid

No change in cluster assignment. TERMINATE!

k-means overview

- Choose the value of k
  - This is how many cluster centroids we're finding
- Choose k points in the set
  - These are the initial centroid locations
- For each point not selected, assign to its nearest centroid
  - Now all points have an initial cluster assignment
- Until "happy" do:
  - 1. Recompute centroids of clusters
  - 2. Reassign all points to closest centroid (forms new clusters)

Discussion:

Is this “hard” or “soft” clustering?

Is this “flat” or “hierarchical” clustering?

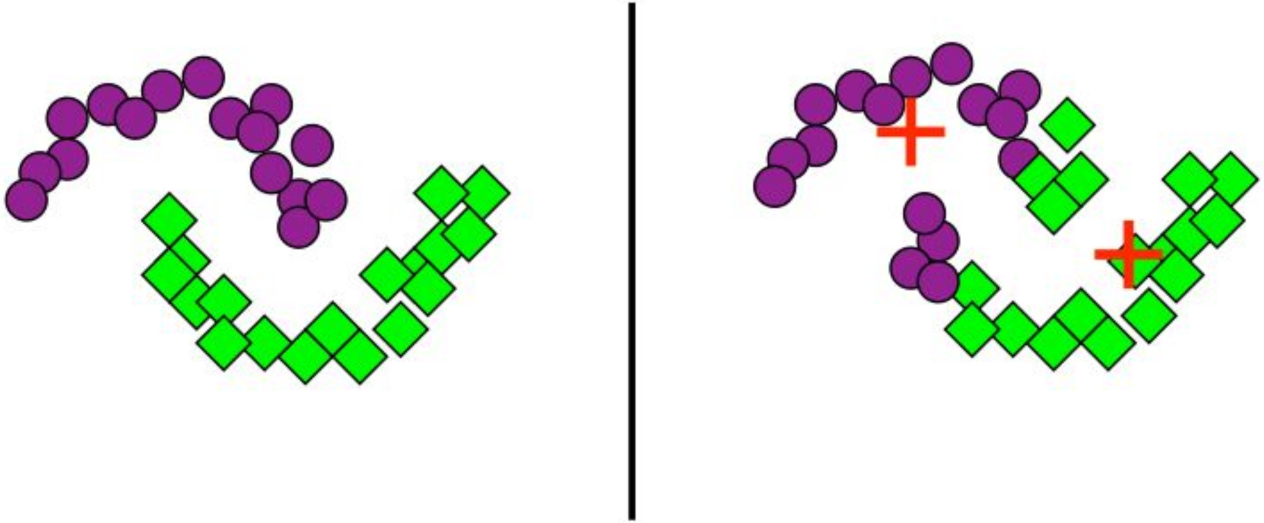
Discussion:

When would k-means fail?
When would it succeed?

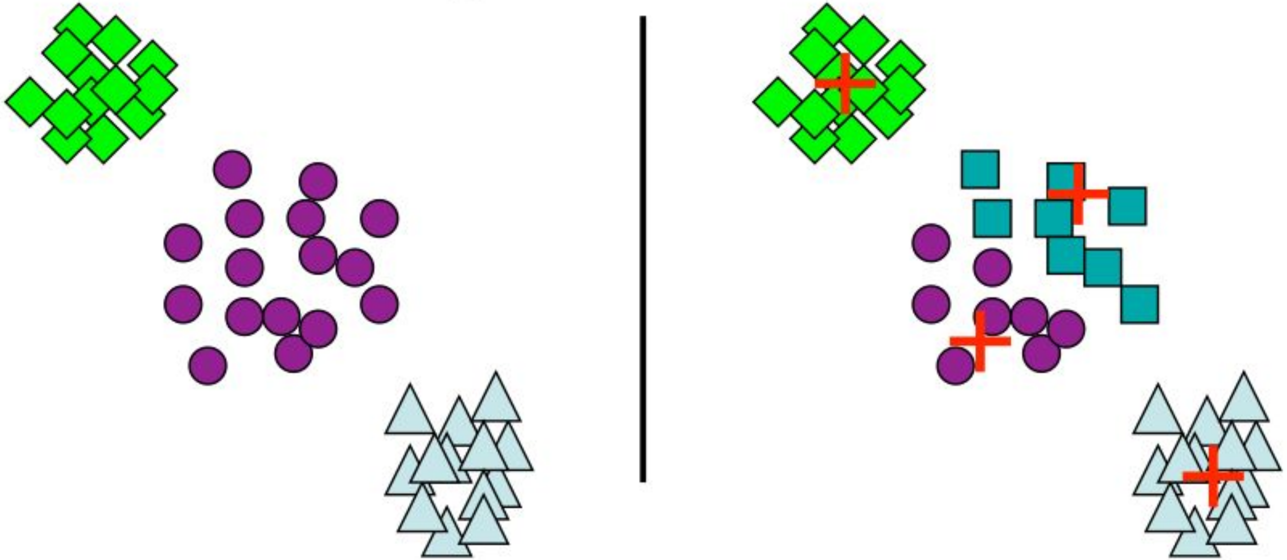
What are some drawbacks?
What are some advantages?

Weaknesses:

Non-globular clusters



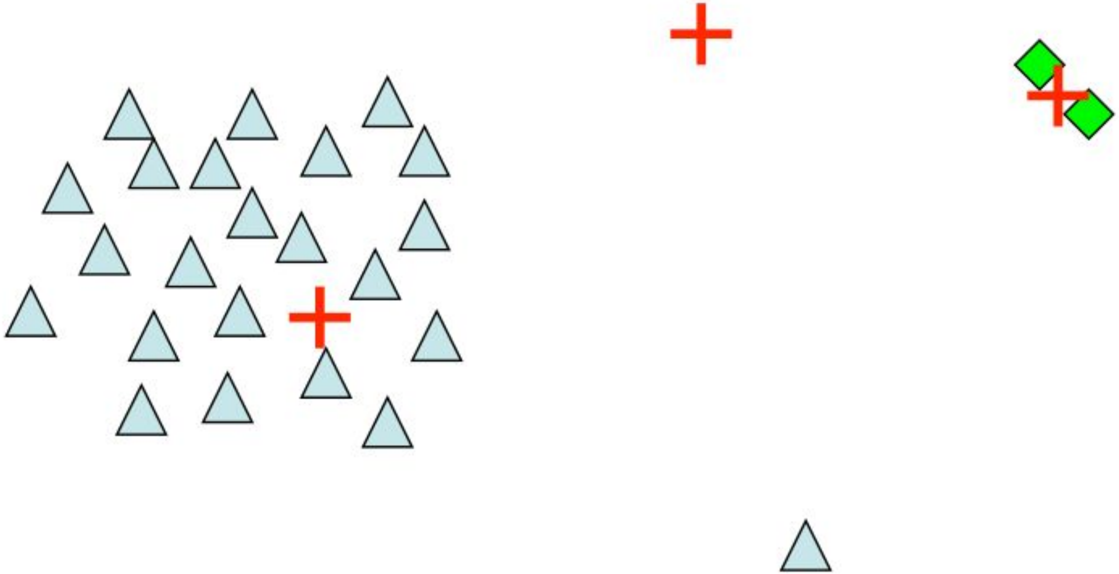
Weaknesses:



Wrong number of clusters

Weaknesses:

Outliers and empty clusters



Algorithm design: Distance metrics and features

- How do we determine what elements are “close”?
  - Spatial distance, similarity of campsite attributes,...
- “Features” are attributes that we use to mathematically compute closeness.
  - Height, age, hometown
- Weights for features
  - Are all features equally important or are some more important than others?
  - Are the features on different scales?

Types of features

- **Categorical features** take a fixed set of values. The values cannot be ordered.
    - For example, the type of phone (android, iphone, windows phone).

- **Ordinal features** take a fixed set of values. The values can be ordered.
    - For example, the ranks of football teams in the PAC-12 conference.

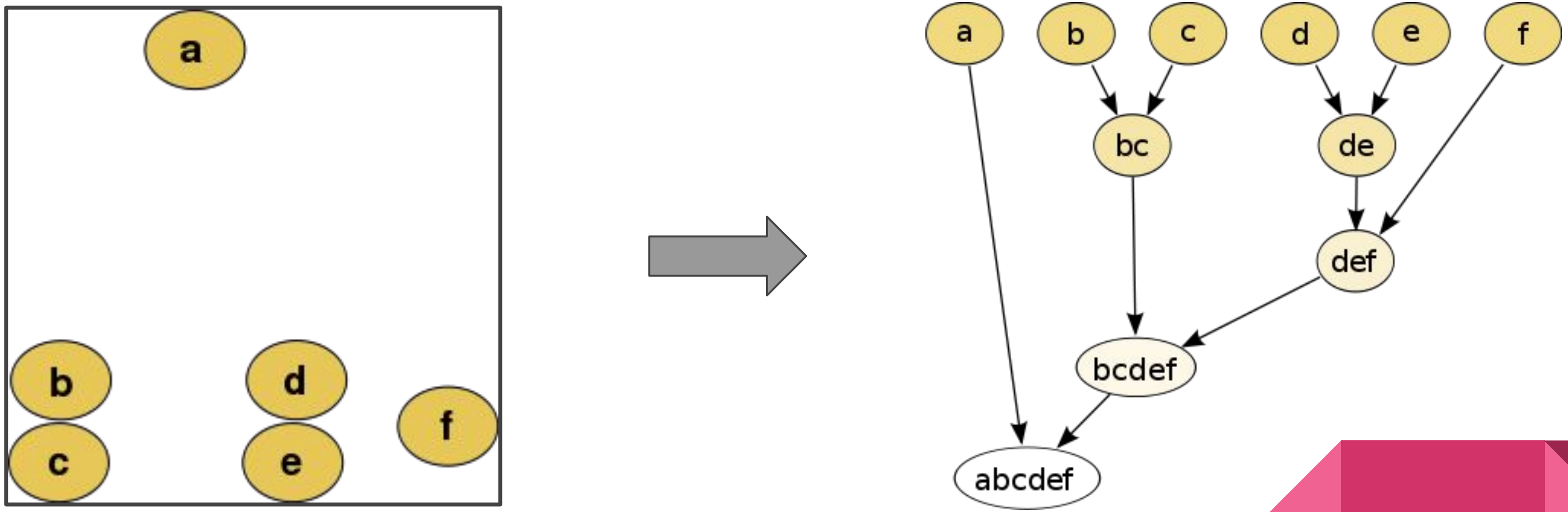
- **Continuous features** can take any real value.
    - For example, the distance of Princeton University from your hometown.

Main types of clustering algorithms

- Connectivity-based (hierarchical) clustering
- Centroid-based clustering
- Distribution-based clustering
- Density-based clustering

Connectivity-based (hierarchical) clustering

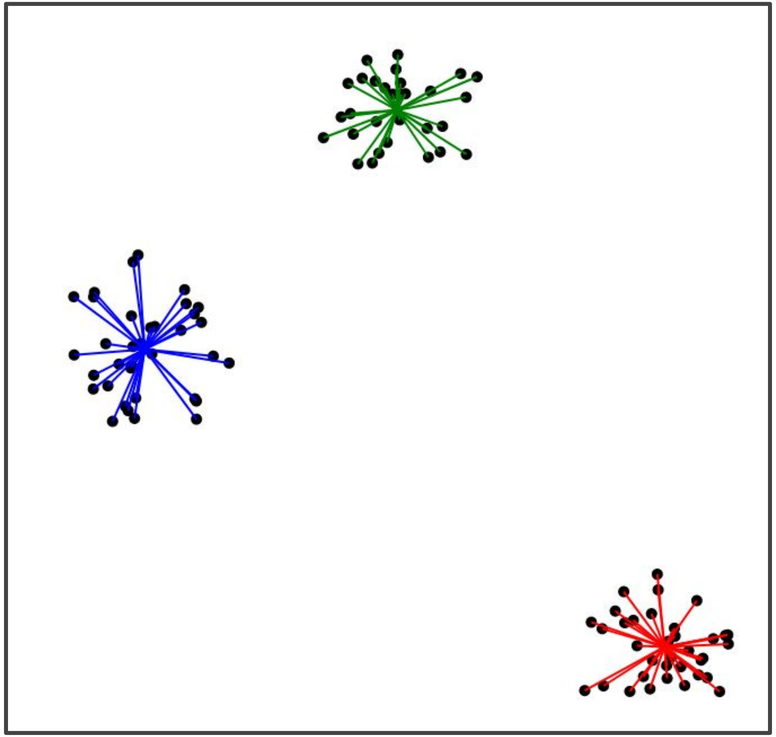
- A hierarchy of clusters based on distance:



- Typically, form a “dendogram”

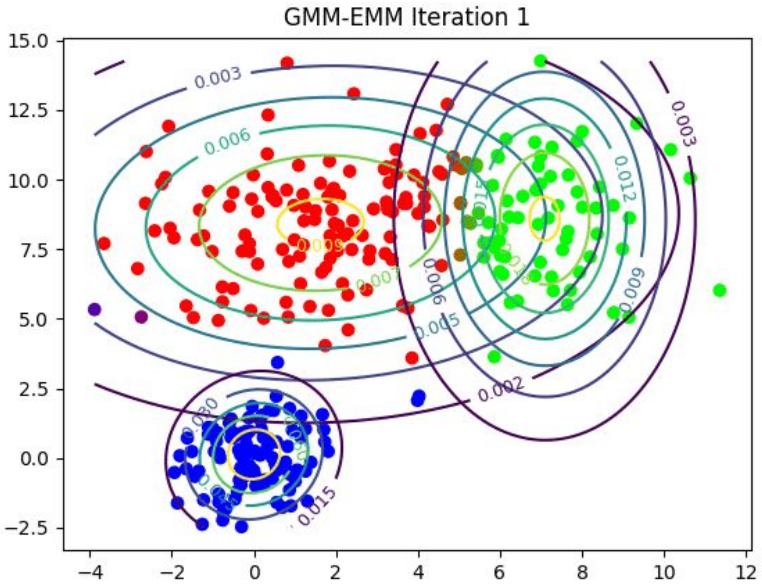
Centroid-based clustering

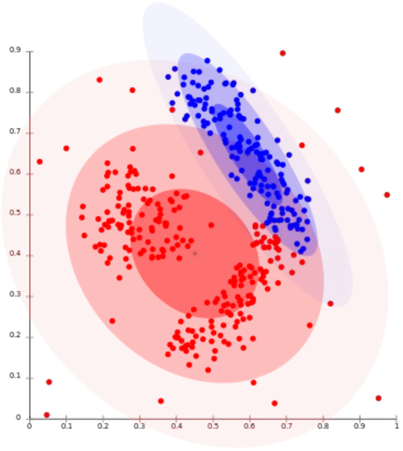
- k-means!



Distribution-based clustering

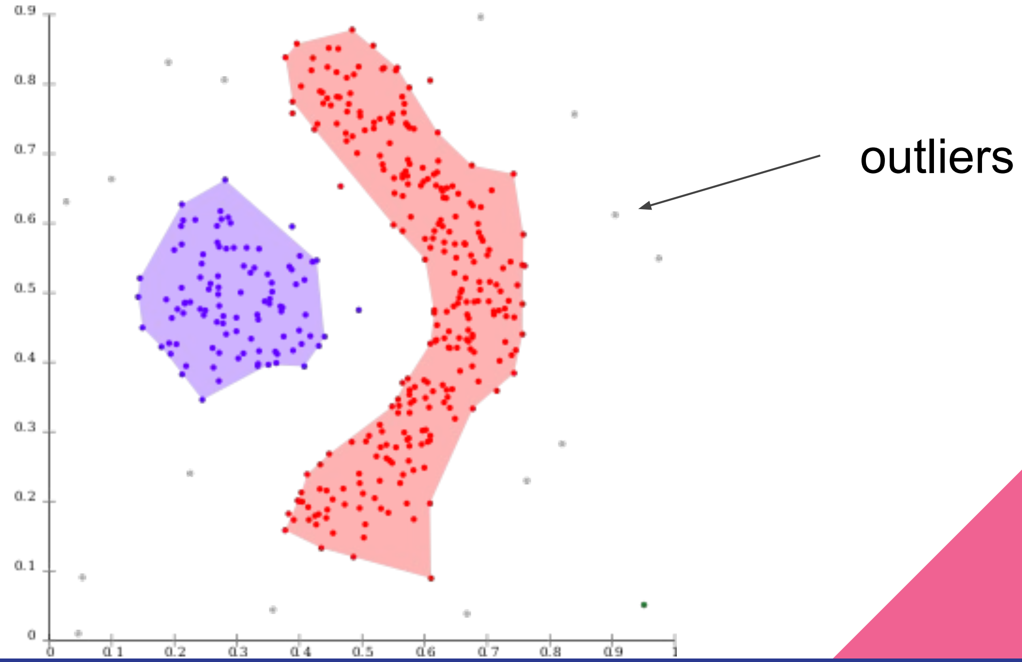
- Cluster according to how likely it is that points lie in a certain distribution





Density-based clustering

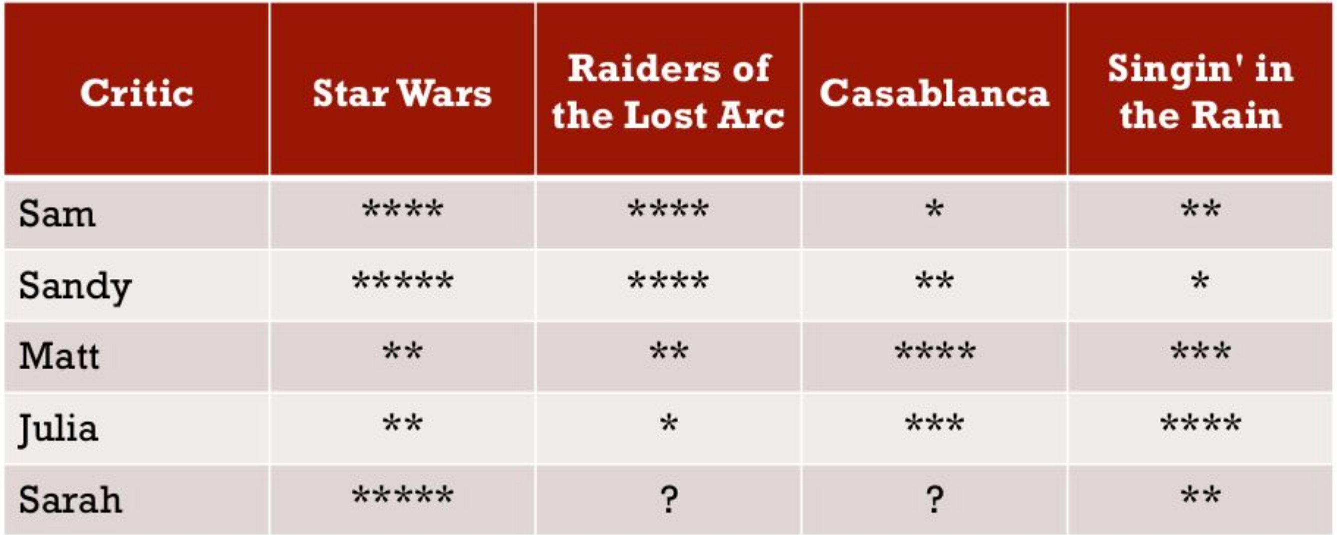
- Group together points that are closely spaced, and marks farther-out points as outliers



outliers

Recommendation system example

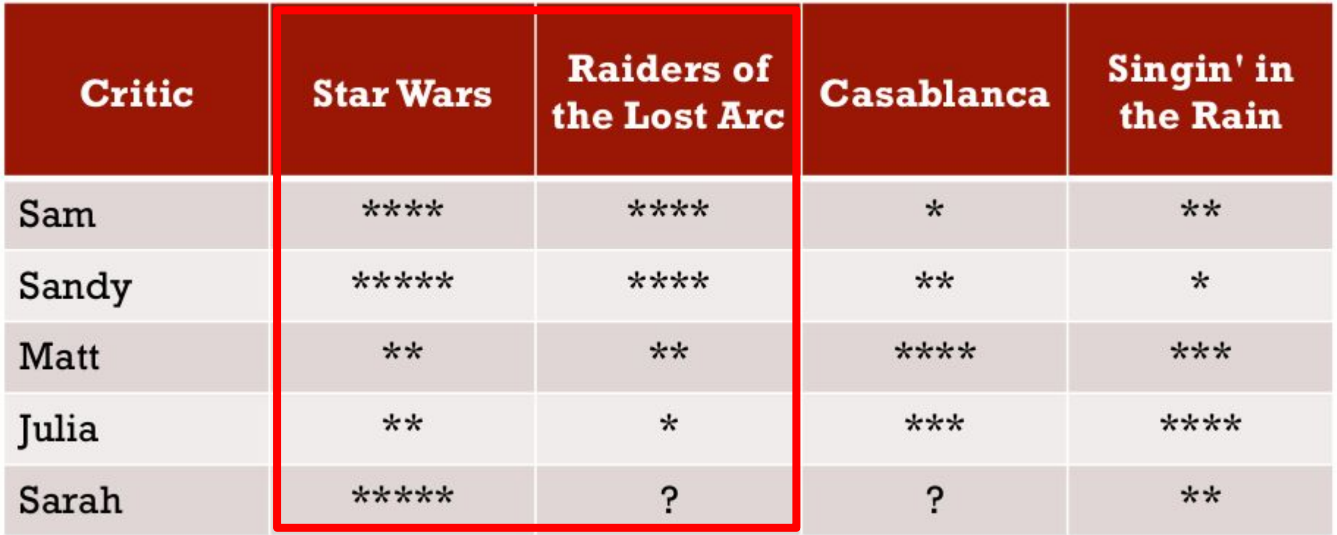
Dataset: Movie critics



Critic	Star Wars	Raiders of the Lost Arc	Casablanca	Singin' in the Rain
Sam	****	****	*	**
Sandy	*****	****	**	*
Matt	**	**	****	***
Julia	**	*	***	****
Sarah	*****	?	?	**

- Could an algorithm use this data to recommend movies?
- How would you do it?

Dataset: Movie critics

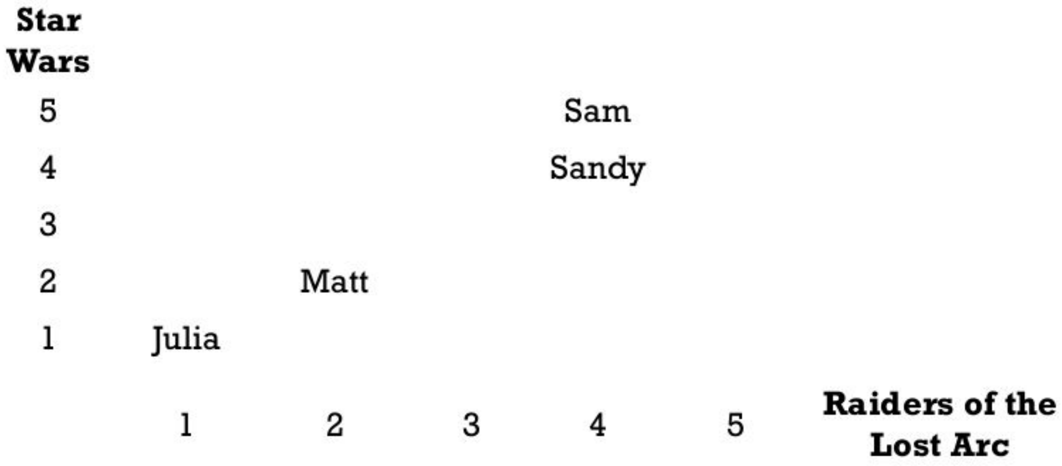


Critic	Star Wars	Raiders of the Lost Arc	Casablanca	Singin' in the Rain
Sam	****	****	*	**
Sandy	*****	****	**	*
Matt	**	**	****	***
Julia	**	*	***	****
Sarah	*****	?	?	**

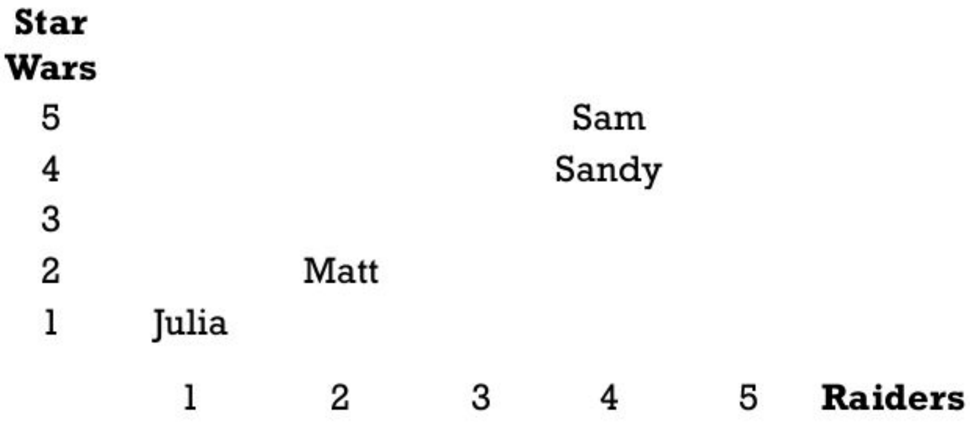
- Could an algorithm use this data to recommend movies?
- How would you do it?

Critics with similar taste

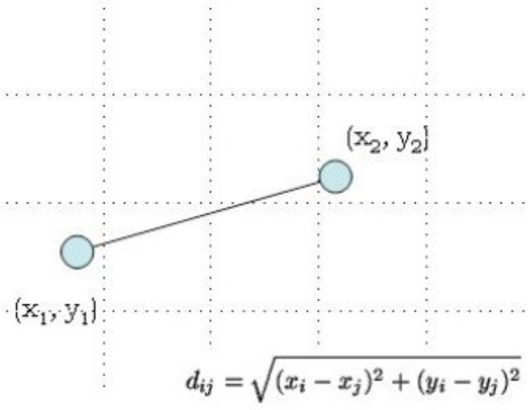
- Preference space



Measuring distance



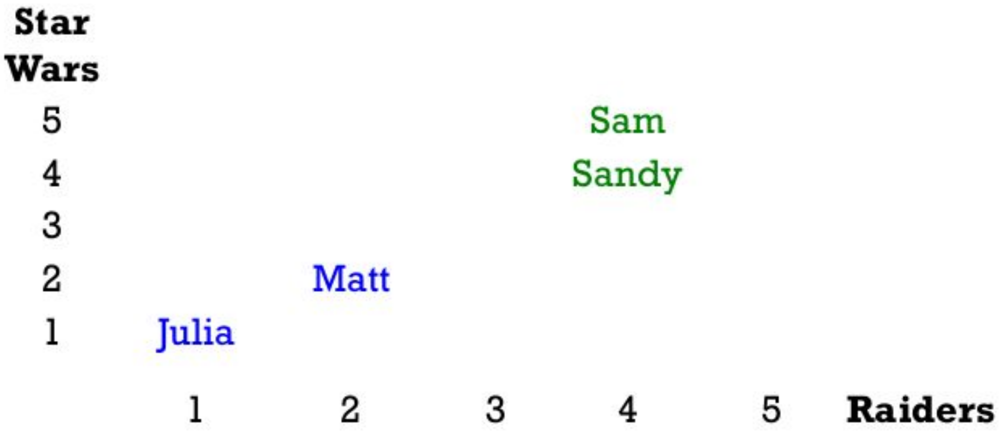
Star Wars						
5				Sam		
4				Sandy		
3						
2		Matt				
1	Julia					
	1	2	3	4	5	Raiders

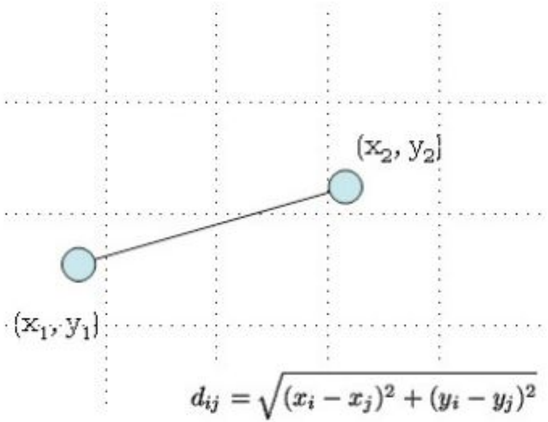


$$d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$$

- Measure similarity between points using a measure of distance

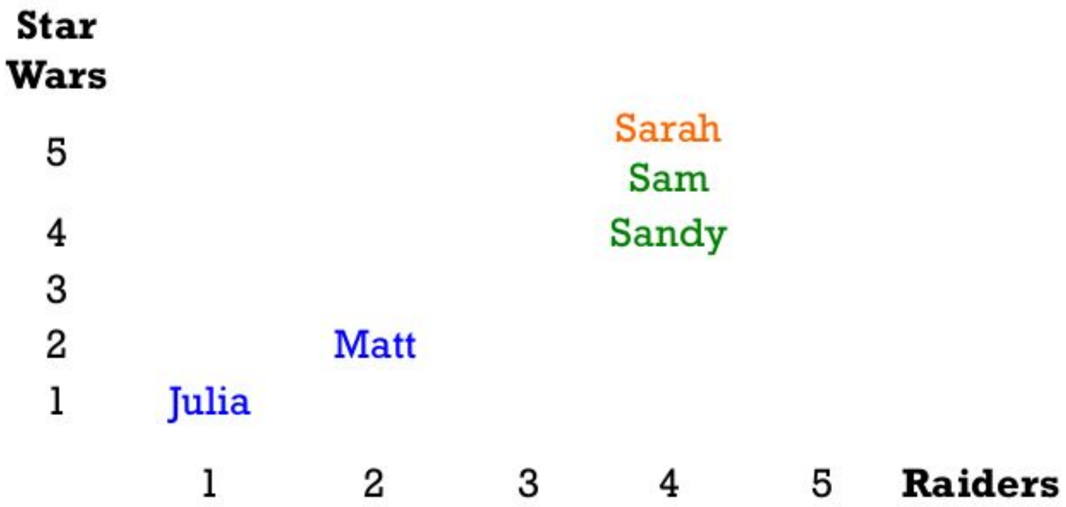
Finding critics with similar taste

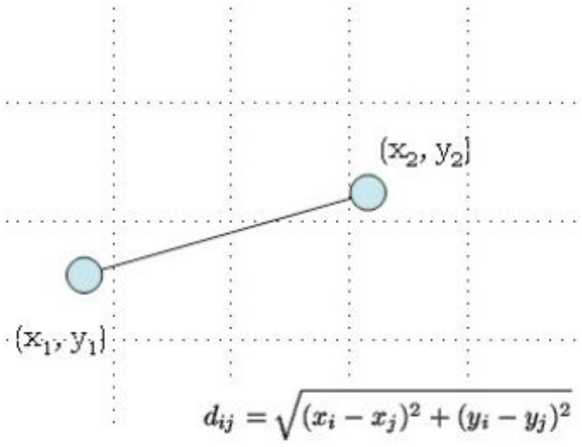




- People who liked Star Wars are close in preference space to those who liked Raiders

Making a recommendation

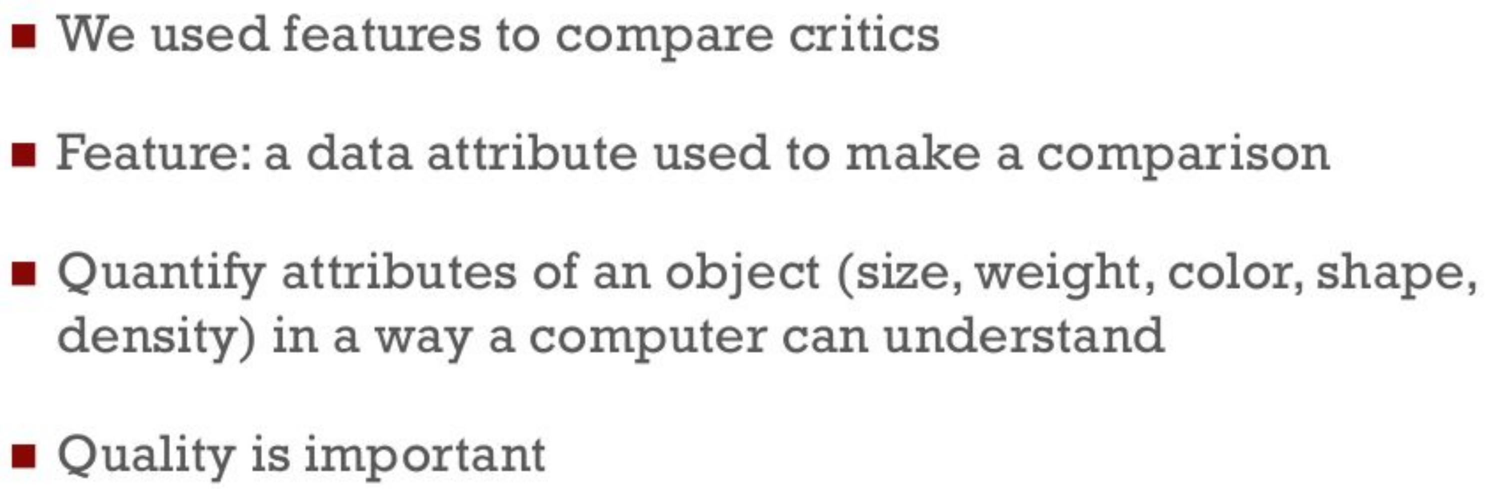




$$d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$$

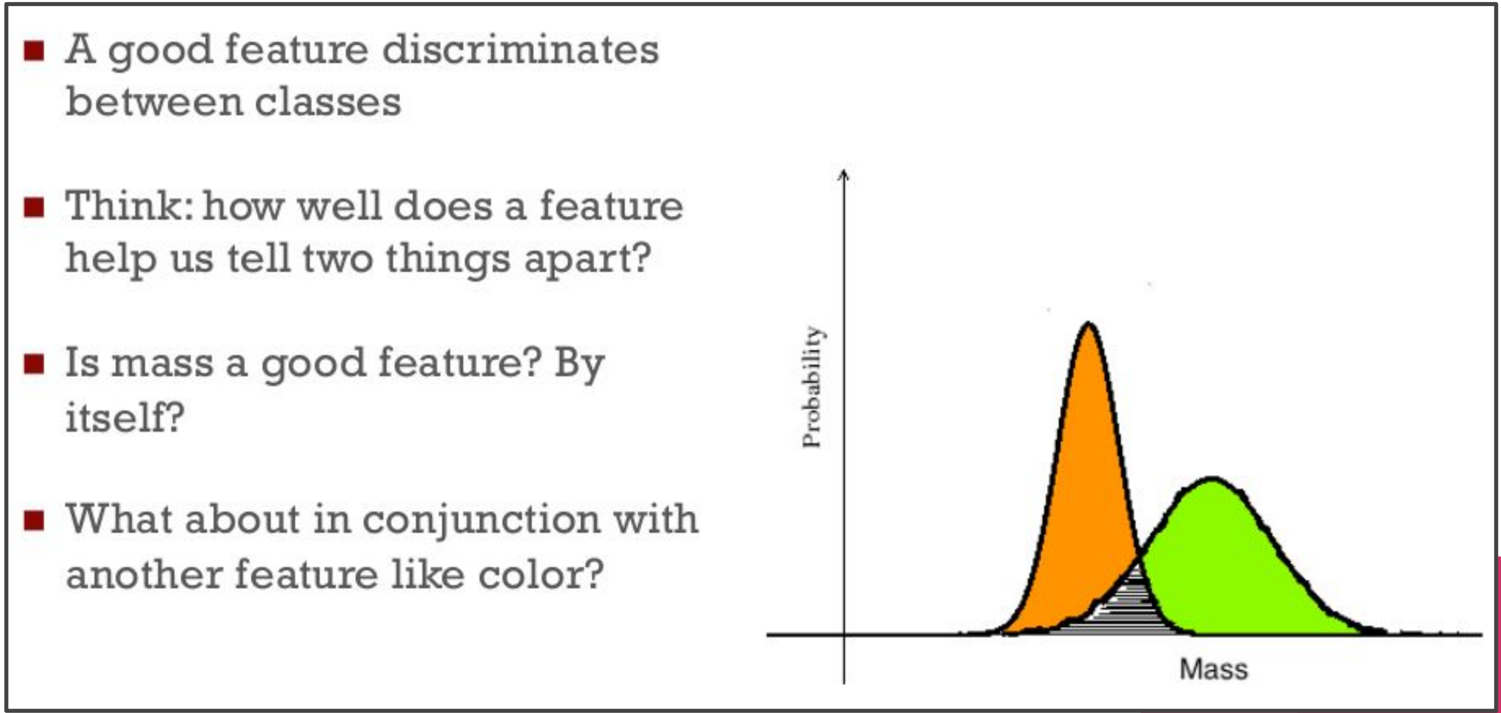
- Sarah hasn’t seen Raiders, but gave Star Wars five stars.
- Chances are she’ll like Raiders too!

Features



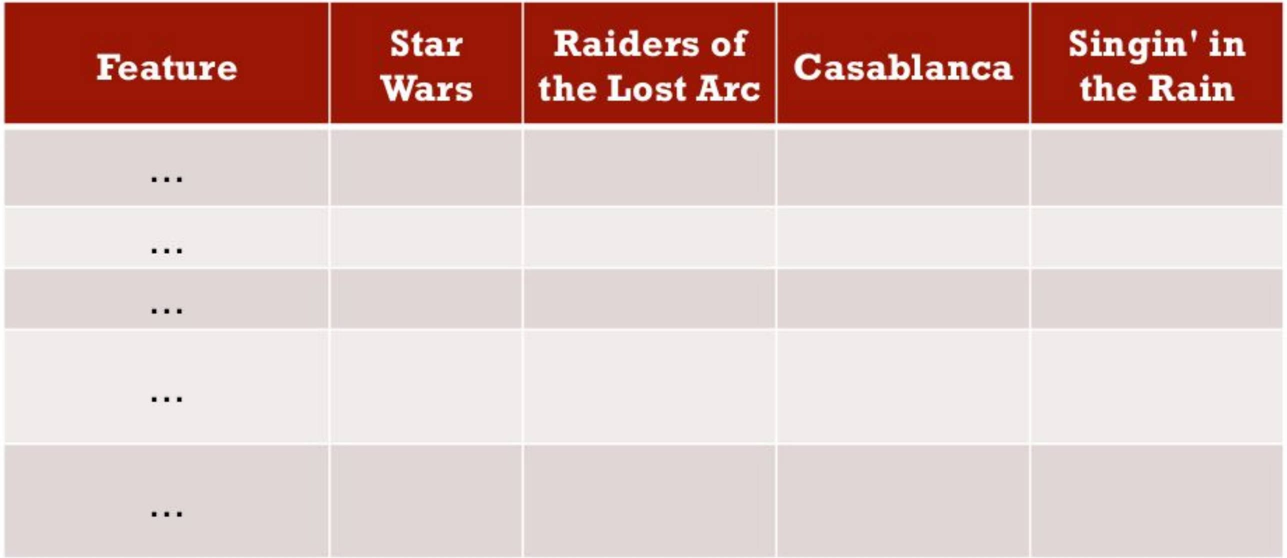
- We used features to compare critics
- Feature: a data attribute used to make a comparison
- Quantify attributes of an object (size, weight, color, shape, density) in a way a computer can understand
- Quality is important

Apples vs. Oranges



- A good feature discriminates between classes
- Think: how well does a feature help us tell two things apart?
- Is mass a good feature? By itself?
- What about in conjunction with another feature like color?

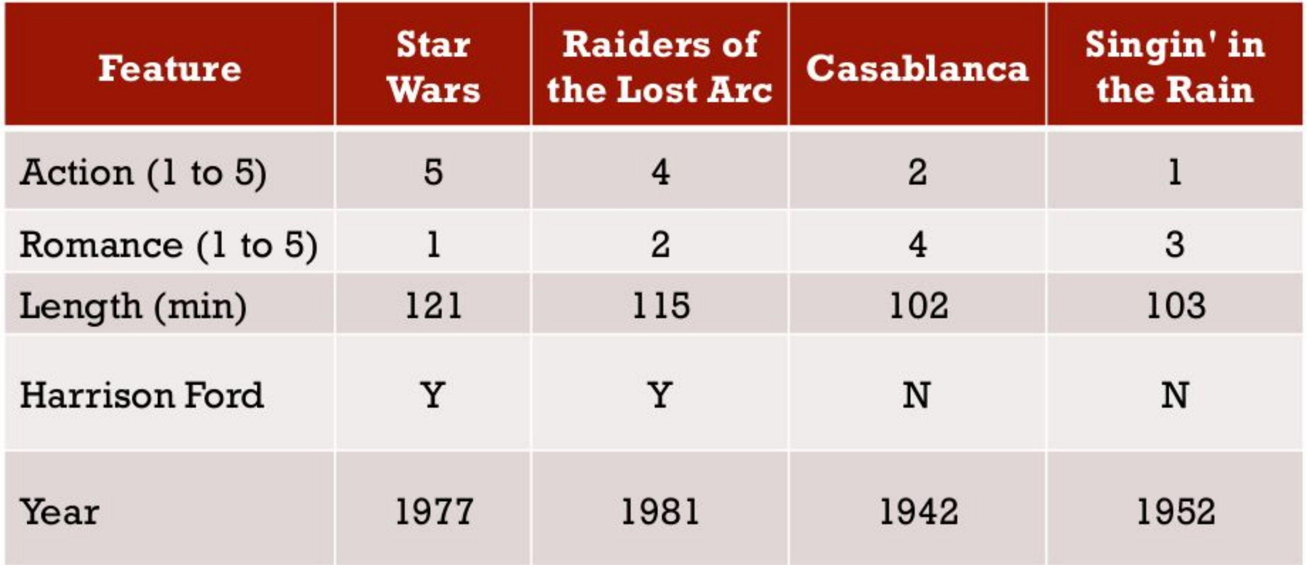
Features to compare movies



Feature	Star Wars	Raiders of the Lost Arc	Casablanca	Singin' in the Rain
...				
...				
...				
...				
...				

- Can you suggest some features to compare movies?

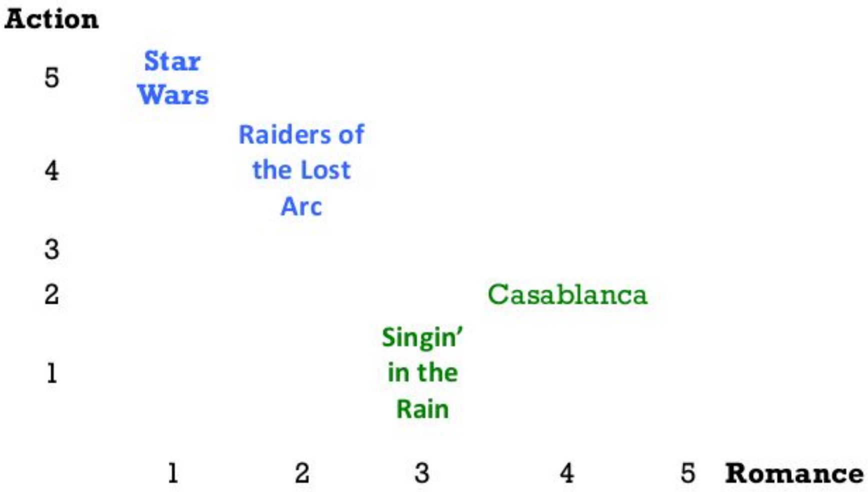
Features to compare movies



Feature	Star Wars	Raiders of the Lost Arc	Casablanca	Singin' in the Rain
Action (1 to 5)	5	4	2	1
Romance (1 to 5)	1	2	4	3
Length (min)	121	115	102	103
Harrison Ford	Y	Y	N	N
Year	1977	1981	1942	1952

- What type of features are these?

Comparing movies in feature space



- Here we can “eyeball” the clusters, but what if we had more features? (i.e. higher dimensional data)?

Exercise time:

Jupyter notebook available at https://github.com/juliennelachance/ai4all_clustering