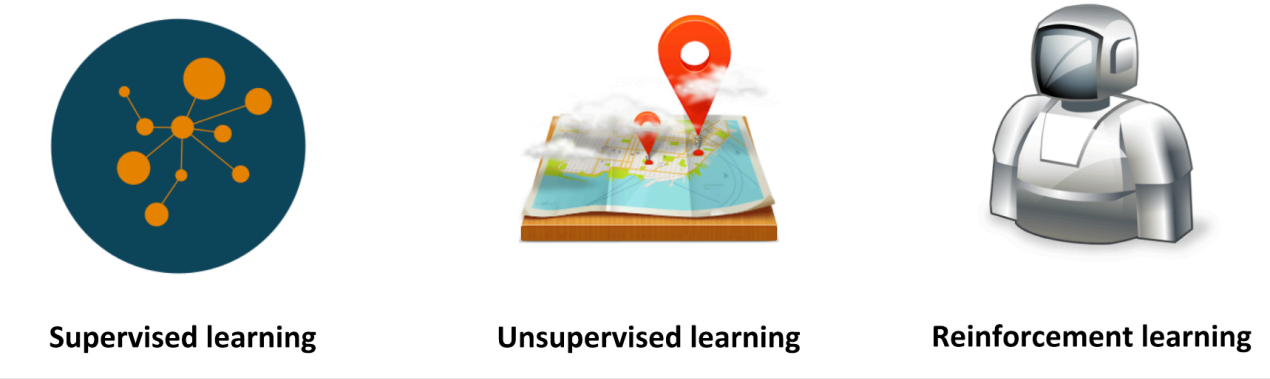
K methods

Week2

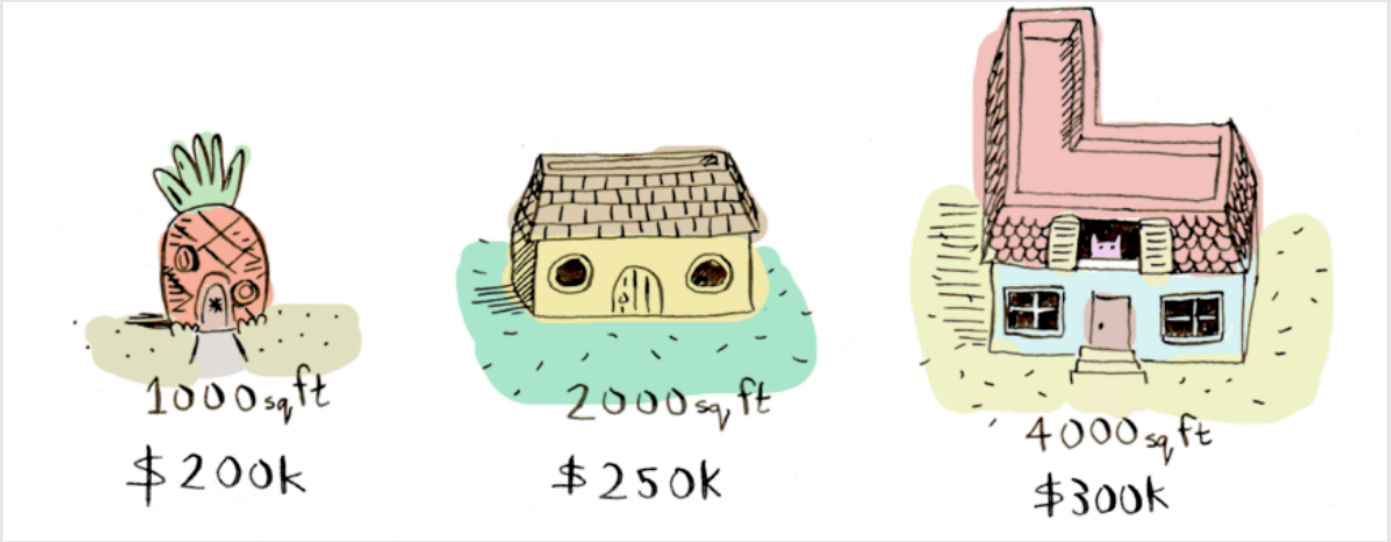
Machine Learning



Supervised learning: given labeled data try to learn a mapping function y← $f(x)$
Unsupervised learning: given unlabeled data try to learn the inherent structure of x
Reinforcement learning: learning based on interaction with environment

Regression

Regression



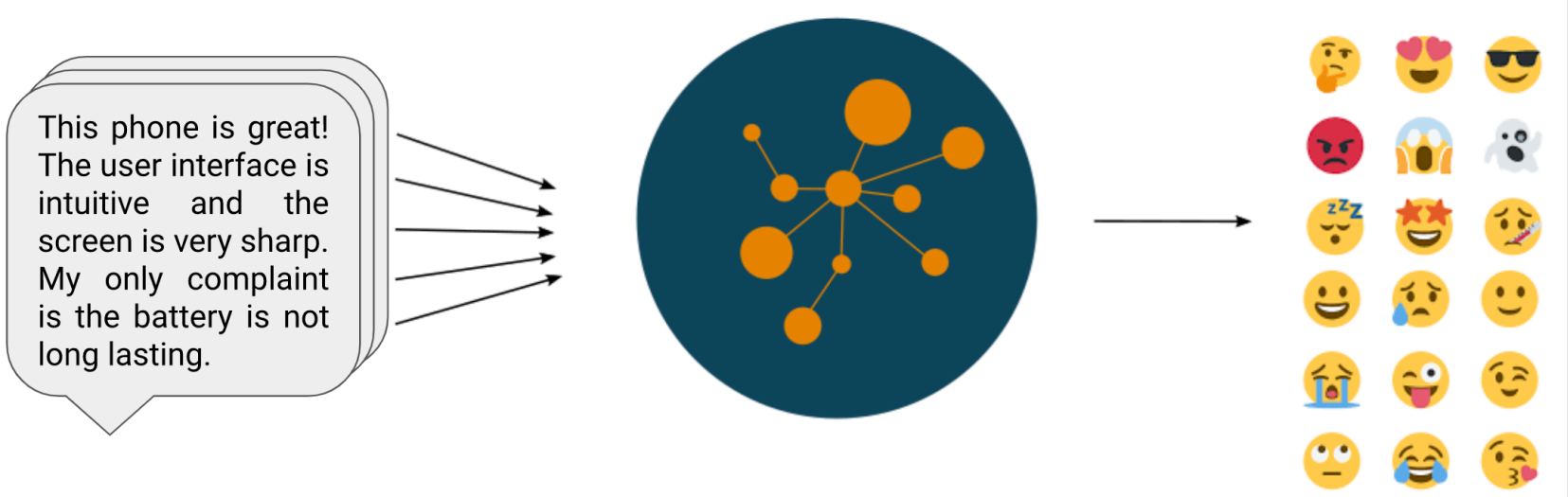
Auction: how much to bid?

Image of an invoice: total amount in dollars?

Classification

Discrete output

Classification



Classification

Dog: 96%

Cat: 29%

Duck: 2%

Bird: 0%

Dog: 36%

Cat: 94%

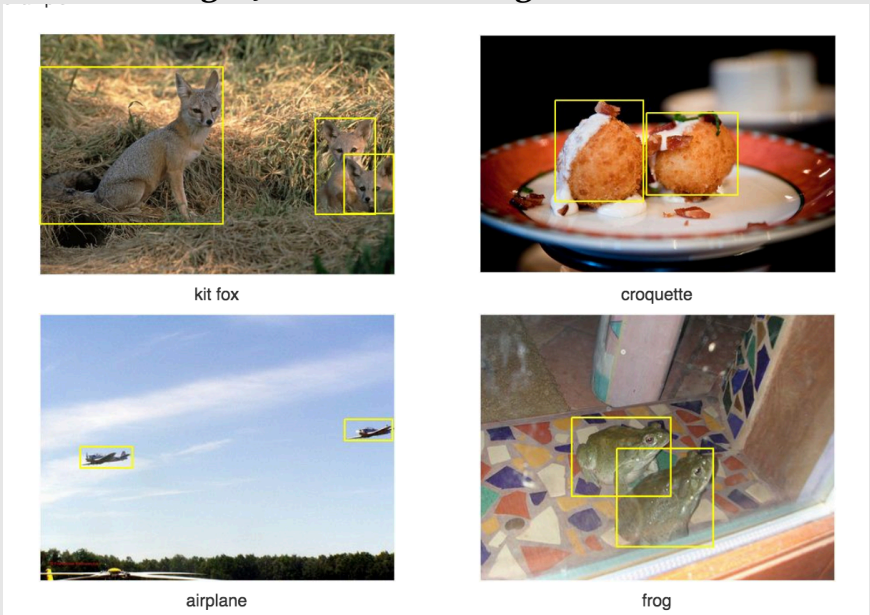
Duck: 2%

Bird: 1%

Supervised Learning in Computer Vision

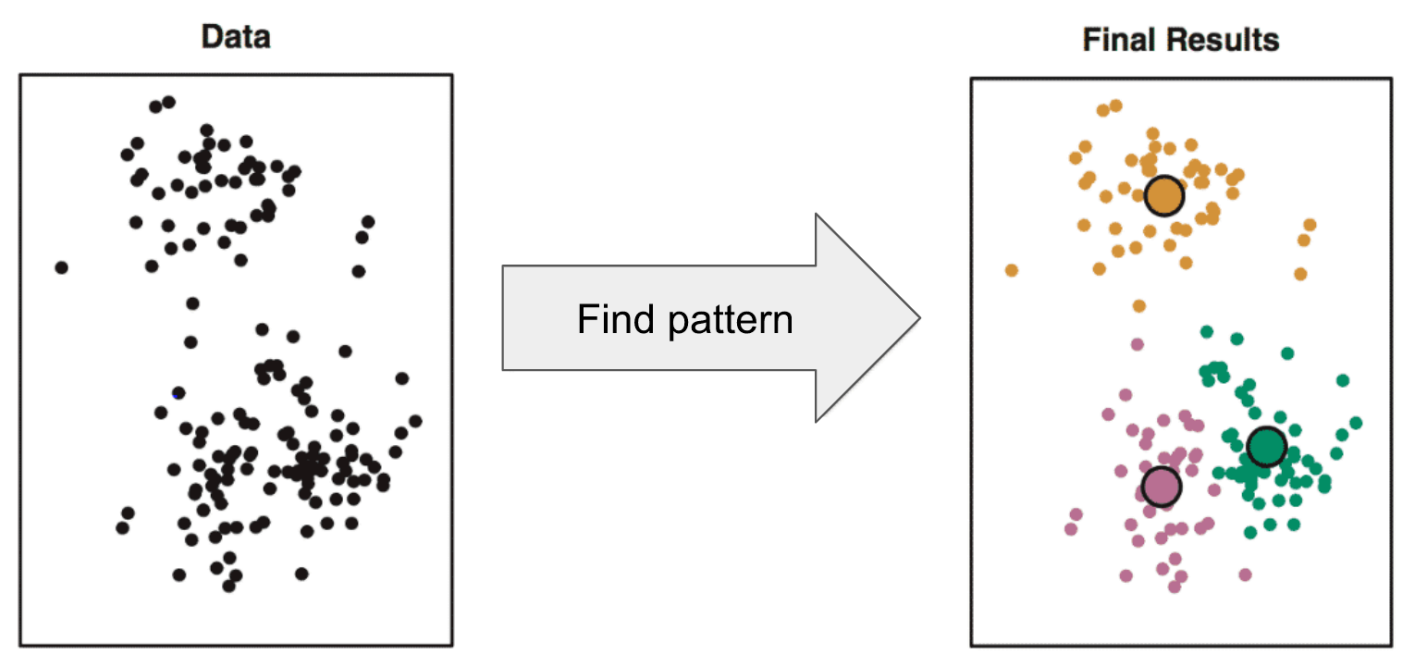
Object localization and detection (both classification and regression)

x = raw pixels of the image, y = the bounding boxes



ImageNet Large Scale Visual Recognition Challenge

Unsupervised Learning: Clustering

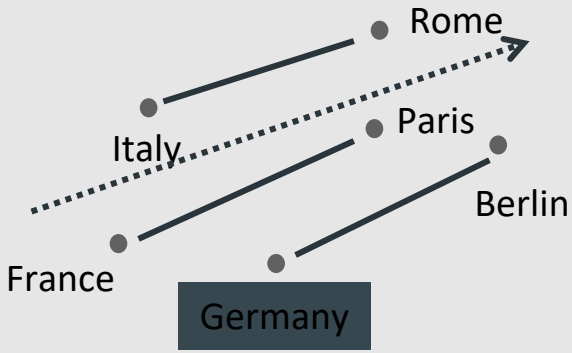


Word Embeddings

Unlabeled dataset

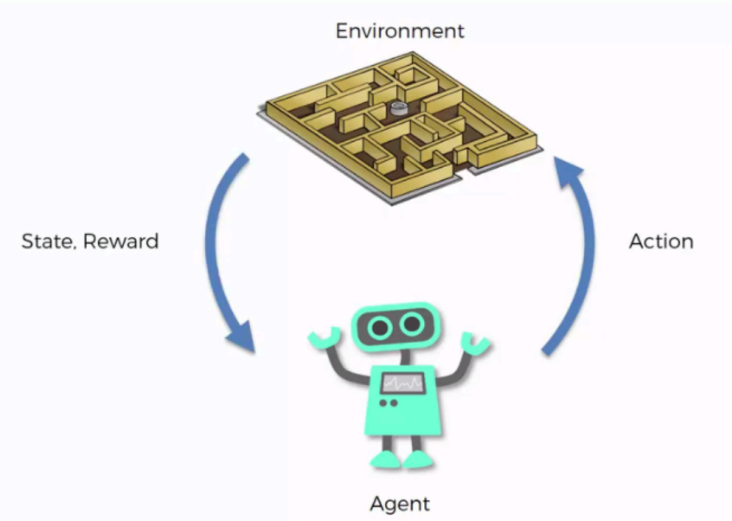
Represent words by vectors

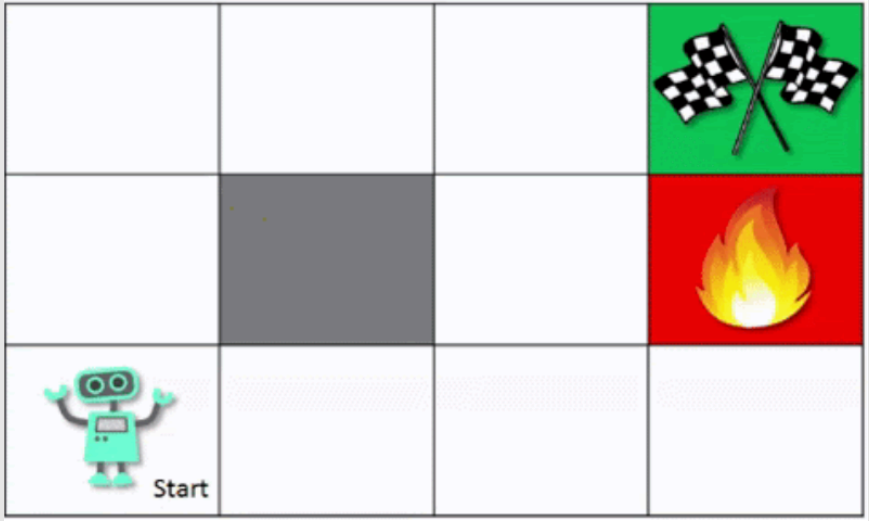
- word $\xrightarrow{\text{encode}}$ vector
- relation $\xrightarrow{\text{encode}}$ direction



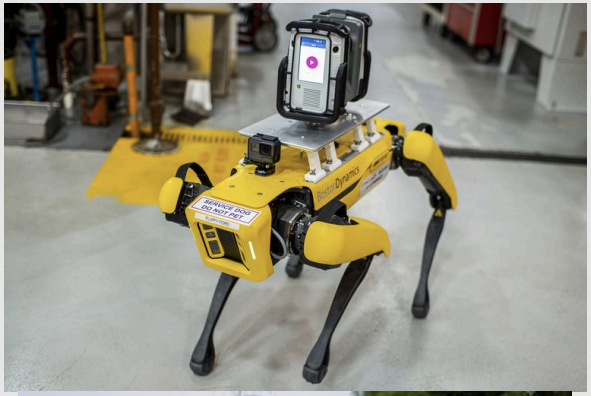
Word2vec [Mikolov et al'13]
GloVe [Pennington et al'14]

Reinforcement Learning

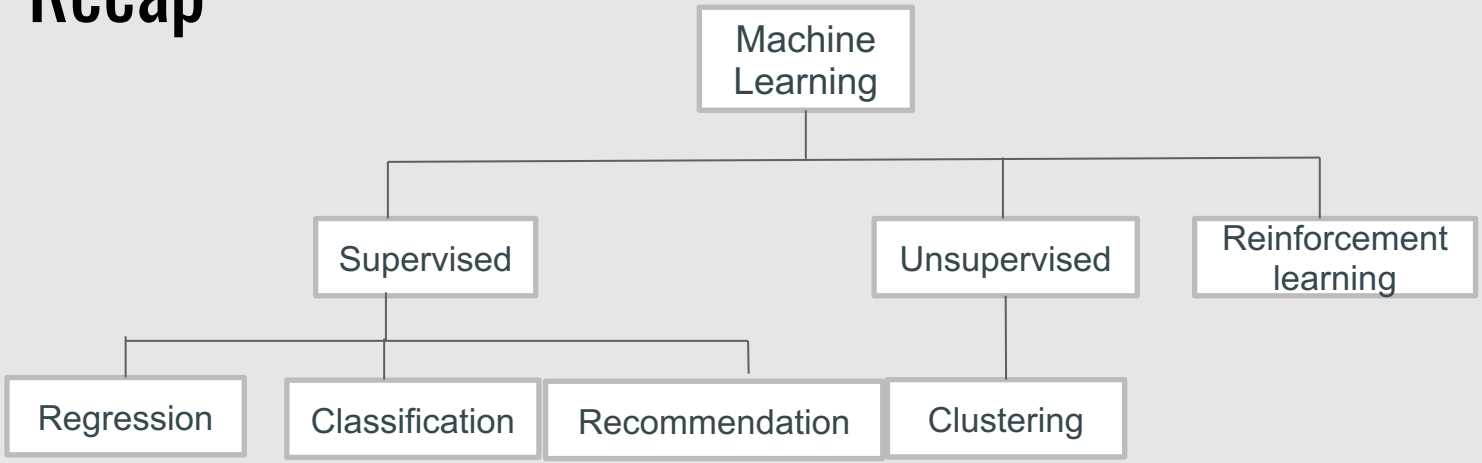




Reinforcement Learning



Recap



Regression: How much?
Classification: which?
Recommendation: recommend me some X based on my previous X's
Clustering: find patterns/groups
Reinforcement learning: what action to take next in an environment

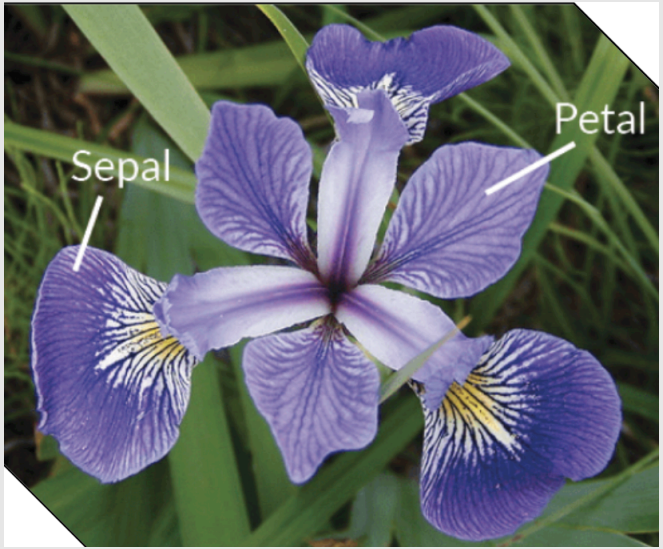
Agenda

- Feature space
- K-means clustering
- K-NN classification
- K-NN regression

Datasets are usually just tables (Structured Data)

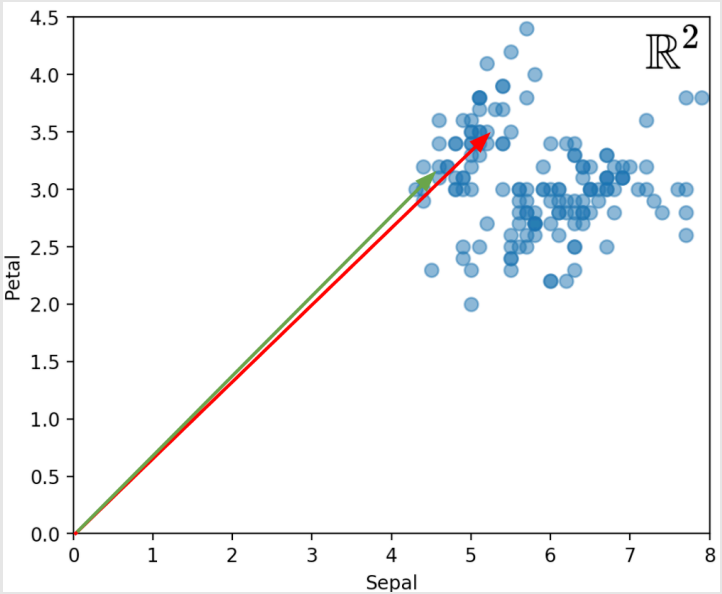
Usually we can think of datasets as tables/excel sheets:

	Sepal	Petal
0	5.1	3.5
1	4.9	3.0
2	4.7	3.2
3	4.6	3.1
4	5.0	3.6



A table as a vector space

	Sepal	Petal
0	5.1	3.5
1	4.9	3.0
2	4.7	3.2
3	4.6	3.1
4	5.0	3.6



Each row/observation is a vector in the **feature vector space**

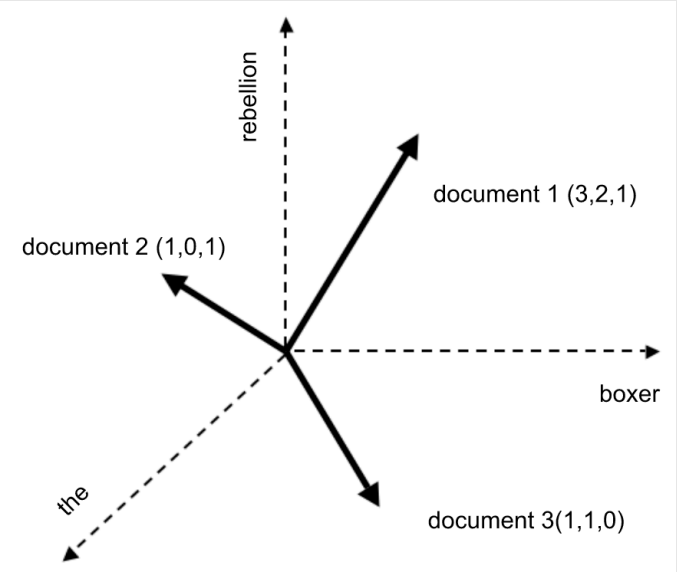
Example: document representation

	The	Boxer	Rebellion	...
Document 1	3	2	1	...
Document 2	1	0	1	...
Document 3	1	1	0	...

The bag of words representation

The Boxer Rebellion (拳亂), **Boxer** Uprising, or Yihetuan Movement (義和團運動) was an anti-imperialist, anti-foreign, and anti-Christian uprising that took place in China between 1899 and 1901, toward **the** end of **the** Qing dynasty.

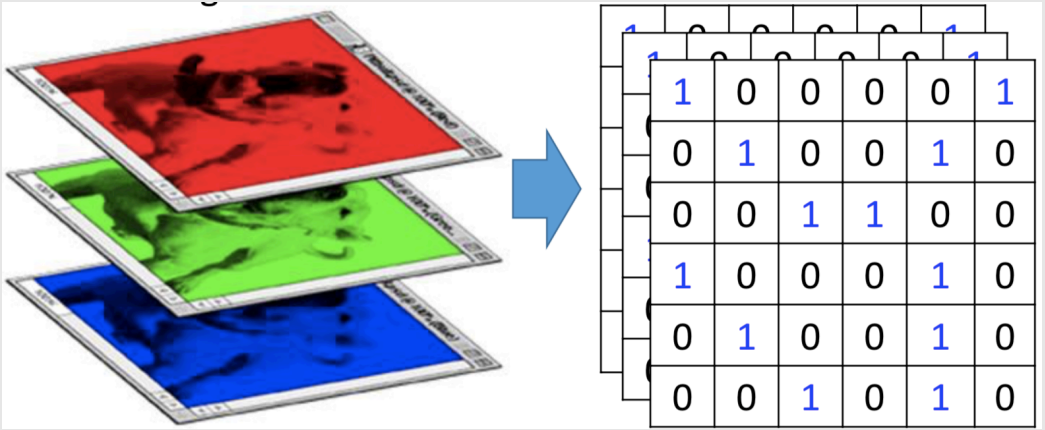
Example: document representation



The bag of words representation

The Boxer Rebellion (拳亂), **Boxer** Uprising, or Yihetuan Movement (義和團運動) was an anti-imperialist, anti-foreign, and anti-Christian uprising that took place in China between 1899 and 1901, toward **the** end of **the** Qing dynasty.

Example: Image representation

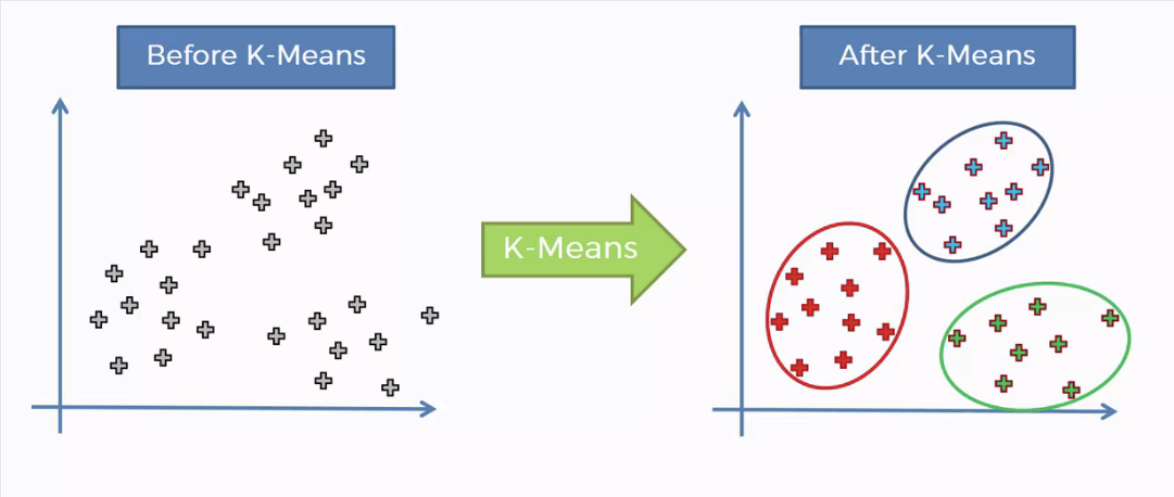


K means clustering

Clustering

- Put “similar” members to the same group
- How many groups do you see?

- We call each group a cluster!



Distance Metrics

- We're in a world of "feature vectors" which can map to co-ordinates

The Minkowski distance of order *p* between two points

$$X = (x_1, x_2, \ldots, x_n) \text{ and } Y = (y_1, y_2, \ldots, y_n) \in \mathbb{R}^n$$

$$D(X, Y) = \left(\sum_{i=1}^{n} |x_i - y_i|^p \right)^{1/p}$$

- The generalized **Minkowski Distance** can often be a useful measure
  - In 2-dimensions, it is the same as the Euclidean Distance when p is 2
  - When p is 1, it's Manhattan distance

Clustering

The goal:

- “Similar” members go into the same group

Key metrics:

- Intra-cluster variance (spread within a group)
- Inter-cluster variance (spread across groups)

Common objective:

- Maximize inter-cluster variance
- Minimize intra-cluster variance

Clustering

Given n data points $x_1, \dots, x_n$, where x_i is a d-dimensional vector. K-Means clustering aims to partition the set of n data points to k sets $\{S_1, \dots, S_k\}$

Common objective:

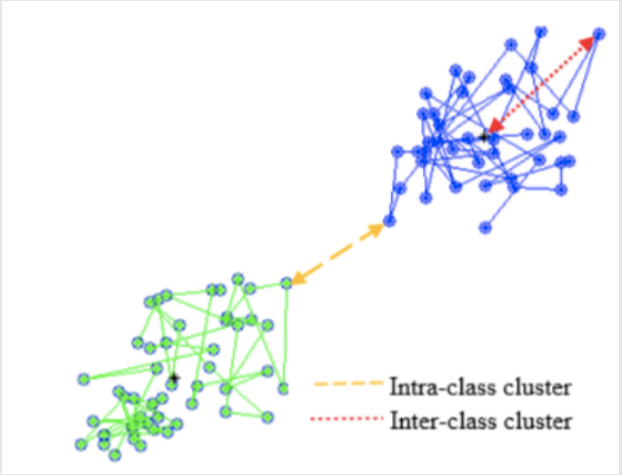
- Minimize intra-cluster variance

$$\sum_{i=1}^{k} \sum_{x \in S_i} \|x - c_i\|^2 = \sum_{i=1}^{k} |S_i| \, Var S_i$$

- Maximize inter-cluster variance

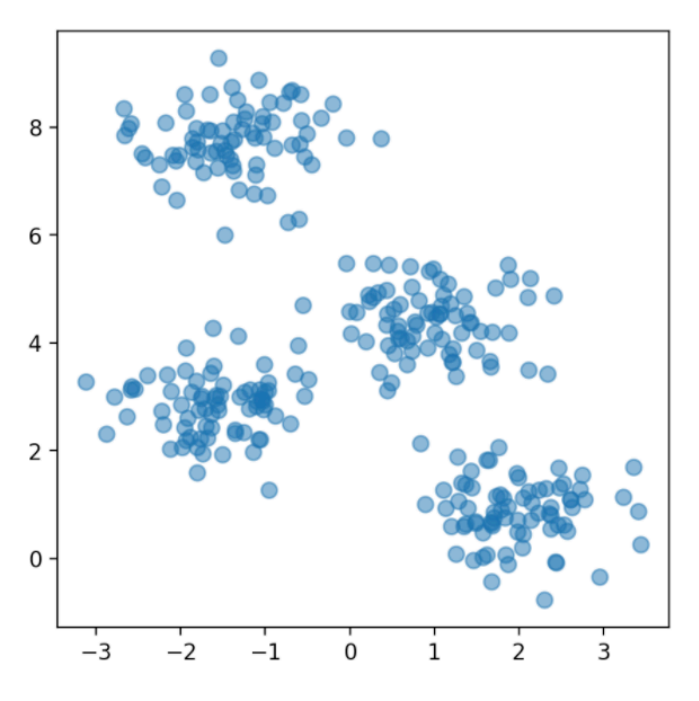
$$\sum_{i=1}^{k} |S_i| \cdot \|c_i - \bar{x}\|^2$$

$c_i = \frac{1}{|S_i|} \sum_{x \in S_i} x_i$ is the centroid for S_i, $\bar{x}$ is the mean of all n points



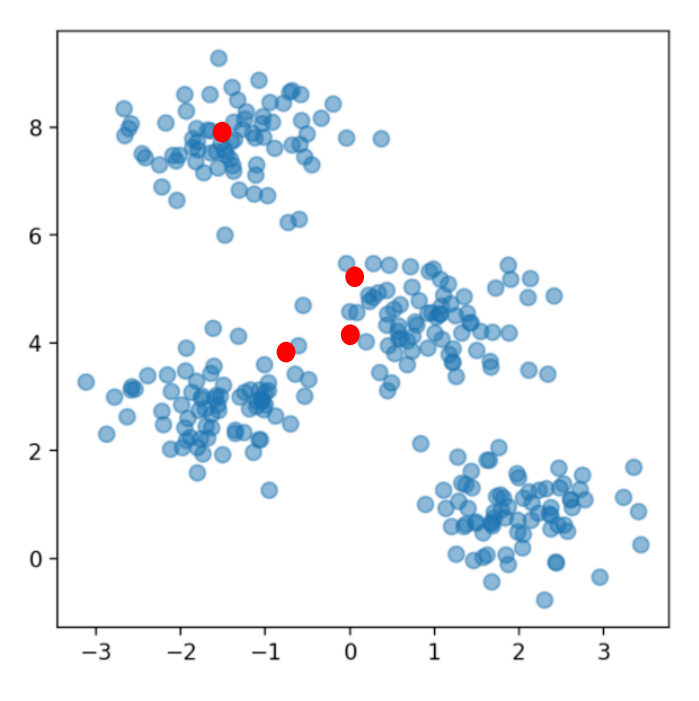
K-means clustering (Steinhaus 1950)

- Assume there are k groups
- Take k random points and assign them as centroids.
- Repeat until converge (i.e. no cluster moves its center).
    - Each member finds its closest cluster, add itself to the cluster
    - Each cluster computes its new centroid
    - If no cluster changes its centroid – done!



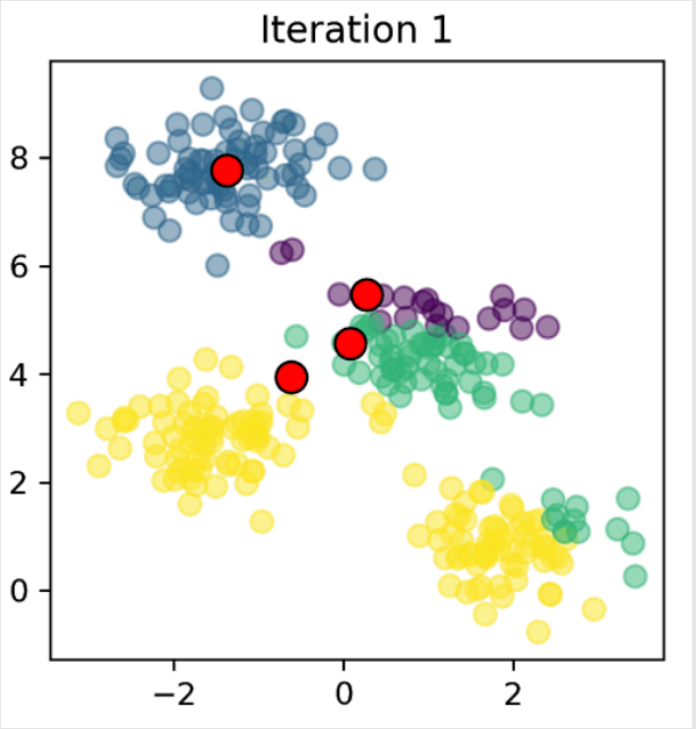
K-means clustering (Steinhaus 1950)

- Assume there are k groups
- Take k random points and assign them as centroids.
- Repeat until converge (i.e. no cluster moves its center).
    - Each member finds its closest cluster, add itself to the cluster
    - Each cluster computes its new centroid
    - If no cluster changes its centroid – done!



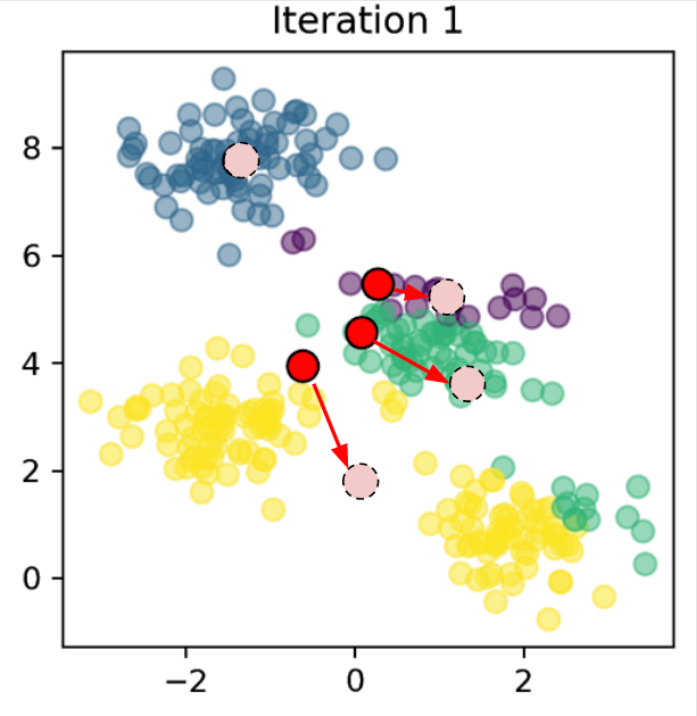
K-means clustering (Steinhaus 1950)

- Assume there are k groups
- Take k random points and assign them as centroids.
- Repeat until converge (i.e. no cluster moves its center).
    - Each member finds its closest cluster, add itself to the cluster
    - Each cluster computes its new centroid
    - If no cluster changes its centroid – done!



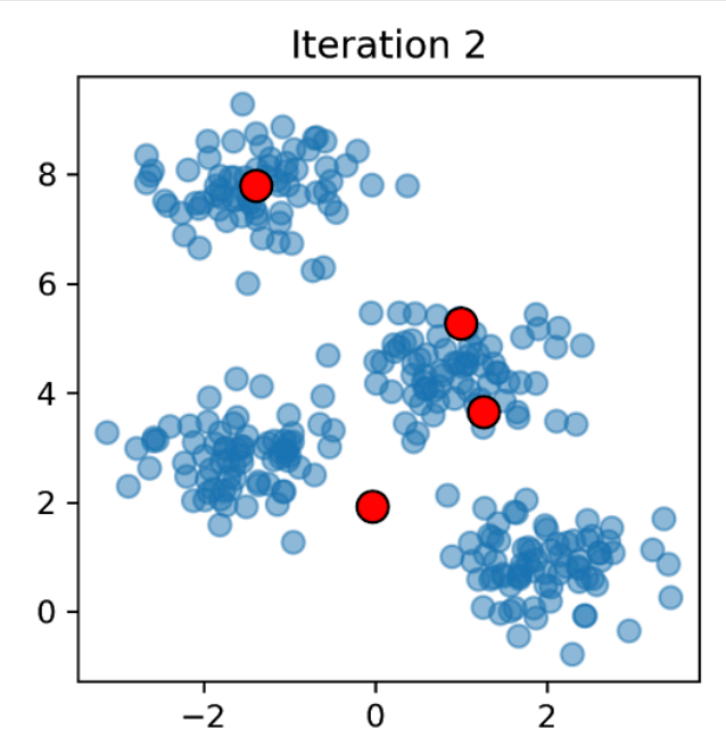
K-means clustering (Steinhaus 1950)

- Assume there are k groups
- Take k random points and assign them as centroids.
- Repeat until converge (i.e. no cluster moves its center).
    - Each member finds its closest cluster, add itself to the cluster
    - Each cluster computes its new centroid
    - If no cluster changes its centroid – done!



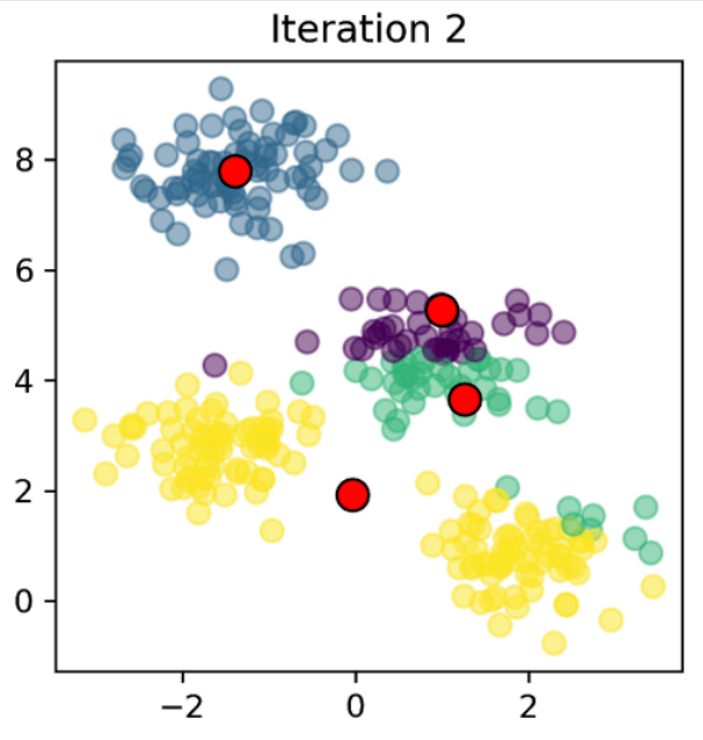
K-means clustering (Steinhaus 1950)

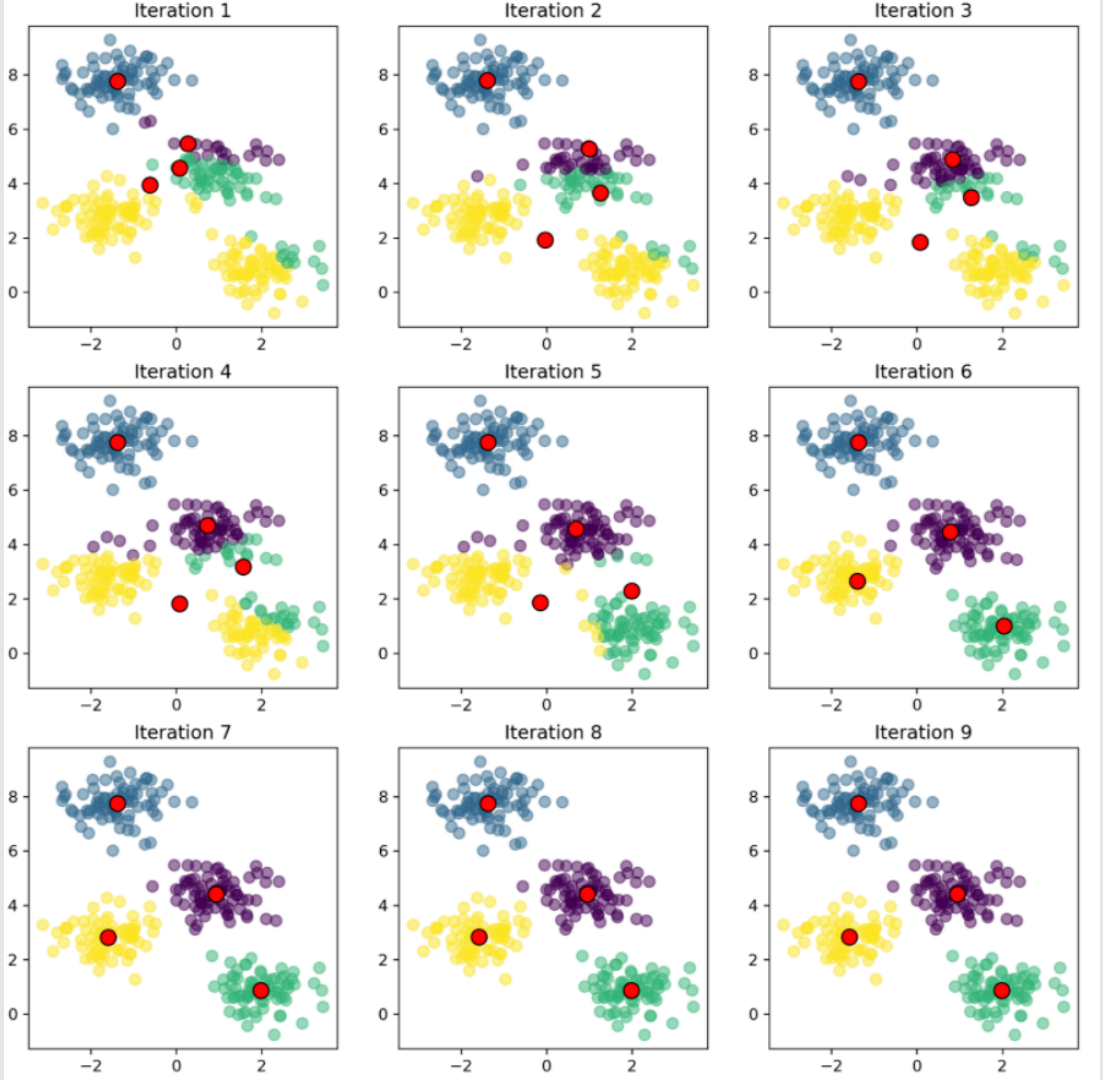
- Assume there are k groups
- Take k random points and assign them as centroids.
- Repeat until converge (i.e. no cluster moves its center).
    - Each member finds its closest cluster, add itself to the cluster
    - Each cluster computes its new centroid
    - If no cluster changes its centroid – done!



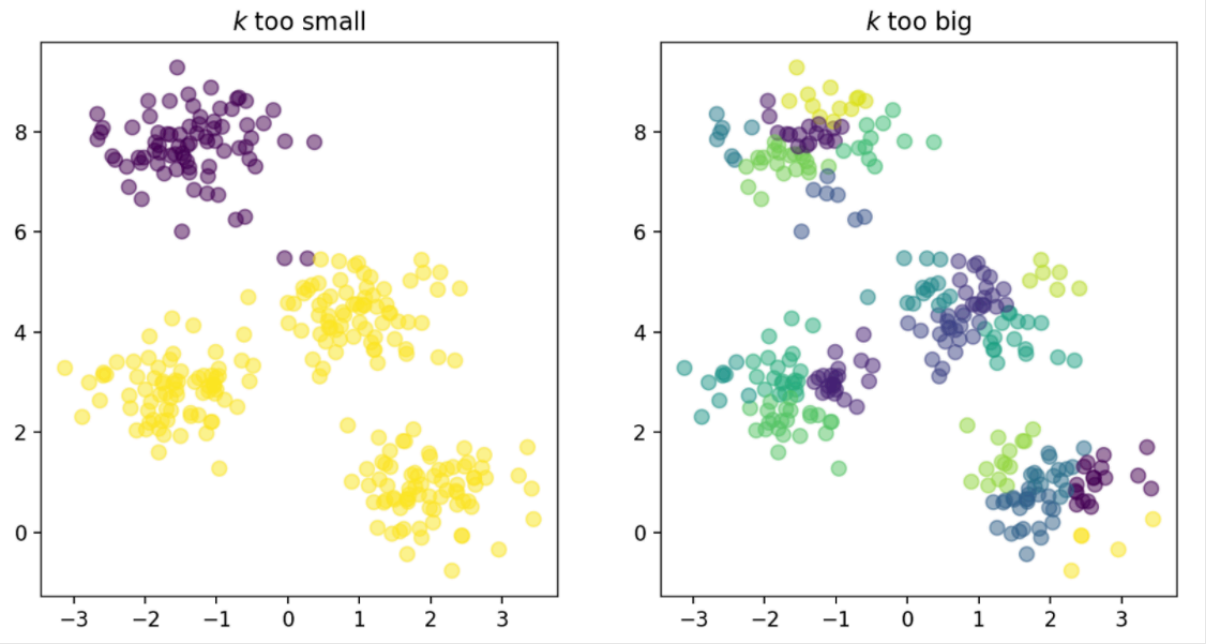
K-means clustering (Steinhaus 1950)

- Assume there are k groups
- Take k random points and assign them as centroids.
- Repeat until converge (i.e. no cluster moves its center).
    - Each member finds its closest cluster, add itself to the cluster
    - Each cluster computes its new centroid
    - If no cluster changes its centroid – done!




Iteration 1
Iteration 2
Iteration 3
Iteration 4
Iteration 5
Iteration 6
Iteration 7
Iteration 8
Iteration 9

Choosing the right value for k

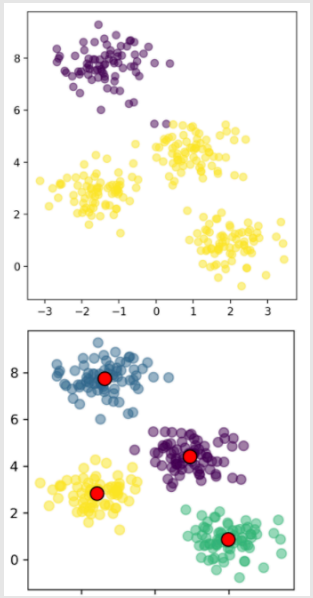


What is a good clustering

Sum of squared distance measures the 'correctness' of the clustering

$$Min \sum_{i=1}^{k} \sum_{x \in S_i} \|x - c_i\|^2 = \sum_{i=1}^{k} |S_i| \, Var S_i$$

where $c_i = \frac{1}{|S_i|} \sum_{x \in S_i} x_i$



K=2 → SSD=1200

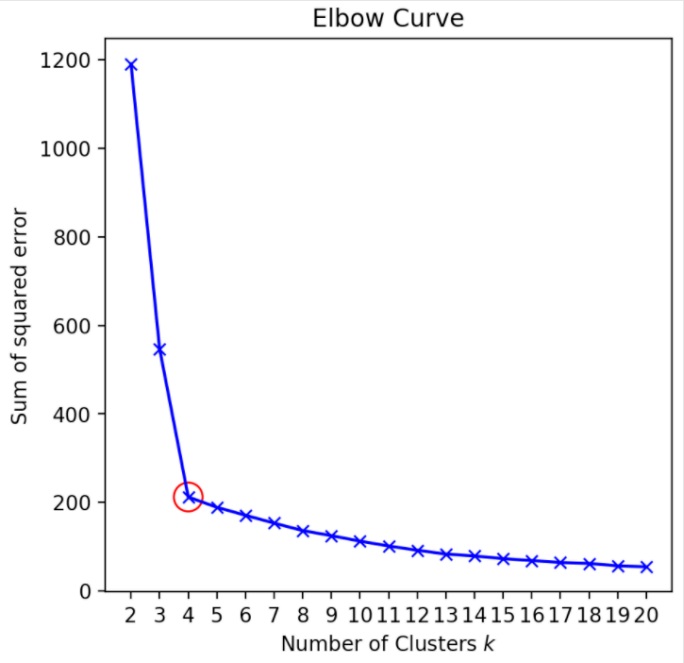
K=4 → SSD=221

$$\sum_{i=1}^{k} \sum_{x \in S_i} \|x - c_i\|^2 = \ ?$$

Choosing the right value for k

- Try various k
- Evaluate sum of squared distance
  $\sum_{i=1}^{k} \sum_{x \in S_i} \|x - c_i\|^2$

- Look for and “elbow” point

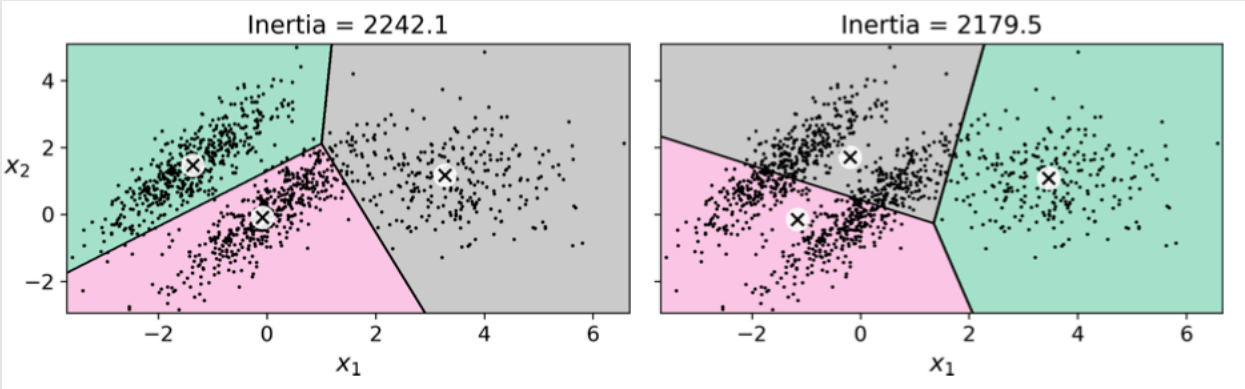


Limits of K-Means

- We use a greedy algorithm to solve the K–Means clustering

  The algorithm is not guaranteed to reach global optima. It's necessary to run the algorithm several times with different initialization

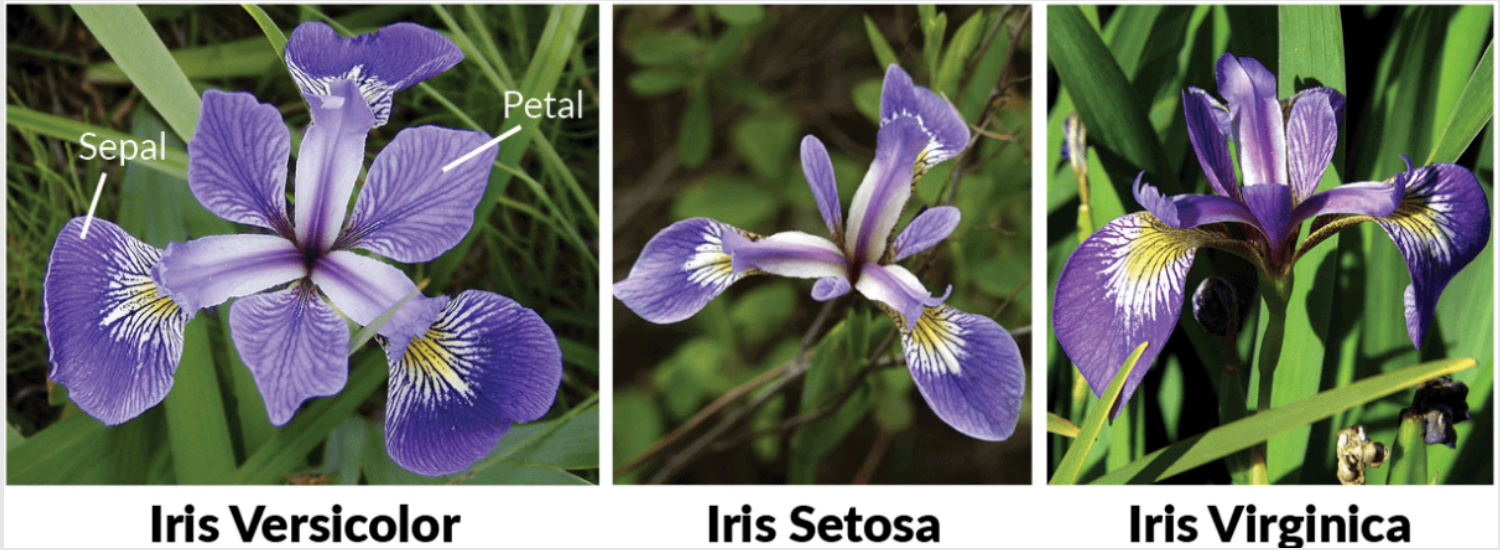
- K-Means does not behave well when are clusters have varying sizes, different densities or non-spherical shapes



K nearest neighbor classification

Iris classification

Given a dataset of observations of items, predict the type of each item

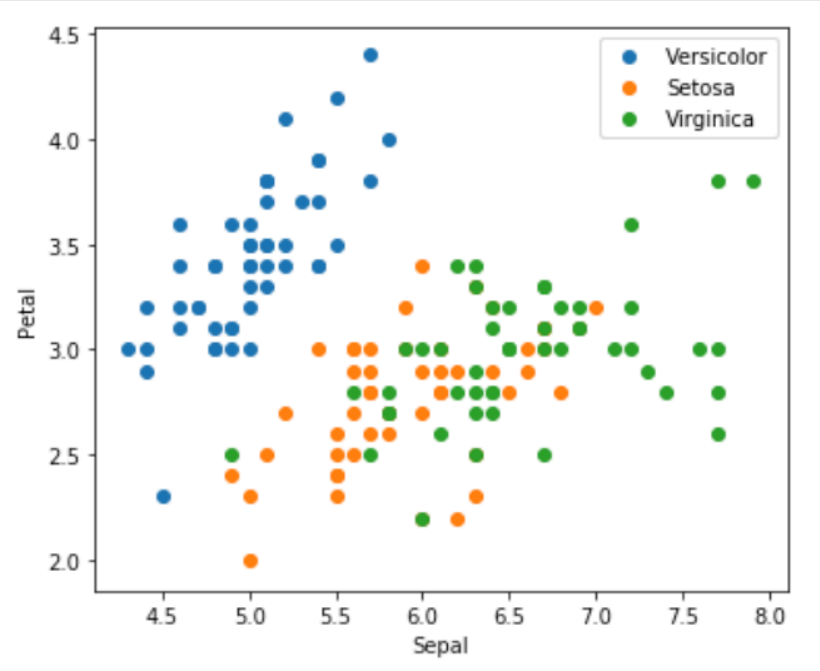


Classification

The features are numeric and the output labels are discrete

	Sepal $\in \mathbb{R}$	Petal $\in \mathbb{R}$	Genus $\in \mathbb{N}^3$
0	5.1	3.5	0
1	4.9	3.0	0
2	4.7	3.2	0
3	4.6	3.1	0
4	5.0	3.6	0

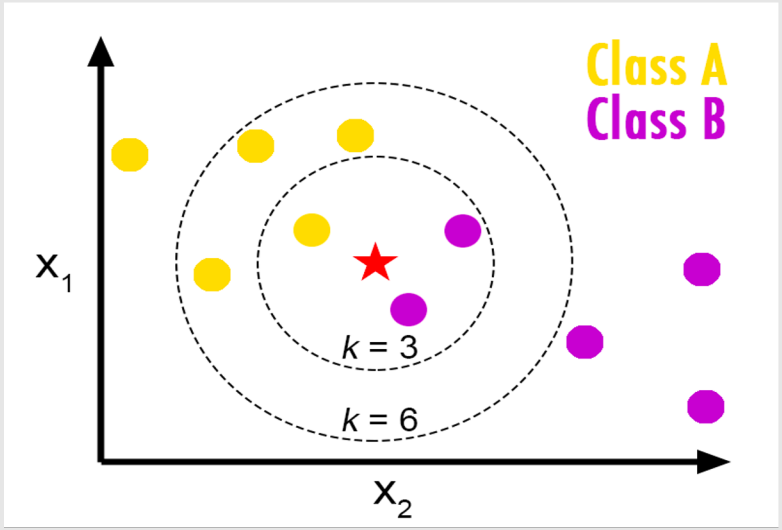
Plotting the data



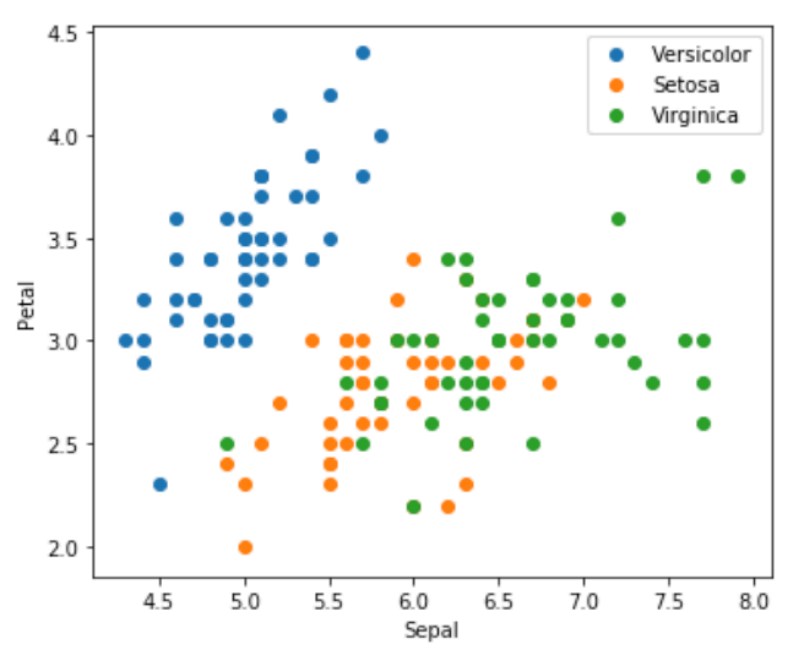
Try something like k-means?

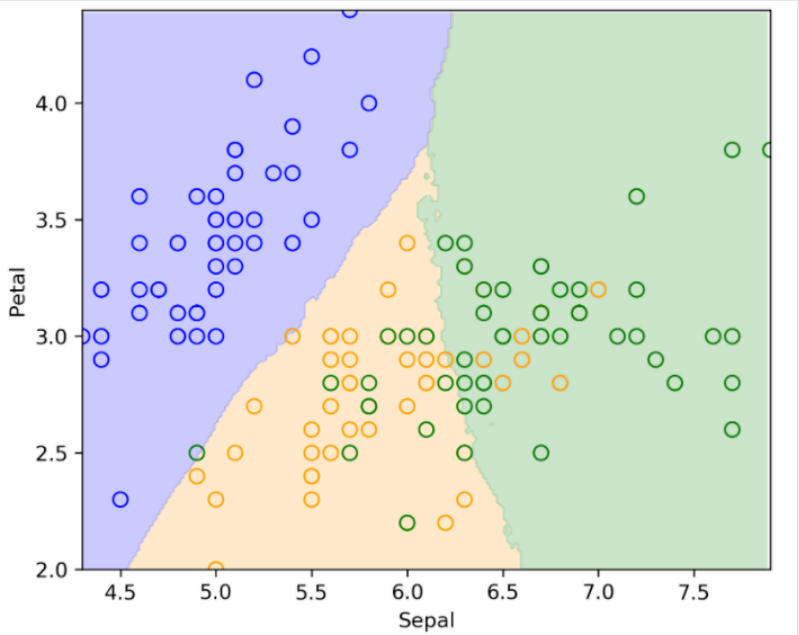
K nearest neighbor (Nilsson, 1965)

- Store all the observations
- Pick a value k
- When asked to make a new prediction, return the most frequent occurring class of the k-nearest neighbors.

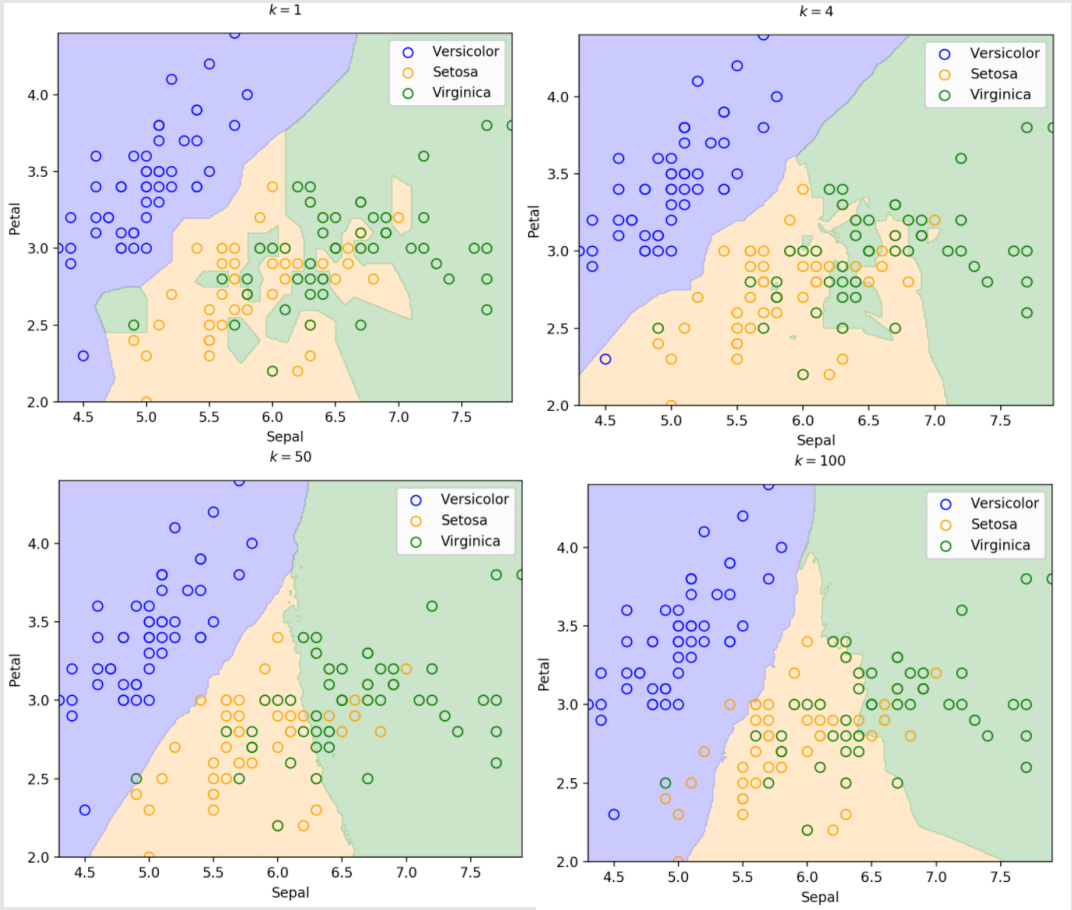


Example


Versicolor
Setosa
Virginica
Petal
Sepal
4.5
4.0
3.5
3.0
2.5
2.0
4.5
5.0
5.5
6.0
6.5
7.0
7.5
8.0


Petal
Sepal
4.0
3.5
3.0
2.5
2.0
4.5
5.0
5.5
6.0
6.5
7.0
7.5

Example – different choice of k



KNN regression

KNN -- Regression

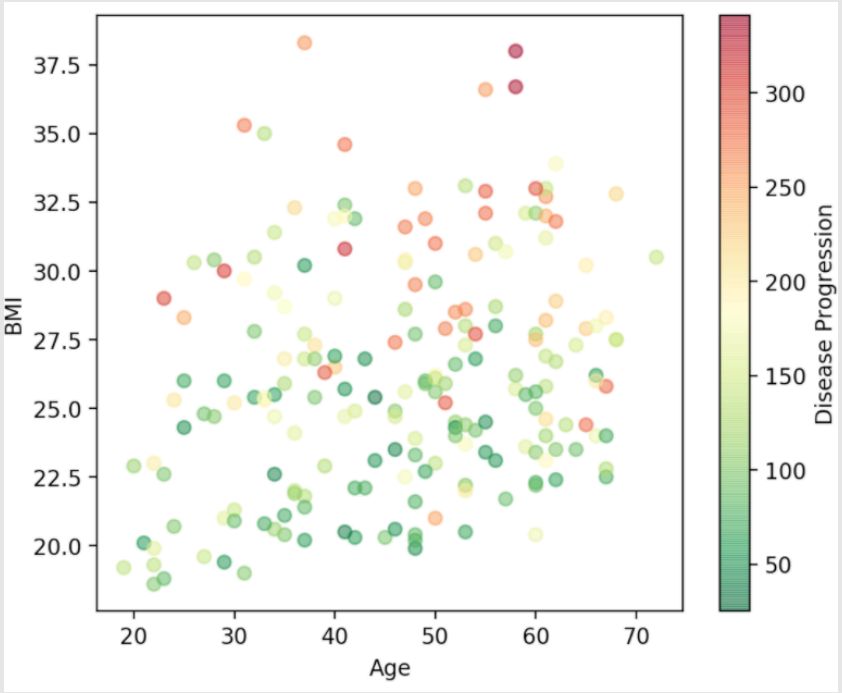
Given a dataset of observations of humans, predict diabetes disease progression (numeric).

Regression

The features are numeric and the output label is continuous

	AGE $\in \mathbb{R}$	BMI $\in \mathbb{R}$	Y $\in \mathbb{R}$
0	59	32.1	151
1	48	21.6	75
2	72	30.5	141
3	24	25.3	206
4	50	23.0	135

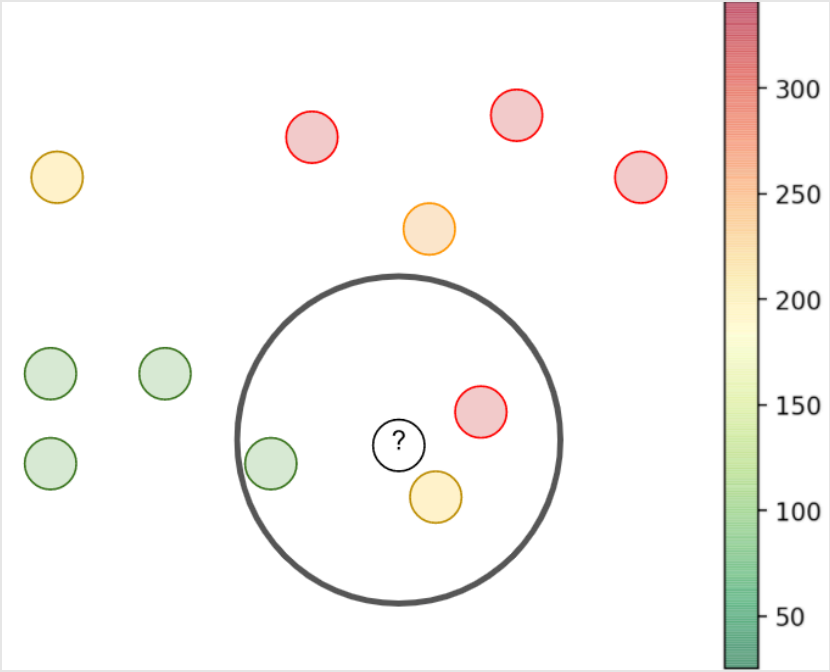
Plot the data



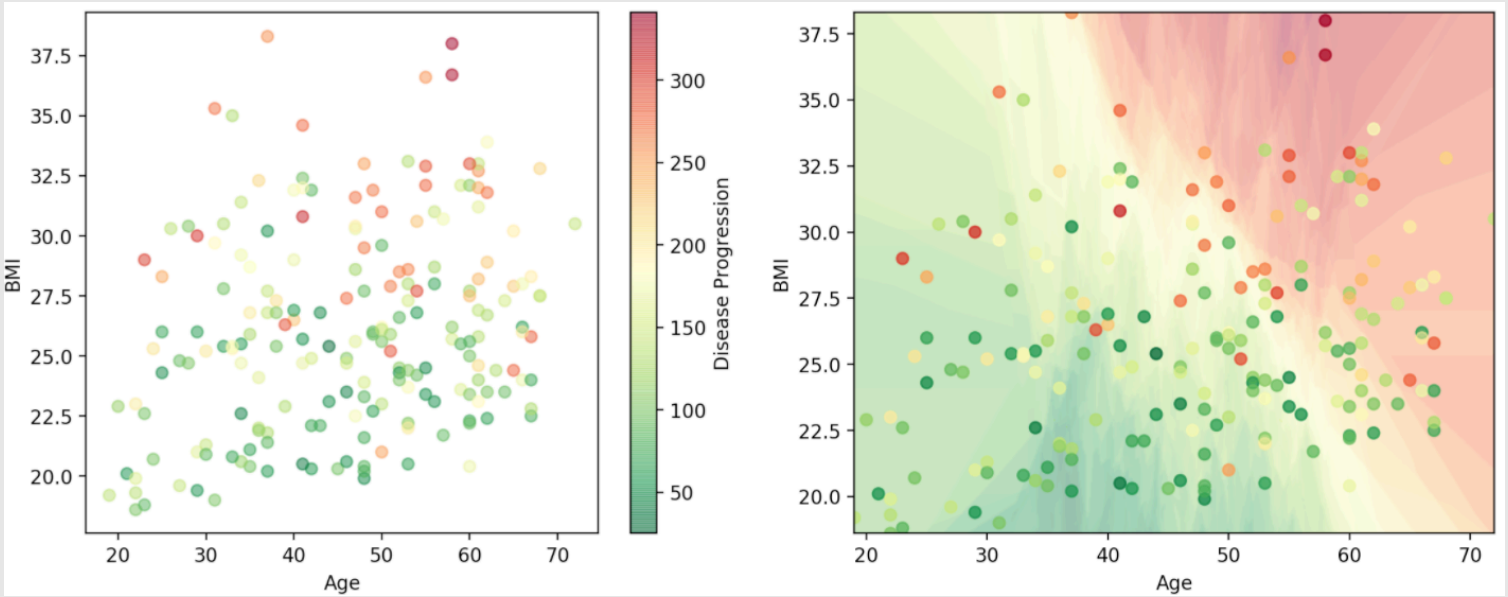
Try something like k-NN classification?

K nearest neighbor (Nilsson, 1965)

- Store all the observations
- Pick a value k
- When asked to make a new prediction, return **the average of the k-nearest neighbors.**



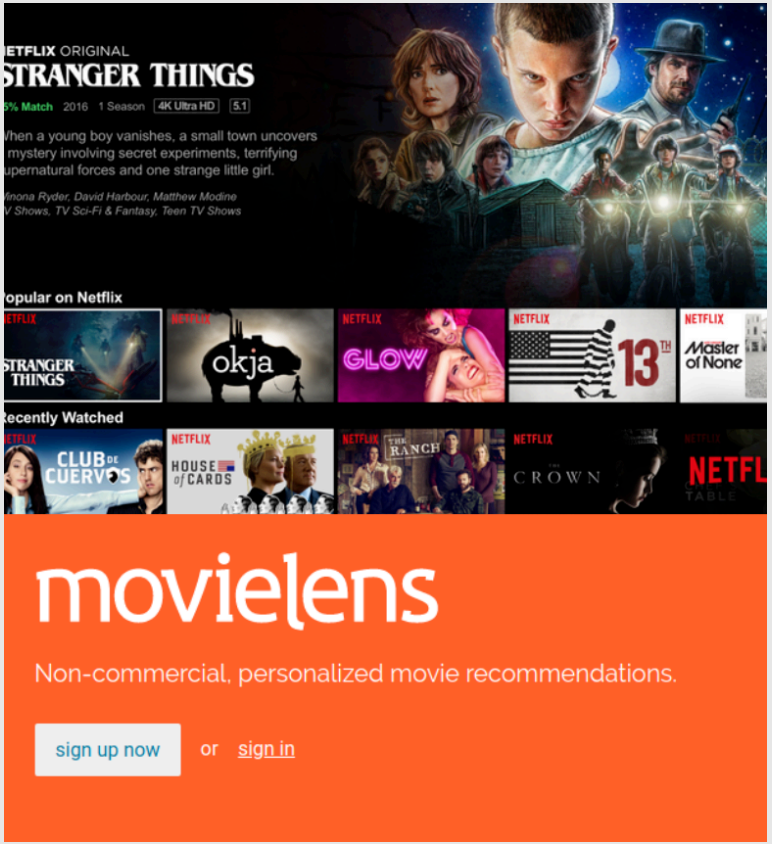
Example



For any future point, apply the rule

Example: KNN used for recommendation system

Given rating of a user, recommend new movies

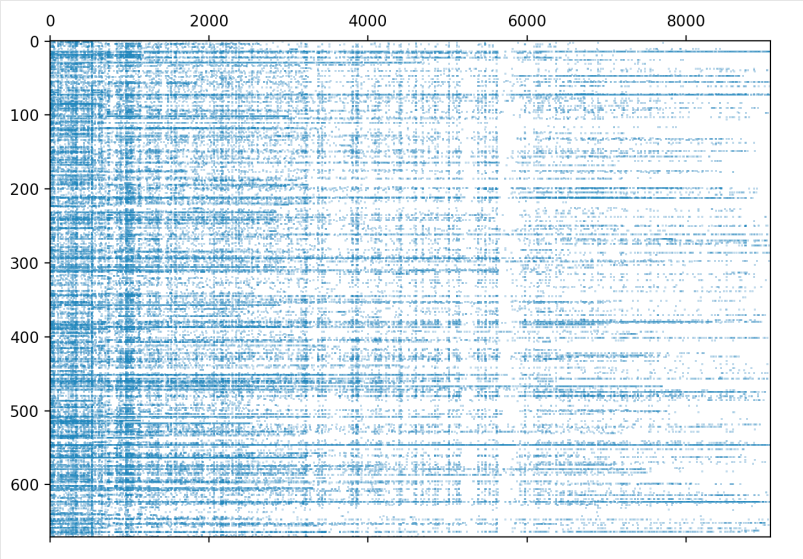


The rating dataset

User	M_1	M_2	M_3	M_4
0	0	0	2.5	0
1	0	3.0	5.0	4.0
2	0	0	3.5	0
3	0	4.5	0	0
...	...	...	...	...

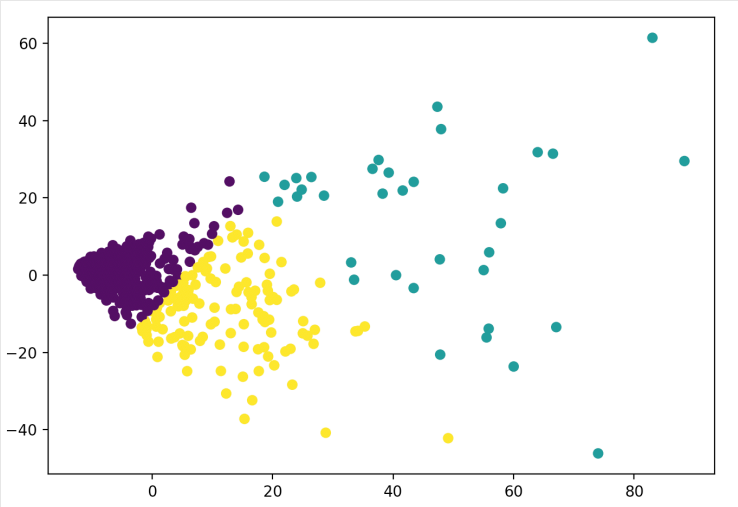
The rating dataset: 671 users × 9066 movies

Plotting the data



The data matrix is sparse with some very popular movies and some completely unknown ones

Plotting the data using k-means



Each user can be represented using a 9066 dimensional vector. Compress 9066 movies in 2 dimensions using PCA and apply k-means clustering algorithm

Collaborative Filtering

- User-Based Collaborative Filtering
  - Find the K-nearest neighbors (KNN) to the user A, using a similarity function to measure the distance between each pair of users. Collect the movies that A's 'neighbors' have watched and A has not.

- Item-Based Collaborative Filtering
  - Calculate similarity among the items. Based on the item that the user is interested in, recommend other similar items.

Test user1: the “epic adventure lover”

	title	genres
movieId		
260	Star Wars: Episode IV - A New Hope (1977)	Action|Adventure|Sci-Fi
69524	Raiders of the Lost Ark: The Adaptation (1989)	Action|Adventure|Thriller

	movieId	title	genres
232	260	Star Wars: Episode IV - A New Hope (1977)	Action|Adventure|Sci-Fi
966	1210	Star Wars: Episode VI - Return of the Jedi (1983)	Action|Adventure|Sci-Fi
953	1196	Star Wars: Episode V - The Empire Strikes Back (1980)	Action|Adventure|Sci-Fi
8603	112852	Guardians of the Galaxy (2014)	Action|Adventure|Sci-Fi
2062	2571	Matrix, The (1999)	Action|Sci-Fi|Thriller
522	589	Terminator 2: Judgment Day (1991)	Action|Sci-Fi
4135	5445	Minority Report (2002)	Action|Crime|Mystery|Sci-Fi|Thriller
3869	4993	Lord of the Rings: The Fellowship of the Ring, The (2001)	Adventure|Fantasy
1024	1270	Back to the Future (1985)	Adventure|Comedy|Sci-Fi

Test user2: the “thriller killer”

	title	genres
movieId		
296	Pulp Fiction (1994)	Comedy|Crime|Drama|Thriller
2571	Matrix, The (1999)	Action|Sci-Fi|Thriller

	movieId	title	genres
2062	2571	Matrix, The (1999)	Action|Sci-Fi|Thriller
266	296	Pulp Fiction (1994)	Comedy|Crime|Drama|Thriller
2288	2858	American Beauty (1999)	Drama|Romance
472	527	Schindler's List (1993)	Drama|War
535	608	Fargo (1996)	Comedy|Crime|Drama|Thriller
1288	1617	L.A. Confidential (1997)	Crime|Film-Noir|Mystery|Thriller
525	593	Silence of the Lambs, The (1991)	Crime|Horror|Thriller
2212	2762	Sixth Sense, The (1999)	Drama|Horror|Mystery
263	293	Léon: The Professional (a.k.a. The Professional) (Léon) (1994)	Action|Crime|Drama|Thriller