18-661 Introduction to Machine Learning

Clustering, Part I

Fall 2020

ECE – Carnegie Mellon University

Outline

1. Clustering
2. k-means
3. k-means++

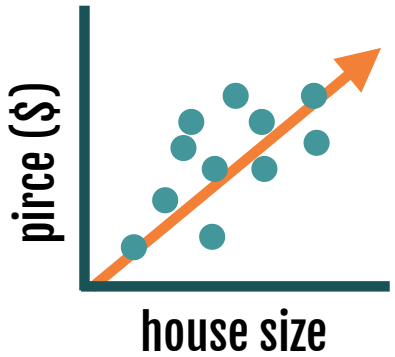
Outline

1. Clustering
2. k-means
3. k-means++

Clustering

Supervised Learning: Regression

How much should you sell your house for?

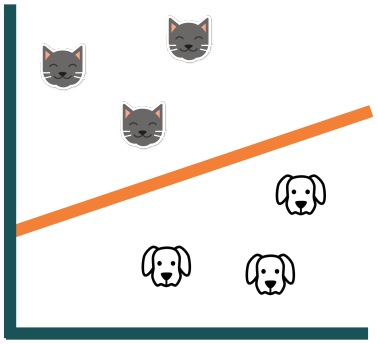


input: houses & features **learn**: $x \rightarrow y$ relationship **predict**: y (*continuous*)

Supervised Learning: Classification

Cat or dog?

input: cats and dogs



learn: $x \rightarrow y$ relationship

predict: y (*categorical*)

Supervised versus Unsupervised Learning

Supervised Learning: labeled observations $\{(\boldsymbol{x}_1, y_1), \ldots (\boldsymbol{x}_n, y_n)\}$

- Labels 'teach' algorithm to learn mapping from observations to labels
- Examples: Classification (Logistic Reg., SVMs, Neural Nets, Nearest Neighbors, Decision Trees), Regression (Linear Reg., Neural Nets)

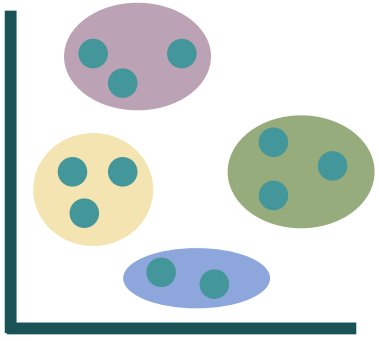
Unsupervised Learning: unlabeled observations $\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}$

- Learning algorithm must find patterns from features alone
- Can be goal in itself (discover hidden patterns, exploratory analysis)
- Can be means to an end (pre-processing for supervised task)
- Examples:
  - K-means clustering (today), Gaussian Mixture Models (next week)
  - Dimensionality Reduction: Transform an initial feature representation into a more concise representation

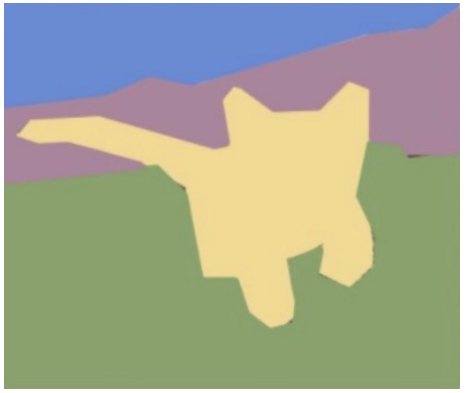
Clustering

How to segment an image?

input: raw pixels $\{x\}$



separate: $\{x\}$ into sets



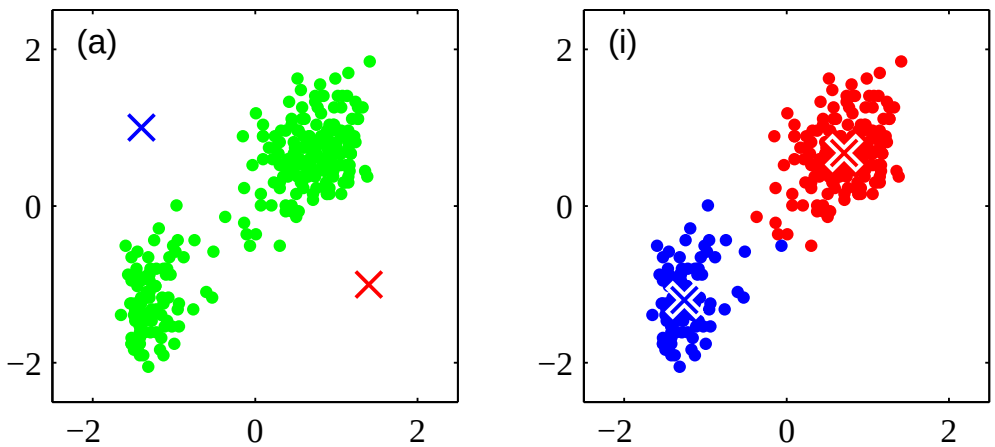
output: cluster labels $\{z\}$

Clustering

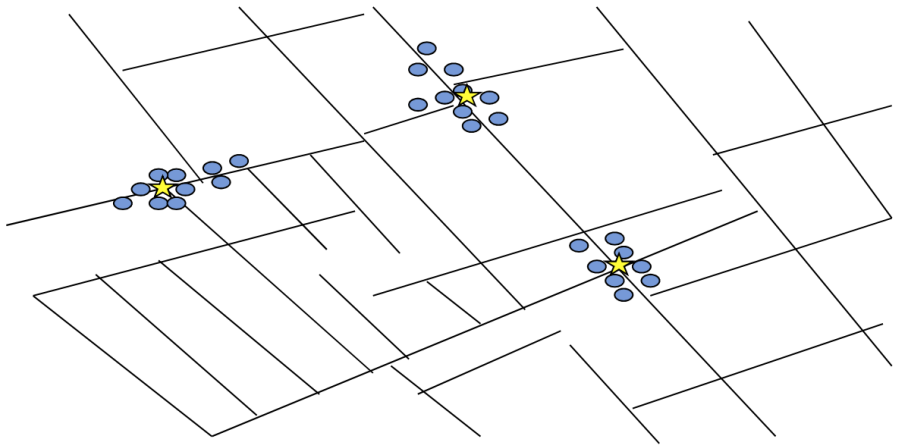
Setup Given $\mathcal{D} = \{\boldsymbol{x}_n\}_{n=1}^N$ and K, we want to output:

- $\{\boldsymbol{\mu}_k\}_{k=1}^K$: prototypes of clusters
- $A(\boldsymbol{x}_n) \in \{1, 2, \ldots, K\}$: the cluster membership

Toy Example Cluster data into two clusters.



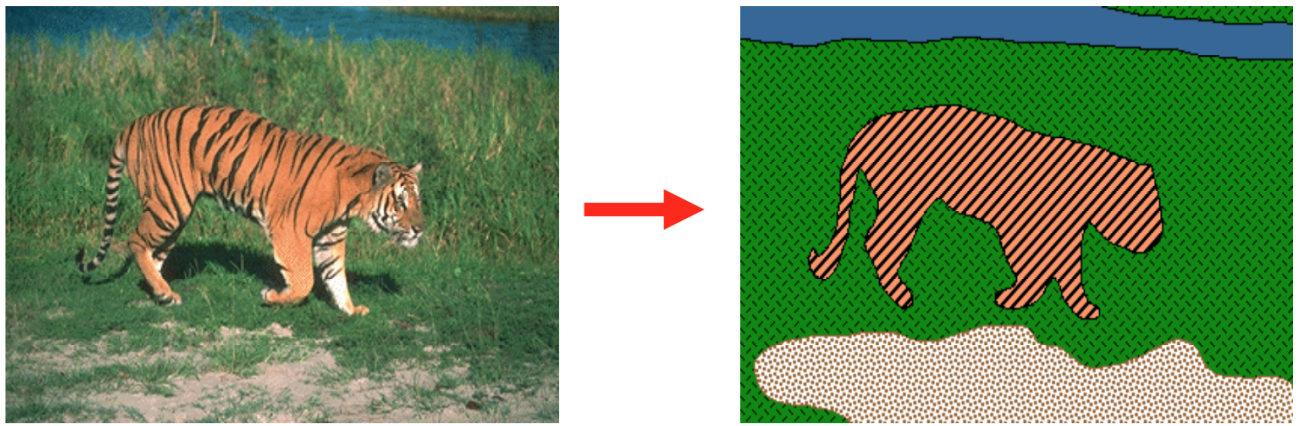
First? Example of Clustering



- John Snow, a London physician plotted the location of cholera deaths on a map during an outbreak in the 1850s.
- The locations indicated that cases were clustered around certain intersections where there were polluted wells – thus exposing both the problem and the solution.
- This story is all the more relevant today as we are trying to overcome the COVID-19 outbreak

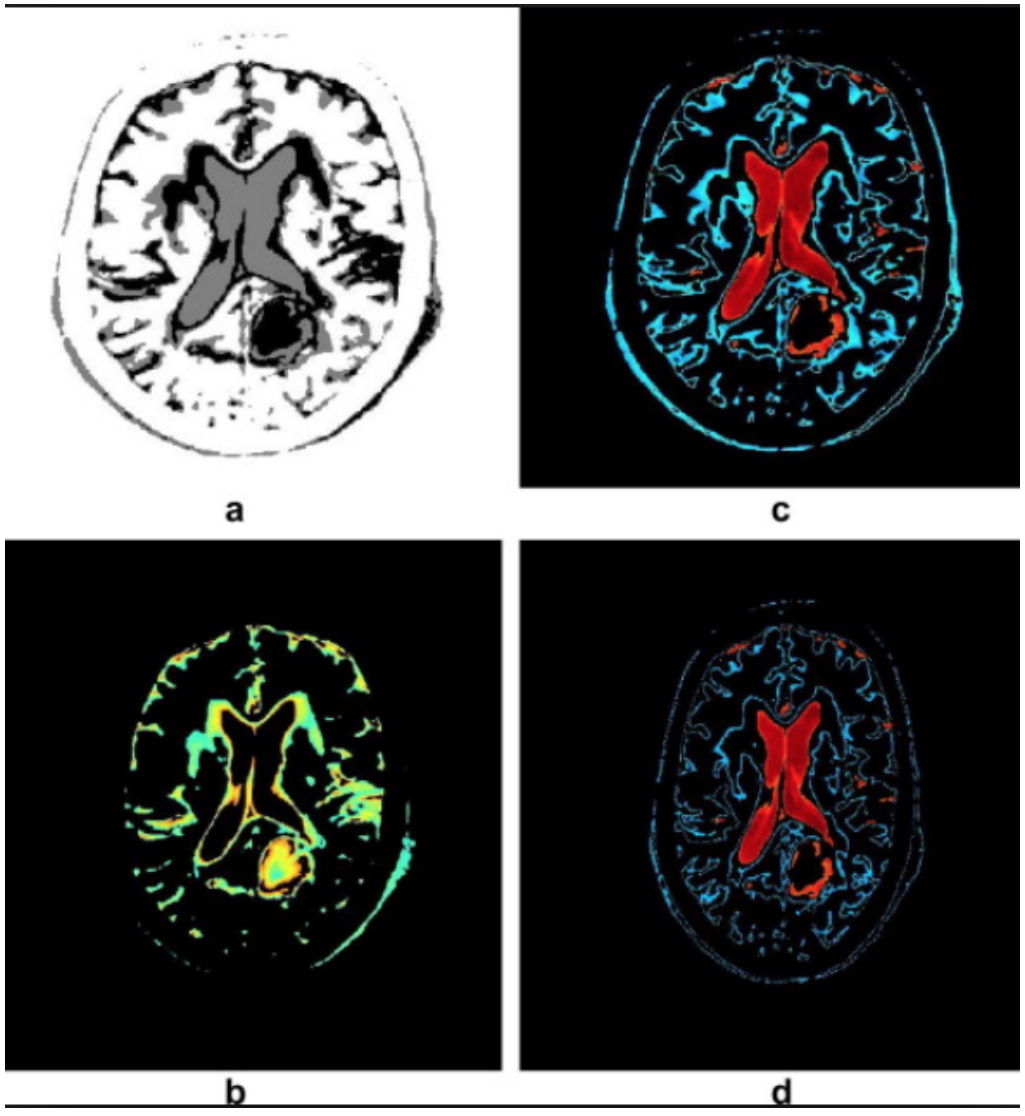
More examples

Image segmentation into foreground and background



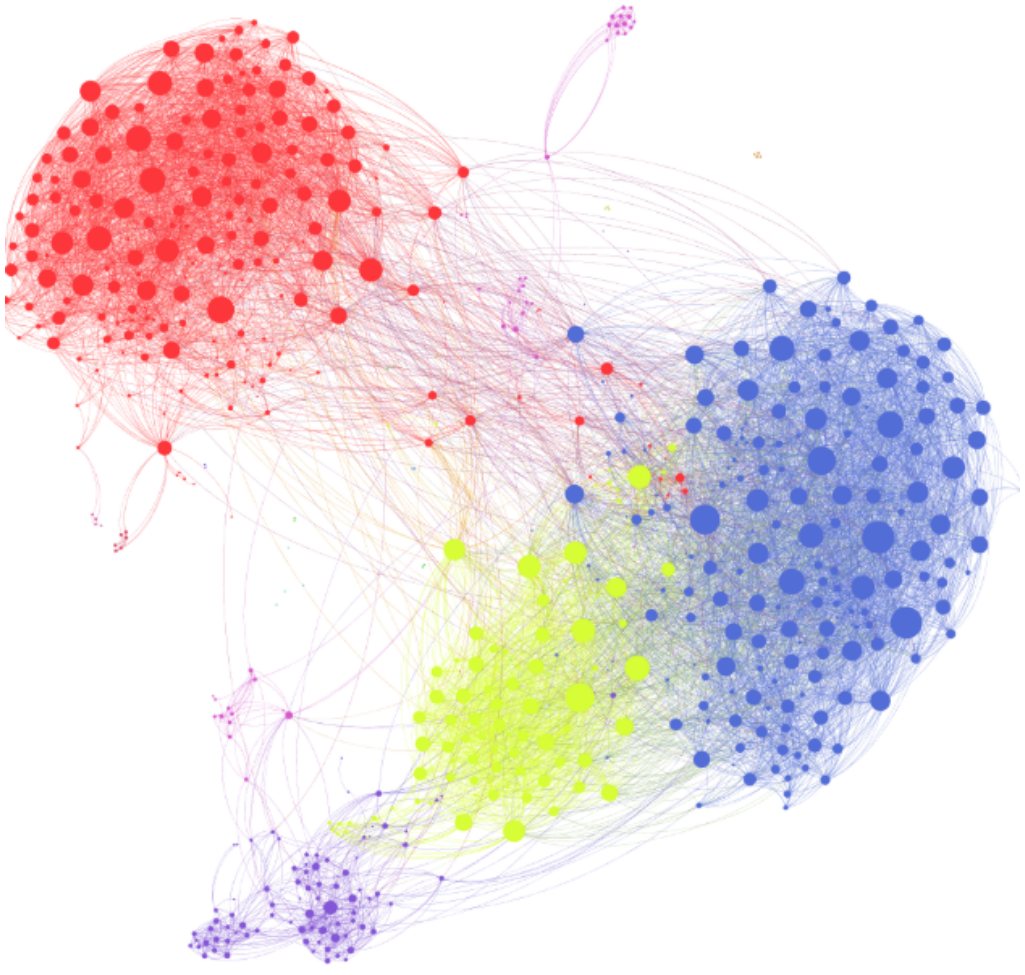
More examples

Detecting brain lesions from MRI Scans



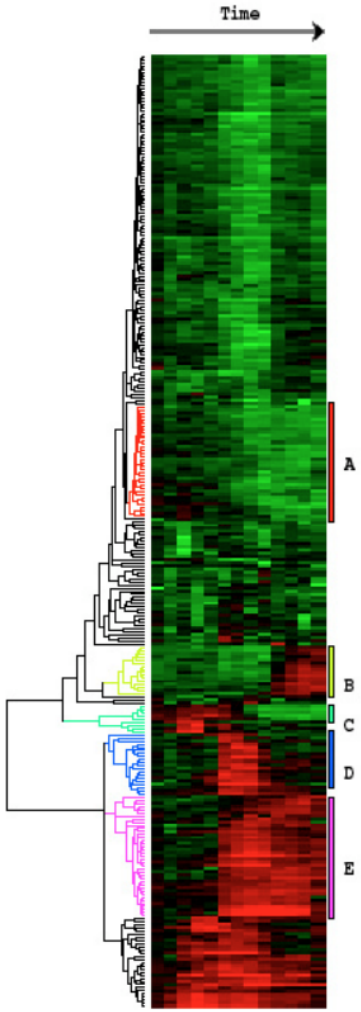
More examples

Social network analysis



More examples

Clustering gene expression data



Clustering

Today we will cover two methods for clustering

- k-means
- k-means++

k-means

k-means

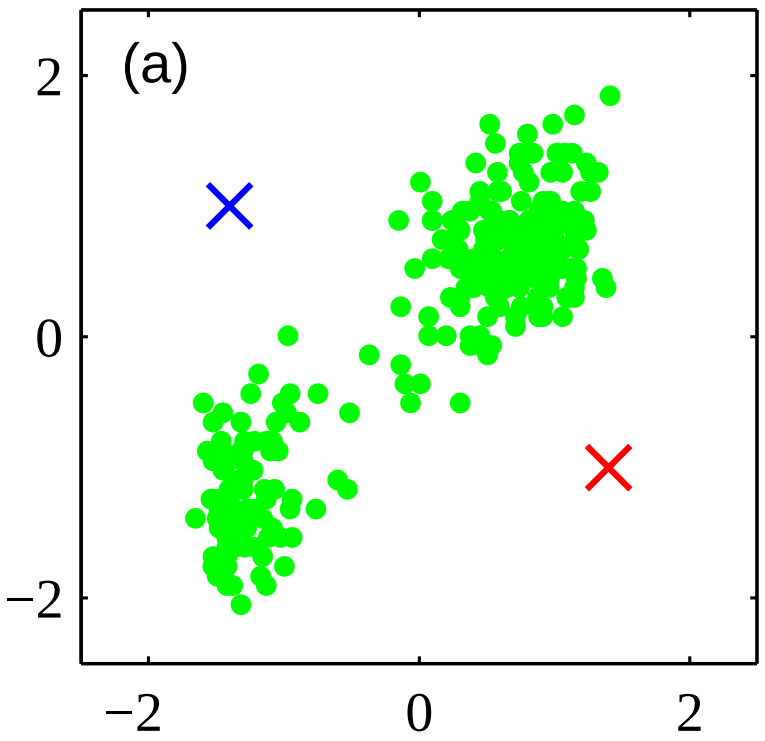
k-means: an iterative clustering method

High-level idea:

- Initialize: Pick k random points as cluster centers, $\{\mu_1, \ldots, \mu_k\}$
- Alternate:
  1. Assign data points to closest cluster center in $\{\mu_1, \ldots, \mu_k\}$
  2. Change each cluster center to the average of its assigned points
- Stop: When the clusters are stable

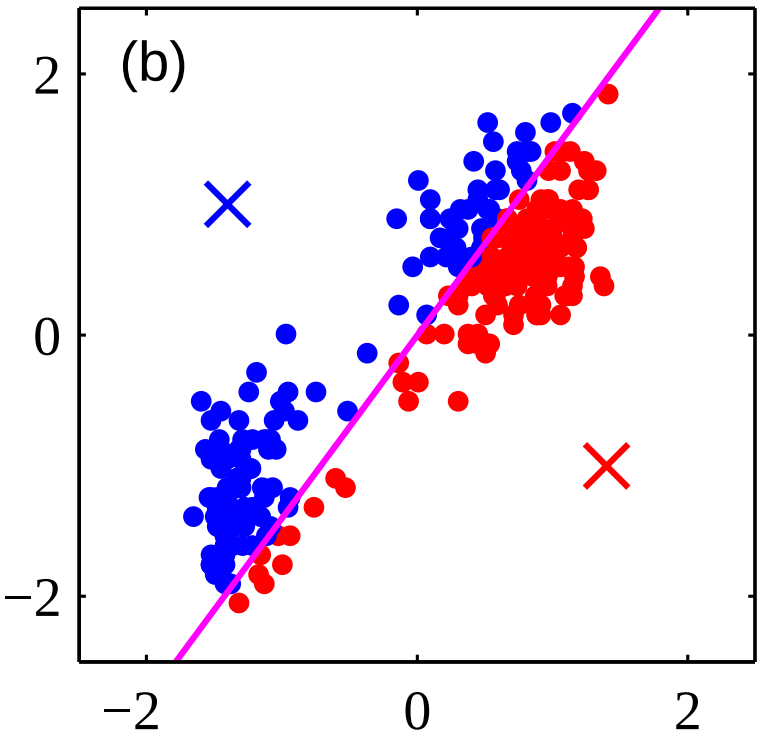
k-means example

- Initialize: Pick k random points as cluster centers
- (Shown here for k=2)



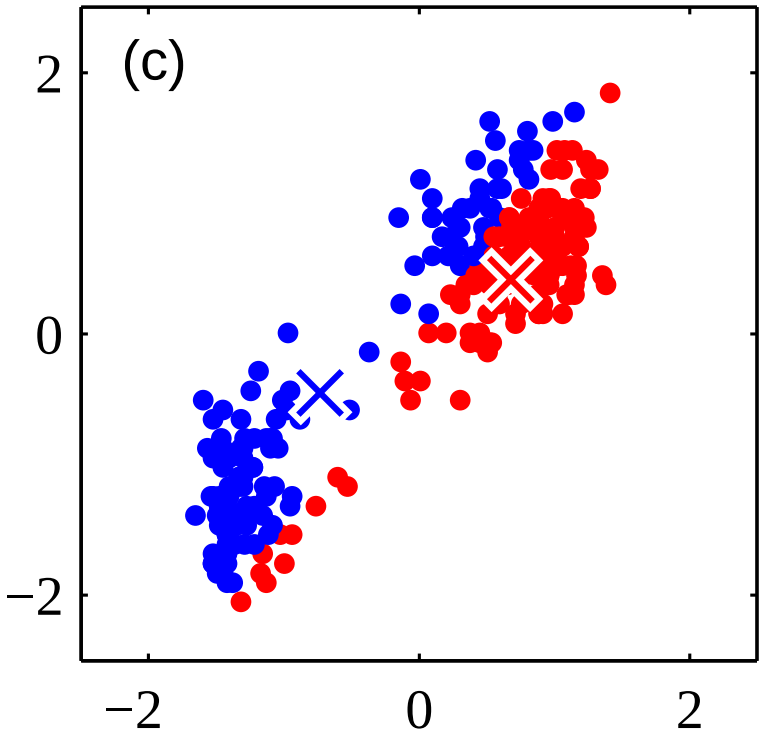
k-means example

- Alternating Step 1: Assign data points to closest cluster center



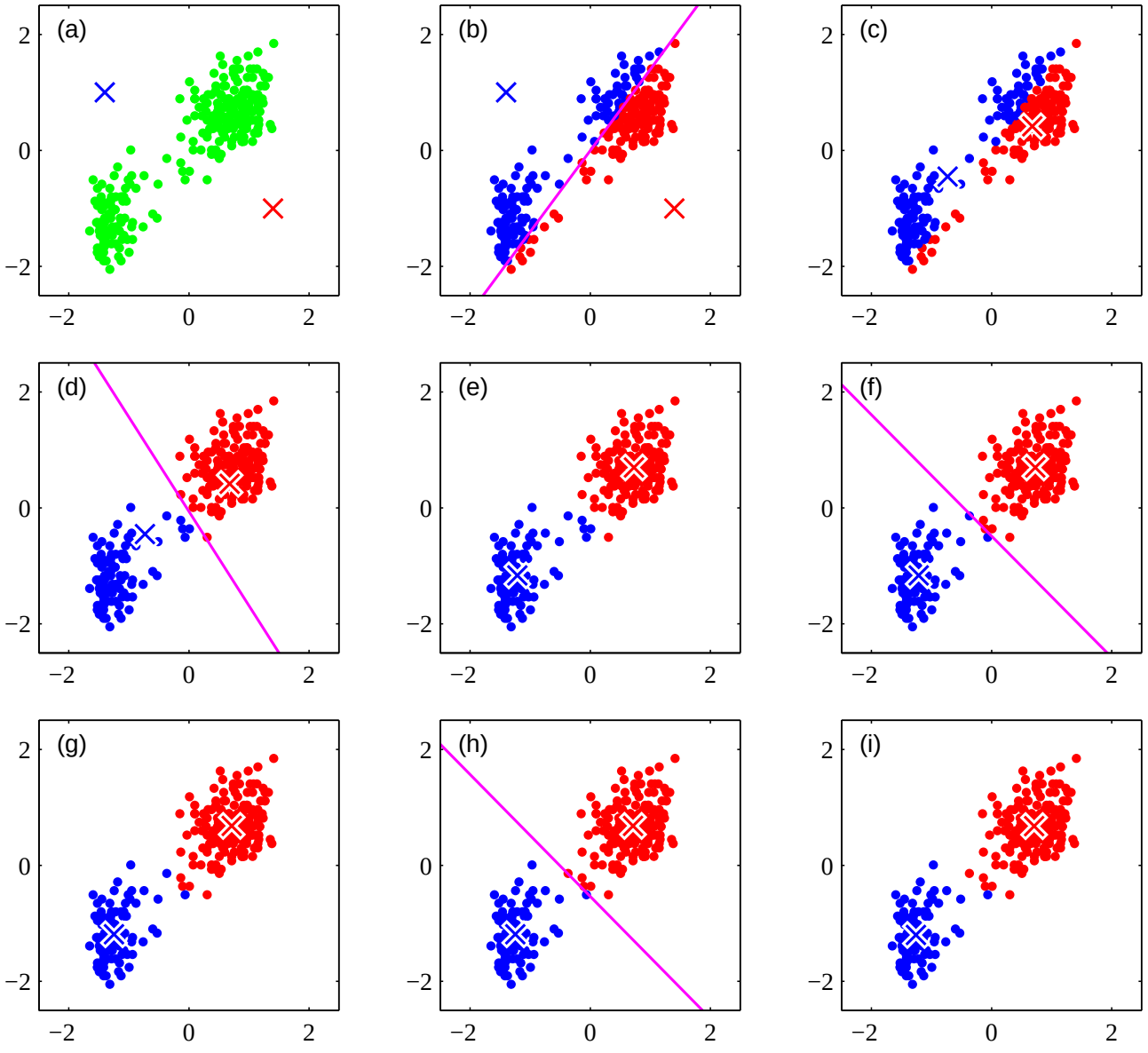
k-means example

- Alternating Step 2: Change the cluster center to the average of the assigned points



Then: Repeat . . .

k-means example (several iterations)



k-means clustering: details

Intuition: Data points assigned to cluster k should be near prototype $\boldsymbol{\mu}_k$

Distortion measure: (clustering objective function, cost function)

$$J = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{nk} \|\boldsymbol{x}_n - \boldsymbol{\mu}_k\|^2 = \sum_{k=1}^{K} \underbrace{\sum_{n:A(\boldsymbol{x}_n)=k} \|\boldsymbol{x}_n - \boldsymbol{\mu}_k\|^2}_{\text{spread within the } k\text{th cluster}}$$

where $r_{nk} \in \{0, 1\}$ is an indicator variable

$$r_{nk} = 1 \quad \text{if and only if} \quad A(\boldsymbol{x}_n) = k$$

Notes:

- Distance measure: $\|\boldsymbol{x}_n - \boldsymbol{\mu}_k\|^2$ calculates how far $\boldsymbol{x}_n$ is from the cluster center $\boldsymbol{\mu}_k$
- Canonical example is the 2-norm, i.e., $\|\cdot\|_2^2$, but could be some other distance measure!

Algorithm

$$J = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{nk} \|\boldsymbol{x}_n - \boldsymbol{\mu}_k\|^2$$

- What are the variables that we need to optimize? $\{r_{nk}\}$ and $\{\boldsymbol{\mu}_k\}$
- Difficult to jointly optimize both
- Solution: Alternative optimization between $\{r_{nk}\}$ and $\{\boldsymbol{\mu}_k\}$

- **Step 0** Initialize $\{\boldsymbol{\mu}_k\}$ to some values
- **Step 1** Fix $\{\boldsymbol{\mu}_k\}$ and minimize over $\{r_{nk}\}$, to get this assignment:

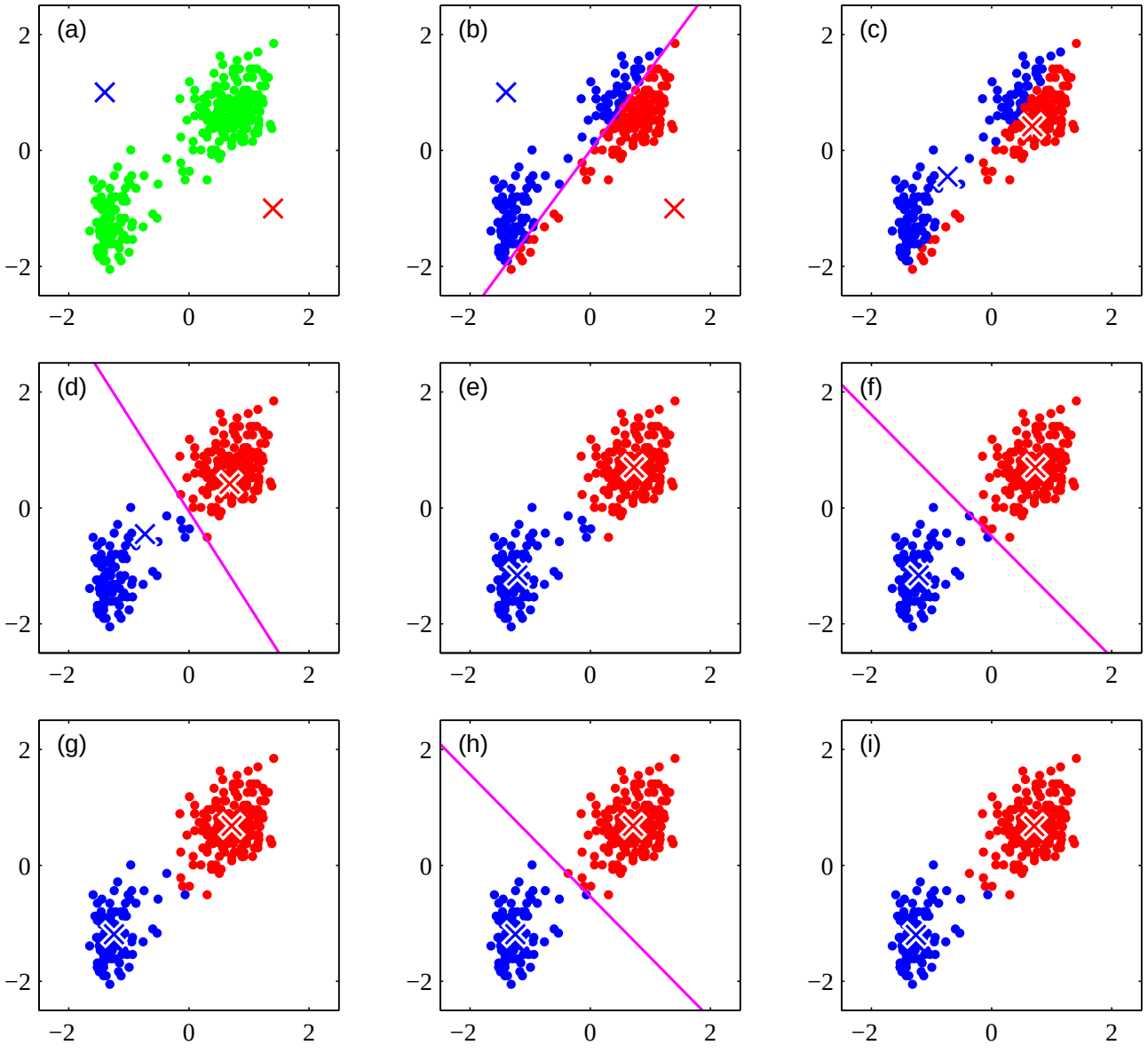
$$r_{nk} = \begin{cases} 1 & \text{if } k = \text{argmin}_j \|\boldsymbol{x}_n - \boldsymbol{\mu}_j\|^2 \\ 0 & \text{otherwise} \end{cases}$$

- **Step 2** Fix $\{r_{nk}\}$ and minimize over $\{\boldsymbol{\mu}_k\}$ to get this update:

$$\boldsymbol{\mu}_k = \frac{\sum_n r_{nk} \boldsymbol{x}_n}{\sum_n r_{nk}}$$

- **Step 3** Return to Step 1 unless stopping criterion is met

k-means example (several iterations)



Properties of k-means algorithm

Does it converge?

- **Guaranteed to converge in a finite number of iterations**
  - Key idea: k-means is an alternating optimization approach
  - Each step is guaranteed to decrease the objective/cost function—thus guaranteed to converge
  - *However*, may converge to a *local minimum* (objective is non-convex)

What's the runtime?

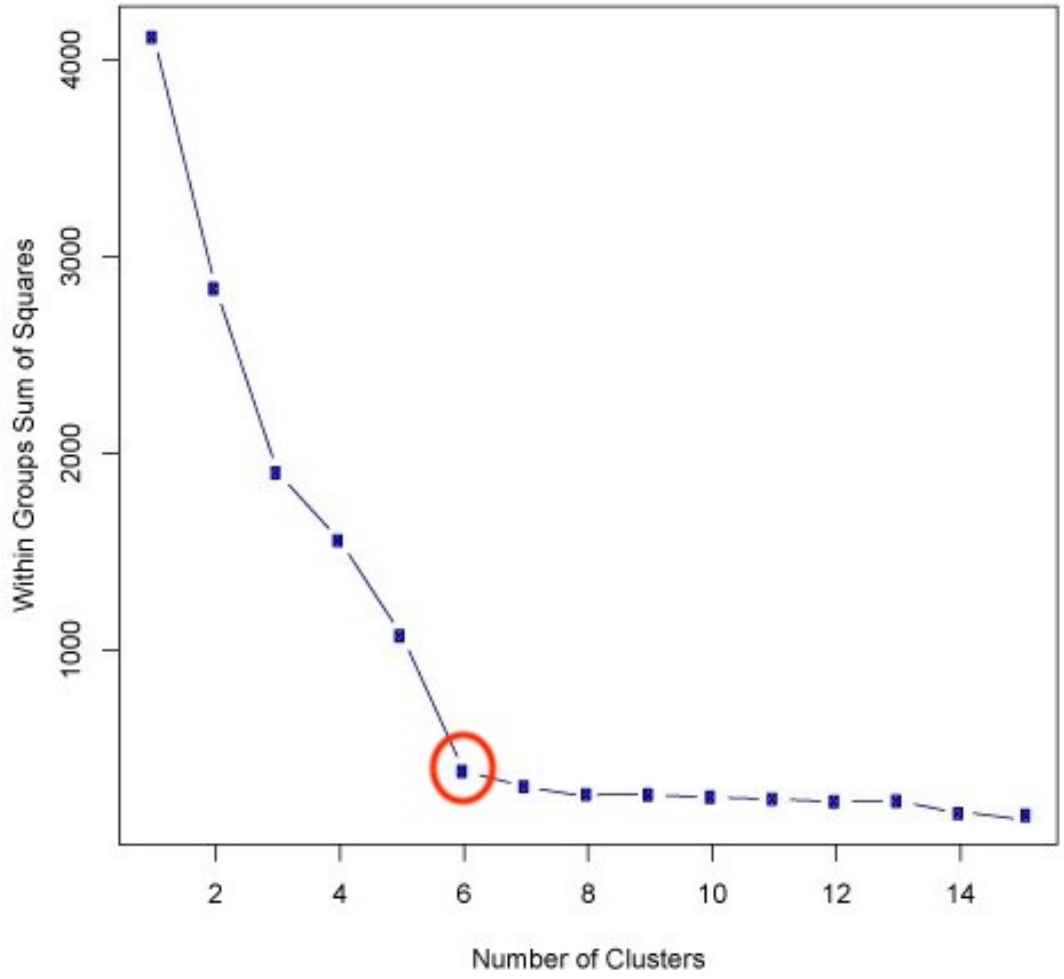
- **Running time per iteration**:
  - Assume: n data points, each with d features, and k clusters
  - Assign data points to closest cluster: $O(ndk)$
  - Re-compute cluster centers: $O(ndk)$
- **Thus, total runtime is**: $O(ndki)$, where i is the number of iterations

Practical Issues with k-means

- How to select k?
  - Prior knowledge
  - Heuristics (e.g., elbow method)

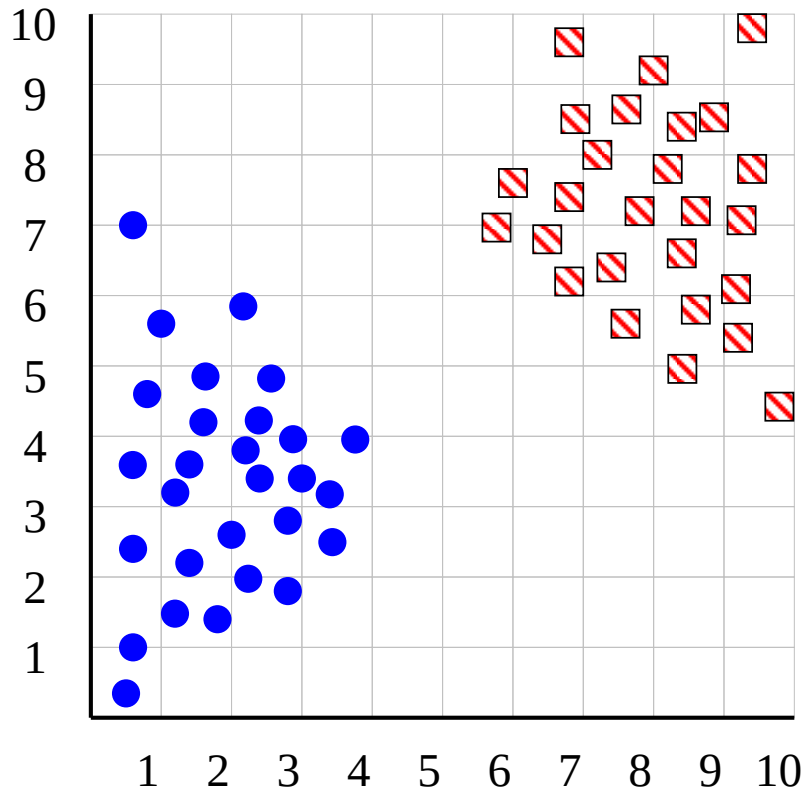
Elbow method

Key idea: select a small value of k that adding a new cluster doesn't reduce the within-cluster distances much

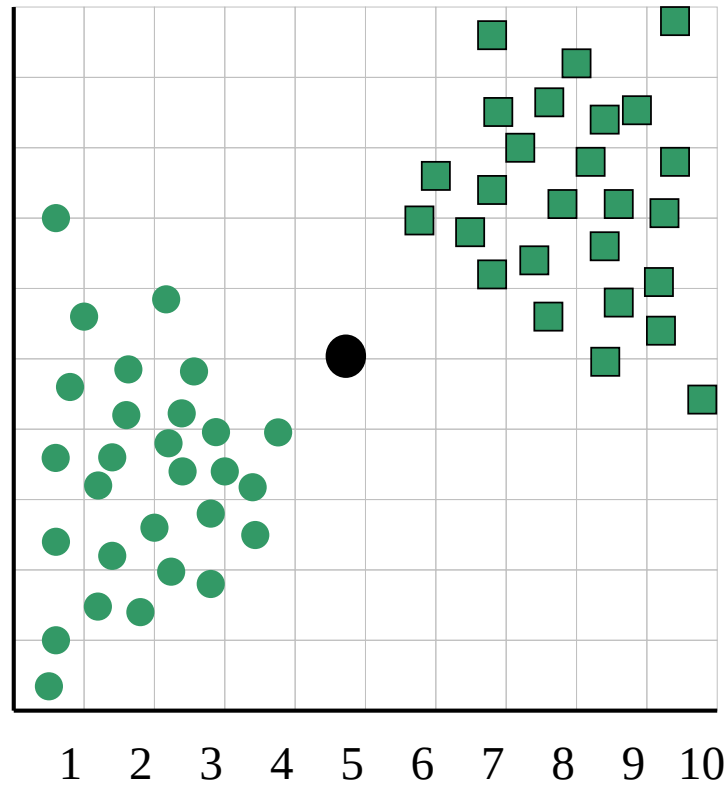


How can we tell the *right* number of clusters?

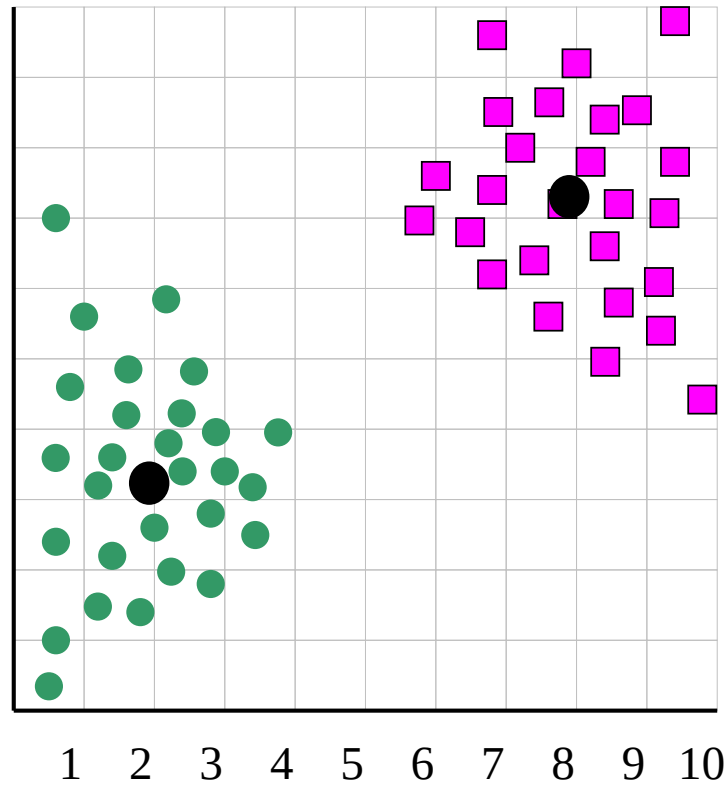
In general, this is a unsolved problem. However there are many approximate methods. In the next few slides we will see an example.



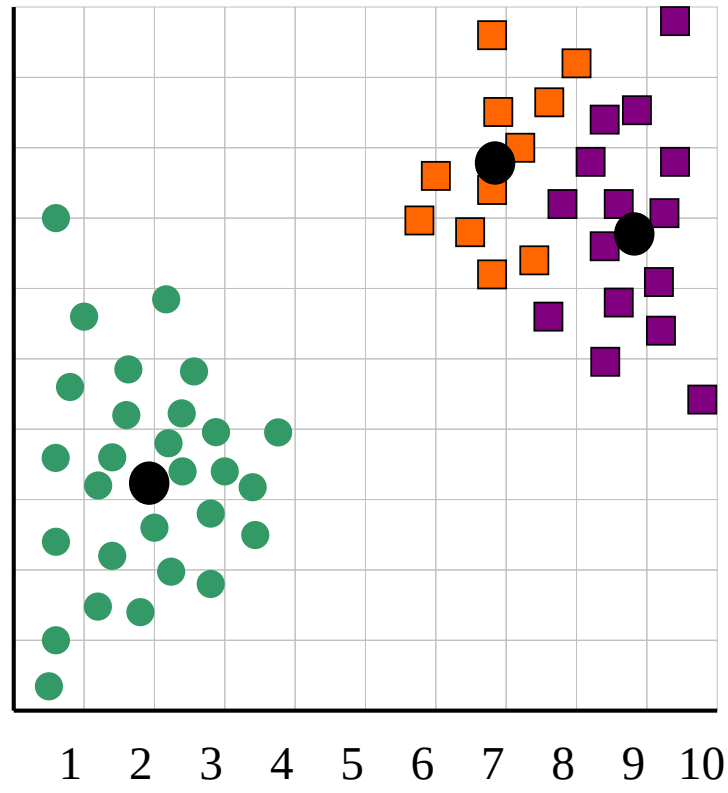
When k = 1, the objective function is 873.0



When k = 2, the objective function is 173.1



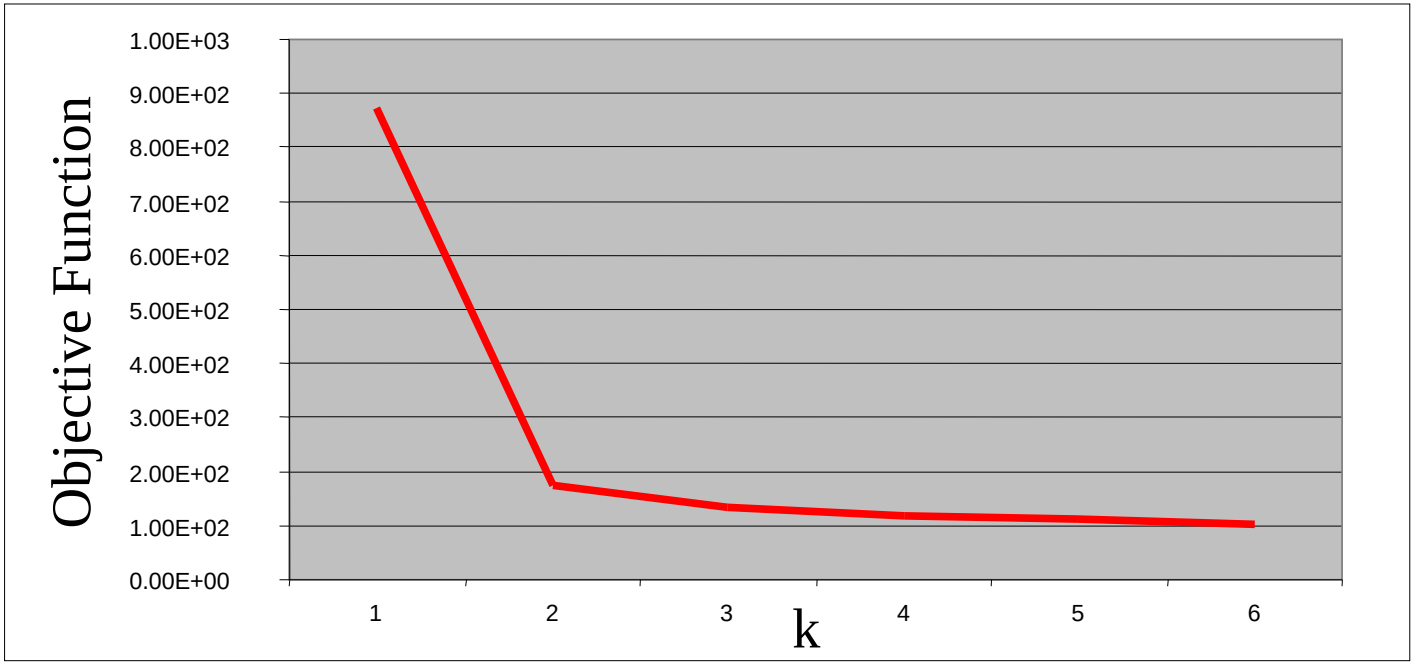
When k = 3, the objective function is 133.6



Elbow Method

We can plot the objective function values for k equals 1 to 6…

The abrupt change at k = 2, is highly suggestive of two clusters in the data. This technique for determining the number of clusters is known as “knee finding” or “elbow finding”.

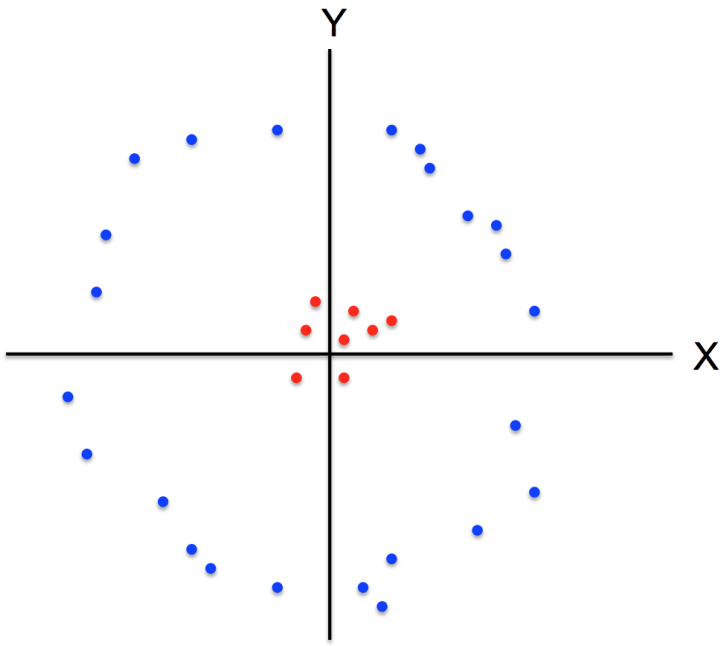


Note that the results are not always as clear cut as in this toy example

Practical Issues with k-means

- How to select k?
  - Prior knowledge
  - Heuristics (e.g., elbow method)
- How to select distance measure?
  - Often requires some knowledge of problem
  - Some examples: Euclidean distance (for images), Hamming distance (distance between two strings), shared key words (for websites)

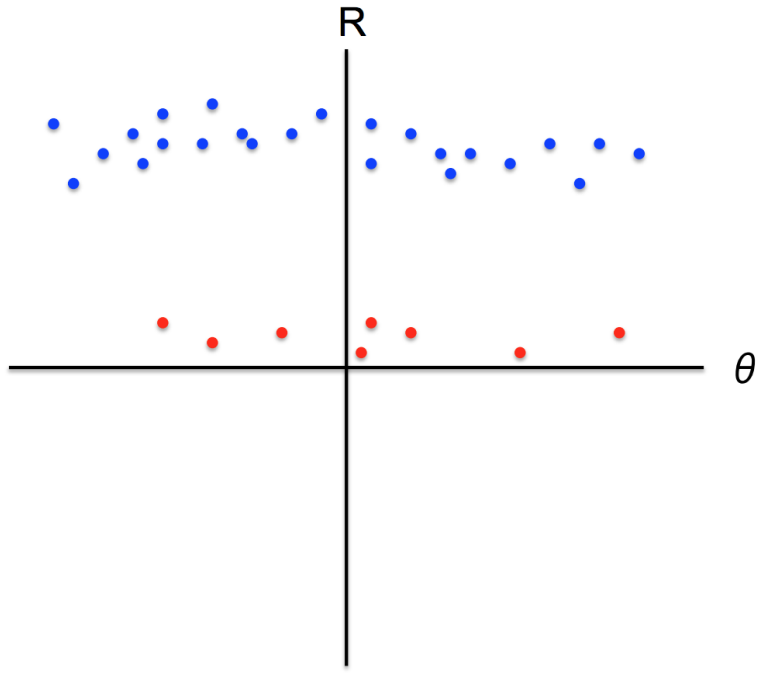
How to get k-means to work on this data?



Should look at the distance of the data points from the origin $\sqrt{x_n^2 + y_n^2}$

Distance measure

Changing features (distance measure) can help



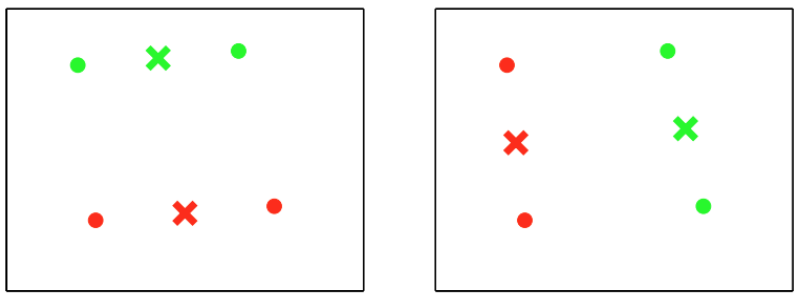
If the cluster i mean is $(\mu_{i,x}, \mu_{i,y})$, the distance of (x_n, y_n) from it can be defined as $|\sqrt{\mu_{i,x}^2 + \mu_{i,y}^2} - \sqrt{x_n^2 + y_n^2}|$

Practical Issues with k-means

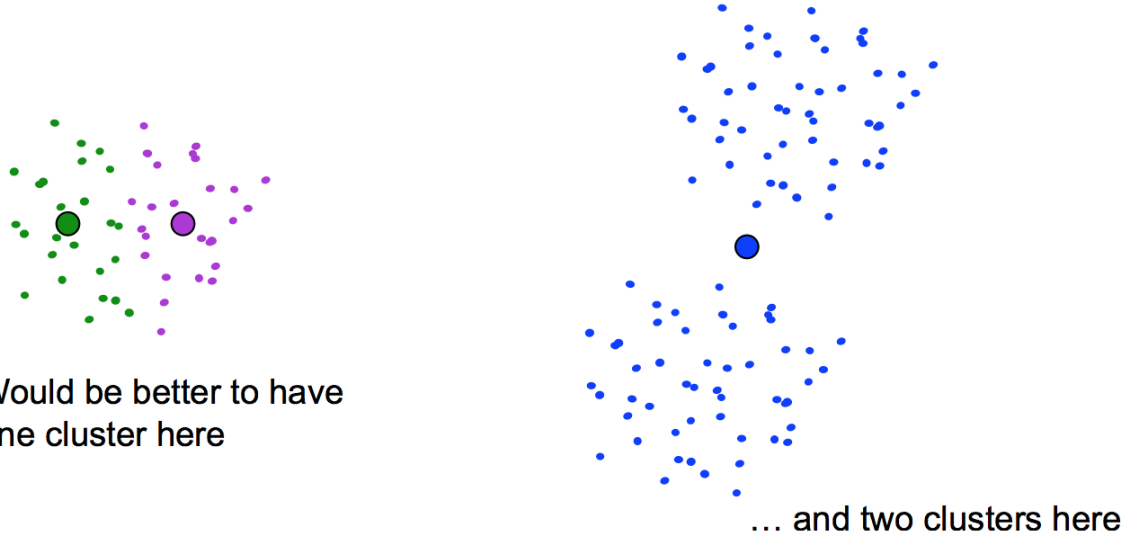
- How to select k?
  - Prior knowledge
  - Heuristics (e.g., elbow method)
- How to select distance measure?
  - Often requires some knowledge of problem
  - Some examples: Euclidean distance (for images), Hamming distance (distance between two strings), shared key words (for websites)
- How to initialize cluster centers?
  - The final clustering can depend significantly on the initial points you pick!

How to initialize cluster centers?

Random initialization can lead to *different results*



Choosing k is also non-trivial



k-means++

k-means++

Key idea: Run k-means, but with a better initialization

- Choose center μ_1 at random
- For $j = 2, \ldots, k$
  - Choose μ_j among $x_1, \ldots, x_n$ with probability:

$$P(\mu_j = x_i) \propto min_{j' < j} \|x_i - \mu_{j'}\|^2$$

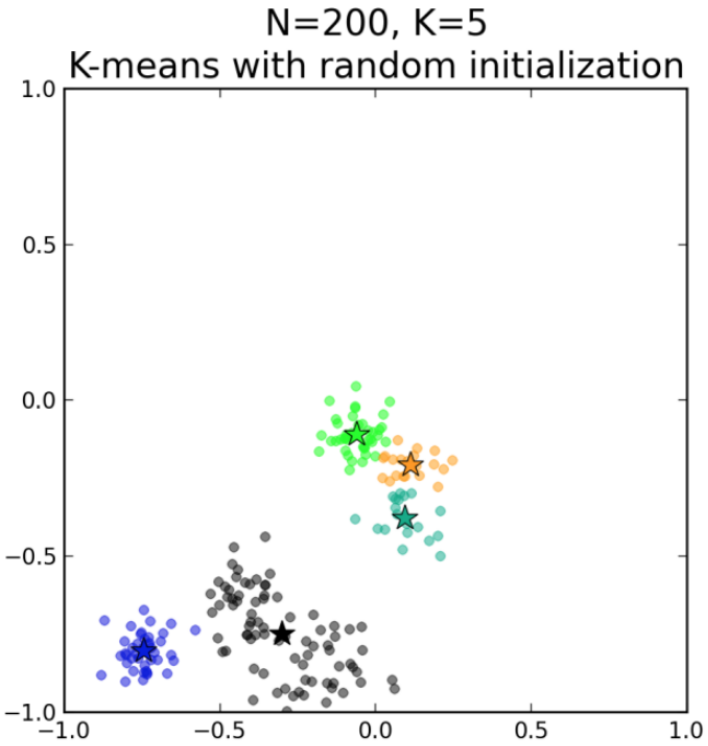
This means that if x_i is close to one of the already chosen cluster means $\mu_1, \ldots \mu_{j-1}$, then we assign a lower probability of selecting it as the next cluster mean.

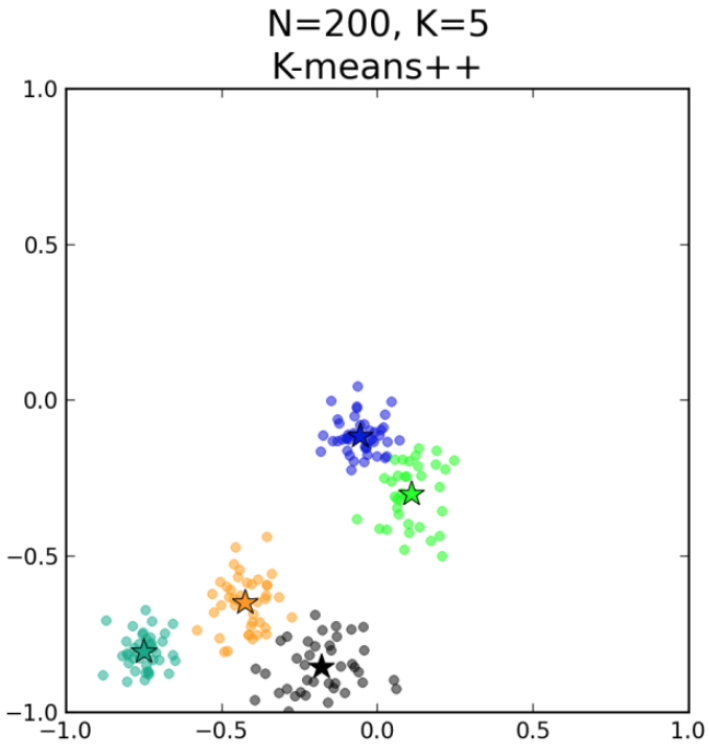
Initialization helps to get good coverage of the space

Theorem: k-means++ always obtains a $O(logk)$ approximation to the optimal solution in expectation.

Running k-means after this initialization can only improve on the result

k-means++


N=200, K=5
K-means with random initialization
1.0
0.5
0.0
−0.5
−1.0
−1.0
−0.5
0.0
0.5
1.0


N=200, K=5
K-means++
1.0
0.5
0.0
−0.5
−1.0
−1.0
−0.5
0.0
0.5
1.0

Connection to k-Nearest Neighbors

- Nearest Neighbors is a supervised learning method
  - Each training point $\mathbf{x}_n$ has a corresponding given label y_n
  - Objective: Assign label to a new $\mathbf{x}$ by looking at the labels of its k nearest points
- Clustering is an unsupervised learning method
  - We are given training points $\mathbf{x}_n$ without labels
  - Objective: Divide them into k groups to understand patterns in the data

The meaning of the parameter k is also different in these two methods

Clustering can make Nearest Neighbors more efficient

- A drawback of nearest neighbors is that we have to remember the training data
- Clustering can help compress the training data into a small number of representative points

Algorithm to Improve Nearest Neighbors

- For all training data points $\mathbf{x}_n$ with label $y_n = c$, for C classes $c = 1, \ldots C$, cluster the $\mathbf{x}_n$ into R groups.
- Store these R cluster means for each of the C classes
- For a test data point $\mathbf{x}$, find the k nearest neighbors among the RC cluster means and assign their majority label to $\mathbf{x}$

You should know . . .

- What unsupervised learning is
- What clustering is
- How to cluster using k-means
- Practical issues with k-means
- How k-means++ improves on k-means