K-Means
an example of unsupervised learning

CMSC 422

MARINE CARPUAT
marine@cs.umd.edu

Exercise: When are DT vs kNN appropriate?

Properties of classification problem	Can Decision Trees handle them?	Can K-NN handle them?
Binary features	yes	yes
Numeric features	yes	yes
Categorical features	yes	yes
Robust to noisy training examples	no (for default algorithm)	yes (when k > 1)
Fast classification is crucial	yes	no
Many irrelevant features	yes	no
Relevant features have very different scale	yes	no

Today's Topics

- A new algorithm
  - K-Means Clustering
- Fundamental Machine Learning Concepts
  - Unsupervised vs. supervised learning
  - Decision boundary

Clustering

- Goal: automatically partition examples into groups of similar examples

- Why? It is useful for
  - Automatically organizing data
  - Understanding hidden structure in data
  - Preprocessing for further analysis

What can we cluster in practice?

- news articles or web pages by topic
- protein sequences by function, or genes according to expression profile
- users of social networks by interest
- customers according to purchase history
- ...

Clustering

- Input
  - a set S of n points in feature space
  - a distance measure specifying distance $d(x_i,x_j)$ between pairs (x_i,x_j)
- Output
  - A partition $\{S_1,S_2, \ldots S_k\}$ of S

Supervised Machine Learning as Function Approximation

Problem setting

- Set of possible instances X
- Unknown target function $f: X \rightarrow Y$
- Set of function hypotheses $H = \{h \mid h: X \rightarrow Y\}$

Input

- Training examples $\{(x^{(1)}, y^{(1)}), \ldots (x^{(N)}, y^{(N)})\}$ of unknown target function f

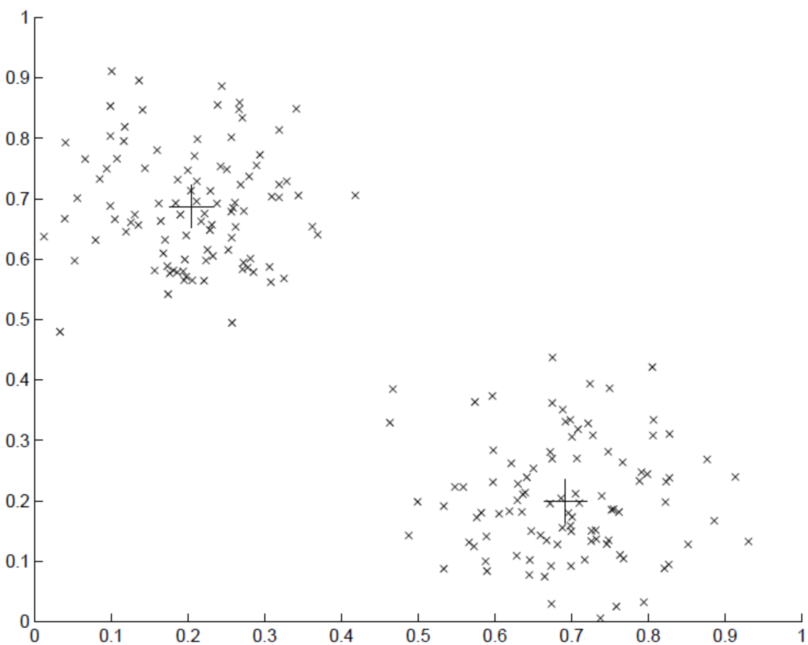
Output

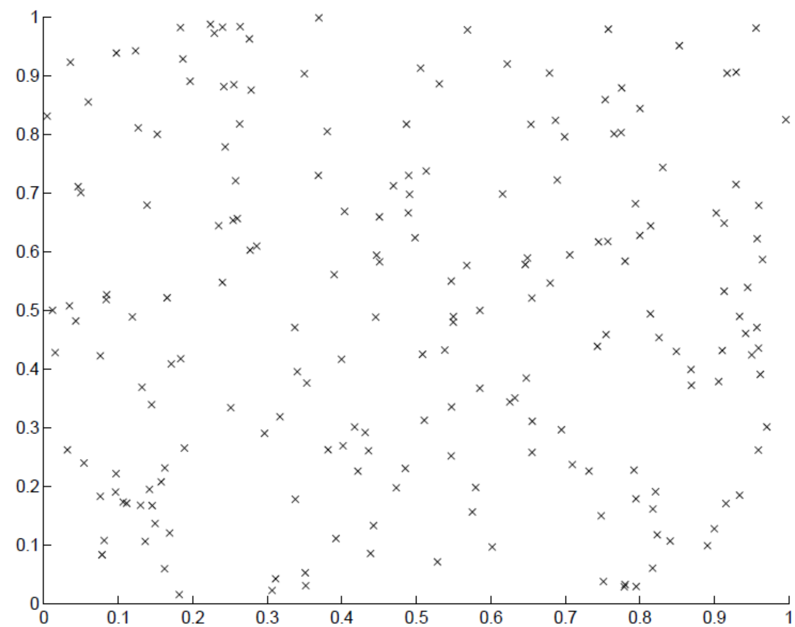
- Hypothesis $h \in H$ that best approximates target function f

Supervised vs. unsupervised learning

- Clustering is an example of unsupervised learning
- We are not given examples of classes y
- Instead we have to discover classes in data

2 datasets with very different underlying structure!





The K-Means Algorithm

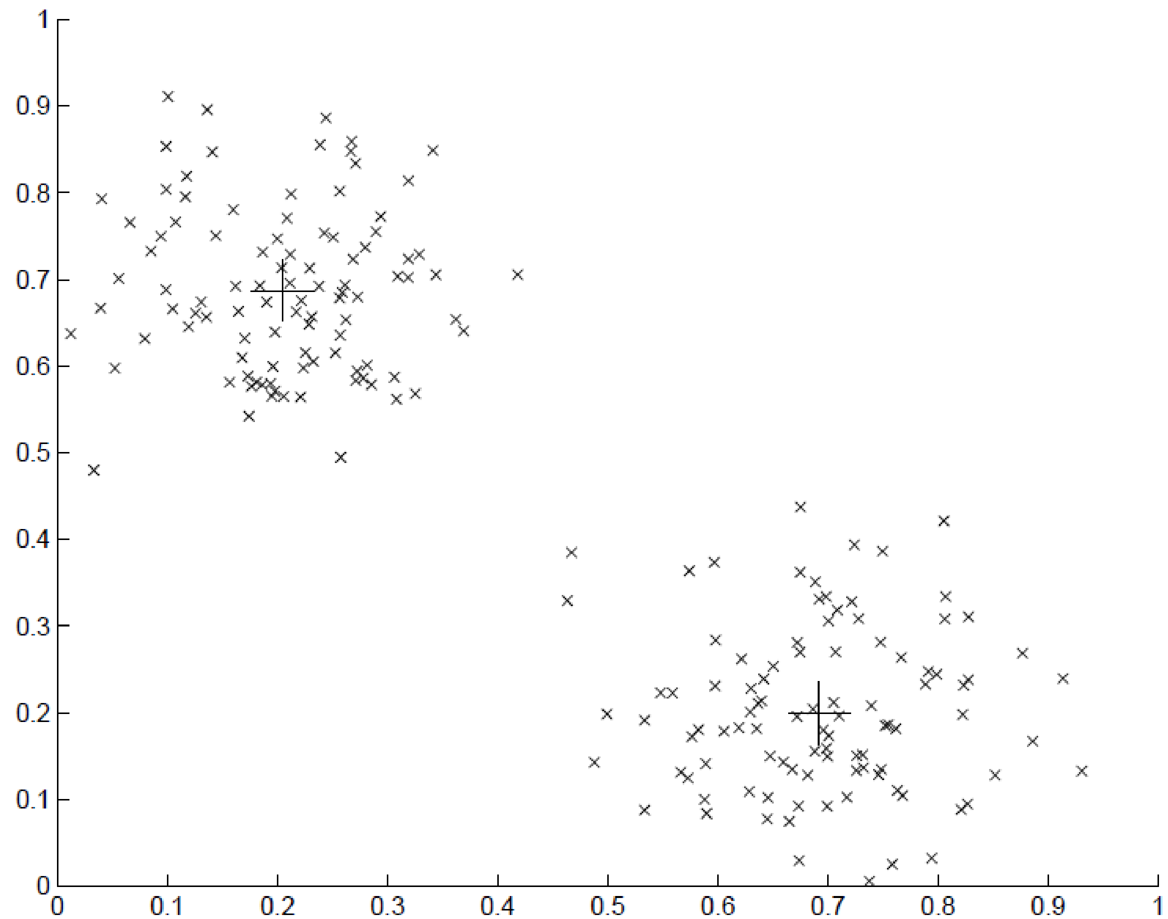
Training Data

K: number of clusters to discover

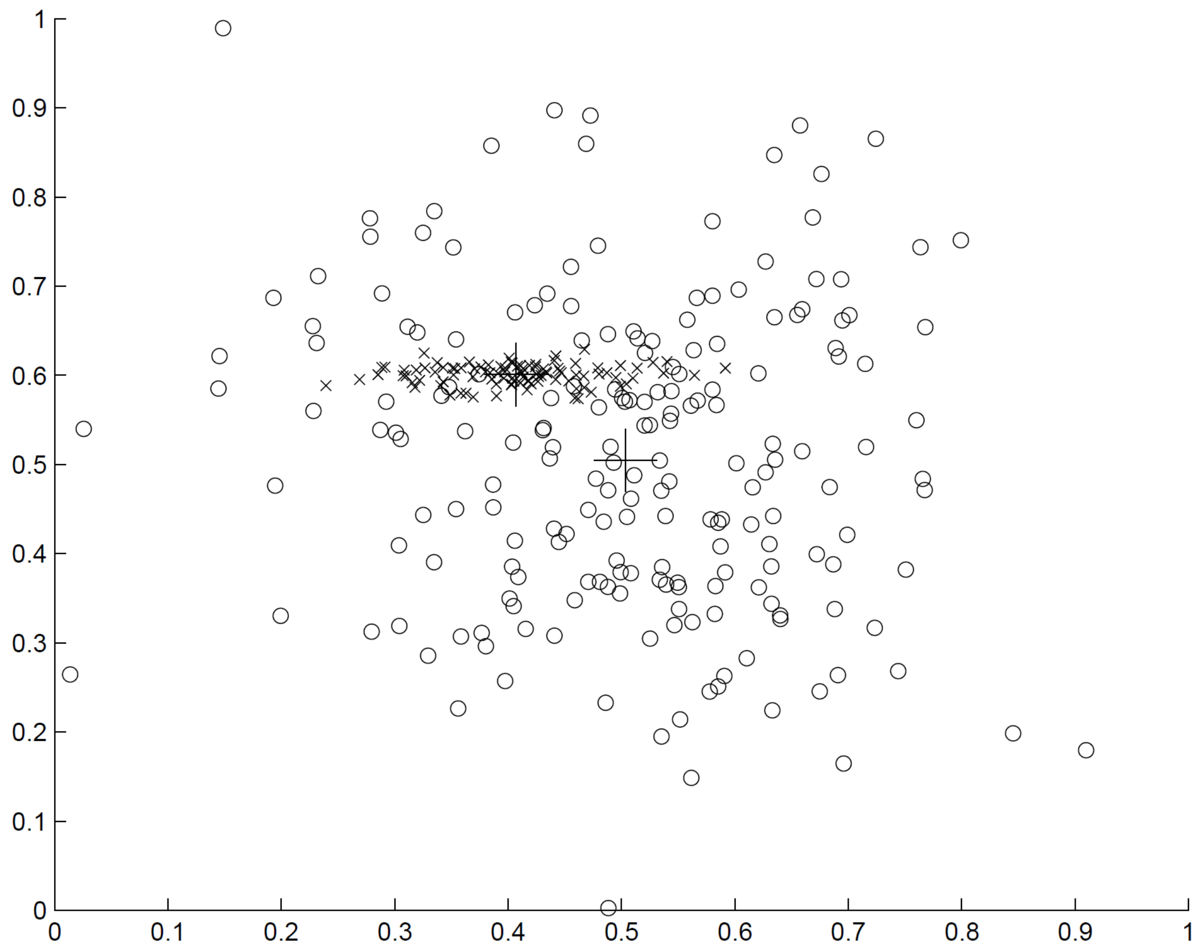
Algorithm 4 K-MEANS($\mathbf{D}$, K)

for $k = 1$ **to** K **do**
    $\boldsymbol{\mu}_k \leftarrow$ some random location   // randomly initialize mean for kth cluster
end for
repeat
    for $n = 1$ **to** N **do**
        $z_n \leftarrow \mathrm{argmin}_k \, ||\boldsymbol{\mu}_k - \boldsymbol{x}_n||$ // assign example n to closest center
    end for
    for $k = 1$ **to** K **do**
        $\mathbf{X}_k \leftarrow \{ \boldsymbol{x}_n : z_n = k \}$ // points assigned to cluster k
        $\boldsymbol{\mu}_k \leftarrow$ MEAN($\mathbf{X}_k$)   // re-estimate mean of cluster k
    end for
until $\boldsymbol{\mu}$s stop changing
return $\boldsymbol{z}$ // return cluster assignments

Example: using K-Means to discover 2 clusters in data



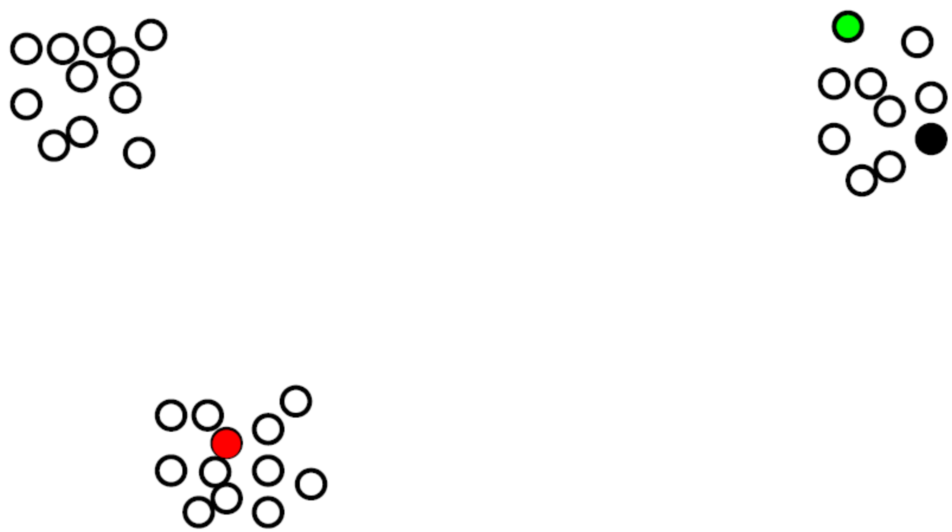
Example: using K-Means to discover 2 clusters in data



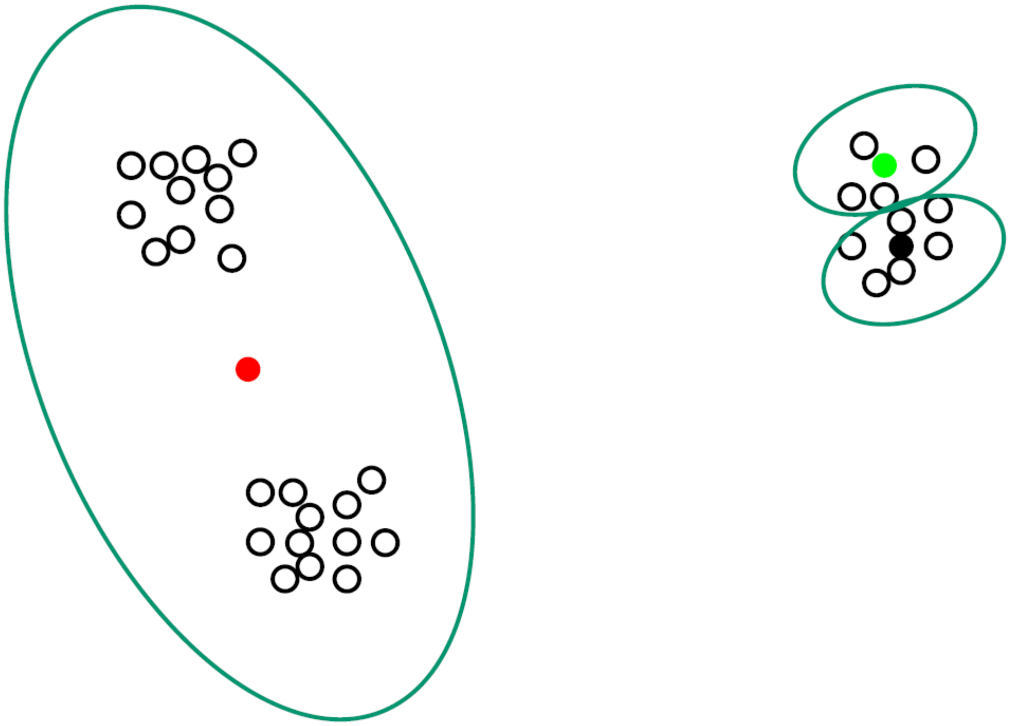
K-Means properties

- Time complexity: O(KNL) where
  - K is the number of clusters
  - N is number of examples
  - L is the number of iterations
- K is a hyperparameter
  - Needs to be set in advance (or learned on dev set)
- Different initializations yield different results!
  - Doesn't necessarily converge to best partition
- "Global" view of data: revisits all examples at every iteration

Impact of initialization



Impact of initialization



Questions for you…

- Are there clusters that cannot be discovered using k-means?

- Do you know any other clustering algorithms?

Aside: Curse of dimensionality

- Challenges of working with high dimensional spaces
  - Hard to visualize
  - Computational cost
  - Many of our intuitions about 2D or 3D spaces don't hold
    - High dimensional hyperspheres "look more like porcupines than balls"
    - Distances between two random points in high dimensions are approximately the same

(CIML Section 3.5 + HW #3)

What you should know

- New Algorithms
  - K-NN classification
  - K-means clustering

- Fundamental ML concepts
  - How to draw decision boundaries
  - What decision boundaries tells us about the underlying classifiers
  - The difference between supervised and unsupervised learning