# K-Means: an example of unsupervised learning

CMSC 422

SOHEIL FEIZI

sfeizi@cs.umd.edu

Slides adapted from SAMUEL DOOLEY and MARINE CARPUAT

# Recap

- Decision Trees
  - Top-down induction to minimize classification error
- Nearest Neighbors (NN) algorithms for classification
  - K-NN, Epsilon ball NN
  - Take a geometric view of learning
- Fundamental Machine Learning Concepts
  - Decision boundary
    - Visualizes predictions over entire feature space
    - Characterizes complexity of learned model

# Exercise: When are DT vs kNN appropriate?

| **Properties of classification problem** | **Can Decision Trees handle them?** | **Can K-NN handle them?** |
|---|---|---|
| Binary features | | |
| Numeric features | | |
| Categorical features | | |
| Robust to noisy training examples | | |
| Fast classification is crucial | | |
| Many irrelevant features | | |
| Relevant features have very different scale | | |

# 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

# 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

# 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?

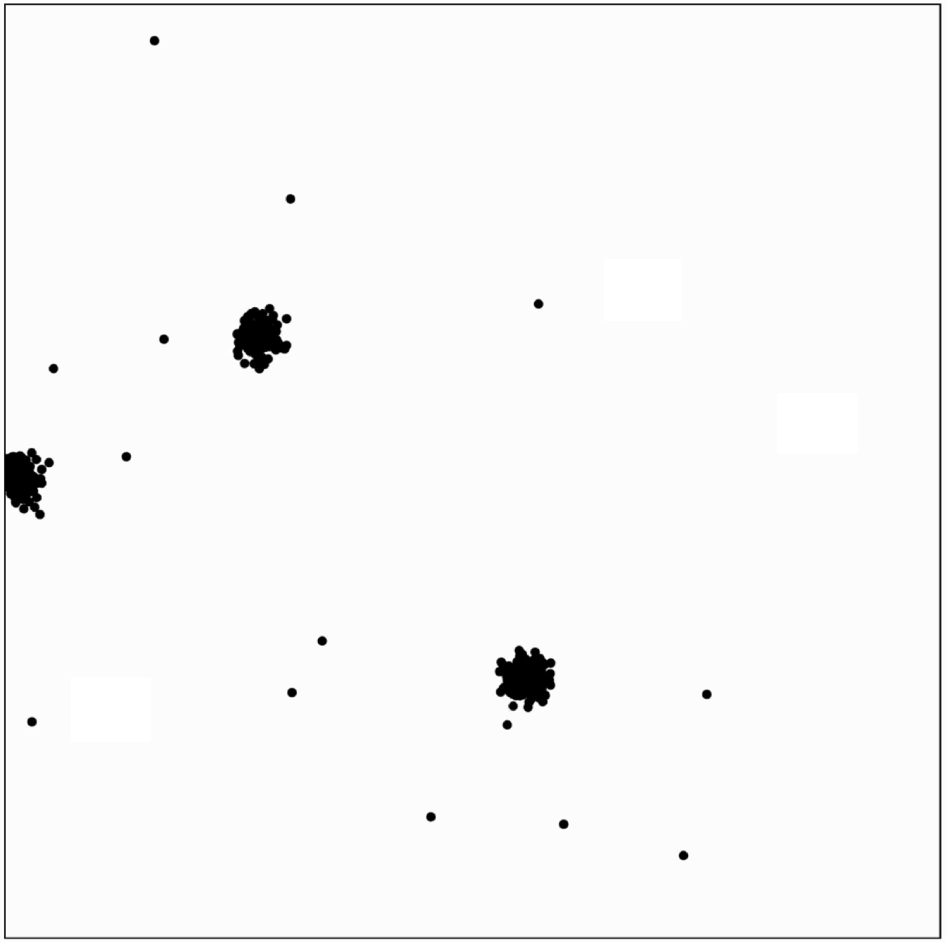
- news articles or web pages by topic
- protein sequences by function, or genes according to expression profile
- users of social networks by interest
- customers according to purchase history
- …

# Clustering

- Input
  - a set S= $\{x_1, x_2, \ldots, x_n\}$ 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
  - Also represented as $\{z_1, \ldots, z_n\}$ where $z_i$ is the index of the partition to which $x_i$ belongs.

# Illustration of Algorithm

Steps of Algorithm
- Gather data

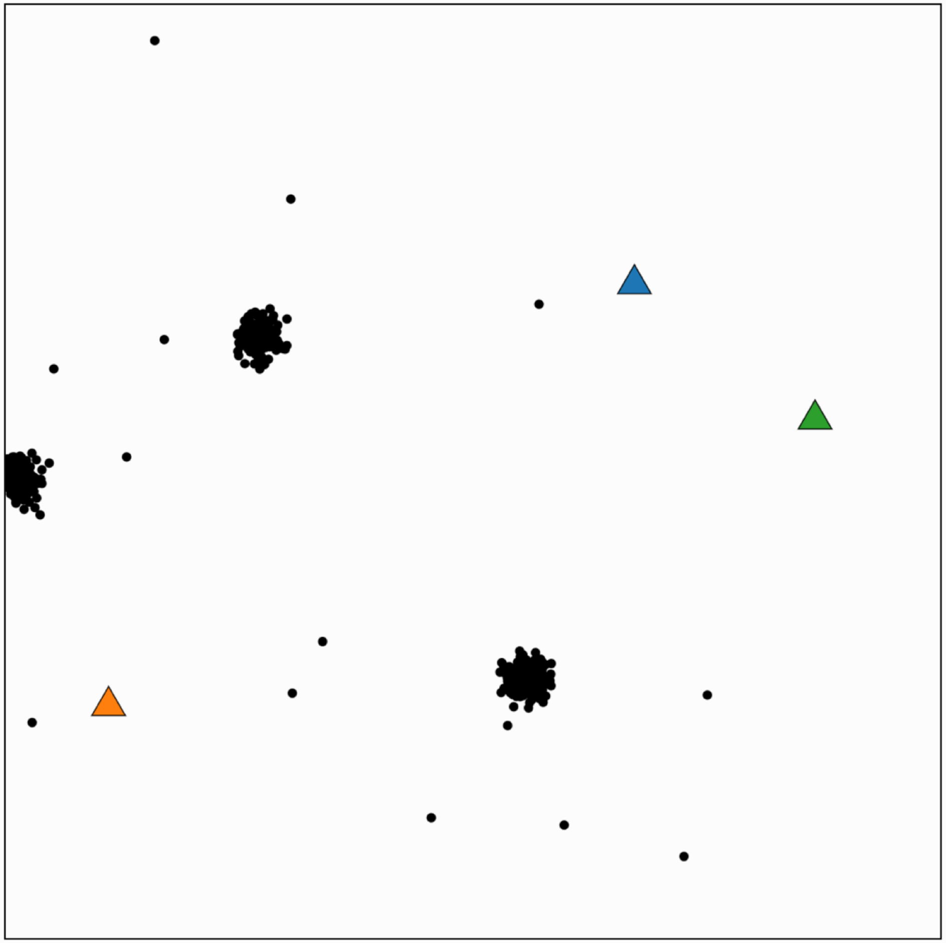


Visualization from: http://stanford.edu/class/ee103/visualizations/kmeans/kmeans.html

# Illustration of Algorithm

Steps of Algorithm

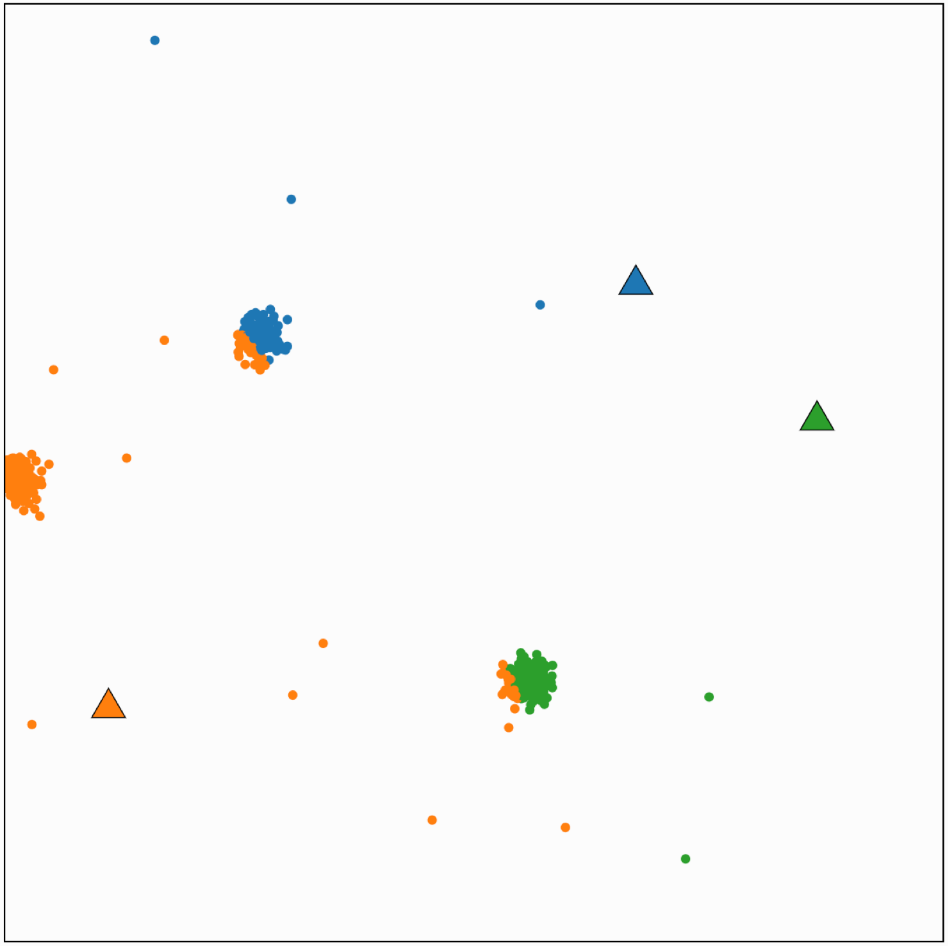
- Gather data
- Initialize means



Visualization from: http://stanford.edu/class/ee103/visualizations/kmeans/kmeans.html

# Illustration of Algorithm

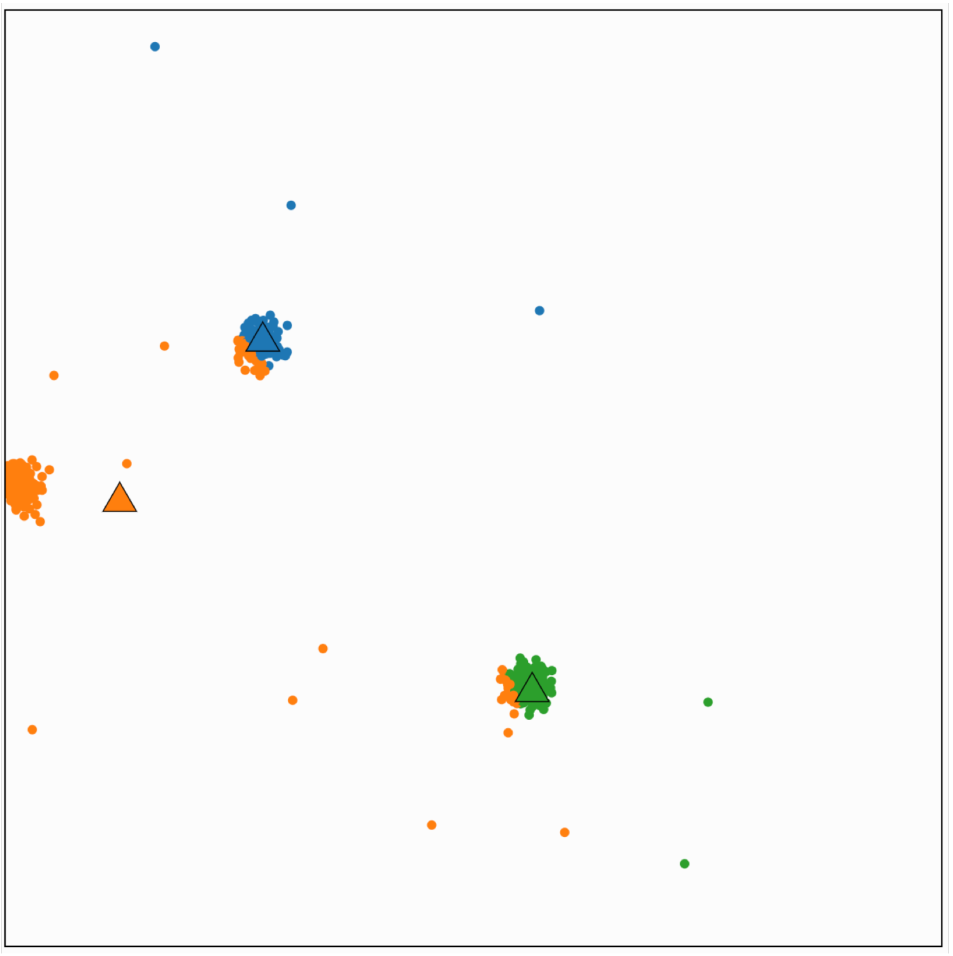
Steps of Algorithm
- Gather data
- Initialize means
- Repeat:
  - Assign classes



Visualization from: http://stanford.edu/class/ee103/visualizations/kmeans/kmeans.html

# Illustration of Algorithm

Steps of Algorithm
- Gather data
- Initialize means
- Repeat:
  - Assign classes
  - Adjust means

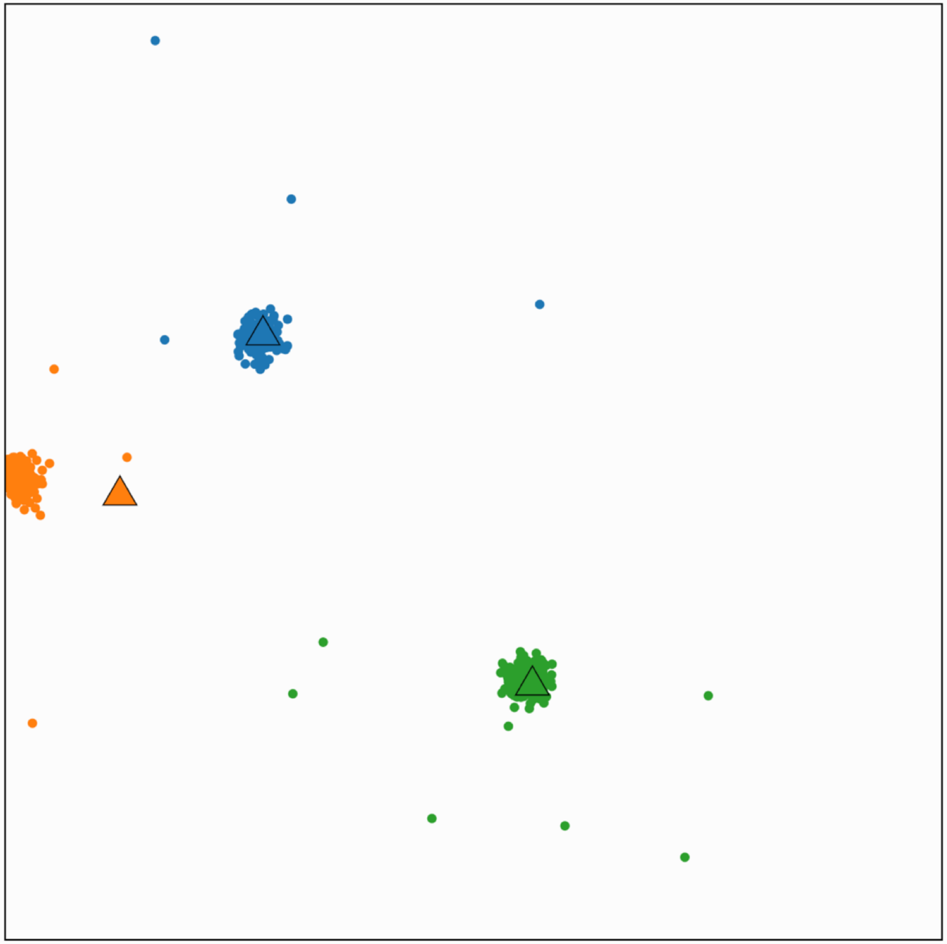


Visualization from: http://stanford.edu/class/ee103/visualizations/kmeans/kmeans.html

# Illustration of Algorithm

Steps of Algorithm

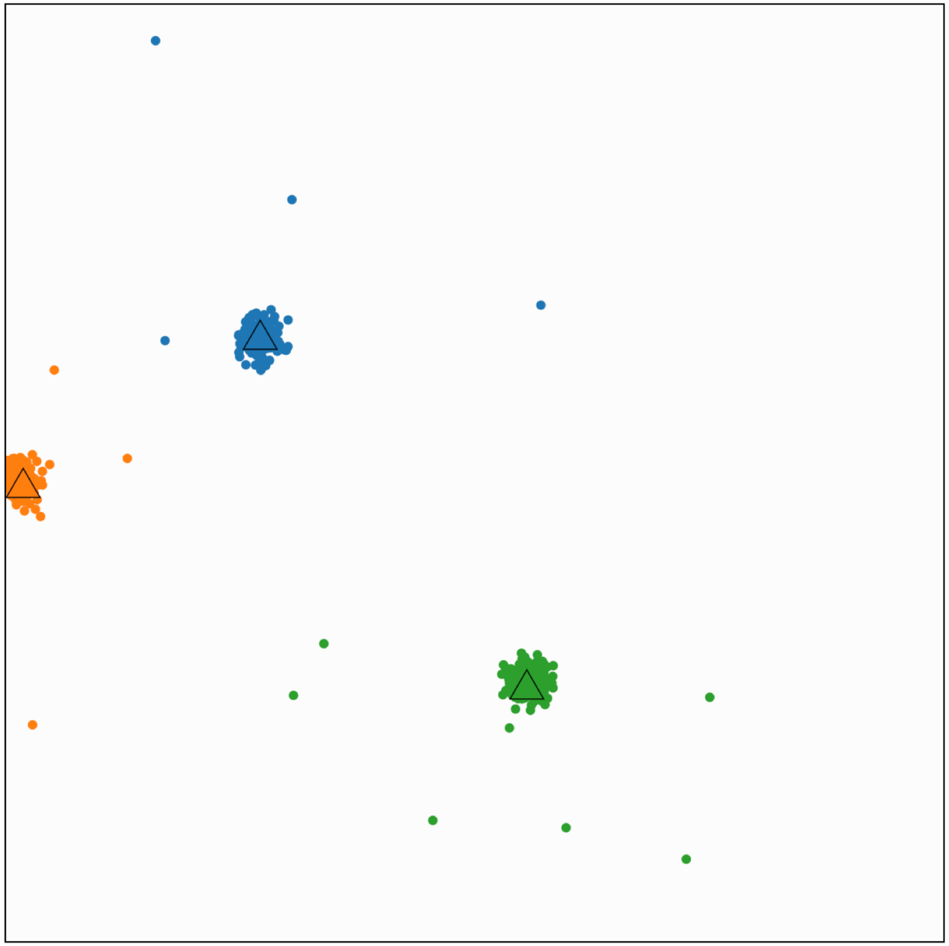
- Gather data
- Initialize means
- Repeat:
  - Assign classes
  - Adjust means



Visualization from: http://stanford.edu/class/ee103/visualizations/kmeans/kmeans.html

# Illustration of Algorithm

Steps of Algorithm
- Gather data
- Initialize means
- Repeat:
  - Assign classes
  - Adjust means



Visualization from: http://stanford.edu/class/ee103/visualizations/kmeans/kmeans.html

# 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 $k$th cluster
**end for**
**repeat**
    **for** $n = 1$ **to** $N$ **do**
        $z_n \leftarrow \text{argmin}_k \, ||\boldsymbol{\mu}_k - \boldsymbol{x}_n||$ // 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 $k$
        $\boldsymbol{\mu}_k \leftarrow \text{MEAN}(\mathbf{X}_k)$ // re-estimate mean of cluster $k$
    **end for**
**until** $\boldsymbol{\mu}$s stop changing
**return** $z$ // return cluster assignments

# The K-Means Optimization

- Assignment objective: For each *i,*

$$\arg\min_{z_i} \|x_i - \mu_{z_i}\|^2$$

- Mean update objective:

$$\min_{\mu_j} \sum_{\{i|z_i=j\}} \|x_i - \mu_{z_i}\|^2$$

- Overall objective:

$$\arg\min_{S} \sum_{i=1}^{k} \sum_{x \in S_i} \|x - \mu_i\|^2 = \arg\min_{S} \sum_{i=1}^{k} |S_i| Var S_i$$

where $\mu_i$ is the mean of the points in $S_i$.

# 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

# Questions for you…

- For what types of data can we not use k-means?
- Are we sure it will find an optimal clustering?
- Does the initialization of the random means impact the result?
- Are there clusters that cannot be discovered using k-means?
- Do you know any other clustering algorithms?

# 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