K-Means
an example of
unsupervised learning

CMSC 422
MARINE CARPUAT
marine@cs.umd.edu
When applying a learning algorithm, some things are properties of the problem you are trying to solve, and some things are up to you to choose as the ML programmer.

Which of the following are properties of the problem?

- The data generating distribution
- The train/dev/test split
- The learning model
- The loss function
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
- galaxies or nearby stars
- ...
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(D, K)**

1. for $k = 1$ to $K$ do
2. \( \mu_k \leftarrow \text{some random location} \) \hspace{1cm} // randomly initialize mean for $k$th cluster
3. end for

4. repeat
5. for $n = 1$ to $N$ do
6. \( z_n \leftarrow \arg \min_k ||\mu_k - x_n|| \) \hspace{1cm} // assign example $n$ to closest center
7. end for
8. for $k = 1$ to $K$ do
9. \( X_k \leftarrow \{ x_n : z_n = k \} \) \hspace{1cm} // points assigned to cluster $k$
10. \( \mu_k \leftarrow \text{MEAN}(X_k) \) \hspace{1cm} // re-estimate mean of cluster $k$
11. end for
12. until \( \mu \)s stop changing
13. return \( z \) \hspace{1cm} // 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...

• Can you think of clusters that cannot be discovered using k-means?

• Do you know any other clustering algorithms?
Aside: High Dimensional Spaces are Weird

• High dimensional spheres look more like porcupines than balls

• Distances between two random points in high dimensions are approximately the same

(CIML Section 2.5)
Exercise: When are DT vs kNN appropriate?

<table>
<thead>
<tr>
<th>Properties of classification problem</th>
<th>Can Decision Trees handle them?</th>
<th>Can K-NN handle them?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Binary features</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td>Numeric features</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td>Categorical features</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td>Robust to noisy training examples</td>
<td>no (for default algorithm)</td>
<td>yes (when k &gt; 1)</td>
</tr>
<tr>
<td>Fast classification is crucial</td>
<td>yes</td>
<td>no</td>
</tr>
<tr>
<td>Many irrelevant features</td>
<td>yes</td>
<td>no</td>
</tr>
<tr>
<td>Relevant features have very different scale</td>
<td>yes</td>
<td>no</td>
</tr>
</tbody>
</table>
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