Unsupervised Learning: Clustering

Vibhav Gogate
The University of Texas at Dallas

Slides adapted from Carlos Guestrin, Dan Klein & Luke Zettlemoyer
Machine Learning

Supervised Learning

Parametric

Y Continuous

1. Gaussians
   Learned in closed form

2. Linear Functions
   1. Learned in closed form
   2. Using gradient descent

Non-parametric

Y Discrete

Unsupervised Learning

Reinforcement Learning

Decision Trees
Greedy search; pruning
Probability of class | features
1. Learn $P(Y)$, $P(X|Y)$; apply Bayes
2. Learn $P(Y|X)$ w/ gradient descent
Non-probabilistic
Linear: perceptron gradient descent
Nonlinear: neural net: backprop
Support vector machines
### Overview of Learning

#### Type of Supervision
(eg, Experience, Feedback)

<table>
<thead>
<tr>
<th>What is Being Learned?</th>
<th>Labeled Examples</th>
<th>Reward</th>
<th>Nothing</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Discrete Function</strong></td>
<td>Classification</td>
<td></td>
<td><strong>Clustering PCA</strong></td>
</tr>
<tr>
<td><strong>Continuous Function</strong></td>
<td>Regression</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Policy</strong></td>
<td>Apprenticeship Learning</td>
<td>Reinforcement Learning</td>
<td></td>
</tr>
</tbody>
</table>
Key Perspective on Learning

• Learning as Optimization
  – Closed form
  – Greedy search
  – Gradient ascent

• Loss Function
  – Error + regularization
Clustering systems:

- Unsupervised learning
- Requires data, but no labels
- Detect patterns eg in
  - Group emails or search results
  - Customer shopping patterns
  - Program executions (intrusion detection)
- Useful when don’t know what you’re looking for
- But: often get gibberish
Clustering

• Basic idea: group together similar instances
• Example: 2D point patterns

What could “similar” mean?
– One option: small (squared) Euclidean distance

$$\text{dist}(x, y) = (x - y)^T (x - y) = \sum_i (x_i - y_i)^2$$
Outline

• K-means & Agglomerative Clustering
• Expectation Maximization (EM)
• Principle Component Analysis (PCA)
K-Means

• An iterative clustering algorithm
  – Pick K random points as cluster centers (means)
  – Alternate:
    • Assign data instances to closest cluster center
    • Change the cluster center to the average of its assigned points
  – Stop when no points’ assignments change
K-Means

• An iterative clustering algorithm
  – Pick K random points as cluster centers (means)
  – Alternate:
    • Assign data instances to closest cluster center
    • Change the cluster center to the average of its assigned points
  – Stop when no points’ assignments change
K-means clustering: Example

- Pick K random points as cluster centers (means)
K-means clustering: Example

Iterative Step 1
- Assign data instances to closest cluster center
K-means clustering: Example

Iterative Step 2
• Change the cluster center to the average of the assigned points
K-means clustering: Example

- Repeat until convergence
K-means clustering: Example
K-means clustering: Example
K-means clustering: Example
Goal of Segmentation is to partition an image into regions each of which has reasonably homogenous visual appearance.
Example: K-Means for Segmentation

K=2

K=3

Original

Original
Example: K-Means for Segmentation

K=2

K=3

K=10

Original

4%

8%

17%
K-Means as Optimization

• Consider the total distance to the means:

\[ \phi(\{x_i\}, \{a_i\}, \{c_k\}) = \sum_i \text{dist}(x_i, c_{a_i}) \]

• Two stages each iteration:
  – Update assignments: fix means c, change assignments a
  – Update means: fix assignments a, change means c

• Co-ordinate Gradient Descent

• Will it converge?
  – Yes!, if you can argue that each update can’t increase \( \Phi \)
Phase I: Update Assignments

• For each point, re-assign to closest mean:

\[ a_i = \arg\min_k \text{dist}(x_i, c_k) \]

• Can only decrease total distance \( \phi \! \):

\[ \phi(\{x_i\}, \{a_i\}, \{c_k\}) = \sum_i \text{dist}(x_i, c_{a_i}) \]
Phase II: Update Means

• Move each mean to the average of its assigned points:

$$ c_k = \frac{1}{|\{i : a_i = k\}|} \sum_{i: a_i = k} x_i $$

• Also can only decrease total distance… (Why?)

• Fun fact: the point \( y \) with minimum squared Euclidean distance to a set of points \( \{x\} \) is their mean
Initialization

• **K-means is non-deterministic**
  - Requires initial means
  - It does matter what you pick!
  - What can go wrong?

  - Various schemes for preventing this kind of thing: variance-based split / merge, initialization heuristics
K-Means Getting Stuck

A local optimum:
K-Means Questions

• Will K-means converge?
  – To a global optimum?

• Will it always find the true patterns in the data?
  – If the patterns are very very clear?

• Runtime?

• Do people ever use it?

• How many clusters to pick?
Agglomerative Clustering

• **Agglomerative clustering:**
  – First merge very similar instances
  – Incrementally build larger clusters out of smaller clusters

• **Algorithm:**
  – Maintain a set of clusters
  – Initially, each instance in its own cluster
  – Repeat:
    • Pick the two closest clusters
    • Merge them into a new cluster
    • Stop when there’s only one cluster left

• Produces not one clustering, but a family of clusterings represented by a **dendrogram**
Agglomerative Clustering

• How should we define “closest” for clusters with multiple elements?
Agglomerative Clustering

• How should we define “closest” for clusters with multiple elements?
  
  • Many options:
    – Closest pair (single-link clustering)
    – Farthest pair (complete-link clustering)
    – Average of all pairs
    – Ward’s method (min variance, like k-means)

• Different choices create different clustering behaviors
Clustering Behavior

Average

Farthest

Nearest

Mouse tumor data from [Hastie]
Agglomerative Clustering Questions

• Will agglomerative clustering converge?
  – To a global optimum?

• Will it always find the true patterns in the data?

• Do people ever use it?

• How many clusters to pick?
Soft Clustering

• Clustering typically assumes that each instance is given a “hard” assignment to exactly one cluster.
• Does not allow uncertainty in class membership or for an instance to belong to more than one cluster.
• *Soft clustering* gives probabilities that an instance belongs to each of a set of clusters.
• Each instance is assigned a probability distribution across a set of discovered categories (probabilities of all categories must sum to 1).
Expectation Maximization (EM)

• Probabilistic method for soft clustering.
• Direct method that assumes $k$ clusters: $\{c_1, c_2, \ldots, c_k\}$
• Soft version of $k$-means.
• Assumes a probabilistic model of categories that allows computing $P(c_i \mid E)$ for each category, $c_i$, for a given example, $E$.
• For text, typically assume a naïve-Bayes category model.
  – Parameters $\theta = \{P(c_i), P(w_j \mid c_i): i \in \{1, \ldots, k\}, j \in \{1, \ldots, |V|\}\}$
EM Algorithm

• Iterative method for learning probabilistic categorization model from unsupervised data.
• Initially assume random assignment of examples to categories.
• Learn an initial probabilistic model by estimating model parameters $\theta$ from this randomly labeled data.
• Iterate following two steps until convergence:
  – Expectation (E-step): Compute $P(c_i \mid E)$ for each example given the current model, and probabilistically re-label the examples based on these posterior probability estimates.
  – Maximization (M-step): Re-estimate the model parameters, $\theta$, from the probabilistically re-labeled data.
Acknowledgements

• K-means & Gaussian mixture models presentation contains material from excellent tutorial by Andrew Moore:
  – http://www.autonlab.org/tutorials/

• K-means Applet:
  – http://www.elet.polimi.it/upload/matteucc/Clustering/tutorial_html/AppletKM.html

• Gaussian mixture models Applet:
  – http://www.neurosci.aist.go.jp/%7Eakaho/MixtureEM.html