Lecture 22:
- Clustering
- Distance measures
- K-Means
Last time... Boosting

- **Idea:** given a weak learner, run it multiple times on (rewighted) training data, then let the learned classifiers vote

- On each iteration $t$:
  - weight each training example by how incorrectly it was classified
  - Learn a hypothesis – $h_t$
  - A strength for this hypothesis – $a_t$

- Final classifier:
  - A linear combination of the votes of the different classifiers weighted by their strength $H(X) = \text{sign} \left( \sum a_t h_t(X) \right)$

- Practically useful
- Theoretically interesting
Last time.. The AdaBoost Algorithm

0) Set $\tilde{W}_i^{(0)} = 1/n$ for $i = 1, \ldots, n$

1) At the $m^{th}$ iteration we find (any) classifier $h(x; \hat{\theta}_m)$ for which the weighted classification error $\epsilon_m$

$$\epsilon_m = 0.5 - \frac{1}{2} \left( \sum_{i=1}^{n} \tilde{W}_i^{(m-1)} y_i h(x_i; \hat{\theta}_m) \right)$$

is better than chance.

2) The new component is assigned votes based on its error:

$$\hat{\alpha}_m = 0.5 \log \left( \frac{(1 - \epsilon_m)}{\epsilon_m} \right)$$

3) The weights are updated according to ($Z_m$ is chosen so that the new weights $\tilde{W}_i^{(m)}$ sum to one):

$$\tilde{W}_i^{(m)} = \frac{1}{Z_m} \cdot \tilde{W}_i^{(m-1)} \cdot \exp \{ -y_i \hat{\alpha}_m h(x_i; \hat{\theta}_m) \}$$
This week

• Clustering
• Distance measures
• K-Means
• Spectral clustering
• Hierarchical clustering
• What is a good clustering?
Distance measures
Distance measures

• In studying clustering techniques we will assume that we are given a matrix of distances between all pairs of data points:

```
<table>
<thead>
<tr>
<th></th>
<th>X_1</th>
<th>X_2</th>
<th>X_3</th>
<th>X_4</th>
</tr>
</thead>
<tbody>
<tr>
<td>X_1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X_2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X_3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X_4</td>
<td>*</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X_m</td>
<td>*</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```

d(x_i, x_j)
What is Similarity/Dissimilarity?

- The real meaning of similarity is a philosophical question. We will take a more pragmatic approach.
- Depends on representation and algorithm. For many rep./alg., easier to think in terms of a distance (rather than similarity) between vectors.
Defining Distance Measures

- **Definition:** Let $O_1$ and $O_2$ be two objects from the universe of possible objects. The distance (dissimilarity) between $O_1$ and $O_2$ is a real number denoted by $D(O_1, O_2)$.
A few examples:

- **Euclidean distance**
  \[
  d(x, y) = \sqrt{\sum_i (x_i - y_i)^2}
  \]

- **Correlation coefficient**
  \[
  s(x, y) = \frac{\sum_i (x_i - \mu_x)(y_i - \mu_y)}{\sigma_x \sigma_y}
  \]
  - Similarity rather than distance
  - Can determine similar trends
What properties should a distance measure have?

- **Symmetric**
  - \( D(A,B) = D(B,A) \)
  - Otherwise, we can say \( A \) looks like \( B \) but \( B \) does not look like \( A \)

- **Positivity, and self-similarity**
  - \( D(A,B) \geq 0 \), and \( D(A,B) = 0 \) iff \( A = B \)
  - Otherwise there will different objects that we cannot tell apart

- **Triangle inequality**
  - \( D(A,B) + D(B,C) \geq D(A,C) \)
  - Otherwise one can say “\( A \) is like \( B \), \( B \) is like \( C \), but \( A \) is not like \( C \) at all”
Distance measures

• Euclidean (L_2)
  \[ d(x, y) = \sqrt{\sum_{i=1}^{d} (x_i - y_i)^2} \]

• Manhattan (L_1)
  \[ d(x, y) = |x - y| = \sum_{i=1}^{d} |x_i - y_i| \]

• Infinity (Sup) Distance L_∞
  \[ d(x, y) = \max_{1 \leq i \leq d} |x_i - y_i| \]

• Note that L_∞ < L_1 < L_2, but different distances do not induce the same ordering on points.
Distance measures

\[ \mathbf{x} = (x_1, x_2) \]
\[ \mathbf{y} = (x_1 - 2, x_2 + 4) \]

Euclidean: \( (4^2 + 2^2)^{1/2} = 4.47 \)
Manhattan: \( 4 + 2 = 6 \)
Sup: \( \text{Max}(4, 2) = 4 \)
Distance measures

- Different distances do not induce the same ordering on points

\[
L_\infty (a, b) = 5
\]
\[
L_2 (a, b) = (5^2 + \varepsilon^2)^{1/2} = 5 + \varepsilon
\]
\[
L_\infty (c, d) = 4
\]
\[
L_2 (c, d) = (4^2 + 4^2)^{1/2} = 4\sqrt{2} = 5.66
\]

\[
L_\infty (c, d) < L_\infty (a, b)
\]
\[
L_2 (c, d) > L_2 (a, b)
\]
Distance measures

• Clustering is sensitive to the distance measure.

• Sometimes it is beneficial to use a distance measure that is invariant to transformations that are natural to the problem:
  - Mahalanobis distance:
    ✓ Shift and scale invariance
Mahalanobis Distance

\[ d(x, y) = \sqrt{(x - y)^T \Sigma (x - y)} \]

\( \Sigma \) is a (symmetric) Covariance Matrix:

\[
\mu = \frac{1}{m} \sum_{i=1}^{m} x_i, \text{ (average of the data)}
\]

\[
\Sigma = \frac{1}{m} \sum_{i=1}^{m} (x - \mu)(x - \mu)^T, \text{a matrix of size } m \times m
\]

Translates all the axes to a mean = 0 and variance = 1 (shift and scale invariance)
Distance measures

• Some algorithms require distances between a point x and a set of points A \( d(x, A) \)
  This might be defined e.g. as min/max/avg distance between x and any point in A.

• Others require distances between two sets of points A, B, \( d(A, B) \).
  This might be defined e.g as min/max/avg distance between any point in A and any point in B.
Clustering algorithms

- **Partitioning algorithms**
  - Construct various partitions and then evaluate them by some criterion
    - K-means
    - Mixture of Gaussians
    - Spectral Clustering

- **Hierarchical algorithms**
  - Create a hierarchical decomposition of the set of objects using some criterion
    - Bottom-up – agglomerative
    - Top-down – divisive
Desirable Properties of a Clustering Algorithm

• Scalability (in terms of both time and space)
• Ability to deal with different data types
• Minimal requirements for domain knowledge to determine input parameters
• Ability to deal with noisy data
• Interpretability and usability

• Optional
  - Incorporation of user-specified constraints
K-Means
K-Means

- **An iterative clustering algorithm**
  - **Initialize:** Pick $K$ random points as cluster centers (means)
  - **Alternate:**
    - Assign data instances to closest mean
    - Assign each mean to the average of its assigned points
  - **Stop when no points’ assignments change**
K-Means

• An iterative clustering algorithm

- Initialize: Pick \( K \) random points as cluster centers (means)

- Alternate:
  • Assign data instances to closest mean
  • Assign each mean 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)

Shown here for $K=2$
K-Means Clustering: Example

Iterative Step 1
- Assign data points to closest cluster centers
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
Properties of K-Means Algorithms

• Guaranteed to converge in a finite number of iterations

• Running time per iteration:
  1. Assign data points to closest cluster center
     \( O(KN) \) time
  2. Change the cluster center to the average of its assigned points
     \( O(N) \) time
K-Means Convergence

Objective

\[ \min_{\mu} \min_{C} \sum_{i=1}^{k} \sum_{x \in C_i} |x - \mu_i|^2 \]

1. Fix \( \mu \), optimize \( C \):

\[ \min_{C} \sum_{i=1}^{k} \sum_{x \in C_i} |x - \mu_i|^2 = \min_{C} \sum_{i} |x_i - \mu_{x_i}|^2 \]

2. Fix \( C \), optimize \( \mu \):

\[ \min_{\mu} \sum_{i=1}^{k} \sum_{x \in C_i} |x - \mu_i|^2 \]

- Take partial derivative of \( \mu_i \) and set to zero, we have

\[ \mu_i = \frac{1}{|C_i|} \sum_{x \in C_i} x \]

K-Means takes an alternating optimization approach, each step is guaranteed to decrease the objective – thus guaranteed to converge.
Demo time...
Example: K-Means for Segmentation

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

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  K=10  Original

Goal of Segmentation is to partition an image into regions each of which has reasonably homogenous visual appearance.
Example: Vector quantization

**FIGURE 14.9.** Sir Ronald A. Fisher (1890 – 1962) was one of the founders of modern day statistics, to whom we owe maximum-likelihood, sufficiency, and many other fundamental concepts. The image on the left is a 1024×1024 grayscale image at 8 bits per pixel. The center image is the result of 2×2 block VQ, using 200 code vectors, with a compression rate of 1.9 bits/pixel. The right image uses only four code vectors, with a compression rate of 0.50 bits/pixel.
Our energy exploration, production, and distribution operations span the globe, with activities in more than 100 countries.

At TOTAL, we draw our greatest strength from our fast-growing oil and gas reserves. Our strategic emphasis on natural gas provides a strong position in a rapidly expanding market.

Our expanding refining and marketing operations in Asia and the Mediterranean Rim complement already solid positions in Europe, Africa, and the U.S.

Our growing specialty chemicals sector adds balance and profit to the core energy business.
Object \rightarrow Bag of ‘words’
Interest Point Features

- **Compute SIFT descriptor**
  
  - [Lowe'99](https://www.cs.ubc.ca/~csivc/papers/lowe_sift.pdf)

- **Detect patches**
  
  - [Mikojaczyk and Schmid '02](https://www.cs.ubc.ca/~csivc/papers/mikojaczyk_schmid.pdf)
  
  - [Matas et al. '02](https://www.cs.ubc.ca/~csivc/papers/matasetal02.pdf)
  
  - [Sivic et al. '03](https://www.cs.ubc.ca/~csivc/papers/sivicetal03.pdf)

- **Normalize patch**
Patch Features
Dictionary Formation
Clustering (usually K-means)

Vector quantization
Clustered Image Patches
Visual synonyms and polysemy

Visual Polysemy. Single visual word occurring on different (but locally similar) parts on different object categories.

Visual Synonyms. Two different visual words representing a similar part of an object (wheel of a motorbike).
Image Representation
K-Means Clustering: Some Issues

- How to set k?
- Sensitive to initial centers
- Sensitive to outliers
- Detects spherical clusters
- Assuming means can be computed