Statistics 202: Data Mining

Week 8
Based in part on slides from textbook, slides of Susan Holmes

©Jonathan Taylor

December 2, 2012
Part I

Clustering
Clustering

- Goal: Finding groups of objects such that the objects in a group will be similar (or related) to one another and different from (or unrelated to) the objects in other groups.
- An unsupervised problem that tries to produce “labelled” data from “unlabelled” data.
- Many different techniques, but most revolve around forming groups with small “within cluster” distances relative to “between cluster” distances.
Cluster analysis

What is Cluster Analysis?

Finding groups of objects such that the objects in a group will be similar (or related) to one another and different from (or unrelated to) the objects in other groups.

Inter-cluster distances are maximized.

Intra-cluster distances are minimized.
Clustering

Applications

- Understanding of some structure in a dataset.
  - Group related documents for browsing;
  - Group genes and proteins that have similar functionality;
  - Group stocks with similar price fluctuations

- Summarization: reduce the size of large data sets
  (sometimes known as vector quantization . . . )
Cluster analysis

FIGURE 14.14. DNA microarray data: average linkage hierarchical clustering has been applied independently to the rows (genes) and columns (samples), determining the ordering of the rows and columns (see text). The colors range from bright green (negative, under-expressed) to bright red (positive, over-expressed).
Cluster analysis

Original image ...
Cluster analysis

Some compression . . .
Cluster analysis

Too much compression?
Cluster analysis

Notion of a Cluster can be Ambiguous

How many clusters?

Four Clusters  
Two Clusters  
Six Clusters

Clusters can be ambiguous.
Clustering

Types of clustering

**Partitional**  A division data objects into non-overlapping subsets (clusters) such that each data object is in exactly one subset.

**Hierarchical**  A set of nested clusters organized as a hierarchical tree. Each data object is in exactly one subset for any horizontal cut of the tree . . .
Cluster analysis

Cluster analysis is also used to form descriptive statistics to ascertain whether or not the data consists of a set distinct subgroups, each group representing objects with substantially different properties. This latter goal requires an assessment of the degree of difference between the objects assigned to the respective clusters.

Central to all of the goals of cluster analysis is the notion of the degree of similarity (or dissimilarity) between the individual objects being clustered. A clustering method attempts to group the objects based on the definition of similarity supplied to it. This can only come from subject matter considerations. The situation is somewhat similar to the specification of a loss or cost function in prediction problems (supervised learning). There the cost associated with an inaccurate prediction depends on considerations outside the data.

Figure 14.4 shows some simulated data clustered into three groups via the popular K-means algorithm. In this case two of the clusters are not well separated, so that “segmentation” more accurately describes the part of this process than “clustering.”

K-means clustering starts with guesses for the three cluster centers. Then it alternates the following steps until convergence:

• for each data point, the closest cluster center (in Euclidean distance) is identified;
Cluster analysis

A hierarchical example
Clustering

Other distinctions

**Exclusivity**  Are points in only one cluster?

**Soft vs. hard**  Can we give a “score” for each case and each cluster?

**Partial vs. complete**  Do we cluster all points, or only some?

**Heterogeneity**  Are the clusters similar in size, shape, etc.
Types of clusters: well-separated

This type of cluster is a set of points such that any point in a cluster is closer (or more similar) to every other point in the cluster than to any point not in the cluster.
Types of clusters: center-based

This type of cluster is a set of objects such that an object in a cluster is closer (more similar) to the “center” of a cluster, than to the center of any other cluster.
Types of clusters: center-based

The center of these clusters is usually, the average of all the points in the cluster, or a medoid, the most “representative” point of a cluster.
Types of clusters: contiguity-based

This type of cluster is a set of points such that a point in a cluster is closer (or more similar) to one or more other points in the cluster than to any point not in the cluster.
Types of clusters: contiguity-based

These types of clusters are made up of dense region of points, often described by one of the other cluster types, but separated by low-density regions, often in the form of noise.
Clustering

Mathematical characterizations

- Most clustering algorithms are based on a dissimilarity measure $d$.
- Data may be of mixed type so some of the similarities we saw earlier may be used.
- Most clustering algorithms do not insist that the dissimilarity is truly a *distance*.
- *(Local)* Given two elements of the partition $C_r, C_s$, we might consider

$$d_{SL}(C_r, C_s) = \min_{x \in C_r, y \in C_s} d(x, y)$$

These types of measures are often used in *hierarchical clustering* algorithms.
Clustering

Mathematical characterizations

- (Global) A clustering is a partition $C = \{C_1, \ldots, C_k\}$ of the cases. It writes

$$T = \frac{1}{2} \sum_{i,j=1}^{n} d^2(x_i, x_j) = W(C) + B(C)$$

and tries to minimize $W(C)$.

- This problem is NP Hard . . .

- These types of measures are often used in *partitional clustering* algorithms.
Clustering

Most common models

Partitional \( K \)-means / medoid ; mixture models

Hierarchical agglomerative (bottom-up) hierarchical clustering.

We’ll spend a some time on each of these . . .
Cluster analysis

$K$-means clustering

$X_1$

$X_2$

$K$-means clustering starts with guesses for the three cluster centers. Then it alternates the following steps until convergence:

• for each data point, the closest cluster center (in Euclidean distance) is identified;
Agglomerative hierarchical clustering

Cluster analysis

FIGURE 14.12. Dendrogram from agglomerative hierarchical clustering with average linkage to the human tumor microarray data.

Hierarchical methods impose hierarchical structure whether or not such structure actually exists in the data. The extent to which the hierarchical structure produced by a dendrogram actually represents the data itself can be judged by the cophenetic correlation coefficient. This is the correlation between the \( \frac{N(N-1)}{2} \) pairwise observation dissimilarities \( d_{ii}' \) input to the algorithm and their corresponding cophenetic dissimilarities \( C_{ii}' \) derived from the dendrogram. The cophenetic dissimilarity \( C_{ii}' \) between two observations \( (i, i') \) is the intergroup dissimilarity at which observations \( i \) and \( i' \) are first joined together in the same cluster. The cophenetic dissimilarity is a very restrictive dissimilarity measure. First, the \( C_{ii}' \) over the observations must contain many ties, since only \( N-1 \) of the total \( \frac{N(N-1)}{2} \) values can be distinct. Also these dissimilarities obey the ultrametric inequality

\[
C_{ii}' \leq \max\{C_{ik}, C_{i'k}\} \quad (14.40)
\]
Part II

*K*-means clustering
$K$-means

Outline

- $K$-means, $K$-medoids
- Choosing the number of clusters: Gap test, silhouette plot.
- Mixture modelling, EM algorithm.
Figure: Simulated data in the plane, clustered into three classes (represented by red, blue and green) by the $K$-means clustering algorithm. From *ESL*. 

\[ x_1 \]
\[ x_2 \]
**K-means**

**Algorithm (Euclidean)**

1. For each data point, the closest cluster center (in Euclidean distance) is identified;
2. Each cluster center is replaced by the coordinatewise average of all data points that are closest to it.
3. Steps 1. and 2. are alternated until convergence. Algorithm converges to a local minimum of the within-cluster sum of squares.

Typically one uses multiple runs from random starting guesses, and chooses the solution with lowest within cluster sum of squares.
**$K$-means**

---

**Non-Euclidean**

1. We can replace the Euclidean distance squared with some other dissimilarity measure $d$, this changes the assignment rule to minimizing $d$. is identified;

2. Each cluster center is replaced by the point that minimizes the sum of all pairwise $d$’s.

3. Steps 1. and 2. are alternated until convergence. Algorithm converges to a local minimum of the within-cluster sum of $d$’s.
Successive iterations of the K-means clustering algorithm for the simulated data of Figure 14.4.
**Figure**: Decrease in $W(C)$, the within cluster sum of squares.
Importance of Choosing Initial Centroids

Figure: Another example of the iterations of $K$-means
**K-means**

Two different K-means Clusterings

- Sub-optimal Clustering
- Optimal Clustering

Original Points

Optimal Clustering

Sub-optimal Clustering
The Iris data ($K$-means)
$K$-means

**Issues to consider**

- Non-quantitative features, e.g. categorical variables, are typically coded by dummy variables, and then treated as quantitative.
- How many centroids $k$ do we use? As $k$ increases, both training and test error decrease!
- By test error, we mean the within-cluster sum of squares for data held-out when fitting the clusters . . .
- Possible to get empty clusters . . .
Choosing $K$

- Ideally, the within cluster sum of squares flattens out quickly and we might choose the value of $K$ at this “elbow”.

- We might also compare the observed within cluster sum of squares to a null model, like uniform on a box containing the data.

- This is the basis of the gap statistic.
**K-means**

**Figure**: Blue curve is the $W_K$ for uniform, green curve is for data.
**K-means**

**Figure**: Largest gap is at 2, and the formal rule also takes into account the variability of estimating the gap.
**Algorithm**

- Same as $K$-means, except that centroid is estimated not by the average, but by the observation having minimum pairwise distance with the other cluster members.

- Advantage: centroid is one of the observations— useful, eg when features are 0 or 1. Also, one only needs pairwise distances for $K$-medoids rather than the raw observations.

- In R, the function `pam` implements this using Euclidean distance (not distance squared).
Example: Country Dissimilarities

This example comes from a study in which political science students were asked to provide pairwise dissimilarity measures for 12 countries.

<table>
<thead>
<tr>
<th></th>
<th>BEL</th>
<th>BRA</th>
<th>CHI</th>
<th>CUB</th>
<th>EGY</th>
<th>FRA</th>
<th>IND</th>
<th>ISR</th>
<th>USA</th>
<th>USS</th>
<th>YUG</th>
</tr>
</thead>
<tbody>
<tr>
<td>BRA</td>
<td>5.58</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CHI</td>
<td>7.00</td>
<td>6.50</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CUB</td>
<td>7.08</td>
<td>7.00</td>
<td>3.83</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EGY</td>
<td>4.83</td>
<td>5.08</td>
<td>8.17</td>
<td>5.83</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FRA</td>
<td>2.17</td>
<td>5.75</td>
<td>6.67</td>
<td>6.92</td>
<td>4.92</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IND</td>
<td>6.42</td>
<td>5.00</td>
<td>5.58</td>
<td>6.00</td>
<td>4.67</td>
<td>6.42</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ISR</td>
<td>3.42</td>
<td>5.50</td>
<td>6.42</td>
<td>6.42</td>
<td>5.00</td>
<td>3.92</td>
<td>6.17</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>USA</td>
<td>2.50</td>
<td>4.92</td>
<td>6.25</td>
<td>7.33</td>
<td>4.50</td>
<td>2.25</td>
<td>6.33</td>
<td>2.75</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>USS</td>
<td>6.08</td>
<td>6.67</td>
<td>4.25</td>
<td>2.67</td>
<td>6.00</td>
<td>6.17</td>
<td>6.17</td>
<td>6.92</td>
<td>6.17</td>
<td></td>
<td></td>
</tr>
<tr>
<td>YUG</td>
<td>5.25</td>
<td>6.83</td>
<td>4.50</td>
<td>3.75</td>
<td>5.75</td>
<td>5.42</td>
<td>6.08</td>
<td>5.83</td>
<td>6.67</td>
<td>3.67</td>
<td></td>
</tr>
<tr>
<td>ZAI</td>
<td>4.75</td>
<td>3.00</td>
<td>6.08</td>
<td>6.67</td>
<td>5.00</td>
<td>5.58</td>
<td>4.83</td>
<td>6.17</td>
<td>5.67</td>
<td>6.50</td>
<td>6.92</td>
</tr>
</tbody>
</table>
Figure: Left panel: dissimilarities reordered and blocked according to 3-medoid clustering. Heat map is coded from most similar (dark red) to least similar (bright red). Right panel: two-dimensional multidimensional scaling plot, with 3-medoid clusters indicated by different colors.
The Iris data: $K$-medoid (PAM)
$K$-medoid

**Silhouette**

- For each case $1 \leq i \leq n$, and set of cases $C$ and dissimilarity $d$ define

\[
\bar{d}(i, C) = \frac{1}{\#C} \sum_{j \in C} d(i, j).
\]

- Each case $1 \leq i \leq n$ is assigned to a cluster $C_{l(i)}$. The silhouette width is defined for each case as

\[
\text{silhouette}(i) = \frac{\min_{j \neq l(i)} \bar{d}(i, C_j) - \bar{d}(i, C_{l(i)})}{\max(\bar{d}(i, C_{l(i)}), \min_{j \neq l(i)} \bar{d}(i, C_j))}.
\]

- High values of silhouette indicate good clusterings.
- In $R$ this is computable for `pam` objects.
The Iris data: silhouette plot for $K$-medoid

Silhouette plot of pam($x = \text{iris[-5]}$, $k = 3$)

3 clusters $C_j$

1: 50 | 0.80
2: 62 | 0.42
3: 38 | 0.45

Average silhouette width: 0.55
The Iris data: average silhouette width
A soft clustering algorithm

- Imagine we actually had labels $Y$ for the cases, then this would be a classification problem.
- For this classification problem, we might consider using a Gaussian discriminant model like LDA or QDA.
- We would then have to estimate $(\mu_j, \Sigma_j)$ within each “cluster.” This would be easy . . .
- The next model is based on this realization . . .
Mixture modelling

**EM algorithm**

- The abbreviation: \( E=\text{expectation}, \ M=\text{maximization} \).
- A special case of an *majorization-minimization* algorithm and widely used throughout statistics.
- Particularly useful for situations in which there might be some hidden data that would make the problem easy . . .
Mixture modelling

**EM algorithm**

- In this mixture model framework, we assume that the data were drawn from the same model as in QDA (or LDA).
  
  \[ Y \sim \text{Multinomial}(1, \pi) \quad \text{(choose a label)} \]
  
  \[ X|Y = \ell \sim N(\mu_\ell, \Sigma_\ell) \]

- Only, we have lost our labels and only observe \( X_{n \times p} \).
- The goal is still the same, to estimate \( \pi, (\mu_\ell, \Sigma_\ell)_{1 \leq \ell \leq k} \).
Mixture modelling

- The algorithm keeps track of $(\mu_\ell, \Sigma_\ell)_{1 \leq \ell \leq k}$
- It also tracks “guesses” at $\mathbf{Y}$ in the form of $\Gamma_{n \times k}$.
- Alternates between “guessing” $\mathbf{Y}$ and estimating $\pi, (\mu_\ell, \Sigma_\ell)_{1 \leq \ell \leq k}$. 

**EM algorithm**
Mixture modelling

**EM algorithm**

Initialize $\Gamma, \mu, \Sigma, \pi$.

Repeat For $1 \leq t \leq T$,

Estimate $\Gamma$ These are called the **responsibilities**

$$\hat{\gamma}_{i\ell}^{(t+1)} = \frac{\hat{\pi}_\ell^{(t)} \phi_{\hat{\mu}_\ell^{(t)}, \hat{\Sigma}_\ell^{(t)}}(X_i)}{\sum_{l=1}^K \hat{\pi}_l^{(t)} \phi_{\hat{\mu}_l^{(t)}, \hat{\Sigma}_l^{(t)}}(X_i)}$$

Estimate $\mu_\ell, 1 \leq k$

$$\hat{\mu}_\ell^{(t+1)} = \frac{\sum_{i=1}^n \hat{\gamma}_{i\ell}^{(t+1)} X_i}{\sum_{i=1}^n \hat{\gamma}_{i\ell}^{(t+1)}}$$

This is just weighted average with weights $\hat{\gamma}_{\ell}^{(t+1)}$. 
Mixture modelling

EM algorithm

Estimate $\Sigma_\ell, 1 \leq k$

$$\hat{\Sigma}^{(t+1)}_\ell = \frac{\sum_{i=1}^n \hat{\gamma}_{i\ell}^{(t+1)}(X_i - \hat{\mu}_{\ell}^{(t+1)})(X_i - \hat{\mu}_{\ell}^{(t+1)})^T}{\sum_{i=1}^n \hat{\gamma}_{i\ell}^{(t+1)}}$$

This is just a weighted estimate of the covariance matrix with weights $\hat{\gamma}_{i\ell}^{(t+1)}$.

Estimate $\pi_\ell$

$$\hat{\pi}_\ell^{(t+1)} = \frac{1}{n} \sum_{i=1}^n \hat{\gamma}_{i\ell}^{(t+1)}$$
Mixture modelling

**EM algorithm**

- The quantities $\Gamma$ are not really parameters, they are “estimates” of the random labels $Y$ which were unobserved.
- If we had observed $Y$ then the rows of $\Gamma$ would be all zero except one entry, which would be 1.
- In this case, estimation of $\pi_\ell, \mu_\ell, \Sigma_\ell$ is just as it would have been in QDA . . .
- The EM simply replaces the unobserved $Y$ with a guess . . .
The Iris data: Gaussian mixture modelling
The Iris data ($K$-means)
The Iris data: silhouette plot for $K$-medoid

Silhouette plot of pam(x = iris[, -5], k = 3)

$n = 150$

3 clusters $C_j$

$\mid n_i \mid \text{avg}_i \in C_j s_i$

1: 50 | 0.80

2: 62 | 0.42

3: 38 | 0.45

Average silhouette width : 0.55