Chapter 9. Clustering Analysis

PubH 5470

April 2004

©Wei Pan

Email: weip@biostat.umn.edu

Http: www.biostat.umn.edu/~weip
Outline

• Introduction

• Hierarchical clustering

• Combinatorial algorithms

• K-means clustering

• K-medoids clustering

• Mixture model-based clustering

• Practical issues
  
  # of clusters, stability of clusters,...
Introduction

• Given: \( X_i = (X_{i1}, \ldots, X_{ip})', \) \( i = 1, \ldots, n \)

• Goal: Cluster or group \( X_i \)'s “similar” to each other together;
Or, predict \( X_i \)'s class \( Y_i \) with no training info on \( Y \)'s.

• Unsupervised learning, class discovery,…

• Ref: 1. textbook, Chap 14;
4. Many many papers...

- Define a metric of distance (or similarity):

\[d(X_i, X_j) = \sum_{k=1}^{p} w_k d_k(X_{ik}, X_{jk})\]

- \(X_{ik}\) quantitative: \(d_k\) can be Euclidean distance, absolute distance, Pearson correlation, etc.

- \(X_{ik}\) ordinal: coded as \((i - 1/2)/M\) (or simply as \(i\)?) for \(i = 1, ..., M\); then treated as quantitative.

- \(X_{ik}\) categorical: specify \(L_{l,m} = d_k(l, m)\) based on subject-matter knowledge; 0-1 loss is commonly used.

- \(w_k = 1\) for all \(k\) commonly used, but it
may not treat each variable (or attribute) equally!
standardize each variable to have \( \text{var}=1 \).
- Distance \( \leftrightarrow \) similarity, e.g. \( \text{sim} = 1 - d \)

Hierarchical Clustering

- A dendrogram (an upside-down tree):
  Leaves represent observations \( X_i \)'s; each sub-tree represents a group/cluster, and the height of the subtree represents the degree of dissimilarity within the group.

- Fig 14.12
• Bottom-up (agglomerative) algorithm

given: a set of observations \( \{X_1, ..., X_n\} \).

for \( i := 1 \) to \( n \) do

\[ c_i := \{Y_i\} \] /* each obs is initially a cluster */

\( C := \{c_1, ..., c_n\} \)

\( j := n + 1 \)

while \( |C| > 1 \)

\[ (c_a, c_b) := \arg \max_{(c_u, c_v)} \text{sim}(c_u, c_v) \]

/* find most similar pair */

\[ c_j := c_a \cup c_b \] /* combine to generate a new cluster */

\( C := [C - \{c_a, c_b\}] \cup c_j \)

\( j := j + 1 \)
• Similarity of two clusters

Similarity of two clusters can be defined in three ways:

– single link: similarity of two most similar members

\[ \text{sim}(C_1, C_2) = \max_{i \in C_1, j \in C_2} \text{sim}(Y_i, Y_j) \]

– complete link: similarity of two least similar members

\[ \text{sim}(C_1, C_2) = \min_{i \in C_1, j \in C_2} \text{sim}(Y_i, Y_j) \]

– average link: average similarity b/w two members

\[ \text{sim}(C_1, C_2) = \text{ave}_{i \in C_1, j \in C_2} \text{sim}(Y_i, Y_j) \]

• R: hclust()
Combinatorial Algorithms

- No probability model; group observations to min/max a criterion

- Clustering: find a mapping $C$: $\{1, 2, \ldots, n\} \rightarrow \{1, \ldots, K\}$, $K < n$

- A criterion

\[
W(C) = \frac{1}{2} \sum_{c=1}^{K} \sum_{C(i)=c} \sum_{C(j)=c} d(X_i, X_j)
\]

- $T = \frac{1}{2} \sum_{i=1}^{K} \sum_{j=1}^{K} d(X_i, X_j) = W(C) + B(C)$,

\[
B(C) = \frac{1}{2} \sum_{c=1}^{K} \sum_{C(i)=c} \sum_{C(j)\neq c} d(X_i, X_j)
\]

- Min $B(C) \leftrightarrow$ Max $W(C)$

- Algorithms: search all possible $C$ to find $C_0 = \text{argmin}_C W(C)$
• Only feasible for small $n$ and $K$: # of possible $C$’s

$$S(n, K) = \frac{1}{K!} \sum_{k=1}^{K} (-1)^{K-k} C(K, k) k^n$$

E.g. $S(10, 4) = 34105$, $S(19, 4) \approx 10^{10}$.

• Alternatives: iterative greedy search!

  K-means Clustering

• Each observation is a point in a $p$-dim space

• Suppose we know/want to have $K$ clusters

• First, (randomly) decide $K$ cluster centers, $M_k$

• Then, iterate the two steps:
- assignment of each obs to a cluster

\[ C(i) = \arg\min_k d(X_i, M_k) \]

- new cluster center is the mean of obs’s in each cluster

\[ M_k = \text{Ave}_{C(i)=k} X_i \]

- Euclidean distance \( d() \) is used

- May stop at a local minimum for \( W(C) \); multiple tries

- R: \text{kmeans()}

- +: simple and intuitive

- -: Euclidean distance \( \rightarrow 1 \) sensitive to outliers; 2) if \( X_{ij} \) is categorical then?
K-medoids Clustering

- Similar to K-means; rather than using the mean of a cluster to represent the cluster, use an observation within it!

- First, (randomly) start with a $C$

- Find $M_k = X_{i_k^*}$ with

$$i_k^* = \text{argmin}_{\{i : C(i) = k\}} \sum_{C(j) = k} d(x_i, x_j)$$

- Update $C$:

$$C(i) = \text{argmin}_k d(X_i, M_k)$$

- Repeat the above 2 steps until convergence

- R: cluster package, containing pam() for partitioning around medoids, clara() for large
datasets with pam, silhouette() for calculating silhouette widths, diana() for divisive hierarchical clustering, etc.

  \[ \implies \text{An alternative: model-based clustering} \]

  \textbf{Mixture Model-based Clustering}

- Assume each \( X_i \) is from a mixture of Normal distributions with pdf

  \[ f(x; \Phi_K) = \sum_{r=1}^{K} \pi_r \phi(x; \mu_r, V_r) \]

  where \( \phi(x; \mu_r, V_r) \) is the pdf of \( N(\mu_r, V_r) \).

- Each component \( r \) is a cluster; probabilistic
• For a fixed $K$, use the EM to estimate $\Phi_K$ (to obtain MLE).

• Try various values of $K = 1, 2, \ldots$, then use AIC/BIC to select the one with the first local minimum.

$$\log L(\Phi_K) = \sum_{i=1}^{n} \log f(X_i; \Phi_K)$$

$$AIC = -2 \log L(\hat{\Phi}_K) + 2\nu_K$$

$$BIC = -2 \log L(\hat{\Phi}_K) + \nu_K \log(n)$$

where $\nu_K$ is #para. in $\Phi_K$.

• Or, test $H_0: K = k_0$ vs $H_A: K = k_0 + 1$; use bootstrap
- EM algorithm

Given: a set of observations \( \{X_1, \ldots, X_n\} \)

\( k < -1 \); init \( \pi_r^{(0)}, \mu_r^{(0)}, \)’s and \( V_r^{(0)} \)’s

While (not convergent) do

For all \( i = 1, \ldots, n \) and \( r = 1, \ldots, K \) do

\[
\tau_{ri}^{(k)} = \frac{\pi_r^{(k)} \phi(X_i; \mu_r^{(k)}, V_r^{(k)})}{f(X_i; \Phi(k))}
\]

/* \( \tau_{ri} \) is posterior prob \( Y_i \) in component \( r \) */

\[
\pi_r^{(k+1)} = \frac{\sum_{i=1}^n \tau_{ri}^{(k)}}{n}
\]

\[
\mu_r^{(k+1)} = \frac{\sum_{i=1}^n \tau_{ri}^{(k)} X_i}{\sum_{i=1}^n \tau_{ri}^{(k)}}
\]

\[
V_r^{(k+1)} = \frac{\sum_{i=1}^n \tau_{ri}^{(k)} (X_i - \mu_r^{(k+1)}) (X_i - \mu_r^{(k+1)})^T}{\sum_{i=1}^n \tau_{ri}^{(k)}}
\]

\( k < -k + 1 \)

Each \( X_i \) is assigned to the component with largest \( \tau_{ri} \)
• +: a cluster is a set of obs’s from a Normal distribution—clear def; can model $V_r$ and thus shape/size of clusters; probabilistic

• −: why Normal?

  Slow

  Cluster size $\geq$ dim of $X_i$ if no restriction on $V_r$ $\implies$ have to do variable selection or dim reduction if $p$ is large

• K-means: a special case of Normal mixture model-based clustering by assuming all $V_r = \sigma^2 I$

• Software: (Fortran) EMMIX or EMMIX-GENE free at
http://www.maths.uq.edu/au/~gjm/emmix-gene/

R: mclust package

An Example

- Ref.: Pan et al (2002, Genome Biology), data available
- 2+4 samples (w/o + with pneumococcal infection), 1176 genes of rats, radiolabeled cDNA arrays
- Goal: detecting differential gene expression
- Clustering two-sample t-statistics
- The fitted mixture model is

\[ f(y; \hat{\Phi}) = 0.042 \times N(6.74, 77.07) + \]
\[0.510 \times N(0.88, 5.56) + 0.448 \times N(-0.31, 1.15).\]

- Fig 4

Figure 1: Posterior probability of being in each cluster as a function of the t-statistic \( y \).

Other Methods

- Hierarchical clustering: divisive (top-down) algorithm (p. 478, 480)
• Self-Organizing Maps: a constrained version of K-means (section 14.4)

Practical Issues

• How to select the number of clusters? Anyway, what is a cluster?

Stability or significance of clusters

• Any clusters?

  – A global test: a parametric bootstrap
    Ref: McShane et al (Bioinformatics, 2002)
    – $H_0$: a Normal distr
      or a uniform or ...?
    – (optional) Principal component analysis (PCA): use first 3 PC’s for each obs
PC’s are orthogonal

- Under $H_0$, simulate data $Y_i^b$ from a MVN component-wise mean/var same as that of the data’s PC’s

- For each obs $Y_i$
  
  $d_i$ is the distance from $Y_i$ to its closest neighbor

  similarly for $d_i^{(b)}$ using $Y_i^{(b)}$, $b = 1, \ldots, B$

- $G_0$ is the empirical distr func (EDF) of $d_i$’s

  $G_b$ is the EDF of $d_i^{(b)}$’s

- Test stat

  $$u_k = \int [G_0(y) - \bar{G}(y)]^2 dy$$

  for $k = 0, 1, \ldots, B$, and $\bar{G} = \sum_b G_b / B$. 
\[ P = \# \{ b : u_b > u_o \} / B \]

- Reproducibility

- Use of the bootstrap
  
  Ref: Zhang & Zhao (FIG, 2000); Kerr & Churchill (PNAS, 2001)

- Reproducibility indices
  
  * Ref: McShane (et al, 2002)
  
  * Robustness (R) index and Discrepancy (D) index
  
  * Again, parametric bootstrap
  
  * \( Y_i \)'s: original obs's
  
  \[ Y_{ij}^{(b)} = Y_{ij} + \epsilon_{ij}^{(b)} \], where \( \epsilon_{ij}^{(b)} \) iid \( N(0, \nu_0) \),
  
  and \( \nu_0 = median(v_i's) \),
\[ v_i = \text{var}(Y_{i1}, \ldots, Y_{iK}) \]

* Cluster \( \{Y_j^{(b)} : j = 1, \ldots, K\} \) for each \( b = 1, \ldots, B \)

* Find the best-matched clusters from \( \{Y_j^{(b)}\} \) and \( \{Y_j\} \),

* For each paired clusters, \( r_k^{(b)} \) = proportion of pairs of obs’s in both clusters (i.e. kth clusters)

* \( R \) is an average of \( r_k^{(b)} \)’s

* \( D \) is an average of proportions of pairs of obs’s not in the same cluster

* Note: Finding best-matched clusters may not be easy
• Determine # of clusters
  – In general, a tough problem; many many methods
  ref’s therein
  – Clustering and classification
  – Main idea: suppose we have a training dataset and a test dataset; comparing the agreement b/w the two clustering results; $k = k_0$ will give the best agreement
  1) Cluster the test data into $k$ clusters;
2) Cluster the training data into $k$ clusters;
3) Measure how well the training set cluster centers predict c-membership in the test set.

* Fig 1

– Define “prediction strength”:

$$ps(k) = \min_{1 \leq k \leq k} \frac{1}{n_{kj}(n_{kj} - 1)} \sum_{i \neq i' \in A_{kj}} I(D[C(X_{tr}, k), X_{te}][i] = 1)$$

where $A_{k,j}$: test observations in test cluster $j$, and $n_{kj} = |A_{kj}|$.

– Choice of $k$: largest $k$ such that $ps(k) > ps_0$.

$ps_0$: 0.8-0.9

$ps(1) = 1$

– Fig 2
– In practice, use repeated 2-fold (or 5-fold) cross-validation

• Other criteria

– Let $B(k)$ and $W(k)$ be the between- and within-cluster sum of squares

– Calinski & Harabasz (1974):

$$\hat{k} = \arg \max_k \frac{B(k)/(k - 1)}{W(k)/(n - k)}$$

note: $CH(1)$ not defined.

– Hartigan (1975):

$$H(k) = \frac{W(k)/W(k + 1) - 1}{n - k - 1}$$

$\hat{k}$: smallest $k \geq 1$ such that $H(k) \leq 10$. 
- Krzanowski & Lai (1985):

\[ \hat{k} = \arg \max_k \left| \frac{D I F F (k)}{D I F F (k + 1)} \right| \]

where \( D I F F (k) = (k - 1)^2/pW_{k-1} - k^2/pW_k \), \( p \) is the dim of an obs.

- Gap stat (Tibshirani et al, JRSS-B, 2001)

  * Motivation: as \( k \) increases, \( W_k \) ...?

  Fig 1

  * \( \text{Gap}(k) = E^* [\log(W_k)] - \log(W_k) \), where \( E^* \) is expectation under a reference distribution (e.g. uniform).

  * Algorithm:

    Step 1. Cluster the observed data and obtain \( W_k \),

    \[ k = 1, \ldots, k_{max} \]
Step 2. Generate $B$ reference data sets (e.g. using the uniform distr), and obtain $W_{k}^{(b)}$, $b = 1, \ldots, B$ and $k = 1, \ldots, k_{\text{max}}$. Compute the gap stat

$$\text{Gap}(k) = \log(W)_{k} - \log(W_{k})$$

where $\log(W)_{k} = \sum_{b} \log(W_{k}^{(b)}) / B$.

Step 3. Compute SD

$$sd_{k} = \sum_{b} [\log(W_{k}^{(b)}) - \log(W)_{k}]^{2} / B$$

and define $s_{k} = sd_{k} \sqrt{1 + 1 / B}$.

Step 4. Choose a smallest $k$ such that

$$\text{Gap}(k) \leq \text{Gap}(k + 1) - s_{k+1}$$

* Fig 2
- Use of bagging: Dudoit & Fridlyand (Genome Biology, 2002)
more ref’s

• Assessing clustering results

- Define $a_i =$ average dissimilarity between obs $i$ and all other obs’s of the cluster to which obs $i$ belong;
- For all other clusters $A$, $d(i, A) =$ average dissimilarity of obs $i$ to all obs’s of cluster $A$;
- $b_i = min_A d(i, A)$
- Silhouette width: $s_i = \frac{b_i - a_i}{max(a_i, b_i)}$
- a large $s_i \implies$ obs $i$ is well clustered; a
small $s_i$ (close to 0) $\implies$ obs $i$ lies between two clusters; a negative $s_i$ $\implies$ obs $i$ is probably in a wrong cluster.