Distance Between Clusterings

Biswa Parajuli
parajuli@cs.fsu.edu

January 28, 2015

“Clustering: Bridging Theory and Practice”
Margareta Ackerman
http://www.cs.fsu.edu/~ackerman/
Comparing Clusterings: Motivation

- External evaluation
- Different algorithms $\rightarrow$ Different clusterings
- Ground Truth or Gold Standard vs Output of a clustering algorithm
Comparing Clusterings

Clusterings (list of cluster labels) by algorithms $A, D$:

$$X = [a, b, c, d, e, f, g, h, i, j]$$

$$C_A = A(X) = [1, 1, 1, 1, 0, 0, 0, 1, 1, 0]$$

$$C_D = D(X) = [0, 1, 0, 1, 1, 0, 1, 0, 1, 1]$$
Clusterings (list of cluster labels) by algorithms $B$, $C$:

\[ X = [a, b, c, d, e, f, g, h, i, j] \]

\[ C_B = B(X) = [0, 0, 0, 0, 1, 1, 1, 0, 0, 1] \]

\[ C_C = C(X) = [0, 0, 2, 2, 1, 1, 1, 0, 2, 1] \]
- Cluster labels are assigned arbitrarily.
- Number of clusters can vary. $C_C$ has 3 clusters while others have 2.
Distance function

Given a dataset $X$, there are two clusterings $C$ and $C'$.

$$C = \{C_1, C_2, ..., C_K\}$$

$$C' = \{C'_1, C'_2, ..., C'_K\}$$

Distance function $d(C, C')$ should measure how different the two clusterings are.

- Symmetric, non-negative
- If $C = C'$, $d(C, C') = 0$
Comparing clusterings: Hamming Distance

- Based on edges
- Measure related to *in-cluster* or *between-cluster* edges
- Clusterings are visualized as graphs
Hamming Distance: Disagreeing edge

An edge is disagreeing if it is \textit{in-cluster} in either $C$ or $C'$ and \textit{between-cluster} in the other. Consider $X$, $C_A$ and $C_D$.

- Take edge $(a, b)$
- $C_A[a] = C_A[b]$ but $C_D[a] \neq C_D[b]$
- So, $(a, b)$ is disagreeing.
Hamming Distance

\[ d_H(C, C') = \frac{e_{\text{disagree}}}{e_{\text{total}}} \]

- \( e_{\text{disagree}} \rightarrow \text{number of disagreeing edges} \)
- \( e_{\text{total}} \rightarrow \text{total number of edges in the graph} \)
Hamming Distance

- Look at all edges between every pair of items and count how many disagree.
- Polynomial time since $\frac{n(n-1)}{2}$ total edges in the complete graph.
- Also known as **Disagreement Distance**.
Classification Error Metric

- Instead of edges, look at how many points disagree.
- How many points should we move to make two clusters in $C$ and $C'$ same?
- Clusters in each clustering are observed as sets and the size of their intersection gives a measure of dissimilarity.
Classification Error Metric

\[ d_{CE}(C, C') = 1 - \frac{1}{n} \max_{\sigma} \sum_{k=1}^{K} n_{k,\sigma(k)} \]

- \( K, K' \) are the number of clusters in \( C, C' \) respectively and \( K \leq K' \)
- \( \sigma \) is a mapping between sets \{1, 2, 3...K\} and \{1, 2, 3...K'\}
- \( n_{k,\sigma(k)} \) is the number of items in an intersection \( C_k \cap C'_{\sigma(k)} \)
- \( n = |X| \)
Possible mappings:
\[ \sigma = \{(1_{C_A} \leftrightarrow 1_{C_B}, 0_{C_A} \leftrightarrow 0_{C_B}), (1_{C_A} \leftrightarrow 0_{C_B}, 0_{C_A} \leftrightarrow 1_{C_B})\} \]
Which mapping gives the max for \( \sum_{k=1}^{K} n_{k, \sigma(k)} \)?
\[ d_{CE}(C_A, C_B) ? \]
Classification Error Metric

- Optimal mapping?
- $d_{CE}(C_B, C_C)$?
Situation \( \neq \) Rosy

- \( K \cdot K' \rightarrow \) Total number of edges
- \( \binom{K \cdot K'}{K'} \rightarrow \) total number of possible mappings

Distance Between Clusterings

Biswa Parajuli | Distance Between Clusterings
Find Optimal Mapping in Polynomial Time

Max weight matching in complete bipartite graph $G = (V, E)$ where $V = \{1, 2, \ldots, K\} \cup \{1, 2, \ldots, K'\}$

Edge weight $w(k, k') = |C_k \cap C'_{k'}|$
Example

Distance Between Clusterings
**Theorem:**

A matching has maximum size iff there is no augmenting path.

**Algorithm Framework:**

- Initialize a feasible vertex labeling \( l \) and a matching \( M \) in equality graph \( G_l \).
- Until \( M \) is not perfect:
  1. Find augmenting path in \( G_l \) and increase size of \( M \) after augmentation.
  2. If no augmenting path exists, modify \( l \) to get a different \( G_l \) by bringing in new edges.
Few terminologies

- Perfect matching
- Neighborhood
- Alternating path
- Augmenting path
Few terminologies

- Vertex labeling
- Feasible vertex labeling
- Equality graph
Kuhn-Munkres/Hungarian Method

1. Initialize vertex labeling $l$. Determine $G_l$. Pick any matching $M$ in $G_l$.

2. If $M$ is a perfect matching, stop.
   Else, pick a free vertex $u \in X$. Set $S = \{u\}$, $T = \emptyset$

3. If $N_{G_l}(S) = T$, update labels:
   \[
   m_l = \min_{x \in S, y \notin T} \{l(x) + l(y) - w(x, y)\}
   \]
   \[
   l'(v) = \begin{cases} 
   l(v) - m_l & \text{if, } v \in S \\
   l(v) + m_l & \text{if, } v \in T \\
   l(v) & \text{otherwise}
   \end{cases}
   \]

4. If $N_{G_l}(S) \neq T$, pick $y \in N_{G_l}(S) - T$.
   - If $y$ is free, $u \leftrightarrow y$ is augmenting path. 
     Augment $M$ and go to 2.
   - If $y$ is matched to $z$ (say), extend alternating path: 
     $S = S \cup \{z\}$, $T = T \cup \{y\}$ and go to 3.

source: https://www.cs.ucsb.edu/~suri/cs231/Matching.pdf