Clustering by Passing Messages Between Data Points

Brendan J. Frey and Delbert Dueck

Presented by
Evan Dong
4/3/2008
Revisit of k-means

Cluster Number

Center

Similarity to Centers

Similarity Grouping

Change of Center?

Y

end

N

start
Motivation

- No priori on the number of clusters
- Independence on initialization
- Processing time to achieve good performance
Affinity Propagation

- Take each data point as a node in the network
- Consider all data points as potential cluster centers
- Start the clustering with a similarity between pairs of data points
- Exchange messages between data points until the good cluster centers are found
Exemplar: cluster center is an actual data point

Similarity $s(i,k)$: single evidence of data $k$ to be the exemplar for data $i$

$$s(i,k) = -\|x_i - x_k\|^2$$
Responsibility $r(i, k)$: accumulated evidence of data $k$ to be the exemplar for data $i$
Availability $a(i, k)$: accumulated evidence of data $i$ pick data $k$ as the exemplar
How does the Affinity Propagation work?

flow chart of affinity propagation
Derivation of Affinity Propagation

- Identifying exemplars can be considered as searching optimum to maximize Energy function (sum of similarity)
- Energy function can be represented by a factor graph
Initialization

\[ X = \{ \vec{x}_1, \ldots, \vec{x}_m \} \]

\[ \vec{x}_i = (x_{i1}, \ldots, x_{in})^T \]

\[
\begin{array}{c|c|c|c|c|c}
 i & 1 & 1 & \cdots & \cdots & 1 \\
 \hline
 k & 2 & 3 & \cdots & \cdots & m \\
 \hline
 s(i,k) & -\|\vec{x}_1 - \vec{x}_3\|^2 & \cdots & \cdots & -\|\vec{x}_1 - \vec{x}_m\|^2 \\
 \end{array}
\]

Similarity Matrix for i\neq k

i.e. i = 1
Input Preference:

\[ s(i, i) = P(s(i, k)) \]

\[ \forall k \neq i, i, k = 1, 2, \ldots, m \]

\[ P : \text{Min} / \text{Max} / \text{Median} \]

No predefined number of clusters!!!
Update

Responsibility

\[ r_{t+1}(i,k) = \lambda r_i(i,k) + (1 - \lambda) \Delta r_i(i,k) \quad i \neq k \]

\[ \Delta r_i(i,k) = s(i,k) - \max_{k' \neq k, k' \neq k} \{ a_t(i,k') + s(i,k') \} \]

\[ a_{t=0}(i,k') = 0 \]

\[ \lambda \in [0,1] \quad \text{damping factor, usually 0.5} \]

Self responsibility

\[ r_{t+1}(k,k) = \lambda r_t(k,k) + (1 - \lambda) \Delta r_t(k,k) \]

\[ \Delta r_t(k,k) = s(k,k) - \max_{k' \neq k} (s(i,k')) \]
Availability

\[ a_{t+1}(i,k) = \lambda a_t(i,k) + (1 - \lambda)\Delta a_t(i,k) \quad i \neq k \]

\[ \Delta a_t(i,k) = \min\{0, r_t(k,k) + \sum_{i' \in \{k\} \setminus \{i,k\}} \max\{0, r(i',k)\}\} \]

\[ a_{t=0}(i,k') = 0 \]

\[ a_t(i,k') \in (-\infty, 0] \]

\[ \lambda \in [0,1] \quad \text{damping factor, usually 0.5} \]

Self availability

\[ a_{t+1}(k,k) = \lambda a_t(k,k) + (1 - \lambda)\Delta a_t(k,k) \]

\[ \Delta a_t(k,k) = \max\{0, r(i',k)\}_{i' \in \{k\} \setminus \{k\}} \]
Combination of responsibility and availability

\[ E = a_{t+1}(i,k) + r_{t+1}(i,k) \]

data point \( k \) is an exemplar if \( E \geq 0 \)

Stopping condition: if the exemplar decision doesn’t change for certain iterations, usually 10
Application in Face Classification

- Use both k-means and affinity propagation to identify exemplars among 900 face images
1 Run vs. 100 Best Run

- **Affinity Propagation much less square error**

  15 images with largest square error

  ![15 images with largest square error using Affinity Propagation](image)

  15 exemplars using Affinity Propagation

  ![15 exemplars using K-Means](image)

  15 exemplars using K-Means
1 Run vs. 10,000 Run
Conclusion

- Picking actual data points as exemplars
- No priori on the number of exemplars
- Less number of parameters
- Better results with less amount of time