Clustering by Passing Messages between Data Points

Michael Pfeiffer
pfeiffer@igi.tugraz.at

Machine Learning A, WS 2007/08

Agenda

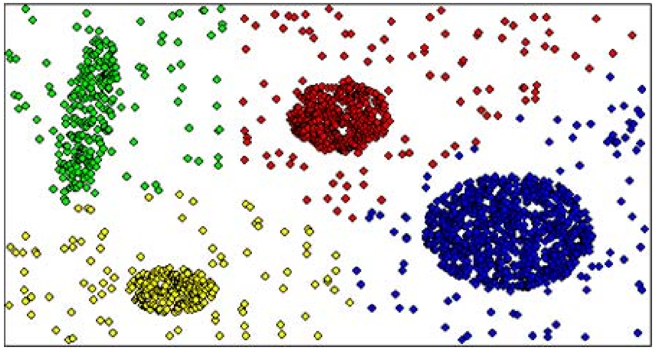
- Introduction
- Unsupervised Learning and Clustering
- Affinity Propagation
  - Algorithm (Message-Passing Concept)
  - Theoretical Considerations
- Examples
- Conclusion

Unsupervised Learning

- Given: training examples {$\mathbf{x}_i$}
  - No class or desired output!

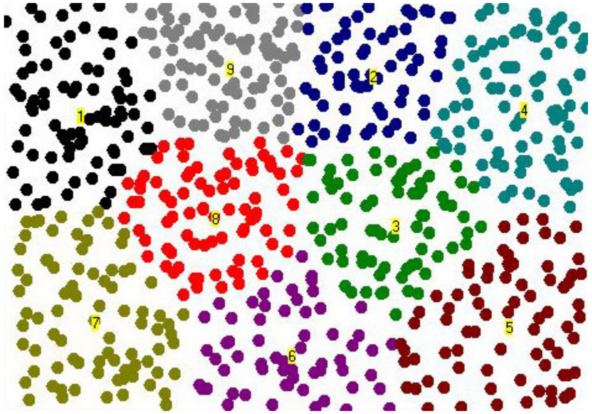
- Task: create a (probabilistic) model of the data p($\mathbf{x}$)
  - E.g. learn a Bayesian network (or any other model) that describes the input distribution

  - Discover groups of similar examples within data
    - Clustering

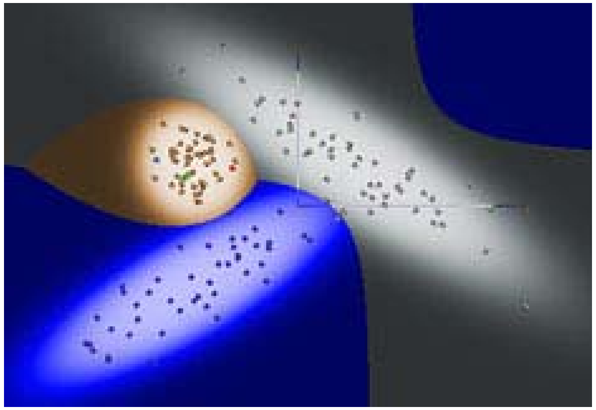


Clustering

- Unsupervised grouping or segmenting data into subsets ("clusters")
- Objects within a cluster are more closely related to one another than objects assigned to different clusters

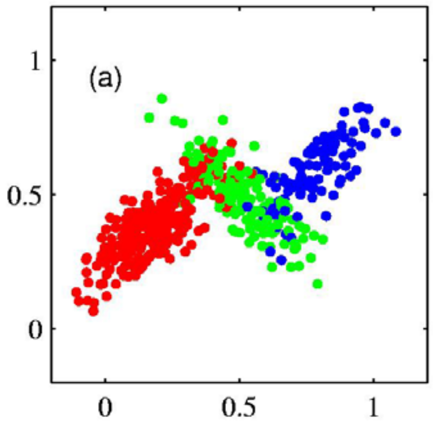


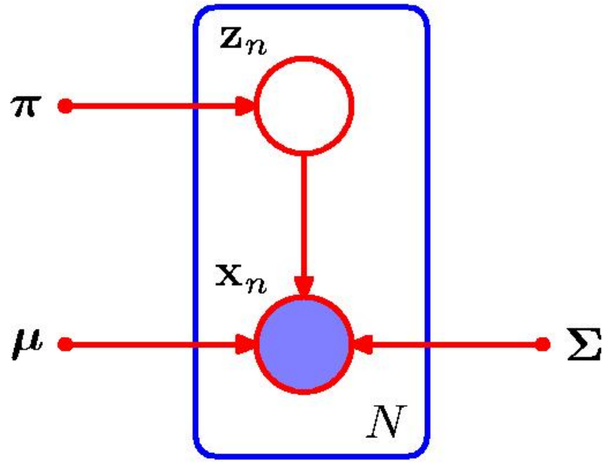
- Approaches:
  - Mixture Models (e.g k-means)
    - Feature space models
  - Linking pairs of data points
    - Pairwise relationships
  - Hierarchical clustering



Mixture Models

- Mixture of Gaussians





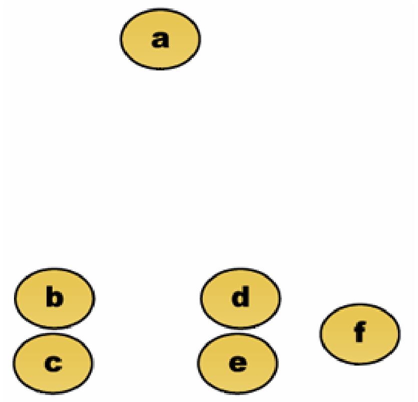
$$p(x) = \sum_{k=1}^{K} p(z_k)\, p(x \mid z_k) = \sum_{k=1}^{K} \pi_k\, N(x \mid \mu_k, \Sigma_k)$$

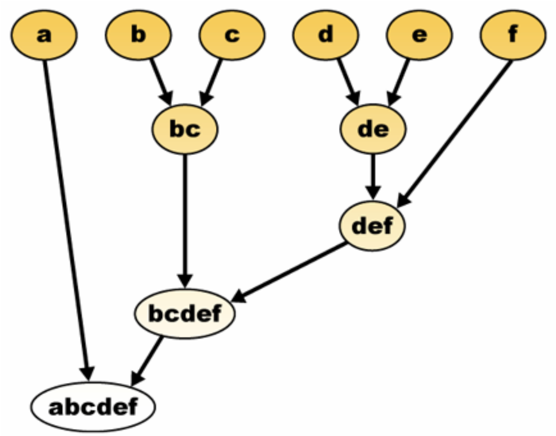
- Clustering Task: given **x**, determine most likely π_k, μ_k and Σ_k parameters

Figures from C. Bishop: Pattern Recognition and Machine Learning

Nonparametric Clustering

- Given datapoints **x** and pairwise similarities $d(x_i, x_j)$





- Group most similar subsets of datapoints
- Allows hierarchical clustering
- Can be applied to non-metric data (e.g. text, graphs, websites, …)

Figures from Wikipedia

Differences between Approaches

Mixture Models

- Data points in common feature space
- Metric to measure distances (usually Euclidean)
- Prototypes may be different from input data
- Parametric probability model for every cluster
  - Sufficient statistics
  - Not always feasible
  - Optimization is prone to local minima
- E.g. mixtures of Gaussians (k-means, EM)

Pairwise Similarities

- Many data sets do not have “coordinates”
  - E.g. text, graphs, websites, spike trains, …
- Still, computing pairwise similarities/distances is possible
  - E.g. word co-occurrence, links, spike time metric, …
- Learn only cluster assignments
- Prototypes for every cluster
  - Exemplars

Similarity Measures / Affinities

- Euclidean Metric: $\left\| \vec{x}_i - \vec{x}_j \right\|$
  - can only be used for quantitative variables
- Qualitative (ordinal, categorical) variables
  - Grades, preferences, …
  - Gender, color, nationality, …
- Non-standard metrics
  - Text documents: bag-of-words
  - Webpages: links
  - Some data points may be non-comparable!
- Similarities can be converted to distances and vice-versa

A bag-of-words
for this slide

```
…
Euclid      1
…
Lunch       0
…
Similarity  2
…
Soccer      0
…
Variable    2
…
```

Criteria for Clustering

- Mixture models:
  - Data likelihood under mixture model

$$P(X \mid C) = \prod_{j=1}^{n} P(x_j \mid \theta_i ; c(j) = i)$$

θ_i … parameters for cluster i

- Pairwise similarities
  - Minimizing within cluster scatter

$$W(C) = \frac{1}{2} \sum_{i=1}^{k} \sum_{c(j)=i} \sum_{c(j')=i} d(x_j, x_{j'})$$

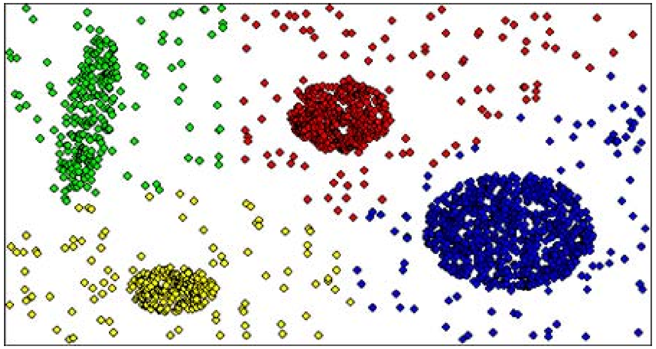
  - equiv. maximize between cluster scatter

$$B(C) = \frac{1}{2} \sum_{i=1}^{k} \sum_{c(j)=i} \sum_{c(j')\neq i} d(x_j, x_{j'})$$

  - W(C) + B(C) = T = const.

Simple k-Means Clustering

- Start with *k* random cluster centers μ_l
- Assign samples to nearest cluster center under the Euclidean distance $d(x_j, \mu_i) = ||x_i - \mu_j||^2$
- Re-compute cluster centers μ_l
- Until assignments do not change



- Variant of EM algorithm
- Simple to extend to non-constant covariance matrices $\Rightarrow$ probability mixture model of data

k-Medoids Clustering

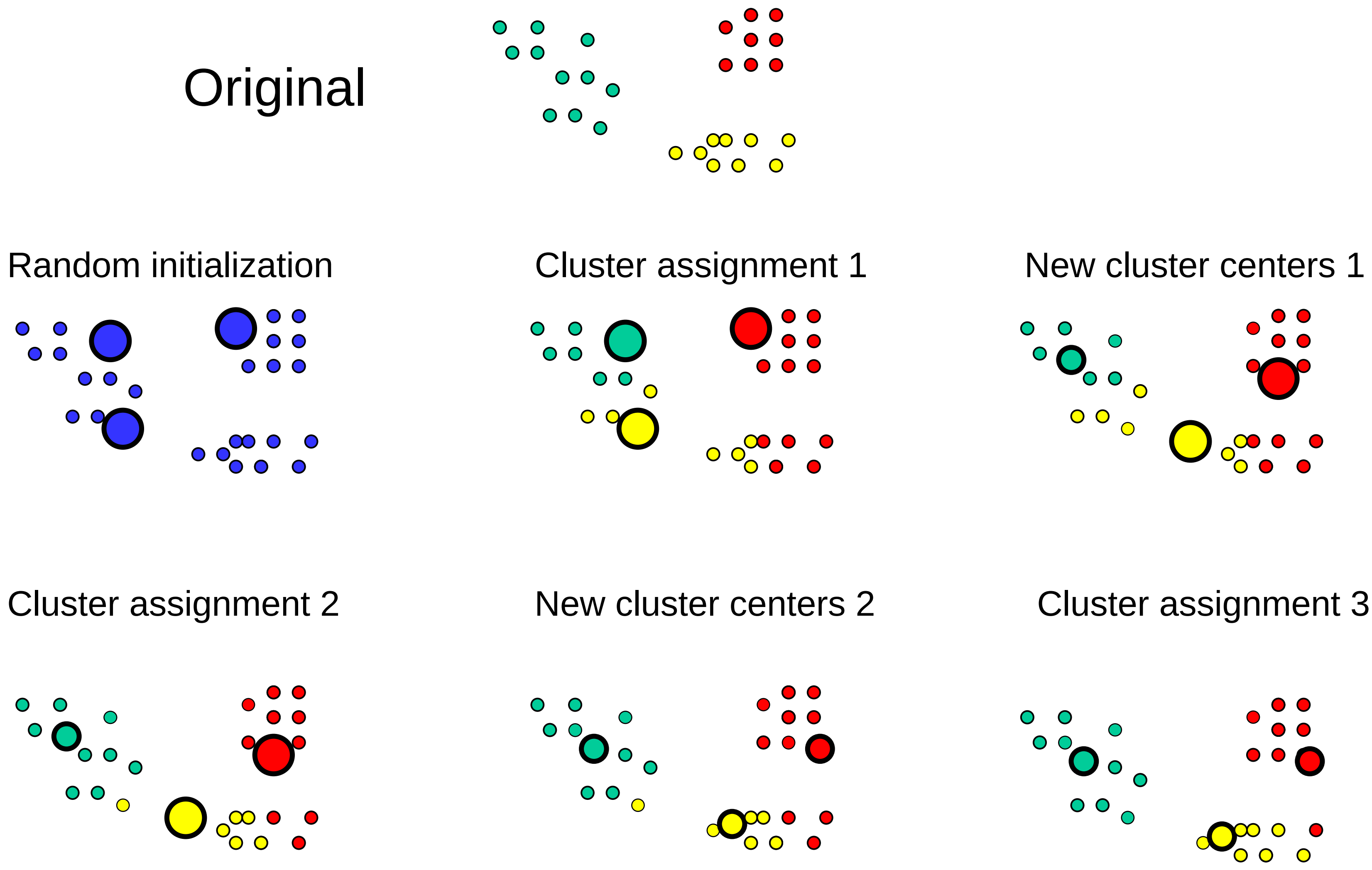
- What if we cannot compute means?
- Cluster centers must be training cases

- Start with *k* random data points as cluster centers μ_i
- Assign every point x_j to closest cluster center $x_{c(j)}$
- For every cluster i:= 1 to k
  - Find $\mu_i := x_j$ with $c(j)=i$ such that total distance to other cluster points is minimized

$$\mu_i := \arg\min_{j:c(j)=i} \sum_{c(j')=i} d(x_j, x_{j'})$$

- Iterate until cluster assignments $c(j)$ do not change

k-Medoids Example



k-Medoids vs. k-Means

k-Medoids

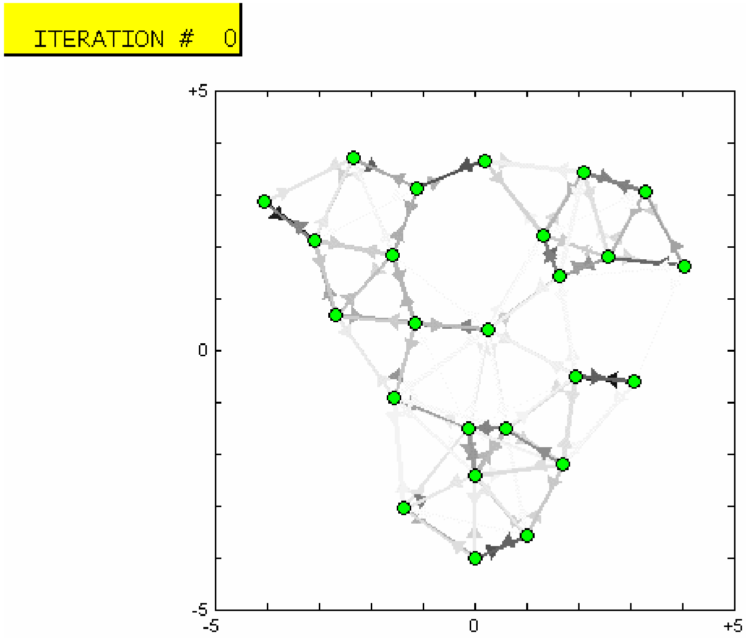
- Works with any metric (only distance matrix required)
- Returns cluster exemplars
- Computationally more intensive
  - Finding cluster exemplar is $O(n_i^2)$, where n_i is the number of points in the cluster
- Exact learning is NP-hard
- Depends on (random) initialization of cluster centers

k-Means

- Requires Euclidean space and metric
- Returns probability model
- Simple to implement
- Converges very quickly in practice
- Prone to local minima
- Depends on (random) initialization of cluster centers

Affinity Propagation [Frey, Dueck 2005,2007]

- Uses only distance / similarity matrix
- Extension of k – Medions:
  - k-Medions randomly chooses k initial cluster exemplars
  - Affinity Propagation simultaneously considers all data points as potential exemplars
- Regards data points as nodes in a network / graph
- Propagates real-valued messages (“affinities”) between data points
- Automatically detects clusters, exemplars and number of clusters



Figures from B. Frey and D. Dueck: Clustering by Passing Messages between Data Points, Science 2007

Advantages of AP

- Deterministic algorithm
  - k-means and k-medoids are both very sensitive to the initial selection of cluster centers
  - Must usually be re-run multiple times
- Automatic determination of number of clusters

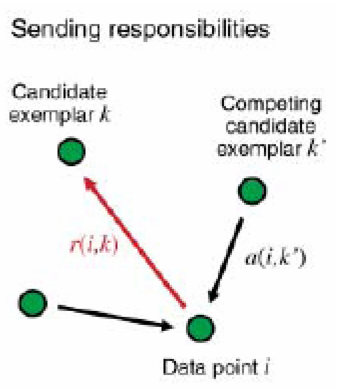
- Works in non-metric spaces
- Doesn’t require any special properties of the distance / similarity measure (e.g. triangle inequality, symmetry)
- Can be extended to a probabilistic mixture model
  - Similarity is (log-) likelihood

- Fast running time

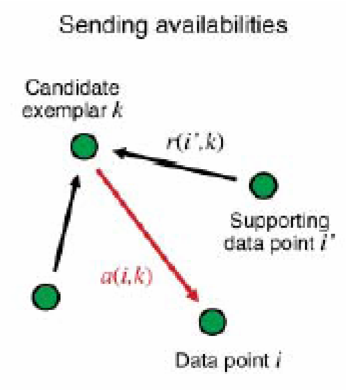
Message Passing

- Messages represent current affinities of one data point for choosing another point as its exemplar

- Two kinds of messages:
  - Responsibility: data point → candidate exemplar
    - how well suited is exemplar for data point, compared to all other possible exemplars



  - Availability: candidate exemplar → data point
    - How appropriate is candidate as exemplar for data point, taking support from other data points into account
    - "soft" cluster assignment



Figures from B. Frey and D. Dueck: Clustering by Passing Messages between Data Points, Science 2007

Responsibilities

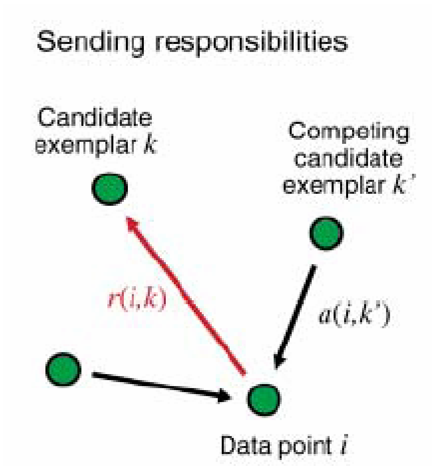
- Responsibility *r*(*i*, *k*):
  - message from data point *i* to potential exemplar *k*
  - Accumulated evidence for *k* being the exemplar for *i*, taking other potential exemplars into account
  - *r*(*k*,*k*): self-responsibility: prior likelihood for *k* to be chosen as exemplar
    - Defined by user, determines number of clusters
    - Good choice: $r(k,k) = \underset{i \neq k}{median}\left(s(i,k)\right)$
- Update rule:

$$\boxed{r(i,k) \leftarrow s(i,k) - \max_{k' \neq k}\left(a(i,k') + s(i,k')\right)}$$

  - s(i,k) … similarity between i and k
  - a(i,k) … availability of k for i

Responsibilities

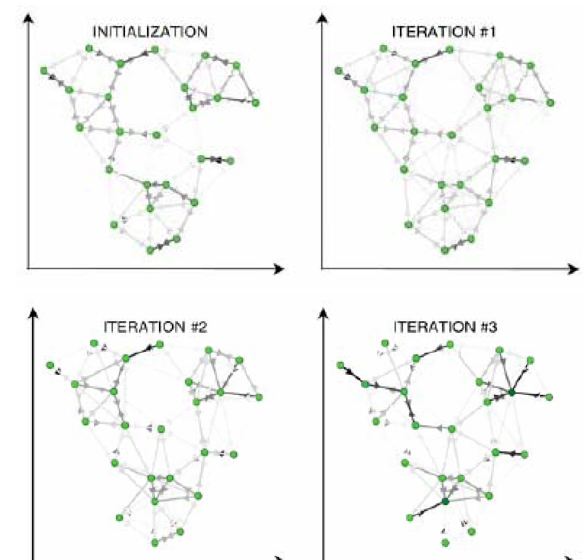
$$r(i,k) \leftarrow s(i,k) - \max_{k' \neq k}\big(a(i,k') + s(i,k')\big)$$



- Candidate exemplars compete for ownership of data points

- a(i,k) initially 0
- First iteration: responsibility = input similarity – largest similarity with other exemplars

- Availability later reflects, how many other points favor *k* as an exemplar
- r(k,k) … evidence that k is an own exemplar



Figures from B. Frey and D. Dueck: Clustering by Passing Messages between Data Points, Science 2007

Availabilities

- Availability $a(i,k)$
  - message sent from potential exemplar k to data point i
  - accumulated evidence for k being the exemplar of i, taking support from other points i' for k into account
  - If many points choose k as exemplar, k should survive as an exemplar

- Update rules:

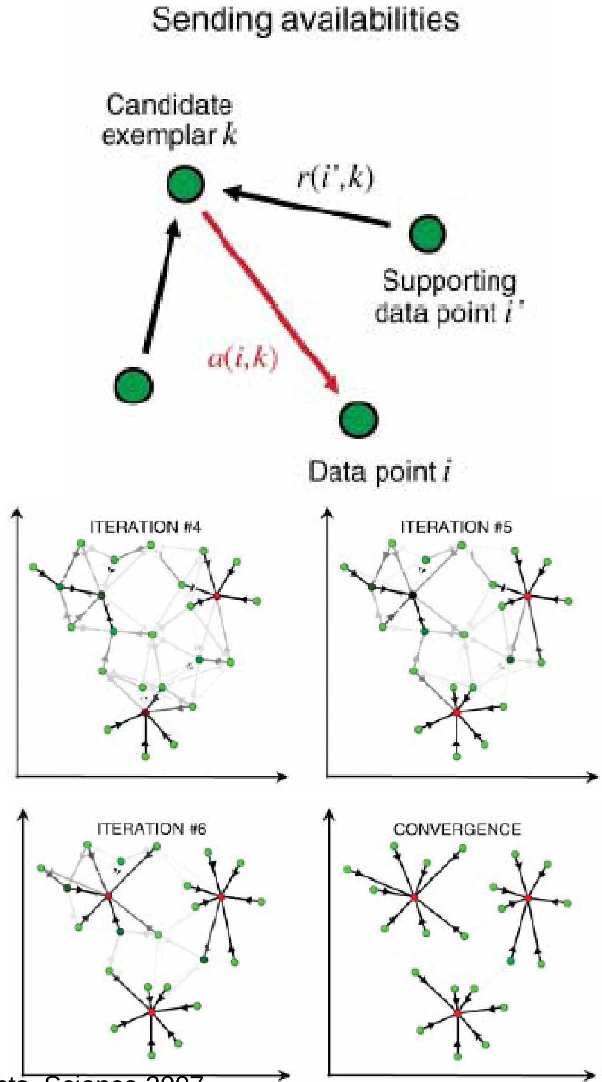
$$\boxed{\begin{aligned} &a(i,k) \leftarrow \min\left\{0,\ r(k,k) + \sum_{i' \notin \{i,k\}} \max\left(0,\ r(i',k)\right)\right\} \\ &a(k,k) \leftarrow \sum_{i' \neq k} \max\left(0,\ r(i',k)\right) \end{aligned}}$$

Availabilities

$$a(i,k) \leftarrow \min\left\{0,\ r(k,k) + \sum_{i' \notin \{i,k\}} \max(0,\ r(i',k))\right\}$$

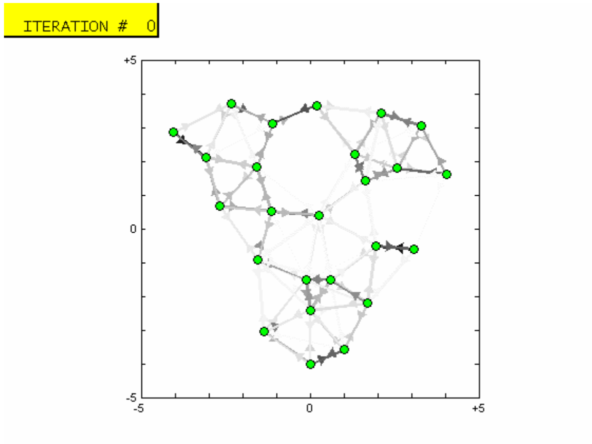
$$a(k,k) \leftarrow \sum_{i' \neq k} \max(0,\ r(i',k))$$

- Collect evidence from data points whether *k* is a good exemplar
- Self-responsibility (prior-likelihood) plus positive responsibilities from other points
- Why only positive?
  - Exemplar needs only explain part of the data
- Self-availability a(k,k): accumulated evidence from other points for *k* as exemplar



Figures from B. Frey and D. Dueck: Clustering by Passing Messages between Data Points, Science 2007

Affinity Propagation Update Rules



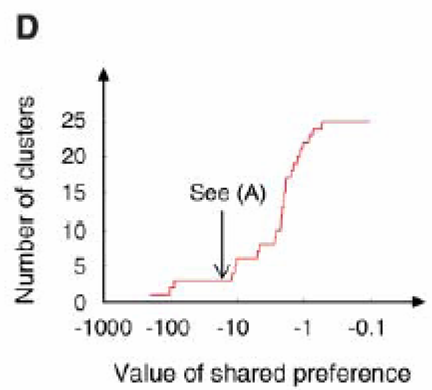
- r(i,j)=0; a(i,j)=0; ∀ i,j
- for i:=1 to num_iterations
  - $r(i,k) \leftarrow s(i,k) - \max\limits_{k' \neq k}\left(a(i,k') + s(i,k')\right)$
  - $a(i,k) \leftarrow \min\left\{0,\ r(k,k) + \sum\limits_{i' \notin \{i,k\}} \max\left(0,\ r(i',k)\right)\right\}$
  - $a(k,k) \leftarrow \sum\limits_{i' \neq k} \max\left(0,\ r(i',k)\right)$
- end;
- for all x_i with (r(i,i)+a(i,i) > 0)
  - x_i is exemplar
  - Assign non-exemplars x_j to closest exemplar under similarity measure s(i,j)
- end;

Figures from B. Frey and D. Dueck: Clustering by Passing Messages between Data Points, Science 2007

Implementation Issues

- Numerical oscillations may occur
  - e.g. s(1,2)=s(2,1) and s(1,1)=s(2,2)
  - i.e. cluster assignments are equally likely
  - Add tiny noise to similarity matrix
  - Use dampening factor λ (usually λ=0.5) :
    - $r_{t+1}(i,j) = \lambda\ r_t(i,j) + (1- \lambda)\ r_{new}(i,j)$
    - $a_{t+1}(i,j) = \lambda\ a_t(i,j) + (1- \lambda)\ a_{new}(i,j)$

- Running time $O(N^2)$ per iteration
- More efficient version for sparse connectivity
  - messages only along existing edges
- Initialization of r(i,i):
  - Determines number of clusters *k*
  - Exact definition of *k* by searching over several scalings for r(i,i)

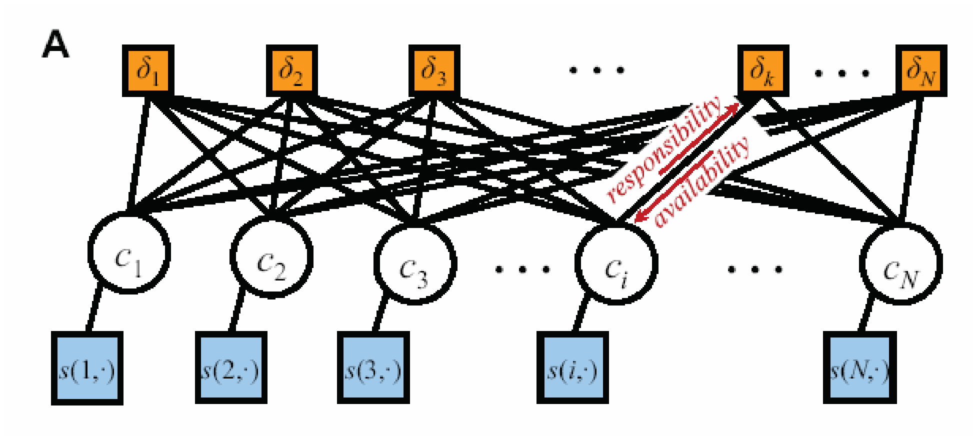


Figures from B. Frey and D. Dueck: Clustering by Passing Messages between Data Points, Science 2007

Theoretical Foundations

- Goal: Maximize $$S(c)=\sum_{i=1}^{N} s(i,c(i))+\sum_{k=1}^{N}\delta_k(c(k))$$
$$\delta_k(\vec{c})=\begin{cases}-\infty & \text{if } c_k \neq k \text{ but } \exists i: c_i = k\\ 0 & \text{otherwise}\end{cases}$$
  - $\delta_k(c)$ … penalty term: -∞ if *k* is chosen as exemplar by another point *i*, but not by itself

- Represented as a **factor graph**

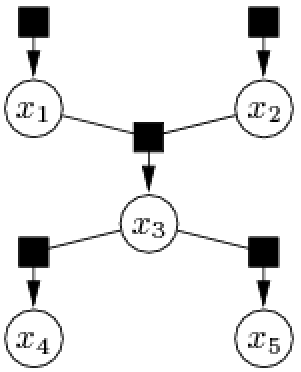


- Function nodes for s and δ
- Variable nodes for c(i)
- Messages from variables to functions and vice-versa

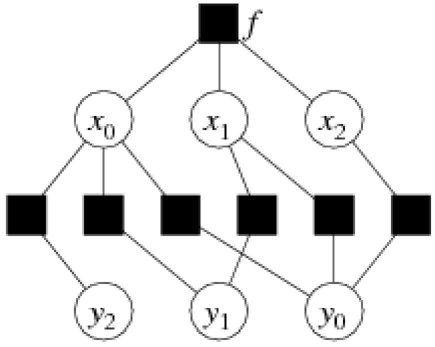
Figures from B. Frey and D. Dueck: Clustering by Passing Messages between Data Points, Science 2007

Factor Graphs

- For computing joint functions of many variables
- Factors into product of smaller variable sets
- Function vs. variable nodes (bipartite graph)
- Local functions depend only on connected variables



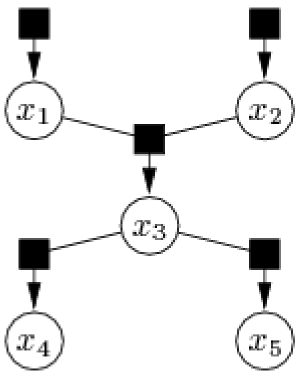
- E.g. belief propagation for joint probabilities (functions are conditional probabilities)

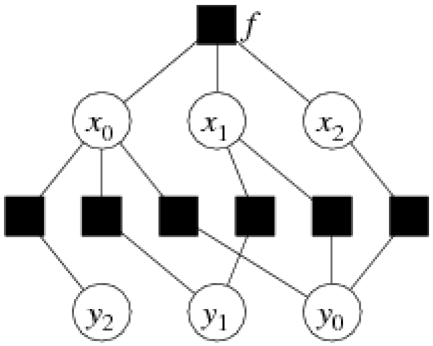


- Approximate inference with probability propagation (loopy belief-propagation)
  - Sum-product or max-product algorithm
  - resp. **max-sum** in log-domain
- Used successfully for error-correcting decoding, random satisfiability, stereo vision,…

Computation in Factor Graph

- Global function: sum (or product) of all function nodes (black boxes)
- Value of function node:
  - sum of all incoming messages
- Message from variable node:
  - sum of all incoming messages, except from target node
- Message from function node:
  - Sum incoming messages and maximize over all variables except target variable

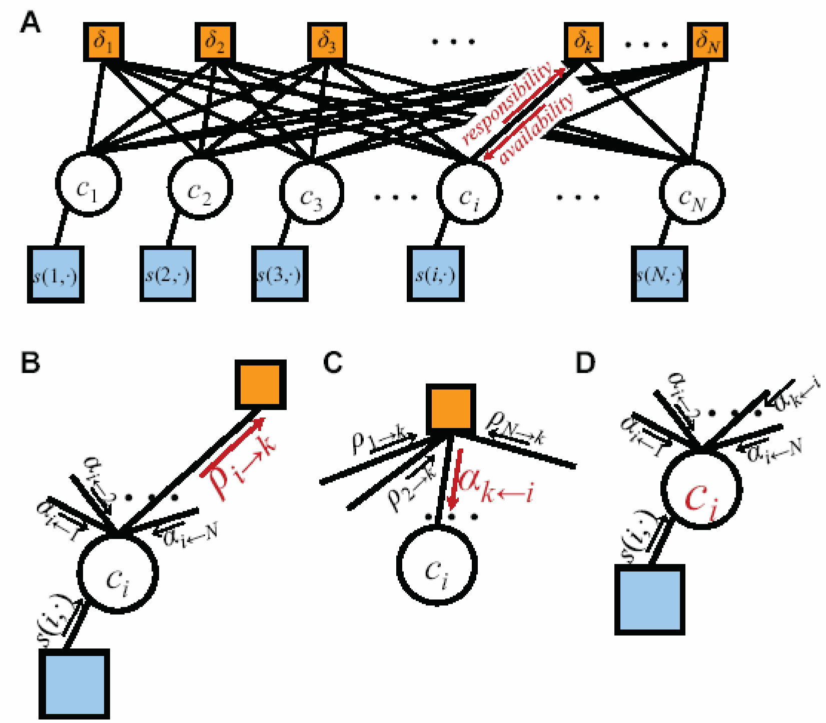




Affinity Propagation with max-sum

$$S(c) = \sum_{i=1}^{N} s(i, c(i)) + \sum_{k=1}^{N} \delta_k(c(k))$$

$$\delta_k(\vec{c}) = \begin{cases} -\infty & \text{if } c_k \neq k \text{ but } \exists i : c_i = k \\ 0 & \text{otherwise} \end{cases}$$

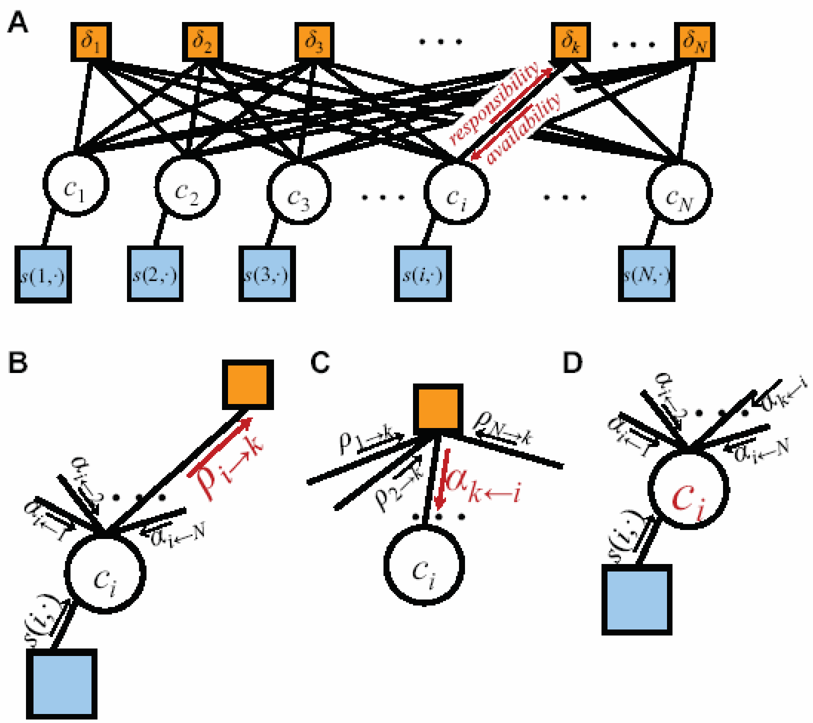


- Function nodes:
  - s … similarity with exemplar
  - δ … valid result
- Maximize S(c): net similarity of data with exemplars under constraints for cluster membership
- Use **max-sum** algorithm
  - = max-product algorithm in logarithmic domain

Figures from B. Frey and D. Dueck: Clustering by Passing Messages between Data Points, Science 2007

Messages in Factor Graph

- From c_i to δ_k: $\rho_{i \to k}(j)$
  - for all possible $c_i = j$
  - “responsibility”
  - sum all incoming messages (except from target)
- From δ_k to c_i: $\alpha_{i \leftarrow k}(j)$
  - “availability”
  - sum incoming and maximize over all but target variable
- Value of c_i is estimated by summing all incoming availability and similarity messages
- Leads to presented algorithm
  - See supplementary material to Paper by Frey and Dueck (Science, 2007) for exact derivation



Figures from B. Frey and D. Dueck: Clustering by Passing Messages between Data Points, Science 2007

Mixture Models by Affinity Propagation

- Similarity ~ likelihood
  - s(i,k) ~ L(i,k) = P(x_i | x_i in cluster with center x_k)

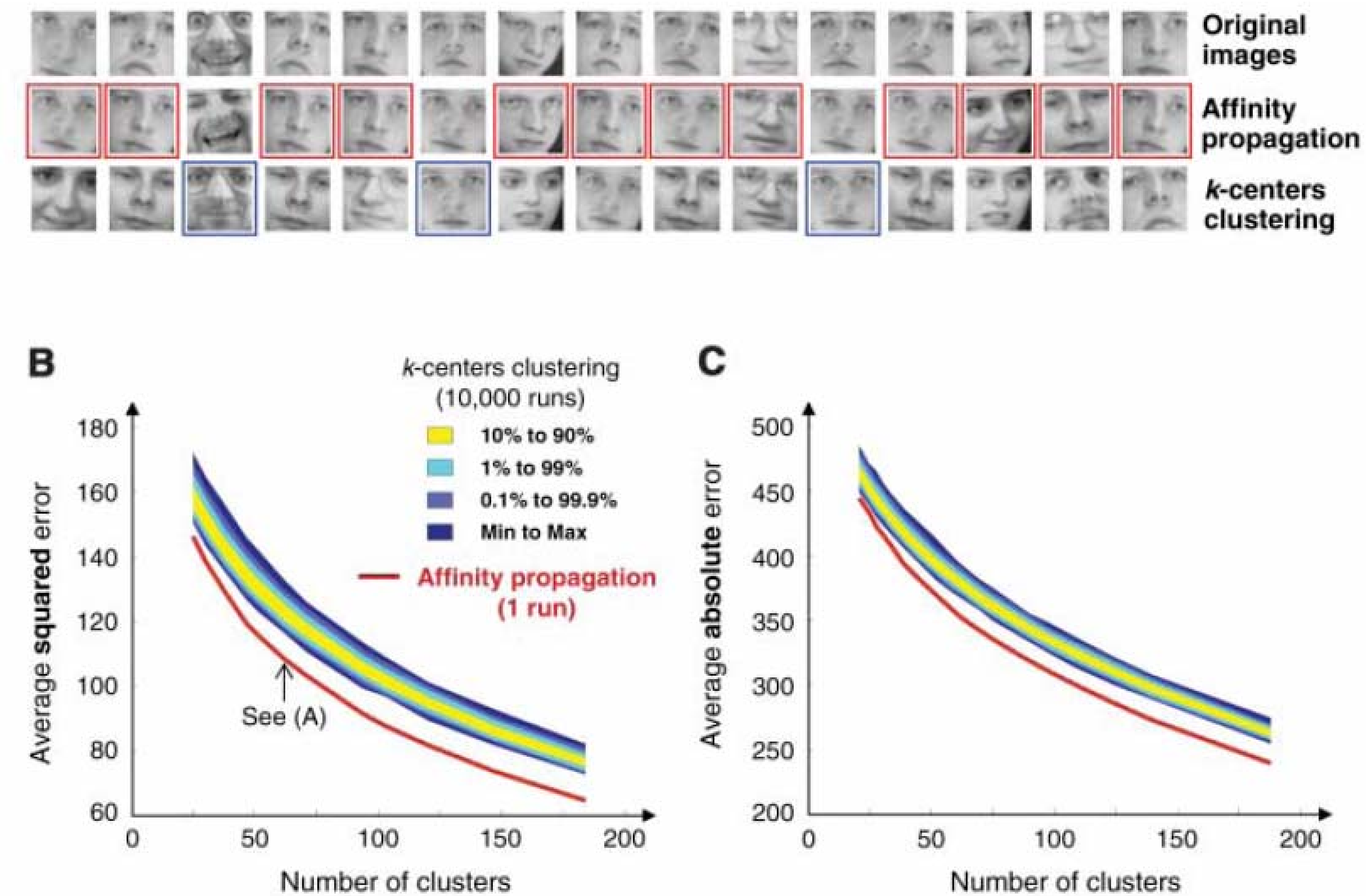
$$r(i,k) \leftarrow L(i,k) \Big/ \left(\sum_{j \neq k} a(i,j) \cdot L(i,j) \right)$$

$$a(k,k) \leftarrow \prod_{j \neq k} (1 + r(j,k)) - 1$$

$$a(i,k) \leftarrow 1 \Big/ \left(\tfrac{1}{r(k,k)} \prod_{j \notin \{i,k\}} (1 + r(j,k))^{-1} + 1 - \prod_{j \notin \{i,k\}} (1 + r(j,k))^{-1} \right)$$

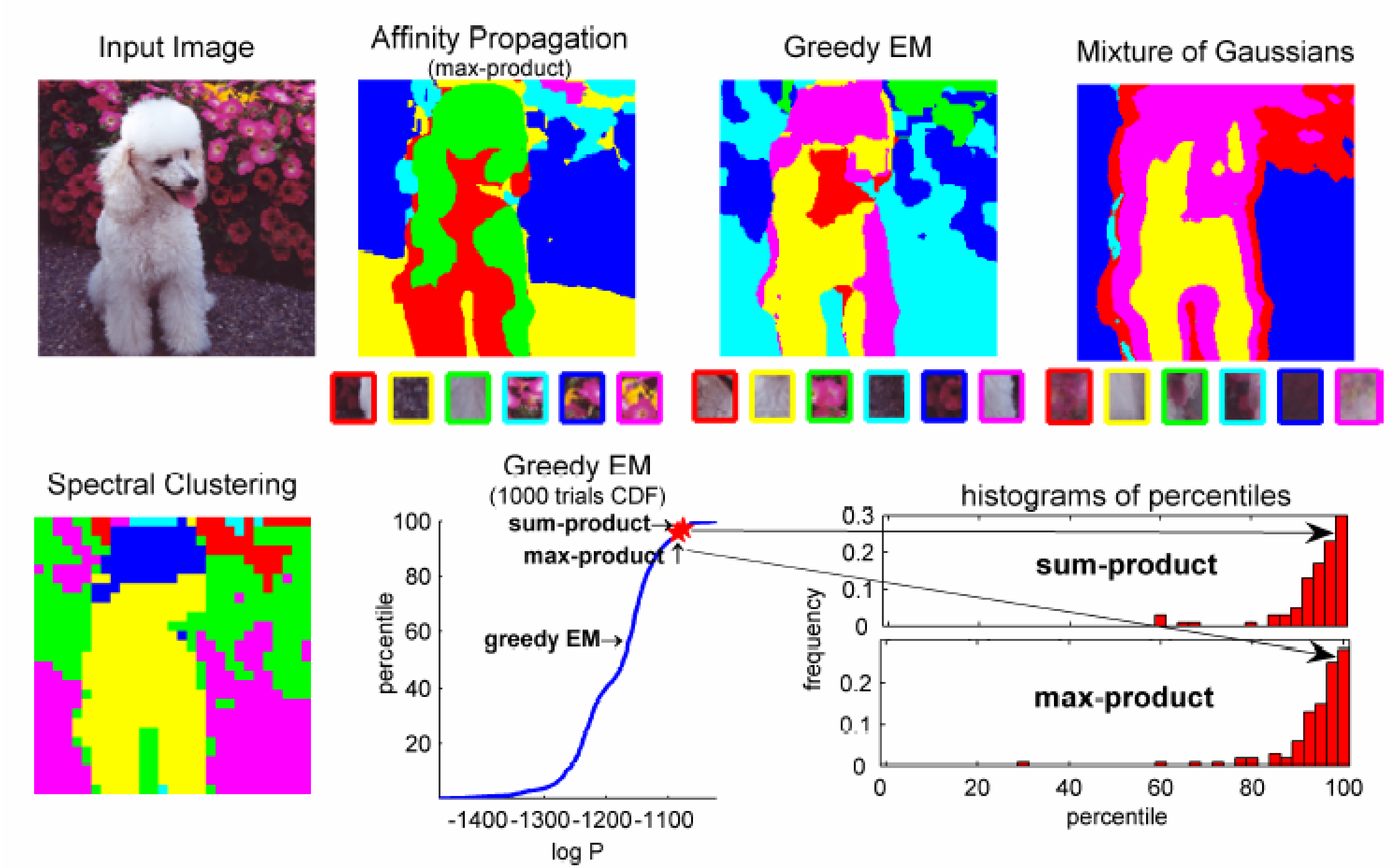
- L_{ii} prior for exemplars
- Variant of sum-product algorithm for belief propagation

Results: Face Image Clustering



Figures from B. Frey and D. Dueck: Clustering by Passing Messages between Data Points, Science 2007

Results: Image Segmentation

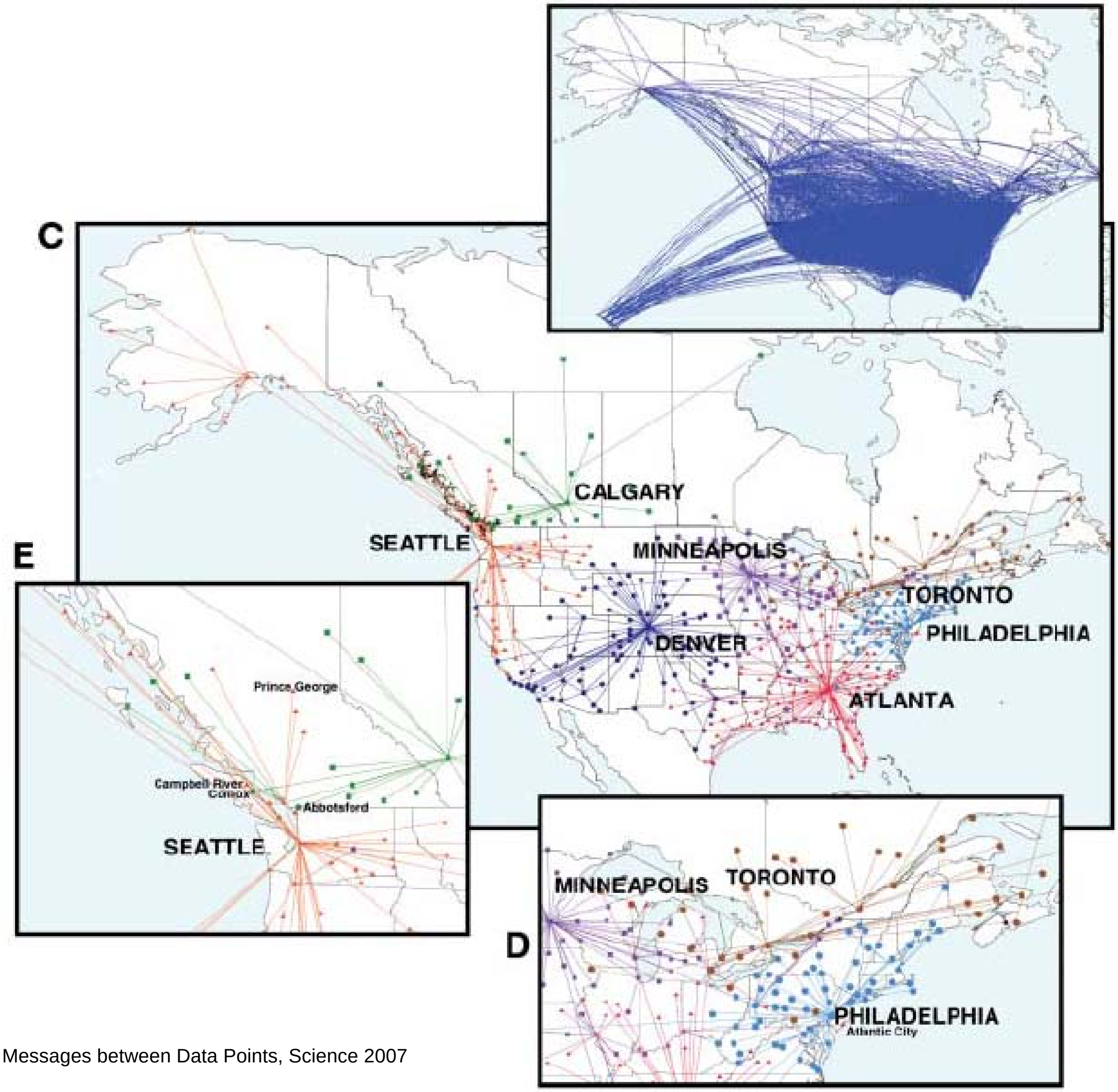


- Clustering of image patches (fixed k=6)
- Affinity propagation is near-best to 1000 EM runs

Figures from B. Frey and D. Dueck: Mixture Modeling by Affinity Propagation, NIPS 2005

Results: Flight Connections

- Most easily accessible cities in Canada and US by domestic airline travel
- Similarity: negative fastest flight time + stopover delay
  - 36% asymmetric
  - Triangle inequality violated (stopover)
- Heavy international traffic disturbs domestic flights
  - E.g. New York, Chicago, LA,…
- Smaller airports have shorter delays
- No direct connections for short distances



Figures from B. Frey and D. Dueck: Clustering by Passing Messages between Data Points, Science 2007

Netflix Prize: 1 Mio. $ Challenge

- Huge database about movie preferences
- Improve Netflix' prediction by 10% and win 1 Mio. $

- Database looks like this:
  - User x gave 5 stars to movie "The Matrix", 5 to "Lord of the Rings" and 1 to "Pretty Woman"
  - Task: How many stars will user x give to "Gladiator"?

  - 17,700 movies and 480,000 users

Results: Movie Clustering

- Netflix database (1 Mio. $ competition)
- Similarity: # common viewers / # total viewers
  - 694 clusters

Lord of the Rings: The Fellowship of the Ring (149866) Lord of the Rings: The Return of the King Lord of the Rings: The Two Towers Lord of the Rings: The Fellowship of the Ring Indiana Jones and the Last Crusade The Matrix Gladiator Pirates of the Caribbean: The Curse of the Black Pearl Harry Potter and the Chamber of Secrets The Last Samurai X2: X-Men United Crouching Tiger, Hidden Dragon X-Men The Count of Monte Cristo Kill Bill: Vol. 2 Spider-Man Kill Bill: Vol. 1 Minority Report	Monsters, Inc. (130243) Finding Nemo (Widescreen) Toy Story Shrek (Full-screen) The Incredibles Monsters, Inc. The Lion King: Special Edition Harry Potter and the Prisoner of Azkaban Aladdin: Platinum Edition Shrek 2 Harry Potter and the Sorcerer's Stone A Bug's Life Ice Age Holes Shark Tale	The Best of Friends: Season 2 (21218) The Best of Friends: Season 3 The Best of Friends: Season 1 Friends: Season 4 The Best of Friends: Season 2 The Best of Friends: Vol. 2 Friends: Season 1 Friends: Season 3 The Best of Friends: Vol. 1 Friends: Season 2

More Movie Clusters

The Spy Who Loved Me (19530)
From Russia With Love
Thunderball
Dr. No
You Only Live Twice
Goldfinger
For Your Eyes Only
Live and Let Die
The Spy Who Loved Me
The Man with the Golden Gun
The Living Daylights
The Thomas Crown Affair

Midway (10201)
The Longest Day
The Tuskegee Airmen
Where Eagles Dare
Kelly's Heroes
Gettysburg
Midway
A Bridge Too Far
Tora! Tora! Tora!
The Final Countdown
Hamburger Hill
Force 10 from Navarone
Bat 21

For a Few Dollars More (17107)
The Magnificent Seven
The Good, the Bad and the Ugly
The Outlaw Josey Wales
For a Few Dollars More
The Dirty Dozen
High Plains Drifter
A Fistful of Dollars
Hang 'Em High
Once Upon a Time in the West
Pale Rider
Silverado
Escape from Alcatraz
The Enforcer
Bullitt
Slap Shot: 25th Anniversary Edition
Magnum Force
The Longest Yard
Two Mules for Sister Sara

Magnetic Storm: Nova (602)
World Almanac Video: The Expanding Universe
Origins: Nova
To the Moon: Nova
Magnetic Storm: Nova
Stephen Hawking's Universe
For All Mankind
MARS Dead or Alive: Nova
The Secret Life of the Brain

NFL: Super Bowl XXXIX (119)
The Boston Red Sox: 2004 World Series Collector's Edition
New England Patriots: NFL Super Bowl XXXVIII Champions
MLB: 2004 World Series
NFL: Dallas Cowboys Team History
Sports Illustrated Swimsuit Edition: 2004
Formula One Review 2004
Golf for Dummies
NFL: History of the Philadelphia Eagles
NFL: Super Bowl XXXIX
Nine Innings from Ground Zero

Seven: Bonus Material (868)
Shrek (Widescreen)
GoodFellas: Special Edition: Bonus Material
The Indiana Jones Trilogy: Bonus Material
The Royal Tenenbaums: Bonus Material
Reservoir Dogs: Bonus Material
The Godfather Trilogy: Bonus Material
Forrest Gump: Bonus Material
Seven: Bonus Material
Pulp Fiction: Bonus Material
Fight Club: Bonus Material
Amelie: Bonus Material
A Bug's Life: Bonus Material
Trainspotting: Collector's Edition: Bonus Material
Scarface: 20th Anniversary Edition: Bonus Material
There's Something About Mary: Special Edition: Bonus Material
Jerry Maguire: Bonus Material
Monsters, Inc.: Bonus Material

Conclusion

- Clustering finds structure in unlabeled data
- Affinity Propagation: clustering by message passing
  - Factor graphs can be used for more than Bayesian inference
  - Sum-product / max-product algorithm for clustering
- AP can be applied to problems with arbitrary similarity measures
  - Does not require continuous space, symmetry or metric that fulfills triangle inequality
- Automatically selects number of clusters
- Efficient algorithm (exploits sparse connectivity)
  - Works very well in practice
- Potential weaknesses:
  - Memory: large similarity matrices
  - Determination of number of clusters

Literature

- B. Frey and D. Dueck: Clustering by Passing Messages between Data Points, Science 315, 972-976, 2007
- B. Frey and D. Dueck: Mixture Modeling by Affinity Propagation, NIPS 2005

- B. Frey: Graphical Models for Machine Learning and Digital Communication, MIT Press, 1998