Computational Linguistics

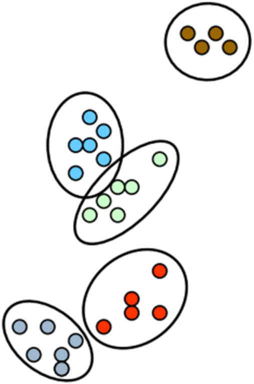
Clustering

Clayton Greenberg and Stefan Thater

Department of Computational Linguistics, Saarland University

23 June 2016

Cluster analysis



Goal:

- group similar items together
- pre-existing labels are not assumed

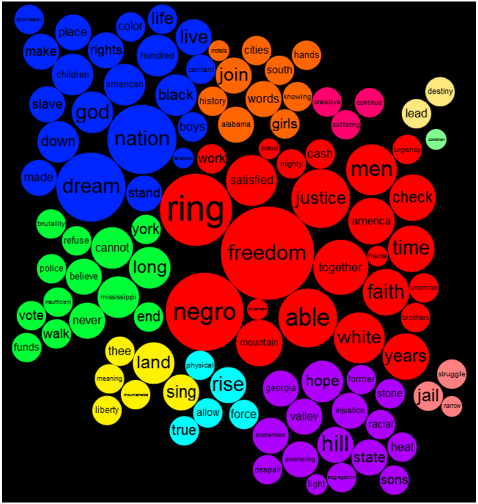
Steps:

1. define distance between points in the sample
2. define a loss function
3. find an algorithm that minimizes the loss function

Outline

1. Clustering examples
2. Unsupervised learning
3. Distance measures
4. *K*-means clustering
5. The Variance Ratio Criterion
6. Other clustering algorithms
7. Application to Named Entity Tagging

Cluster Word (speech “I have a dream”)



http://neoformix.com/2011/wcd_KingIHaveADream.png

Cluster Text (e.g. search results)

Searches related to cluster

cluster **meaning**	cluster **server**
cluster **band**	cluster **computing**
cluster **sampling**	cluster **analysis**
cluster **headaches**	cluster **database**

Goooooooooooogle >

1 2 3 4 5 6 7 8 9 10 Next

Cluster Image Regions: Image Segmentation

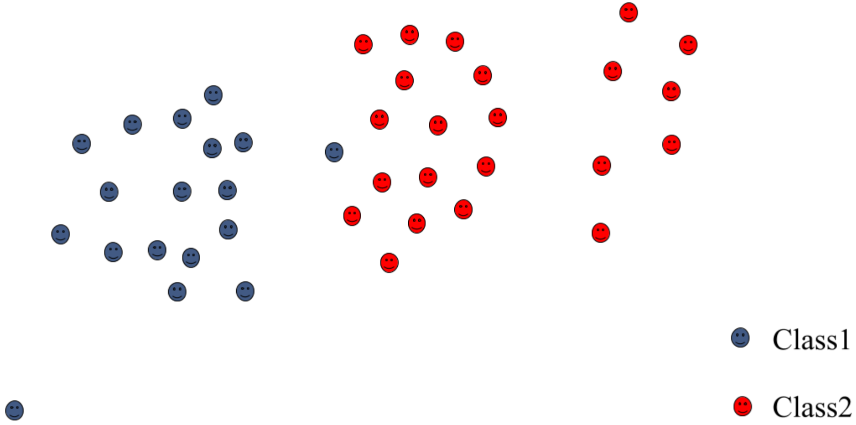
http://cs.brown.edu/~pff/segment/

Cluster Image Regions

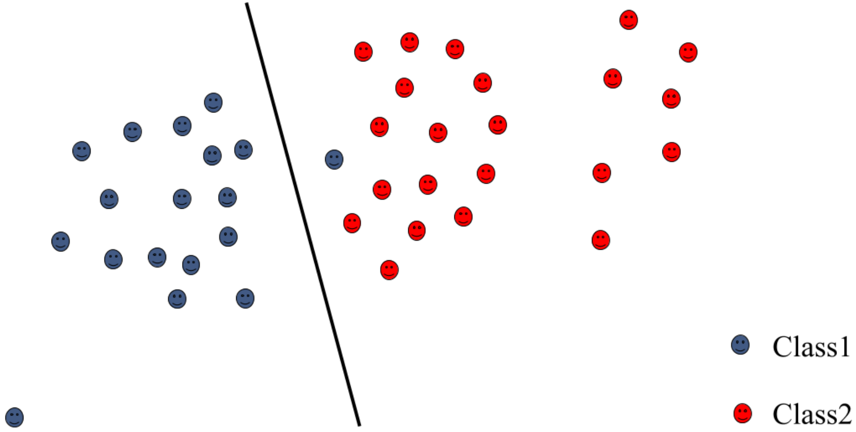
Bishop, PRML

Outline

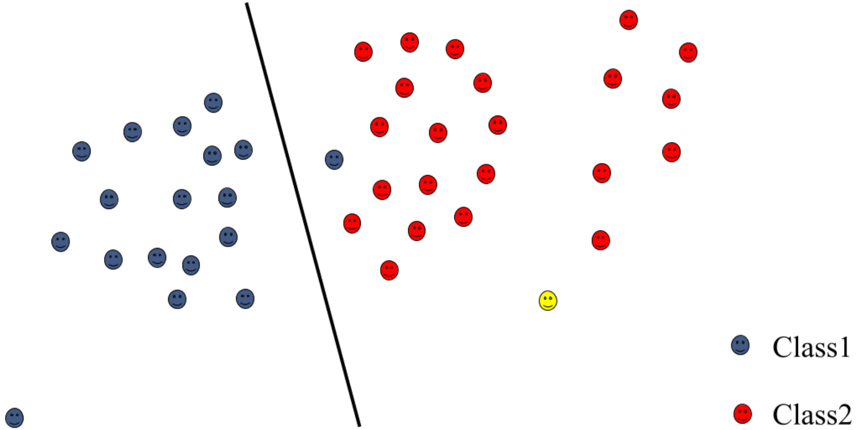
Supervised classification: labels known



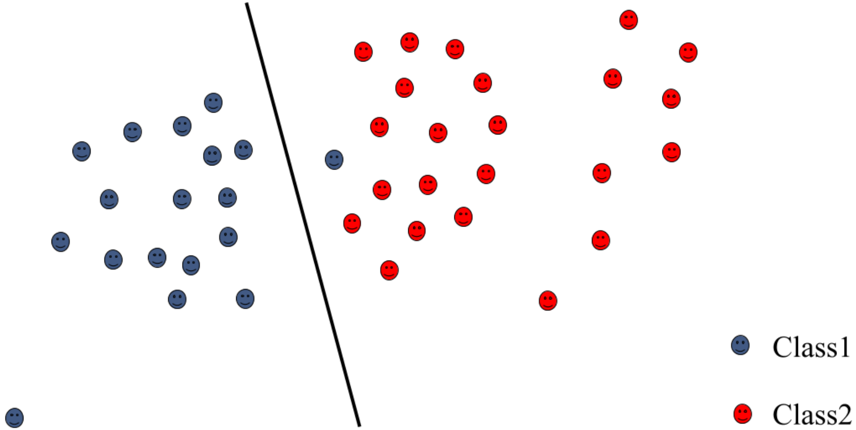
Your classifier determines a boundary



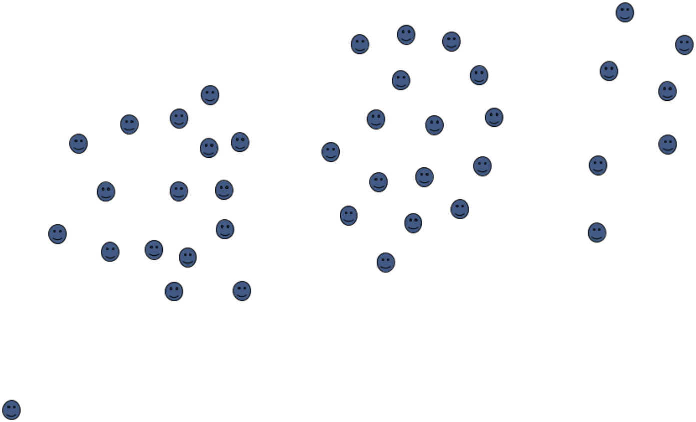
An unseen case



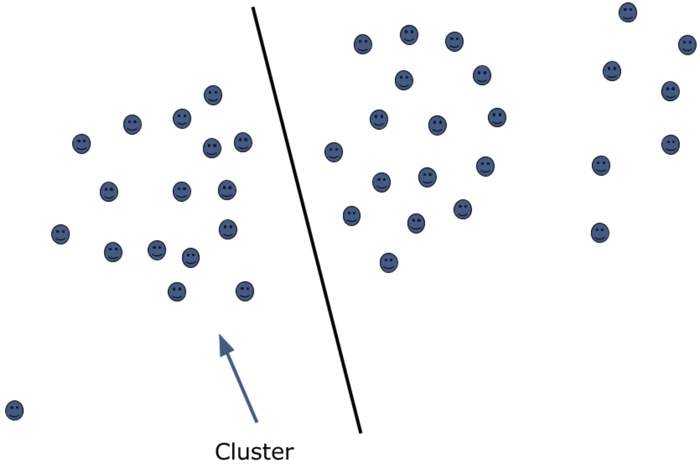
Successfully classified



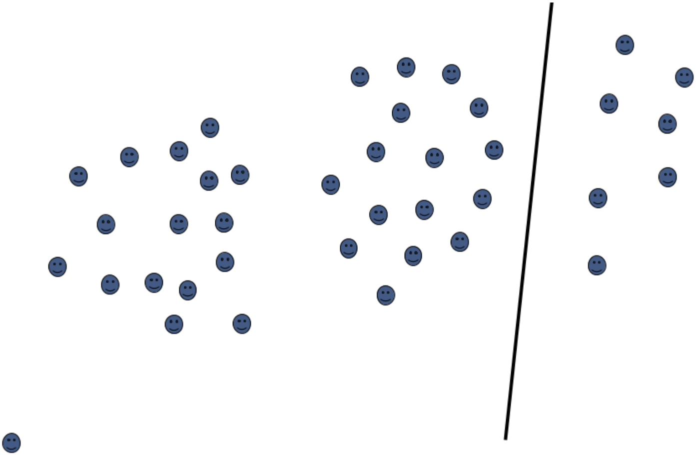
Clustering: No labels!



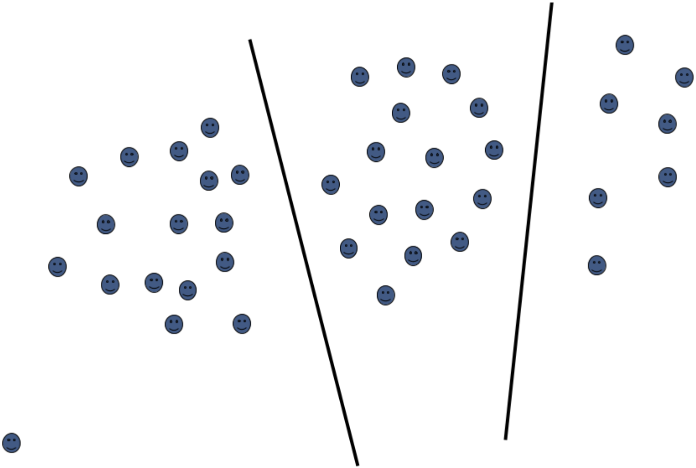
One possible boundary



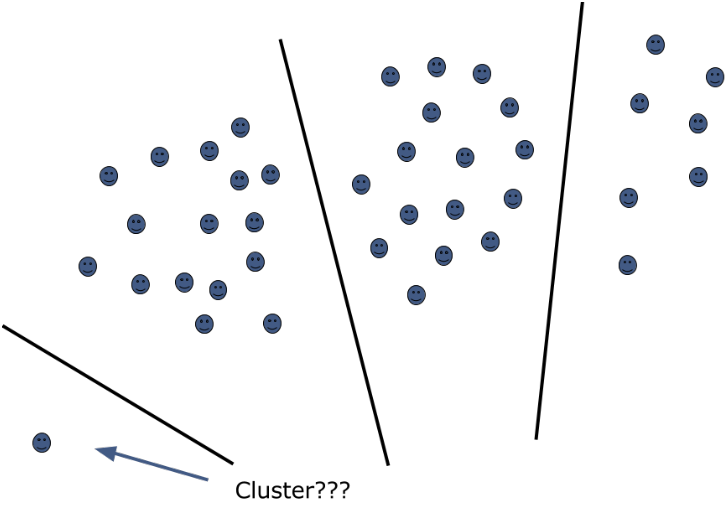
Another possible boundary



Any number of clusters possible

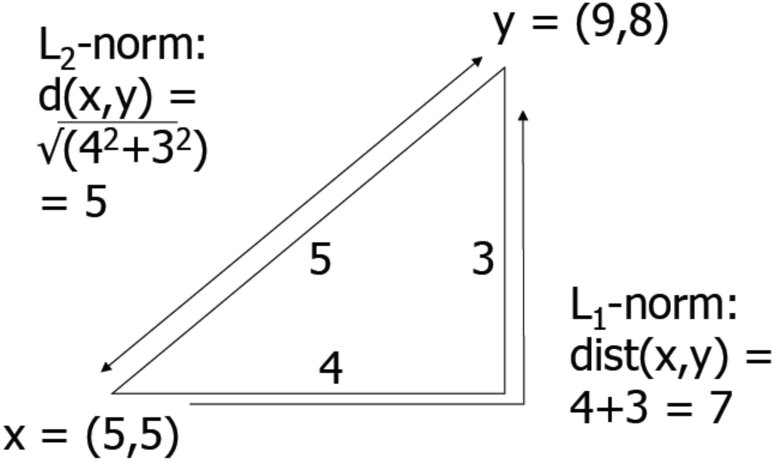


Should we have a cluster for this point?



Outline

Euclidean distances



Axioms of a distance measure

d is a *distance measure* if it is a function from pairs of points to the real numbers such that:

$d(x,y) \geq 0$	(*nonnegativity*)
$d(x,y) = 0$ if and only if $x = y$	(*identity of indiscernables*)
$d(x,y) = d(y,x)$	(*symmetry*)
$d(x,y) \leq d(x,z) + d(z,y)$	(*triangle inequality*)

Distance measures

L_1 distance (Manhattan distance)

$$d_1(\vec{x}, \vec{y}) = \sum_{k=1}^{K} |x_k - y_k|$$

L_2 distance (Euclidean distance)

$$d_2(\vec{x}, \vec{y}) = \sqrt{\sum_{k=1}^{K} |x_k - y_k|^2}$$

r^2 distance (Euclidean squared distance)

$$r^2(\vec{x}, \vec{y}) = \sum_{k=1}^{K} |x_k - y_k|^2$$

L_∞ distance (maximum distance)

$$d_\infty(\vec{x}, \vec{y}) = \max_k(|x_k - y_k|)$$

Example

Calculate the distance between $\vec{x} = \begin{pmatrix} 3 \\ -1 \\ 0 \\ 3 \end{pmatrix}$ and $\vec{y} = \begin{pmatrix} 1 \\ 2 \\ 2 \\ 1 \end{pmatrix}$

Use all four distance measures introduced on the previous slide.

Other distance measures

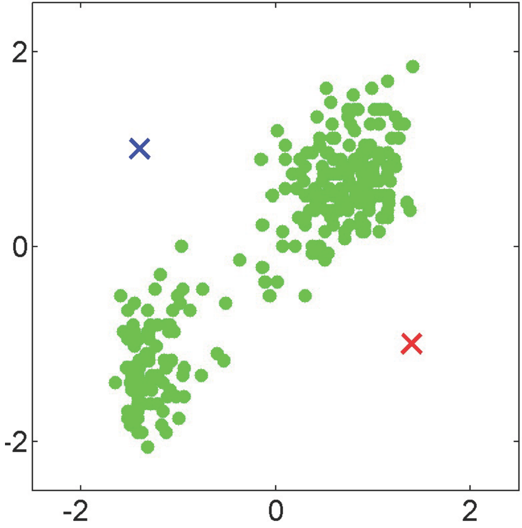
- Cosine*
- Edit distance
- Jaccard
- Kernels

Outline

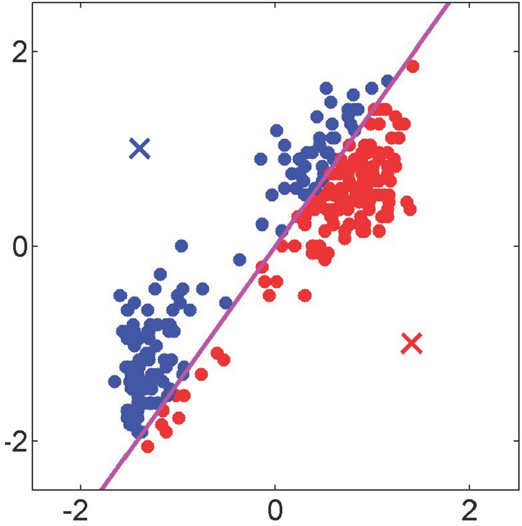
The K-means algorithm

1. For each cluster, decide on a mean.
2. Assign each data point to the nearest mean.
3. Recalculate means according to assignments.
4. If some mean has changed, go back to step 2

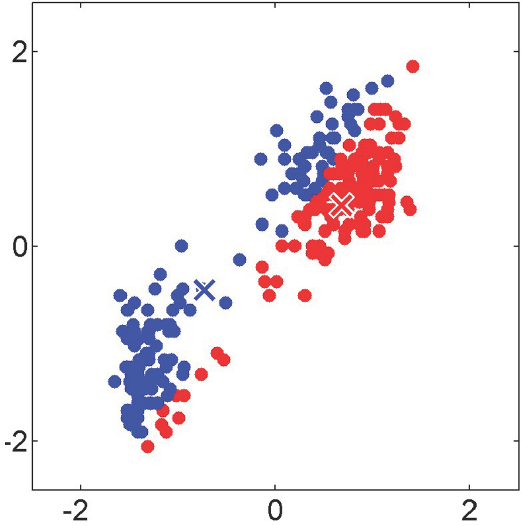
K-means illustration



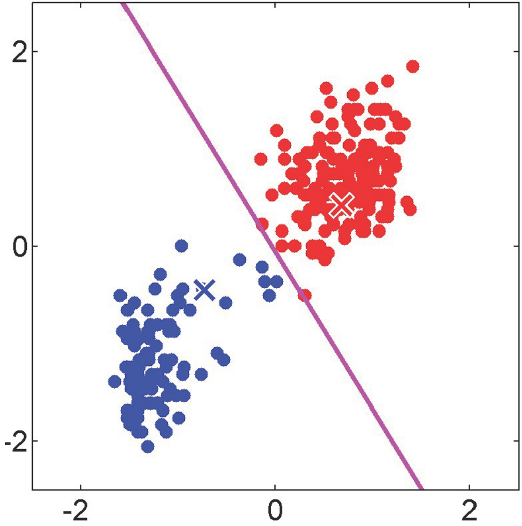
K-means illustration



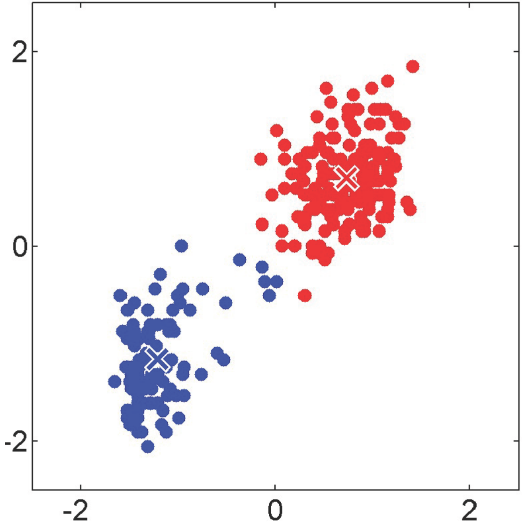
K-means illustration



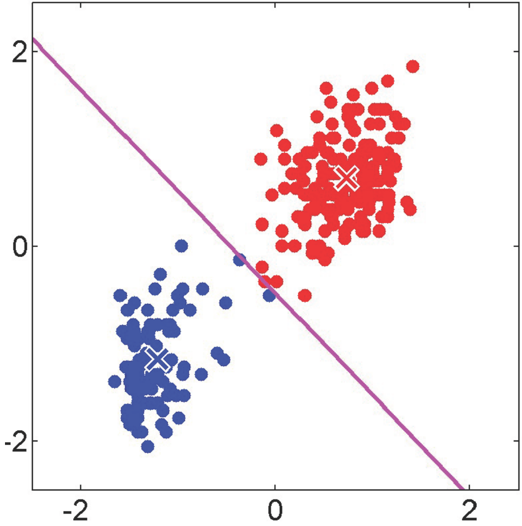
K-means illustration



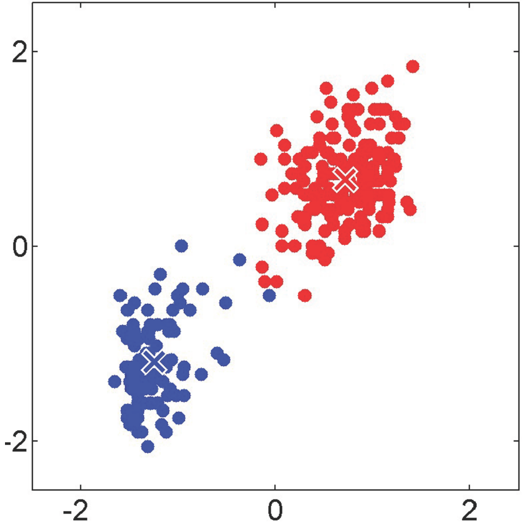
K-means illustration



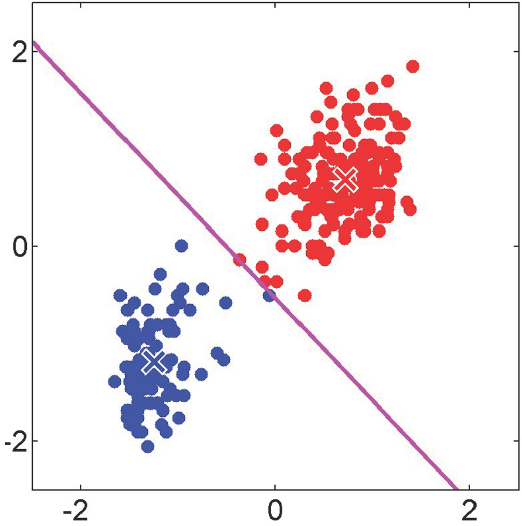
K-means illustration



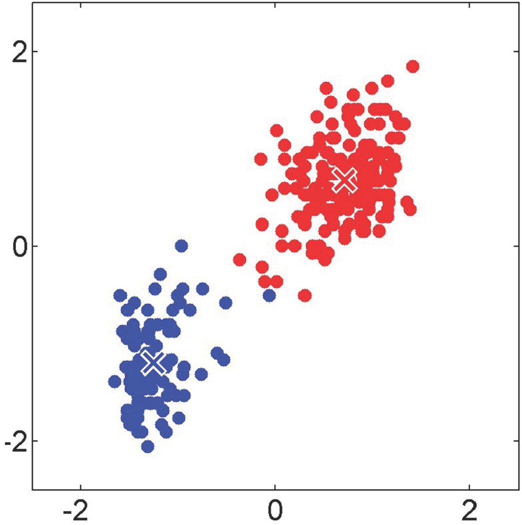
K-means illustration



K-means illustration



K-means illustration



Assigning points to clusters

$$r_{n,k} = \begin{cases} 1 & \text{if } k = \operatorname*{argmin}_j d(\vec{x}_n, \vec{\mu}_j) \\ 0 & \text{otherwise} \end{cases}$$

$\vec{x}_n$: n^{th} training sample (vector)

$\vec{\mu}_j$: mean of the j^{th} cluster

$d(\vec{x}_n, \vec{\mu}_j)$: distance (your choice, e.g. L_2)

Example

$$r_{n,k} = \begin{cases} 1 & \text{if } k = \operatorname*{argmin}_j d(\vec{x}_n, \vec{\mu}_j) \\ 0 & \text{otherwise} \end{cases}$$

See white board

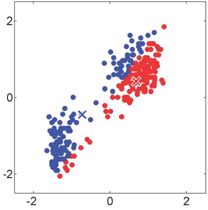
Update mean

$$\vec{\mu}_k = \frac{\sum_{n=1}^{N} r_{n,k}\vec{x}_n}{\sum_{n=1}^{N} r_{n,k}}$$

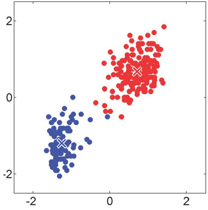
Interpret the denominator

Loss function: distortion measure

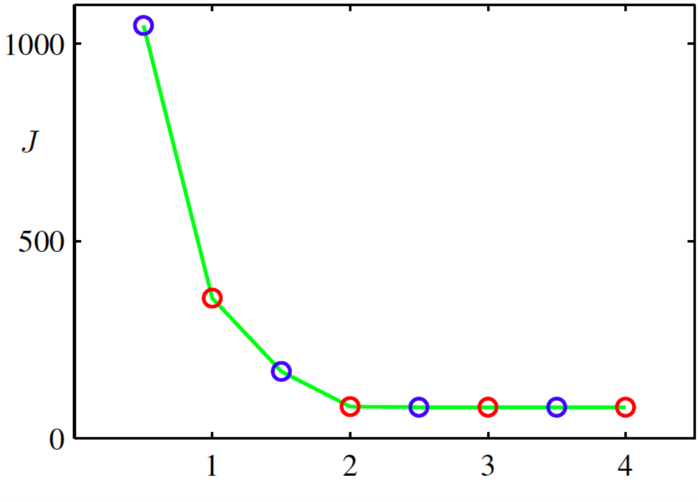
$$J = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{n,k} d(\vec{x}_n, \vec{\mu_k})$$



Which of these has the smaller J?



Distortion function after each iteration



How to initialize K-means

- Converges to local optimum
- Outcome of clustering depends on initialization
- Heuristic: pick K vectors from training data being farthest apart

Outline

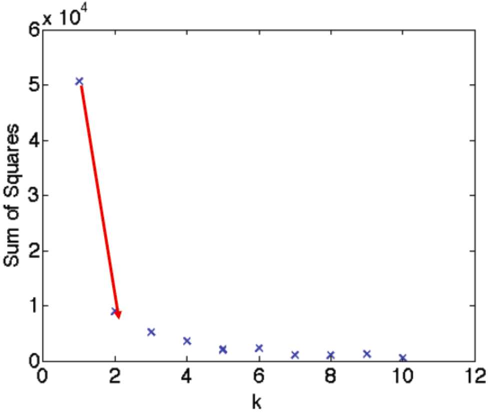
Determining K from the distortion

$$J = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{n,k} d(\vec{x}_n, \vec{\mu}_k)$$

What about picking K such that J becomes as small as possible?

How to determine K

- For $K = N$, the distortion $J = 0$
- Solution: find a “corner”



Sums of squares

- Assume $d = r^2$ and (only for the next line) $K = 1$.
- Then, $J = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{n,k} d(\vec{x}_n, \vec{\mu}_k) = \sum_{n=1}^{N} r^2(\vec{x}_n, \vec{\mu}_G) = SS_{\text{Total}}$
- $\vec{\mu}_G$ is called the grand mean

- We can decompose $SS_{\text{Total}} = SS_{\text{Between}} + SS_{\text{Within}}$, where

$$SS_{\text{Between}} = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{n,k} d(\vec{\mu}_k, \vec{\mu}_G)$$

$$SS_{\text{Within}} = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{n,k} d(\vec{x}_n, \vec{\mu}_k)$$

The Variance Ratio Criterion (*VRC*)

1. Compute VRC_k for each k

$$VRC_k = \frac{SS_{\text{Between}}}{k-1} \Big/ \frac{SS_{\text{Within}}}{n-k}$$

2.

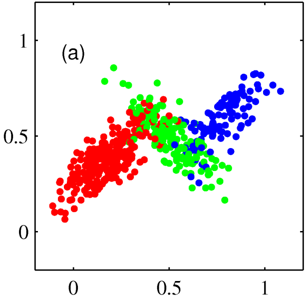
$$\hat{k} = \underset{k}{\operatorname{argmin}}[(VRC_{k+1} - VRC_k) - (VRC_k - VRC_{k-1})]$$

Outline

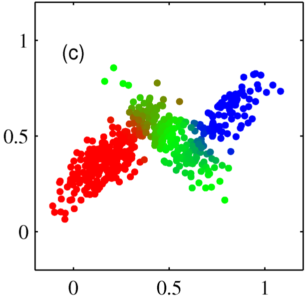
Soft clustering (e.g. Expectation-Maximization)

No strict assignment to a cluster

Just probabilities



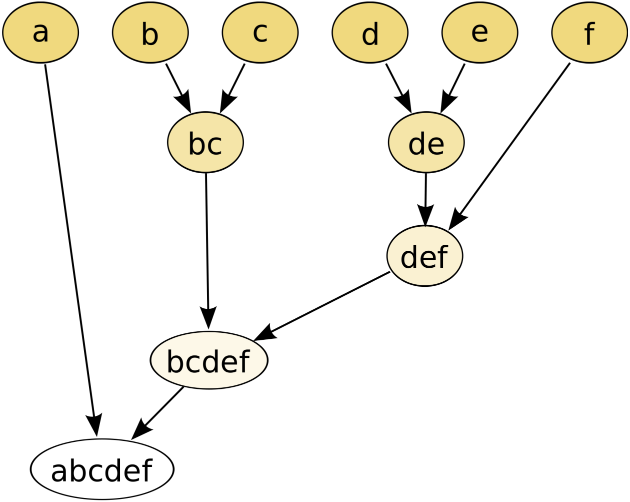
Hard clustering



Soft clustering

Hierarchical clustering (Brown Algorithm)

Organize clusters in a hierarchy



The Exchange Algorithm

g_w: calls of word w

- start with some initial mapping $w \rightarrow g_w$
  - for each word w of the vocabulary do
    - for each class k do
      - tentatively exchange word w from class g_w to class k and update counts
      - compute perplexity for this tentative exchange
    - exchange word w from class g_w to class k with minimum perplexity
- do until stopping criterion is met

Outline

Possible features of words

- Frequency
- TF-IDF
- Stop wording?
- Stemming?

Idea

- Cluster words together that have similar neighbors
- Minimize perplexity on training test

Example clustering

Cluster	Example members
1	Groß, Rau, Muller, Zimmermann, Frei, Becker, Schmidt
2	Düsseldorf, Berlin, München, Köln, Stuttgart, Hannover
3	nahmen, macht, zeigt, gleichen, bringt, biete, machte, enthält

Class labels as features (1/2)

Training

Word	Class label	Tag
Düsseldorf	C2	City
is	X	0
the	X	0
capital	X	0
of	X	0
NRW	X	0

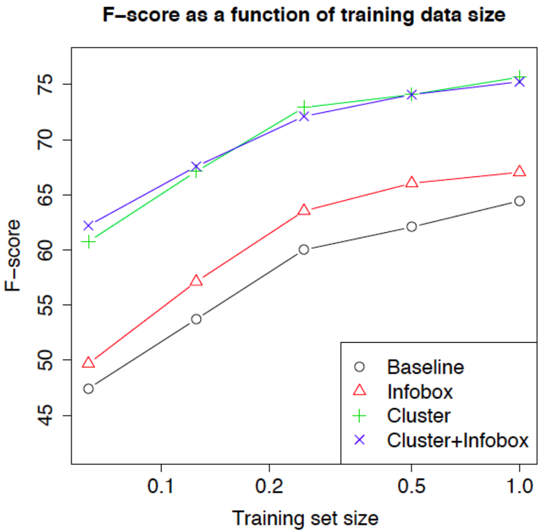
Class labels as features (2/2)

Testing

Word	Class label	Tag
The	X	0
Hofbräuhaus	X	0
is	X	0
in	X	0
Munich	C2	???

How to tag if Munich is not in the training data?

Results


F–score as a function of training data size
F–score
45
50
55
60
65
70
75
Training set size
0.1
0.2
0.5
1.0
Baseline
Infobox
Cluster
Cluster+Infobox

Summary of topics

1. Clustering examples
2. Unsupervised learning
3. Distance measures
4. *K*-means clustering
5. The Variance Ratio Criterion
6. Other clustering algorithms
7. Application to Named Entity Tagging