GÖTEBORGS UNIVERSITET

Statistical Methods for NLP
LT 2202

Clustering with the *k*-means algorithm

March 6, 2014

Richard Johansson

Unsupervised learning

- So far, we have seen how to train classifiers by learning from hand-tagged training sets
- What if there are no tagged data?
- Examples:
  - Discover categories of documents
  - Discover syntactic relations
- This is called **unsupervised learning** or **clustering**

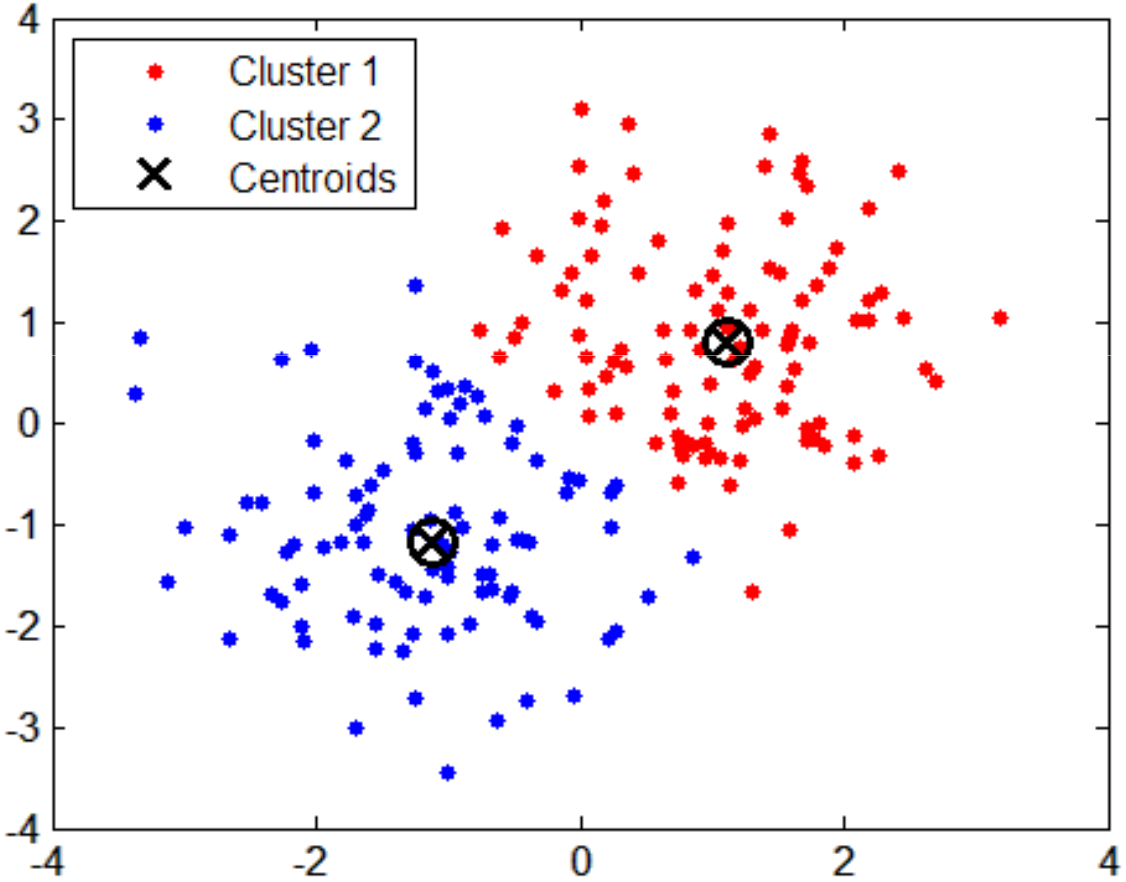
Preliminaries

- We have a collection of objects (e.g. documents)
- Assume that each object can be represented as a point in a geometric space
- Typically the points are constructed by using word frequencies (bag of words)
  - See appendix for details

k-means clustering

- Assume that there are *k* classes
- For every class, create a **centroid**: a point that is in the center of the class
- Find centroids so that all the points in each class are as near as possible
- Computationally hard to do exactly

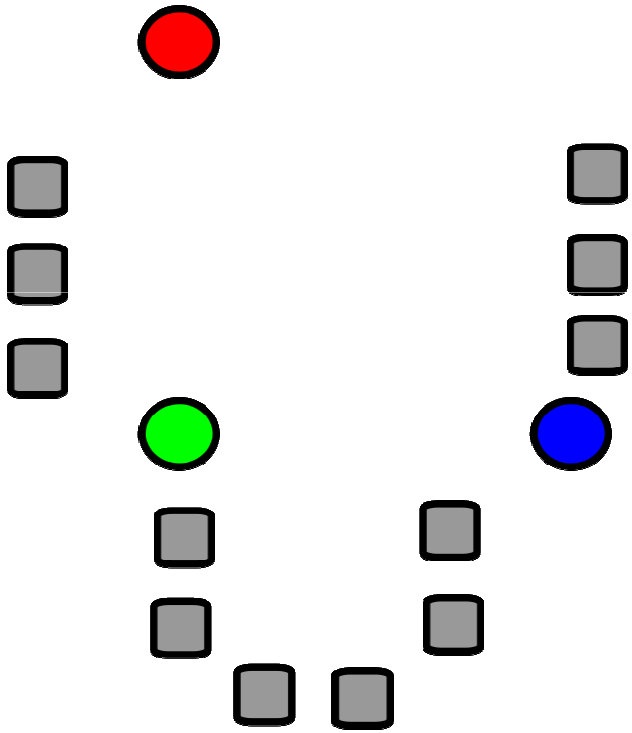
k-means clustering



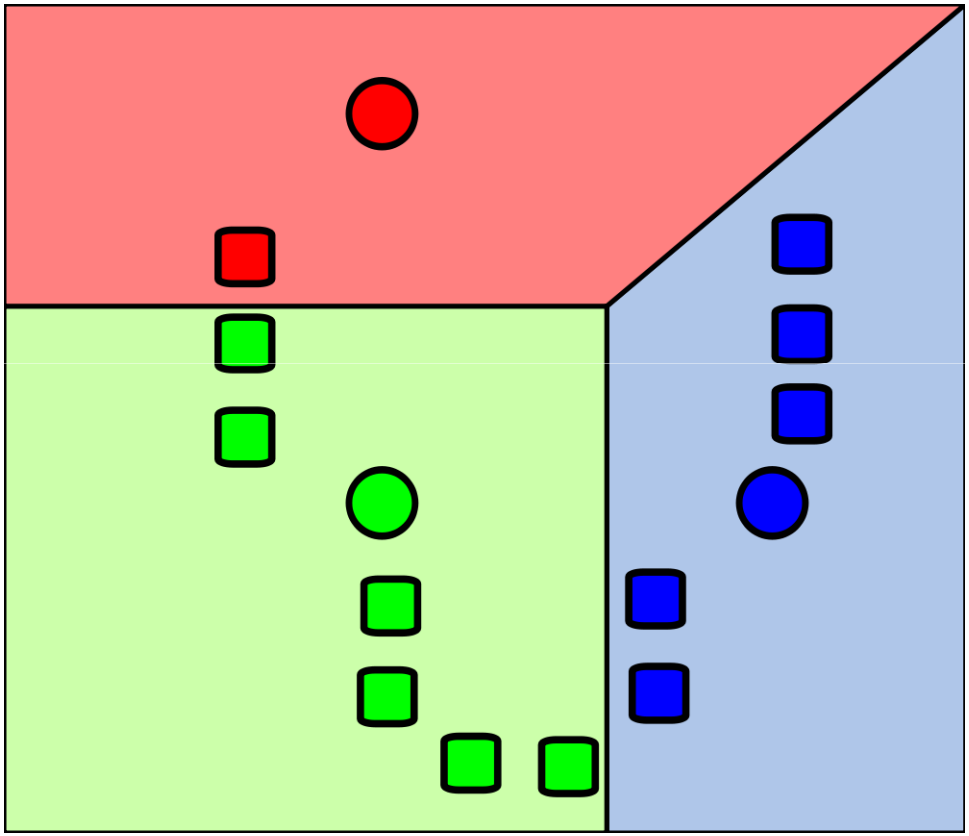
k-means clustering: iterative approach

1. Start with a random assignment into clusters
2. Compute centroids of each cluster
3. Assign each point to the cluster of the nearest centroid
4. If any change, repeat from step 2

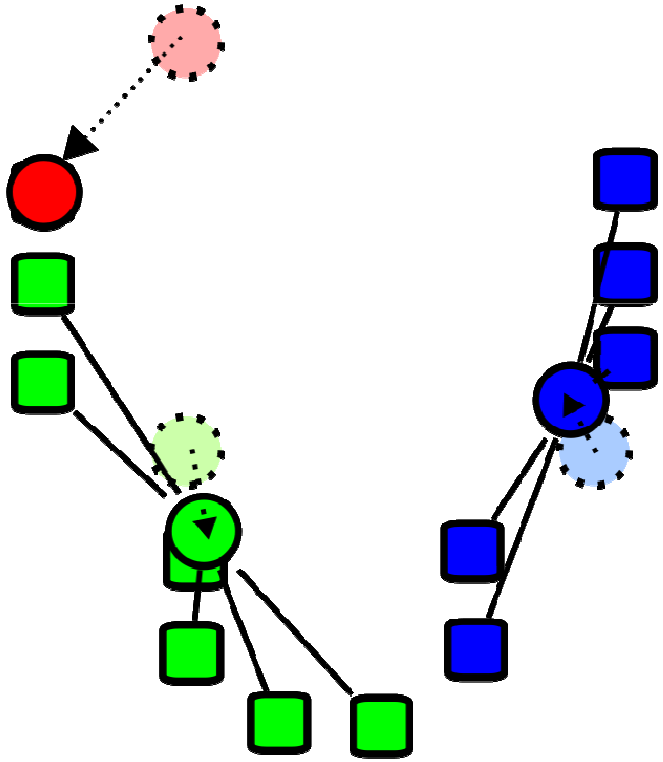
K-means example (1)



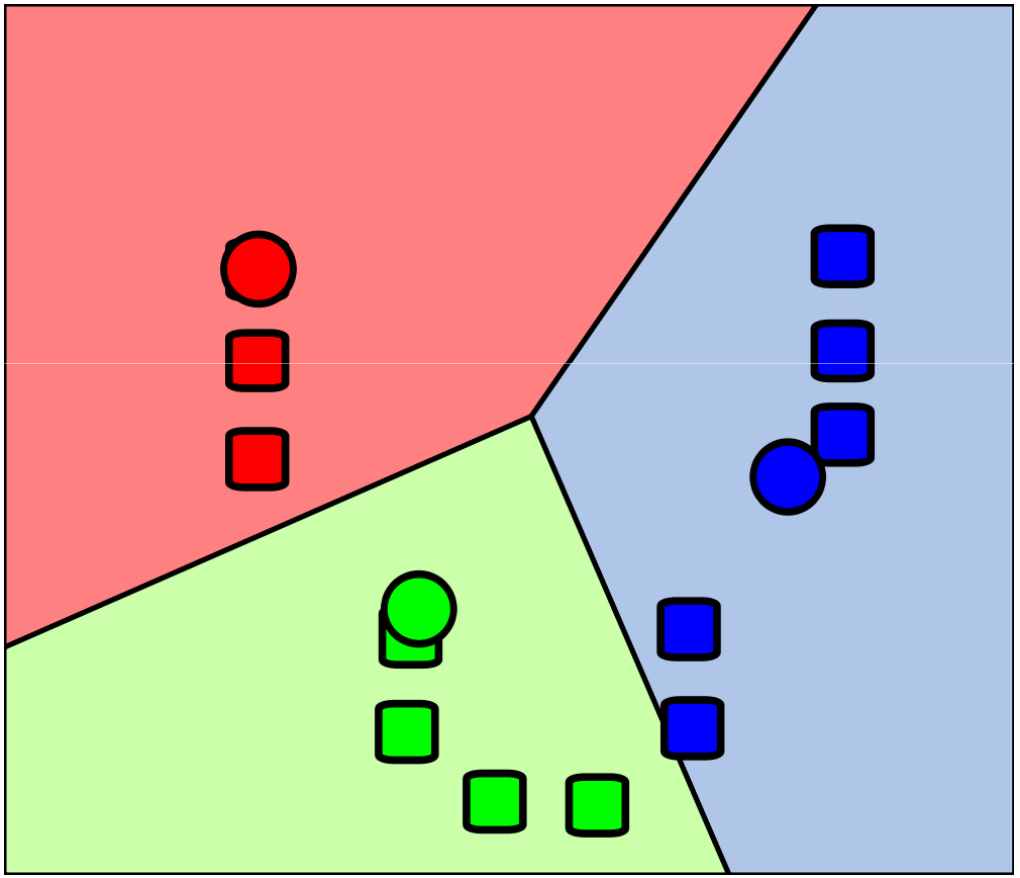
K-means example (2)



K-means example (3)



K-means example (4)



Implementations

- *k*-means is simple and popular and is implemented in many software libraries
  - –NLTK
  - –Scikit-learn

GÖTEBORGS UNIVERSITET

Appendix

Bag of words

- In a **bag-of-words** representation, we assign one dimension for each word in the vocabulary
- To represent a document, we build a list of word frequencies:

[0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 2, 0, 0, 0, 0, 0, ...

- Very often used in document classification

Bag of words: example

"The same year, the Company bought the Waterstone's chain of bookshops"

- Word frequencies in this text:

the: 3
same: 1
year: 1
...

[0, 0, 0, 0, 0, 0, 3, 0, 0, 0, 0, 1, 0, 1, 0, ...]

Implementation note: sparse vectors

- In NLP, most features are very rare
  - –Word, bigram features
- Assume that the feature list is

  [0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 2, 0, 0, 0, 0, 0]
- It is more efficient to use a sparse representation:

  Index list: [6, 12]

  Value list: [1, 2]

  Alternatively: list of pairs: [(6, 1), (12, 2)]