

Lecture 24: Unsupervised learning

CSI 5v93: Introduction to machine learning

Baylor University
Computer Science Department

Dr. Greg Hamerly
<http://cs.baylor.edu/~hamerly/>

CSI 5v93: Introduction to machine learning, Lecture 24 – p. 1/15

Questions?

CSI 5v93: Introduction to machine learning, Lecture 24 – p. 2/15

Unsupervised learning

Unsupervised learning is learning without a training signal.

It can be very different than supervised learning

- no “correct answer”
- goal is different – the concept to learn is not given with the training data
- overfitting and underfitting are still possible, but it’s not as clear when they occur

Data clustering (4.3)

Several views/motivations:

- identify and group items which are similar into several groups
- identify and divide items which are different into several groups
- summarize the data with several prototypes

There are several types of clustering algorithms:

- hierarchical bottom-up
- hierarchical top-down
- iterative “flat” clustering (e.g. k -means and Gaussian EM)
- spectral clustering
- density-seeking (e.g. mean-shift)

Most clustering algorithms are iterative in some fashion.

The k -means algorithm

A “flat”, iterative improvement clustering algorithm.

Very popular: easy to implement, fairly fast algorithm, provides good results.

Input: a set of n data points in d dimensions.

Goal: find k prototypes, or centers, that represent the data “well”.

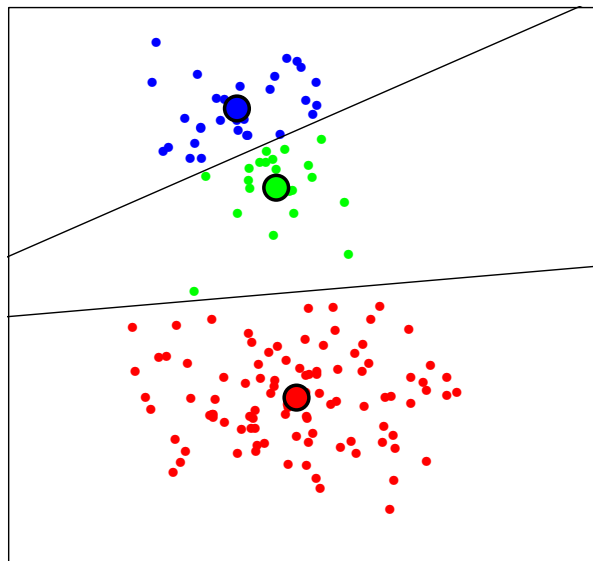
The centers are *not* constrained to be part of the input.

Each center represents the data that is nearest to it.

CSI 5v93: Introduction to machine learning, Lecture 24 – p. 5/15

The k -means algorithm

Iteration Number 20



This is an example of using k -means with $k = 3$ clusters on 2-dimensional data.

CSI 5v93: Introduction to machine learning, Lecture 24 – p. 6/15

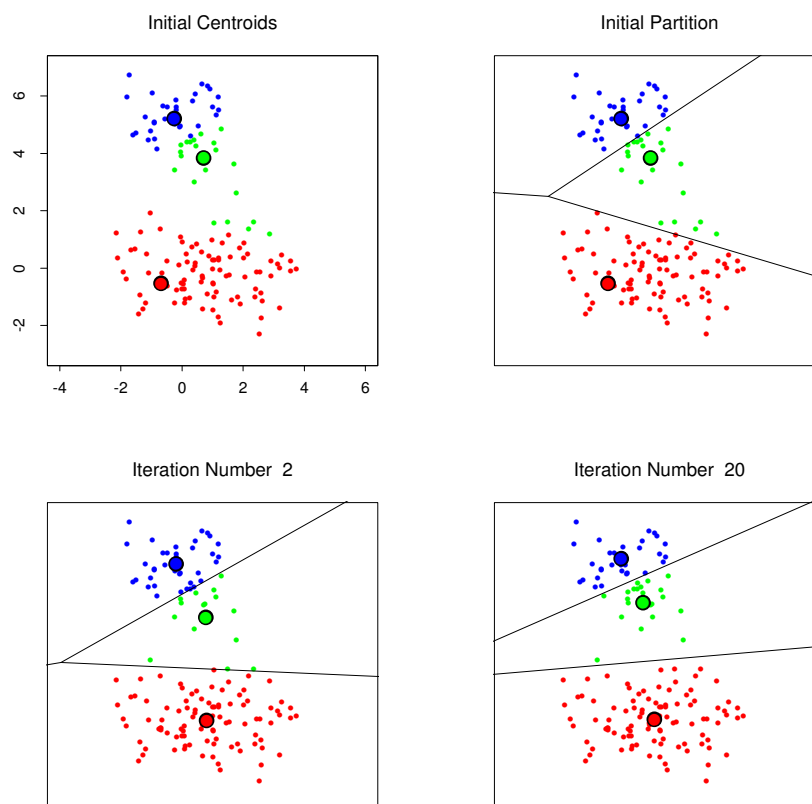
The k -means algorithm

Basic algorithm:

- choose the number of cluster centers k
- position the k centers
- iterate until no change:
 - assign each data point to the cluster center nearest to the point
 - move each cluster center to the average of the points assigned to it

CSI 5v93: Introduction to machine learning, Lecture 24 – p. 7/15

Example: k -means with $k = 3$ on 2d data



CSI 5v93: Introduction to machine learning, Lecture 24 – p. 8/15

Answering questions from last lecture

1. How do you define overfitting or underfitting for unsupervised learning? Is it the same as for supervised learning?
2. What if the input data are not numerical data? How do you calculate the distances between inputs?
3. If we draw boundaries based on the k -means solution, will the boundaries always separate the data points perfectly?

Answering questions from last lecture (continued)

1. Is it possible to have some data points which are classified wrongly?
2. In higher dimensional problems, can we also say that the boundaries are linear?
3. In the example of three k -means clusters, where the linear boundaries meet, a new boundary is formed. How do you know where that boundary is formed?

Reminder: the k -means quality criterion

k -means is actually minimizing the within-cluster sum-of-squared-distances.

Let $\delta(i, j)$ indicate that datapoint x_i belongs to cluster center c_j . Then k -means minimizes

$$\begin{aligned} F(X, C) &= \sum_{i=1}^n \sum_{j=1}^k \|x_i - c_j\|^2 \delta(i, j) \\ &= \sum_{i=1}^n \sum_{j=1}^k \delta(i, j) \sum_{m=1}^d (x_{im} - c_{jm})^2 \end{aligned}$$

Remaining questions in k -means

- How do we choose k ?
- How do we initialize the cluster centers?

Choosing k

How do we choose k ?

- *Best answer*: if you know how many clusters you expect or want, use that.
- Try several different k , and find the sum-of-squared distances for each clustering, and plot the values, and look for a significant dip.
- Regularization methods: minimum description length (MDL), complexity penalty (e.g. Bayesian Information Criterion, or BIC).
- Statistical tests for clusters (e.g. G-means).
- Predictive methods: divide the input data into two or more parts, cluster separately, and compare the clusterings.
- Many other heuristics.

Initializing the clusters

How do we initialize the cluster centers?

- Randomly: choose k different locations
- Randomly: choose k different input datapoints
- Randomly: assign each input data points randomly to one of k clusters, and place each cluster center at the mean of its assigned data points
- Furthest-first: choose the furthest datapoint from the current set of centers

No matter which method is used, there is some randomness involved.

Practical advice: for fixed k , run the algorithm as many times as possible with different random initializations, and take the one that achieves the best k -means score.

2-minute journal

Please write a response to the following on a piece of paper and hand it in immediately. Please make it anonymous (no names). Write about:

- major points you learned today
- areas not understood or requiring clarification