Lecture: Machine Learning for Data Science

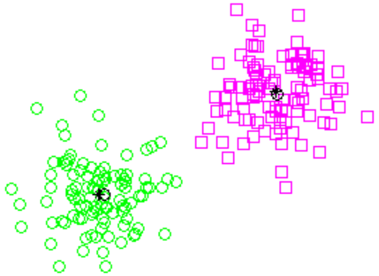
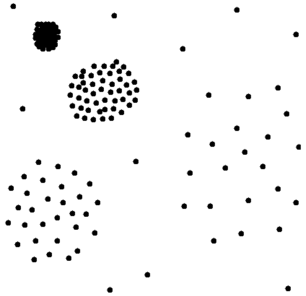
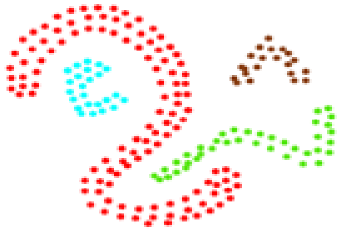
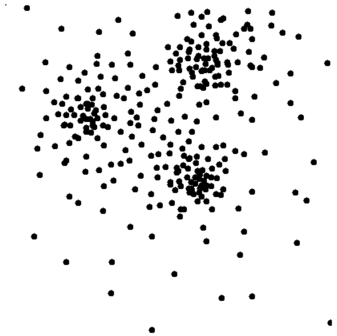
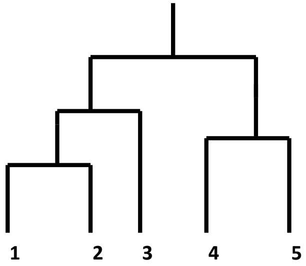
Winter semester 2021/22

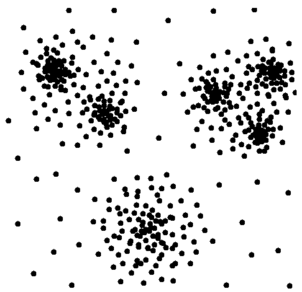
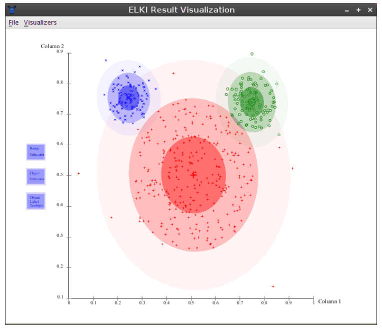
Lecture 13: Unsupervised learning –EM clustering

Prof. Dr. Eirini Ntoutsi

Clustering topics covered in this lecture

- Partitioning-based clustering
  - k-Means, k-Medoids
- Hierarchical clustering
- Density-based clustering
- Grid-based clustering
- Soft clustering
- Clustering evaluation

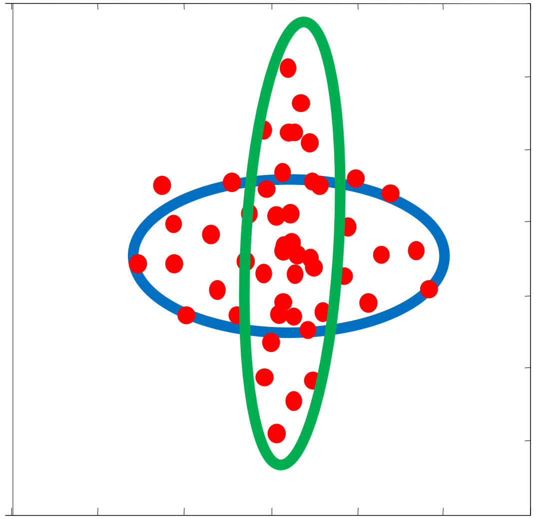



Outline

- Soft vs Hard clustering
- EM-clustering
- Things you should know from this lecture & reading material

Soft clustering

- What if clusters are overlapping?
  - Hard to tell which cluster is right for an instance
  - Maybe we should try to remain uncertain
- Soft clustering allows instances to belong to more than one clusters
  - the membership probabilities must sum to 1.0
- Mixture models are a probabilistically-grounded way of doing soft clustering
  - Each cluster corresponds to a probability distribution
    - Typically a Gaussian or multinomial
  - the parameters of the distribution comprise the cluster description
- How do we find the parameters of the distributions?
  - Expectation Maximization (EM) algorithm [Dempster, Laird and Rubin, 1977]

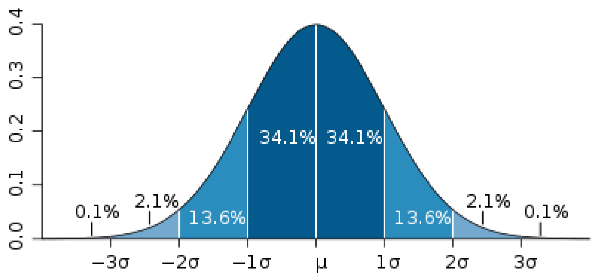


Gaussian Mixture Models

- Let a dataset D of instances to be clustered
  - In the general case, instances are d-dimensional vectors $x = (x_1, \ldots, x_d)$
- Each cluster is represented via a Gaussian distribution
- So D is generated via a mixture of k Gaussian distributions

Univariate case (Univariate Gaussian)

- Each cluster is represented via a Gaussian distribution
- In the simplest case, we assume univariate instances
- Each cluster c is a univariate Gaussian distribution described via
  - mean μ_c (expected value in c)
  - variance σ_c^2 (the average of the squared differences from the mean)
    - standard deviation σ_c (the square root of variance)



Source: http://en.wikipedia.org/wiki/Normal_distribution

- Probability density function of a Gaussian distribution

$$P(x \mid c) = \frac{1}{\sqrt{2\pi\sigma_c^{\,2}}} e^{-\frac{1}{2}\cdot(x-\mu_c)^2/\sigma_c^{\,2}}$$

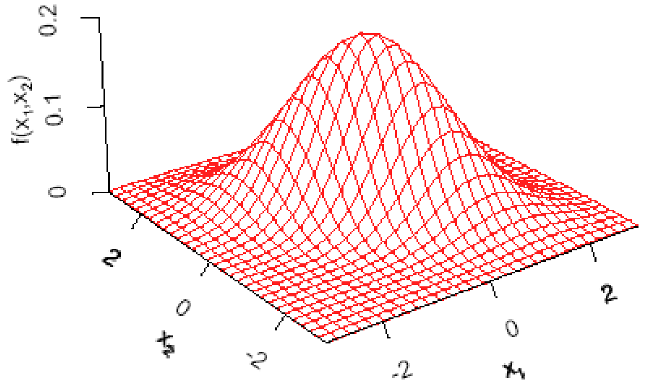
- For a numerical feature X, for which we have a sample $x_1,\ldots,x_n$ we can easily compute the parameters of the Gaussian

$$\mu = \frac{1}{n}\cdot\sum_{i=1}^{n} x_i \qquad \sigma^2 = \frac{1}{n}\sum_{i=1}^{n}(x_i-\mu)^2$$

(See also Lecture 2 on basic data descriptors)

Multivariate case (Multivariate Gaussian)

- Each cluster is represented via a Gaussian distribution
- In the general case, instances are *d*-dimensional vectors $x = (x_1, \ldots, x_d)$



- Each cluster *c* is a multivariate Gaussian distribution represented via
  - mean μ_c (expected value of each attributes *i* in *c*) → this is a vector
  - $d \times d$ covariance matrix Σ_c (how correlated are attributes (*i,j*) in *c*)
    - $|\Sigma_c|$ matrix determinant
- Probability density function of a Gaussian distribution

$$P(x|c) = \frac{1}{\sqrt{(2\pi)^d |\Sigma_c|}} e^{-\frac{1}{2}\cdot(x-\mu_c)^T \cdot \Sigma_c^{-1} \cdot (x-\mu_c)}$$

Multivariate case (Multivariate Gaussian)

- For a dataset *D* of *K*-dimensional instances, we can compute the parameters of the Gaussian
- μ is the mean vector

$$\mu = [\mu^1, \mu^2, \ldots, \mu^d]$$

  - where μ^j is the mean value w.r.t. dimension j

$$\mu^j = \frac{1}{n} \cdot \sum_{i=1}^{n} x^j{}_i$$

$$\mathbf{D} = \begin{array}{c|cccc} & X_1 & X_2 & \cdots & X_d \\ \hline \mathbf{x}_1 & x_{11} & x_{12} & \cdots & x_{1d} \\ \mathbf{x}_2 & x_{21} & x_{22} & \cdots & x_{2d} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{x}_n & x_{n1} & x_{n2} & \cdots & x_{nd} \end{array}$$

- *d x d* covariance matrix *Σ* whose *(i,j)* entry is the covariance between attributes *i* and *j*
- The covariance of two random variables *x* and *y* is given by

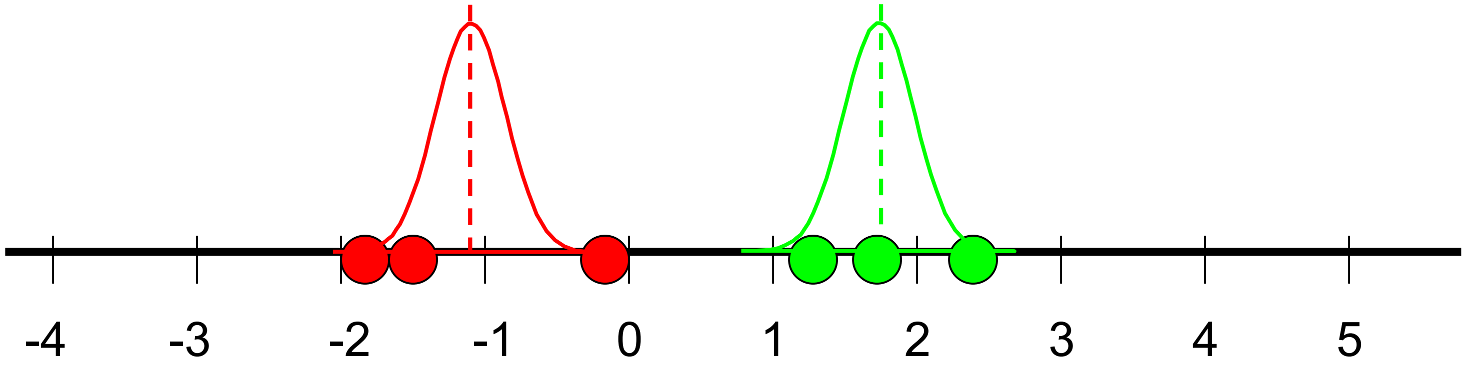
$$\sigma(x, y) = \frac{1}{n-1} \sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y})$$

$$\begin{pmatrix} \sigma(x, x) & \sigma(x, y) \\ \sigma(y, x) & \sigma(y, y) \end{pmatrix}$$

- The variance σ^2 of a variable *x* can be expressed as *σ(x,x)*

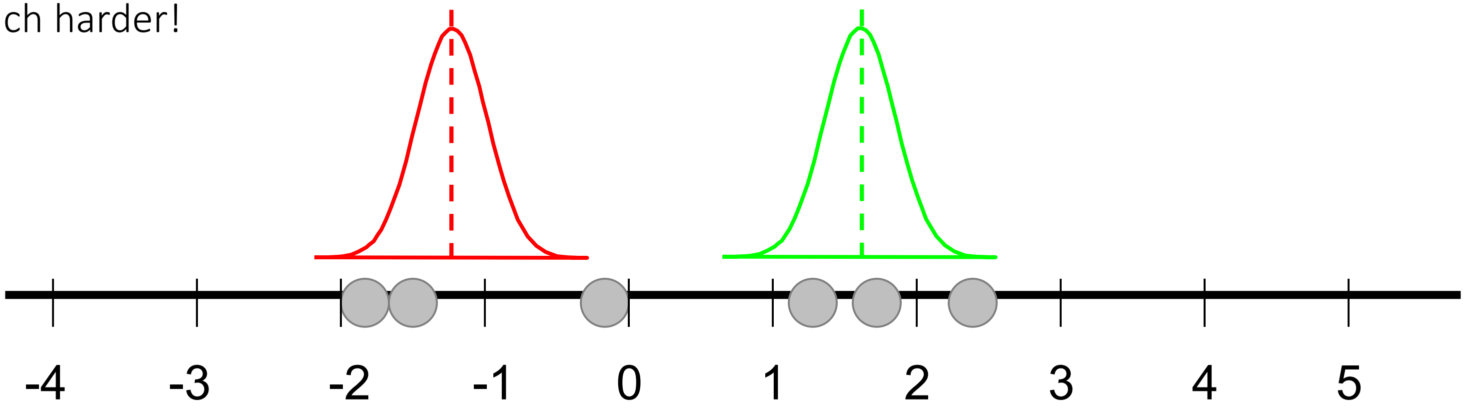
How to compute the parameters of the Gaussians: A small example

- Suppose we have 6 points & we know they come from 2 Gaussian models (red, green)
  - Can we estimate the parameters of the Gaussian?
    - Yes, based on the point assignments



How to compute the parameters of the Gaussians: A small example

- What if we don't know which points came from which source? We still know that the points came from 2 Gaussian sources, but we don't know the assignments
  - Can we estimate the parameters of the Gaussian?
    - Much harder!



- What if we knew the parameters of the Gaussians (μ_c ,Σ_c)?
  - Could we guess whether a point is more likely to come from the red or green Gaussian?
    - Yes!

$$P(x \mid c) = \frac{1}{\sqrt{(2\pi)^d \, |\Sigma_c|}} e^{-\frac{1}{2}\cdot(x-\mu_c)^T \cdot \Sigma_c^{-1} \cdot (x-\mu_c)}$$

Outline

- Soft vs Hard clustering
- EM-clustering
- Things you should know from this lecture & reading material

Expectation Maximization (EM) idea

- Chicken-egg problem
  - We need the parameters of the green and red Gaussians to guess the source of each point
  - We need to know the sources of the points to estimate the parameters of the Gaussian sources

- EM algorithm
  - Start with two randomly placed Gaussians (μ_c, Σ_c)
  - Two alternating steps:
    - E-step ("Expectation"):
      - For each instance, what is the probability of coming from each Gaussian (under the current estimate of the model)?
      - This is a soft assignment (probability)
    - M-step ("Maximization"):
      - re-estimate the model parameters (μ_c, Σ_c) based on the assignments
  - Until convergence

Gaussian Mixture Models

- Let *D* be a dataset of *d*-dimensional instances.
- Let *k* sources, $\{c_1, c_2, \ldots, c_k\}$
- If the instances in *D* are generated in an independent manner from the *k* sources, the probability of the dataset D (|D|=N) is just the product of the probabilities of each instance x_i in *D*:

$$\mathcal{L} = \prod_{i=1}^{N} \mathrm{P}(x_i) \qquad = \prod_{i=1}^{N} \sum_{l=1}^{k} \mathrm{P}(c_l)\mathrm{P}(x_i|c_l)$$

- We want to find the distribution parameters that maximize the likelihood
- We use the EM algorithm to estimate the parameters

EM algorithm 1/3

- Initialize
- Two alternating steps:
  - E-step (“Expectation”): re-estimate the cluster assignments under the current estimate of the model
  - M-step (“Maximization”): re-estimate the model parameters under the current assignment
- Until convergence

EM algorithm 2/3

- E-step ("Expectation"):
  - re-estimate the cluster assignments under the current estimate of the model
  - For each object, calculate the probability of the object being generated by each cluster c_l

$$\mathrm{P}^{new}(c_l|x_i) = \mathrm{P}(c_l)\mathrm{P}(x_i|c_l)$$

  - where $P(x_i|c_l)$ is given by the probability density function of the Gaussian distribution

EM algorithm 3/3

- M-step (“Maximization”): re-estimate the model parameters under the current assignment
- Cluster density/prior: % of instances coming from c_l

$$\mathrm{P}^{new}(c_l) = \frac{1}{N}\sum_{i=1}^{N}\mathrm{P}^{new}(c_l|x_i)$$

- Cluster means: expected value for each attribute coming from c_l

$$\mu_l^{new} = \frac{\sum_{i=1}^{N} x_i \mathrm{P}^{new}(c_l|x_i)}{\sum_{i=1}^{N}\mathrm{P}^{new}(c_l|x_i)}$$

- Cluster covariances: how correlated are attributes in c_l

$$\Sigma_l^{new} = \frac{\sum_{i=1}^{N}(x_i - \mu_l^{new})(x_i - \mu_l^{new})'\mathrm{P}^{new}(c_l|x_i)}{\sum_{i=1}^{N}\mathrm{P}^{new}(c_l|x_i)}$$

EM (Gaussian Mixture Models) overview

- EM clustering: A soft clustering method
  - membership probabilities of instances to different clusters
- Each cluster corresponds to a probability distribution
  - e.g., Gaussian → Gaussian mixture models
  - the parameters of the distribution comprise the cluster description
- We use EM to find the parameters of the distributions

EM and k-Means

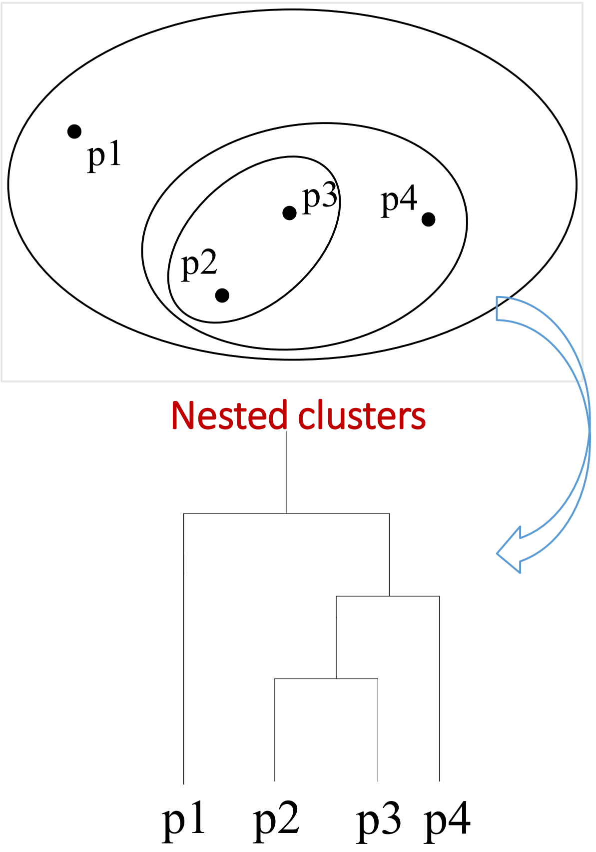
- EM is similar to the k-Means algorithm

- E-step (EM) → assign each object to a cluster step (k-Means)
  - In EM each object is assigned to a cluster with a probability, in k-Means the assignment is hard
- M-step (EM) → compute cluster centroids step (k-Means)
  - In EM, the computation of the mean also considers the fact that each object belong to a distribution with a certain probability, in k-Means hard assignments of points are considered.
- The cluster representation is different
  - In k-Means: each cluster is represented via a centroid
  - In EM: each cluster is represented viaa probability distribution (Gaussian)

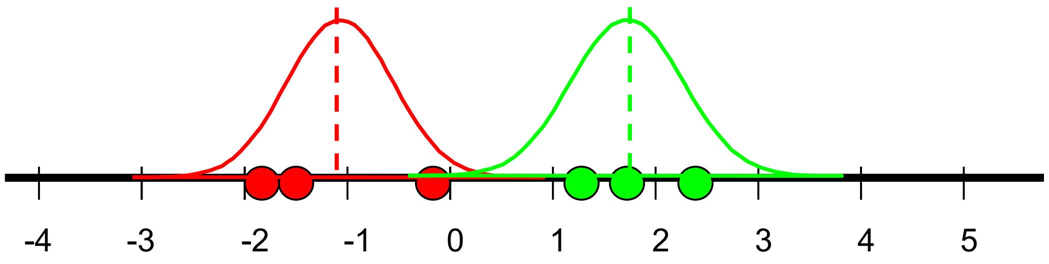
- There exist papers that discuss how are they related, e.g., k-Means is a Variational EM Approximation of Gaussian Mixture Models, https://arxiv.org/pdf/1704.04812.pdf

Soft-clustering vs Hierarchical clustering

Dendrogram



Soft clustering



In hierarchical clustering, an instance belongs to more than one clusters in the hierarchy, still this is a hard assignment.

In soft clustering, an instance belongs to all clusters with some probability. This is a flat clustering.

Outline

- Soft vs Hard clustering
- EM-clustering
- Things you should know from this lecture & reading material

Overview and Reading

- Soft-clustering
- EM clustering

- Reading
  - Tan P.-N., Steinbach M., Kumar V book, Chapter 8.
  - Data Clustering: A Review, https://www.cs.rutgers.edu/~mlittman/courses/lightai03/jain99data.pdf
  - Nando de Freitas youtube video: https://www.youtube.com/watch?v=voN8omBe2r4