ALTERNATIVE METHODS FOR CLUSTERING

K-Means Algorithm

1: Select K points as the initial centroids.
2: **repeat**
3: Form K clusters by assigning all points to the closest centroid.
4: Recompute the centroid of each cluster.
5: **until** The centroids don't change

Termination conditions

- Several possibilities, e.g.,
  - A fixed number of iterations
  - Objects partition unchanged
  - Centroid positions don't change

Convergence of *K*-Means

- Define goodness measure of cluster c as sum of squared distances from cluster centroid:
  - $SSE_c(c,s) = \Sigma_i (d_i - s_c)^2$ (sum over all d_i in cluster c)
  - $G(C,s) = \Sigma_c SSE_c(c,s)$
- Re-assignment monotonically decreases G
  - It is a coordinate descent algorithm (**opt one component at a time**)
- At any step we have some value for G(C,s)

1) Fix s, optimize C → assign d to the closest centroid → G(C',s) < G(C,s)

2) Fix C', optimize s → take the new centroids → G(C',s') < G(C',s) < G(C,s)

The new cost is smaller than the original one → local minimum

Time Complexity: Assign points to clusters

Question

- Assuming the computation of a similarity is linear in the number of attributes |A|, what is the complexity of assigning points to clusters?

Answer

P = the set of points

A = the set of attributes of each point

K = the number of clusters

$$O(k \times |P| \times |A|)$$

Time Complexity: Update centroids

Question

- What is the complexity of updating centroids?

Answer

P = the set of points

A = the set of attributes of each point

K = the number of clusters

$$O(|P| \times |A|)$$

Overall Time Complexity

Question

What is the complexity of k-means if t iterations are necessary to converge?

Answer

P = the set of points

A = the set of attributes of each point

K = the number of clusters

$$O(t \times k \times |P| \times |A|)$$

MIXTURE MODELS AND THE EM ALGORITHM

Model-based clustering (probabilistic)

- In order to understand our data, we will assume that there is a generative process (a model) that creates/describes the data, and we will try to find the model that **best fits** the data.
  - Models of different complexity can be defined, but we will assume that our model is a distribution from which data points are sampled
  - **Example**: the data is the height of all people in **Greece**
- In most cases, a single distribution is not good enough to describe all data points: **different parts of the data follow a different distribution**
  - **Example**: the data is the height of all people in Greece and China
  - We need a mixture model
  - Different distributions correspond to different clusters in the data.

Algorithm 9.2 EM algorithm.

1: Select an initial set of model parameters.
(As with K-means, this can be done randomly or in a variety of ways.)

2: **repeat**

3: **Expectation Step** For each object, calculate the probability that each object belongs to each distribution, i.e., calculate $prob(distribution\ j|\mathbf{x}_i, \Theta)$.

4: **Maximization Step** Given the probabilities from the expectation step, find the new estimates of the parameters that maximize the expected likelihood.

5: **until** The parameters do not change.
(Alternatively, stop if the change in the parameters is below a specified threshold.)

EM (Expectation Maximization) Algorithm

- Initialize the values of the parameters in Θ to some random values
- Repeat until convergence
  - E-Step: Given the parameters Θ estimate the membership probabilities $P(G|x_i)$ and $P(C|x_i)$
  - M-Step: Compute the parameter values that (in expectation) maximize the data likelihood

E-Step: Assignment of points to clusters

K-means: **hard** assignment, EM: soft assignment

M-Step: K-means: Computation of centroids EM: Computation of the new model parameters

Gaussian Distribution

- Example: the data is the height of all people in Greece
  - Experience has shown that this data follows a Gaussian (Normal) distribution
  - Reminder: Normal distribution:

$$P(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

  - μ = mean, σ = standard deviation

Gaussian Model

- What is a model?
  - A Gaussian distribution is fully defined by the mean μ and the standard deviation σ
  - We define our model as the pair of parameters $\theta = (\mu, \sigma)$

- This is a general principle: a model is defined as a vector of parameters θ

Fitting the model

- We want to find the normal distribution that best fits our data
  - Find the best values for μ and σ
  - But what does best fit mean?

Maximum Likelihood Estimation (MLE)

- Suppose that we have a vector $X = (x_1, \ldots, x_n)$ of values
- And we want to fit a Gaussian $N(\mu, \sigma)$ model to the data
- Probability of observing point x_i:

$$P(x_i) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x_i-\mu)^2}{2\sigma^2}}$$

- Probability of observing all points (assume independence)

$$P(X) = \prod_{i=1}^{n} P(x_i) = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x_i-\mu)^2}{2\sigma^2}}$$

- We want to find the parameters $\theta = (\mu, \sigma)$ that maximize the probability $P(X|\theta)$

Maximum Likelihood Estimation (MLE)

- The probability $P(X|\theta)$ as a function of θ is called the Likelihood function

$$L(\theta) = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x_i-\mu)^2}{2\sigma^2}}$$

- It is usually easier to work with the Log-Likelihood function

$$LL(\theta) = -\sum_{i=1}^{n} \frac{(x_i-\mu)^2}{2\sigma^2} - \frac{1}{2} n \log 2\pi - n \log \sigma$$

- Maximum Likelihood Estimation
  - Find parameters μ, σ that maximize $LL(\theta)$

$$\mu = \frac{1}{n}\sum_{i=1}^{n} x_i = \mu_X$$

Sample Mean

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

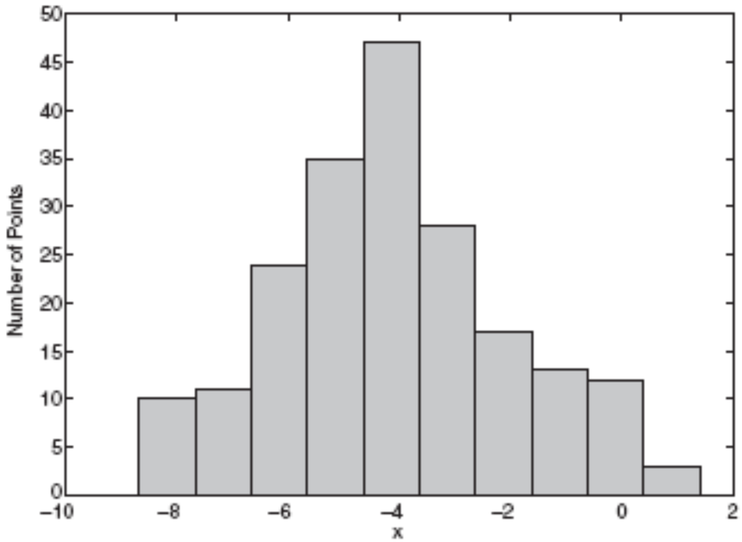
Sample Variance

MLE

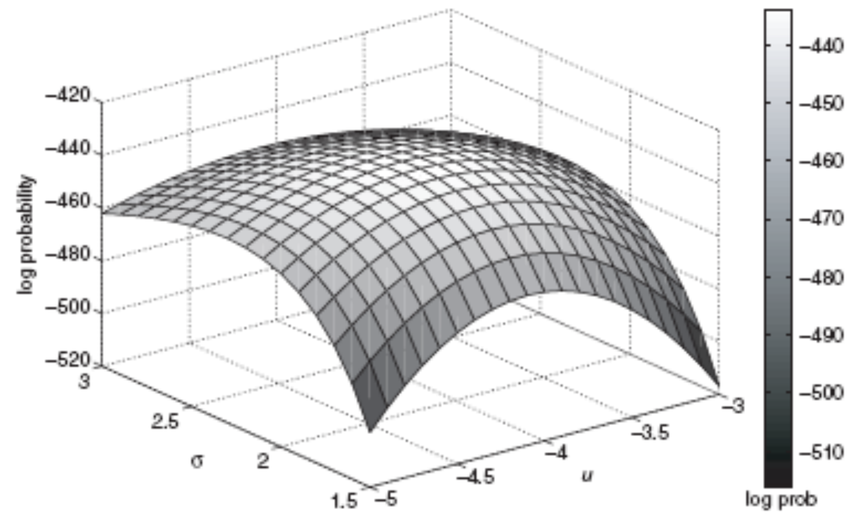
- Note: these are also the most likely parameters given the data

$$P(\theta|X) = \frac{P(X|\theta)P(\theta)}{P(X)}$$

- If we have no prior information about θ, or X, then maximizing $P(X|\theta)$ is the same as maximizing $P(\theta|X)$



(a) Histogram of 200 points from a Gaussian distribution.

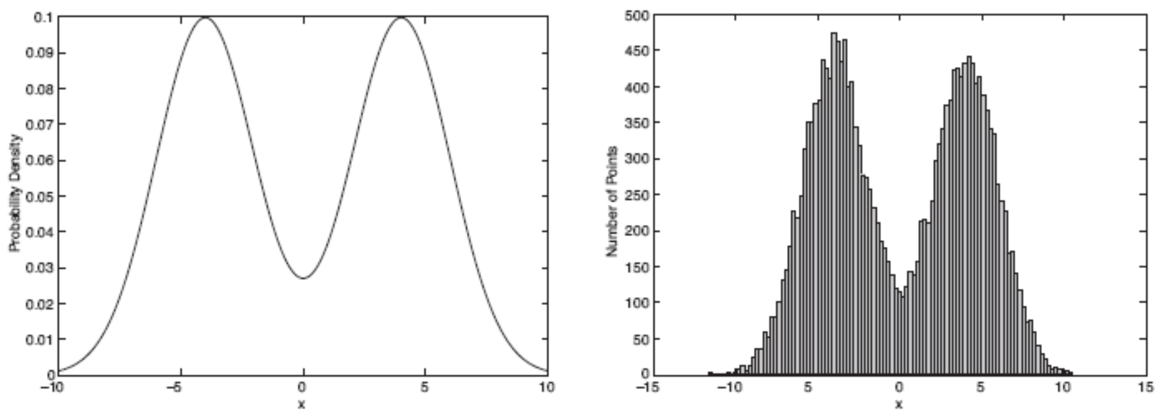


(b) Log likelihood plot of the 200 points for different values of the mean and standard deviation.

Figure 9.3. 200 points from a Gaussian distribution and their log probability for different parameter values.

Mixture of Gaussians

- Suppose that you have the heights of people from Greece and China and the distribution looks like the figure below (dramatization)



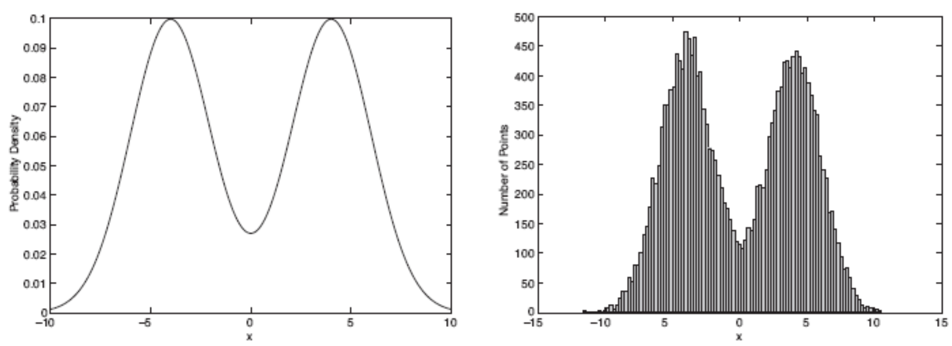
(a) Probability density function for the mixture model.

(b) 20,000 points generated from the mixture model.

Figure 9.2. Mixture model consisting of two normal distributions with means of -4 and 4, respectively. Both distributions have a standard deviation of 2.

Mixture of Gaussians

- In this case the data is the result of the mixture of two Gaussians
  - One for Greek people, and one for Chinese people
  - Identifying for each value which Gaussian is most likely to have generated it will give us a clustering.



(a) Probability density function for the mixture model.

(b) 20,000 points generated from the mixture model.

Figure 9.2. Mixture model consisting of two normal distributions with means of -4 and 4, respectively. Both distributions have a standard deviation of 2.

Mixture Model

- A value x_i is generated according to the following process:
  - First select the nationality
    - With probability π_G select Greek, with probability π_C select China ($\pi_G + \pi_C = 1$)
  - Given the nationality, generate the point from the corresponding Gaussian
    - $P(x_i|\theta_G) \sim N(\mu_G, \sigma_G)$ if Greece
    - $P(x_i|\theta_C) \sim N(\mu_C, \sigma_C)$ if China

Mixture Models

- Our model has the following parameters

$$\Theta = (\pi_G, \pi_C, \mu_G, \mu_C, \sigma_G, \sigma_C)$$

Mixture probabilities | Distribution Parameters

- For value x_i, we have:

$$P(x_i|\Theta) = \pi_G P(x_i|\theta_G) + \pi_C P(x_i|\theta_C)$$

- For all values $X = (x_1, \dots, x_n)$

$$P(X|\Theta) = \prod_{i=1}^{n} P(x_i|\Theta)$$

- We want to estimate the parameters that maximize the Likelihood of the data

Mixture Models

- Once we have the parameters $\Theta = (\pi_G, \pi_C, \mu_G, \mu_C, \sigma_G, \sigma_C)$ we can estimate the membership probabilities $P(G|x_i)$ and $P(C|x_i)$ for each point x_i:
  - This is the probability that point x_i belongs to the Greek or the Chinese population (cluster)

$$P(G|x_i) = \frac{P(x_i|G)P(G)}{P(x_i|G)P(G) + P(x_i|C)P(C)}$$

$$= \frac{P(x_i|G)\pi_G}{P(x_i|G)\pi_G + P(x_i|C)\pi_C}$$

EM (Expectation Maximization) Algorithm

- Initialize the values of the parameters in Θ to some random values
- Repeat until convergence
  - E-Step: Given the parameters Θ estimate the membership probabilities $P(G|x_i)$ and $P(C|x_i)$
  - M-Step: Compute the parameter values that (in expectation) maximize the data likelihood

$$\pi_G = \frac{1}{n}\sum_{i=1}^{n} P(G|x_i) \qquad \pi_C = \frac{1}{n}\sum_{i=1}^{n} P(C|x_i)$$

Fraction of population in G,C

$$\mu_C = \sum_{i=1}^{n} \frac{P(C|x_i)}{n * \pi_C} x_i \qquad \mu_G = \sum_{i=1}^{n} \frac{P(G|x_i)}{n * \pi_G} x_i$$

MLE Estimates if π's were fixed

$$\sigma_C^2 = \sum_{i=1}^{n} \frac{P(C|x_i)}{n * \pi_C} (x_i - \mu_C)^2 \qquad \sigma_G^2 = \sum_{i=1}^{n} \frac{P(G|x_i)}{n * \pi_G} (x_i - \mu_G)^2$$

Bisecting K-means

Finding the best number of clusters

- In k-means the number of clusters *K* is given
  - Partition *n* objects into predetermined number of clusters
  - **Finding the "right" number of clusters is part of the problem**

Bisecting K-means

Variant of K-means that can produce
a hierarchical clustering

Initialize the list of clusters to contain the cluster containing all points.
repeat
  Select a cluster from the list of clusters
  for $i = 1$ to *number_of_iterations* **do**
    Bisect the selected cluster using basic K-means
  end for
  Add the two clusters from the bisection with the lowest SSE to the list of clusters.
until Until the list of clusters contains K clusters

Bisecting K-Means

- The algorithm is exhaustive terminating at singleton clusters (unless K is known)
- Terminating at singleton clusters
  - Is time consuming
  - Singleton clusters are meaningless
  - Intermediate clusters are more likely to correspond to real classes
- No criterion for stopping bisections before singleton clusters are reached.

Bayesian Information Criterion (BIC)

- A strategy to stop the Bisecting algorithm when meaningful clusters are reached to avoid **over-splitting**

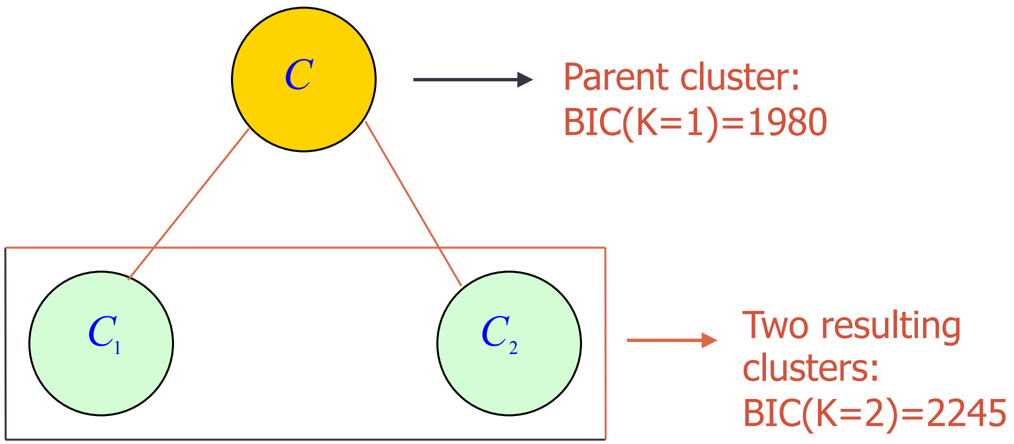
- Using BIC as splitting criterion of a cluster in order to decide whether a cluster should split or no

- BIC measures the improvement of the cluster structure between a cluster and its two children clusters.

- Compute the BIC score of:
  - A cluster
  - Two children clusters

- BIC approximates the probability that the M_j is describing the real clusters in the data

BIC based split



The BIC score of the parent cluster is less than BIC score of the generated cluster structure => we accept the bisection.

X-Means

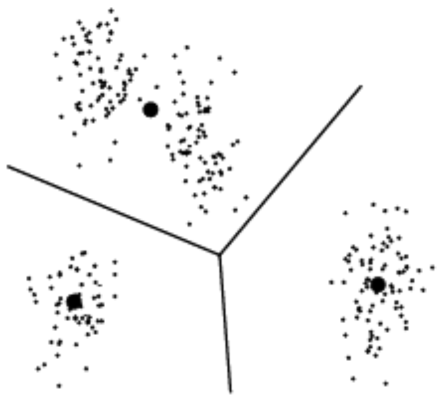
- Forward search for the appropriate value of k in a given range $[r_1, r_{max}]$:
  - Recursively split each cluster and use BIC score to decide if we should keep each split

1. Run K-means with $k=r_1$
2. Improve structure
3. If $k > r_{max}$ Stop and return the best-scoring model

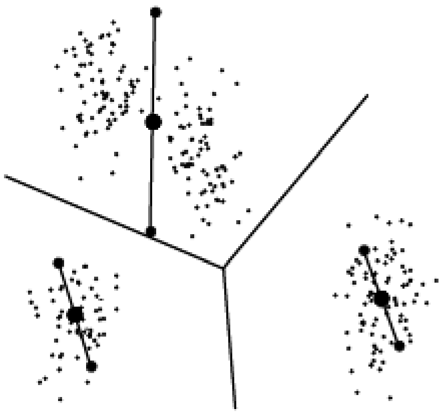
- Use local BIC score to decide on keeping a split
- Use global BIC score to decide which K to output at the end

X-Means

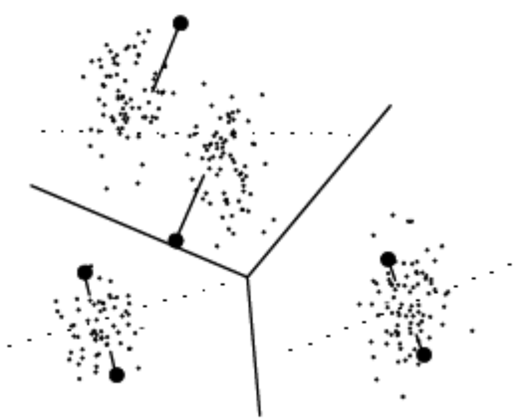
1. K-means with k=3



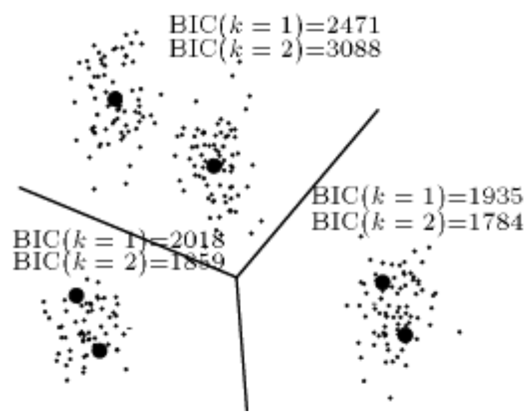
2. Split each centroid in 2 children moved a distance propotional to the region size in opposite direction (random)



3. Run 2-means in each region locally



4. Compare BIC of parent and children



4. Only centroids with higher BIC survives

BIC Formula

- The BIC score of a data collection is defined as (Kass and Wasserman, 1995):

$$BIC(M_j) = \hat{l}_j(D) - \frac{p_j}{2} \log R$$

- $\hat{l}_j(D)$ is the log-likelihood of the data set D
- P_j is a function of the number of independent parameters: centroids coordinates, variance estimation.
- R is the number of points of a cluster

Approximate the probability that the M_j is describing the real clusters in the data

BIC (Bayesian Information Criterion)

- Adjusted Log-likelihood of the model.
- The likelihood that the **data is "explained by" the clusters** according to the spherical-Gaussian assumption of k-means

$$BIC(M_j) = \hat{l}_j(D) - \frac{p_j}{2} \log R$$

Focusing on the set D_n of points which belong to centroid n

$$\begin{aligned} \hat{l}(D_n) \quad &= \quad -\frac{R_n}{2}\log(2\pi) - \frac{R_n \cdot M}{2}\log(\hat{\sigma}^2) - \frac{R_n - K}{2} \\ &\quad +R_n \log R_n - R_n \log R \end{aligned}$$

It estimates how closely to the centroid are the points of the cluster.