UNIVERSITÄT BIELEFELD
Faculty of Technology

Introduction to Data Mining
05 - Clustering

Benjamin Paaßen

WS 2023/2024, Bielefeld University

Exercise Submission

- Sheet 02 is out today

Exercise Submission

UNIVERSITÄT BIELEFELD
Faculty of Technology

- Sheet 02 is out today
- We noticed some difficulties in assigning e-mail submissions to groups

Exercise Submission

UNIVERSITÄT BIELEFELD
Faculty of Technology

- Sheet 02 is out today
- We noticed some difficulties in assigning e-mail submissions to groups

$\Rightarrow$ Please use moodle exclusively for your submissions, from now on

1. K-Means algorithm

2. Gaussian Mixture Models

Motivation: Education Example

- Say we want to personalize a course, but have limited teaching capacity

Motivation: Education Example

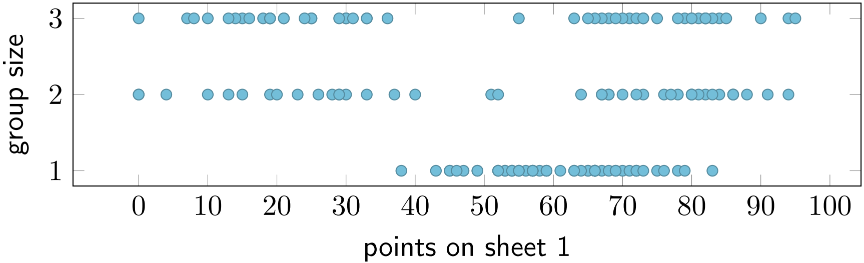
- Say we want to personalize a course, but have limited teaching capacity

$\Rightarrow$ Find **clusters** of students who have similar support needs.

Motivation: Education Example

- Say we want to personalize a course, but have limited teaching capacity

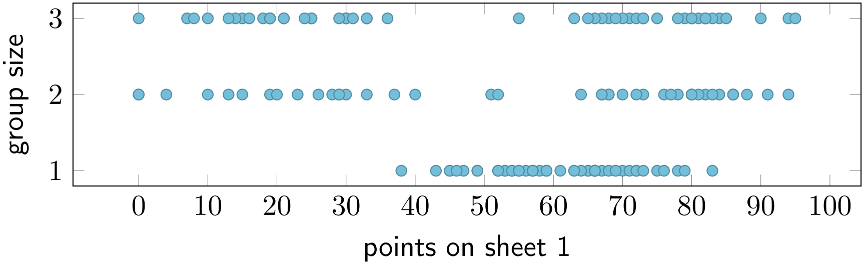
⇒ Find **clusters** of students who have similar support needs.



Motivation: Education Example

- Say we want to personalize a course, but have limited teaching capacity

⇒ Find **clusters** of students who have similar support needs.

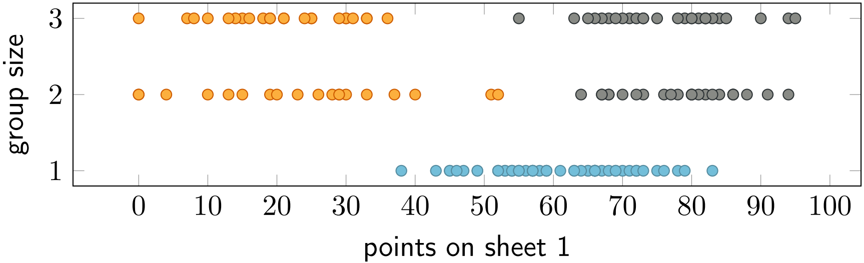


- Which patterns do you notice?

Motivation: Education Example

- Say we want to personalize a course, but have limited teaching capacity

⇒ Find **clusters** of students who have similar support needs.



- Which patterns do you notice?

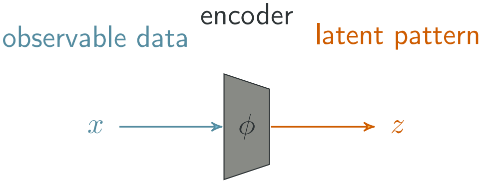
K-Means algorithm

Autoencoding Setup

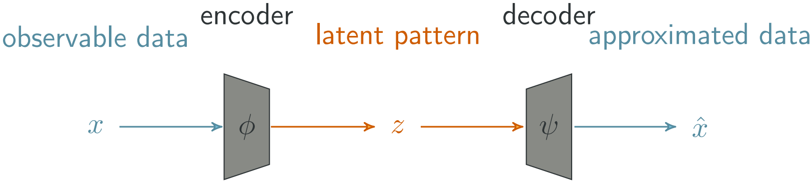
observable data

x

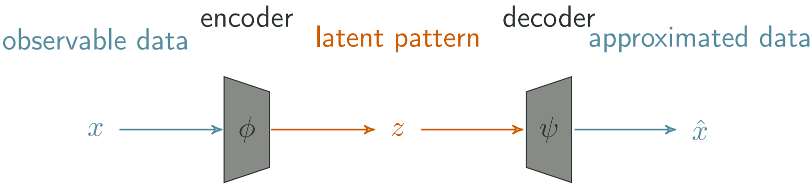
Autoencoding Setup



Autoencoding Setup

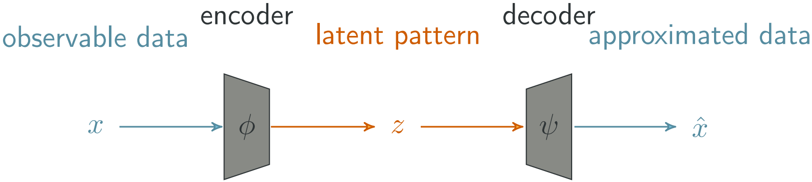


Autoencoding Setup



- Clustering means that $z \in \{1, \ldots, K\}$

Autoencoding Setup



- Clustering means that $z \in \{1, \ldots, K\}$

$\Rightarrow$ decoder: $\psi(z) = \mu_z \in \mathbb{R}^m$ (discrete catalogue of "prototypes")

Optimal encoder

Universität Bielefeld

Faculty of Technology

- For each x, directly optimize reconstruction error

$$\phi(x) = \arg\min_k \|\psi(k) - x\|$$

Optimal encoder

UNIVERSITÄT BIELEFELD
Faculty of Technology

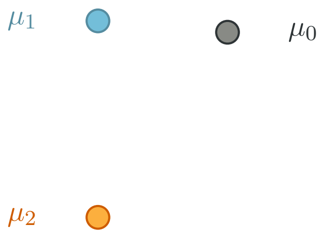
- For each x, directly optimize reconstruction error

$$\phi(x) = \arg\min_k \|\psi(k) - x\| = \arg\min_k \|\mu_k - x\|$$

Optimal encoder

- For each x, directly optimize reconstruction error

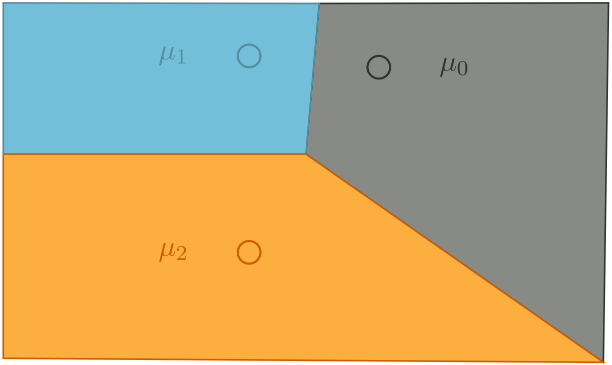
$$\phi(x) = \arg\min_k \|\psi(k) - x\| = \arg\min_k \|\mu_k - x\|$$



Optimal encoder

- For each x, directly optimize reconstruction error

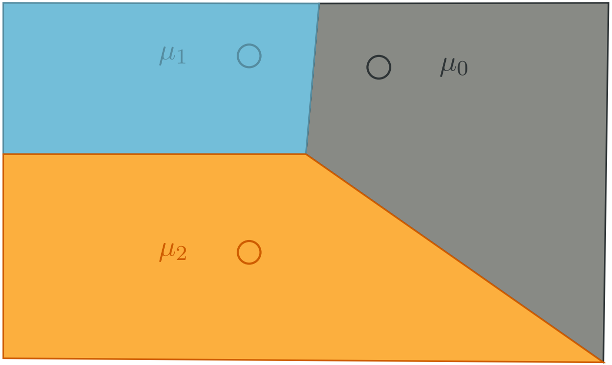
$$\phi(x) = \arg\min_k \|\psi(k) - x\| = \arg\min_k \|\mu_k - x\|$$



Optimal encoder

- For each x, directly optimize reconstruction error

$$\phi(x) = \arg\min_k \|\psi(k) - x\| = \arg\min_k \|\mu_k - x\|$$



Voronoi tesselation

Optimize prototype positions

- Let $x_1, \ldots, x_N \in \mathbb{R}^m$ be example data

Optimize prototype positions

- Let $x_1, \ldots, x_N \in \mathbb{R}^m$ be example data
- Try to find prototypes $\mu_1, \ldots, \mu_K \in \mathbb{R}^m$ that minimize reconstruction error

$$\min_{\mu_1,\ldots,\mu_K} \quad \sum_{i=1}^{N} \|\mu_{z_i} - x_i\|^2$$

$$\text{such that} \quad z_i = \phi(x_i) = \arg\min_k \|\mu_k - x\|$$

Optimize prototype positions

- Let $x_1, \ldots, x_N \in \mathbb{R}^m$ be example data
- Try to find prototypes $\mu_1, \ldots, \mu_K \in \mathbb{R}^m$ that minimize reconstruction error

$$\min_{\mu_1, \ldots, \mu_K} \quad \sum_{i=1}^{N} \|\mu_{z_i} - x_i\|^2$$

$$\text{such that} \quad z_i = \phi(x_i) = \arg\min_k \|\mu_k - x\|$$

- Unfortunately, this is NP-hard :(

Optimize prototype positions (continued)

- Idea: Optimization becomes really simple, if z_i is fixed :)

Optimize prototype positions (continued)

UNIVERSITÄT BIELEFELD
Faculty of Technology

- Idea: Optimization becomes really simple, if z_i is fixed :)
- Let $\mathcal{P}_k : \{i | z_i = k\}$; the "receptive field" of prototype k

Optimize prototype positions (continued)

- Idea: Optimization becomes really simple, if z_i is fixed :)
- Let $\mathcal{P}_k : \{i|z_i = k\}$; the "receptive field" of prototype k
- Calculate gradient:

$$\nabla_{\mu_k} \sum_{i=1}^{N} \|\mu_{z_i} - x_i\|^2$$

Optimize prototype positions (continued)

UNIVERSITÄT BIELEFELD
Faculty of Technology

- Idea: Optimization becomes really simple, if z_i is fixed :)
- Let $\mathcal{P}_k : \{i|z_i = k\}$; the "receptive field" of prototype k
- Calculate gradient:

$$\nabla_{\mu_k} \sum_{i=1}^{N} \|\mu_{z_i} - x_i\|^2 = \nabla_{\mu_k} \sum_{i \in \mathcal{P}_k} \|\mu_k - x_i\|^2$$

Optimize prototype positions (continued)

UNIVERSITÄT BIELEFELD
Faculty of Technology

- Idea: Optimization becomes really simple, if z_i is fixed :)
- Let $\mathcal{P}_k : \{i | z_i = k\}$; the "receptive field" of prototype k
- Calculate gradient:

$$\nabla_{\mu_k} \sum_{i=1}^{N} \|\mu_{z_i} - x_i\|^2 = \nabla_{\mu_k} \sum_{i \in \mathcal{P}_k} \|\mu_k - x_i\|^2$$
$$= \sum_{i \in \mathcal{P}_k} 2(\mu_k - x_i)$$

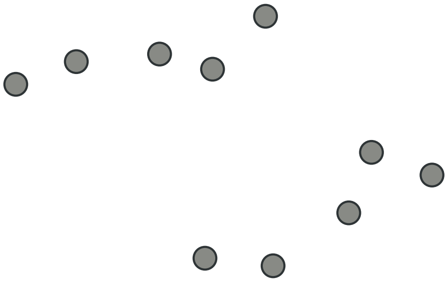
Optimize prototype positions (continued)

- Idea: Optimization becomes really simple, if z_i is fixed :)
- Let $\mathcal{P}_k : \{i | z_i = k\}$; the "receptive field" of prototype k
- Calculate gradient:

$$\nabla_{\mu_k} \sum_{i=1}^{N} \|\mu_{z_i} - x_i\|^2 = \nabla_{\mu_k} \sum_{i \in \mathcal{P}_k} \|\mu_k - x_i\|^2$$
$$= \sum_{i \in \mathcal{P}_k} 2(\mu_k - x_i)$$

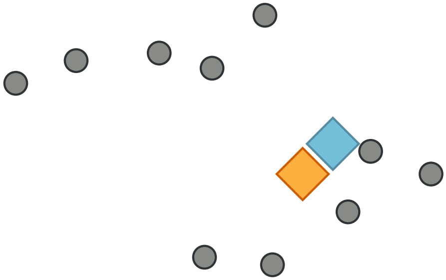
- Setting gradient to zero yields $\mu_k = \frac{1}{|\mathcal{P}_k|} \sum_{i \in \mathcal{P}_k} x_i$.

Visualization

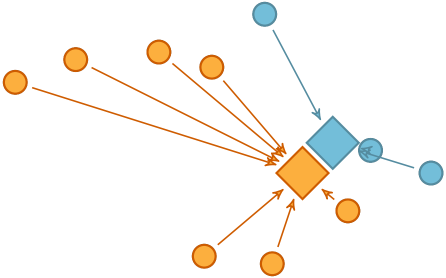


Visualization

UNIVERSITÄT BIELEFELD
Faculty of Technology

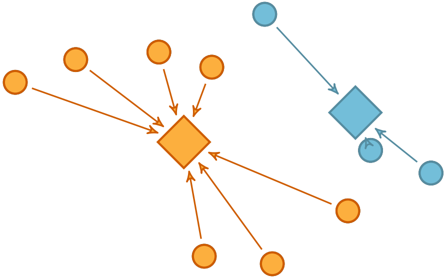


Visualization



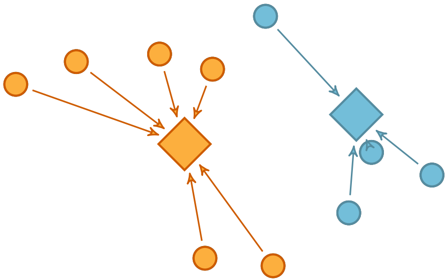
assignment step

Visualization



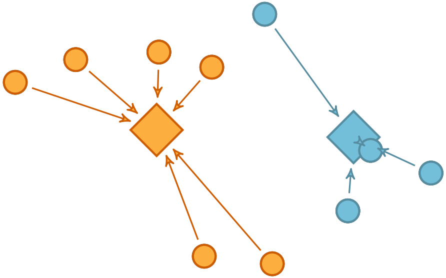
mean step

Visualization



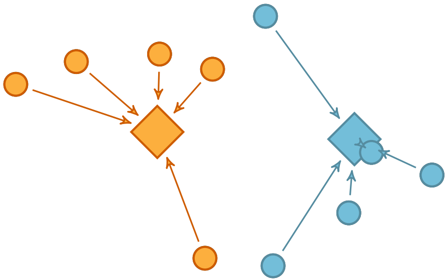
assignment step

Visualization



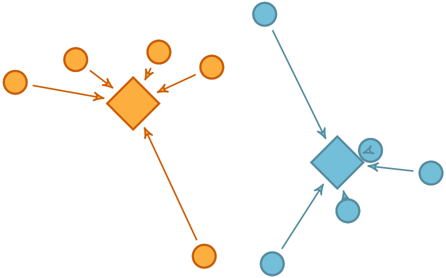
mean step

Visualization



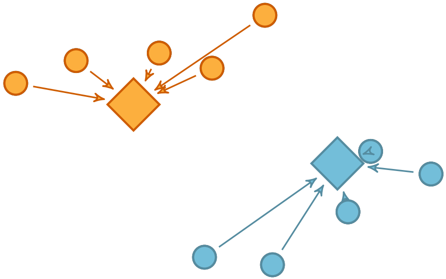
assignment step

Visualization



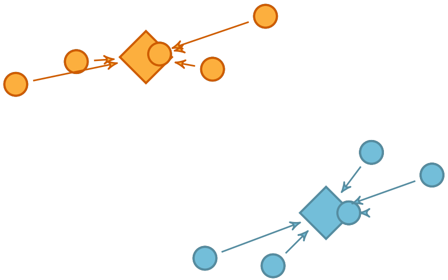
mean step

Visualization



assignment step

Visualization



mean step

K-means procedure

function KMeans(data matrix $\boldsymbol{X}$ with N rows and m columns, desired number of clusters K)

end function

K-means procedure

function KMeans(data matrix $\boldsymbol{X}$ with N rows and m columns, desired number of clusters K)

Randomly initialize $\mu_1, \ldots, \mu_K \in \mathbb{R}^m$ (in the convex hull of the data).

end function

K-means procedure

UNIVERSITÄT BIELEFELD
Faculty of Technology

function KMeans(data matrix $\boldsymbol{X}$ with N rows and m columns, desired number of clusters K)

Randomly initialize $\mu_1, \ldots, \mu_K \in \mathbb{R}^m$ (in the convex hull of the data).

while at least one z_i changed in the last iteration **do**

end while

end function

K-means procedure

function KMeans(data matrix $\boldsymbol{X}$ with N rows and m columns, desired number of clusters K)

Randomly initialize $\mu_1, \ldots, \mu_K \in \mathbb{R}^m$ (in the convex hull of the data).

while at least one z_i changed in the last iteration **do**

Compute $z_i \leftarrow \arg\min_k \|\mu_k - x_i\|$ for all $i \in \{1, \ldots, N\}$.

end while

end function

K-means procedure

UNIVERSITÄT BIELEFELD

Faculty of Technology

function KMeans(data matrix $\boldsymbol{X}$ with N rows and m columns, desired number of clusters K)

Randomly initialize $\mu_1, \ldots, \mu_K \in \mathbb{R}^m$ (in the convex hull of the data).

while at least one z_i changed in the last iteration **do**

Compute $z_i \leftarrow \arg\min_k \|\mu_k - x_i\|$ for all $i \in \{1, \ldots, N\}$.

Compute $\mathcal{P}_k \leftarrow \{i | z_i = k\}$ for all $k \in \{1, \ldots, K\}$.

end while

end function

K-means procedure

UNIVERSITÄT BIELEFELD
Faculty of Technology

function KMeans(data matrix $\boldsymbol{X}$ with N rows and m columns, desired number of clusters K)

Randomly initialize $\mu_1, \ldots, \mu_K \in \mathbb{R}^m$ (in the convex hull of the data).

while at least one z_i changed in the last iteration **do**

Compute $z_i \leftarrow \arg\min_k \|\mu_k - x_i\|$ for all $i \in \{1, \ldots, N\}$.

Compute $\mathcal{P}_k \leftarrow \{i | z_i = k\}$ for all $k \in \{1, \ldots, K\}$.

Compute $\mu_k \leftarrow \frac{1}{|\mathcal{P}_k|} \sum_{i \in \mathcal{P}_k} x_i$ for all $k \in \{1, \ldots, K\}$.

end while

end function

K-means procedure

function KMeans(data matrix $\boldsymbol{X}$ with N rows and m columns, desired number of clusters K)
  Randomly initialize $\mu_1, \ldots, \mu_K \in \mathbb{R}^m$ (in the convex hull of the data).
  while at least one z_i changed in the last iteration **do**
    Compute $z_i \leftarrow \arg\min_k \|\mu_k - x_i\|$ for all $i \in \{1, \ldots, N\}$.
    Compute $\mathcal{P}_k \leftarrow \{i | z_i = k\}$ for all $k \in \{1, \ldots, K\}$.
    Compute $\mu_k \leftarrow \frac{1}{|\mathcal{P}_k|} \sum_{i \in \mathcal{P}_k} x_i$ for all $k \in \{1, \ldots, K\}$.
  end while
  return $\mu_1, \ldots, \mu_K$.
end function

Proof of stopping (Sketch)

UNIVERSITÄT BIELEFELD
Faculty of Technology

- Let $\boldsymbol{z}_1, \ldots \boldsymbol{z}_t$ be the series of assignment vectors for subsequent iterations of K-means

Proof of stopping (Sketch)

UNIVERSITÄT BIELEFELD
Faculty of Technology

- Let $\boldsymbol{z}_1, \ldots \boldsymbol{z}_t$ be the series of assignment vectors for subsequent iterations of K-means
- There are only finitely many possible assignments – so for sufficiently high t, one must repeat

Proof of stopping (Sketch)

- Let $\boldsymbol{z}_1, \ldots \boldsymbol{z}_t$ be the series of assignment vectors for subsequent iterations of K-means
- There are only finitely many possible assignments – so for sufficiently high t, one must repeat
- An immediate repeat stops the algorithm

Proof of stopping (Sketch)

UNIVERSITÄT BIELEFELD
Faculty of Technology

- Let $\boldsymbol{z}_1, \ldots \boldsymbol{z}_t$ be the series of assignment vectors for subsequent iterations of K-means
- There are only finitely many possible assignments – so for sufficiently high t, one must repeat
- An immediate repeat stops the algorithm
- Otherwise, note that each assignment vector $\boldsymbol{z}$ corresponds to **one** reconstruction error (after re-computing the means)

Proof of stopping (Sketch)

- Let $\boldsymbol{z}_1, \ldots \boldsymbol{z}_t$ be the series of assignment vectors for subsequent iterations of K-means
- There are only finitely many possible assignments – so for sufficiently high t, one must repeat
- An immediate repeat stops the algorithm
- Otherwise, note that each assignment vector $\boldsymbol{z}$ corresponds to **one** reconstruction error (after re-computing the means)
- K-means only changes the assignments to **reduce** the reconstruction error

Proof of stopping (Sketch)

UNIVERSITÄT BIELEFELD
Faculty of Technology

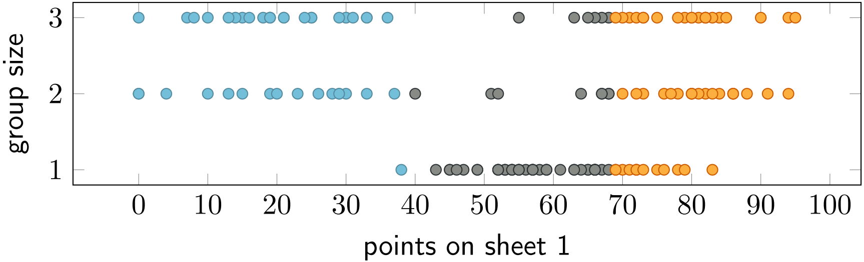
- Let $\boldsymbol{z}_1, \ldots \boldsymbol{z}_t$ be the series of assignment vectors for subsequent iterations of K-means
- There are only finitely many possible assignments – so for sufficiently high t, one must repeat
- An immediate repeat stops the algorithm
- Otherwise, note that each assignment vector $\boldsymbol{z}$ corresponds to **one** reconstruction error (after re-computing the means)
- K-means only changes the assignments to **reduce** the reconstruction error
- Hence, repeating an assignment vector is impossible (this would increase the loss)

Proof of stopping (Sketch)

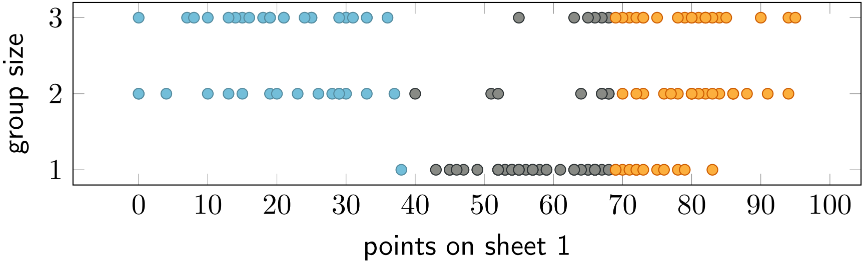
- Let $z_1, \ldots z_t$ be the series of assignment vectors for subsequent iterations of K-means
- There are only finitely many possible assignments – so for sufficiently high t, one must repeat
- An immediate repeat stops the algorithm
- Otherwise, note that each assignment vector z corresponds to **one** reconstruction error (after re-computing the means)
- K-means only changes the assignments to **reduce** the reconstruction error
- Hence, repeating an assignment vector is impossible (this would increase the loss)
- **note!** in practice, K-means mostly stops already after ca. 10-30 iterations

Example: non-normalized data

UNIVERSITÄT BIELEFELD
Faculty of Technology



Example: non-normalized data



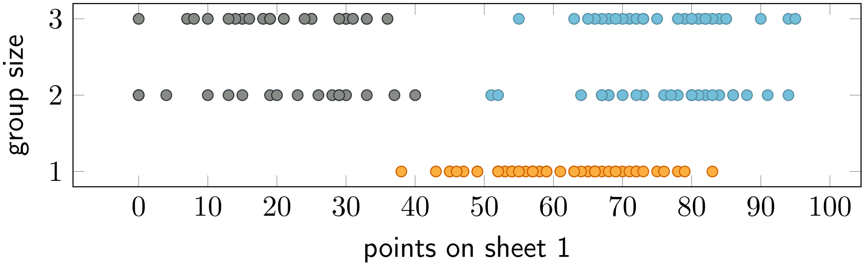
- What is the problem here?

Example: normalized data

- Trick: z-normalize data before using K-means to avoid scaling issues: $x_{i,j} \leftarrow \frac{x_{i,j} - \mu_j}{\sigma_j}$

Example: normalized data

- Trick: z-normalize data before using K-means to avoid scaling issues: $x_{i,j} \leftarrow \frac{x_{i,j} - \mu_j}{\sigma_j}$



How to choose K?

- Idea: Compute "quality indices" for clusterings and take K that yields the best quality index

How to choose K?

- Idea: Compute "quality indices" for clusterings and take K that yields the best quality index
- Silhouette Score: Let d_i^+ be average distance to other samples in the same cluster as i and let d_i^- be the distance of i to the closest, other cluster

How to choose K?

- Idea: Compute "quality indices" for clusterings and take K that yields the best quality index
- Silhouette Score: Let d_i^+ be average distance to other samples in the same cluster as i and let d_i^- be the distance of i to the closest, other cluster $\Rightarrow$ $\frac{1}{N}\sum_{i=1}^{N} \frac{d_i^- - d_i^+}{\max\{d_i^-, d_i^+\}}$

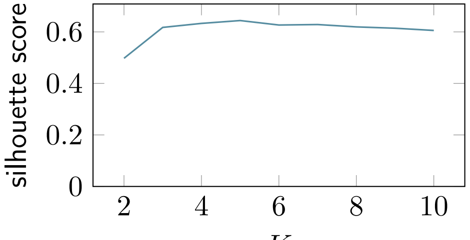
How to choose K?

- Idea: Compute "quality indices" for clusterings and take K that yields the best quality index
- Silhouette Score: Let d_i^+ be average distance to other samples in the same cluster as i and let d_i^- be the distance of i to the closest, other cluster $\Rightarrow$ $\frac{1}{N} \sum_{i=1}^N \frac{d_i^- - d_i^+}{\max\{d_i^-, d_i^+\}}$
- Bayesian information criterion (BIC): reconstruction loss plus number of model parameters: $K \cdot m$

How to choose K?

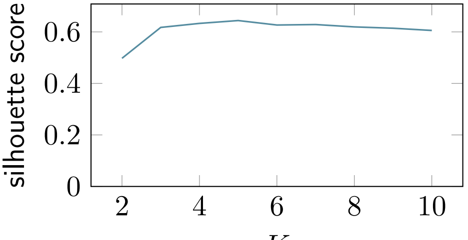
UNIVERSITÄT BIELEFELD
Faculty of Technology

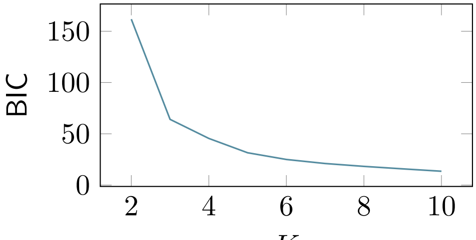
- Idea: Compute "quality indices" for clusterings and take K that yields the best quality index
- Silhouette Score: Let d_i^+ be average distance to other samples in the same cluster as i and let d_i^- be the distance of i to the closest, other cluster $\Rightarrow$ $\frac{1}{N}\sum_{i=1}^N \frac{d_i^- - d_i^+}{\max\{d_i^-, d_i^+\}}$
- Bayesian information criterion (BIC): reconstruction loss plus number of model parameters: $K \cdot m$



How to choose K?

- Idea: Compute "quality indices" for clusterings and take K that yields the best quality index
- Silhouette Score: Let d_i^+ be average distance to other samples in the same cluster as i and let d_i^- be the distance of i to the closest, other cluster $\Rightarrow$ $\frac{1}{N}\sum_{i=1}^{N} \frac{d_i^- - d_i^+}{\max\{d_i^-, d_i^+\}}$
- Bayesian information criterion (BIC): reconstruction loss plus number of model parameters: $K \cdot m$





Gaussian Mixture Models

Probabilistic Model

- Assume data is generated by a mixture of K Gaussians as follows:

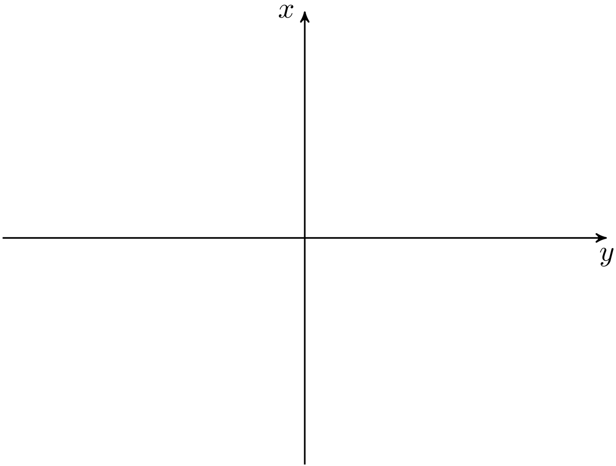
Probabilistic Model

- Assume data is generated by a mixture of K Gaussians as follows:
- Step 1: sample latent cluster index k via marginal probability $p_Z(k)$

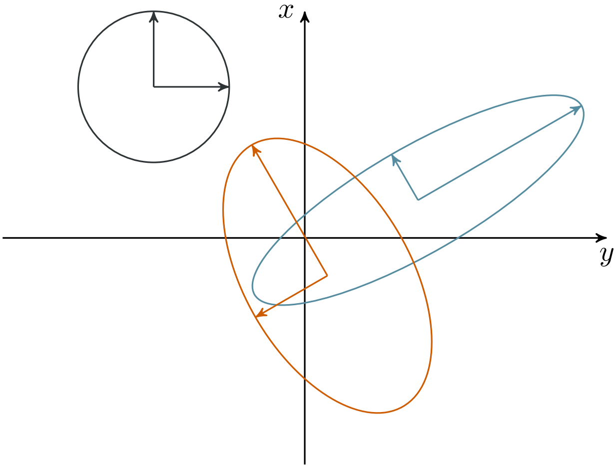
Probabilistic Model

- Assume data is generated by a mixture of K Gaussians as follows:
- Step 1: sample latent cluster index k via marginal probability $p_Z(k)$
- Step 2: Sample observable data point x from Gaussian $p_{X|Z}(x|k)$ with mean μ_k and covariance matrix Σ_k

Visualization



Visualization



Marginal and Bayes

$$p_X(x) = \sum_{k=1}^{K} p_{X|Z}(x|k) \cdot p_Z(k)$$

Marginal and Bayes

$$p_X(x) = \sum_{k=1}^{K} p_{X|Z}(x|k) \cdot p_Z(k)$$

$$p_{Z|X}(k|x) = \frac{p_{X|Z}(x|k) \cdot p_Z(k)}{\sum_{z=1}^{K} p_{X|Z}(x|z) \cdot p_Z(z)}$$

Optimization procedure

- Negative log likelihood:

$$\ell = \sum_{i=1}^{N} -\log\left[\sum_{k=1}^{K} p_{X|Z}(x_i|k) \cdot p_Z(k)\right]$$

Optimization procedure

- Negative log likelihood:

$$\ell = \sum_{i=1}^{N} - \log \Big[\sum_{k=1}^{K} p_{X|Z}(x_i|k) \cdot p_Z(k) \Big]$$

$\Rightarrow$ Nasty to optimize (log of sum, non-convex) :(

Optimization procedure

- Negative log likelihood:

$$\ell = \sum_{i=1}^{N} -\log\left[\sum_{k=1}^{K} p_{X|Z}(x_i|k) \cdot p_Z(k)\right]$$

$\Rightarrow$ Nasty to optimize (log of sum, non-convex) :(

- Trick: Compute $\gamma_{k,i} = p_{Z|X}(k|x_i)$, then optimize the **expected** negative log likelihood instead:

$$Q = \sum_{i=1}^{N} \sum_{k=1}^{K} -\gamma_{k,i} \cdot \log\left[p_{X|Z}(x_i|k) \cdot p_Z(k)\right]$$

Optimization procedure

UNIVERSITÄT BIELEFELD
Faculty of Technology

- Negative log likelihood:

$$\ell = \sum_{i=1}^{N} -\log\left[\sum_{k=1}^{K} p_{X|Z}(x_i|k) \cdot p_Z(k)\right]$$

$\Rightarrow$ Nasty to optimize (log of sum, non-convex) :(

- Trick: Compute $\gamma_{k,i} = p_{Z|X}(k|x_i)$, then optimize the **expected** negative log likelihood instead:

$$Q = \sum_{i=1}^{N}\sum_{k=1}^{K} -\gamma_{k,i} \cdot \log\left[p_{X|Z}(x_i|k) \cdot p_Z(k)\right]$$
$$= \sum_{i=1}^{N}\sum_{k=1}^{K} \gamma_{k,i} \cdot \left(\frac{1}{2}\log[2\pi \det(\Sigma_k)] + \frac{1}{2}(x_i - \mu_k)^T \Sigma_k^{-1}(x_i - \mu_k) - \log[p_Z(k)]\right)$$

Optimization procedure

- Negative log likelihood:

$$\ell = \sum_{i=1}^{N} - \log \Big[\sum_{k=1}^{K} p_{X|Z}(x_i|k) \cdot p_Z(k) \Big]$$

⇒ Nasty to optimize (log of sum, non-convex) :(

- Trick: Compute $\gamma_{k,i} = p_{Z|X}(k|x_i)$, then optimize the **expected** negative log likelihood instead:

$$Q = \sum_{i=1}^{N} \sum_{k=1}^{K} -\gamma_{k,i} \cdot \log \Big[p_{X|Z}(x_i|k) \cdot p_Z(k) \Big]$$
$$= \sum_{i=1}^{N} \sum_{k=1}^{K} \gamma_{k,i} \cdot \Big(\frac{1}{2} \log[2\pi \det(\Sigma_k)] + \frac{1}{2}(x_i - \mu_k)^T \Sigma_k^{-1} (x_i - \mu_k) - \log[p_Z(k)] \Big)$$

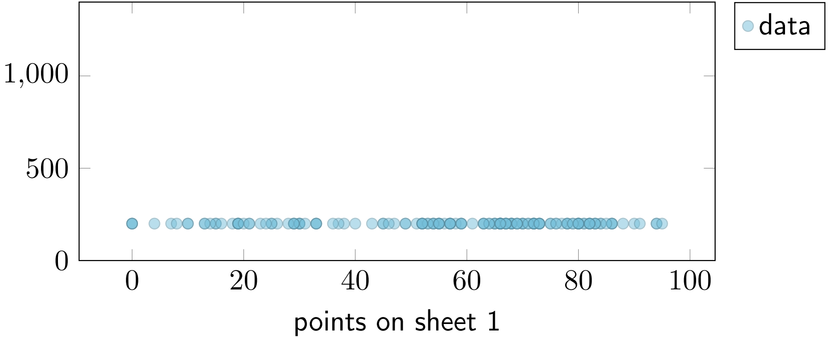
⇒ Convex and closed-form solution (see exercises) :)

Visualization

- Scenario: only consider one dimension, $K = 2$, $\sigma_1 = \sigma_2 = 10$, $\mu_1 = 60$, only μ_2 varies

Visualization

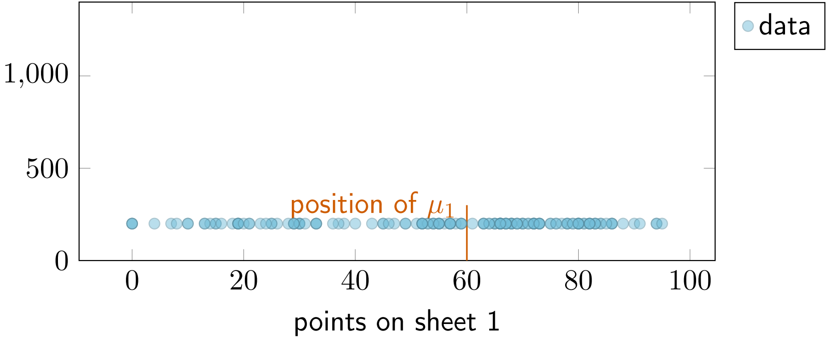
- Scenario: only consider one dimension, $K = 2$, $\sigma_1 = \sigma_2 = 10$, $\mu_1 = 60$, only μ_2 varies



Visualization

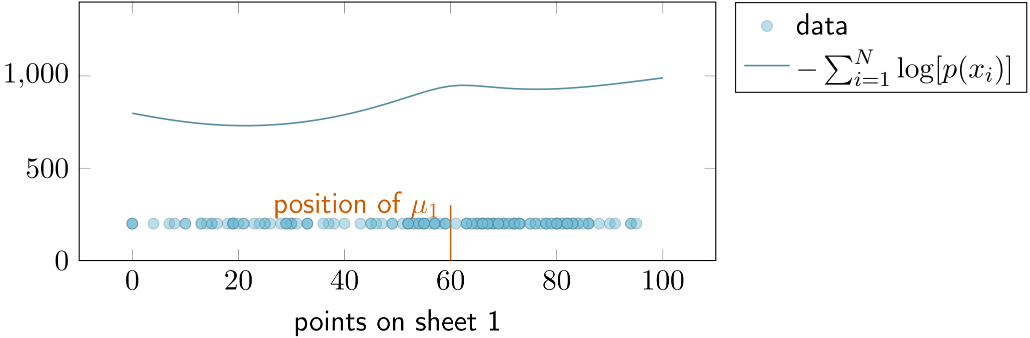
UNIVERSITÄT BIELEFELD
Faculty of Technology

- Scenario: only consider one dimension, $K = 2$, $\sigma_1 = \sigma_2 = 10$, $\mu_1 = 60$, only μ_2 varies



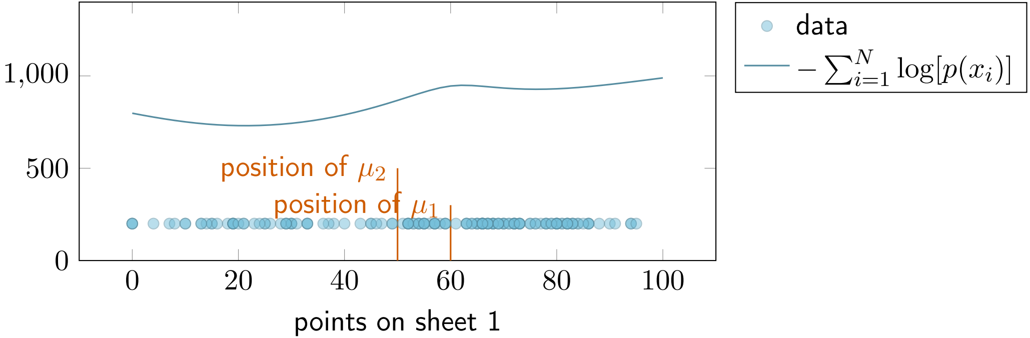
Visualization

- Scenario: only consider one dimension, $K = 2$, $\sigma_1 = \sigma_2 = 10$, $\mu_1 = 60$, only μ_2 varies



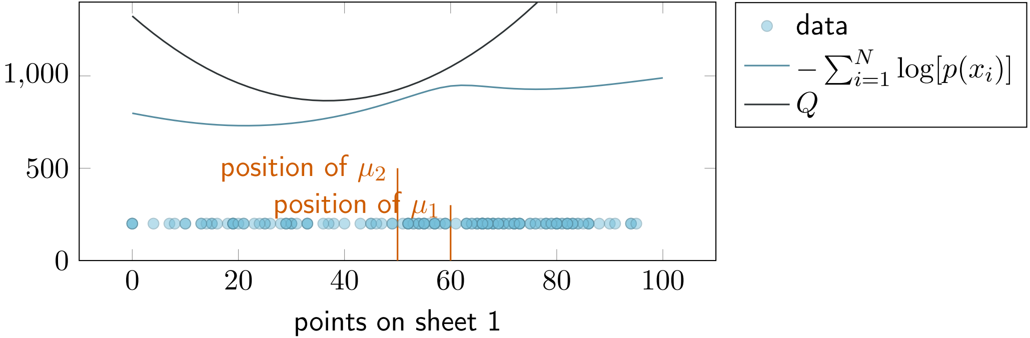
Visualization

- Scenario: only consider one dimension, $K = 2$, $\sigma_1 = \sigma_2 = 10$, $\mu_1 = 60$, only μ_2 varies



Visualization

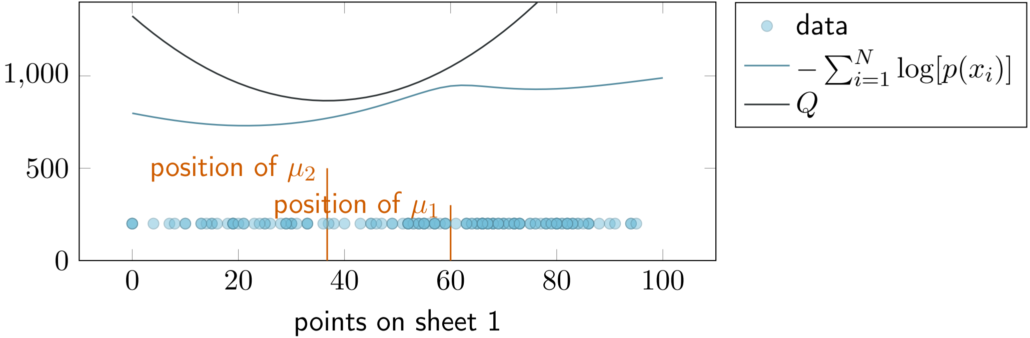
- Scenario: only consider one dimension, $K = 2$, $\sigma_1 = \sigma_2 = 10$, $\mu_1 = 60$, only μ_2 varies



Visualization

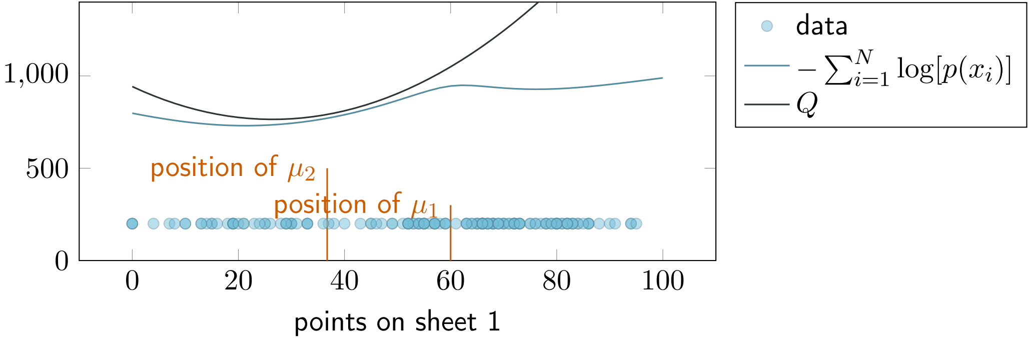
UNIVERSITÄT BIELEFELD
Faculty of Technology

- Scenario: only consider one dimension, $K = 2$, $\sigma_1 = \sigma_2 = 10$, $\mu_1 = 60$, only μ_2 varies



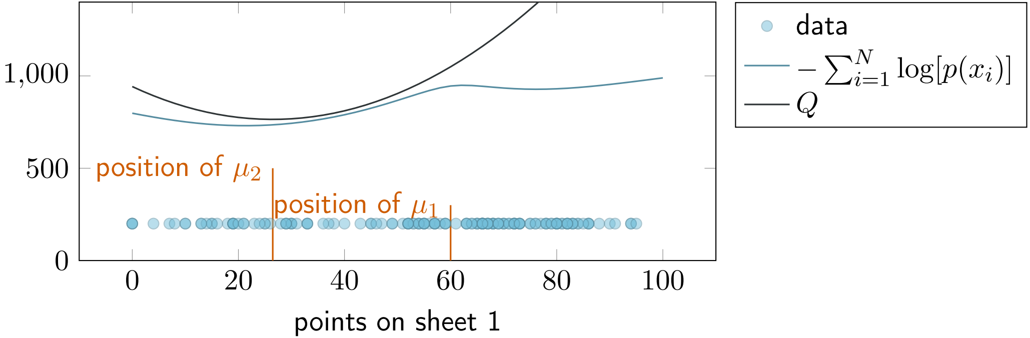
Visualization

- Scenario: only consider one dimension, $K = 2$, $\sigma_1 = \sigma_2 = 10$, $\mu_1 = 60$, only μ_2 varies



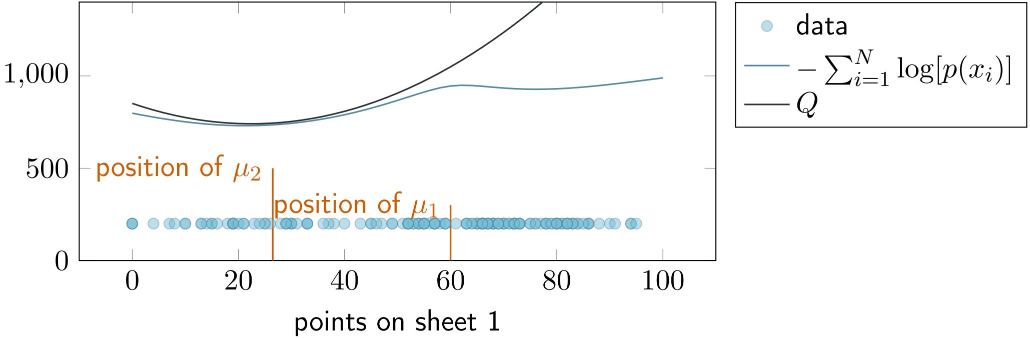
Visualization

- Scenario: only consider one dimension, $K = 2$, $\sigma_1 = \sigma_2 = 10$, $\mu_1 = 60$, only μ_2 varies



Visualization

- Scenario: only consider one dimension, $K = 2$, $\sigma_1 = \sigma_2 = 10$, $\mu_1 = 60$, only μ_2 varies



Theory of EM (part 1)

Let $\gamma_{i,k}$ be arbitrary real number s.t. $\sum_{k=1}^{K} \gamma_{i,k} = 1$ and $\gamma_{i,k} \geq 0$ for all i and all k.
For every x_i, we have:

Theory of EM (part 1)

Let $\gamma_{i,k}$ be arbitrary real number s.t. $\sum_{k=1}^{K} \gamma_{i,k} = 1$ and $\gamma_{i,k} \geq 0$ for all i and all k.
For every x_i, we have:

$$\log[p_X(x_i)] = \sum_{k=1}^{K} \gamma_{i,k} \cdot \log[p_X(x_i)]$$

Theory of EM (part 1)

UNIVERSITÄT BIELEFELD
Faculty of Technology

Let $\gamma_{i,k}$ be arbitrary real number s.t. $\sum_{k=1}^{K} \gamma_{i,k} = 1$ and $\gamma_{i,k} \geq 0$ for all i and all k.
For every x_i, we have:

$$\begin{aligned}
\log[p_X(x_i)] &= \sum_{k=1}^{K} \gamma_{i,k} \cdot \log[p_X(x_i)] \\
&= \sum_{k=1}^{K} \gamma_{i,k} \cdot \log\left[\frac{p_{X,Z}(x_i, k)}{p_{Z|X}(k|x_i)}\right]
\end{aligned}$$

Theory of EM (part 1)

Let $\gamma_{i,k}$ be arbitrary real number s.t. $\sum_{k=1}^{K} \gamma_{i,k} = 1$ and $\gamma_{i,k} \geq 0$ for all i and all k.
For every x_i, we have:

$$
\begin{aligned}
\log[p_X(x_i)] &= \sum_{k=1}^{K} \gamma_{i,k} \cdot \log[p_X(x_i)] \\
&= \sum_{k=1}^{K} \gamma_{i,k} \cdot \log\left[\frac{p_{X,Z}(x_i, k)}{p_{Z|X}(k|x_i)}\right] \\
&= \sum_{k=1}^{K} \gamma_{i,k} \cdot \log[p_{X,Z}(x_i, k)] - \sum_{k=1}^{K} \gamma_{i,k} \cdot \log[p_{Z|X}(k|x_i)]
\end{aligned}
$$

Theory of EM (part 1)

UNIVERSITÄT BIELEFELD
Faculty of Technology

Let $\gamma_{i,k}$ be arbitrary real number s.t. $\sum_{k=1}^{K} \gamma_{i,k} = 1$ and $\gamma_{i,k} \geq 0$ for all i and all k.
For every x_i, we have:

$$
\begin{aligned}
\log[p_X(x_i)] &= \sum_{k=1}^{K} \gamma_{i,k} \cdot \log[p_X(x_i)] \\
&= \sum_{k=1}^{K} \gamma_{i,k} \cdot \log\left[\frac{p_{X,Z}(x_i,k)}{p_{Z|X}(k|x_i)}\right] \\
&= \sum_{k=1}^{K} \gamma_{i,k} \cdot \log[p_{X,Z}(x_i,k)] - \sum_{k=1}^{K} \gamma_{i,k} \cdot \log[p_{Z|X}(k|x_i)] \\
&= \sum_{k=1}^{K} \gamma_{i,k} \cdot \log[p_{X,Z}(x_i,k)] - \sum_{k=1}^{K} \gamma_{i,k} \cdot \Big(\log[p_{Z|X}(k|x_i)] - \log[\gamma_{i,k}] + \log[\gamma_{i,k}]\Big)
\end{aligned}
$$

Theory of EM (part 2)

UNIVERSITÄT BIELEFELD
Faculty of Technology

$$\log[p_X(x_i)] = \sum_{k=1}^{K} \gamma_{i,k} \cdot \log[p_{X,Z}(x_i, k)] - \sum_{k=1}^{K} \gamma_{i,k} \cdot \log\left[\frac{p_{Z|X}(k|x_i)}{\gamma_{i,k}}\right] - \sum_{k=1}^{K} \gamma_{i,k} \cdot \log[\gamma_{i,k}]$$

Theory of EM (part 2)

UNIVERSITÄT BIELEFELD
Faculty of Technology

$$\log[p_X(x_i)] = \sum_{k=1}^{K} \gamma_{i,k} \cdot \log[p_{X,Z}(x_i, k)] - \sum_{k=1}^{K} \gamma_{i,k} \cdot \log\left[\frac{p_{Z|X}(k|x_i)}{\gamma_{i,k}}\right] - \sum_{k=1}^{K} \gamma_{i,k} \cdot \log[\gamma_{i,k}]$$

- We summarize:

$$Q = -\sum_{i=1}^{N} \sum_{k=1}^{K} \gamma_{i,k} \cdot \log[p_{X,Z}(x_i, k)] \qquad \text{(exp. neg. log likelihood)}$$

$$\mathcal{D}_{\mathsf{KL}}(\boldsymbol{\gamma_i} || p_{Z_i|X_i}) = -\sum_{k=1}^{K} \gamma_{i,k} \cdot \log\left[\frac{p_{Z|X}(k|x_i)}{\gamma_{i,k}}\right] \qquad \text{(Kullback-Leibler-Divergence)}$$

$$\mathcal{H}(\boldsymbol{\gamma_i}) = -\sum_{k=1}^{K} \gamma_{i,k} \cdot \log[\gamma_{i,k}] \qquad \text{(entropy)}$$

Theory of EM (part 3)

$$-\sum_{i=1}^{N} \log[p_X(x_i)] = Q - \sum_{i=1}^{N} \mathcal{D}_{\mathsf{KL}}(\boldsymbol{\gamma_i} || p_{Z_i|X_i}) + \mathcal{H}(\boldsymbol{\gamma_i})$$

Theory of EM (part 3)

$$-\sum_{i=1}^{N} \log[p_X(x_i)] = Q - \sum_{i=1}^{N} \mathcal{D}_{\mathsf{KL}}(\boldsymbol{\gamma_i}||p_{Z_i|X_i}) + \mathcal{H}(\boldsymbol{\gamma_i})$$

Interpretation:

Theory of EM (part 3)

$$-\sum_{i=1}^{N} \log[p_X(x_i)] = Q - \sum_{i=1}^{N} \mathcal{D}_{\mathsf{KL}}(\boldsymbol{\gamma_i} || p_{Z_i|X_i}) + \mathcal{H}(\boldsymbol{\gamma_i})$$

Interpretation:

- The true neg. log likelihood $\mathcal{L}$ is equal to Q (our much nicer loss) minus KL divergences minus entropies

Theory of EM (part 3)

$$-\sum_{i=1}^{N} \log[p_X(x_i)] = Q - \sum_{i=1}^{N} \mathcal{D}_{\mathsf{KL}}(\boldsymbol{\gamma_i} || p_{Z_i|X_i}) + \mathcal{H}(\boldsymbol{\gamma_i})$$

Interpretation:

- The true neg. log likelihood $\mathcal{L}$ is equal to Q (our much nicer loss) minus KL divergences minus entropies
- Both KL divergence and entropy are non-negative!

Theory of EM (part 3)

UNIVERSITÄT BIELEFELD
Faculty of Technology

$$-\sum_{i=1}^{N} \log[p_X(x_i)] = Q - \sum_{i=1}^{N} \mathcal{D}_{\mathsf{KL}}(\boldsymbol{\gamma_i} || p_{Z_i|X_i}) + \mathcal{H}(\boldsymbol{\gamma_i})$$

Interpretation:

- The true neg. log likelihood $\mathcal{L}$ is equal to Q (our much nicer loss) minus KL divergences minus entropies
- Both KL divergence and entropy are non-negative!

$\Rightarrow$ $\mathcal{L}$ is upper-bounded by $Q \Rightarrow$ optimizing Q improves our upper bound on $\mathcal{L}$

Theory of EM (part 3)

$$-\sum_{i=1}^{N} \log[p_X(x_i)] = Q - \sum_{i=1}^{N} \mathcal{D}_{\mathsf{KL}}(\boldsymbol{\gamma_i} || p_{Z_i|X_i}) + \mathcal{H}(\boldsymbol{\gamma_i})$$

Interpretation:

- The true neg. log likelihood $\mathcal{L}$ is equal to Q (our much nicer loss) minus KL divergences minus entropies
- Both KL divergence and entropy are non-negative!

$\Rightarrow$ $\mathcal{L}$ is upper-bounded by $Q \Rightarrow$ optimizing Q improves our upper bound on $\mathcal{L}$

- Even better: KL divergence gets zero if $\gamma_{i,k} = p_{Z_i|X_i}(k|x_i)$

Theory of EM (part 3)

$$-\sum_{i=1}^{N} \log[p_X(x_i)] = Q - \sum_{i=1}^{N} \mathcal{D}_{\mathsf{KL}}(\boldsymbol{\gamma_i} || p_{Z_i|X_i}) + \mathcal{H}(\boldsymbol{\gamma_i})$$

Interpretation:

- The true neg. log likelihood $\mathcal{L}$ is equal to Q (our much nicer loss) minus KL divergences minus entropies
- Both KL divergence and entropy are non-negative!

$\Rightarrow$ $\mathcal{L}$ is upper-bounded by Q $\Rightarrow$ optimizing Q improves our upper bound on $\mathcal{L}$

- Even better: KL divergence gets zero if $\gamma_{i,k} = p_{Z_i|X_i}(k|x_i)$

$\Rightarrow$ Every time we re-compute $\gamma_{i,k} = p_{Z_i|X_i}(k|x_i)$, we improve how well Q approximates $\mathcal{L}$

Theory of EM (part 3)

UNIVERSITÄT BIELEFELD
Faculty of Technology

$$-\sum_{i=1}^{N} \log[p_X(x_i)] = Q - \sum_{i=1}^{N} \mathcal{D}_{\mathsf{KL}}(\boldsymbol{\gamma_i}||p_{Z_i|X_i}) + \mathcal{H}(\boldsymbol{\gamma_i})$$

Interpretation:

- ▶ The true neg. log likelihood $\mathcal{L}$ is equal to Q (our much nicer loss) minus KL divergences minus entropies
- ▶ Both KL divergence and entropy are non-negative!
- $\Rightarrow$ $\mathcal{L}$ is upper-bounded by Q $\Rightarrow$ optimizing Q improves our upper bound on $\mathcal{L}$
- ▶ Even better: KL divergence gets zero if $\gamma_{i,k} = p_{Z_i|X_i}(k|x_i)$
- $\Rightarrow$ Every time we re-compute $\gamma_{i,k} = p_{Z_i|X_i}(k|x_i)$, we improve how well Q approximates $\mathcal{L}$
- ▶ Entropy gets zero if every point belongs exactly to one cluster

Theory of EM (part 3)

UNIVERSITÄT BIELEFELD
Faculty of Technology

$$-\sum_{i=1}^{N} \log[p_X(x_i)] = Q - \sum_{i=1}^{N} \mathcal{D}_{\mathsf{KL}}(\boldsymbol{\gamma_i} || p_{Z_i|X_i}) + \mathcal{H}(\boldsymbol{\gamma_i})$$

Interpretation:

- The true neg. log likelihood $\mathcal{L}$ is equal to Q (our much nicer loss) minus KL divergences minus entropies
- Both KL divergence and entropy are non-negative!

$\Rightarrow$ $\mathcal{L}$ is upper-bounded by Q $\Rightarrow$ optimizing Q improves our upper bound on $\mathcal{L}$

- Even better: KL divergence gets zero if $\gamma_{i,k} = p_{Z_i|X_i}(k|x_i)$

$\Rightarrow$ Every time we re-compute $\gamma_{i,k} = p_{Z_i|X_i}(k|x_i)$, we improve how well Q approximates $\mathcal{L}$

- Entropy gets zero if every point belongs exactly to one cluster

$\Rightarrow$ The more clearly defined our clusters get over time, the better Q approximates $\mathcal{L}$

EM algorithm for GMMs

function GMM(data matrix $\boldsymbol{X}$ with N rows and m columns, desired number of Gaussians K)

end function

EM algorithm for GMMs

function GMM(data matrix $\boldsymbol{X}$ with N rows and m columns, desired number of Gaussians K)

Randomly initialize $\mu_1, \ldots, \mu_K \in \mathbb{R}^m$ (in the convex hull of the data).

end function

UNIVERSITÄT BIELEFELD
Faculty of Technology

EM algorithm for GMMs

function GMM(data matrix $\boldsymbol{X}$ with N rows and m columns, desired number of Gaussians K)

  Randomly initialize $\mu_1, \ldots, \mu_K \in \mathbb{R}^m$ (in the convex hull of the data).

  Initialize $p_Z(k) \leftarrow \frac{1}{K}$, initialize $\Sigma_k \leftarrow \boldsymbol{I}$.

end function

EM algorithm for GMMs

function GMM(data matrix $\boldsymbol{X}$ with N rows and m columns, desired number of Gaussians K)

  Randomly initialize $\mu_1, \ldots, \mu_K \in \mathbb{R}^m$ (in the convex hull of the data).

  Initialize $p_Z(k) \leftarrow \frac{1}{K}$, initialize $\Sigma_k \leftarrow \boldsymbol{I}$.

  for desired number of iterations **do**

  end for

end function

function GMM(data matrix $\boldsymbol{X}$ with N rows and m columns, desired number of Gaussians K)

Randomly initialize $\mu_1, \ldots, \mu_K \in \mathbb{R}^m$ (in the convex hull of the data).

Initialize $p_Z(k) \leftarrow \frac{1}{K}$, initialize $\Sigma_k \leftarrow \boldsymbol{I}$.

for desired number of iterations **do**

Compute $\gamma_{k,i} = p_{Z|X}(k|x_i)$ for all $i \in \{1, \ldots, N\}$ and all $k \in \{1, \ldots, K\}$. ▷ Expectation step

end for

end function

EM algorithm for GMMs

function GMM(data matrix $\boldsymbol{X}$ with N rows and m columns, desired number of Gaussians K)

  Randomly initialize $\mu_1, \ldots, \mu_K \in \mathbb{R}^m$ (in the convex hull of the data).

  Initialize $p_Z(k) \leftarrow \frac{1}{K}$, initialize $\Sigma_k \leftarrow \boldsymbol{I}$.

  for desired number of iterations **do**

    Compute $\gamma_{k,i} = p_{Z|X}(k|x_i)$ for all $i \in \{1, \ldots, N\}$ and all $k \in \{1, \ldots, K\}$.

    ▷ Expectation step

    Set $p_Z(k) \leftarrow \frac{1}{N} \sum_{i=1}^{N} \gamma_{i,k}$ for all $k \in \{1, \ldots, K\}$.

    Set $\mu_k \leftarrow \frac{\sum_{i=1}^{N} \gamma_{i,k} \cdot x_k}{\sum_{i=1}^{N} \gamma_{i,k}}$ for all $k \in \{1, \ldots, K\}$.

    Set $\Sigma_k \leftarrow \frac{\sum_{i=1}^{N} \gamma_{i,k} \cdot (x_k - \mu_k)(x_k - \mu_k)^T}{\sum_{i=1}^{N} \gamma_{i,k}}$ for all $k \in \{1, \ldots, K\}$.

    ▷ Maximization step

  end for

end function

EM algorithm for GMMs

function GMM(data matrix $\boldsymbol{X}$ with N rows and m columns, desired number of Gaussians K)

Randomly initialize $\mu_1, \ldots, \mu_K \in \mathbb{R}^m$ (in the convex hull of the data).

Initialize $p_Z(k) \leftarrow \frac{1}{K}$, initialize $\Sigma_k \leftarrow \boldsymbol{I}$.

for desired number of iterations **do**

Compute $\gamma_{k,i} = p_{Z|X}(k|x_i)$ for all $i \in \{1, \ldots, N\}$ and all $k \in \{1, \ldots, K\}$.

▷ Expectation step

Set $p_Z(k) \leftarrow \frac{1}{N} \sum_{i=1}^N \gamma_{i,k}$ for all $k \in \{1, \ldots, K\}$.

Set $\mu_k \leftarrow \frac{\sum_{i=1}^N \gamma_{i,k} \cdot x_k}{\sum_{i=1}^N \gamma_{i,k}}$ for all $k \in \{1, \ldots, K\}$.

Set $\Sigma_k \leftarrow \frac{\sum_{i=1}^N \gamma_{i,k} \cdot (x_k - \mu_k)(x_k - \mu_k)^T}{\sum_{i=1}^N \gamma_{i,k}}$ for all $k \in \{1, \ldots, K\}$.

▷ Maximization step

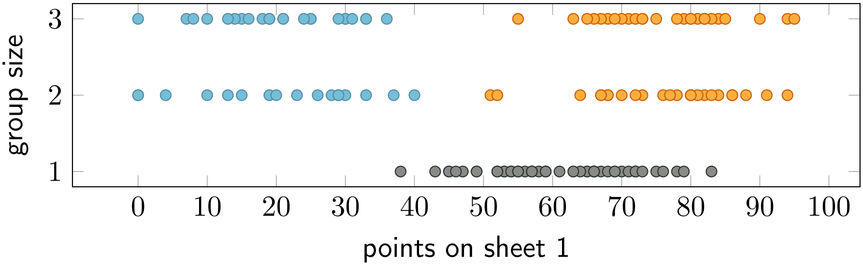
end for

return p_Z, $\mu_1, \ldots, \mu_K$, $\Sigma_1, \ldots, \Sigma_K$.

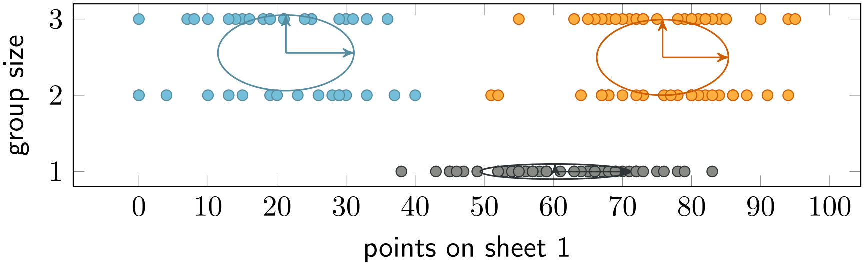
end function

GMM on example data

UNIVERSITÄT BIELEFELD
Faculty of Technology



GMM on example data



Summary

- K-Means discovers clusters by assigning data to closest prototype and prototypes to cluster means

Summary

UNIVERSITÄT BIELEFELD
Faculty of Technology

- K-Means discovers clusters by assigning data to closest prototype and prototypes to cluster means
- Sensitive to data scaling and "unlucky" initializations

Summary

- K-Means discovers clusters by assigning data to closest prototype and prototypes to cluster means
- Sensitive to data scaling and "unlucky" initializations
- Selection of K: Silhouette score and/or BIC

Summary

UNIVERSITÄT BIELEFELD
Faculty of Technology

- K-Means discovers clusters by assigning data to closest prototype and prototypes to cluster means
- Sensitive to data scaling and "unlucky" initializations
- Selection of K: Silhouette score and/or BIC
- Gaussian mixture models can handle scaling and elliptical cluster shapes – but need much more computation

Literature I

Barber, David (2012). **Bayesian Reasoning and Machine Learning**. Cambridge, UK: Cambridge University Press. url: http://www.cs.ucl.ac.uk/staff/d.barber/brml/.

Jain, Anil K. (2010). "Data clustering: 50 years beyond K-means". In: **Pattern Recognition Letters** 31.8. Award winning papers from the 19th International Conference on Pattern Recognition (ICPR), pp. 651–666. doi: 10.1016/j.patrec.2009.09.011.