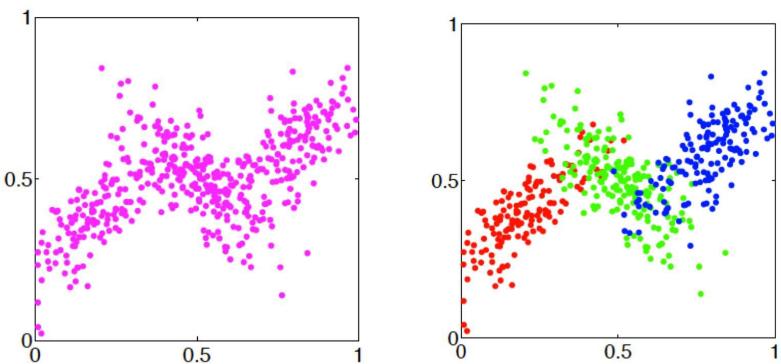


# Gaussian Mixture Models and EM algorithm

Radek Danecek

# Gaussian Mixture Model

- Unsupervised method
- Fit multimodal Gaussian distributions



# Formal Definition

- The model is described as:

$$p(\mathbf{x}|\theta) = \sum_{k=1}^K \pi_k \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k), \quad \pi_k > 0, \quad \sum_k \pi_k = 1,$$

- The parameters of the model are:

$$\theta = (\pi_1, \dots, \pi_K, \boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_K, \boldsymbol{\Sigma}_1, \dots, \boldsymbol{\Sigma}_K)$$

- The training data is unlabeled – unsupervised setting

- Why not fit with MLE?

# Optimization problem

- Model:

$$p(\mathbf{x}|\theta) = \sum_{k=1}^K \pi_k \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k), \quad \pi_k > 0, \quad \sum_k \pi_k = 1,$$

$$\theta = (\pi_1, \dots, \pi_K, \boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_K, \boldsymbol{\Sigma}_1, \dots, \boldsymbol{\Sigma}_K)$$

- Apply MLE:

- Maximize:

$$L(\theta) = \sum_{n=1}^N \log \sum_{k=1}^K \pi_k \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

- Difficult, non convex optimization with constraints

- Use EM algorithm instead

# EM Algorithm for GMMs

- Idea:
  - Objective function:  $L(\theta) = \sum_{n=1}^N \log \sum_{k=1}^K \pi_k \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$
  - Split optimization of the objective into two parts
- Algorithm:
  - Initialize model parameters (randomly):  $\theta = (\pi_1, \dots, \pi_K, \boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_K, \boldsymbol{\Sigma}_1, \dots, \boldsymbol{\Sigma}_K)$
  - Iterate until convergence:
    - **E-step**
      - Assign cluster probabilities ("soft labels") to each sample
    - **M-step**
      - Solve the MLE using the soft labels

# Initialization

- Initialize model parameters (randomly)

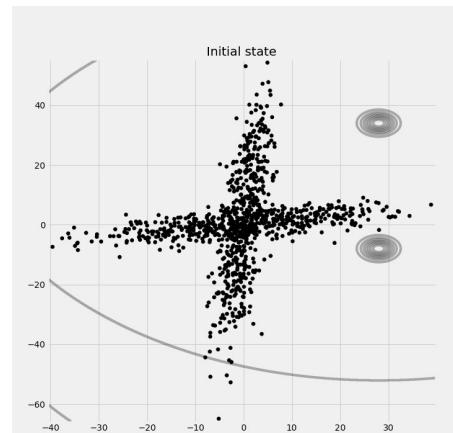
$$\theta = (\pi_1, \dots, \pi_K, \mu_1, \dots, \mu_K, \Sigma_1, \dots, \Sigma_K)$$

- Uniform for cluster probabilities

- Centers
  - Random
  - K-means heuristics

- Covariances:

- Spherical, according to empirical variance



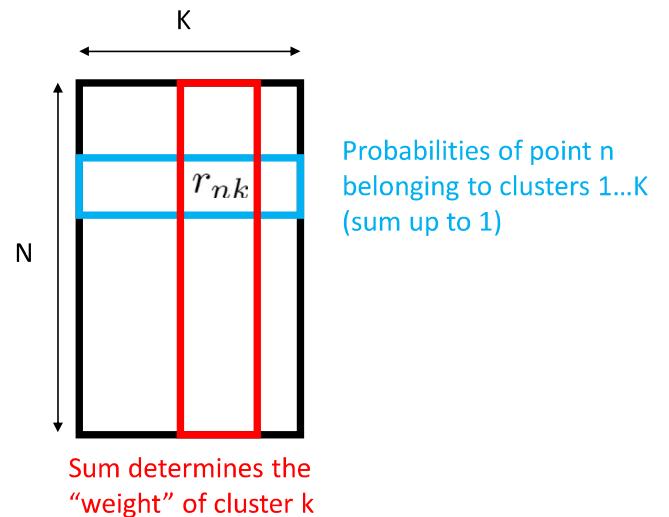
## E-step

- For each data point  $x_n$  and each cluster  $k$ , compute the probability that  $x_n$  belongs to  $k$  (given current model parameters)

$$\theta = (\pi_1, \dots, \pi_K, \boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_K, \boldsymbol{\Sigma}_1, \dots, \boldsymbol{\Sigma}_K)$$

$$r_{nk} := p(z_n = k | \mathbf{x}_n) = \frac{\pi_k \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{j=1}^K \pi_j \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)}.$$

“soft labels”

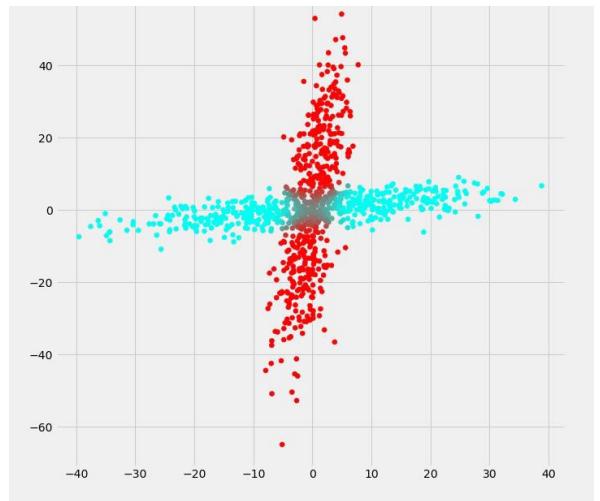


## E-step

- For each data point  $x_n$  and each cluster  $k$ , compute the probability that  $x_n$  belongs to  $k$  (given current model parameters)

$$\theta = (\pi_1, \dots, \pi_K, \boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_K, \boldsymbol{\Sigma}_1, \dots, \boldsymbol{\Sigma}_K)$$

$$r_{nk} := p(z_n = k | \mathbf{x}_n) = \frac{\pi_k \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{j=1}^K \pi_j \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)}.$$



“soft labels”

# M-step

- Now we have “soft labels” for the data -> fall back to supervised MLE

- Optimize the log likelihood:

- Instead of the original (difficult objective):  $L(\theta) = \sum_{n=1}^N \log \sum_{k=1}^K \pi_k \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$   
We optimize the following:

$$L(\theta) = \mathbb{E}[p(x, z | \boldsymbol{\theta})] = \sum_{n=1}^N \sum_{k=1}^K r_{nk} (\log (\pi_k \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)))$$

- Differentiate w.r.t.  $\theta = (\pi_1, \dots, \pi_K, \boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_K, \boldsymbol{\Sigma}_1, \dots, \boldsymbol{\Sigma}_K)$

$$\pi_k = \frac{\sum_{n=1}^N r_{nk}}{\sum_{k=1}^K \sum_{n=1}^N r_{nk}} \quad \boldsymbol{\mu}_k = \frac{\sum_{n=1}^N r_{nk} \mathbf{x}_n}{\sum_{n=1}^N r_{nk}} \quad \boldsymbol{\Sigma}_k = \frac{\sum_{n=1}^N r_{nk} (\mathbf{x}_n - \boldsymbol{\mu}_k)(\mathbf{x}_n - \boldsymbol{\mu}_k)^T}{\sum_{n=1}^N r_{nk}}$$

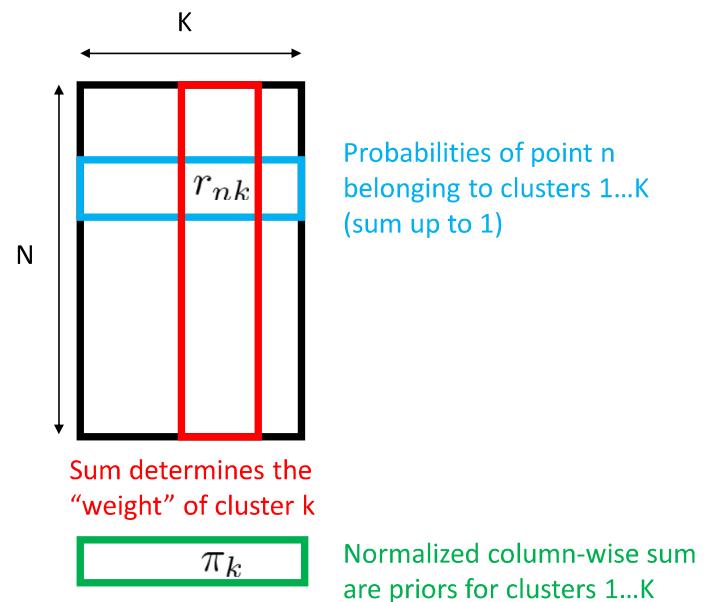
# M-step

- Update model parameters:

$$\theta = (\pi_1, \dots, \pi_K, \mu_1, \dots, \mu_K, \Sigma_1, \dots, \Sigma_K)$$

- Update prior for each cluster:

$$\pi_j = \frac{\sum_{n=1}^N r_{nj}}{\sum_{k=1}^K \sum_{n=1}^N r_{nk}}$$



# M-step

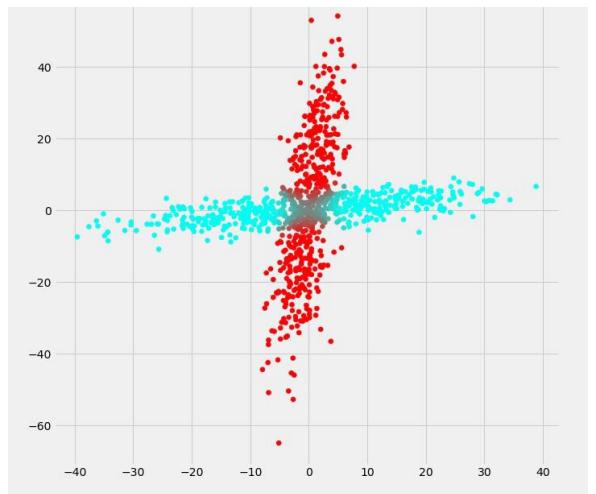
- Update model parameters:

$$\theta = (\pi_1, \dots, \pi_K, \boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_K, \boldsymbol{\Sigma}_1, \dots, \boldsymbol{\Sigma}_K)$$

- Update mean and covariance of each cluster

$$\boldsymbol{\mu}_k = \frac{\sum_{n=1}^N r_{nk} \mathbf{x}_n}{\sum_{n=1}^N r_{nk}}$$

$$\boldsymbol{\Sigma}_k = \frac{\sum_{n=1}^N r_{nk} (\mathbf{x}_n - \boldsymbol{\mu}_k)(\mathbf{x}_n - \boldsymbol{\mu}_k)^T}{\sum_{n=1}^N r_{nk}}$$



# M-step

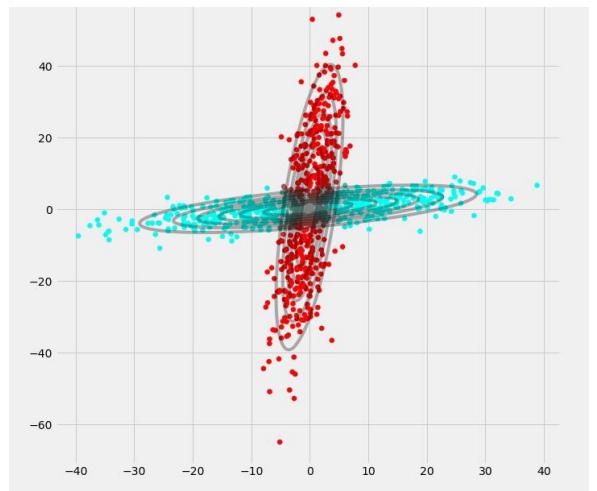
- Update model parameters:

$$\theta = (\pi_1, \dots, \pi_K, \boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_K, \boldsymbol{\Sigma}_1, \dots, \boldsymbol{\Sigma}_K)$$

- Update mean and covariance of each cluster

$$\boldsymbol{\mu}_k = \frac{\sum_{n=1}^N r_{nk} \mathbf{x}_n}{\sum_{n=1}^N r_{nk}}$$

$$\boldsymbol{\Sigma}_k = \frac{\sum_{n=1}^N r_{nk} (\mathbf{x}_n - \boldsymbol{\mu}_k)(\mathbf{x}_n - \boldsymbol{\mu}_k)^T}{\sum_{n=1}^N r_{nk}}$$



# EM Algorithm for GMMs

- Idea:

- Objective function:  $L(\theta) = \sum_{n=1}^N \log \sum_{k=1}^K \pi_k \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$
- Split optimization of the objective into two parts

- Algorithm:

- Initialize model parameters (randomly):  $\theta = (\pi_1, \dots, \pi_K, \boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_K, \boldsymbol{\Sigma}_1, \dots, \boldsymbol{\Sigma}_K)$
- Iterate until convergence:

- E-step

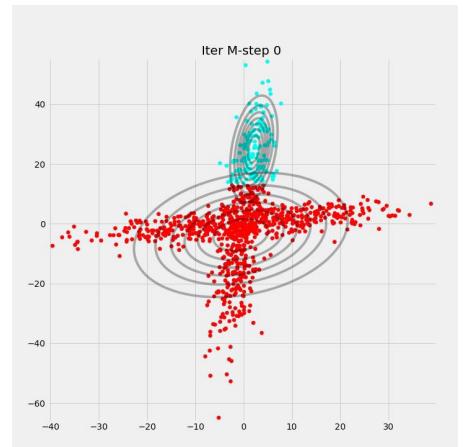
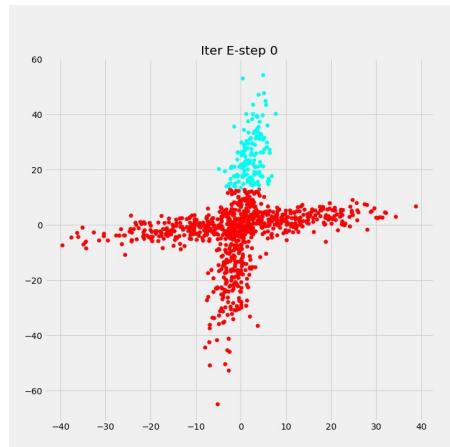
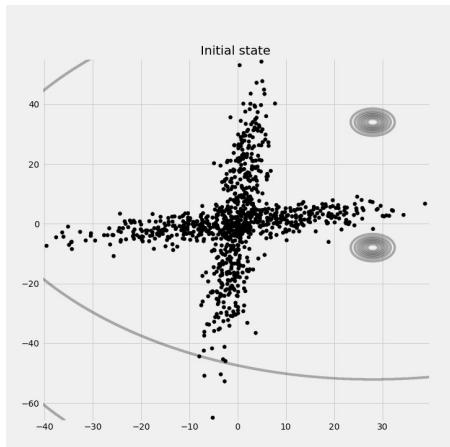
- Assign cluster probabilities (“soft labels”) to each sample  $r_{nk} := p(z_n = k | \mathbf{x}_n) = \frac{\pi_k \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{j=1}^K \pi_j \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)}$

- M-step

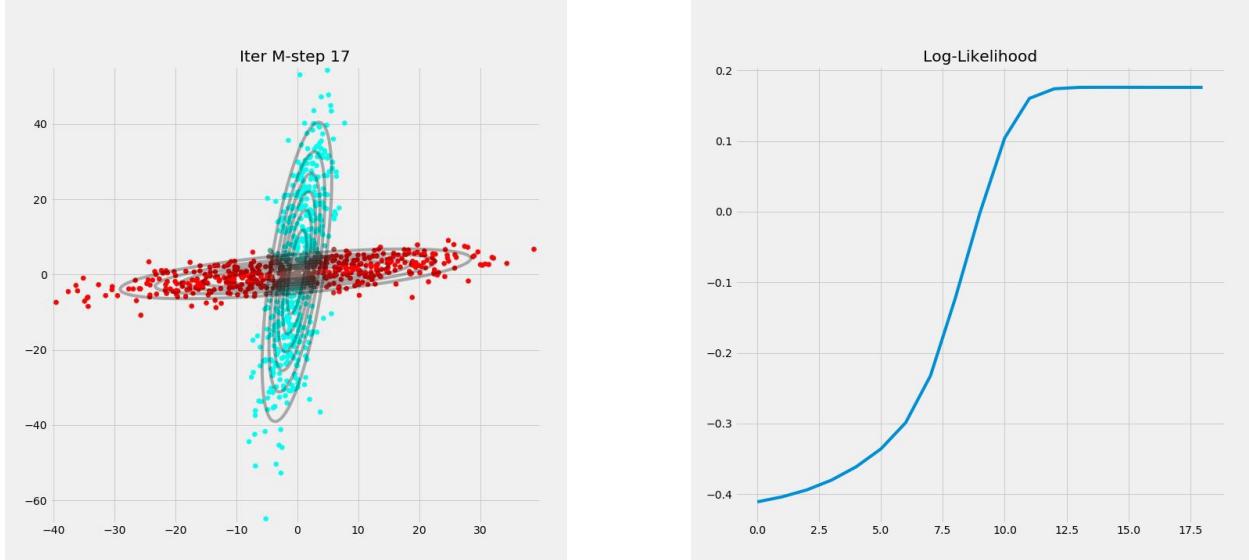
- Find optimal parameters given the soft labels

$$\pi_k = \frac{\sum_{n=1}^N r_{nk}}{\sum_{k=1}^K \sum_{n=1}^N r_{nk}} \quad \boldsymbol{\mu}_k = \frac{\sum_{n=1}^N r_{nk} \mathbf{x}_n}{\sum_{n=1}^N r_{nk}} \quad \boldsymbol{\Sigma}_k = \frac{\sum_{n=1}^N r_{nk} (\mathbf{x}_n - \boldsymbol{\mu}_k)(\mathbf{x}_n - \boldsymbol{\mu}_k)^T}{\sum_{n=1}^N r_{nk}}$$

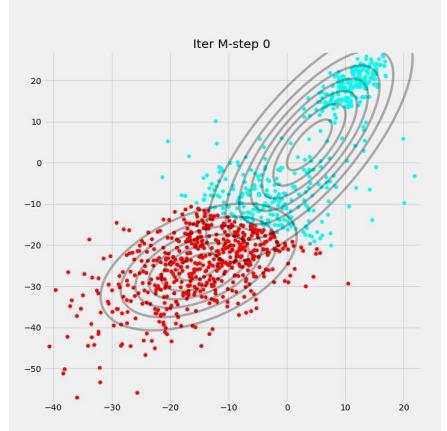
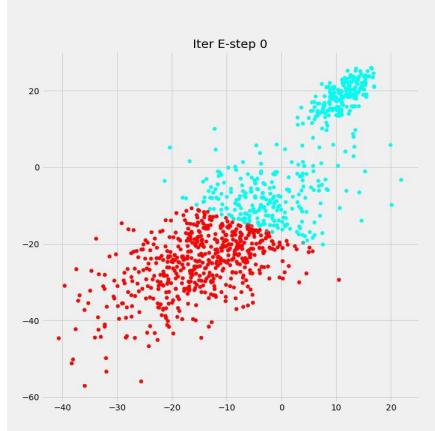
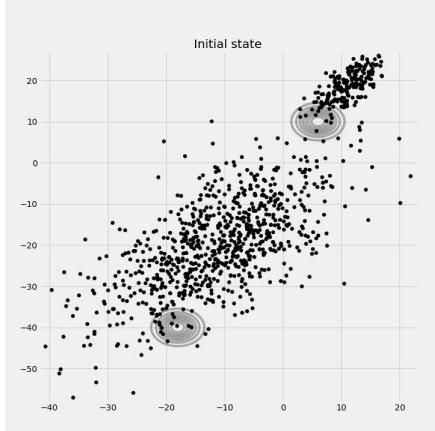
# Overlapping clusters



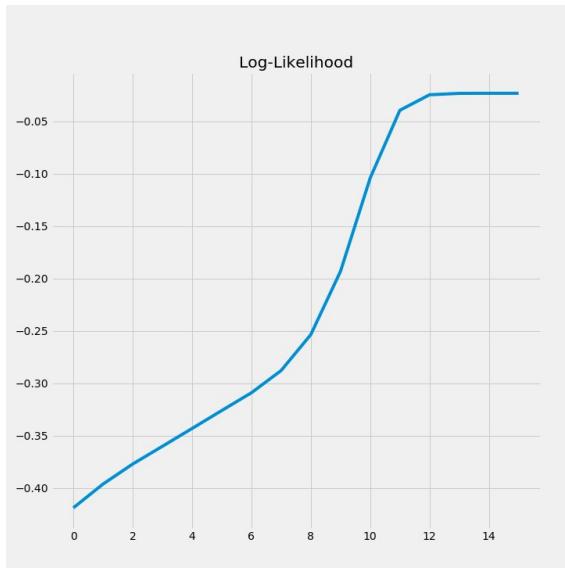
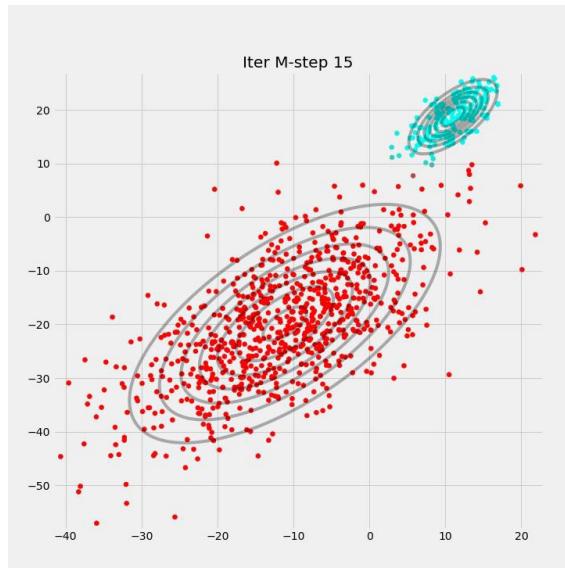
# Overlapping clusters



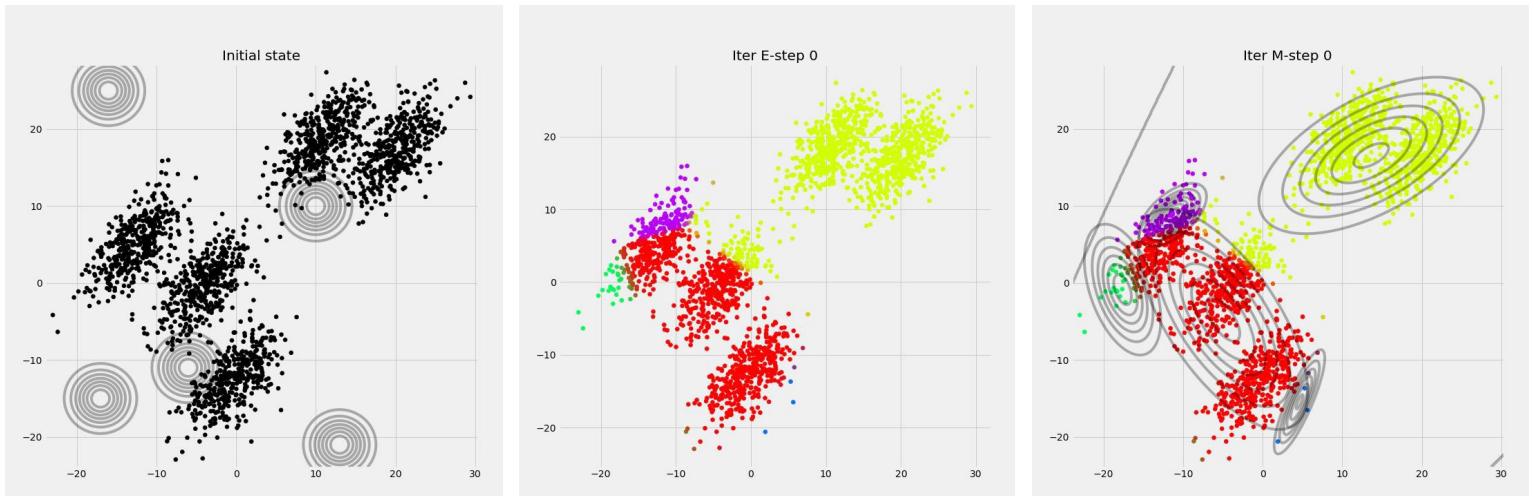
# Unequal cluster size



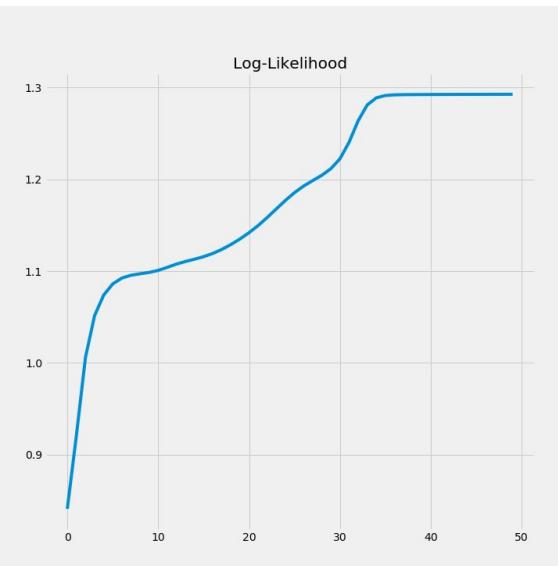
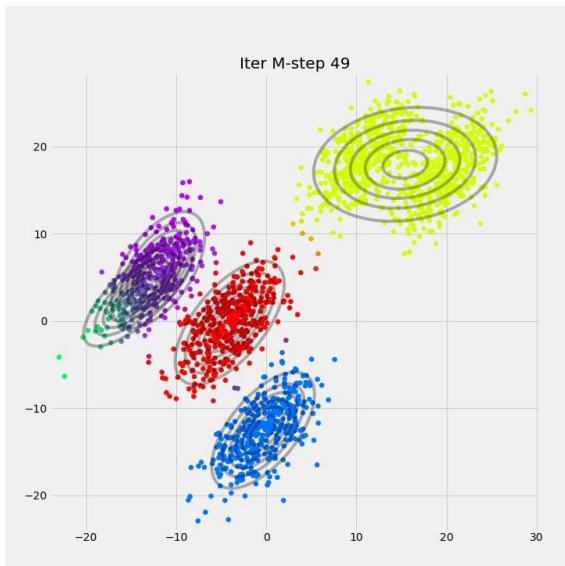
# Imbalanced cluster size



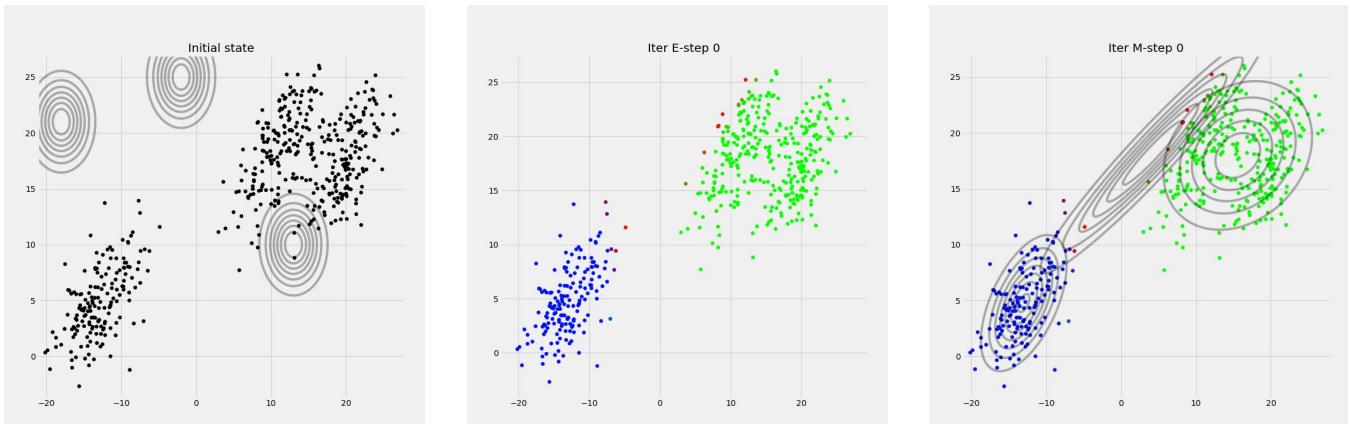
# Sensitivity to Initialization



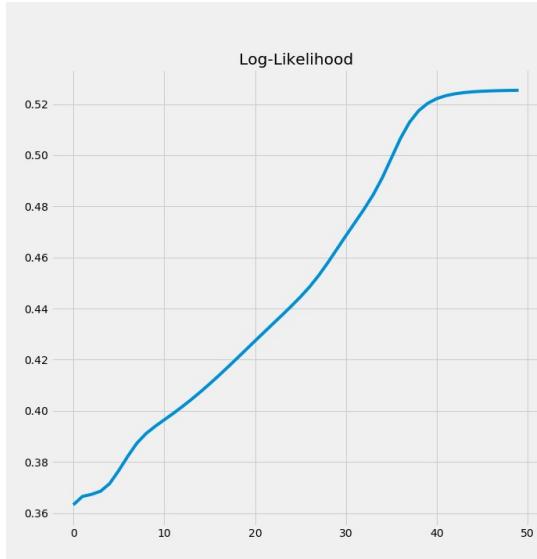
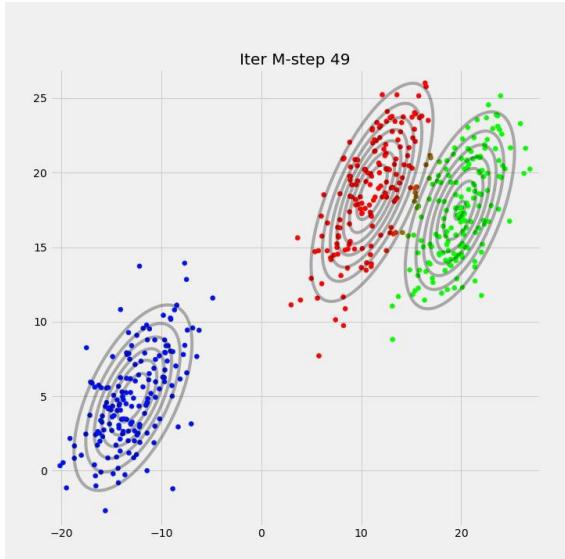
# Sensitivity to Initialization



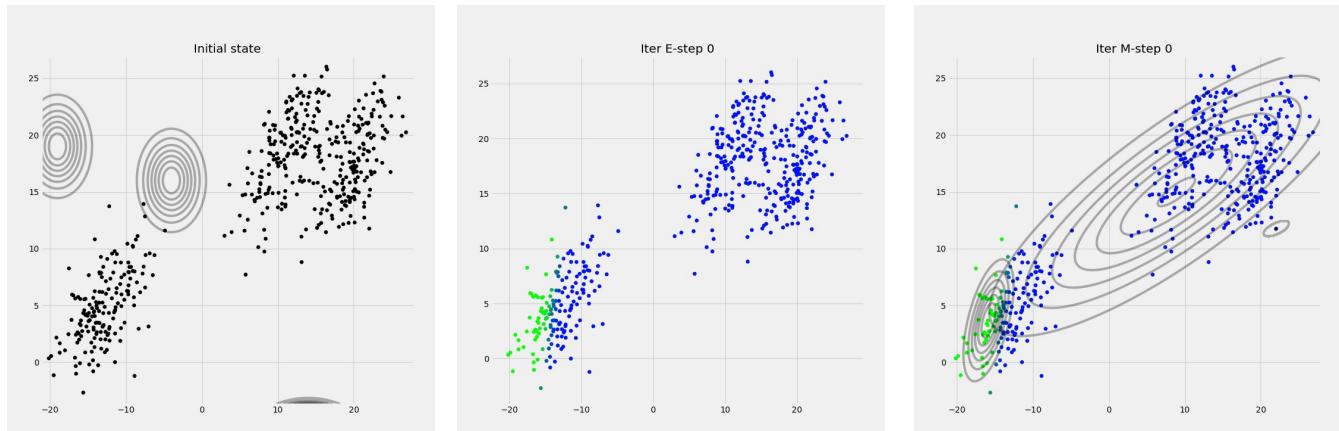
# Sensitivity to Initialization



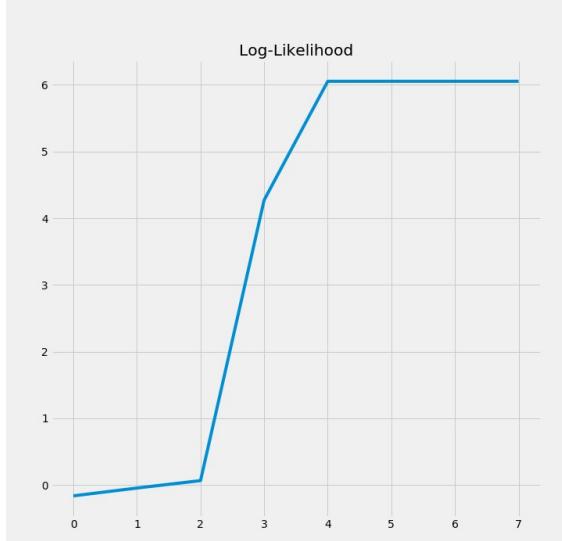
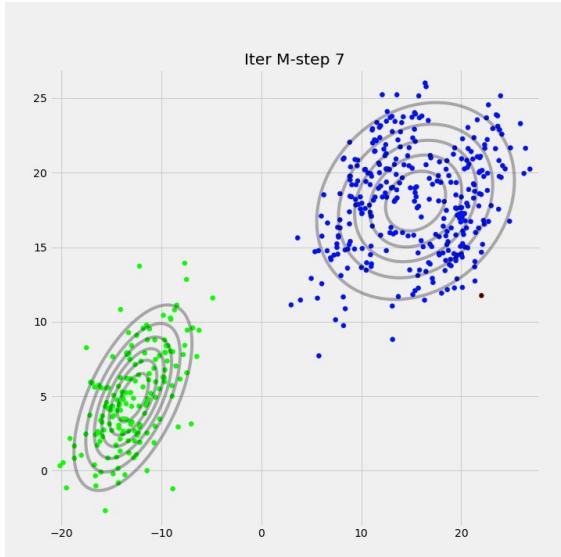
# Sensitivity to Initialization



# Sensitivity to Initialization



# Sensitivity to initialization



# Degenerate covariance

- The determinant of the covariance matrix tends to 0

$$\boldsymbol{\Sigma}_k = \frac{\sum_{n=1}^N r_{nk}(\mathbf{x}_n - \boldsymbol{\mu}_k)(\mathbf{x}_n - \boldsymbol{\mu}_k)^T}{\sum_{n=1}^N r_{nk}} + \lambda \mathbf{I}$$

# Practical Example – Color segmentation

- Input: an image  $\mathcal{I} \in \mathbb{R}^{w \times h \times c}$
- Can be thought of as a dataset of 3D (color) samples

$$\mathbf{X} \in \mathbb{R}^{wh \times c}$$

- Run 3D GMM clustering over  $\mathbf{X}$

# Practical Example – Color segmentation

Input Image



# Practical Example – Color segmentation

3 clusters



# Practical Example – Color segmentation

4 clusters



# Practical Example – Color segmentation

5 clusters



# Practical Example – Color segmentation

7 clusters



# Practical Example – Color segmentation

8 clusters



# Practical Example – Color segmentation

9 clusters



# Practical Example – Color segmentation

10 clusters



# Practical Example – Color segmentation

15 clusters



# Practical Example – Color segmentation

20 clusters



# Practical Example – Color segmentation

Input Image



# EM Algorithm for GMMs

- Idea:

- Objective function:  $L(\theta) = \sum_{n=1}^N \log \sum_{k=1}^K \pi_k \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$
- Split optimization of the objective into two parts

- Algorithm:

- Initialize model parameters (randomly):  $\theta = (\pi_1, \dots, \pi_K, \boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_K, \boldsymbol{\Sigma}_1, \dots, \boldsymbol{\Sigma}_K)$
- Iterate until convergence:

- E-step

- Assign cluster probabilities (“soft labels”) to each sample  $r_{nk} := p(z_n = k | \mathbf{x}_n) = \frac{\pi_k \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{j=1}^K \pi_j \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)}$

- M-step

- Find optimal parameters given the soft labels

$$\pi_k = \frac{\sum_{n=1}^N r_{nk}}{\sum_{k=1}^K \sum_{n=1}^N r_{nk}}$$

$$\boldsymbol{\mu}_k = \frac{\sum_{n=1}^N r_{nk} \mathbf{x}_n}{\sum_{n=1}^N r_{nk}}$$

$$\boldsymbol{\Sigma}_k = \frac{\sum_{n=1}^N r_{nk} (\mathbf{x}_n - \boldsymbol{\mu}_k)(\mathbf{x}_n - \boldsymbol{\mu}_k)^T}{\sum_{n=1}^N r_{nk}}$$

# Generalized EM

- Idea:

- Objective function:  $L(\theta) = \sum_{n=1}^N \log \sum_{k=1}^K \pi_k \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$
- Split optimization of the objective into two parts

- Algorithm:

- Initialize model parameters (randomly):  $\theta = (\pi_1, \dots, \pi_K, \boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_K, \boldsymbol{\Sigma}_1, \dots, \boldsymbol{\Sigma}_K)$
- Iterate until convergence:

- E-step

- Assign cluster probabilities (“soft labels”) to each sample  $r_{nk} := p(z_n = k | \mathbf{x}_n) = \frac{\pi_k \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{j=1}^K \pi_j \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)}$

- M-step

- Find optimal parameters given the soft labels

$$\pi_k = \frac{\sum_{n=1}^N r_{nk}}{\sum_{k=1}^K \sum_{n=1}^N r_{nk}}$$
$$\boldsymbol{\mu}_k = \frac{\sum_{n=1}^N r_{nk} \mathbf{x}_n}{\sum_{n=1}^N r_{nk}}$$
$$\boldsymbol{\Sigma}_k = \frac{\sum_{n=1}^N r_{nk} (\mathbf{x}_n - \boldsymbol{\mu}_k)(\mathbf{x}_n - \boldsymbol{\mu}_k)^T}{\sum_{n=1}^N r_{nk}}$$

# Generalized EM

- Idea:

- Objective function:  $L(\theta) = \sum_{n=1}^N \log \sum_{k=1}^K \pi_k p_k(\mathbf{x}_n | \boldsymbol{\theta}_k)$
- Split optimization of the objective into two parts

- Algorithm:

- Initialize model parameters (randomly):  $\theta = (\pi_1, \dots, \pi_K, \theta_1, \dots, \theta_K)$

- Iterate until convergence:

- E-step

- Assign cluster probabilities (“soft labels”) to each sample  $r_{nk} := p(z_n = k | \mathbf{x}_n) = \frac{\pi_k \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{j=1}^K \pi_j \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)}$

- M-step

- Find optimal parameters given the soft labels

$$\pi_k = \frac{\sum_{n=1}^N r_{nk}}{\sum_{k=1}^K \sum_{n=1}^N r_{nk}}$$
$$\boldsymbol{\mu}_k = \frac{\sum_{n=1}^N r_{nk} \mathbf{x}_n}{\sum_{n=1}^N r_{nk}}$$
$$\boldsymbol{\Sigma}_k = \frac{\sum_{n=1}^N r_{nk} (\mathbf{x}_n - \boldsymbol{\mu}_k)(\mathbf{x}_n - \boldsymbol{\mu}_k)^T}{\sum_{n=1}^N r_{nk}}$$

# Generalized EM

- Idea:

- Objective function:  $L(\theta) = \sum_{n=1}^N \log \sum_{k=1}^K \pi_k p_k(\mathbf{x}_n | \boldsymbol{\theta}_k)$
- Split optimization of the objective into two parts

- Algorithm:

- Initialize model parameters (randomly):  $\theta = (\pi_1, \dots, \pi_K, \theta_1, \dots, \theta_K)$

- Iterate until convergence:

- E-step

- Assign cluster probabilities (“soft labels”) to each sample  $r_{nk} := p(z_n = k | \mathbf{x}_n) = \frac{\pi_k \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{j=1}^K \pi_j \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)}$

- M-step

- Find optimal parameters given the soft labels

$$\pi_k = \frac{\sum_{n=1}^N r_{nk}}{\sum_{k=1}^K \sum_{n=1}^N r_{nk}}$$

$$\boldsymbol{\mu}_k = \frac{\sum_{n=1}^N r_{nk} \mathbf{x}_n}{\sum_{n=1}^N r_{nk}}$$

$$\boldsymbol{\Sigma}_k = \frac{\sum_{n=1}^N r_{nk} (\mathbf{x}_n - \boldsymbol{\mu}_k)(\mathbf{x}_n - \boldsymbol{\mu}_k)^T}{\sum_{n=1}^N r_{nk}}$$

# Generalized EM

- Idea:

- Objective function:  $L(\theta) = \sum_{n=1}^N \log \sum_{k=1}^K \pi_k p_k(\mathbf{x}_n | \boldsymbol{\theta}_k)$
- Split optimization of the objective into two parts

- Algorithm:

- Initialize model parameters (randomly):  $\theta = (\pi_1, \dots, \pi_K, \theta_1, \dots, \theta_K)$
- Iterate until convergence:

- E-step

- Assign cluster probabilities (“soft labels”) to each sample  $r_{nk} := p(z_n = k | \mathbf{x}_n) = \frac{\pi_k p_k(\mathbf{x}_n | \theta_k)}{\sum_{j=1}^K \pi_j p_j(\mathbf{x}_n | \theta_j)}$

- M-step

- Find optimal parameters given the soft labels

$$\pi_k = \frac{\sum_{n=1}^N r_{nk}}{\sum_{k=1}^K \sum_{n=1}^N r_{nk}}$$
$$\boldsymbol{\mu}_k = \frac{\sum_{n=1}^N r_{nk} \mathbf{x}_n}{\sum_{n=1}^N r_{nk}}$$
$$\boldsymbol{\Sigma}_k = \frac{\sum_{n=1}^N r_{nk} (\mathbf{x}_n - \boldsymbol{\mu}_k)(\mathbf{x}_n - \boldsymbol{\mu}_k)^T}{\sum_{n=1}^N r_{nk}}$$

# Generalized M-step

- What is the objective function?

- GMM:

$$L(\theta) = \mathbb{E}[p(x, z|\theta)] = \sum_{n=1}^N \sum_{k=1}^K r_{nk} (\log (\pi_k \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)))$$

- General:

$$L(\theta) = \mathbb{E}[p(x, z|\theta)] = \sum_{n=1}^N \sum_{k=1}^K r_{nk} (\log (\pi_k p_k(\mathbf{x}_n | \theta_k)))$$

# Exercise

- Consider a mixture of K multivariate Bernoulli distributions with parameters  $\mu = \{\mu_1, \dots, \mu_K\}$ , where  $\mu_k = \{\mu_{k1}, \dots, \mu_{kd}\}$

- Multivariate Bernoulli distribution:

$$p_k(\mathbf{x}|\boldsymbol{\mu}) = \prod_{d=1}^D \mu_{kd}^{x_d} (1 - \mu_{kd})^{1-x_d}$$

- Question 1: Write down the equation for the E-step update

hint GMM:

$$r_{nk} := p(z_n = k | \mathbf{x}_n) = \frac{\pi_k \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{j=1}^K \pi_j \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)}$$

Answer:

# Exercise

- Consider a mixture of K multivariate Bernoulli distributions with parameters  $\mu = \{\mu_1, \dots, \mu_K\}$ , where  $\mu_k = \{\mu_{k1}, \dots, \mu_{kd}\}$

- Multivariate Bernoulli distribution:

$$p_k(\mathbf{x}|\boldsymbol{\mu}) = \prod_{d=1}^D \mu_{kd}^{x_d} (1 - \mu_{kd})^{1-x_d}$$

- Question 1: Write down the equation for the E-step update

hint GMM:

$$r_{nk} := p(z_n = k | \mathbf{x}_n) = \frac{\pi_k \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{j=1}^K \pi_j \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)}$$

Answer:

$$r_{nk} := p(z_n = k | \mathbf{x}_n) = \frac{\pi_k \prod_{d=1}^D \mu_{kd}^{x_d} (1 - \mu_{kd})^{1-x_d}}{\sum_{j=1}^K \pi_j \prod_{d=1}^D \mu_{jd}^{x_d} (1 - \mu_{jd})^{1-x_d}}$$

# Exercise

- Consider a mixture of K multivariate Bernoulli distributions with parameters  $\mu = \{\mu_1, \dots, \mu_K\}$ , where  $\mu_k = \{\mu_{k1}, \dots, \mu_{kd}\}$
- Multivariate Bernoulli distribution:

$$p_k(\mathbf{x}|\boldsymbol{\mu}) = \prod_{d=1}^D \mu_{kd}^{x_d} (1 - \mu_{kd})^{1-x_d}$$

- Question 2: Write down the EM objective:

$$L(\theta) = \mathbb{E}[p(x, z|\boldsymbol{\theta})] = \sum_{n=1}^N \sum_{k=1}^K r_{nk} (\log (\pi_k p_k(\mathbf{x}_n|\boldsymbol{\theta}_k)))$$

# Exercise

- Consider a mixture of K multivariate Bernoulli distributions with parameters  $\mu = \{\mu_1, \dots, \mu_K\}$ , where  $\mu_k = \{\mu_{k1}, \dots, \mu_{kd}\}$
- Multivariate Bernoulli distribution:

$$p_k(\mathbf{x}|\boldsymbol{\mu}) = \prod_{d=1}^D \mu_{kd}^{x_d} (1 - \mu_{kd})^{1-x_d}$$

- Question 2: Write down the EM objective:

$$\begin{aligned} L(\theta) &= \mathbb{E}[p(x, z|\boldsymbol{\theta})] = \sum_{n=1}^N \sum_{k=1}^K r_{nk} (\log (\pi_k p_k(\mathbf{x}_n|\boldsymbol{\theta}_k))) \\ L(\theta) &= \mathbb{E}[p(x, z|\boldsymbol{\theta})] = \sum_{n=1}^N \sum_{k=1}^K r_{nk} \left( \log \left( \pi_k \prod_{d=1}^D \mu_{kd}^{x_{nd}} (1 - \mu_{kd})^{1-x_{nd}} \right) \right) \end{aligned}$$

# Exercise

- Multivariate Bernoulli distribution:

$$p_k(\mathbf{x}|\boldsymbol{\mu}) = \prod_{d=1}^D \mu_{kd}^{x_d} (1 - \mu_{kd})^{1-x_d}$$

- EM objective:

$$L(\theta) = \mathbb{E}[p(x, z|\boldsymbol{\theta})] = \sum_{n=1}^N \sum_{k=1}^K r_{nk} (\log (\pi_k p_k(\mathbf{x}_n|\boldsymbol{\theta}_k)))$$

# Exercise

- Multivariate Bernoulli distribution:

$$p_k(\mathbf{x}|\boldsymbol{\mu}) = \prod_{d=1}^D \mu_{kd}^{x_d} (1 - \mu_{kd})^{1-x_d}$$

- EM objective:

$$L(\theta) = \mathbb{E}[p(x, z|\boldsymbol{\theta})] = \sum_{n=1}^N \sum_{k=1}^K r_{nk} (\log (\pi_k p_k(\mathbf{x}_n|\boldsymbol{\theta}_k)))$$

$$L(\theta) = \mathbb{E}[p(x, z|\boldsymbol{\theta})] = \sum_{n=1}^N \sum_{k=1}^K r_{nk} \left( \log \left( \pi_k \prod_{d=1}^D \mu_{kd}^{x_{nd}} (1 - \mu_{kd})^{1-x_{nd}} \right) \right)$$

# Exercise

- Multivariate Bernoulli distribution:

$$p_k(\mathbf{x}|\boldsymbol{\mu}) = \prod_{d=1}^D \mu_{kd}^{x_d} (1 - \mu_{kd})^{1-x_d}$$

- EM objective:

$$L(\theta) = \mathbb{E}[p(x, z|\boldsymbol{\theta})] = \sum_{n=1}^N \sum_{k=1}^K r_{nk} (\log (\pi_k p_k(\mathbf{x}_n|\boldsymbol{\theta}_k)))$$

$$L(\theta) = \mathbb{E}[p(x, z|\boldsymbol{\theta})] = \sum_{n=1}^N \sum_{k=1}^K r_{nk} \left( \log \left( \pi_k \prod_{d=1}^D \mu_{kd}^{x_{nd}} (1 - \mu_{kd})^{1-x_{nd}} \right) \right)$$

$$L(\theta) = \mathbb{E}[p(x, z|\boldsymbol{\theta})] = \sum_{n=1}^N \sum_{k=1}^K r_{nk} \left( \log (\pi_k) + \sum_{d=1}^D \log (\mu_{kd}^{x_{nd}}) + \log ((1 - \mu_{kd})^{1-x_{nd}}) \right)$$

# Exercise

- Multivariate Bernoulli distribution:

$$p_k(\mathbf{x}|\boldsymbol{\mu}) = \prod_{d=1}^D \mu_{kd}^{x_d} (1 - \mu_{kd})^{1-x_d}$$

- EM objective:

$$L(\theta) = \mathbb{E}[p(x, z|\boldsymbol{\theta})] = \sum_{n=1}^N \sum_{k=1}^K r_{nk} (\log (\pi_k p_k(\mathbf{x}_n|\boldsymbol{\theta}_k)))$$

$$L(\theta) = \mathbb{E}[p(x, z|\boldsymbol{\theta})] = \sum_{n=1}^N \sum_{k=1}^K r_{nk} \left( \log \left( \pi_k \prod_{d=1}^D \mu_{kd}^{x_{nd}} (1 - \mu_{kd})^{1-x_{nd}} \right) \right)$$

$$L(\theta) = \mathbb{E}[p(x, z|\boldsymbol{\theta})] = \sum_{n=1}^N \sum_{k=1}^K r_{nk} \left( \log (\pi_k) + \sum_{d=1}^D \log (\mu_{kd}^{x_{nd}}) + \log ((1 - \mu_{kd})^{1-x_{nd}}) \right)$$

$$L(\theta) = \mathbb{E}[p(x, z|\boldsymbol{\theta})] = \sum_{n=1}^N \sum_{k=1}^K r_{nk} \left( \log (\pi_k) + \sum_{d=1}^D x_{nd} \log (\mu_{kd}) + (1 - x_{nd}) \log ((1 - \mu_{kd})) \right)$$

# Exercise

- Question 3: Write down the M-step update

$$L(\theta) = \mathbb{E}[p(x, z|\boldsymbol{\theta})] = \sum_{n=1}^N \sum_{k=1}^K r_{nk} \left( \log(\pi_k) + \sum_{d=1}^D x_{nd} \log(\mu_{kd}) + (1 - x_{nd}) \log((1 - \mu_{kd})) \right)$$

- Differentiate wrt.:  $\boldsymbol{\theta} = (\pi_1, \dots, \pi_k, \boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_K)$

$$\frac{\partial L}{\partial \pi_j} = 0 \quad \pi_j = \frac{\sum_{n=1}^N r_{nj}}{\sum_{k=1}^K \sum_{n=1}^N r_{nk}}$$

# Exercise

- Question 3: Write down the M-step update

$$L(\theta) = \mathbb{E}[p(x, z|\boldsymbol{\theta})] = \sum_{n=1}^N \sum_{k=1}^K r_{nk} \left( \log(\pi_k) + \sum_{d=1}^D x_{nd} \log(\mu_{kd}) + (1 - x_{nd}) \log((1 - \mu_{kd})) \right)$$

- Differentiate wrt.:  $\boldsymbol{\theta} = (\pi_1, \dots, \pi_k, \boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_K)$

$$\frac{\partial L}{\partial \pi_j} = 0 \quad \pi_j = \frac{\sum_{n=1}^N r_{nj}}{\sum_{k=1}^K \sum_{n=1}^N r_{nk}}$$

$$\frac{\partial L}{\partial \mu_{kd}} = \sum_{n=1}^N r_{nk} \left( \frac{x_{nd}}{\mu_{kd}} + \frac{1 - x_{nd}}{1 - \mu_{kd}} \right) = 0$$

# Exercise

- Question 3: Write down the M-step update

$$L(\theta) = \mathbb{E}[p(x, z|\boldsymbol{\theta})] = \sum_{n=1}^N \sum_{k=1}^K r_{nk} \left( \log(\pi_k) + \sum_{d=1}^D x_{nd} \log(\mu_{kd}) + (1 - x_{nd}) \log((1 - \mu_{kd})) \right)$$

- Differentiate wrt.:  $\boldsymbol{\theta} = (\pi_1, \dots, \pi_k, \boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_K)$

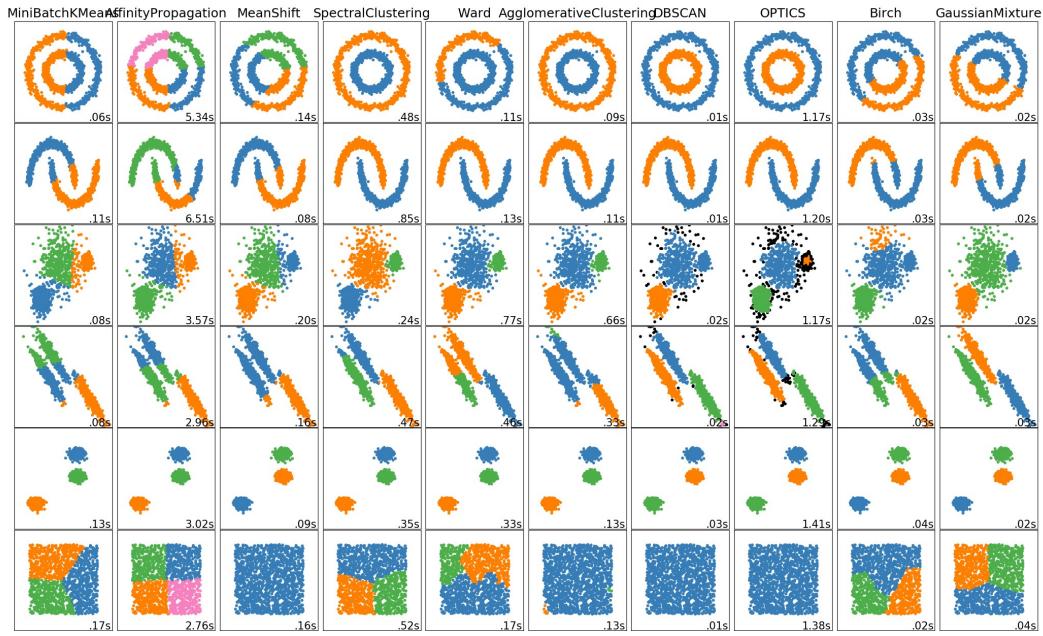
$$\frac{\partial L}{\partial \pi_j} = 0 \quad \pi_j = \frac{\sum_{n=1}^N r_{nj}}{\sum_{k=1}^K \sum_{n=1}^N r_{nk}}$$

$$\frac{\partial L}{\partial \mu_{kd}} = \sum_{n=1}^N r_{nk} \left( \frac{x_{nd}}{\mu_{kd}} - \frac{1 - x_{nd}}{1 - \mu_{kd}} \right) = 0 \quad \mu_{kd} = \frac{\sum_{n=1}^N r_{nk} x_{nd}}{\sum_{n=1}^N r_{nk}}$$

# Summary

- EM algorithm is useful for fitting GMMs (or other mixtures) in an unsupervised setting
- Can be used for:
  - Clustering
  - Classification
  - Distribution estimation
  - Outlier detection

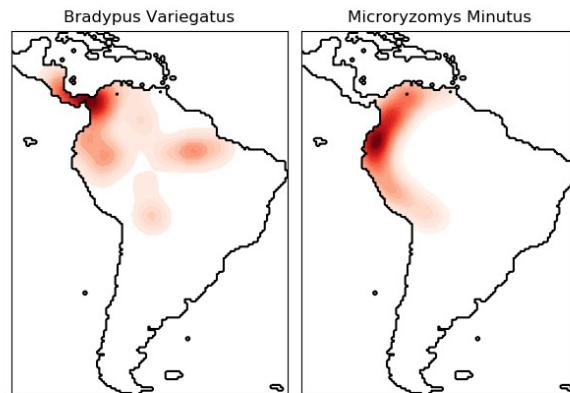
# Other unsupervised clustering techniques



Source: <https://scikit-learn.org/stable/modules/clustering.html>

# Alternative for density estimation

- Kernel density estimation



Source: [https://scikit-learn.org/stable/auto\\_examples/neighbors/plot\\_species\\_kde.html](https://scikit-learn.org/stable/auto_examples/neighbors/plot_species_kde.html)

# References

- Lecture slides/videos
- [https://www.python-course.eu/expectation maximization and gaussian mixture models.php](https://www.python-course.eu/expectation_maximization_and_gaussian_mixture_models.php)
- <https://scikit-learn.org/stable/modules/clustering.html>