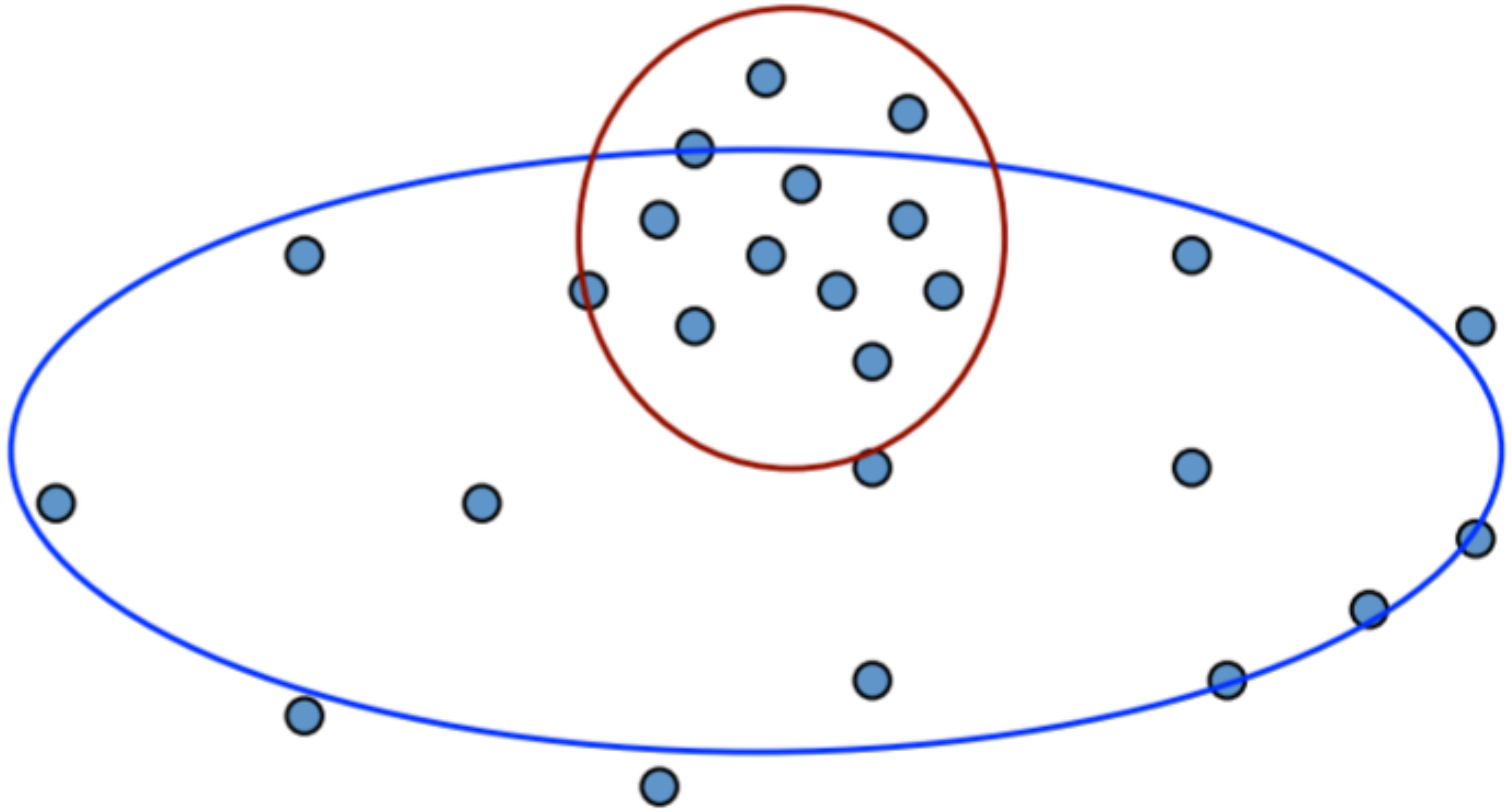
Gaussian Mixture Models

Recap

- Unsupervised learning

- **K-means clustering**
  - Each cluster is represented by the centroid
  - Data belongs to a cluster with nearest centroid

- **Limitations**
  - Brittle to initialization
  - Overlapping clusters
  - Wider clusters



Today

- **Mixture Model**
  - Tackle clusters with overlap & various sizes
  - Will take a generative approach

  - Focus on the most famous case
    - Gaussian mixture models (GMM)

Mixture Model

Mixture Model

- Take a generative approach
  - Posit that data are coming from some well-defined distribution
  - Fit the parameters of the distribution

- Have done this for naïve Bayes
  - **Difference.** Do not observe the “labels”

Mixture Model

- **Solution.** Introduce latent variables of cluster identity
  - Not necessarily reflecting reality — rather an instrument

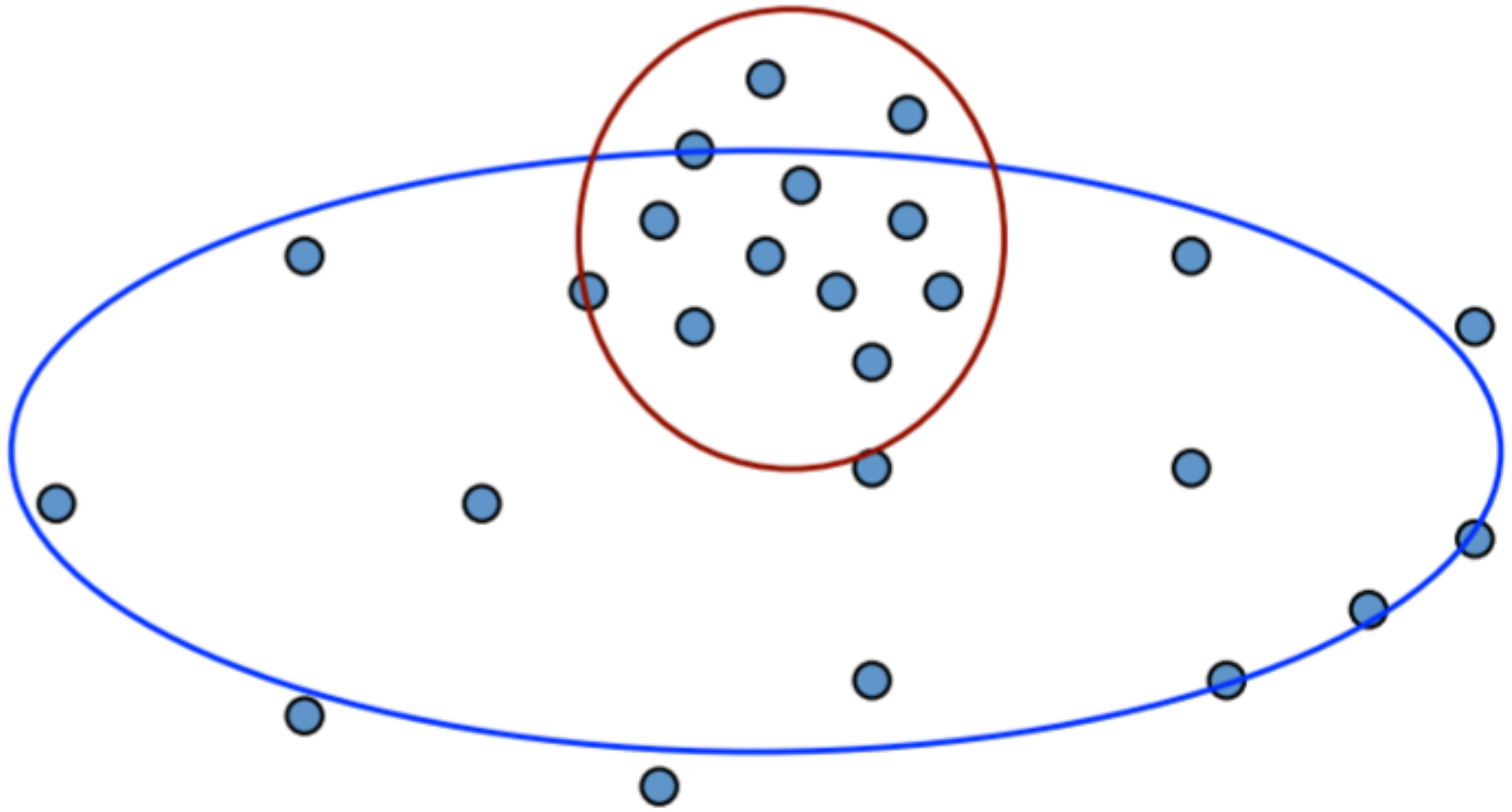
- **Modeling.** We consider:
  - $P_\phi(\text{cluster})$: Latent group identity
  - $P_\theta(\text{feature} \mid \text{cluster})$ Data distribution of each cluster

- **Fitting.** Use training data to fit the parameters

$$P_{\text{train}} \approx P_{\theta,\phi}(\text{feature})$$

Mixture Model

- **Example.** Suppose the case of two clusters
  - Draw $Y \in \{0,1\} \sim \mathrm{Bern}(p)$
    - If $Y = 0$, then $X \sim \mathcal{N}(\mu_0, \sigma_0^2)$
    - If $Y = 1$, then $X \sim \mathcal{N}(\mu_1, \sigma_1^2)$
  - Allows overlap and varying widths



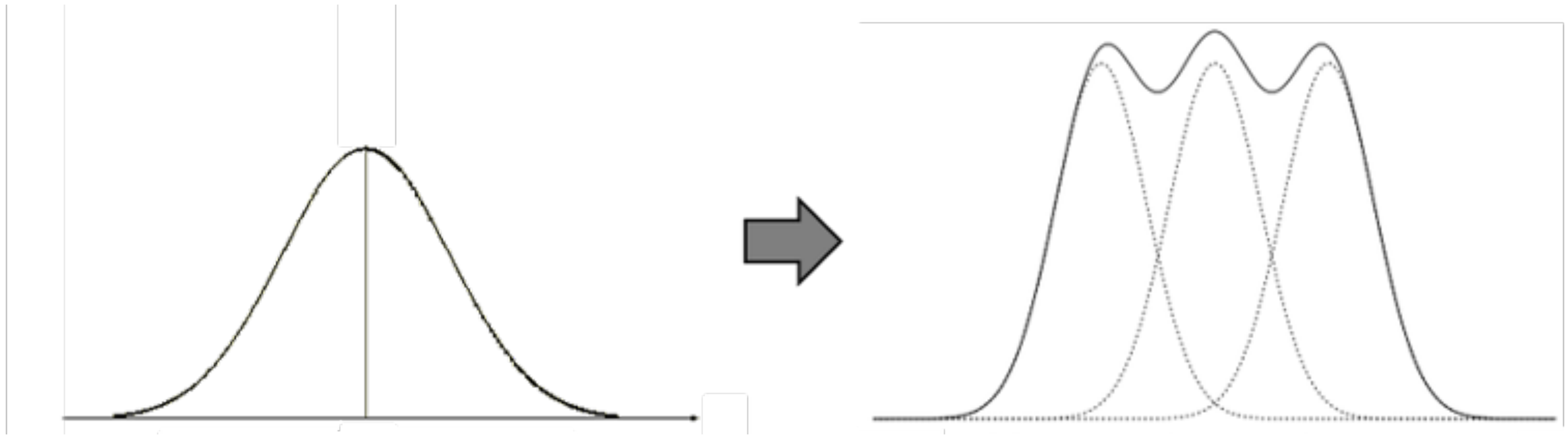
Generative approach

- **Perk.** If you have learned a nice probabilistic model from the data you can sample a new data from this $P_{\theta,\phi}(\,\cdot\,)$

(Finite) Mixture Models

- A set of generative models where $P(\cdot)$ takes the form of a weighted sum of finite elementary distributions

$$p(\mathbf{x}) = \sum_{k=1}^{K} \pi_k \cdot p_k(\mathbf{x}), \qquad \pi_k \in [0,1], \sum \pi_k = 1$$



Gaussian Mixture Models

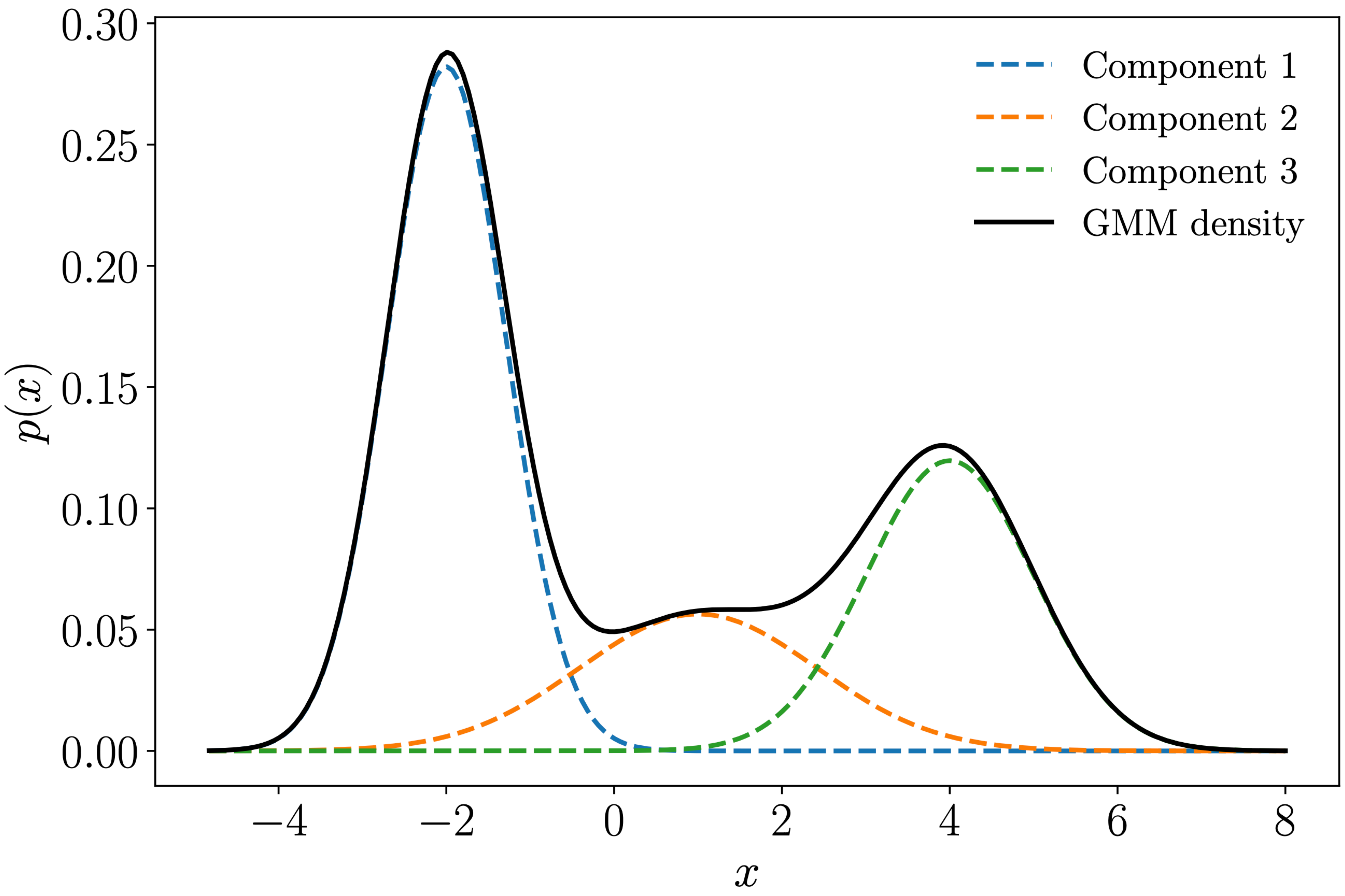
- **Gaussian MM.** Each base distribution is a Gaussian distribution

$$p(\mathbf{x} \mid \theta) = \sum_{k=1}^{K} \pi_k \cdot \mathcal{N}(\mathbf{x} \mid \mu_k, \Sigma_k)$$

  - Here, θ is the total parameter set

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

Gaussian Mixture Models



$$p(x \mid \boldsymbol{\theta}) = 0.5\mathcal{N}\left(x \mid -2, \tfrac{1}{2}\right) + 0.2\mathcal{N}\left(x \mid 1, 2\right) + 0.3\mathcal{N}\left(x \mid 4, 1\right)$$

Optimizing GMMs

- As in naïve Bayes, our optimization objective comes from the maximum likelihood principle
  - The likelihood of mixture distribution can be written as:

$$p(\mathbf{x}_{1:n} \,|\, \theta) = \prod_{i=1}^{n} p(\mathbf{x}_i \,|\, \theta)$$

$$= \prod_{i=1}^{n} \sum_{k=1}^{K} \pi_{k(i)} \cdot \mathcal{N}(\mathbf{x}_i \,|\, \mu_{k(i)}, \Sigma_{k(i)})$$

- **Goal.** Maximize this quantity by selecting $\theta = \{\mu_k, \Sigma_k, \pi_k \mid k \in [K]\}$

Optimizing GMMs

- Again, consider the log-likelihood to make it a summation:

$$\mathscr{L}(\theta) := \log p(\mathbf{x}_{1:n} \,|\, \theta) = \sum_{i=1}^{n} \log \left(\sum_{k=1}^{K} \pi_k \cdot \mathcal{N}(\mathbf{x}_i \,|\, \mu_k, \Sigma_k) \right)$$

  - We want to solve the maximization

$$\max_{\theta} \mathscr{L}(\theta)$$

- **Problem.** Very difficult to optimize by the critical point analysis
  - We'll go through what we call expectation-maximization

Expectation-Maximization (Advanced!)

Expectation-Maximization

- An iterative algorithm for optimizing probabilistic latent-variable models
  - Can be thought of as a specialized form of alternating optimization

- **Idea.** Repeat the following steps
  - Construct a lower bound on the likelihood

$$g(\theta) \leq \mathscr{L}(\theta)$$

  - Maximizes the lower bound $g(\theta)$

$$\theta^{(\text{new})} = \arg\max_{\theta} g(\theta)$$

Expectation-Maximization

- Formally, let y_i be the latent variable associate with $\mathbf{x}_i$
  - In GMM, y_i is the "cluster identity," i.e., which Gaussian $\mathbf{x}_i$ is from

- Then, we know that:

$$
\begin{aligned}
\mathcal{L}(\theta) &:= \sum_{i=1}^{n} \log p(\mathbf{x}_i \,|\, \theta) \\
&= \sum_{i=1}^{n} \log \left(\sum_{k=1}^{K} p(\mathbf{x}_i, y_i = k \,|\, \theta) \right)
\end{aligned}
$$

Expectation-Maximization

- Define any distribution $Q(k)$
- Then, we have, for any single sample-group pair $(\mathbf{x}, y)$:

$$\begin{aligned}\log p(\mathbf{x} \mid \theta) &= \log\left(\sum_{k=1}^{K} p(\mathbf{x}, y = k \mid \theta)\right) \\ &= \log\left(\sum_{k=1}^{K} Q(k) \cdot \frac{p(\mathbf{x}, y = k \mid \theta)}{Q(k)}\right) \\ &\geq \sum_{k=1}^{K} Q(k) \cdot \log\left(\frac{p(\mathbf{x}, y = k \mid \theta)}{Q(k)}\right)\end{aligned}$$

  - The inequality is due to Jensen's inequality

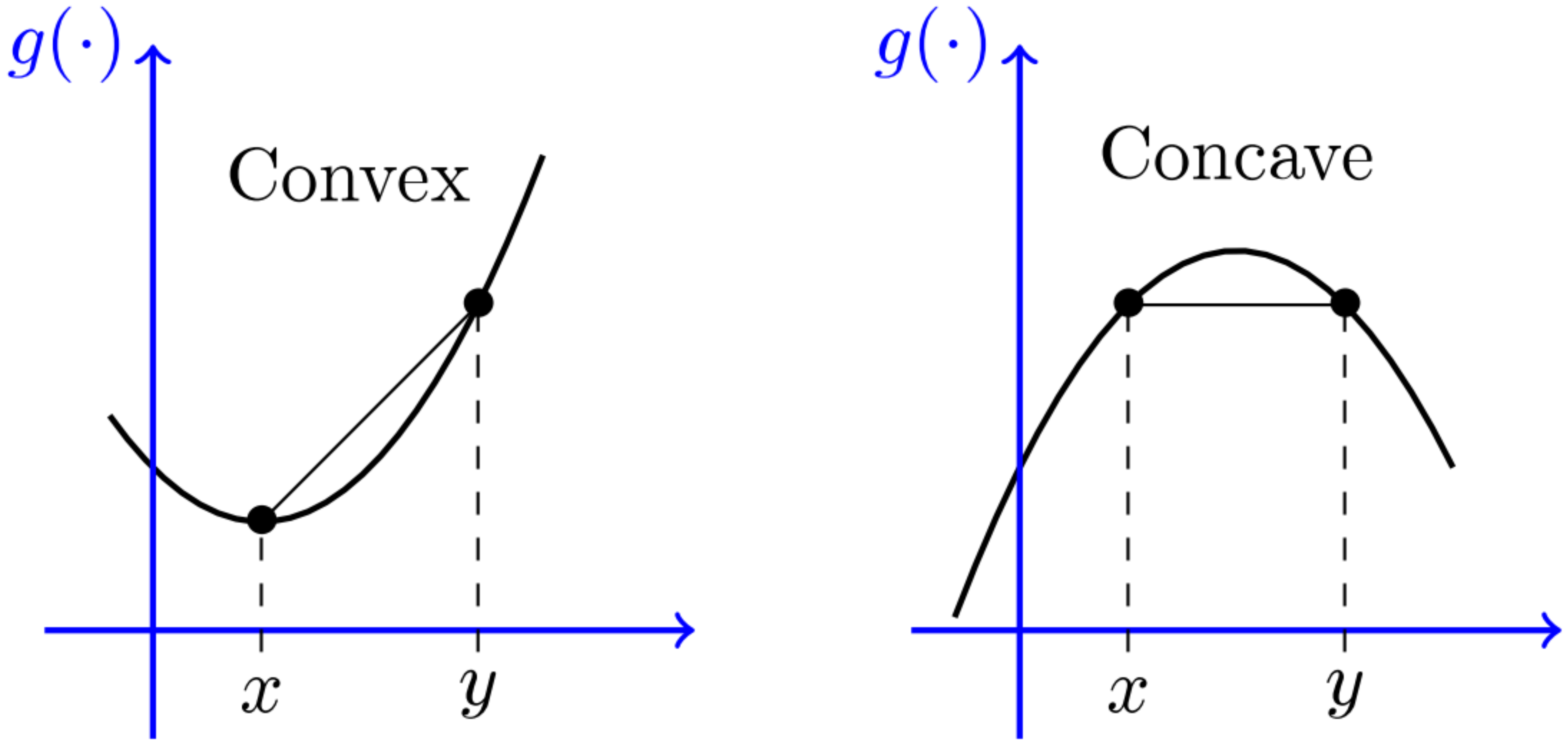
Jensen's inequality (Advanced!)

Convex functions

- Recall that convex functions are functions such that:

$$\lambda f(x) + (1-\lambda) f(y) \geq f(\lambda x + (1-\lambda) y), \qquad \forall x, y, \forall \lambda \in [0,1]$$

- Concave functions are the opposite (negative of convex functions)
  - Example. Log function



Jensen's inequality

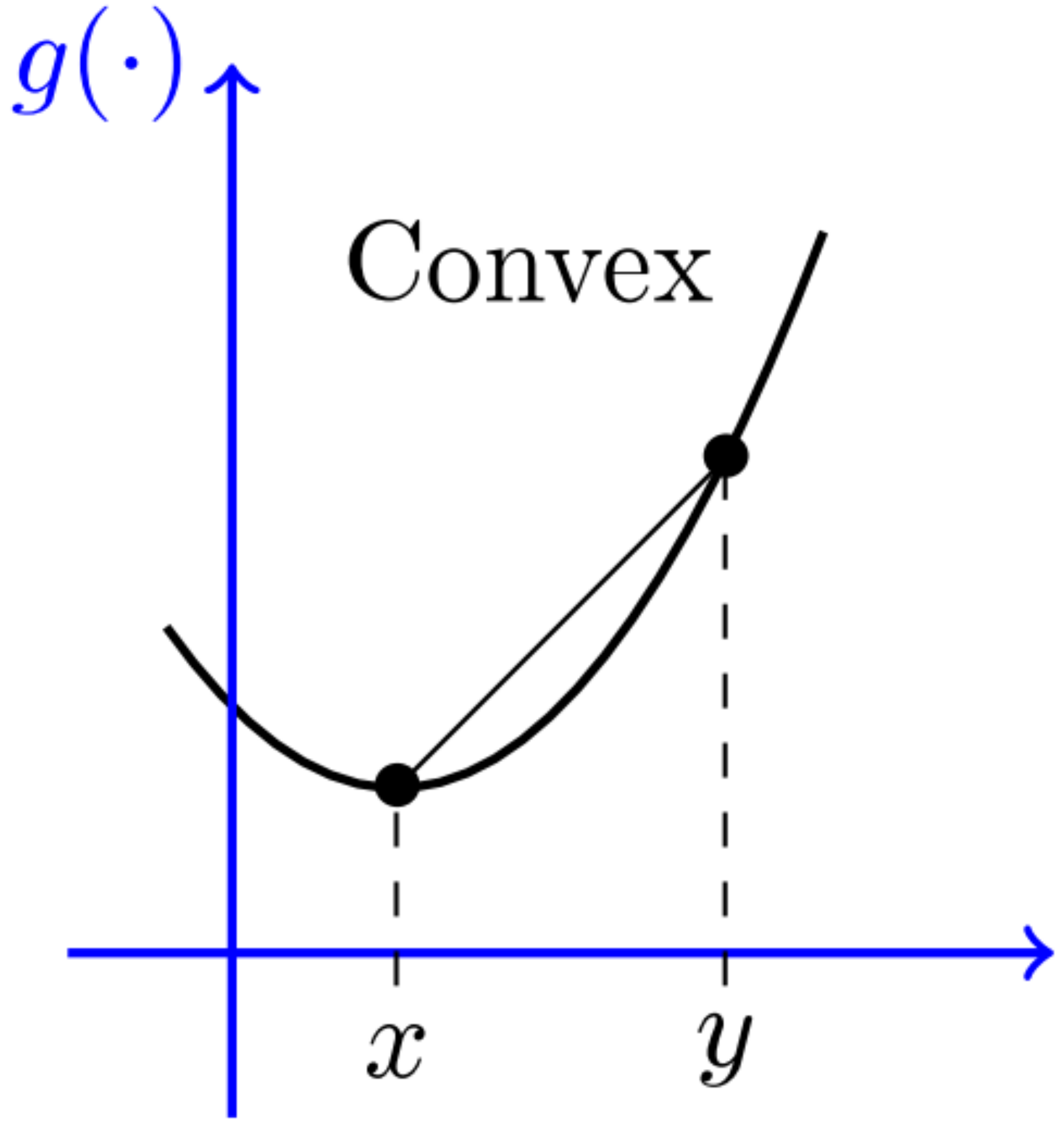
- For convex functions, we have

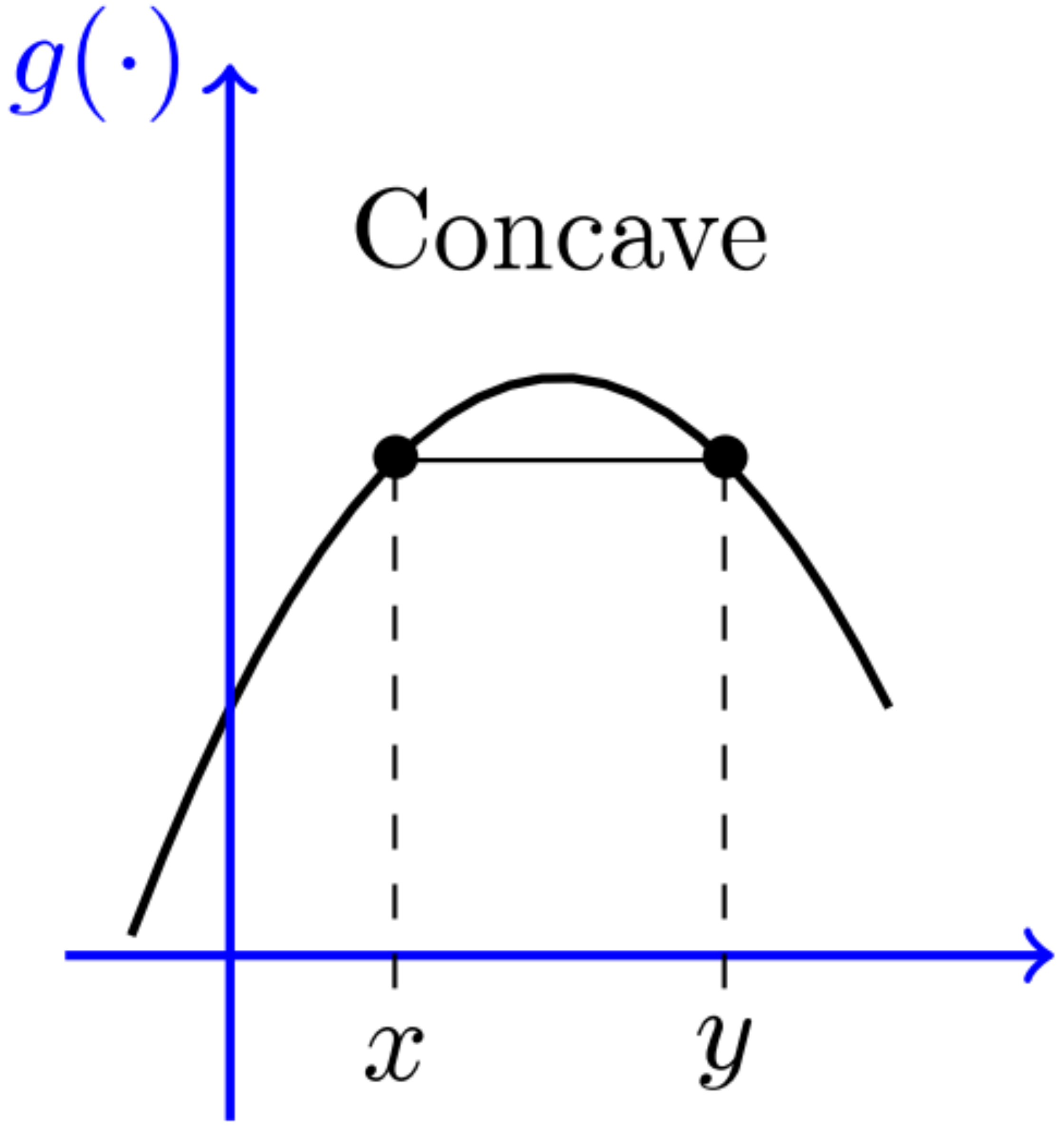
$$\mathbb{E}[f(X)] \leq f(\mathbb{E}[X])$$

- For concave functions, we have

$$\mathbb{E}[f(X)] \geq f(\mathbb{E}[X])$$

- Equality, if X is a constant variable





</Jensen's inequality>

Expectation-Maximization

$$\log\left(\sum_{k=1}^{K} Q(k) \cdot \frac{p(\mathbf{x}, y = k \mid \theta)}{Q(k)}\right) \geq \sum_{k=1}^{K} Q(k) \cdot \log\left(\frac{p(\mathbf{x}, y = k \mid \theta)}{Q(k)}\right)$$

- This is applying Jensen's inequality to a concave function $\log(\cdot)$
  - Here, the random variable is:

$$\frac{p(\mathbf{x}, y = k \mid \theta)}{Q(k)}$$

- This lower bound on the likelihood is called evidence lower bound (ELBO)

$$\mathrm{ELBO}(\mathbf{x} \mid Q, \theta)$$

Expectation-Maximization

$$\log p(\mathbf{x} \mid \theta) \geq \text{ELBO}(\mathbf{x} \mid Q, \theta)$$

- Now, we want to make this bound tightest by selecting good Q
  - Recall that Jensen's inequality is tightest for constant R.V.
    - That is,

$$\text{const} = \frac{p(\mathbf{x}, y = k \mid \theta)}{Q(k)} = \frac{p(y = k \mid \mathbf{x}, \theta)}{Q(k)} p(\mathbf{x} \mid \theta)$$

  - Thus, best if we choose

$$Q(k) = p(y = k \mid \mathbf{x}, \theta)$$

Expectation-Maximization

- Let's go back to the multi-sample case:

- We have

$$\mathscr{L}(\theta) = \sum_{i=1}^{n} \log\left(\sum_{k=1}^{K} p(\mathbf{x}_i, y_i = k \,|\, \theta) \right) \geq \sum_{i=1}^{n} \mathrm{ELBO}(\mathbf{x}_i \,|\, Q_i, \theta)$$

  - Here, we have Q_i as samplewise posteriors

$$Q_i(k) = p(y_i = k \,|\, \mathbf{x}_i, \theta)$$

EM Algorithm

- Now, the EM algorithm can be written as:

  - **1. Initialization:** Initialize θ
  - **2. Expectation:** Compute the ELBO-maximizing Q

$$Q_i(k) = p(y_i = k \,|\, \mathbf{x}_i, \theta)$$

  - **3. Maximization:** Compute the ELBO-maximizing θ

$$\theta^{(\text{new})} = \arg\max_{\theta} \sum_{i=1}^{n} \text{ELBO}(\mathbf{x}_i \,|\, Q_i, \theta)$$

  - **4. Repeat!**

</Expectation-Maximization>

EM for GMMs

- Now, let's apply EM for GMMs

- First, recall that:
  - Multivariate Gaussians

$$\mathcal{N}(\mathbf{x} \mid \mu, \Sigma) = \frac{1}{\sqrt{(2\pi)^d |\Sigma|}} \cdot \exp\left(-\frac{1}{2}(\mathbf{x} - \mu)^\top \Sigma^{-1} (\mathbf{x} - \mu)\right)$$

  - Taking log, we get

$$\log \mathcal{N}(\mathbf{x} \mid \mu, \Sigma) = -\frac{1}{2}\left(d \log(2\pi) + \log|\Sigma| + (\mathbf{x} - \mu)^\top \Sigma^{-1} (\mathbf{x} - \mu)\right)$$

EM for GMMs

- **Expectation.** This step computes the posterior for each sample

$$Q(k) = p(y_i = k \mid \mathbf{x}_i, \theta)$$

  - In clustering, we call this responsibility

$$r_{ik} = p(y_i = k \mid \mathbf{x}_i, \theta) = \frac{p(\mathbf{x}_i, y_i = k \mid \theta)}{p(\mathbf{x}_i \mid \theta)}$$

$$p(y_i = k \mid \theta) = \quad = \frac{\pi_k \mathcal{N}(\mathbf{x}_i \mid \mu_k, \Sigma_k)}{\sum_j \pi_j \mathcal{N}(\mathbf{x}_i \mid \mu_j, \Sigma_j)}$$

$$= p(\mathbf{x}_i \mid y_i = k, \theta)$$

$$= p(\mathbf{x}_i \mid \theta)$$

EM for GMMs

$$r_{ik} = \frac{\pi_k \mathcal{N}(\mathbf{x}_i \mid \mu_k, \Sigma_k)}{\sum_j \pi_j \mathcal{N}(\mathbf{x}_i \mid \mu_j, \Sigma_j)}$$

- Note. If we plug in:
  - uniform prior $\pi_k = 1/K$
  - uniform variance $\sigma_k = 1/\beta$

  then we recover the soft K-means objective

$$r_{ik} = \frac{\exp(-\beta \|\mathbf{x}_i - \mu_k\|_2^2)}{\sum_j \exp(-\beta \|\mathbf{x}_i - \mu_j\|_2^2)}$$

EM for GMMs

- **Maximization.** Given the r_{ik} fixed, we solve the maximization

$$\max_{\theta} \sum_{i=1}^{n} \text{ELBO}(\mathbf{x}_i \,|\, Q_i, \theta)$$

  - Recall that the ELBO was:

$$\sum_{i=1}^{n} \sum_{k=1}^{K} r_{ik} \cdot \log\left(\frac{p(\mathbf{x}_i, y_i = k \,|\, \theta)}{r_{ik}} \right)$$

  - Dropping constants, we are solving:

$$\max_{\theta} \sum_{i=1}^{n} \sum_{k=1}^{K} r_{ik} \cdot (\log p(\mathbf{x}_i \,|\, y_i = k, \theta) + \log p(y_i = k \,|\, \theta))$$

EM for GMMs

$$\max_{\theta} \sum_{i=1}^{n} \sum_{k=1}^{K} r_{ik} \cdot (\log p(\mathbf{x}_i \,|\, y_i = k, \theta) + \log p(y_i = k \,|\, \theta))$$

- We can divide into two subproblems:

$$\max_{\{\pi_k\}} \sum_{i=1}^{n} \sum_{k=1}^{K} r_{ik} \cdot \log \pi_k$$

$$\max_{\{\mu\},\{\Sigma\}} \sum_{i=1}^{n} \sum_{k=1}^{K} r_{ik} \cdot \log \mathcal{N}(\mathbf{x}_i \,|\, \mu_k, \Sigma_k)$$

EM for GMMs

$$\max_{\{\pi_k\}} \sum_{i=1}^{n} \sum_{k=1}^{K} r_{ik} \cdot \log \pi_k$$

- 1st subproblem. Constrained optimization problem
  - Solve this by the method of Lagrangian multipliers, to get

$$\pi_k = \frac{n_k}{n}$$

  - Here, we use the shorthand n_k as the total responsibility in cluster k

$$n_k = \sum_{i=1}^{n} r_{ik}$$

EM for GMMs

$$\max_{\{\mu\},\{\Sigma\}} \sum_{i=1}^{n} \sum_{k=1}^{K} r_{ik} \cdot \log \mathcal{N}(\mathbf{x}_i \,|\, \mu_k, \Sigma_k)$$

- 2nd subproblem. Unconstrained maximization
  - Analyze the critical point, to get:

$$\mu_k = \frac{\sum_i r_{ik}\mathbf{x}_i}{n_k}, \qquad \Sigma_k = \frac{1}{n_k} \sum_{i=1}^{n} r_{ik}(\mathbf{x}_i - \mu_k)(\mathbf{x}_i - \mu_k)^\top$$

  - For a full derivation, see section 11.2.3 of the MML textbook

1. Initialize $\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \pi_k$.
2. *E-step:* Evaluate responsibilities r_{nk} for every data point $\boldsymbol{x}_n$ using current parameters $\pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k$:

$$r_{nk} = \frac{\pi_k \mathcal{N}(\boldsymbol{x}_n \mid \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_j \pi_j \mathcal{N}(\boldsymbol{x}_n \mid \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)} . \tag{11.53}$$

3. *M-step:* Reestimate parameters $\pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k$ using the current responsibilities r_{nk} (from E-step):

$$\boldsymbol{\mu}_k = \frac{1}{N_k} \sum_{n=1}^{N} r_{nk} \boldsymbol{x}_n , \tag{11.54}$$

$$\boldsymbol{\Sigma}_k = \frac{1}{N_k} \sum_{n=1}^{N} r_{nk} (\boldsymbol{x}_n - \boldsymbol{\mu}_k)(\boldsymbol{x}_n - \boldsymbol{\mu}_k)^\top , \tag{11.55}$$

$$\pi_k = \frac{N_k}{N} . \tag{11.56}$$

Original Data
k-Means Clustering
EM Clustering
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1

Next up

- Dimensionality reduction

</lecture 6>