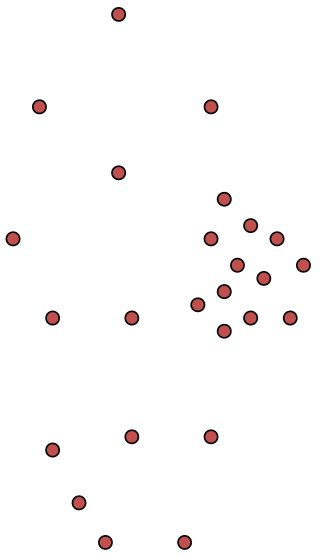
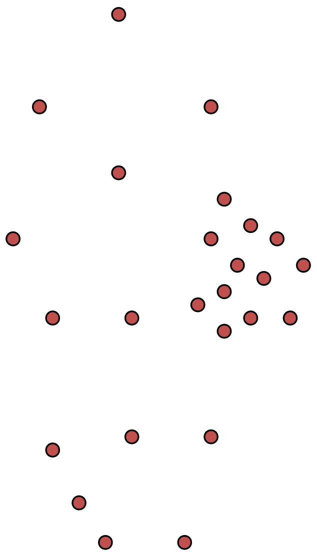
# (One) bad case for K-means



- Clusters may overlap
- Some clusters may be “wider” than others
- Clusters may not be linearly separable

# (One) bad case for K-means



- Clusters may overlap
- Some clusters may be “wider” than others
- Clusters may not be linearly separable

# Partitioning Algorithms

- K-means
  - **hard assignment**: each object belongs to only one cluster

- Mixture modeling
  - **soft assignment**: probability that an object belongs to a cluster

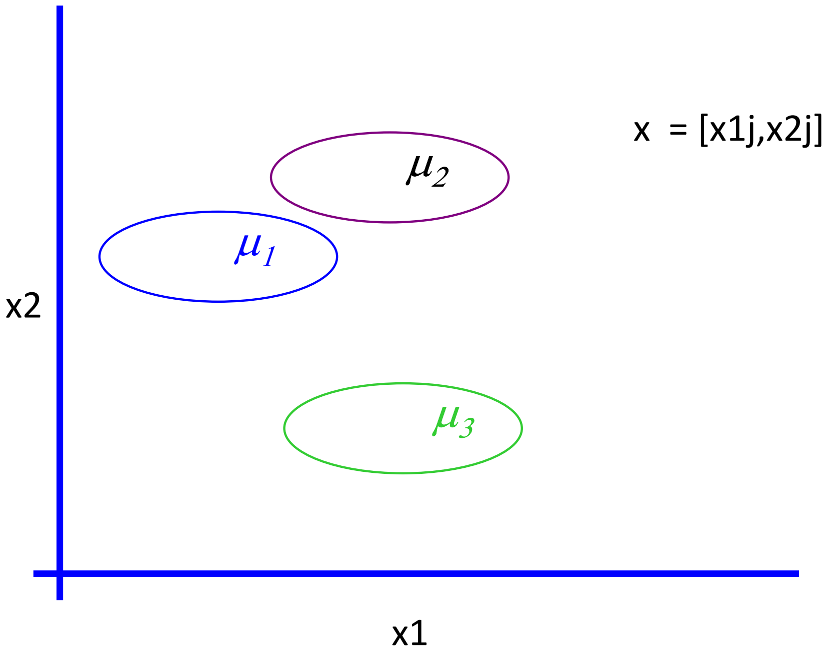
*Generative approach*

# Gaussian Mixture Model

Mixture of K Gaussian distributions: (Multi-modal distribution)

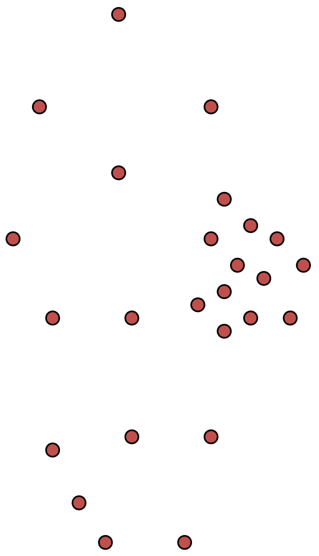
$$p(x|y=i) \sim \mathrm{N}(\mu_i, \sigma^2 \boldsymbol{I})$$

$$p(x) = \sum_i \underset{\substack{\downarrow \\ \textbf{Mixture} \\ \textbf{component}}}{p(x|y=i)} \; \underset{\substack{\downarrow \\ \textbf{Mixture} \\ \textbf{proportion}}}{P(y=i)}$$



x = [x1j,x2j]

# (One) bad case for K-means



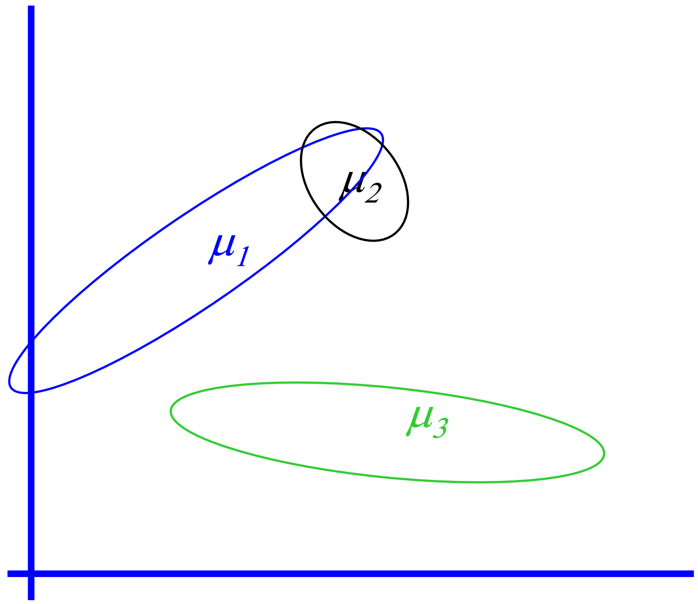
- Clusters may overlap
- Some clusters may be “wider” than others
- Clusters may not be linearly separable

# General GMM

GMM – Gaussian Mixture Model (Multi-modal distribution)

$$p(x|y=i) \sim \mathrm{N}(\mu_i, \Sigma_i)$$

$$p(x) = \sum_i \underbrace{p(x|y=i)}_{\text{Mixture component}} \underbrace{P(y=i)}_{\text{Mixture proportion}}$$



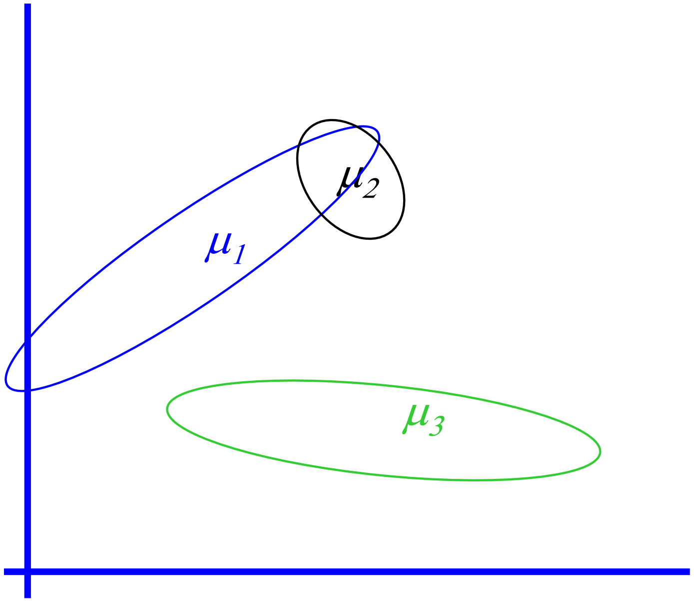
# General GMM

GMM – Gaussian Mixture Model (Multi-modal distribution)

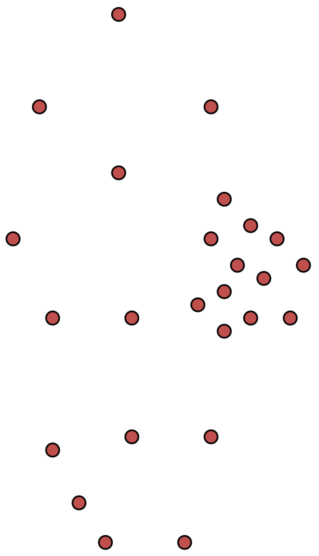
- There are k components
- Component *i* has an associated mean vector $\mu_i$
- Each component generates data from a Gaussian with mean $\mu_i$ and covariance matrix $\Sigma_i$

Each data point is generated according to the following recipe:

1) Pick a component at random: Choose component i with probability *P(y=i)*
2) Datapoint x $\sim N(\mu_i, \Sigma_i)$



# (One) bad case for K-means



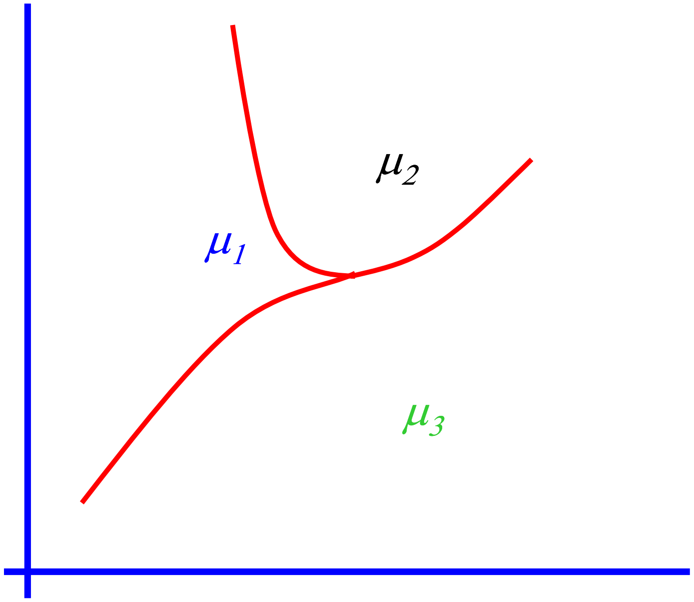
- Clusters may overlap
- Some clusters may be “wider” than others
- Clusters may not be linearly separable

# General GMM

GMM – Gaussian Mixture Model (Multi-modal distribution)

$$p(x|y=i) \sim \mathrm{N}(\mu_i, \Sigma_i)$$

Decision boundary when probabilities are equal:



$$\log \frac{\mathrm{P}(y = i \mid x)}{\mathrm{P}(y = j \mid x)}$$

$$= \log \frac{p(x \mid y = i)\mathrm{P}(y = i)}{p(x \mid y = j)\mathrm{P}(y = j)}$$

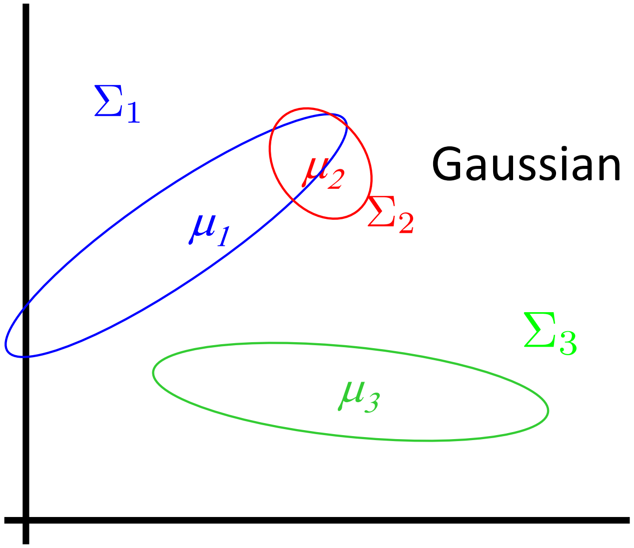
$$= x^T \enclose{circle}{W} x + \enclose{circle}{w}^T x$$

→ Depend on $\mu_1, \mu_2, .. , \mu_K, \Sigma_1, \Sigma_2, .. , \Sigma_K$, P(y=1),…, P(Y=k)

**"Quadratic Decision boundary"** – second-order terms don't cancel out

# Learning General GMM

$$x_1, \ldots, x_m \sim p(x) = \sum_{i=1}^{k} \underset{\substack{\downarrow \\ \textbf{Mixture} \\ \textbf{component}}}{p(x|Y = i)} \underset{\substack{\downarrow \\ \textbf{Mixture} \\ \textbf{proportion, } p_i}}{P(Y = i)}$$



Gaussian mixture model

$$p(x|Y = i) \sim \mathcal{N}(\mu_i, \Sigma_i)$$

**Parameters:** $\{p_i, \mu_i, \Sigma_i\}_{i=1}^{K}$

- How to estimate parameters? Max Likelihood
  But don’t know labels Y

# Learning General GMM

Maximize marginal likelihood:

$$\arg\max \prod_j P(x_j) = \arg\max \prod_j \sum_{i=1}^{K} P(y_j=i,x_j)$$

$$= \arg\max \prod_j \sum_{i=1}^{K} P(y_j=i)p(x_j|y_j=i)$$

$P(y_j=i) = P(y=i)$ Mixture component i is chosen with prob $P(y = i)$

$$= \arg\max \prod_{j=1}^{m} \sum_{i=1}^{k} P(y=i) \frac{1}{\sqrt{\det(\Sigma_i)}} \exp\left[ -\frac{1}{2}(x_j-\mu_i)^T \Sigma_i (x_j-\mu_i) \right]$$

How do we find the $\mu_i$, $\Sigma_i$ s and P(y=i)s which give max. marginal likelihood?

* Set $\frac{\partial}{\partial \mu_i}$ log Prob (….) = 0 and solve for $\mu_i$'s. Non-linear not-analytically solvable

* Use gradient descent: Doable, but often slow