Mixtures of Gaussians

Uses: - Clustering

- Model non-Gaussian distributions

Model:

Same as Gaussian Bayes classifier

Except "lables" are hidden or "latent" variables

Cluster choice $z^{(n)} \sim \text{Discrete}(\underline{\pi})$

$z^{(n)} \in \{1, 2, 3 \ldots K\}$

$\underline{\pi}$: +ve vector, sums to 1

Features

$$\underline{x}^{(n)} \mid z^{(n)} = k \sim N(\underline{x}^{(n)}; \underline{\mu}^{(k)}, \Sigma^{(k)})$$

Likelihood

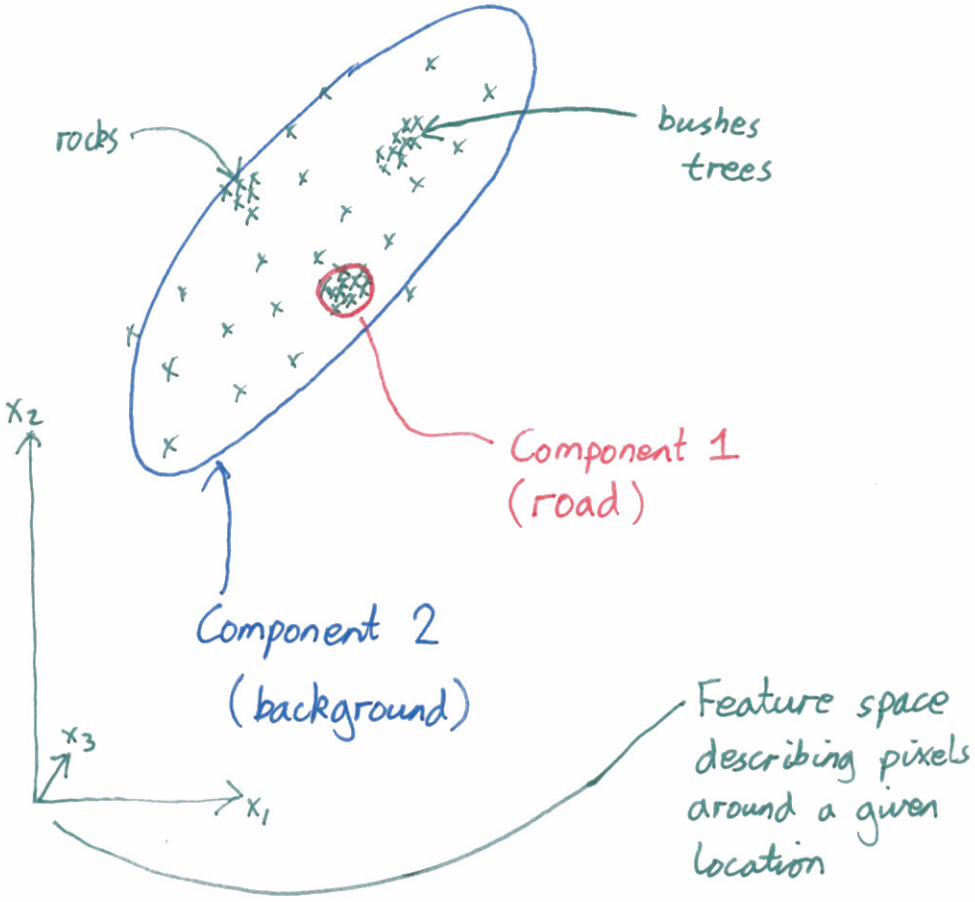
Given one observation: $p(\underline{x}^{(n)} \mid \theta = (\underline{\pi}, \{\underline{\mu}^{(k)}, \Sigma^{(k)}\}))$

$$= \sum_{k=1}^{K} p(\underline{x}^{(n)}, z^{(n)} = k \mid \theta)$$

$$= \sum_k p(\underline{x}^{(n)} \mid z^{(n)} = k, \theta)\, p(z^{(n)} = k \mid \theta)$$

$$= \sum_k N(\underline{x}^{(n)}; \underline{\mu}^{(k)}, \Sigma^{(k)})\, \pi_k$$

Use in "Stanley" car



Gradient-based fitting

$$\text{Cost} = -\sum_{n=1}^{N} \log(p(\underline{x}^{(n)} \mid \theta))$$

Write as function of unconstrained parameters

$\tilde{L}^{(k)}$ lower-triangular matrix

$$L^{(k)} = \begin{cases} L_{ij} = e^{\tilde{L}_{ii}} & i=j \\ L_{ij} = \tilde{L}_{ij} & i>j \\ L_{ij} = 0 & i<j \end{cases}$$

$$\Sigma^{(k)} = L^{(k)} L^{(k)\top}$$

$k = 1 \ldots K$

$\underline{c}$ vector K params

$\underline{\pi} = \text{softmax}(\underline{c})$

neg. log likelihood

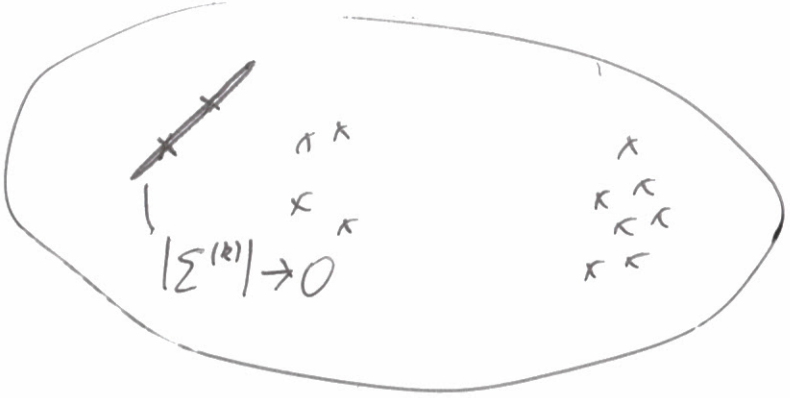
Initialization matters

$$\underline{M}^{(k)} = \underline{0}$$

$$\underline{\Sigma}^{(k)} = \mathbb{I}$$

$$\underline{M}^{(k)} \leftarrow \underline{M}^{(k)} - \eta \nabla_{\underline{W}} NLL$$

Maximum Likelihood is ∞



The alternative EM

Idea: Pretend we observe $\{z^{(n)}\}$

Responsibility initialization:

$$r_k^{(n)} = \begin{cases} 1 & \text{if } z^{(n)} = k \\ 0 & \text{otherwise} \end{cases}$$

Maximum likelihood parameters:

$$\pi_k = \frac{r_k}{N} \qquad r_k = \sum_{n=1}^{N} r_k^{(n)}$$

$$\underline{\mu}^{(k)} = \frac{1}{r_k} \sum_{n=1}^{N} r_k^{(n)} \underline{x}^{(n)}$$

$$\Sigma^{(k)} = \frac{1}{r_k} \sum_{n} r_k^{(n)} \underline{x}^{(n)} \underline{x}^{(n)T} - \underline{\mu}^{(k)} \underline{\mu}^{(k)T}$$

EM Algorithm

0) Initialize params θ

1) E-step

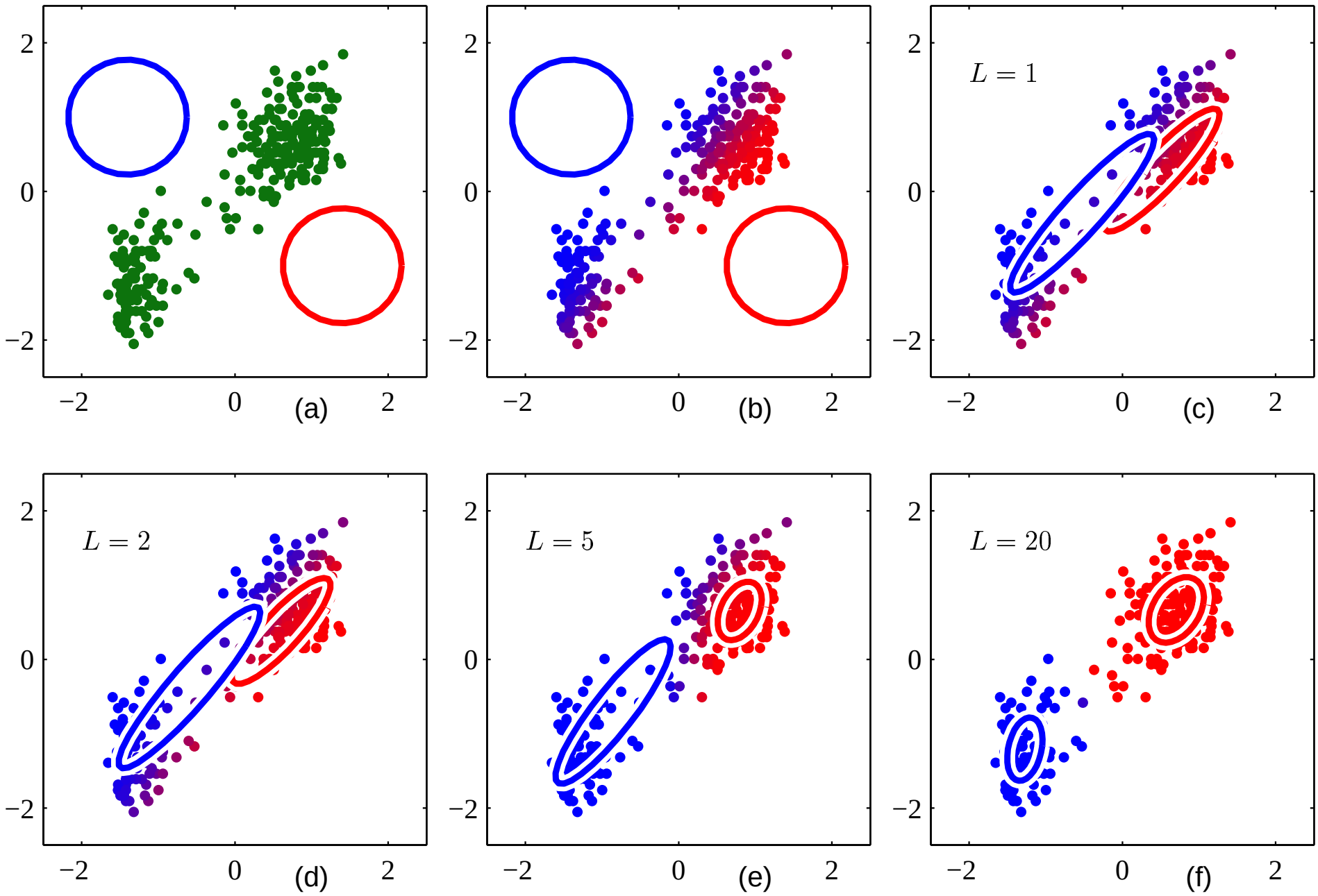
$$r_k^{(n)} = p(z^{(n)} = k \mid \underline{x}^{(n)}, \theta)$$

2) M-step

Uses these real-valued $r_k^{(n)}$ in those equations to fit θ.

3) GoTo 1) if not converged.

EM algorithm for Gaussian mixtures



Bishop Figure 9.8, or see Murphy p353

Interpretation

$$r_k^{(n)} = Q(z^{(n)}=k) = P(z^{(n)}=k \mid \underline{x}^{(n)}, \theta^{(old)})$$

Compare distributions

$$D_{KL}(Q \| P) = \sum_{\underline{z}} Q(\underline{z}) \log \frac{Q(\underline{z})}{P(\underline{z} \mid X, \theta)} \geq 0$$

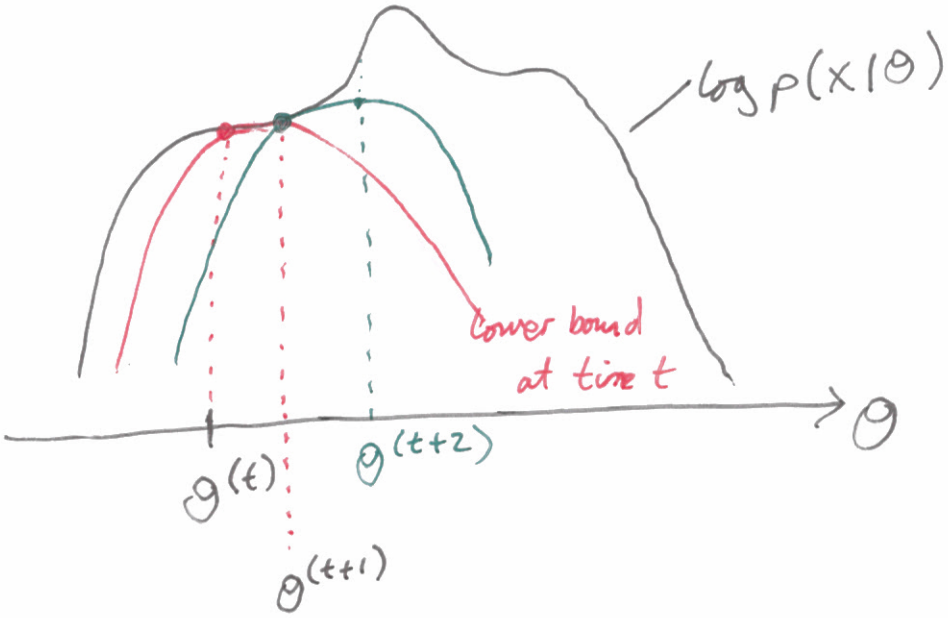
$$\left[P(\underline{z} \mid X, \theta) = \frac{P(\underline{z}, X \mid \theta)}{P(X \mid \theta)} \leftarrow \text{likelihood} \right]$$

$$\Rightarrow \sum_{\underline{z}} Q(\underline{z}) \log \frac{Q(\underline{z})}{P(\underline{z}, X \mid \theta)} \geq -\log P(X \mid \theta)$$

Bound on log-likelihood

Tight if $\theta = \theta^{(old)}$

Bound-based optimizer



If loose bound: