Machine Learning

Lecture 9 - Unsupervised learning, mixture models and clustering

Mattias Villani

Department of Statistics
Stockholm University

Department of Computer and Information Science
Linköping University

 mattiasvillani.com  @matvil  mattiasvillani

Lecture overview

- Supervised Mixture-of-Normals
- Unsupervised Mixture-of-Normals
- k-means clustering

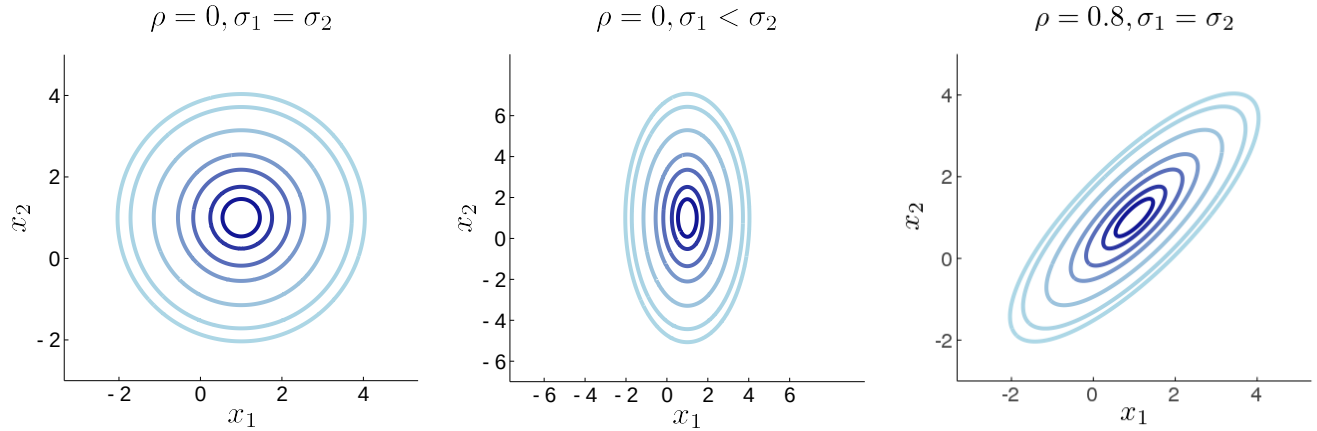
Multivariate normal distribution

- $\mathrm{x} \in \mathbb{R}^p$ is a **multivariate normal**, $\mathrm{x} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, with **density**

$$p(\mathrm{x}) = |2\pi\boldsymbol{\Sigma}|^{-1/2} \exp\left(-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\right)$$

- **Mean** and **covariance** matrix

$$\mathbb{E}(\boldsymbol{x}) = \boldsymbol{\mu} \quad \text{and} \quad \mathrm{Cov}(\boldsymbol{x}) = \boldsymbol{\Sigma}$$



Multivariate normal distribution - properties

- $\mathrm{x} \in \mathbb{R}^p$ is **multivariate normal**, $\mathrm{x} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ with

$$\mathbb{E}(\boldsymbol{x}) = \boldsymbol{\mu} \text{ and } \mathrm{Cov}(\boldsymbol{x}) = \boldsymbol{\Sigma}$$

- Decompose

$$\mathrm{x} = \begin{pmatrix} \boldsymbol{x}_1 \\ \boldsymbol{x}_2 \end{pmatrix}, \boldsymbol{\mu} = \begin{pmatrix} \boldsymbol{\mu}_1 \\ \boldsymbol{\mu}_2 \end{pmatrix} \text{ and } \boldsymbol{\Sigma} = \begin{pmatrix} \boldsymbol{\Sigma}_{11} & \boldsymbol{\Sigma}_{12} \\ \boldsymbol{\Sigma}_{21} & \boldsymbol{\Sigma}_{22} \end{pmatrix}$$

- **Marginal distributions** are normal

$$\mathrm{x}_1 \sim \mathcal{N}(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_{11})$$

- **Conditional distributions** are normal

$$\mathrm{x}_1 | \boldsymbol{x}_2 \sim \mathcal{N}(\tilde{\boldsymbol{\mu}}_1, \tilde{\boldsymbol{\Sigma}}_1)$$

with

$$\tilde{\boldsymbol{\mu}}_1 = \boldsymbol{\mu}_1 + \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}(\boldsymbol{x}_2 - \boldsymbol{\mu}_2)$$
$$\tilde{\boldsymbol{\Sigma}}_1 = \boldsymbol{\Sigma}_{11} - \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}\boldsymbol{\Sigma}_{21}$$

Discriminative vs Generative models

- **Discriminative models**: modeling of $p(y|\mathrm{x})$. No model for x.
  - Logistic regression
  - Regression trees and ensembles
  - Deep neural networks
- **Generative models**: *joint* distribution of labels and features

$$p(y, \mathrm{x}) = p(\mathrm{x}|y)p(y).$$

  - Discriminant analysis
  - Mixture models
- Generative models require more effort:
  - **need also a model for *x***
  - features may be high-dimensional. Hard!
- Generative models:
  - **better understanding** of the mechanisms.
  - can be extended to **unsupervised** (all labels missing) and **semi-supervised** (some labels missing).

Gaussian mixture model - supervised case

- Let $\mathcal{N}(\mathrm{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma})$ denote the density function of $\mathrm{x} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$
- Joint model for label $y \in \{1, \ldots, M\}$ and features $\mathrm{x} \in \mathbb{R}^p$:

$$p(y, \mathrm{x}) = p(\mathrm{x}|y)p(y)$$

where the **class-conditional densities** are normal:

$$p(\mathrm{x}|y) = \mathcal{N}(\mathrm{x}|\boldsymbol{\mu}_y, \boldsymbol{\Sigma}_y)$$

- Known labels: **MLE for each class separately**

$$\hat{\boldsymbol{\mu}}_m = \frac{1}{n_m} \sum_{i:y_i=m} \boldsymbol{x}_i$$

$$\hat{\boldsymbol{\Sigma}}_m = \frac{1}{n_m} \sum_{i:y_i=m} (\boldsymbol{x}_i - \hat{\boldsymbol{\mu}}_m)(\boldsymbol{x}_i - \hat{\boldsymbol{\mu}}_m)^\top$$

- **MLE for discrete labels** $p(y = m) = p_m$

$$\hat{p}_m = \frac{n_m}{n}.$$

Discriminant analysis

- **Classification probability** by Bayes' theorem

$$p(y|\mathrm{x}) = \frac{p(y,\mathrm{x})}{p(\mathrm{x})} = \frac{p(\mathrm{x}|y)p(y)}{p(\mathrm{x})} \propto p(\mathrm{x}|y)p(y)$$

- **Quadratic discriminant analysis** (**QDA**)

$$\begin{aligned}
&\arg\max_m \left(\log \hat{p}(\mathrm{x}_\star|y=m) + \log \hat{p}(y=m)\right) \\
&\quad = \arg\max_m \left(\log \mathcal{N}(\mathrm{x}_\star|\hat{\boldsymbol{\mu}}_m, \hat{\boldsymbol{\Sigma}}_m) + \log \hat{p}_m\right) \\
&\quad = \arg\max_m \left(-\frac{1}{2}\log\left|\hat{\boldsymbol{\Sigma}}_m\right| - \frac{1}{2}(\mathrm{x}_\star - \hat{\boldsymbol{\mu}}_m)^\top \hat{\boldsymbol{\Sigma}}_m^{-1}(\mathrm{x}_\star - \hat{\boldsymbol{\mu}}_m) + \log \hat{p}_m\right)
\end{aligned}$$

- So QDA has a quadratic decision boundary.
- **Linear discriminant analysis** (**LDA**) if we assume

$$\boldsymbol{\Sigma}_1 = \boldsymbol{\Sigma}_2 = \ldots = \boldsymbol{\Sigma}_M = \boldsymbol{\Sigma}$$

Discriminant analysis

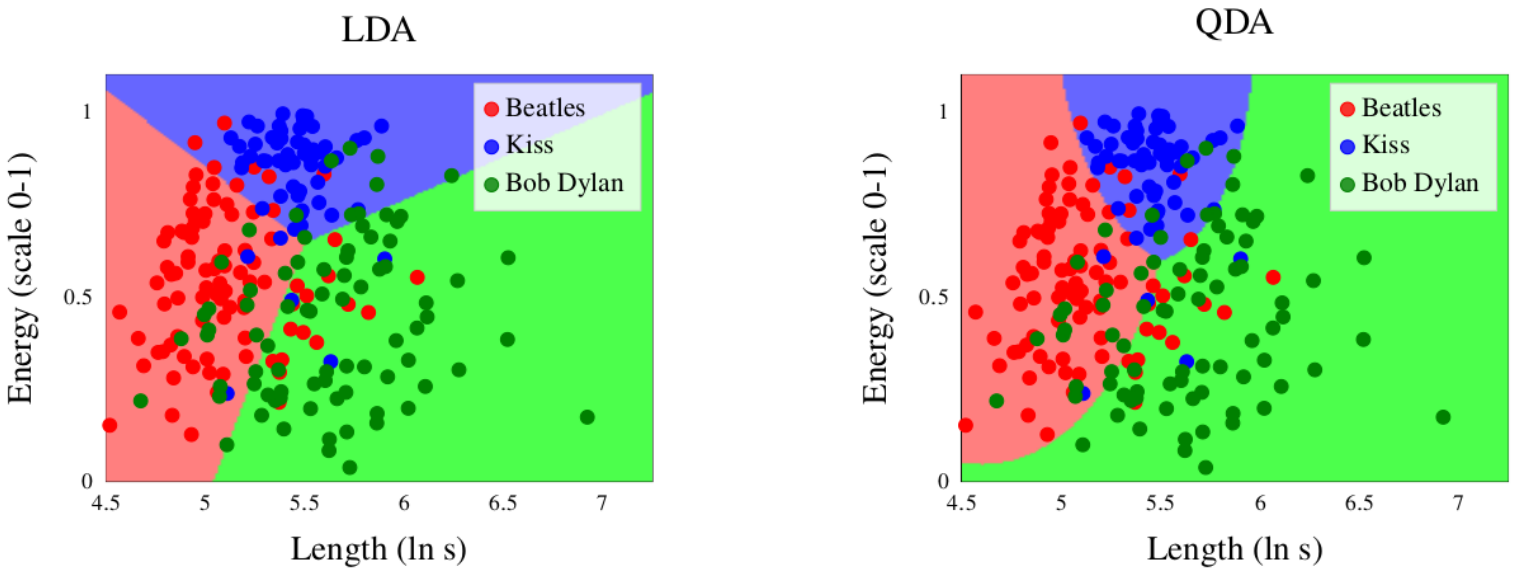


Figure from Lindholm et al (2021).

Gaussian mixture model - unsupervised case

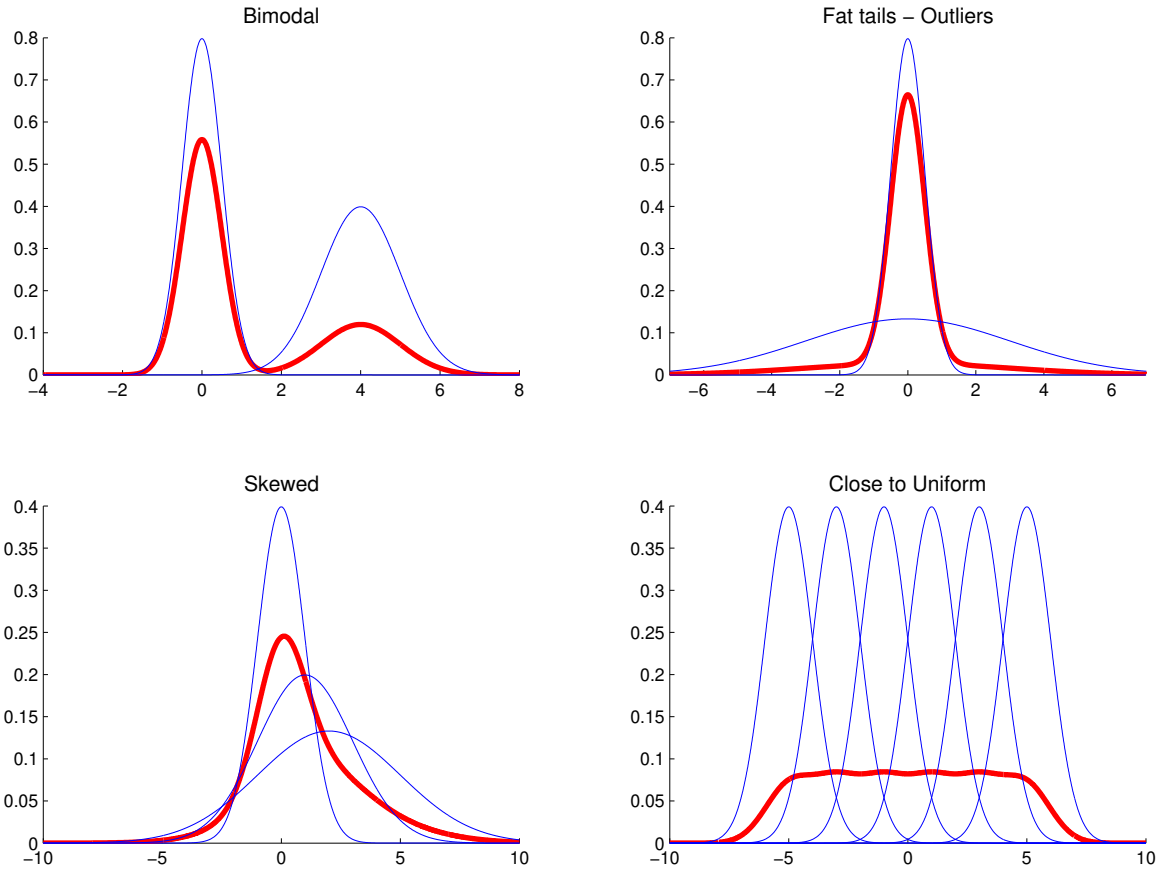
- **Unsupervised**: no labels.
- Don't know from which distribution $\mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m)$ each observation $\boldsymbol{x}_i$ comes from.
- **Two-component univariate mixture of normals**

$$p(x) = \pi \cdot \mathcal{N}(x|\mu_1, \sigma_1^2) + (1 - \pi) \cdot \mathcal{N}(x|\mu_2, \sigma_2^2)$$

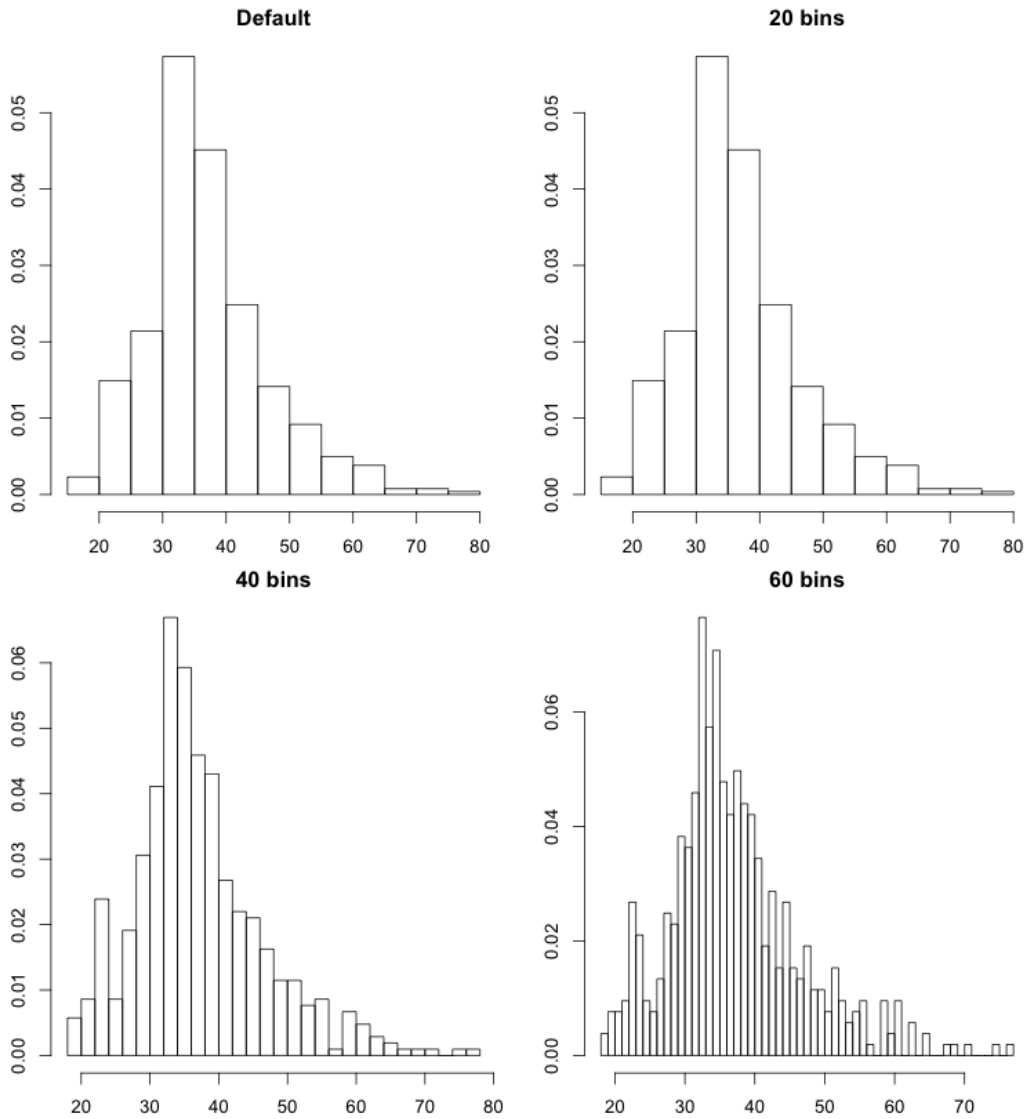
- **Simulate** from a two-component mixture of normals:
  - Simulate $y_i \in \{1, 2\}$, with $\Pr(y_i = 1) = \pi$.
  - If $y_i = 1$, simulate x from $N(\mu_1, \sigma_1^2)$
  - If $y_i = 2$, simulate x from $N(\mu_2, \sigma_2^2)$.
- *M*-**component mixture of multivariate normals**

$$p(\boldsymbol{x}) = \sum_{m=1}^{M} \pi_m \mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m), \quad \sum_{m=1}^{M} \pi_m = 1$$

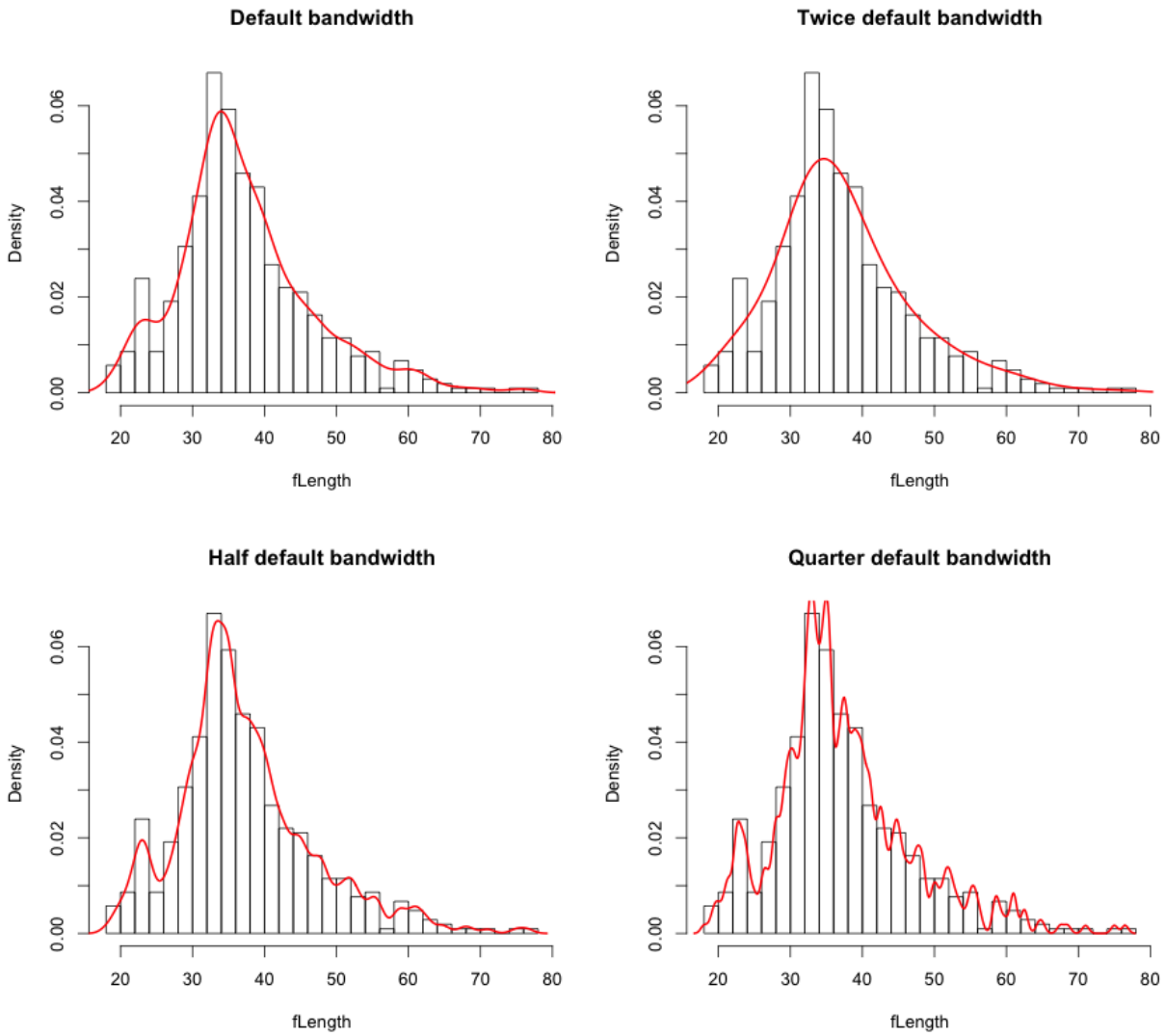
Illustration of mixture distributions



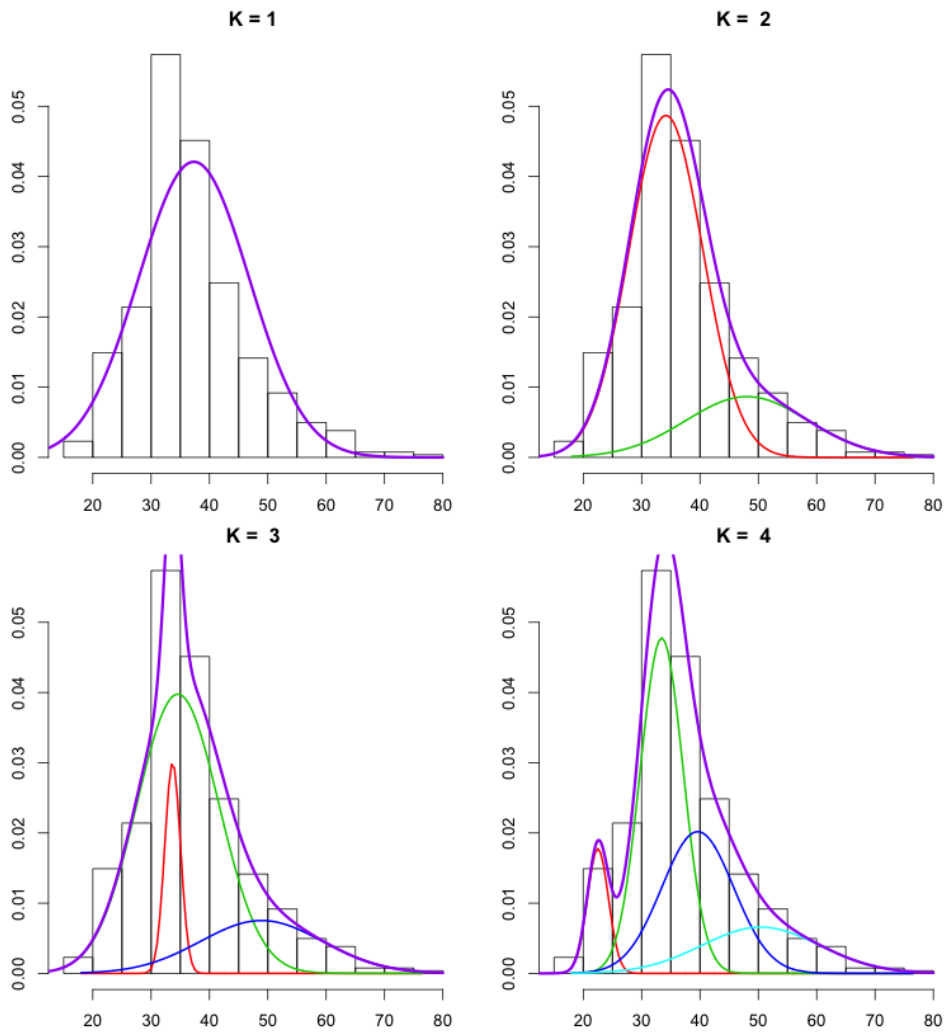
Fish length - Histogram density estimates



Fish length - Kernel density estimates



Fish length - Mixture of normals



- See code `GMM_EM.R` on web page.

EM algorithms for unsupervised Gaussian mixtures

- GMM - the likelihood is a messy product of sums

$$p(\boldsymbol{x}_1, ..., \boldsymbol{x}_n | \boldsymbol{\mu}_{1:M}, \boldsymbol{\Sigma}_{1:M}, \pi_{1:M}) = \prod_{i=1}^{n} \sum_{m=1}^{M} \pi_m \mathcal{N}(\boldsymbol{x}_i | \boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m).$$

- **Complete data likelihood**:

$$p(\boldsymbol{x}_1, ..., \boldsymbol{x}_n, y_1, \ldots, y_n | \boldsymbol{\mu}_{1:M}, \boldsymbol{\Sigma}_{1:M}, \pi_{1:M}) = \prod_{i=1}^{n} \pi_{y_i} \mathcal{N}(\boldsymbol{x}_i | \boldsymbol{\mu}_{y_i}, \boldsymbol{\Sigma}_{y_i}).$$

- Let $\boldsymbol{\theta} = (\boldsymbol{\mu}_{1:M}, \boldsymbol{\Sigma}_{1:M}, \pi_{1:M})$ be all model parameters.
- Iterative **EM-algorithm** for MLE. Given previous estimate $\hat{\boldsymbol{\theta}}$
  - **E-step**: Compute $Q(\boldsymbol{\theta}) \equiv \mathbb{E}_{\boldsymbol{y}} \left(\log p(\boldsymbol{X}, \boldsymbol{y} | \boldsymbol{\theta}) | \boldsymbol{X}, \hat{\boldsymbol{\theta}} \right)$
  - **M-step**: Update $\hat{\boldsymbol{\theta}} \leftarrow \arg\max_{\boldsymbol{\theta}} Q(\boldsymbol{\theta})$
- The **expected log-likelihood** is

$$Q(\boldsymbol{\theta}) = \sum_{i=1}^{n} \sum_{m=1}^{M} w_i(m) \left[\log \mathcal{N}(\boldsymbol{x}_i | \boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m) + \log \pi_m \right]$$

where $w_i(m) \equiv p\left(y_i = m | \hat{\boldsymbol{\theta}}, \boldsymbol{x}_i\right)$.

EM-algorithm for a Gaussian mixture model

- (**E-step**) Compute probabilities for the latent y_i

$$w_i(m) = \frac{\pi_m \mathcal{N}(\boldsymbol{x}_i|\boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m)}{\sum_{j=1}^{M} \pi_j \mathcal{N}(\boldsymbol{x}_i|\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)} \quad \text{and} \quad \hat{n}_m = \sum_{i=1}^{n} w_i(m)$$

- (**M-step**) Given $w_i(m)$, compute ML estimates by maximizing $Q(\boldsymbol{\theta})$:

$$\hat{\pi}_m = \frac{\hat{n}_m}{n}$$

$$\hat{\boldsymbol{\mu}}_m = \frac{1}{\hat{n}_m} \sum_{i=1}^{n} w_i(m) \boldsymbol{x}_i$$

$$\hat{\boldsymbol{\Sigma}}_m = \frac{1}{\hat{n}_m} \sum_{i=1}^{n} w_i(m)(\boldsymbol{x}_i - \hat{\boldsymbol{\mu}}_m)(\boldsymbol{x}_i - \hat{\boldsymbol{\mu}}_m)^\top$$

- Iterate until log-likelihood

$$\sum_{i=1}^{N} \log \left\{ \sum_{m=1}^{M} \pi_m \mathcal{N}(\boldsymbol{x}_i|\boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m) \right\}$$

satisfies some stopping rule.

Gaussian mixture model for old Faithful data

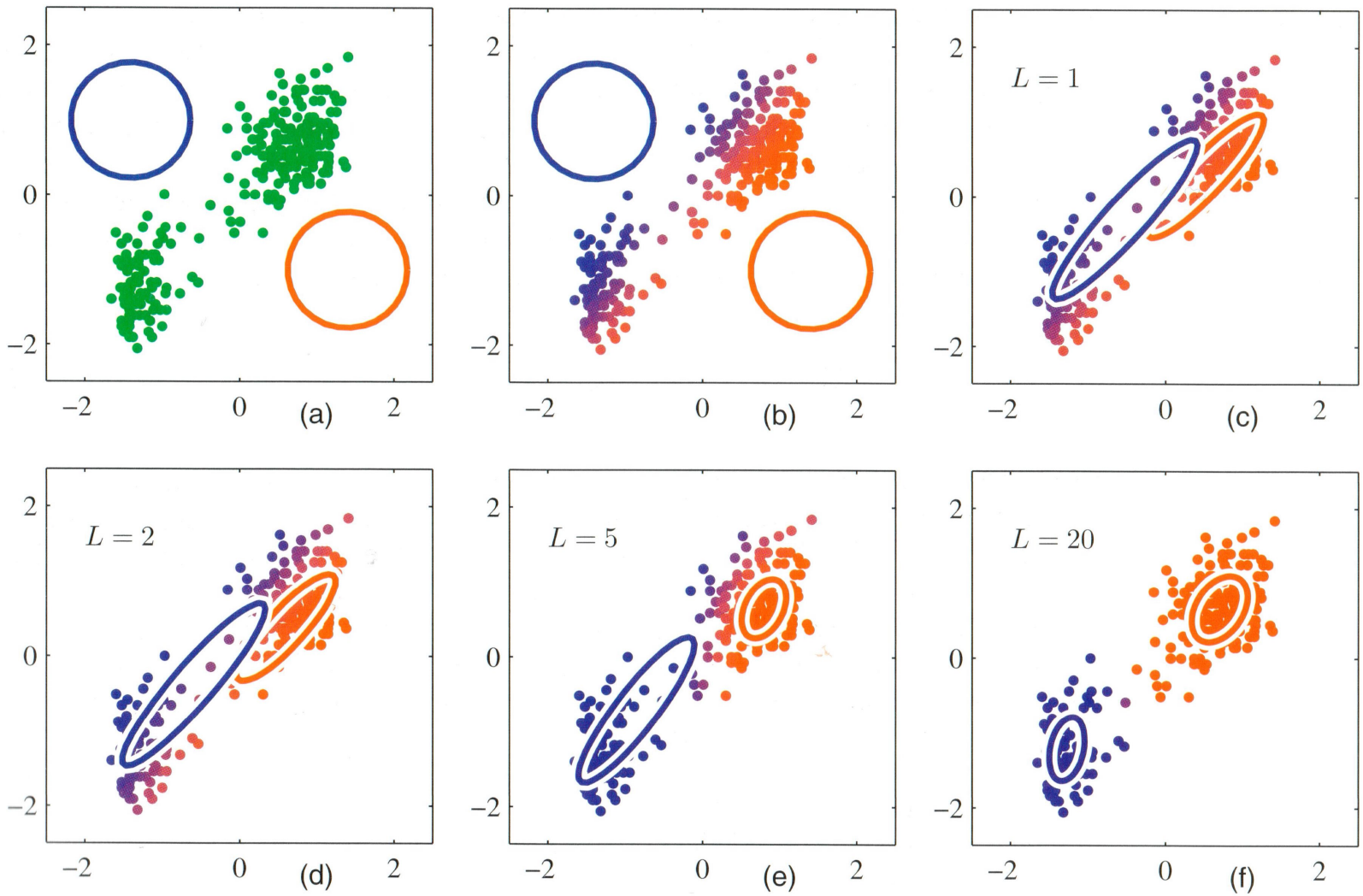


Figure from Bishop (2011).

EM-algorithm for a Gaussian mixture model

- Log-likelihood is **guaranteed to not decrease** at any iteration
- EM is typically **slow**.
  Switch to Newton-Raphson when closer to the maximum.
- EM solution often depends on the **initial values**.
  Restart with different initial values and pick the best solution.
- **Label-switching** for mixtures. There are actually $M!$ maxima.
- EM tends to find a finite maximum even though the **likelihood is unbounded**. Singularities. Likelihood becomes arbitrarily large if mth component is $\mathcal{N}(\boldsymbol{x}_j, \boldsymbol{\Sigma}_m)$ with $\boldsymbol{\Sigma}_m \to 0$.
- (pssst: use a Bayesian prior!)

k-Means clustering

- Data in $\mathbb{R}^p$: $x_1, ..., x_n$.
- Aim: **partition** the data into M **clusters**.
- Each observation x_i is represented by a **centroid** $\boldsymbol{\mu}_m \in \mathbb{R}^p$.
- Let responsibility $r_{im} = 1$ if x_i belongs to $\boldsymbol{\mu}_m$.
- **k-means clustering** minimizes

$$\sum_{i=1}^{n} \sum_{m=1}^{M} r_{im} \|x_i - \boldsymbol{\mu}_m\|^2$$

  with respect to the r_{im} and the $\boldsymbol{\mu}_1, ..., \boldsymbol{\mu}_M$.
- Iterative algorithm: initialize $\boldsymbol{\mu}_1, ..., \boldsymbol{\mu}_M$, then:
  - Allocate each observations to nearest centroid ($r_{im} = 1$)
  - Recompute each centroid as mean of its allocated observations.

k-Means clustering of old faithful data

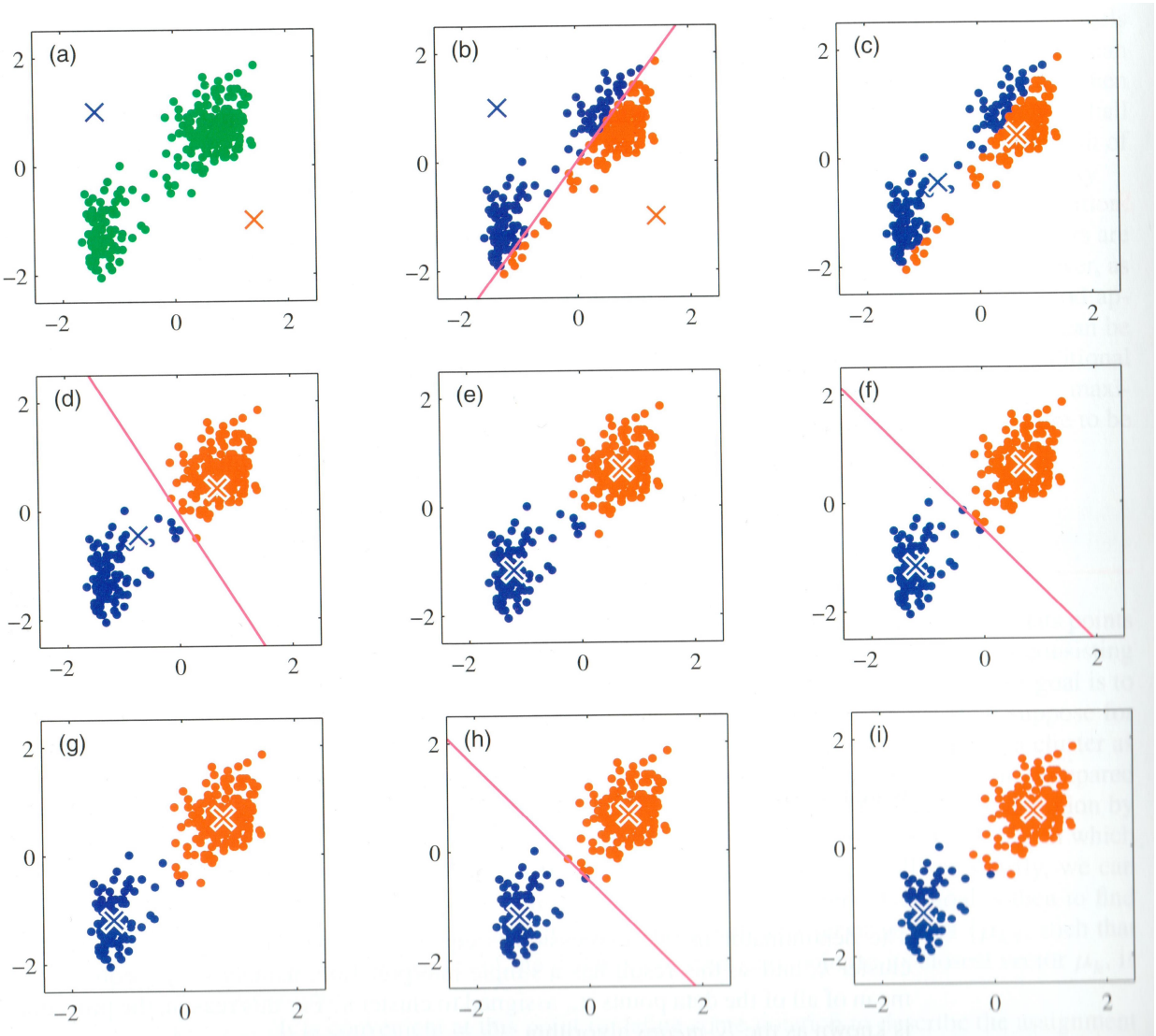


Figure from Bishop (2011).

Clusterings songs by k-means

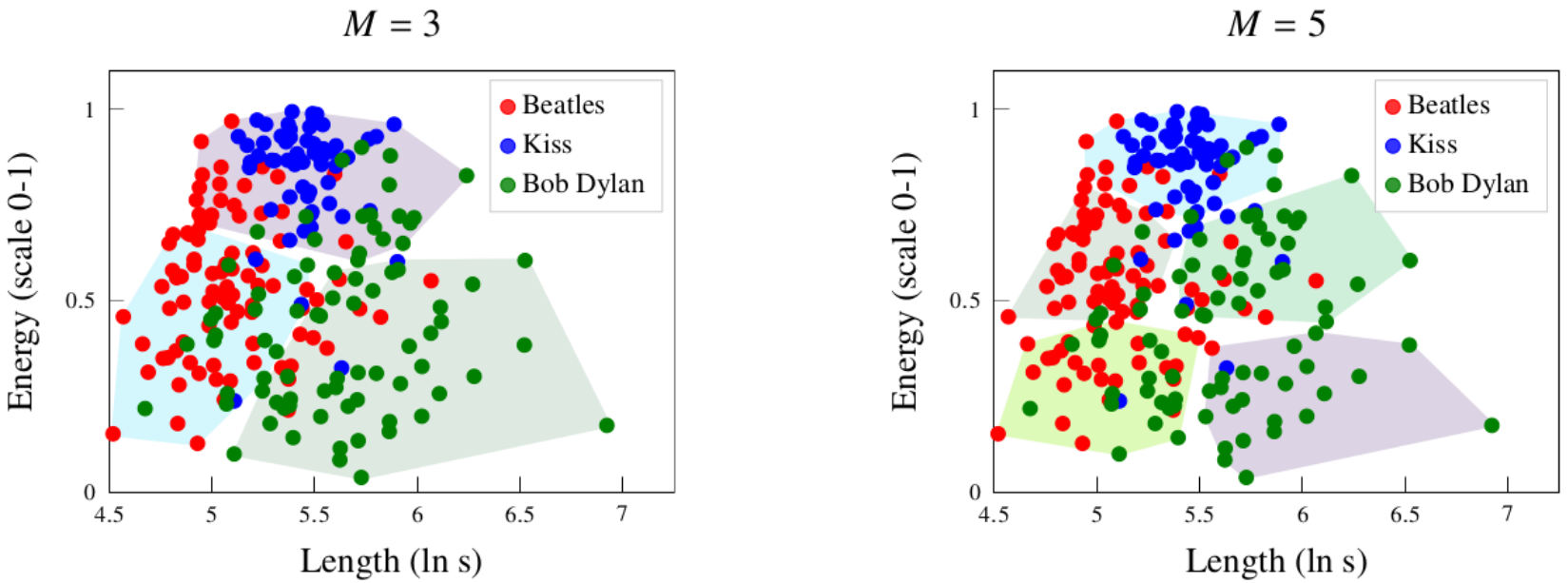


Figure from Lindholm et al. (2021).

Selecting the number of clusters - elbow

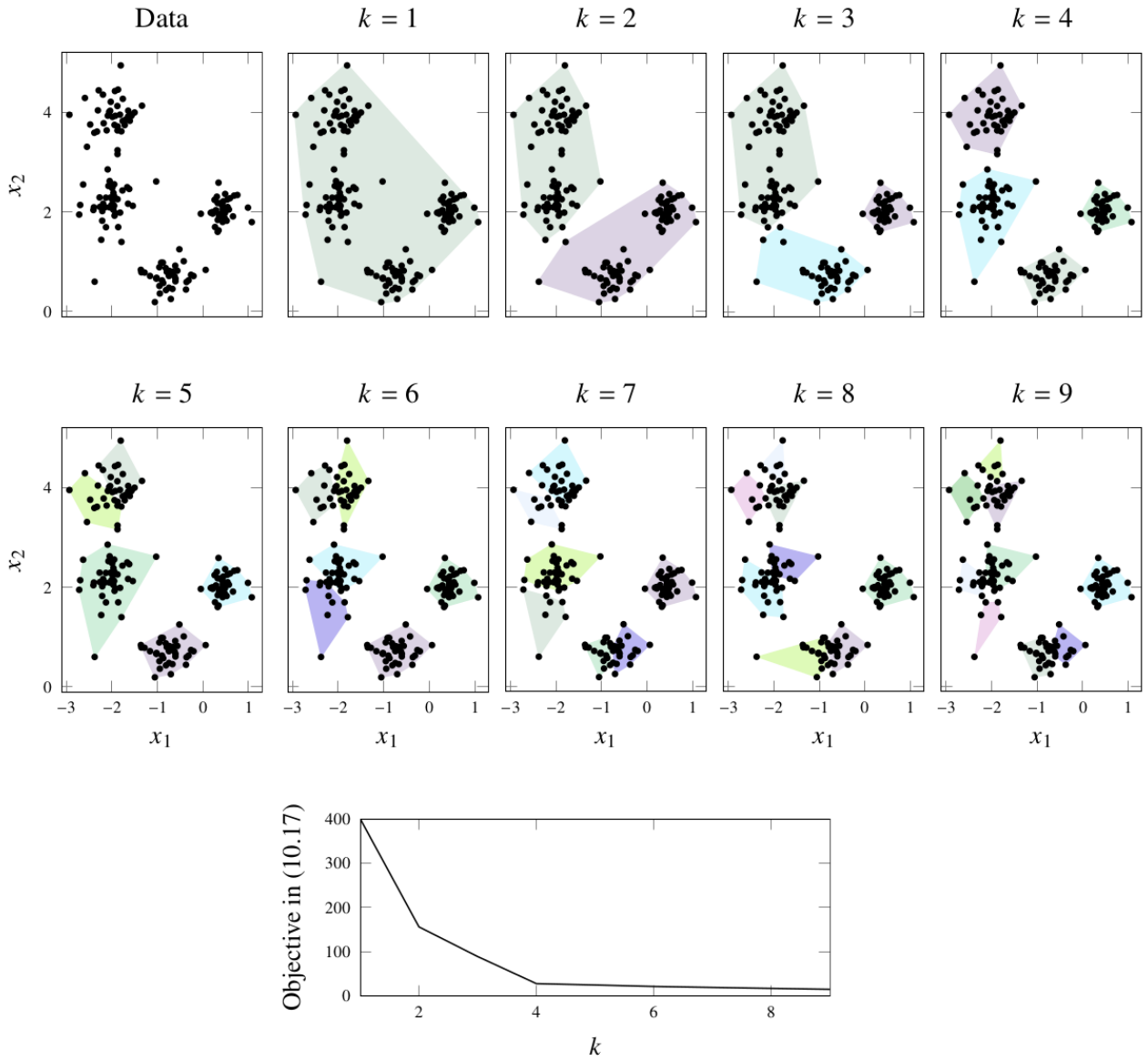


Figure from Lindholm et al. (2021).

Image compression by k-means

Figure from Bishop (2011).