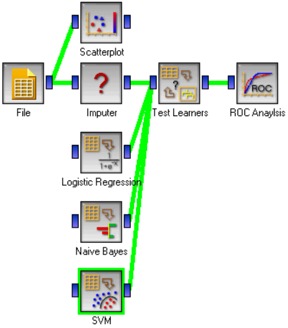
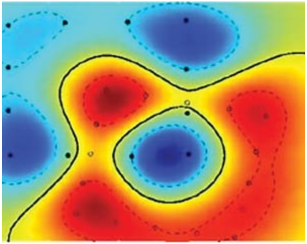
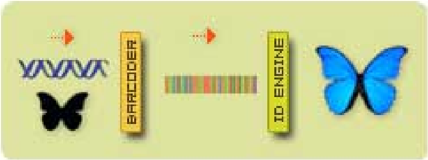
Unsupervised Learning - Clustering







Goal of clustering

- Overall objective: form a partition $C_1, \ldots, C_K$ of a data sample $\{X_1, \ldots, X_n\}$ so that *"data belonging to the same group are more similar to each other than to data lying in different groups"*
- Issues:
  - *"Similar"* in which sense? Metric? Groups should correspond to *modes*, reflect *distribution structure*? How to deal with qualitative data?
  - **Combinatorial** problem: there are

    $$\sum_{m=1}^{K}(-1)^{K-m}\binom{k}{m}m^n$$

    partitions with $K \leq n$ non empty groups
    How to cluster the data in practice?
  - How to choose K?

Techniques for clustering

- Very diverse methods, implemented as a preprocessing stage
- Three groups:
  - hierarchical techniques: agglomerative vs. divisive
  - (nonparametric) Bayesian methods
  - partitional: centroïds, model-based, graph theoretic, spectral clustering
- Most popular procedures:
  - $K - means$
  - Agglomerative hierarchical clustering
  - The EM-algorithm

K-means

- Input: data points in $\mathcal{X}$, distance d on $\mathcal{X}$, number K of clusters
- The clusters are defined by means of **centroïds** $c_1, \ldots, c_K$ in $\mathcal{X}$

$$x \in C_k \Leftrightarrow k = \arg\min_{1 \leq l \leq K} d(x, c_l)$$

- General principle:
  - start with an initial clustering,
    1. define centroïds by means of a given method (ex: cluster means, cluster medians)
    2. reassign the data to new clusters defined by proximity to centroïds
- How many iterations?

K-means

- Usually, centroïds are the current **cluster means**:

$$c_k = \frac{1}{\#\{i : X_i \in C_k\}} \sum_{i:\, X_i \in C_k} X_i$$

- When the metric is the **square Euclidean distance**, the goal is to minimize over $c_1, \ldots, c_K$

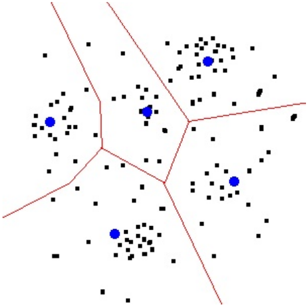
$$\sum_{k=1}^{K} \sum_{i:\, X_i \in C_k} ||X_i - c_k||^2$$

- Minimizing intra-cell variability is equivalent to maximizing inter-cell variability

$$\sum_{(i,j)} ||X_i - X_j||^2 = \sum_{k \neq l} \sum_{(i,j):\, (X_i, X_j) \in C_k \times C_l} ||X_i - X_j||^2 + \sum_{k} \sum_{(i,j):\, (X_i, X_j) \in C_k^2} ||X_i - X_j||^2$$

K-means

- Monotonicity: the within-cluster variability **decreases**
- Convergence to a **possibly local** minimum
- Use the function KMEANS(.)



A typical application - Vector Quantization

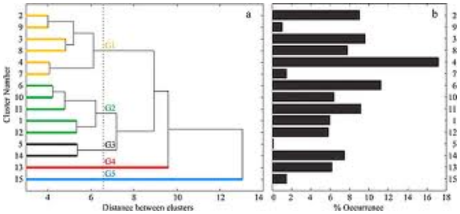
Compress high-dimensional pixellated images into low resolution images

Hierarchical clustering

- Produce a **nested sequence** of clusterings
- The sequence can be represented by a **tree schematic** (dendrogram)
- Either **agglomerative** or else **divisive**
- No need to specify the number K of clusters in advance

Agglomerative hierarchical clustering

- Initially, start with n clusters: the singletons $\{X_i\}$
- Merge the pair of singletons $\{X_i\}$ and $\{X_j\}$ with minimum dissimilarity, yielding $n-1$ clusters
- Iterate: merge the two clusters with minimum dissimilarity
- ...
- Stop when all points have been agglomerated into a single cluster of cardinality n



Agglomerative hierarchical clustering - How to measure dissimilarity between clusters

- "Single linkage"

$$D(C, C') = \min_{(x,x') \in C \times C'} d(x, x')$$

- "Complete linkage"

$$D(C, C') = \max_{(x,x') \in C \times C'} d(x, x')$$

- "Centroïd linkage"

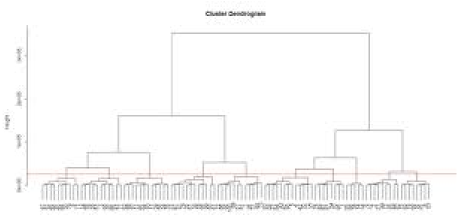
$$D(C, C') = d(\bar{x}_C, \bar{x}_{C'})$$

- "Average linkage"

$$D(C, C') = \frac{1}{\#C\#C'} \sum_{(x,x') \in C \times C'} d(x, x')$$

Divisive hierarchical clustering

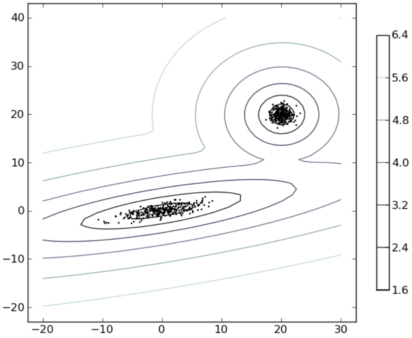
- Start with a single cluster of cardinality n and fix a threshold λ
- Determine the pair (X_i, X_j) with maximum dissimilarity $d_{\max}$
- Compare $d_{\max}$ and t.
  If $d_{\max} < \lambda$, then stops.
  If $d_{\max} > \lambda$, form two clusters: assign X_m to X_i's cluster if $d(X_i, X_m) < d(X_j, X_m)$, to X_j's cluster otherwise
- ...



Model-based clustering

- **Mixture** density model: $f_\theta(x) = \sum_{k=1}^{K} \omega_k f_{\theta_k}(x)$
  $\omega_k \geq 0,\ \sum_{k=1}^{K} \omega_k = 1,\ \theta = ((\theta_1, \omega_1),\ \ldots,\ (\theta_K, \omega_K))$
- Consider Y, **"hidden" class label**:

$$X \mid Y \sim f_{\theta_Y}(x)dx \text{ and } \omega_k = \mathbb{P}\{Y = k\}$$



Model-based clustering - The EM algorithm

- Goal: find a (local) maximum for the log-likelihood

$$L(\theta, \mathbf{X}^{(n)}) = \mathbb{E}_\theta \left[\sum_{i=1}^{n} \log \left(\sum_{k=1}^{K} \mathbb{I}\{Y_i = k\} f_{\theta_k}(X_i) \right) \mid \mathbf{X}^{(n)} \right]$$

- Initialization: start with a guess $\widehat{\theta}^{(0)}$
- Iterations:
  1. E-step: compute $Q(\theta', \widehat{\theta}^{(j)}) = \mathbb{E}_{\widehat{\theta}^{(j)}}[L(\theta', \mathbf{X}^{(n)}) \mid \mathbf{X}^{(n)}]$ for any θ'
  2. M-step: find $\widehat{\theta}^{(j+1)} = \arg\max_{\theta'} Q(\theta', \widehat{\theta}^{(j)})$
- stops when $\widehat{\theta}^{(j+1)} - \widehat{\theta}^{(j)}$ becomes negligible
- The EM-algorithm increases the log-likelihood