Machine Learning
Exercise 9

Marc Toussaint
Machine Learning & Robotics lab, U Stuttgart
Universitätsstraße 38, 70569 Stuttgart, Germany

June 25, 2015

1 Graph cut objective function & spectral clustering

One of the central messages of the whole course is: To solve (learning) problems, first formulate an objective function that defines the problem, then derive algorithms to find/approximate the optimal solution. That should also hold for clustering...

k-means finds centers μ_k and assignments $c : i \mapsto k$ to minimize $\min \sum_i (x_i - \mu_{c(i)})^2$.

An alternative class of objective functions for clustering are graph cuts. Consider n data points with similarities w_{ij}, forming a weighted graph. We denote by $W = (w_{ij})$ the weight matrix, and $D = \text{diag}(d_1, .., d_n)$, with $d_i = \sum_j w_{ij}$, the degree matrix. For simplicitly we consider only 2-cuts, that is, cutting the graph in two disjoint clusters, $C_1 \cup C_2 = \{1, .., n\}$, $C_1 \cap C_2 = \emptyset$. The normalized cut objective is

$$\text{RatioCut}(C_1, C_2) = \Big(1/|C_1| + 1/|C_2|\Big) \sum_{i \in C_1, j \in C_2} w_{ij}$$

a) Let $f_i = \begin{cases} +\sqrt{|C_2|/|C_1|} & \text{for } i \in C_1 \\ -\sqrt{|C_1|/|C_2|} & \text{for } i \in C_2 \end{cases}$ be a kind of indicator function of the clustering. Prove that

$$f^\top (D - W) f = 2n \ \text{RatioCut}(C_1, C_2)$$

b) Further prove that $\sum_i f_i = 0$ and $\sum_i f_i^2 = \sqrt{n}$. Therefore finding a clustering that minimizes $\text{RatioCut}(C_1, C_2)$ is equivalent to

$$\min f^\top (D - W) f \quad \text{s.t.} \quad \|f\|_1 = 0 \ , \quad \|f\|_2 = 1$$

Note that *spectral clustering* computes eigenvectors f of the graph Laplacian $D - W$ with smallest eigenvalues, which is a relaxation of the above problem that minimizes over continuous functions $f \in \mathbb{R}^n$ instead of discrete clusters C_1, C_2.

2 Clustering the Yale face database

On the webpage find and download the Yale face database http://ipvs.informatik.uni-stuttgart.de/mlr/marc/teaching/data/yalefaces.tgz. (Optinally use `yalefaces_cropBackground.tgz`.) The file contains gif images of 165 faces.

We'll cluster the faces using k-means in $K = 4$ clusters.

a) First compute the distance matrix D with $D_{ij} = \|x_i - x_j\|^2$, the squared distance between each pair of images. The rest operates only on this distance matrix.

b) Compute a k-means clustering starting with random initializations of the centers. Repeat k-means clustering 10 times. For each run, report on the clustering error $\min \sum_i (x_i - \mu_{c(i)})^2$ and pick the best clustering.

c) Ideally, repeat the above for various K and plot the clustering error over K.

Display the center faces μ_k and perhaps some samples for each cluster.