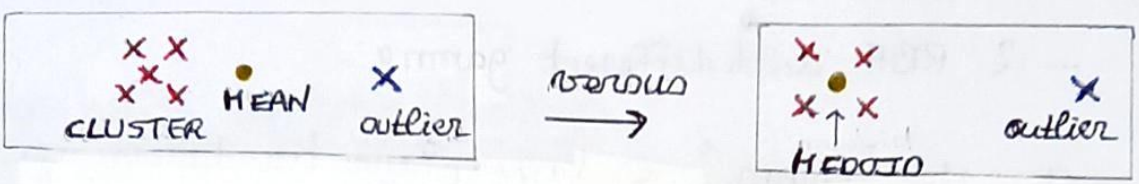
MLH - HW3 - NATHAN LEVY

I) Clustering

(a) In the K-Means algorithm, the centroid of a cluster is the point μ such that the euclidean metric between the examples and μ is minimal. If an outlier is far from the other points of the cluster, we expect μ to be shifted towards the outlier to minimize the distance. Conversely, the K-medoids uses an actual point in the cluster, and we seek to minimize the sum of distances between the medoid and the other points of the cluster. An outlier will probably maximize this distance and thus will not be chosen as the *medoid*. The medoid will be at the center of the cluster. To conclude K-medoid is more robust to outliers.

Example:



(b) We consider m examples $(x_i)_{1 \leq i \leq m}$ in $\mathbb{R}$.
We want to minimize

$$J(\mu) = \sum_{i=1}^{m} (x_i - \mu)^2$$

Which is the sum of the distances of the examples to μ.

We recognize that J is similar to the linear regression cost function, which is convex.

Thus, $J(\mu^*)$, with

$$\mu^* = \underset{\mu \in \mathbb{R}}{\operatorname{argmin}} \left[J(\mu) \right]$$

is a global minima. Thus:

$$\frac{dJ}{d\mu} = 0 \iff -2 \times \sum_{i=1}^{m} (x_i - \mu) = 0.$$

$$\iff \sum_{i=1}^{m} x_i = \sum_{i=1}^{m} \mu = m \times \mu.$$

$$\iff \mu = \frac{1}{m} \sum_{i=1}^{m} x_i.$$

Thus J is minimal when $\mu^* = \mu = \frac{1}{m} \sum_{i=1}^{m} x_i$

which is the mean of the m examples.

II) SVM

We are given 6 different SVM settings:

- 2 linear with different capacities C.
- 2 polynomial with different degrees
- 2 RBF with different γ.

- To begin with, the "linear" or hyperplane boundaries are associated with the linear SVMs.
So A-D are 1 or 2.
The difference is the capacity, namely how much do we want to penalize misclassifications.
In case A, we "allowed" some observations to be within the margin, "in the street". It means C was lower than in case D, where there are no such tolerance as C is higher:

A → linear kernel, $C = 0.01$	= CASE 1
D → linear kernel, $C = 1$	= CASE 2

- Now we can deal with the RBF.
γ controls { the "shape" / the "spread" } ⇒ { γ gets large → narrow shape / γ small → large, less "tight". }
The boundary shape is closed, circle-like, like E and B.
B is less "spreaded" than E, so its γ is higher.

B → RBF, $\gamma = 1$	CASE 6
E → RBF, $\gamma = 0.2$	CASE 5

- Let's end with the 2 polynomial kernels, namely C and F. The "complexity" of the shape of the boundary increases with the degree, thus:

C → polynomial, degree = 2 → CASE 3
F → polynomial, degree = 10 → CASE 4

III) Capability of generalization

(a) Our balance is the bias-variance tradeoFF.
Excessively complex models will have low bias/high variance and can lead to overFitting. Conversely, "simpler" models have a high bias and underFit your training set.
Thus we should be *parsimonious* in our model choice to ensure a good capability of generalization, without being so simple that we would lose all the explanatory power.

(b) - The First term of AIC is a measure of the model complexity (the more parameters you learn, the higher is the complexity), therefore it affects the "variance" term, i.e. the model will behave very differently on different data sets.
- The second term is a measure of the goodness of fit of the model, therefore it affects the *bias* term.
The higher the likelihood is, the more the model explains the data and captures the relationship between the variables - lowering the bias.

(C) If the balance is violated, two options may happen:

- The model will overfit the training examples, leading to bad generalization (balance broken by too high variance)

- IF we violate the balance due to high bias, it means that the algorithm will miss the relationships between the features, leading to underFitting.

(D) In GMM, we seek to maximize the log-likelihood L (minimize $-L$)
But to achieve a parsimonious model, we should minimize the number of parameters.

Thus we want the AIC to be low, such that we penalize models that are too complicated and reward models that explain well our data.