HW3 – Theoretical Assignment

1. Clustering

A. The K-medoid algorithm is indeed more robust to noise (or outliers) than the K-means algorithm.

K-medoid tries to minimize the L1 distance between all the examples in one cluster from their *medoid* point, which is an example from the data set that was selected to be the center of that cluster. K-means, on the other hand, tries to minimize the Euclidian distance between all the examples in one cluster from their *centroid* point, which is a certain point in that cluster's space. This difference makes K-medoid more robust to noise since it will be less affected by it due to its limitation in selecting the medoid point. Since this point can only be one of the examples from the given data set (and not any point in the cluster's space like in K-means), it reduces the effect of noise on the results. In addition, minimizing the Euclidian distance is more sensitive to noise and outliers than minimizing the L1 distance.

B. In order to minimize the given term, we will differentiate it and compare it to 0:

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

$$\frac{dy}{d\mu} = -2\sum_{i=1}^{m} (x_i - \mu) = 0$$

$$\sum_{i=1}^{m} (x_i - \mu) = 0$$

$$\sum_{i=1}^{m} x_i - m \cdot \mu = 0$$

$$\boldsymbol{\hat{\mu} = \frac{\sum_{i=1}^{m} x_i}{m} = \bar{x}}$$

2. SVM

As can be seen from the figures, figures A and D are the results of SVM with linear kernel (straight separation lines), figures B and E are the results of SVM with RBF kernel (Gaussian-shaped separation) and figures C and F are the results of SVM with polynomial kernel.

Specifically, for using a *linear* kernel, the difference is the *parameter C*, which is the parameter that represents the level of penalization. When C is small, the separating

When using a *linear kernel*, the difference is the parameter C, which represents the level of penalization. When C is small, the separation line will have larger margins, even if it will lead to misclassification (as can be seen in figure **A**), while for a larger value of C, the separation line will have smaller margins in order to better classify the training set (as can be seen in figure **D**).

For an *RBF kernel*, the difference is the parameter γ, which represents the level of fit to the training data. When γ is small, the separation lines will fit less to the data (as can be seen in figure **E**) comparing to the separation lines calculated with a larger γ (as can be seen in figure **B**).

Finally, figure **C** represents a 2^{nd} order polynomial kernel according to its shape, and figure F represents 10^{th} order polynomial kernel.

To sum up:

A – 1; B – 6; C – 3; D – 2; E – 5; F – 4.

3. Capability of generalization

A. The scientific term of the balance that Einstein meant to in ML aspect is that the model we search for our data should be both *the best one* and *the simplest one*. This means the desired model should contain the minimum number of parameters which will allow a good-enough fitting to the data, and not one parameter less. Hence, it is as simple as possible but not simpler.

B. The terms in AIC affect the above-mentioned balance from section (A) in the following manner:

* *2p* – Since *p is the total number of learned parameters*, if it is too large, the model will fit better to the data (and might reach overfitting), and the model's complexity will increase, so it will not be as simple as possible (as required) anymore. On the other hand, if p is too small, the model will not fit well to the data (and might reach underfitting) and it will be simpler than necessary.

* *2ln($\hat{L}$)* – Since $\hat{L}$ *is the estimated likelihood given the mentioned parameters*, if it is too large, it means that the goodness of fit is very high, and we might be overfitting the model to the data. This probably means that the model we are using is too complex, and a simpler model to use can be found. On the other hand, if $\hat{L}$ is too small, it means that the goodness of fit is very low, and we are probably using a too-simple model for the given data.

C. The options that are likely to happen if this balance is violated are:

1. Overfitting: if we choose a too-complex model, which contains more parameters than the minimum parameters needed for that data, we might reach overfitting of the model. In that case, the model will "learn" the given data so well, that it will not be able to generalize new data given to it.

2. Underfitting: if we choose a simpler model (a model that uses less parameters) than the "ideal model" (which contains the minimum parameters needed for that data), we might reach underfitting of the model. In that case, the model will not be able to learn the given data as well as expected.

D. The AIC should be as low as possible, since we want to find the ideal balance between a high number of parameters (larger p), and its goodness of fit (which is represented by $\hat{L}$). The criteria "penalizes" for a high number of parameters - p, and "rewards" a high goodness of fit - $\hat{L}$.