# ***HW3***

## 1. Clustering:

a. K-medoid is more robust to noise and outliers than the K-means algorithm. After choosing the centroids' locations randomly, the K-means algorithm update the centroids' locations according to the mean value of the all the distances of the examples which assigned to it. K-means algorithm rely on the mean values, which are influenced from outlier values (extremely big and small), and therefore it is more sensitive to noise and outliers. By contrast, the K-medoid algorithm does not rely on the mean value. It takes the most centrally located medoids from the data, so the energy of the entire system is minimized. Because K-medoid algorithm chooses the medoids from the data, it is less sensitive to noise and outliers like the K-means algorithm.

b. In order to minimize the term, we will calculate the derivative and compare to zero:

$$\left(\sum_{i=1}^{m}(x_i-\mu)^2\right)' = 2\sum_{i=1}^{m}(x_i-\mu) = 0$$

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

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

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

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

We will check that it is a minimum point:

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

The answer is positive so the point we found is the minimum point.

We got that the minimum value for the term is when $\mu$ is the mean of m examples.

## 2. SVM

Linear kernels:

A and D – because in those images the classifiers are linear function (linear line). C is the punishment- how much we "pay" for misclassification. When C is low, there is more misclassification, but also more generalization. When C is high, there are less misclassification, but higher risk to overfitting.

1. **Linear kernel with C = 0.01 – match to image A.** This is because the classifier is more general and there is more misclassification than in D, (we can see the 2 samples enter the margins).
2. **Linear kernel with C = 1 - match to image D.** This is because at this image, the classifier is more fitted to the data. There is less misclassification than in A, and therefore it is also less general.

RBF kernels:

We can see that the images that describe classifiers with RBF kernel are E, B. we can identify them, because RBF kernel can classify data in a close radial polygon. In RBF kernel, we can influence the overfitting with $\gamma$ parameter. As $\gamma$ is higher there is less misclassification, but also less generalization, and higher risk to overfitting. When the $\gamma$ is lower, there is more generalization and lower risk to overfitting, but also higher risk to misclassification.

5. **RBF kernel with $\gamma$= 0.2 – match to image E**. In this image the classifier more general, according to lower $\gamma$.

6. **RBF kernel with $\gamma$ = 1 – match to image B**. In this image we can see that the classifier is more fitted to the data and have a shape that is more similar to the shape of the samples' spread. Therefore, it matches to the classifier with higher $\gamma$.

Polynomial kernels:

We can see that images C and F are Polynomial kernels because there are not linear and does not have the radial shape of RBF. As the order of the polynomial kernel is higher, the classifier is more fitted to the data. That means that there is there is less misclassification, but also less generalization, and higher risk to overfitting.

3. **2nd order polynomial kernel – match to image C**. In this image we can see that the classifier is less fitted to the data from image F. It is more general and therefore matches to the classifier with lower order the polynomial kernel.
4. **10th order polynomial kernel - match to image F**. In this image we can see classifier that looks like overfitting. The classifier is very fitted to the data and have a shape that is very similar to the shape of the samples' spread. It does not look general and therefore, it matches to the classifier with higher order the polynomial kernel.

3. Capability of generalization:

   a. The scientific term of balance that Einstein meant to in machine learning aspect is generalization. We want our model/ classifier to be simple, and that way to generalize many different samples. In the other hand, we do not want our model to be too much simple (general), because this will cause a lot of misclassifications and inaccuracy. We want our model to classify the samples correctly, but also to be simple and have capability of generalization, so it could also predict the data it did not practice before.
   b. $2p$ – represents the number of learned parameters. As the number of parameters is higher- the model is more complicated and therefore less simple and less general. As the number rise, the model becomes more accurate for specific data and there is a risk to overfitting.
      $2\ln(\hat{L}) - (\hat{L})$ is the estimated likelihood given those parameters. Therefore, as the term increase, it represents that our model gets better in exploring our data and fit it well. Refer to generalization, it means that our model is less general, fit to our data and therefore know how to classify the data.
      The two terms balanced each other, when the model fit to the data $2\ln(\hat{L})$ will increase, but $2p$ will also increase (more complicated). When $2p$ the model is simpler, but then $2\ln(\hat{L})$ will be also lower.
   c. If the balance was violated:
      - In a case of too much generalization it will lead to underfitting. The model will be too general, so it will not be able to classify well our data.
      - In the opposite case of lack in generalization, it will lead to overfitting. The model will be specific for the data that it already saw and fitted to it, so when it will get different data it will mot know how to classify it.
   d. AIC should be low:
      $2p$ low represents low number of parameters and therefore a simpler model. It means the model is more general and there is less risk to overfitting.
      $2\ln(\hat{L})$ –represents that our model gets better in exploring our data. In the term it comes with a minus, so as the model is better, AIC is lower.