HW3

1. a. K-medoid is indeed more robust to outliers than K-means. This is due to the fact that when we calculate the median, we do not actually consider the "distance" from the median, but only the number of values above and below the median value in the data set (where the median is exactly the middle value). However, when we calculate the mean, we are actually taking into account how small or big each value is, thus outliers can offset the mean value from our expected value.

   b. Let's prove that μ which minimizes the term $\sum_{i=1}^{m}(x_i-\mu)^2$ is the mean.

   Let us use $\bar{x}$ as the mean of our m examples, so we write:

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

   Let's say now that $a=x_i-\bar{x}$ and $b=\bar{x}-\mu$ so since $(a+b)^2=a^2+2ab+b^2$:

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

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

   We could take $(\bar{x}-\mu)$ out of the sum since it's a constant. Now we need to remember what the mean actually is:

$$\bar{x}=\frac{1}{m}\sum_{i=1}^{m}x_i\rightarrow\sum_{i=1}^{m}x_i=m\bar{x}$$

   So if we look at our middle expression in the calculation we get:

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

   So we plug that in our calculation and continue:

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

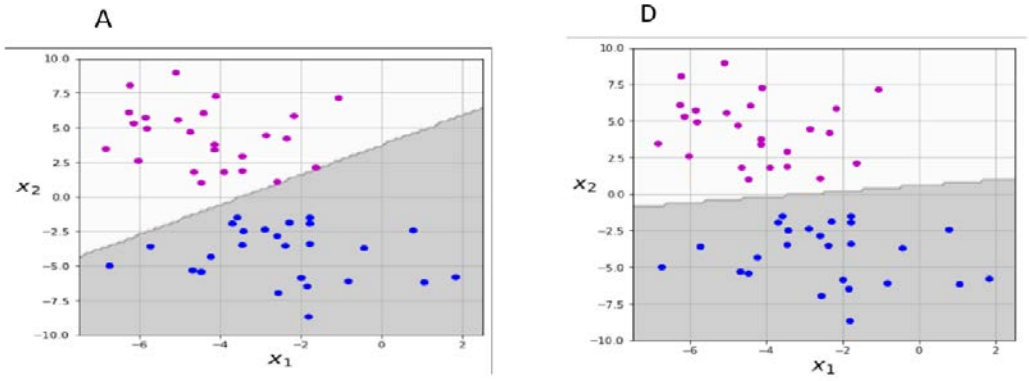
   Minimizing this expression as a function of μ would be achieved when $\mu=\bar{x}$, Since the right part of this expression would be equal to zero (and it can't be negative).

Bonus: Let's try and prove that μ which minimizes the term $\sum_{i=1}^{m}|(x_i-\mu)|$ is the median (assuming μ is indeed in the dataset). This time we will work with the derivative.
Since we want to minimize this expression as a function of μ, and since the derivative of an absolute value is the sign function, we write:

$$\frac{d}{d\mu}\sum_{i=1}^{m}|(x_i-\mu)| = \sum_{i=1}^{m} sign(x_i-\mu) \overset{want}{=} 0$$

So what we have now is a sum of sign function values (either +1 or -1). This sum is equal to zero only when the number of positive values is equal to the number of negative values. This is achieved only when the chosen μ is exactly the median, since according to definition, a median is the value where exactly half of the dataset is smaller than him and half is bigger.

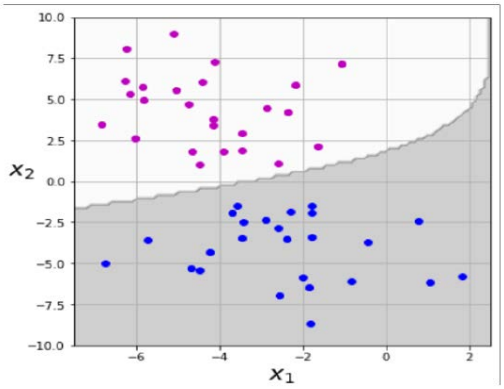
2. In this part, we will match each SVM visualization with the fitting setting:
First thing we can notice is that there are two figures that have a linear kernel:



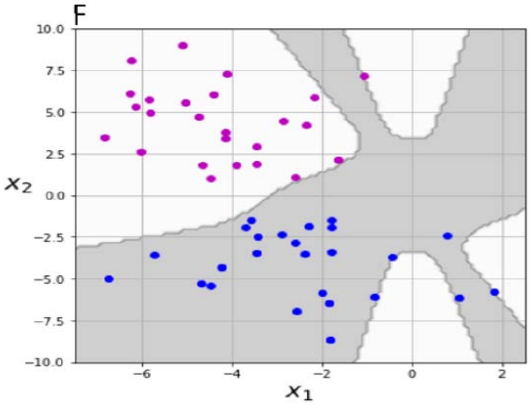
This can be easily seen because the boundary is a linear line. The only difference is that for one of these boundaries the value of C is 1 and for the other the value is 0.01 .
Since we know that the margin is defined by our support vectors, which are supposed to be symmetrical on both sides of the boundary and parallel to it, we can assume that the 2 purple dots located close to the boundary in figure A are inside the margin. This means our degree of tolerance is higher than in figure D, since we tolerate some data in the margin, unlike figure D where the closest dots to the boundary are farther away.
Therefore, since the C parameter dictates the degree of tolerance, we assume figure D has a higher C value – matching a C of 1, and figure A has a C value of 0.01 .

As for the polynomial kernel, we can again notice two figures that fit:

C



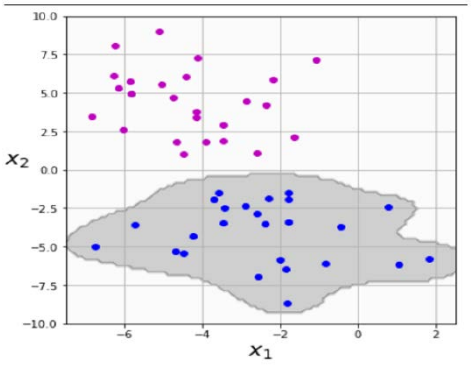
F



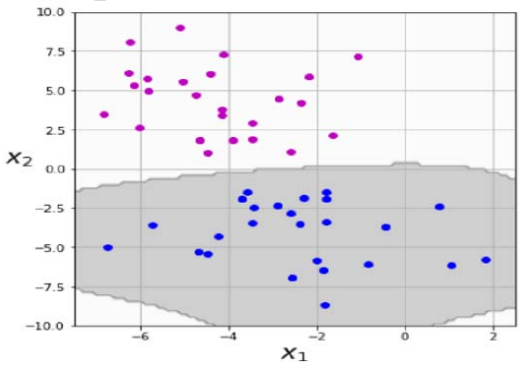
We can see that in picture C we have a margin which fits a 2nd degree polynomial because of the parabolic shape. We can also see that the fit looks generalized which also fits a low degree polynomial kernel. However, in F we can see a much more complex representation, which fits the data set “too good”. This indicates we might have an overfitted model – which fits a high degree polynomial kernel.

We are left with the last two figures:

B



E



These figures fit the settings of an RBF kernel. The RBF kernel works as a transformer to generate new features by measuring the distance between all other dots to specific dots, which are the ”centers”. We are left to decide which is the right γ parameter. We know that the γ parameter control how well we fit the data – the higher γ is, the higher the penalty, which causes a higher fit. This means that high γ values can cause overfitting. We can see that in figure B the classification is fitted “tighter” than in E which is more generalized. So, we conclude B has $\gamma = 1$ and E has $\gamma = 0.2$.

In summary, our matchings are as follows:

A - 1. Linear kernel with C=0.01

B - 6. RBF kernel with γ=1

C - 3. 2nd order polynomial kernel.

D - 2. Linear kernel with C=1

E - 5. RBF kernel with γ=0.2

F - 4. 10th order polynomial kernel.

3. a. In Machine Learning, we also want to keep things as simple as possible. A simpler hypothesis representation is less prone to overfitting. We are achieving this by applying **Generalization**. When we generalize our solution – we keep the balance between how complex our solution is, and how well it describes our problem. If we make the solution too complex, we might fit the data we trained with well, but the "phenomenon" will not be described as well as the training data. By generalizing we can avoid this issue.

   b. When we look on the AIC, we can see that the two parameters are exactly the two that are needed to be kept in balanced. p – which is the total number of learned parameters, actually describes the complexity of the problem – the more parameters we have, the more complex the model is. $\hat{L}$ – which is the estimated likelihood, actually describes the "goodness of fit" to our data. By keeping the balance between the goodness of fit and the complexity of the problem, we make sure our problem is generalized, thus minimizing the AIC. We can also see that $\hat{L}$ is represented as ln($\hat{L}$), thus indicating that an increase in the number of parameters should be accompanied in a large increase in the likelihood to be "worth it". Otherwise, the AIC would increase.

   c. If the balance is violated, we are risking running into **overfitting** and **underfitting**. If our p – the numbers of parameters is very large, we might have an overfit model, because we defined too many parameters based on our training data in order to fit it.

However, if the number of parameters is too small, resulting in a small likelihood based on these parameters (or not much of the variance explained as a result of the variables), we might have an underfit model.

d. We aim to keep the AIC low. The best-fit model according to AIC is the one that explains the greatest amount of variation using the fewest possible independent variables. The AIC penalizes models that use more parameters, so for example - if we have 2 models with a different number of parameters which explain the same amount of variation – or fit the data equally, we will choose the one with the lower AIC.
Thus, to each the best model, we can either start with a small number of parameters and increase them, thus decreasing the AIC until we reach a tipping point where the increase in parameters does not provide a large increase in the fit, or we can work the other way around, start with a large number of parameters and then decrease, until we reach a tipping point where removing parameters becomes “harmful” to our model.