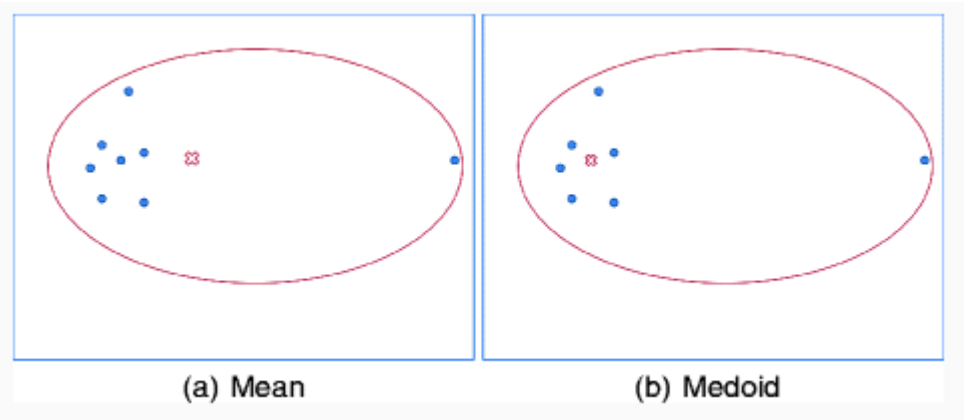
Hw3

Noam Talor 206345852

1. **Clustering (10%)**
   a. Is K-medoid is more robust to noise (or outliers) than K-means algorithm? Explain your answer.

   K- Medoids is more robust as compared to K-Means, as K-Means has a greater sensitivity to outliers. This happens because in K-Means the center of each cluster is calculated according to the sum of squared Euclidean distances for data objects (meaning shortest distance any vector in the cluster member will have to reach). i.e., outliers will affect the determination of the centroid. However, in the K-Medoids algorithm, there is much less outliers' effect due to the fact that the centroid is selected from the data itself, which means that it is one vector of the data itself.

   This concept is illustrated in the following image:



   (The image source: https://link.springer.com/referenceworkentry/10.1007%2F978-0-387-30164-8_426)

   b. Prove that for the 1D case ($x \in R^1$) of K-means, the centroid (μ) which minimizes the term $\sum_{i=1}^{m}(x_i - \mu)^2$ is the mean of m examples.

   Before proving the claim that the number C that minimizes the distance $\sum_{i=1}^{m}(x_i - C)^2$ is the average, I will first prove another claim that will help me with the necessary proof:

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

   Proof: $\sum_{i=1}^{n}(x_i - \bar{x}) = \sum_{i=1}^{n} x_i - \sum_{i=1}^{n} \bar{x} = \sum_{i=1}^{n} x_i - n\bar{x} \underbrace{=}_{*} \sum_{i=1}^{n} x_i - \sum_{i=1}^{n} x_i = 0$

   $(*)\ \bar{x} = \frac{\sum_{i=1}^{n} x_i}{n} \rightarrow\rightarrow\rightarrow n\bar{x} = \sum_{i=1}^{n} x_i$

   Our proof: $\sum_{i=1}^{n}(x_i - C)^2 = \sum_{i=1}^{n}(x_i - \bar{x} + \bar{x} - C)^2 =$

$$\sum_{i=1}^{n}(x_i-\bar{x})^2+2\sum_{i=1}^{n}(x_i-\bar{x})\,(\bar{x}-C)+\sum_{i=1}^{n}(\bar{x}-C)^2$$

$$=\sum_{i=1}^{n}(x_i-\bar{x})^2+2(\bar{x}-C)\underbrace{\sum_{i=1}^{n}(x_i-\bar{x})}_{0-I\ have\ prooved\ before}+n(\bar{x}-C)^2$$

$$=\underbrace{\sum_{i=1}^{n}(x_i-\bar{x})^2}_{constant-doesnt\ depend\ on\ C}+\underbrace{n(\bar{x}-C)^2}_{non-negative\ constant}$$

The minimum is obtained when $n(\bar{x}-C)^2$ is zero - and this happens when C is equal to the average.

Bonus: <u>Prove that the centroid (practically, the medoid) which minimizes the term $\sum_{i-1}^{n}|(x_i-\mu)|$ is the median of m examples given that μ belongs to the dataset.</u>

First we will arrange the x_i in ascending order from the smallest-$x_1$to the biggest x_n. Now we can say that (μ belongs to the dataset means $\forall\mu\in[x_1,x_n]$):

$$\sum_{i=1}^{n}|(x_i-\mu)|=\sum_{i=2}^{n-1}|(x_i-\mu)|+|x_1-\mu|+|x_n-\mu|,\qquad\forall\mu\in[x_1,x_n]$$

$$if\ \mu=x_1,\qquad then\ \sum_{i=1}^{n}|(x_i-\mu)|=\sum_{i=2}^{n-1}|(x_i-\mu)|+0+x_n-x_1$$

$$if\ \mu=x_n,\qquad then\ \sum_{i=1}^{n}|x_i-\mu|=\sum_{i=2}^{n-1}|x_i-\mu|-(x_1-x_n)+0$$

$$if\ x_1<\mu<x_n,\qquad then\ \sum_{i=1}^{n}|(x_i-\mu)|=\sum_{i=2}^{n-1}|(x_i-\mu)|-(x_1-\mu)+(x_n-\mu)$$

i.e., anyway: $\sum_{i=1}^{n}|(x_i-\mu)|=\sum_{i=2}^{n-1}|x_i-\mu|+(x_n-x_1)$

Now let generalize it:

Assuming that n is odd-

$$\sum_{i=1}^{n}|x_i-\mu|=\left|x_{\frac{n+1}{2}}-\mu\right|+(x_n-x_1)+(x_{n-1}-x_2)+\cdots+\left(\underbrace{x_{\frac{n+3}{2}}}_{\frac{n+1}{2}+1}-\underbrace{x_{\frac{n-1}{2}}}_{\frac{n+1}{2}-1}\right)$$

i.e. $\sum_{i=1}^{n}|x_i-\mu|=\left|x_{\frac{n+1}{2}}-\mu\right|+constant$

The minimum is obtained when $x_{\frac{n+1}{2}}-\mu$ is zero - and this happens when μ is equal to the median- to $x_{\frac{n+1}{2}}$.

Assuming that n is even-

$$\sum_{i=1}^{n}|x_i-\mu| = +(x_n - x_1) + (x_{n-1} - x_2) + \cdots + \left|x_{\frac{n}{2}} - \mu\right| + \left|x_{\frac{n+2}{2}} - \mu\right|$$

i.e. $\sum_{i=1}^{n}|x_i-\mu| = \left|x_{\frac{n}{2}} - \mu\right| + \left|x_{\frac{n+2}{2}} - \mu\right| + constant$

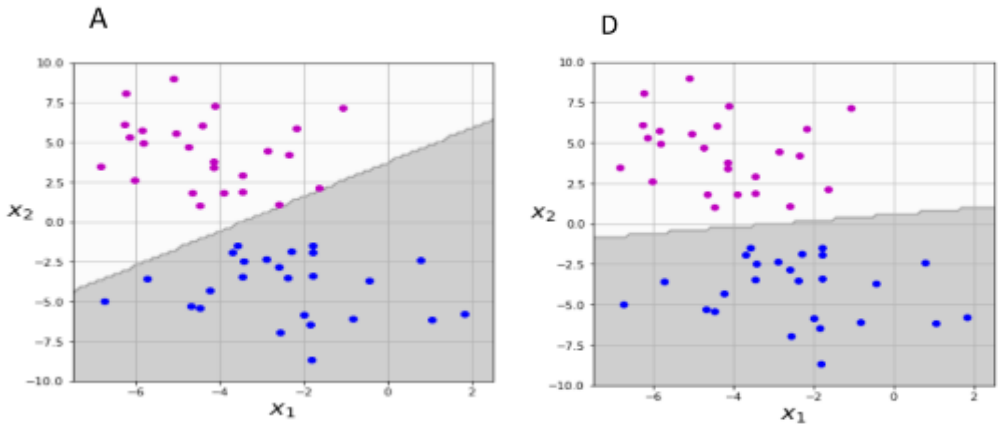
let derivate it and compare to 0 to find the minimum of the function-

$$\frac{\left|x_{\frac{n}{2}} - \mu\right|}{x_{\frac{n}{2}} - \mu} + \frac{\left|x_{\frac{n+2}{2}} - \mu\right|}{x_{\frac{n+2}{2}} - \mu} = 0$$

Observe that the median $\left(\frac{x_{\frac{n}{2}} + x_{\frac{n+2}{2}}}{2}\right)$ satisfies the equation above, therefore μ is equal to the median.

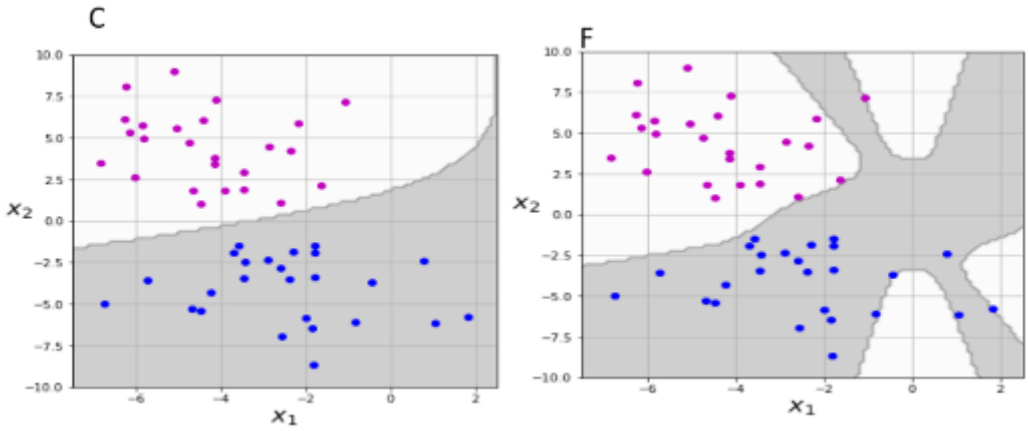
2. SVM (30%)

those 2 following figures uses linear kernel (as it can be seen- the line separating the two classes is linear):



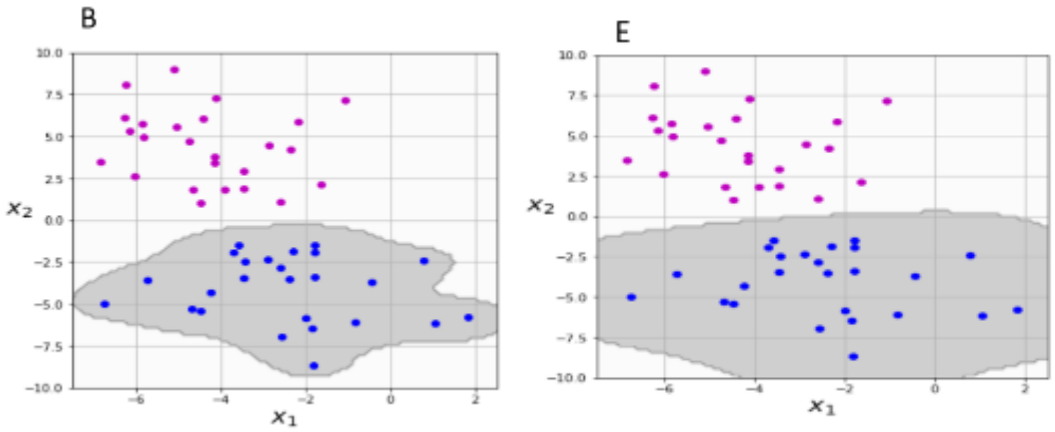
The C parameter tells the SVM optimization how much avoiding misclassifying each training example. For large values of C, the optimization will choose a smaller-margin hyperplane if that hyperplane does a better job of getting all the training points classified correctly. Conversely, a very small value of C will cause the optimizer to look for a larger margin separating hyperplane, even if that hyperplane misclassifies more points. therefore, **A is a linear kernel with C=0.01** and **D is a linear kernel with C=1.**

those 2 following figures uses polynomial kernel (as it can be seen- the line separating the two classes is polynomial):



The larger the degree of the polynom, the greater the danger of reaching to overfitting. In F we can see an example of over fitting. Therefore, **C is 2^{nd} order polynomial kernel** and **F is 10^{nd} order polynomial kernel**.

2 following figures uses RBF kernel (as it can be seen- the shape correlates to the RBF kernel shape):



Higher values of Gamma implies that the influence of a single training example will be close while lower values of Gamma implies that the influence will be much far. The higher the gamma, the greater the danger of reaching overfitting (we have to watch out our model keeps a general behavior since it is prone to adjust too much to the training examples). As it can be seen, B appears to be closer to overfitting. Therefore, **B uses RBF kernel with $\gamma = 1$** and **E uses RBF kernel with $\gamma = 0.2$**.

3. **Capability of generalization**

a. What is the scientific term of the balance that Einstein meant to in machine learning aspect?

Generalization. One of the most important consideration when learning a model is how well it will generalize to new observations. This is called generalization and it refers to how well the concepts learned by a machine learning model will translate to new observations not seen by the model when it was trained. This is related to the concept of overfitting and underfitting. We need to find the model that best fits our data on the one hand, and on the other hand not to be tailored to our specific data- but stay generalize. From the moment you arrive at a suitable model ("but not simpler") do not try to continue to fit the data because it will result in over-fitting ("simple as possible").

b. How does each of the terms $(2p,\ 2\ln(\hat{L}))$in AIC affect the terms of the balance you defined in (a)?

p- the total number of learned parameters. grows as we increase the number of parameters in our model. This penalizes us for building models that are complicated.

ln(L)- the estimated likelihood. Decreases the penalization as the model gets better at explaining our data (our integer increases but causing our result to decrease). this rewards us for building a model that fit out data well.

c. What are the two options that are likely to happen if this balance was violated?

The balance can be violated in the direction of over fitting or in the direction of underfitting. Underfitting and overfitting are not desirable effects and reflect some limitations on our choice made of the hypothesis function. Both overfitting and underfitting will end in bad prediction while overfitting is a result of memorization instead of learning while underfitting is result of lack of information and lack of understanding of the model.

d. What are we aiming for with the AIC? Should it be high or low? Explain

AIC designed to help us strike a balance between models that are complex and models that are good at explaining our data. We may start with a simple model and add parameters to it. As we add more parameters our model will get better at explaining our data, but it will also become more complex. We need to reach the tipping point where the benefit of increasing explanatory power by adding a new parameter are offset by increase in model complexity. We can also start with a complex model and remove parameters so the model will get simpler but also explain our data less well.

When it comes to AIC -**the smaller the better**. we prefer models that minimize AIC (that is why when the AIC is high we define it as penalize).