1) Clustering (10%)

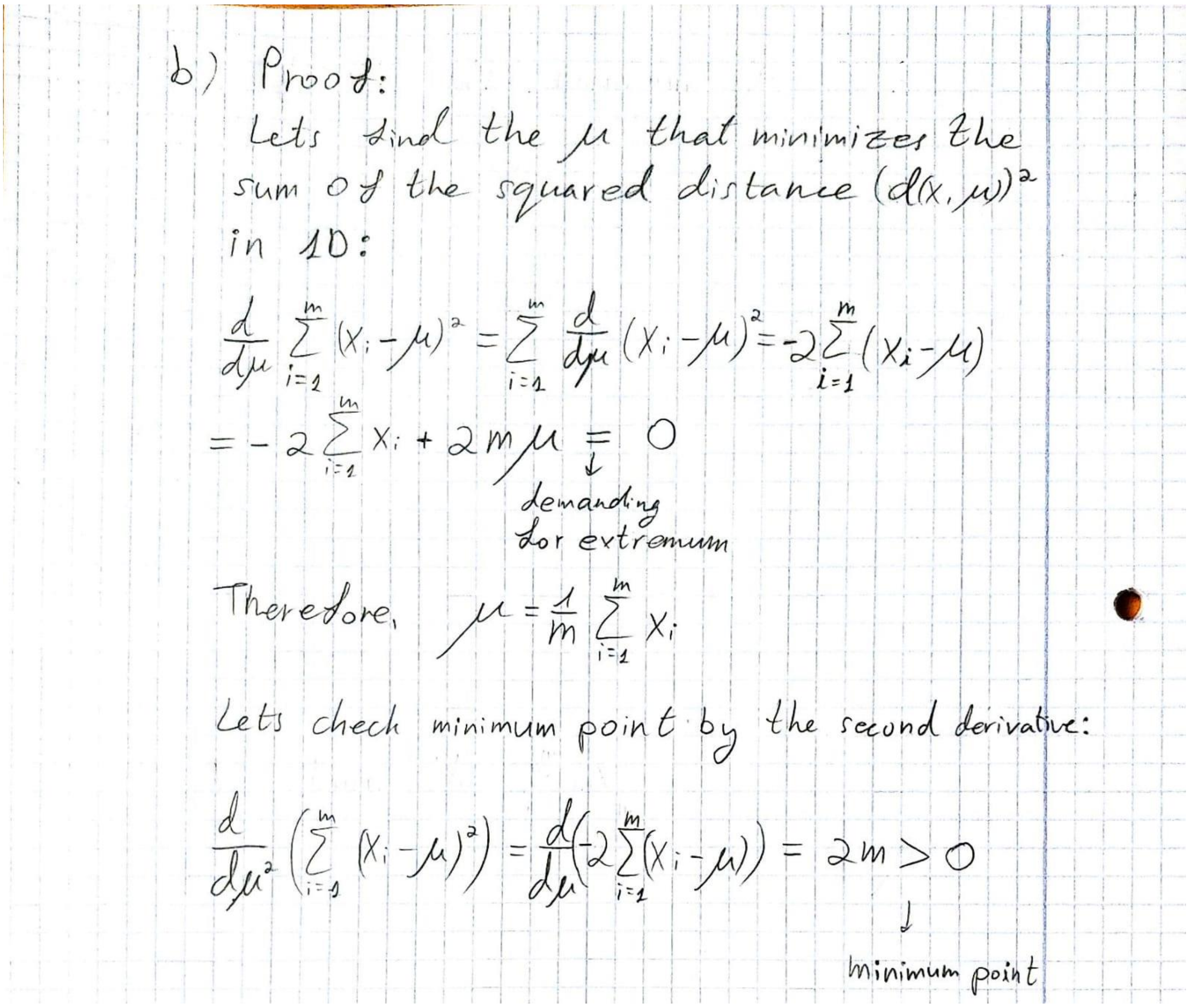
a. Is K-medoid more robust to noise (or outliers) than the K-means algorithm?
Explain your answer.

Answer: Yes.

K-means, based on the squared Euclidean distance, $d(x,\mu)^2$, which defined by the following equation: $\mu_i = \mu(C_i) = argmin_{\mu\in\mathbb{R}^n}\{\sum_{x\in c_i} d(x,\mu)^2\}$, μ_i defines the mean of the centroid of each cluster. The squared distance can be influenced by noises (or outliers) that their distance to the centroid is expected to be bigger than the examples (the main points we would like to cluster).

In contrast to K-mean, K-medoid, based on the differences between the examples, points labeled to be in a cluster and their medoid which this algorithm peaks and changes iteratively according to the L1 distances. Therefore, it is less affected from the noises are mostly far from the points.

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



b) Proof:

Lets find the μ that minimizes the sum of the squared distance $(d(x,\mu))^2$ in 1D:

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

$$= -2\sum_{i=1}^{m}x_i + 2m\mu \underset{\substack{\downarrow \\ \text{demanding} \\ \text{for extremum}}}{=} 0$$

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

Lets check minimum point by the second derivative:

$$\frac{d}{d\mu^2}\left(\sum_{i=1}^{m}(x_i-\mu)^2\right) = \frac{d}{d\mu}\left(-2\sum_{i=1}^{m}(x_i-\mu)\right) = \underset{\substack{\downarrow \\ \text{minimum point}}}{2m > 0}$$

Bonus:

Assume that M Vectors $\bar{X}_1, \bar{X}_2 \ldots, \bar{X}_M \in \mathbb{R}^n$

$$\sum_{i=1}^{m} \|\bar{X}_i - \mu\|^2 = \|X_1 - \mu\|^2 + \|X_2 - \mu\|^2 + \cdots + \|X_N - \mu\|^2 \overset{\textcircled{\star}}{=}$$

$$\circledast \; \|X_1 - \mu\|^2 \underset{\downarrow}{=} \bar{X}_1^T \bar{X}_1 - 2\bar{X}_1^T \bar{\mu} + \bar{\mu}^T \bar{\mu}$$

$$\overset{\textcircled{\star}}{=} \sum_{i=1}^{m} \left(((X_1)_i - \mu_i)^2 + ((X_2)_i - \mu_i)^2 + \cdots + ((X_M)_i - \mu_i)^2 \right) =$$

$$= \sum_{i=1}^{M} \left(M\mu_i^2 - 2\mu_i((X_1)_i + (X_2)_i + \cdots + (X_M)_i) + (X_1)_i^2 + (X_2)_i^2 + \cdots + (X_M)_i^2 \right)$$

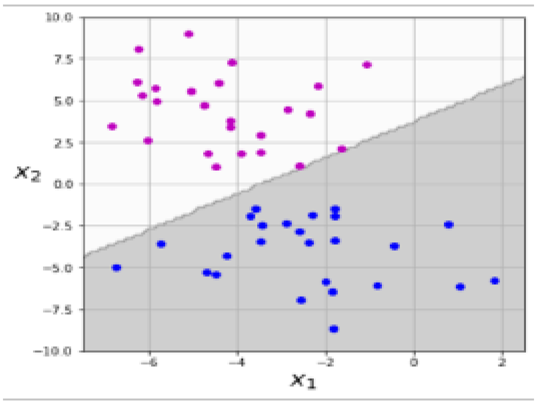
- $(X_j)_i$ is ith element of the vector x_j, $j \in \{1, \ldots, M\}$
- i in the sum is minimized by: $\mu_i = \frac{1}{M}((X_1)_i + \ldots + (X_M)_i)$

As we can see the solution $\mu = \frac{1}{M}(X_1 + \ldots + X_M)$ is the one I proved above.

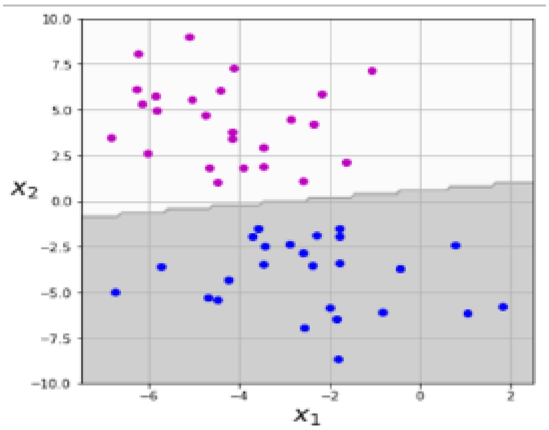
Therefore, we can generalized to m examples.

2) SVM (30%)

A



D



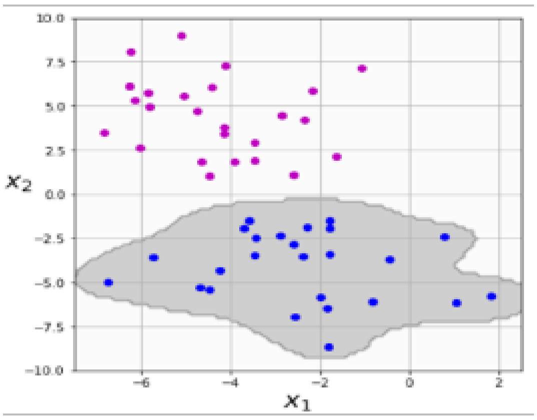
Answer: As we can see from the graphs both are linear separable. However, they are different from one another in the margin length.

According to the optimization problem: $argmin_{w,b}\left\{C\sum_{i=1}^{m}\xi_i+\frac{1}{2}\|w\|^2\right\}$

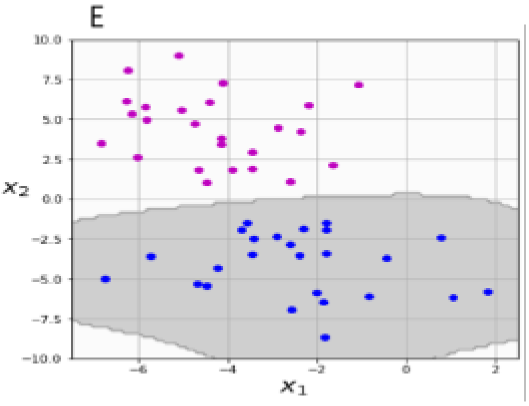
In order to minimize this expression, we have two values that we have to take in account: The first one, C represents the penalty of misclassification and the second one is the margin, which defined as $\frac{2}{||w||}$. Therefore, we have to find a tradeoff between maximizing the margin and minimizing the mistakes. When C is small, misclassification is given less importance and focus more on maximizing the margin, **- option 1**, while, when C is large, the focus is more on avoiding misclassification at the expense of small –**option 2**.

Graph A has higher margin, so smaller C **- option 1.**
Graph D has smaller margin, so higher C **- option 2.**

B



E

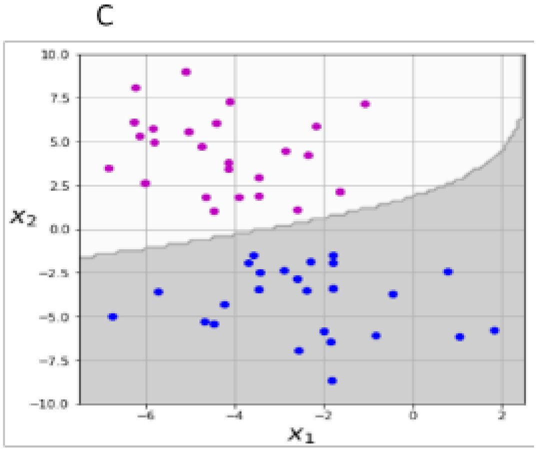


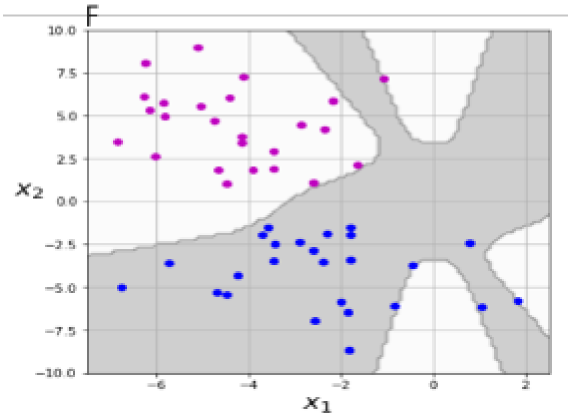
RBF kernel – Radial basis function: $k(\mathbf{x},\mathbf{z})=exp(-||\mathbf{x}-\mathbf{z}||^2/(2\sigma^2))$

$$= \exp(-||x - z\,||^2 \cdot \gamma) \;\rightarrow\; \gamma = \frac{1}{\sigma^2}$$

These graphs have the highest correlation one the other, as we can see from their shapes that that **graph E** is bigger than **B** in by factor gamma. Which makes sense because exponent is a monotonic function.

Gamma is a parameter that represents the fitting to the training data. As the higher gamma is **option 6** the classification fits more to the training.
Therefore, **graph B** fits more to the training set – **option 6**, whereas Graph E has less fitting to the data **- option 5.**

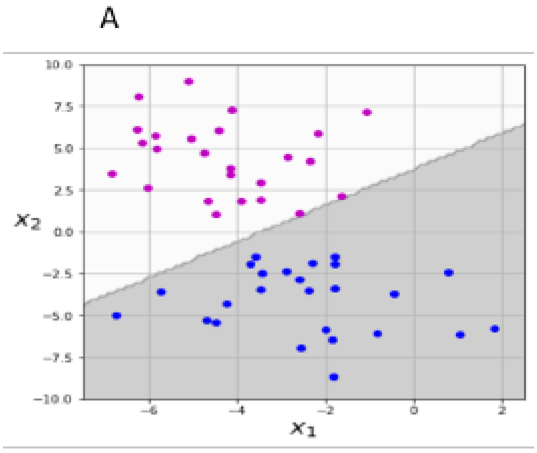




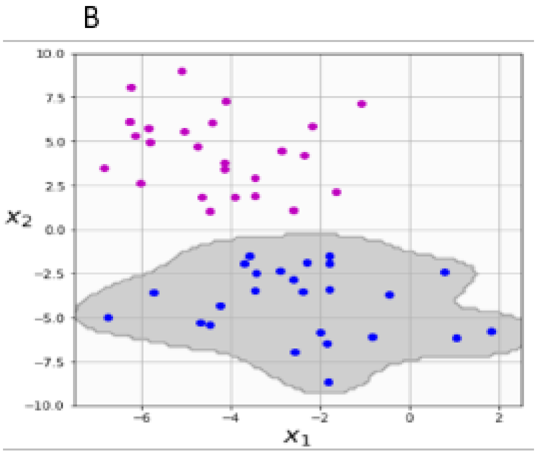
The two left options are the graphs above, so they have polynomial kernels.

As we can see the **graph C** fits less to the training than **graph F.**
As high the polynomial's order is the complexity of the problem is higher and the graph fits more to the training points – **option 4,** on the contrary to lower polynomial's order – **option 3**

Summary of my matchings:

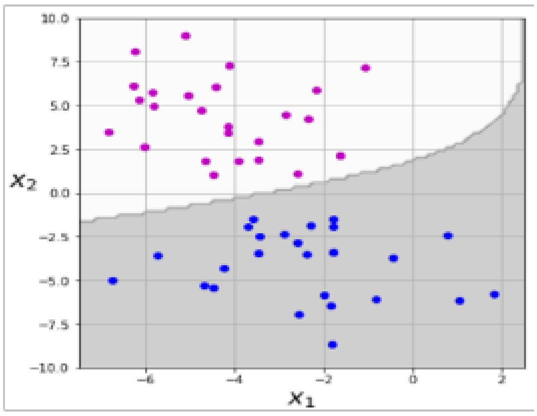


1. Linear kernel with $C = 0.01$.



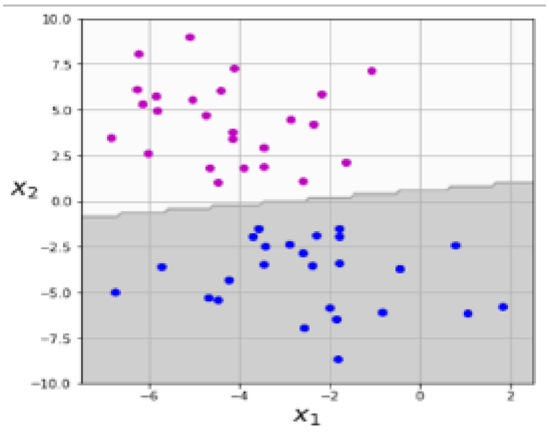
6. RBF kernel with $\gamma = 1$.

C



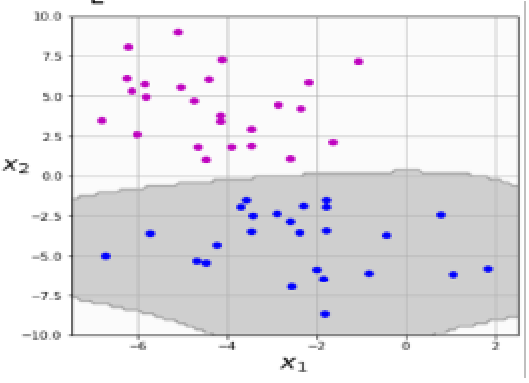
3. 2^{nd} order polynomial kernel.

D



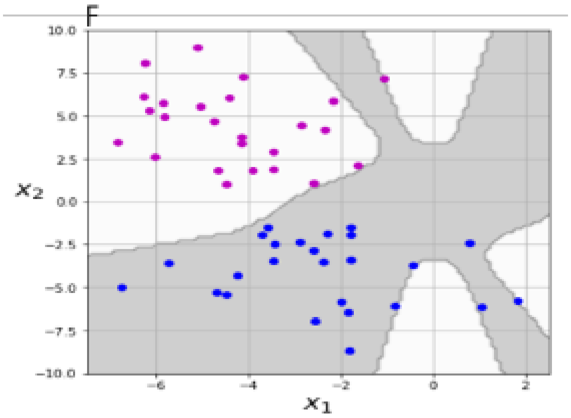
2. Linear kernel with $C = 1$.

E



5. RBF kernel with $\gamma = 0.2$.

F



4. 10^{th} order polynomial kernel.

3) Capability of generalization (20%)

Einstein saying: "Everything should be made as simple as possible but not simpler".

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

Answer: The scientific term of the balance that Einstein meant to in machine learning aspect is "generalization". It means the goodness of model to be fitted to new observation that it hasn't seen before – learning part. If the learned hypothesis fits the training set too much, it happens when the complexity of the model is high, unlike Einstein saying "it should be made as simple as possible", and leads to overfitting / high variance which means bad generalization. On the other hand, if our model will be simplest we miss the goodness-of-fit between the hypothesis representation and the training set – underfitting / high bias.

In order to maintain the balance, we use regularization technics for finding a trade-off between simple hypothesis – low complexity and goodness-of-fit.

b. How does each of the terms (2p, 2ln(^ L)) in AIC affect the terms of the balance you defined in (a)?

→ $AIC = 2\text{p} - 2\ln(\text{^ L})$

Answer: 2*p - P is the total numbers of parameters, as much as p is smaller the model is easier to analyze, low complexity rather than high-dimensional model with large numbers of parameters which can lead to overfitting. When p is small AIC is small as well, the dimension of the model is small which means high bias and low variance, while high p means high variance and small bias. However, if the p is too small the model can face with misclassification.
The likelihood ^L indicates the goodness fit of a class to the model, 2ln(^ L) is a number is the probability that estimates p. Therefore, if ^ L is high, according to the equation, AIC decreases.
The balance that I defined in (a) can be represented by a trade-off between bias and variance, (P) in resulting estimators (^L)

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

Answer:

1) If we increase the number of parameters p, it can lead to overfitting, and add complexity to the classification that takes a more time and money.
2) If we decrease the number of parameters p, it can lead to underfitting, that means we won't have enough parameters for a good classification.

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

Answer: AIC is a mathematical method which evaluates how well a model fits the data. The best model is the one that has minimum AIC among all the other ones.
The best-fit model according to AIC is the one that has the greatest amount of variation that means highest likelihood (^L) using the fewest possible parameters (p) which indicate also the parsimony of the model.