HW 3 – Machine learning in healthcare

Hadas Ben-Atya, 316364470

1. Clustering:

a. K-Medoids is more robust than the K-Means.

In K-means we minimize the sum of squared Euclidian distances between the data and the cluster's centroid. In other words, K mean clustering the data based on their closeness to each other.

The sum of squared of the Euclidian distance is : $D(point_i, C_j) = \sqrt{(point_i - C_j)^2}$

we will calculate this distance for each point in our data and each centroid from our clusters.

After one epoch, we will calculate the centroid for each new cluster and repeat the algorithm until the centroids do change more than our threshold.

This method is highly sensitive to noise and outliers because the cluster's centroid is calculated by the mean of all the cluster's points: $C_j = \frac{1}{N_j}\sum_{i=1}^{N} P_i$.

we know from previous lectures that the mean is not a robust metric in manner of outliers and noise.

Another drawback of the K-meaning is that for different initial centroids, we may converge to different clusters.

In addition, the K-means cannot handle different size or density clusters, which may affect the accuracy of the result.

K-Medoids is used to find set of cluster representatives and then assign other points to them based on the L1 distance, which measure the dissimilarities of the point P_i with all the other points in the data:

$$D = \sum_{C_j} \sum_{P_i \in C_j} |P_i - C_j|$$

Thus, the K-Medoids tries to minimize the sum of dissimilarities of the data points.

This method reduces noise and outliers and it more robust than the K-means.

the mainly drawback of the K-Medoids is the high complexity compared to the K-means.

b. For the 1D case of K-means, we are minimizing the $\sum_{i=1}^{m}(x_i-\mu)^2$.

$$\sum_{i=1}^{m}(x_i-\mu)^2=\sum_{i=1}^{m}(x_i-\bar{x}+\bar{x}-\mu)^2 \underbrace{=}_{\substack{(a+b)^2=a^2+2ab+b^2\\ a=x_i-\bar{x},b=\bar{x}-\mu}} \sum_{i=1}^{m}(x_i-\bar{x})^2 + 2\sum_{i=1}^{m}(x_i-\bar{x})(\bar{x}-\mu)+\sum_{i=1}^{m}(\bar{x}-\mu)^2$$

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

notice that

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

we got that $\sum_{i=1}^{m}(x_i-\mu)^2=\sum_{i=1}^{m}(x_i-\bar{x})^2+m(\bar{x}-\mu)^2$
we will get minimum only if $\bar{x}=\mu$, which means that the centroid (μ) equivalent to the mean of the m examples ($\overline{x}$).

c. The centroid which minimizes the term : $\sum_{i=1}^{m}|x_i-\mu|$:

$for\ x_1<x_2<\cdots<x_n$,

$$if\ \mu<x_1 \rightarrow \sum_{i=1}^{m}|x_i-\mu|=\sum_{i=1}^{m}x_i-\mu$$

we can notice that as the value of μ increased, the value of $\sum_{i=1}^{m}x_i-\mu$ decrease until $\mu>x_1$.

$$f(\mu)=\sum_{i=1}^{m}x_i-\mu\ <f(x_1)=\sum_{i=1}^{m}x_i-x_1\ \ \forall\mu<x_1$$

let: $x_k\le\mu\le\mu+d\le x_{k+1};\quad d\in\mathcal{R}\ ,1\le k\le m$

$$f(\mu+d)=\sum_{i=1}^{m}|x_i-(\mu+d)|=\sum_{i=1}^{k}|x_i-(\mu+d)|+\sum_{i=k}^{m}|x_i-(\mu+d)|$$

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

$$=d*k+\sum_{i=1}^{k}\mu-x_i+d*(m-k)+\sum_{i=k}^{m}x_i-\mu$$

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

$$=d(2k-m)+\sum_{i=k}^{m}|x_i-\mu|=d(2k-m)+f(\mu)$$

$$\Rightarrow d(2k-m) = f(\mu+d) - f(\mu)$$
$$\rightarrow (2k-m) = \frac{f(\mu+d)-f(\mu)}{d} \xrightarrow{d\to 0} f'(\mu)$$

$$\Rightarrow \begin{cases} for\ k < \frac{m}{2} \rightarrow\ f'(\mu) < 0 \\ for\ k > \frac{m}{2} \rightarrow f'(\mu) > 0 \\ for\ k = \frac{m}{2} \rightarrow f'(\mu) = 0 \end{cases}$$

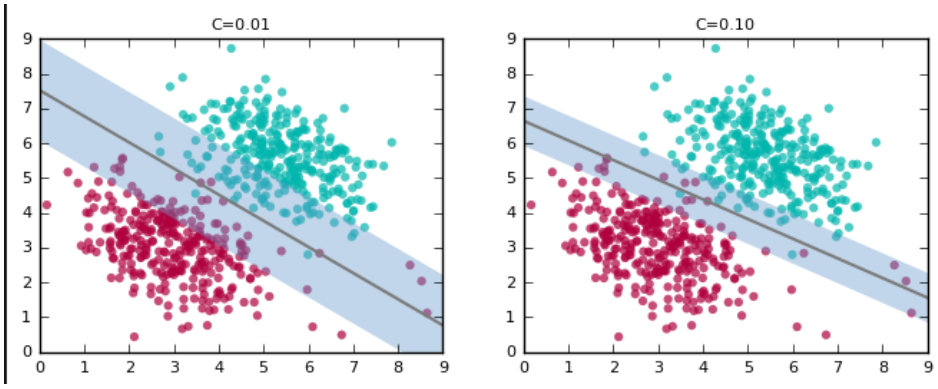
Thus, the minimum of $\sum_{i=1}^{m}|x_i - \mu|$ obtained when $k = \frac{m}{2}$ which means that the value that minimized the function is a value such that half of the population is lower them it, and half higher – which fits to the definition of the median.
note that the median is not necessarily an item within the population.

2. SVN:

In Support vector machine, different kernels and parameters will affect our results.

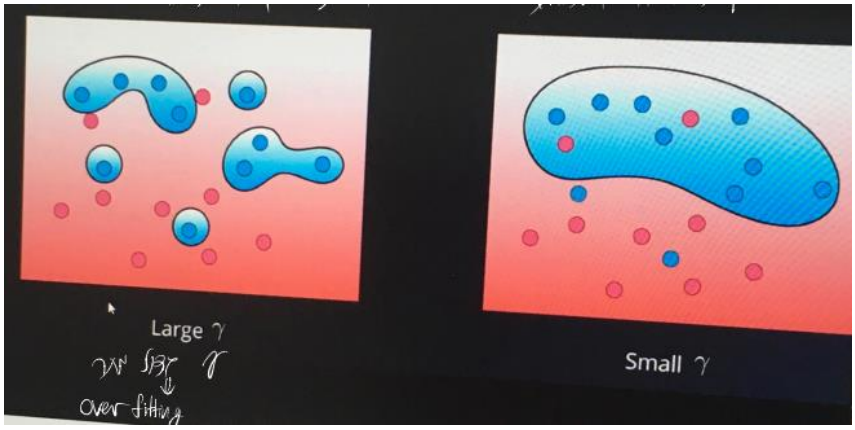
The kernel influences:
Linear kernel will allow us to separate our data by drawing a straight line, while non-linear kernels will transform non-linear spaces into linear spaces by transforming the data into another dimension. When using non-linear kernel, we will get non-linear separating space and when one plot the separating data we will be able to see the kernel shape.

The C influences:
this hyper-parameter controls the trade-off between maximizing the margin and decreasing the number of misclassified samples.
for larger C, we will get over-fitting to our data, the number of misclassified points will decrease, and the margin will be smaller.



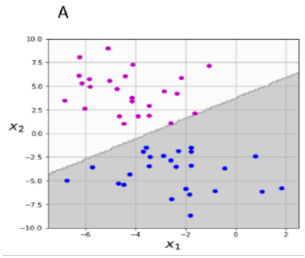
The γ influences:
γ is the Gaussian RBF parameter, it defines how far the influences of a single training example reaches, when low values means "far" and large values meaning "close".

Thus, a larger γ will create smaller radius of influences of the samples selected and will increase the risk of over fitting.



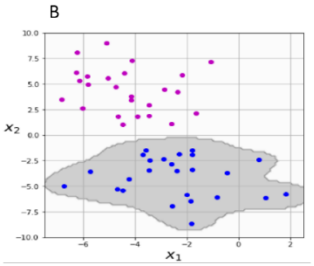
A match to 2 (linear kernel with C=1).

First, we can recognize a linear kernel by the straight line separated our data.
in addition, we can see that the separating line is very tied to the samples which implies for small margin and large C.

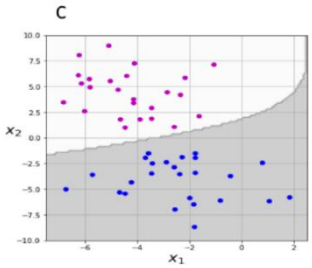


B match to 6 (RBF kernel with $\gamma = 1$)

We can notice the RBF kernel which looks like sphere shape.
we have a high fitting to the blue data thus we can conclude it fits to a higher γ .

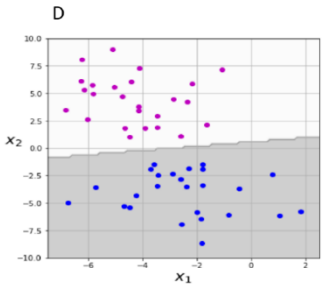


C match to 3 (2nd order polynomial kernel)
we can see that the separating line is from a shape of 2nd polynomial order.



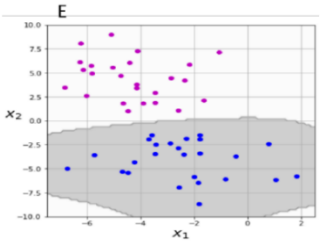
D match to 1 (linear kernel with C=0.01)

We can recognize a linear kernel by the straight line separated our data.
Also, we can see that the separating line is far from the samples which implies for large margin and small C.



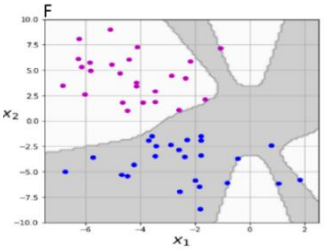
E match to 5 (RBF kernel with $\gamma = 0.2$)

We can see the characteristic shape of the RBF kernel.
the kernel is less tied to the samples which implies for small γ .



F match to 4 (10th order polynomial kernel)

The kernel shape fits to a high order polynomial kernel.



3. Capability of generalization :

a. Einstein's concept is – " Everything should be made as simple as possible but not simpler ". This concept is a key concept in machine learning and it refers to the Generalization concept. As we learned in the lecture, a simpler hypothesis representation is less prone to overfitting, which we always try to avoid of.

Thus, an appropriate generalized model will help us to achieve the balance between the effective complexity of the model and avoiding over/under fitting to our training data. If we will use a high complexity model, we will risk with overfitting to our training date, but if we will use too simple model, we might get under-fitting model to our training data.

b. The AIC is composed of two terms: $AIC = \ 2p - 2\ln(\hat{L})$

p = the total number of learned parameters. as we mentioned earlier, the higher number of learned parameters the more complexity our model and we will risk in over-fitting. But too small amount of learning parameters will create an under-fitting model.
thus, we would like to optimize p to get the suitable and a proper generalized model, by minimize the amount of the learned parameters.

$\hat{L}$ = the estimated likelihood given these parameters, which means how well the model reproduces the data – the likelihood that the model could have produced our observed values. Thus, we will try to maximize the model reproduce the data ($\hat{L}$) by minimize the inverse of the likelihood = $-2\ln(\hat{L}) = \frac{2}{\ln(\hat{L})}$

The best model according to the AIC is the model that explains the greatest amount of variation (minimize the $\{-\ln(\hat{L})\}$) using the fewest possible independent variables (minimize p).
This method is an accurate "translation" of the Einstein's statement.

c. The two options that are likely to happen if this balance was violated are over-fitting and under-fitting.
if we use a model with too high complexity, that in machine learning refers to a larger number of learned parameters, we might get over-fitting to our training date, which corresponds to high statistical variance.
In contrast, if we will learn too few parameters, we will get low complexity model and under-fitting model which corresponds to high statistical bias.

d. We are aiming to a lower AIC. If the AIC is lower, it means that we used as fewer number of learned parameters as possible – which indicates the simplicity of the model, with as greater likelihood as possible, which indicates the goodness of fit of the model.

[1] “Akaike Information Criterion | When & How to Use It,” *Scribbr*, Mar. 26, 2020. https://www.scribbr.com/statistics/akaike-information-criterion/ (accessed Jan. 14, 2021).

[2] P. Arora, Deepali, and S. Varshney, “Analysis of K-Means and K-Medoids Algorithm For Big Data,” *Procedia Computer Science*, vol. 78, pp. 507–512, 2016, doi: 10.1016/j.procs.2016.02.095.

[3] John A. Bullinaria, 2004 , "Improving Generalization Introduction to Neural Networks : Lecture 10" https://www.cs.bham.ac.uk/~jxb/NN/l10.pdf

[4] https://math.stackexchange.com/questions/113270/the-median-minimizes-the-sum-of-absolute-deviations-the-ell-1-norm