Homework 5 - Machine Learning from Data

Students: Ben Kapner ID311146021, Micaela Singer ID807724

1. Kernels and mapping functions

a. Consider two kernels K_1 and K_2, with the mapping φ_1 and φ_2 respectively. Show that $K = 5K_1 + 4K_2$ is also a kernel and find its corresponding φ.

To prove that K is also a kernel, we must find a mapping function and show that the inner product after applying the mapping equals to the kernel output. Consider the following mapping function:

$$\varphi = [\sqrt{5}\varphi_1, \sqrt{4}\varphi_2]$$

Where φ is the concatenation of $\sqrt{5}\varphi_1$ and $\sqrt{4}\varphi_2$.

Since we know that K_1 and K_2 are kernels with mappings φ_1 and φ_2 we can write K_1 and K_2 for every x, y as:

$$K_1(x, y) = \varphi_1(x) \cdot \varphi_1(y)$$

$$K_2(x, y) = \varphi_2(x) \cdot \varphi_2(y)$$

For K to be a valid kernel, we must prove that for every x, y:

$$K(x, y) = \varphi(x) \cdot \varphi(y)$$

If K is a kernel, then for two instances x, y we can express it as:

$$K(x, y) = 5K_1(x, y) + 4K_2(x, y)$$

$$K(x, y) = 5(\varphi_1(x) \cdot \varphi_1(y)) + 4(\varphi_2(x) \cdot \varphi_2(y))$$

$$K(x, y) = \sqrt{5}\varphi_1(x) \cdot \sqrt{5}\varphi_1(y) + \sqrt{4}\varphi_2(x) \cdot \sqrt{4}\varphi_2(y)$$

On the other hand, we have that:

$$\varphi(x) \cdot \varphi(y) = \sqrt{5}\varphi_1(x) \cdot \sqrt{5}\varphi_1(y) + \sqrt{4}\varphi_2(x) \cdot \sqrt{4}\varphi_2(y)$$

Hence,

$$K(x, y) = \varphi(x) \cdot \varphi(y)$$

Therefore K is a valid kernel with mapping $\varphi = [\sqrt{5}\varphi_1, \sqrt{4}\varphi_2]$.

b. Consider a kernel K_1 and its corresponding mapping φ_1 that maps from the lower space R^n to a higher space R^m (m>n). We know that the data in the higher space R^m, is separable by a linear classifier with the weights vector w. Given a different kernel K_2 and its corresponding mapping φ_2, we create a kernel $K = 5K_1 + 4K_2$. Can you find a linear classifier in the higher space to which φ, the mapping corresponding to the kernel K, is mapping? If yes, find the linear classifier weight vector. If no, prove why not.

It is possible to find a linear classifier in the higher space to which φ is mapping.

If there is a linear classifier in the higher space to which φ is mapping, then it follows the decision function z:

$$z(x) = sgn(x \cdot w)$$

Having w be the weights vector.

When applying the z formula for φ we have:

$$z(x) = sgn(\varphi(x) \cdot w)$$

Recall that from the previous section we established the mapping for K:

$$\varphi = [\sqrt{5}\varphi_1, \sqrt{4}\varphi_2]$$

Hence,

$$z(x) = sgn([\sqrt{5}\varphi_1, \sqrt{4}\varphi_2] \cdot w)$$

Since we are performing an inner product, z can also be written as:

$$z(x) = sgn\left(\left(\left[\frac{1}{\sqrt{5}}w, \vec{0}\right] \cdot [\sqrt{5}\varphi_1, \sqrt{4}\varphi_2]\right) + \left(\left[\vec{0}, \frac{1}{\sqrt{4}}w\right] \cdot [\sqrt{5}\varphi_1, \sqrt{4}\varphi_2]\right)\right)$$

When performing the inner products, because of the zero vector the previous equation will be simplified to:

$$z(x) = sgn\left(\frac{1}{\sqrt{5}}w\sqrt{5}\varphi_1 + \frac{1}{\sqrt{4}}w\sqrt{4}\varphi_2\right)$$

$$z(x) = sgn((\varphi_1 + \varphi_2) \cdot w)$$

c. Consider the space $S = \{1, 2, \ldots N\}$ for some finite N (each instance in the space is a 1-dimension vector and the possible values are 1, 2,…,N) and the function $K(x, y) = 9 \cdot f(x, y)$ for every $x, y \in S$. Prove that K is a valid kernel by finding a mapping φ such that:

$$\boldsymbol{\varphi(x) \cdot \varphi(y) = 9min(x, y) = K(x, y)}$$

For example, if the instances are $x = 4, y = 8$, for some $N \geq 8$, then:

$$\boldsymbol{\varphi(x) \cdot \varphi(y) = \varphi(4) \cdot \varphi(8) = 9 \cdot min(4, 8) = 36}$$

The idea is to prove that $K(x, y) = 9 \cdot f(x, y)$ is valid when $f(x, y) = min(x, y)$ by finding a mapping φ. The inner product $\varphi(x) \cdot \varphi(y)$ must be the sum of the pairwise product of the components $\varphi(x)$ and $\varphi(y)$ such that $\varphi(x) \cdot \varphi(y) = min(x, y)$. Hence, we analyze two cases:

- $x < y \Rightarrow K(x, y) = 9 \cdot x$

Consider the vectors:

- $\varphi(x) = [3,3,3, \ldots ,3,0,0, \ldots 0]$ of length N such that its first x entries are equal to 3 and the rest are zeros.
- $\varphi(y) = [3,3,3, \ldots ,3,0,0, \ldots 0]$ of length N such that its first y entries are equal to 3 and the rest are zeros.

When performing the inner product between $\varphi(x)$ and $\varphi(y)$, the product of the first x entries will have the value 9 and the rest will be zeros. Hence, $\varphi(x) \cdot \varphi(y) = 9 \cdot x = K(x, y)$.

- $x > y \Rightarrow\ K(x, y) =\ 9 \cdot y$

Similarly, we consider the same two vectors, however this time the inner product between $\varphi(x)$ and $\varphi(y)$ we obtain the product of the first y entries (will have the value 9) and the rest will be zeros. Hence, $\varphi(x) \cdot \varphi(y) = 9 \cdot y = K(x, y)$.

Therefore, K is a valid kernel with the following mapping:

$$\varphi(j) = [3,3,3, \dots ,3,0,0, \dots ,0]$$

Where $\varphi(j)$ is of length N and the first j entries are 3 and the rest are zeros.

2. Lagrange multipliers

The revenue is modeled by:

$$R(h, s) = 200 \times h^{\frac{2}{3}} \times s^{\frac{1}{3}}$$

Where:

- h represents hours of label ($20 per hour)
- s represents the steel ($170 per ton)

We have a budget of $20,000, so our total cost equation is as follows:

$$20{,}000 = 20h + 170s$$

Hence, we have:

$$\mathcal{L} = 200 \times h^{\frac{2}{3}} \times s^{\frac{1}{3}} + \lambda[20{,}000 - 20h - 170s]$$

And proceed to calculate the partial derivatives of $\mathcal{L}$ with respect to h and c and make it equal to zero.

$$\frac{\partial \mathcal{L}}{\partial h} = 200 \times 2 \times h^{\frac{2}{3}-1} \times s^{\frac{1}{3}} - 20\lambda = 400 \times h^{\frac{-1}{3}} \times s^{\frac{1}{3}} - 20\lambda = 0$$

$$\frac{\partial \mathcal{L}}{\partial s} = 200 \times 1 \times h^{\frac{2}{3}} \times s^{\frac{1}{3}-1} - 170s = 200 \times h^{\frac{2}{3}} \times s^{\frac{-2}{3}} - 170s = 0$$

The constraint will be determined by the partial derivative of $\mathcal{L}$ with respect to λ and make it equal to zero:

$$\frac{\partial \mathcal{L}}{\partial \lambda} = 20{,}000 - 20h - 170s = 0$$

Next, we solve each of the first two equations for λ. From $\frac{\partial \mathcal{L}}{\partial h}$ we obtain:

$$\lambda = \frac{400 \times h^{\frac{-1}{3}} \times s^{\frac{1}{3}}}{20}$$

From $\frac{\partial \mathcal{L}}{\partial s}$ we obtain:

$$\lambda = \frac{200 \times h^{\frac{2}{3}} \times s^{\frac{-2}{3}}}{170}$$

Set both λ equations equal:

$$\frac{400 \times h^{\frac{-1}{3}} \times s^{\frac{1}{3}}}{20} = \frac{200 \times h^{\frac{2}{3}} \times s^{\frac{-2}{3}}}{170}$$

$$400 \times h^{\frac{-1}{3}} \times s^{\frac{1}{3}} \times 170 = 20 \times 200 \times h^{\frac{2}{3}} \times s^{\frac{-2}{3}}$$

$$68{,}000 \times s^{\frac{1}{3}} \times s^{\frac{2}{3}} = 4{,}000 \times h^{\frac{2}{3}} \times h^{\frac{1}{3}}$$

$$68{,}000s = 4{,}000h$$

$$17s = h$$

Use this result in the cost constraint equation:

$$20{,}000 = 20 \times 17s + 170s$$

$$20{,}000 = 510s$$

$$s = 39.2157$$

Using this result to obtain h:

$$h = 17 \times 39.2157 = 666.6667$$

Hence, the maximum revenue is:

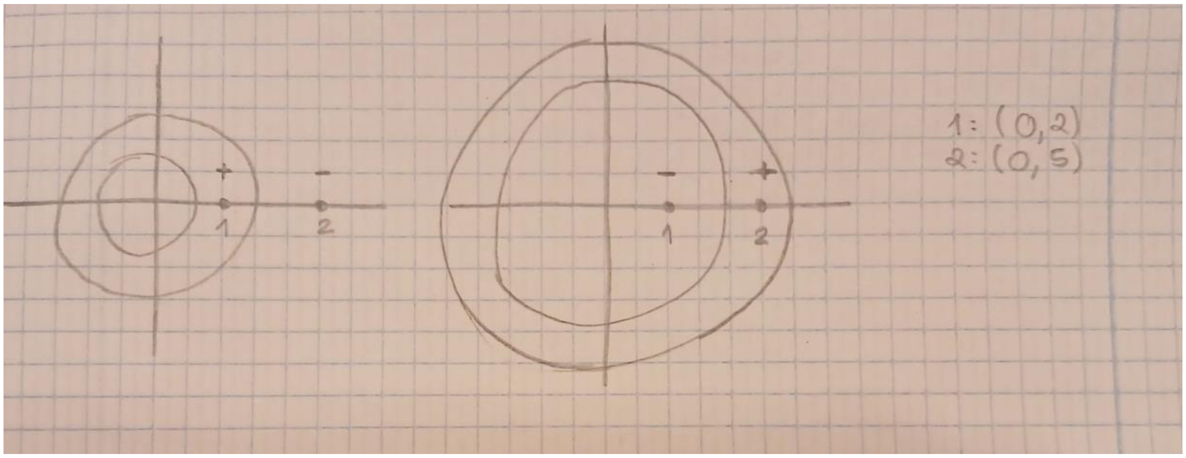
$$R(h, s) = 200 \times h^{\frac{2}{3}} \times s^{\frac{1}{3}}$$

$$R(h, s) = 200 \times (666.6667)^{\frac{2}{3}} \times (39.2157)^{\frac{1}{3}} = 51{,}854.8236$$

3. PAC learning and VC Dimension

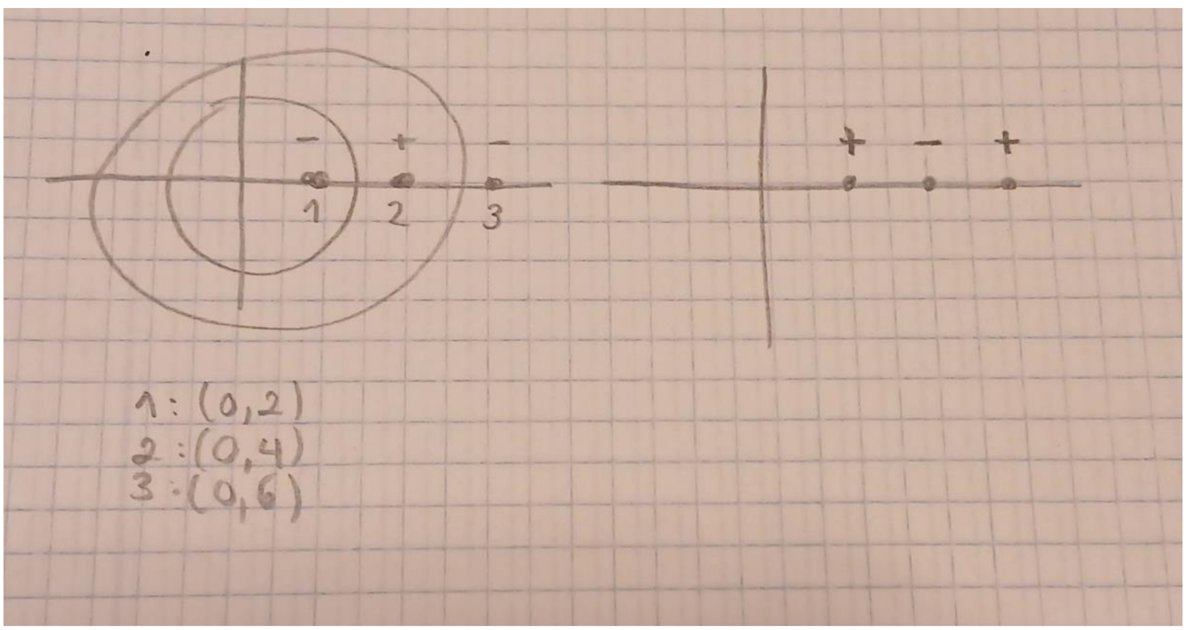
a. What is the $VC(H)$? Prove your answer.

For 2 points, it is possible to assign the labels to each point in such a way that the labels are correctly separated by region:



Therefore, $VC(H) \geq 2$.

However, for 3 points, it is possible to assign the labels in such a way that an origin-centered ring cannot shatter them. Consider three points with coordinates (0,2), (0,4) and (0,6) respectively. When assigning positive labels to points 1 and 3 and negative label to point 2 (in the middle) they won't be shattered by origin-centered rings.



Hence, $VC(H) < 3$ and since $VC(H) \geq 2$ we can conclude that $VC(H) = 2$.

b. Describe a polynomial sample complexity algorithm L that learns C using H. State the time complexity of your suggested algorithm. Prove all your steps.

"C is PAC learnable by L using H if and only if learner L will, with probability $1-\delta$ output a hypothesis $h \in H$ such that $error_D \leq \varepsilon$ in time and samples polynomial in $\frac{1}{\varepsilon}, \frac{1}{\delta}$ and n."

<u>Algorithm:</u> the most specific hypothesis. L fits a hypothesis from H to the training set by choosing:

- r_2 to be the distance between the origin and the furthest away data point in the dataset with positive label (meaning that such point belongs to the concept C). This refers to the outer circle that composes the ring.
- r_1 to be the distance between the origin and the closest data point in the dataset with label positive label (meaning that such point belongs to the concept C). This refers to the inner circle that composes the ring.

Consistent learner

L is a consistent learner because all positive points in the dataset are inside of H, and all negative points are outside of H. Therefore, L is a consistent with the concept C.

Time complexity

Every point is visited once, so its time complexity is polynomial.

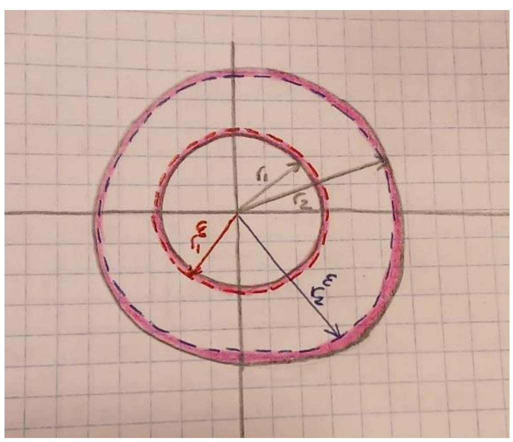
Sample complexity

We want to prove that given the desired parameters ε and δ, the number of training samples m that is required to guarantee the desired error and confidence, is polynomial in $\frac{1}{\varepsilon} > 0$, and $\frac{1}{\delta} > 0$. In other words, we are interested in an m large enough so that this will be true:

$$\pi(error_{\pi}(L(D), C) > \varepsilon) < \delta$$

Let $\varepsilon > 0$ and $\delta > 0$.

Consider the circle c_1^{ε} of radius $r_1^{\varepsilon} > r_1$ (recall that r_1 refers to the radius of the inner circle that composes the ring) and the circle c_2^{ε} of radius $r_2^{\varepsilon} < r_2$ (recall that r_2 refers to the radius of the outer circle that composes the ring). The following figure shows the different circles and radiuses:



The error is contained in the shaded annuluses A_1 and A_2 in the previous figure, such that both A_1 and A_2 will sum to ε. Furthermore, the probability of all my positive samples not visiting an annulus is at most $\left(1-\frac{\varepsilon}{2}\right)^m$. The probability of all my positive samples not visiting any of the 2 annuluses is at most $2\left(1-\frac{\varepsilon}{2}\right)^m$.

We want to find the number of samples m such that:

$$P(error \geq \varepsilon) \leq 2\left(1-\frac{\varepsilon}{2}\right)^m$$

By Taylor series we know that we can upper bound the previous expression:

$$P(error \geq \varepsilon) \leq 2\left(1-\frac{\varepsilon}{2}\right)^m \leq 2exp\left(\frac{-m\varepsilon}{2}\right)$$

We want the previous expression to be smaller than δ:

$$2exp\left(\frac{-m\varepsilon}{2}\right) \leq \delta$$

$$exp\left(\frac{-m\varepsilon}{2}\right) \leq \frac{\delta}{2}$$

$$\frac{-m\varepsilon}{2} \leq ln\left(\frac{\delta}{2}\right)$$

$$-m \leq \frac{2}{\varepsilon} ln\left(\frac{\delta}{2}\right)$$

$$m > \frac{2}{\varepsilon} ln\left(\frac{2}{\delta}\right)$$

c. **You want to get 95% confidence a hypothesis with at most 5% error. Calculate the sample complexity with the bound that you found in b and the above bound for infinite $|H|$. In which one did you get a smaller m? Explain.**

We have, $\varepsilon = 0.05$ and $\delta = 1 - 0.95 = 0.05$.

- From the previous section we obtained a bound for m:

$$m > \frac{2}{\varepsilon} ln\left(\frac{2}{\delta}\right)$$

$$m > \frac{2}{0.05} ln\left(\frac{2}{0.05}\right)$$

$$m > 147.555$$

Hence, $m > 148$.

- For an infinite $|H|$:

$$m \geq \frac{1}{\varepsilon}\left(4log_2\left(\frac{2}{\delta}\right) + 8VC(H)log_2\left(\frac{13}{\varepsilon}\right)\right)$$

$$m \geq \frac{1}{0.05}\left(4log_2\left(\frac{2}{0.05}\right) + 8 \times 2 \times log_2\left(\frac{13}{0.05}\right)\right)$$

$$m \geq 2992.91$$

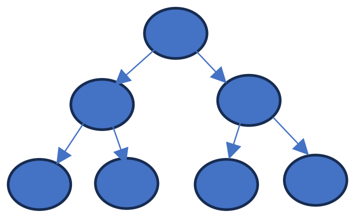
Hence, $m \geq 2993$.

We obtain a smaller number of samples when calculating the amount according to the geometry method. The result is expected since the geometry method returns a tighter bound whereas the method using VC dimension returns a more lose bound for a more general case.

4. VC Dimension

a. What is the $VC(H_3)$? Prove your answer.

For $VC(H_3)$, $m = 3$ and since $n \leq m$, the maximum number n can take is 3. We therefore obtain a binary decision tree of $x = 2^n - 1 = 2^3 - 1 = 7$, which includes leaf nodes. The representation of a binary decision tree with 7 nodes in total is:



We obtain a total of 4 leaf nodes, which represent 4 decision boundaries, so $VC(H_3) \geq 4$. Now, to prove that $VC(H_3) = 4$ we need to prove that $VC(H_3) < 5$. Since we only have 4 leaf nodes, the 5th analyzed point would have to fall in one leaf node that already has 1 point. Since not all 5 points can be separated individually with the 4 decision boundaries, then they cannot shatter H_3. Taking for example that the point that was already classified by the leaf node had a label +1 and that the 5th point that arrived at the same leaf node had a label -1. Hence, $VC(H_3) = 4$ because any set of size 4 is shattered and any set of size 5 is not shattered.

b. What is the $VC(H_m)$? Prove your answer.

For $VC(H_m)$, since $n \leq m$, the maximum number n can take is m. The decision tree would hence have a total of $x = 2^m - 1$ nodes. On the other hand, the total number of leaf nodes would be 2^{m-1} which represent the decision boundaries. Following the same logic as in the previous question we have that:

- $VC(H_m) \geq 2^{m-1}$

With 2^{m-1} decision boundaries, 2^{m-1} can be separated individually and shatter H_m.

- $VC(H_m) < 2^m$

If we have 2^m different points with only 2^{m-1} decision boundaries represented by the leaf nodes of the binary decision tree, the last point (point number m) will fall into a leaf node that already has one point on it. Therefore, there will always exist some dichotomy of the 2^m points that cannot be achieved because all 2^m points cannot be separated individually with 2^{m-1} leaf nodes. Indeed, in a binary decision tree, the decision boundaries aren't independent. When tracing from the root to a leaf, every step in the path is a decision based on a certain condition. This shows how the structure of the binary decision tree limits its ability to realize all possible dichotomies of a set of points when the number of points is more than the number of leaf nodes in the tree.

Hence, $VC(H_m) = 2^{m-1}$