

# Solutions to Sheet 1

Friedrich May, 355487; Markus Moll, 406263; Mariem Mounir, 415862

April 29, 2020

## Exercise 1

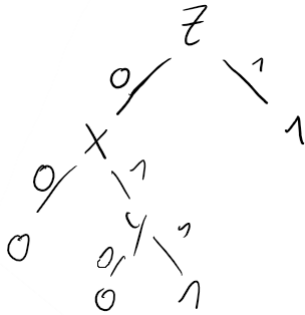
To calculate the class for every point we first need to find the  $k$  closest points (depending on the norm). To archive this, a script like the following can be used:

```
1 function [ nbs ] = knn( spc,pnt,k, norm_p )
2 %KNN Summary of this function goes here
3 % Detailed explanation goes here
4 nbs = [];
5 nonb = 0;
6 while nonb < k
7     if nonb == 0
8         candidates = spc;
9     else
10        candidates = transpose(setdiff(transpose(spc),transpose(nbs),'rows
    ↪ '));
11    end
12    closestp = [0,0,0];
13    closestn = inf;
14    for i = 1:length(candidates)
15        d =transpose(pnt-candidates(:,i));
16        n=norm(d,norm_p);
17        if n < closestn
18            closestn = n;
19            closestp = candidates(:,i);
20        end
21    end
22    nonb = nonb+1;
23    nbs(:,nonb) = closestp;
24 end
25 end
```

Using the results the following table can be created. Note that the ties can be arbitrarily chosen.

Norm	Euclidian		Manhattan	
k	2	3	2	3
4,3,3	1	1	1	1
4,-1,1	tie	-1	tie	1
-2,4,5	-1	-1	tie	1
-2,-6,1	tie	1	tie	1
6,0,2	tie	1	-1	-1

## Exercise2



The first split is done using  $z$ , because this is the only variable for which the result stays the same for a value of the variable. The other two splits could be swapped because the influence on the result is the same for both variables.

## Exercise 3

a)

We have

$$S = ((x_1, y_1), \dots, (x_k, y_k))$$

with

$$x_i \in \{-1, 1\}^n, y_i \in \{-1, 1\} \forall i \in [k]$$

Let  $p$  be the number of updates needed for the Perceptron Algorithm.

So we have (1) :

$$\langle w_p, x_i y_i \rangle \geq 0 \quad \forall i \in [k]$$

And we have also :

$$w_p - w_{p-1} = x_{p-1} y_{p-1}$$

So :

$$\langle w_p - w_{p-1}, x_{p-1} y_{p-1} \rangle = \langle x_{p-1} y_{p-1}, x_{p-1} y_{p-1} \rangle$$

$$\langle w_p - w_{p-1}, x_{p-1} y_{p-1} \rangle = n$$

$$\langle w_p, x_{p-1}y_{p-1} \rangle - \langle w_{p-1}, x_{p-1}y_{p-1} \rangle = n$$

Using (1) we have :

$$\langle w_{p-1}, x_{p-1}y_{p-1} \rangle + n \geq 0$$

$$\|w_{p-1}\| \|x_{p-1}y_{p-1}\| + n \geq \langle w_{p-1}, x_{p-1}y_{p-1} \rangle + n \geq 0$$

And we saw in the lecture that for each  $i \leq p-1$  :

$$\|w_i\| \leq \sqrt{i}$$

Thus :

$$n + n^2(p-1) \geq n + n^2\sqrt{p-1} \geq 0$$

Finally:

$$1 - \frac{1}{n} \leq p$$

**b)**

We have that the function:

$$maj(x_i) = \begin{cases} 1 & \text{if } \sum_{j=1}^n x_{ij} > 0 \\ -1 & \text{else} \end{cases}$$

If we have  $\sum_{j=1}^n x_{ij} > 0$ , we need to find  $w$  such as

$$\langle w, x_i \rangle \geq 0 \quad \forall i$$

$$\langle w, x_i \rangle = \sum_{j=1}^n w_j x_{ij}$$

for  $w=(1,...,1)$  we have

$$\langle w, x_i \rangle = \sum_{j=1}^n x_{ij} > 0$$

We need to normalize this  $w$ , so finally our normalized vector  $w$  is :

$$w = \left(\frac{1}{n}, \dots, \frac{1}{n}\right)$$

To find an upper bound on the number of updates of  $w$  the Perceptron Algorithm performs for any training set for the function  $maj$ , we need to find the margin:

$$\min_{(x,y) \in S} |\langle w, x \rangle|$$

And  $S$  is a normalised set We have for each  $p$ :

$$|\langle w, x_p \rangle| = \left| \sum_{j=1}^n \frac{x_{pj}}{n^2} \right|$$

And because  $n$  is an odd number so will must have :

$$\frac{1}{n^2} \leq \left| \sum_{j=1}^{n^2} \frac{x_{pj}}{n^2} \right|$$

So we found a lower bound, we just need to find a vector that verify it, and a vector that have the majority positive or negative by 1 elements verify it.

So the margin is:

$$\gamma = \frac{1}{n^2}$$

Thus ( using theorem 1.10) we find that the upper bound is :

$$n^4 = \frac{1}{\gamma^2}$$

**c)**

Yes, we will still find a linear separator that realizes maj. Because the Perceptron Algorithm has no restriction on the set.

## Exercise 4

**a)**

The Center is underlined.

### Round 1

Cluster 1: A

Cluster 2: B

Cluster 3: C, D, E, F, G, H

### Round 2

Cluster 1: A

Cluster 2: B, C

Cluster 3: (7.5, 6), D, E, F, G, H

### Round 3

Cluster 1: A, B

Cluster 2: (3, 9.5), C

Cluster 3: (8.5, 5.6), D, E, F, G, H

### Round 4

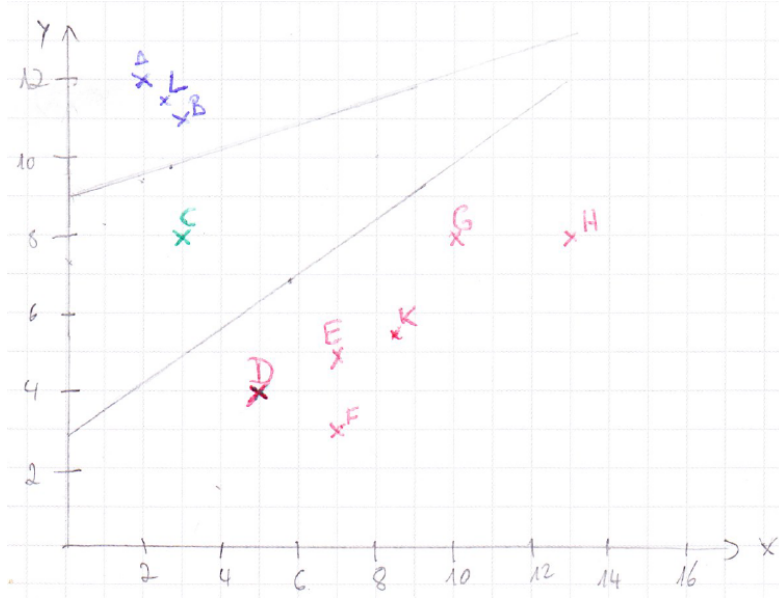
Cluster 1: (3.5, 11.5), A, B

Cluster 2: C

Cluster 3: (8.5, 5.6), D, E, F, G, H

The clustering of the Points stays the same as the round before so the algorithm halts.

b)



$L$  is mean of cluster 1:  $(2.5, 11.5)$ ,  $C$  is mean of cluster 2:  $(3, 8)$ ,  $K$  is mean of cluster 3:  $(8.5, 5.6)$

The dividing lines between the clusters are the lines that contain all points with the same distance to two of the points.

So the line separating the clusters with mean  $A$  and  $B$  are defined by

$$\|(x, f(x)) - A\| = \|(x, f(x)) - B\|$$

$$\Leftrightarrow \|(x - A_1, f(x) - A_2)\| = \|(x - B_1, f(x) - B_2)\|$$

$$\Leftrightarrow \sqrt{(x - A_1)^2 + (f(x) - A_2)^2} = \sqrt{(x - B_1)^2 + (f(x) - B_2)^2}$$

$$\Leftrightarrow (x - A_1)^2 + (f(x) - A_2)^2 = (x - B_1)^2 + (f(x) - B_2)^2$$

$$\Leftrightarrow f(x) = \frac{(x - A_1)^2 - (x - B_1)^2 - A_2^2 - B_2^2}{2(A_2 - B_2)}$$

$$\Leftrightarrow f(x) = \frac{B_1 - A_1}{A_2 - B_2}x + \frac{A_1^2 + B_1^2 - A_2^2 - B_2^2}{2(A_2 - B_2)}$$

The separation of the clusters can be described by three functions:

Cluster 1&2:  $f(x) = \frac{1}{7}x + \frac{181}{7}$

Cluster 2&3:  $f(x) = \frac{11}{4.8}x + \frac{14.11}{4.8}$

Cluster 1&3:  $f(x) = \frac{12}{11.8}x + \frac{85.11}{11.8}$

**c)**

Choosa  $A, F$  and  $H$  as initial cluster means.

$$\|B - A\| = \sqrt{2}, \|C - A\| = \sqrt{17} \rightarrow A, \|D - A\| = \sqrt{73}, \|E - A\| = \sqrt{74}, \|G - A\| = 4\sqrt{5}$$

$$\|B - F\| = 4\sqrt{5}, \|C - F\| = \sqrt{41}, \|D - F\| = \sqrt{5} \rightarrow F, \|E - F\| = 2 \rightarrow F, \|G - F\| = \sqrt{34}$$

$$\|B - H\| = \sqrt{109}, \|C - H\| = 10 \rightarrow A, \|D - H\| = 4\sqrt{5}, \|E - H\| = 3\sqrt{5}, \|G - H\| = 3 \rightarrow H$$

Cluster 1:  $\underline{A}, B, C$

Center is underlined

Cluster 2:  $D, E, \underline{F}$

Cluster 3:  $G, \underline{H}$

Using  $z^j \leftarrow \frac{\sum_{x \in C^j} x}{|C^j|}$  the new centroids are:  $C_1 : (2.667, 10.333), C_2 : (6.333, 4), C_3 : (11.5, 8)$ .

Using the new centroids leads to the following clusters:

$C_1 : (2.667, 10.333), A, B, C$

$C_2 : (6.333, 4), D, E, F$

$C_3 : (11.5, 8), G, H$

Th clusters are the same as in the prevois step so thete starting centroids produce a different result than a).

**d)**

$$\|A - G\| = 4\sqrt{5}, \|B - G\| = \sqrt{58}, \|B - A\| = \sqrt{2}$$

The shortest distance of all the points to eachother is the distance between A and B. So A and B have to be in a cluster except A and B are centers of a cluster each. Then the distance between A and G is longer then B ang G. Therfore G would be in the cluster of B. It's impossible for k-means to return  $\{A, G\}$  as a cluster.