Mohamed Boudiaf University of M'sila

Faculty of Mathematics and Computer Science

Department: Computer Science/ICT

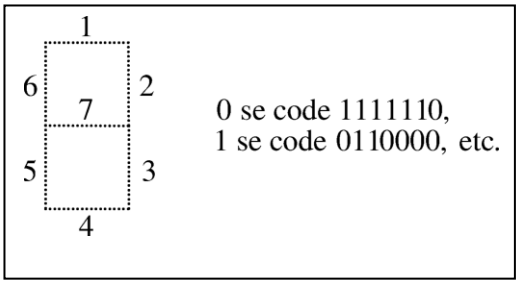
Subject: AI for M1-AI/M2-ISSI

SERIES NO. 04 CORRECTION

Exercise 01

1. Proposal of a binary coding for each digit

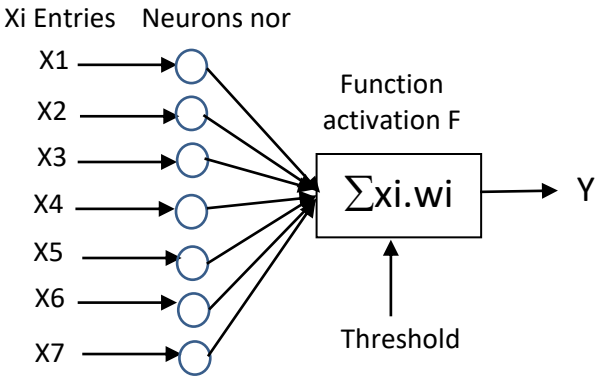
Figure	Binary code
0	1111110
1	0110000
2	1101101
3	1111001
4	0110011
5	1011011
6	1011111
7	1110000
8	1111111
9	1111011



2. Propose an RNA architecture to decide whether the number is even or not

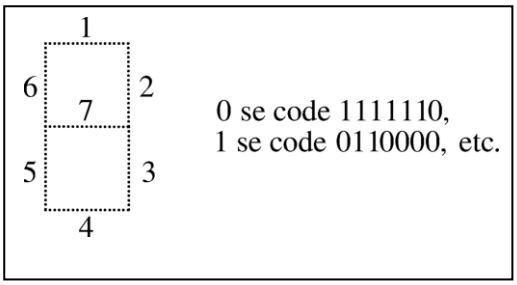
Each digit is expressed in binary on 07 bits

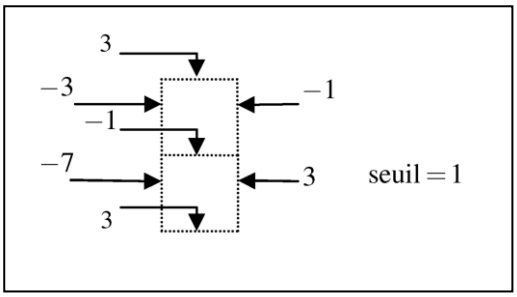
➔Each bit represents the input of a neuron
➔We therefore have: 07 inputs x1, x2, …, x7 and 07 neurons n1, n2, …, n7

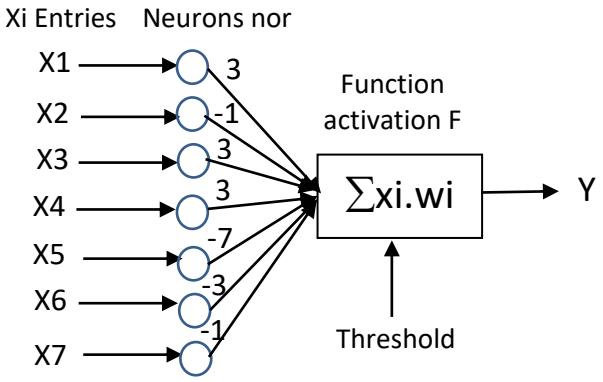


3. Find the learning threshold ε and the corresponding weights wi

The threshold is proposed ε = 1 and the collection of the following weights:





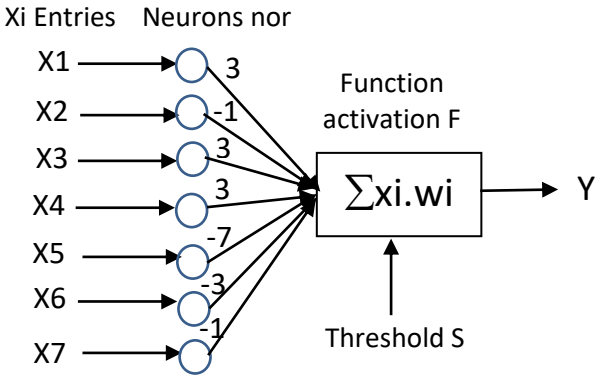


It is also proposed that the network output be calculated according to the following formula:

$$y = \begin{cases} 0 & if \sum W_i.X_i \leq 1 \implies the\ number\ X\ est\ even \\ 1 & if \sum W_i.X_i > 1 \implies the\ number\ X\ is\ odd \end{cases}$$

4. Run the proposed model through examples.

Number	Binary code (network inputs)	∑xi.wi	Y	Kind (Class)
0	1111110	-2	0	even
1	0110000	2	1	odd
4	0110011	-2	0	even
6	1011111	-2	0	even
9	1111011	+4	1	odd

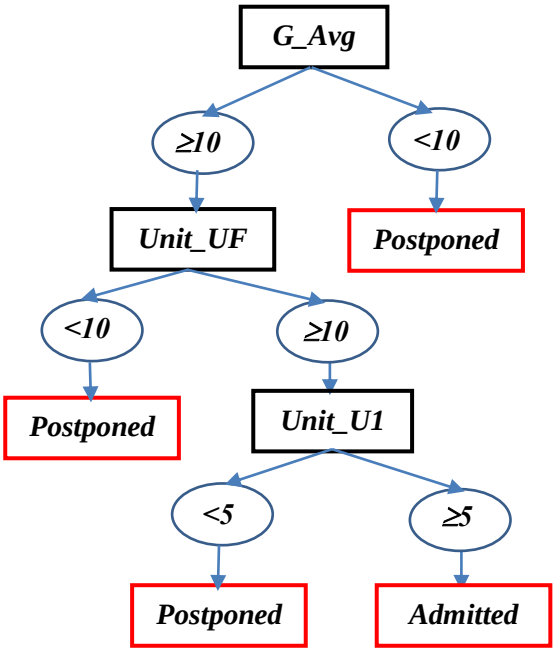


Exercise 2: (Machine Learning and Decision-Making System)

The table below illustrates the contents of the learning base of the company's decision-making system.

Candidate	Mark_U1	Mark_UF	G_Avg	Decision
Ali	4	8	9	Postponed
Mohamed	7	11	10.50	Admitted
Omar	3.50	10	12	Postponed
Farid	7	9	11.50	Postponed
Fatma	15	11	9.80	Postponed
Amira	12.50	13	13.75	Admitted
Salim	3.50	10.50	10	Postponed
Souad	6	14	12	Admitted

1. Construction of a prediction model in the form of a decision tree (from the learning base given above). (02 pts)



2. Reformulate the predictive model in the form of decision rules. (02 pts)

R1: IF (Avg_G <10) Then Postponed
R2: IF (Avg_G≥10) and (Unit_UF <10) Then Postponed
R3: IF (Avg_G≥10) and (Unit_UF≥10)and (Unit_U1 <5) Then Postponed
R4: IF (Avg_G≥10) and (Unit_UF≥10) and (Unit_U1≥5) Then Admitted

Consider the following test base:

3. Prediction of the decision for the candidates given in the test base (based on the predictive model constructed in (1)) (02 pts)

Candidate	Mark_U1	Mark_UF	G_Avg	Decision
Samira	5	10.50	9.99	Postponed
Mourad	7	11	11.25	Admitted
Nassim	2	7	10.50	Postponed
Chahinez	6	12	11.16	Admitted

4. The type of learning used by the decision-making system is supervised, because the classes are predefined (Admitted, Postponed) (01 pt)

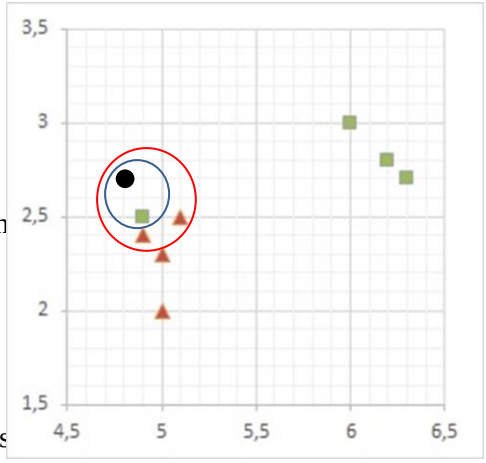
Exercise 03 (Principle of the K-NN algorithm)

The IRIS flower has several variations, among which three are particularly difficult to differentiate. These three variations are: Iris setosa, Iris versicolor and Iris viginica. Recent scientific research suggests that the three flowers can be distinguished by measuring certain elements of the flower, particularly the length and width of the sepal and petal. After taking and analyzing the samples in the lab, here are the results obtained:

Iris	Class	Sepal Length	Sepal Width	Petal Length
1	*Iris setosa*	4.6	3.6	1.0
2	*Iris setosa*	4.3	3.0	1.1
3	*Iris setosa*	5.0	3.2	1.2
4	*Iris setosa*	5.8	4.0	1.2
5	*Iris versicolor*	5.1	2.5	3.0
6	*Iris versicolor*	5.0	2.3	3.3
7	*Iris versicolor*	4.9	2.4	3.3
8	*Iris versicolor*	5.0	2.0	3.5
9	*Iris virginica*	4.9	2.5	4.5

10	*Iris virginica*	6.2	2.8	4.8
11	*Iris virginica*	6.0	3.0	4.8
12	*Iris virginica*	6.3	2.7	4.9

Consider the following graph:



1. Visualize the point with coordinates (4.8, 2.7)
==> (a solid black circle on the graph)
2. What is the closest point to the added one?
==> (the green square)
3. What are the three closest points?
==> (Green square, red triangle, red triangle)
4. What is the majority color (shape) among the three closest poin
==> (red triangle)

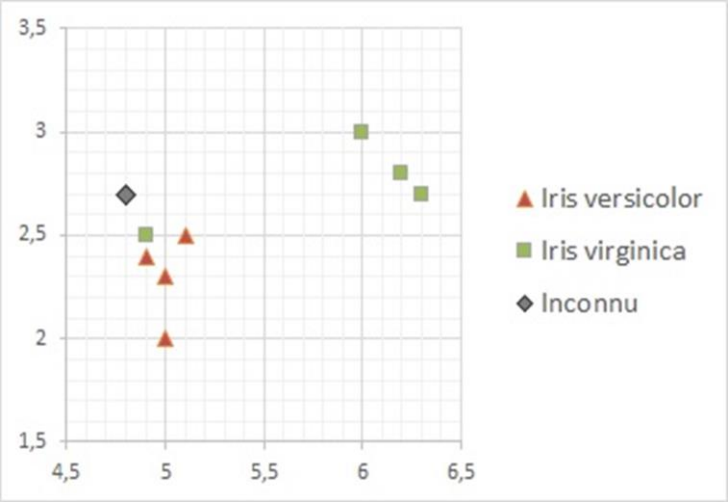
Now let's assume that the Red Triangles represent
Iris versicolor and the Green Squares of Iris virginica.
5. What would be the class of the flower that we just added?
For what ?
==> Iris versicolor according to the majority number of neighbors
(02 neighbors Iris versicolor, 01 neighbor Iris virginica)

Now a counter is associated with each flower class, in our case we have a counter for Iris setosa, one for Iris versicolor and one for Iris virginica. Each counter starts at 0 and increases by 1 each time a neighbor of the counter's class is encountered.
Let's reuse the previous graph in this exercise, considering that the red triangles represent Iris versicolor and the green squares represent Iris virginica. The gray diamond-shaped marker is an unknown Iris.

6. Determine the value of the counters for the 4 nearest neighbors.

==> (Iris versicolor counter = 3, Iris verginica counter = 1)

7. Give a general conclusion regarding the K-NN algorithm



Conclusion :

- The K-nearest neighbors algorithm is based on a fairly simple principle.
- We use known data (training model) to deduce the category (class) of a new element by finding, among the data with the closest values, the one that is most represented.
- We can choose the number of closest data (K). Usually we choose 3.
- Each class has an associated counter. It starts at 0 and increases by 1 each time a neighbor of its class is encountered.
- After doing this operation K times, the class associated with the maximum class counter is the one predicted by the classifier.
- The choice of measures (metrics) for which we compare whether the element is close is important. We will try to choose as best as possible those which allow us to differentiate the classes (discriminating criterion).
- The results are not always reliable! You should not believe 100% of what the classifier tells you, even if it is well done.
- A classifier only recognizes what it has been trained to do. Here, if we give it another flower, a daisy for example, it will deduce the iris that is closest to it.

Exercise 04 (Calculating distances)

Let the two points be: A(xA, yA), B(xB, yB)

We define the following distance metrics between A and B:

1. The distance defined by the inner product (Inner Product) between A and B is given by the following formula:

$Inner(A,B) = x_A * x_B + y_A * y_B$

2. Euclidean distance

$Euc(A,B) = \sqrt{(x_B - x_A)^2 + (y_B - y_A)^2}$

3. Distance from Manhattan (taxi-distance)

$Manh(A,B) = |x_A - x_B| + |y_A - y_B|$

4. Minkowski distance

$Mink(A,B) = \sqrt[q]{(|x_A - x_B|)^q + (|y_A - y_B|)^q}$

with: q=1 Manhattan distance, q=2 Euclidean distance, q>2, Minkowski distance

5. The Cosine distance

$$Cos(A,B) = \frac{x_A * x_B + y_A * y_B}{\sqrt{(x_A{}^2 + y_A{}^2)}.\sqrt{(x_B{}^2 + y_B{}^2)}}$$

6. The Dice Distance

$$Dice(A,B) = \frac{2 * (x_A * x_B + y_A * y_B)}{\sqrt{(x_A{}^2 + y_A{}^2)}.\sqrt{(x_B{}^2 + y_B{}^2)}}$$

6. The Jaccard distance

$$Jacc(A,B) = \frac{(x_A * x_B + y_A * y_B)}{\sqrt{(x_A{}^2 + y_A{}^2)}.\sqrt{(x_B{}^2 + y_B{}^2)} - (x_A * x_B + y_A * y_B)}$$

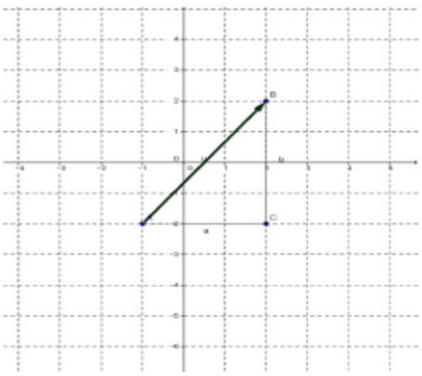
Digital Applications

Calculate the previous distances for:

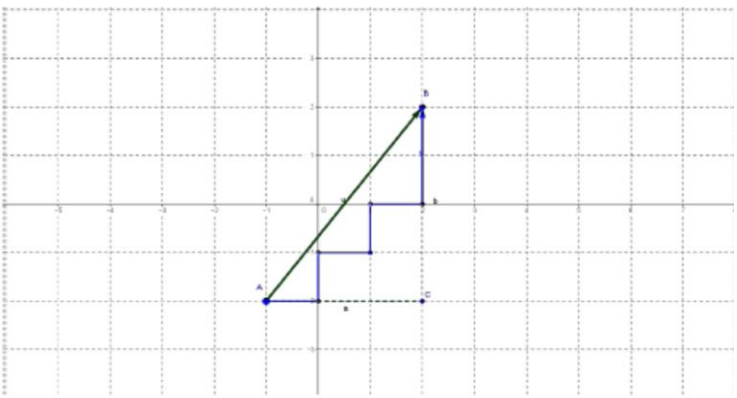
A(-1, -2), B(2, 2)

==>$Inner(A,B) = x_A * x_B + y_A * y_B = (-1) * 2 + (-2) * 2 = -6$

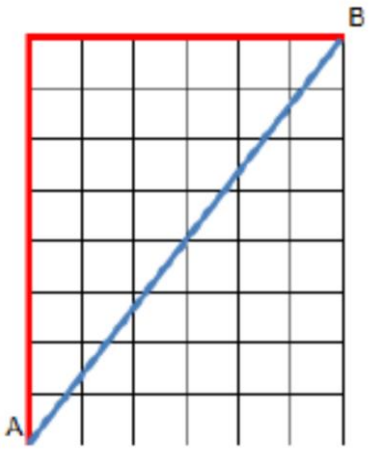
==>$Euc(A,B) = \sqrt{(2-(-1))^2 + (2-(-2))^2} = \sqrt{9+16} = 5$



==>$Manh(A,B) = |-1-2| + |-2-2| = 3 + 4 = 7$



Noticed :



In the plane above, points A and B are separated by 6 units on one variable and 8 units on another variable. The Manhattan distance is therefore 14 (in red) while the Euclidean distance is the square root of $6^2 + 8^2$, or 10 (in blue). Manhattan is therefore 40% greater.
If the points were very far apart, say 20 and 40, Manhattan = 60 and Euclid = 44.7, a difference of 34.16%...

==>$Mink(A,B) = \sqrt[3]{(|-1-2|)^3 + (|-2-2|)^3}$ q=3

4.5= $\sqrt[3]{(3)^3 + (4)^3} = \sqrt[3]{27+64} = \sqrt[3]{91} =$

==> = = -0.95$Cos(A,B) = \frac{x_A * x_B + y_A * y_B}{\sqrt{(x_A{}^2 + y_A{}^2)}.\sqrt{(x_B{}^2 + y_B{}^2)}} = \frac{(-1)*2+(-2)*2}{\sqrt{((-1)^2+(-2)^2)}.((2)^2+(2)^2)} = \frac{-6}{\sqrt{5.8}}\frac{-6}{6.3245}$

==> = -0.923$Dice(A,B) = \frac{2*(x_A * x_B + y_A * y_B)}{(x_A{}^2 + y_A{}^2)+(x_B{}^2 + y_B{}^2)} = \frac{-12}{5+8} = \frac{-12}{13}$

==> -0.316$Jacc(A,B) = \frac{(x_A * x_B + y_A * y_B)}{(x_A{}^2 + y_A{}^2)+(x_B{}^2 + y_B{}^2)-(x_A * x_B + y_A * y_B)} = \frac{-6}{5+8-(-6)} = \frac{-6}{19} =$

Exercise 5 (Naive Bayes Algorithm)

Consider the dataset of 1000 fruits. We have three types: Banana, Orange, and "other". For each fruit, we have 3 characteristics:

- Whether the fruit is long or not
- Whether it is sweet or not
- Whether its color is yellow or not

Our dataset looks like this:

Kind	Long	small (not long)	sugar	Unsweetened	YELLOW	Not yellow	Total
Banana	400	100	350	150	450	50	500
Orange	0	300	150	150	300	0	300
Other fruit	100	100	150	50	50	150	200
Total	500	500	650	350	800	200	1000

Suppose someone asks us to tell him the type of a fruit he has. Its characteristics are as follows:

- ✓ It is yellow
- ✓ It is long
- ✓ It is sweet

To know if it is a banana, or an orange or another fruit, we must calculate the following three possible probabilities:

- ✓ P(Banana | long, yellow, sweet): The probability that it is a banana given that the fruit is long, yellow and sweet
- ✓ P(Orange | long, yellow, sugar): The probability that it is an orange given that the fruit is long, yellow and sweet.
- ✓ P(Other fruit| long, yellow, sweet): The probability that it is another fruit given that the latter is long, yellow and sweet.

The type of "unknown" fruit that we seek to classify will be the one where we have the greatest probability. By applying Bayes' formula, we will have:

$$\text{P(Banane | long, jaune, sucre)} = \frac{P(Long|Banane) * P(Jaune|Banane * P(Sucre|Banane)}{P(Long) * P(Jaune) * P(Sucre)} * P(Banane)$$

$$P(Banane) = \frac{Cardinal(Banane)}{Total\ Fruit} = \frac{500}{1000} = \frac{5}{10} = 0.5$$

$$P(Long|Banane) = \frac{Cardinal(Banane\ ET\ Long)}{Cardinal(Banane)} = \frac{400}{500} = \frac{4}{5} = 0.8$$

$$P(Jaune|Banane) = \frac{Cardinal(Banane\ ET\ Jaune)}{Cardinal(Banane)} = \frac{450}{500} = \frac{45}{50} = 0.9$$

$$P(Sucre|Banane) = \frac{Cardinal(Banane\ ET\ Sucre)}{Cardinal(Banane)} = \frac{350}{500} = \frac{35}{50} = 0.7$$

$$P(Long) = \frac{Cardinal(Long)}{Total\ Fruit} = \frac{500}{1000} = \frac{5}{10} = 0.5$$

$$P(Jaune) = \frac{Cardinal(Jaune)}{Total\ Fruit} = \frac{800}{1000} = \frac{8}{10} = 0.8$$

$$P(Sucre) = \frac{Cardinal(Sucre)}{Total\ Fruit} = \frac{650}{1000} = \frac{65}{100} = 0.65$$

$$\text{P(Banane | long, jaune, sucre)} = \frac{0.8*0.9*0.7}{0.5*0.8*0.65} * 0.5 = \frac{0.252}{0.26} = 0.969$$

In the same way we calculate:

$$\text{P(Orange| long, jaune, sucre)} = \frac{P(Long|Orange) * P(Jaune|Orange * P(Sucre|Orange)}{P(Long) * P(Jaune) * P(Sucre)} * P(Orange)$$

$$\text{P(Autre Fruit| long, jaune, sucre)} = \frac{P(Long|Autre\ Fruit) * P(Jaune|Autre\ Fruit) * P(Sucre|Autre\ Fruit)}{P(Long) * P(Jaune) * P(Sucre)} * P(Autre\ Fruit)$$

$$P(Orange) = \frac{Cardinal(Orange)}{Total\ Fruit} = \frac{300}{1000} = \frac{3}{10} = 0.3$$

$$P(Autre\ Fruit) = \frac{Cardinal(Autre\ Fruit)}{Total\ Fruit} = \frac{200}{1000} = \frac{2}{10} = 0.2$$

SO :

P(Orange| long, jaune, sucre)= 0 (since:$P(Long|Orange) = 0$)

P(Autre Fruit| long, jaune, sucre) $= 0.072$

We notice that the probability that our fruit is a banana P(Banana \mid long, yellow, sugar) is much greater than that of the other probabilities. We classify our unknown fruit as being a banana.

Advantages and limitations of the Naive Bayes Classifier?

Benefits

- The Naive Bayes Classifier is very fast for classification: in fact the probability calculations are not very expensive.
- Classification is possible even with a small dataset

Disadvantages

- The Naive Bayes Classifier algorithm assumes independence of variables: This is a strong assumption and is violated in the majority of real cases.

Counter-intuitively, despite violating the variable independence constraint, Naïve Bayes gives good classification results.

Exercise 6 (SVM Algorithm)

The table below contains data on individuals in a population described according to two attributes: attribute 1 and attribute 2. We want to use the SVM method to classify this data into two classes: C1 and C2.

No.	Attribute 1	Attribute 2	Class
1	0	3	C1
2	3	3	C1
3	1	4	C1
4	2	5	C1
5	1	6	C2
6	2	7	C2
7	3	7	C2
8	3	8	C2
9	2	9	C2
10	4	9	C2
11	8	9	C2

1. Represent on the plan of figure 1 the data of the previous table, symbolizing the class C1 by the symbol (+) and class C2 by a circle (o).
2. Represent the optimal separating hyperplane by clearly showing the vector supports that you use.

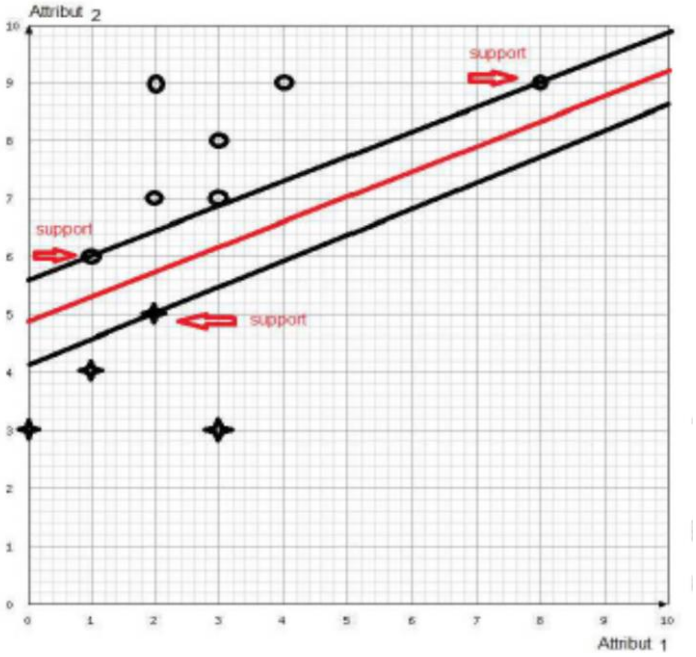


Figure 1

3. We wish to classify a twelfth individual having Attribute1 equal to 4 and Attribute2 equal to 5.6. Redraw all the points in Figure 2, showing the class (+ or o) of the element that comes to be added.
4. Represent the new optimal hyperplane.

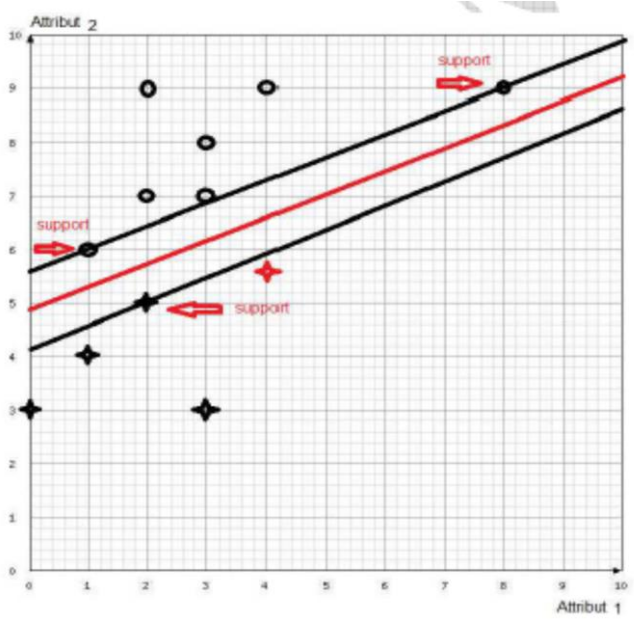


Figure 2

We add another point (2, 3) whose class we know, which is C1.

5. Are the data always linearly separable? If so, give the equation of the new optimal hyperplane. Otherwise, say what SVM predicts in this case.

Yes, the data are always linearly separable. There is always an optimal hyperplane H that separates classes 1 and 2. Its general equation is w. x + b ; w being the weight vector, and b the bias.

Good luck