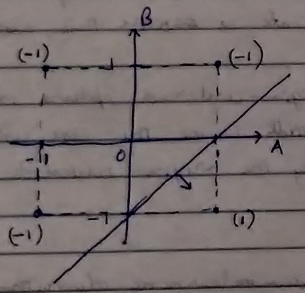
Problem Set 1

2) Design a two-input perceptron that implements the boolean function $A \wedge \neg B$. Design a two-layer network of perceptrons that implements A XOR B

$A \wedge \neg B$

A	B	$A \wedge \neg B$
1	1	-1
1	-1	1 ← Only true output
-1	1	-1
-1	-1	-1



If we design a linear classifier as shown, it will correctly classify $A \wedge \neg B$.
The equation of the line is $1 - A + B = 0$
Therefore, the perceptron can have the following equation:

$$w_0 + w_1 A + w_2 B = 0$$

$w_0 = -1$

$w_1 = 1$

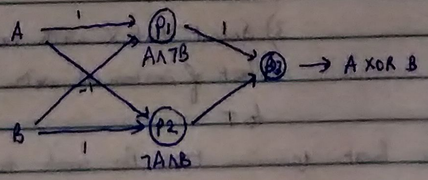
$w_2 = -1$

(The weights are multiplied by -1 to account for the positive example on the right side of the linear classifier)

A XOR B

A	B	A XOR B
-1	-1	-1
1	-1	1
-1	1	1
1	1	-1

To design A XOR B, we first write A XOR B $= (A \wedge \neg B) \vee (\neg A \wedge B)$. Given that information, we can use (i) in the following structure:



7) Give the VC dimension of these hypothesis spaces, briefly explaining your answer.

1. An origin centered circle (2D)
2. An origin centered sphere (3D)

For an origin centered circle, the VC dimension is 2. This is because if we have two points not at the same distance from the center, we can shatter this case using a circle centered at origin. However, if there are three points, they will be at three, possibly different distances $r_1 \leq r_2 \leq r_3$. In this case, ~~the~~ for the labels $+, -, +$ respectively, there is no way for the origin-centered circle to shatter this case.

The same argument holds for ~~an~~ origin-centered sphere in 3 dimensions. The VC dimension is still 2, as the case with 3 points in 3D cannot be shattered by the origin-centered sphere.

5) Suggest a lazy version of the eager decision tree learning algorithm ID3. What are the advantages/disadvantages of your lazy learning algorithm compared to the original eager learning algorithm?

(Adapted from Lazy Decision Trees talk by Ronny Kohavi)

LazyDT:

⇒ Input: training set T, instance I to classify

⇒ Output: label for instance I

⇒ Algorithm: 1) If T is pure (all instances have same label), return label L

2) If all instances have same feature values, return majority class in T

Stopping criterion

3) Select test X with value x for instance I. Assign the same set of instances with X = x to T and apply the algorithm to T

recursive

The best decision tree for each test instance should be selected, but only the path taken by the test instance matters.

Advantages: 1) Since only one path is constructed, the lazyDT algorithm is fast
2) LazyDT never considers a split on a feature with an unknown value
~~So~~ C4.5 will penalize such attributes based on the ratio of missing values, which might lead to important values being missed for certain test cases

Disadvantages: 1) There is no pruning, which means that the algorithm proceeds until the leaf is pure
2) Data has to be discretized in advance, which means very eager and local interactions are lost.