Groupe : Yassine SEDJARI,Mohammed NECHBA, Mohamed MOUHAJIR
Supervisé par : Pr. Abdellatif El Afia

VC dimension Algorithm

Input:

- A hypothesis class H
- Training set S -> D^m

Output:

- Vc dimension of h given the sample S

Begin:

```
SET the vc dimension to 1
SET cantShatter to False
WHILE we can shatter do
    SET the sample size m to vc+1
    SET contsearch to True
    FOR each sample size A of size m sampled from S
        SET cantShatter to False
        IF we can not continue to search THEN
            BREAK from the for loop
        END IF
        SET contsearch to False
        FOR each possible label vector label_ of A
            SET h to be a hypothesis from the hypothesis class H
            TRAIN h on A and label_
            SET y_pred to be the prediction h(A)
            IF y_pred != label_ THEN
                SET cantshatter to True
                SET contsearch to True
                BREAK from the for loop
            END IF
        END FOR
    END FOR
    IF we can shatter THEN
        INCREMENT vc by 1
    END IF
END WHILE
RETURN vc
```

End:

Uniform Covering Number Algorithm

Input:

- H: A hypothesis class
- S -> D^m: Sample S sampled with a distribution D
- Epsilon: the radius of the covering balls
- m: the length of the Sample A
- d: the distance measure

Output:

- Uniform covering number of hypothesis class H

Begin:

GENERATE all the possible samples A of length m from S
SET uniform covering number to 1
FOR each sample A of length m
  CALL HA **WITH** the Sample A and the class H **RETURNING** H_A the restriction of H over A
  CALL Ne **WITH** H_A the restriction of H over A, the radius epsilon and the distance d **RETURNING** the covering number of the set H_A
  IF the covering number is greater than the uniform covering number **THEN**
    SET the uniform covering number to the covering number
  ENDIF
ENDFOR
RETURN the uniform covering number

End:

Covering Number Algorithm

Input:

- W: A set in a metric space (M,d), in our case W is finite and discrete
- Epsilon: the radius of the covering balls
- d: the distance measure

Output:

- Covering number of the set W

Begin:

```
SET m to the length of W
FOR each point in W
    COLOUR point p with black colour
ENDFOR
SET covering number to 0
SET number of points covered to 0
WHILE there exist a point still coloured in black
    FOR each point p coloured in black
        CALCULATE the neighbourhood of p and store it in a list
    ENDFOR
    PICK the point pmax with the biggest neighbourhood
    COLOUR the point pmax and its neighbourhood with colour white
    INCREMENT the number of points covered by the size of the
neighbourhood of pmax
    INCREMENT the covering number by 1
ENDWHILE
RETURN the covering number
```

End:

HA The restriction of H over A Algorithm

Input:

- A: a sample of size m sampled from S
- Epsilon: the radius of the covering balls
- H: A hypothesis class

Output:

- H_A: the restriction set of H over A

Begin:

```
    SET H_A to an empty set
    INSTANTIATE All hypothesis h of the class H and store them in the list all_h
    FOR each hypothesis h in all_h
        CALCULATE the prediction h(A)
        IF h(A) is not already in H_A THEN
            ADD h(A) to the H_A set
        ENDIF
    ENDFOR
    RETURN H_A
```

End: