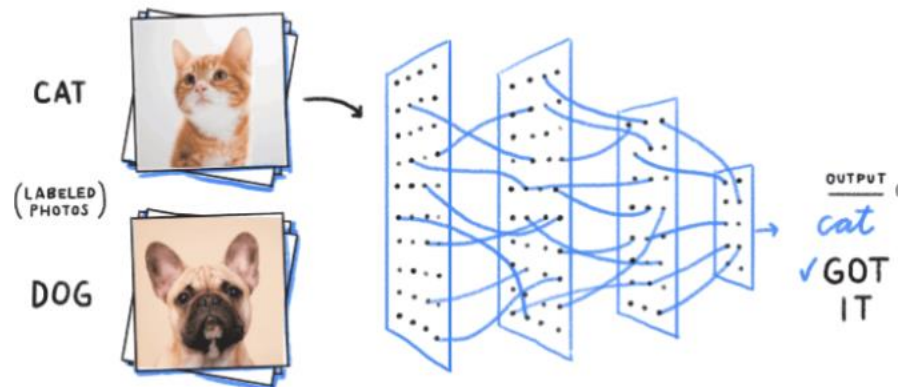


Perceptron



what is the perceptron?

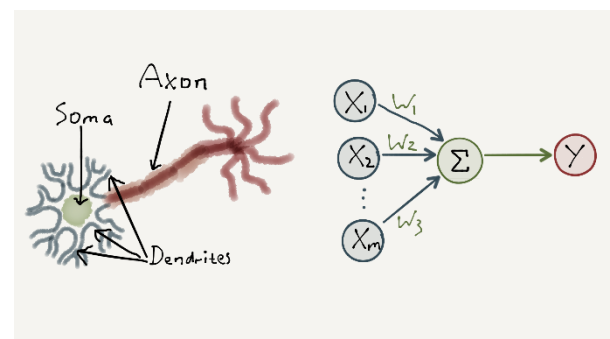
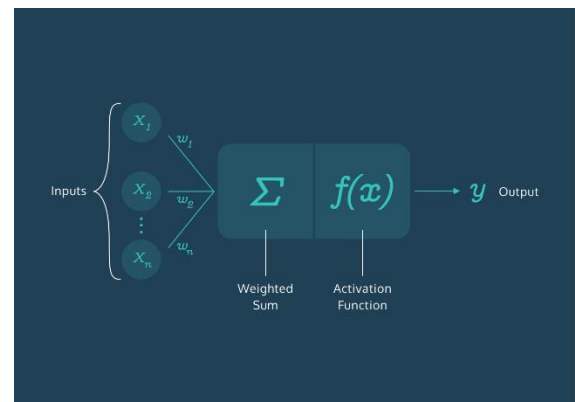
Perceptron one of the most basic linear machine learning algorithm for supervised learning of binary classifiers

Who is it work?

The process begins by taking all the input values and multiplying them by their weights. Then, all of these multiplied values are added together to create the weighted sum. The weighted sum is then applied to the activation function, producing the perceptron's output.

What is the goal for perceptron?

The main goal is to calculate right weights and bias, based on these two parameters, we can make good predicted



Pseudocode and algorithm

STEP 1 : INITIALIZE WEIGHTS TO ZERO FOR ALL x_1 TO x_N

STEP 2 : INITIALIZE BIAS TO ZERO

STEP 3 : MAIN FOR LOOP FORM 1 TO ITERATION

STEP 4 : FOR LOOP PASS THROUGH ALL FEATURES AND LABELS

STEP 5: COMPUTE ACTIVATION ,

$A = w_1 * x_1 + w_2 * x_2 \dots + w_N * x_N$ THEN ADD BIAS ,
CAN BE SIMPLIFIED TO DOT PRODUCT =
 $NP.DOT(WEIGHTS, FEATURE) + BIAS$

STEP 6: CHECK IF THERE ERROR OR NOT BY
MULTIPLY(Y,A) THEN CHECK IF ≤ 0

NOTE Y IS THE LABEL OR CLASS FOR THE FEATURE
 x_N AND Y IS EITHER +1 OR -1

STEP 7 : UPDATE WEIGHTS AND BIAS FOLLOW THE
ALGORITHM

STEP 8: RETURN WEIGHTS AND BIAS

STEP 9 : PERCEPTRON TEST SAME AS STEP 5 BUT
RETURN THE SIGN OF A

Algorithm 5 PERCEPTRONTRAIN($D, MaxIter$)

```
1:  $w_d \leftarrow 0$ , for all  $d = 1 \dots D$  // initialize weights
2:  $b \leftarrow 0$  // initialize bias
3: for  $iter = 1 \dots MaxIter$  do
4:   for all  $(x, y) \in D$  do
5:      $a \leftarrow \sum_{d=1}^D w_d x_d + b$  // compute activation for this example
6:     if  $ya \leq 0$  then
7:        $w_d \leftarrow w_d + yx_d$ , for all  $d = 1 \dots D$  // update weights
8:        $b \leftarrow b + y$  // update bias
9:     end if
10:  end for
11: end for
12: return  $w_0, w_1, \dots, w_D, b$ 
```

Algorithm 6 PERCEPTRONTEST($w_0, w_1, \dots, w_D, b, \hat{x}$)

```
1:  $a \leftarrow \sum_{d=1}^D w_d \hat{x}_d + b$  // compute activation for the test example
2: return SIGN( $a$ )
```

implementation Iris classification with Perceptron.

Task 1 : setosa with versicolor

Task 2 : versicolor with virginica

Task 3 : setosa with virginica

Task 1 :

1- WHAT WOULD BE THE IDEAL NUMBER OF ITERATIONS TO TERMINATE THE TRAINING?

- 2

2 - WHAT IS THE TRAIN AND TEST CLASSIFICATION ACCURACIES FOR EACH ONE OF THE CLASSIFIERS AFTER 30 EPOCHS (ITERATIONS)?

- 100%

perceptron test sample

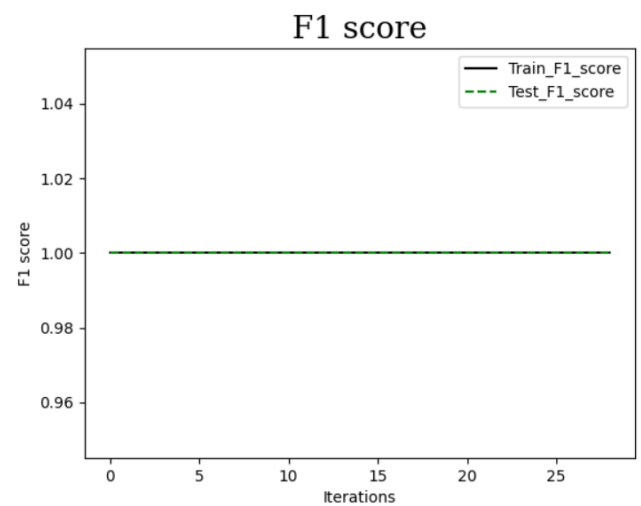
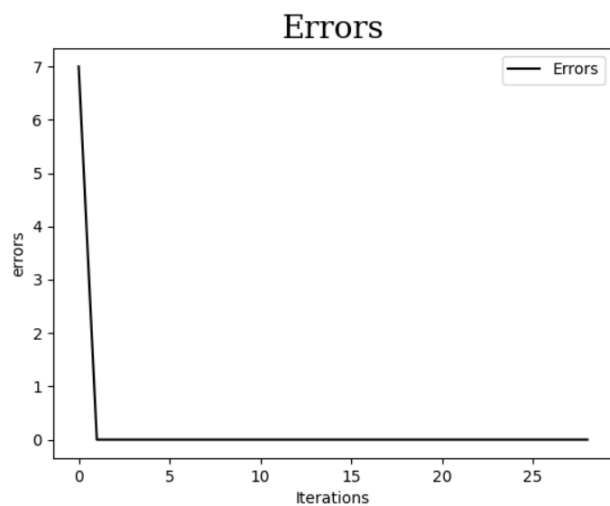
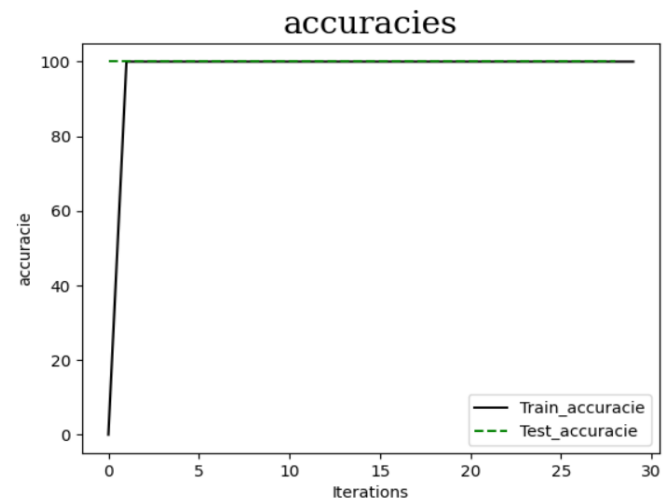
X => [5.6 2.7 4.2 1.3] | LABEL : VERSICOLOR | PREDICTION : VERSICOLOR

X => [5.5 2.6 4.4 1.2] | LABEL : VERSICOLOR | PREDICTION : VERSICOLOR

X => [4.4 3.2 1.3 0.2] | LABEL : SETOSA | PREDICTION : SETOSA

X => [5.1 3.8 1.9 0.4] | LABEL : SETOSA | PREDICTION : SETOSA

X => [5. 2.3 3.3 1.] | LABEL : VERSICOLOR | PREDICTION : VERSICOLOR



Task 2 :

1- what would be the ideal number of iterations to terminate the training?

- 28

2 - What is the train and test classification accuracies for each one of the classifiers after 30 epochs (iterations)?

- **train is 97.5% and test 100%**

perceptron test sample

X => [5.7 3. 4.2 1.2] | label : versicolor | prediction : versicolor

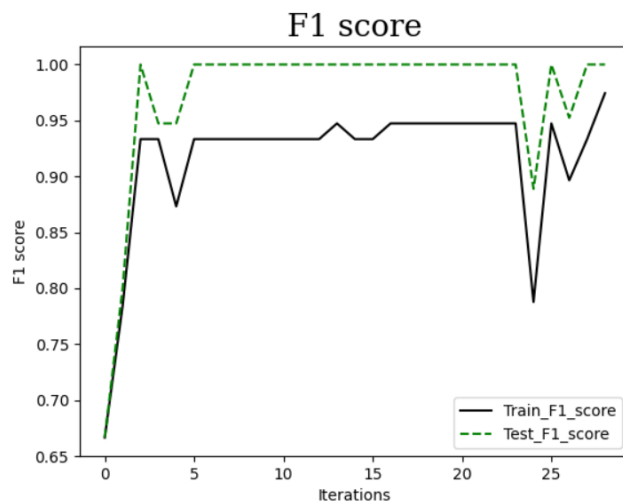
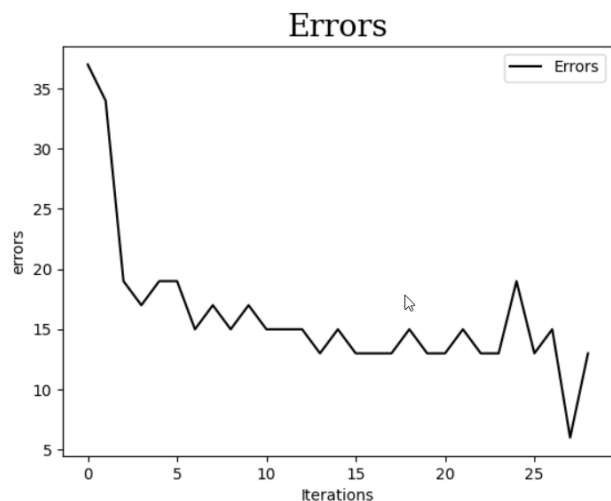
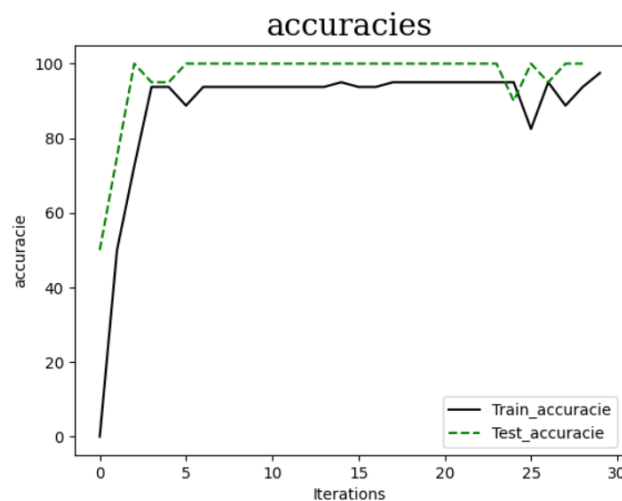
X => [6.7 3.3 5.7 2.5] | label : virginica | prediction : virginica

X => [6.5 3. 5.2 2.] | label : virginica | prediction : virginica

X => [5.6 2.7 4.2 1.3] | label : versicolor | prediction : versicolor

X => [6.2 3.4 5.4 2.3] | label : virginica | prediction : virginica

X => [6.7 3.1 5.6 2.4] | label : virginica | prediction : virginica



Task 3:

1- what would be the ideal number of iterations to terminate the training?

- 2

2 - What is the train and test classification accuracies for each one of the classifiers after 30 epochs (iterations)?

- 100%

perceptron test sample

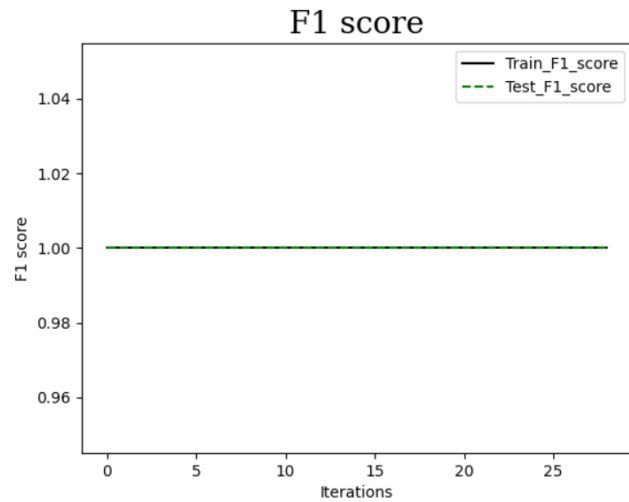
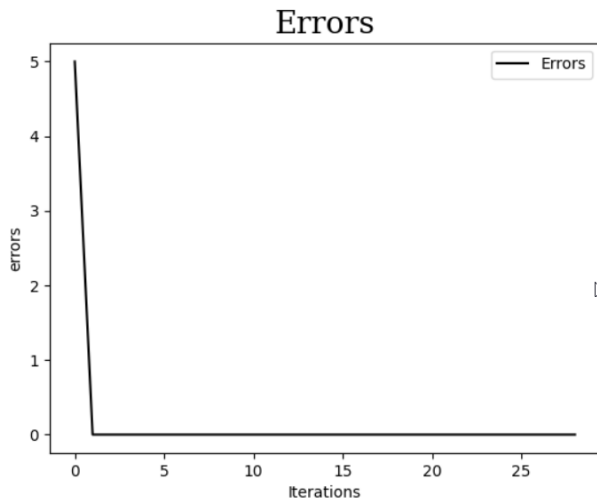
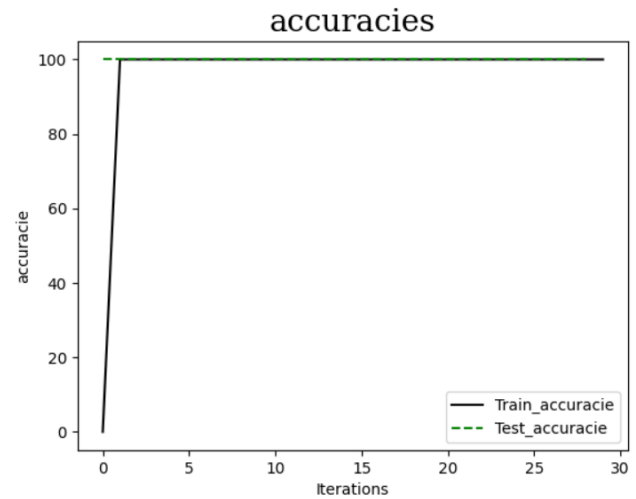
X => [5.1 3.8 1.6 0.2] | label : setosa | prediction : setosa

X => [4.8 3. 1.4 0.3] | label : setosa | prediction : setosa

X => [6.7 3.1 5.6 2.4] | label : virginica | prediction : virginica

X => [6.3 2.5 5. 1.9] | label : virginica | prediction : virginica

X => [5.1 3.8 1.9 0.4] | label : setosa | prediction : setosa



Resources

<https://deepai.org/machine-learning-glossary-and-terms/perceptron>

<https://en.wikipedia.org/wiki/Perceptron>

<https://towardsdatascience.com/what-the-hell-is-perceptron-626217814f53>

<https://www.simplilearn.com/tutorials/deep-learning-tutorial/perceptron>

