CMSC 170: Introduction to Artificial intelligence

Week 06: Linear Classification using Perceptron

John O-Neil V. Geronimo
Institute of Computer Science
University of the Philippines Los Baños

Content

I. Background
II. Implementation of Linear Classification Using Perceptrons

Content

Background

A perceptron is a form of simple neural network, consisting of a **single neuron** that takes a feature vector with n coordinates, assigns each coordinate with a corresponding weight, and outputs the feature vector's classification based on a threshold function.

Background

The perceptron algorithm was conceptualized by Frank Rosenblatt in 1957, and it was one of the first neural networks to be implemented.

Background

Perceptron only works on **linearly separable** data. If the training data is not linearly separable, the perceptron algorithm will not converge

Algorithm

Given values:

- m feature vectors with n coordinates and a corresponding target label/classification, y, each: $[\boldsymbol{x_0}, \boldsymbol{x_1}, \dots, \boldsymbol{x_n} \rightarrow \boldsymbol{y}]$
- n weights, $\boldsymbol{w_0}, \boldsymbol{w_1}, \dots \boldsymbol{w_n}$, for each coordinate of a feature vector
- Learning rate, $\boldsymbol{r}$
- Threshold, $\boldsymbol{t}$
- Bias, $\boldsymbol{b}$

Algorithm

Step 1

Choose initial weights (may be random, but are usually initialized to 0)

Algorithm

Step 2

For each individual feature vector (row)

a. Compute perceptron value, a

$$a = \sum_{i=0}^{n}(x_i w_i) + b w_b$$

b. Determine classification, y

$$y = (a \geq threshold)?1:0$$

c. Adjust weights

$$w_a = w_c + r x_p (z - y)$$

where, w_a is the adjusted weight, w_c is the most recent/current value of the weight, and x_p pertains to the x-value of the previous feature vector

Algorithm

Step 3

if (weights **converge**):

stop learning

else:

repeat step 2 for the next feature vector.

The weights have converged if they stay the **same** throughout all feature vectors in the training data set.

Content

I. Background
II. Implementation of Linear Classification Using Perceptrons

Example

Consider the given values for our perceptron:

Learning rate = 0.1

Threshold = 0.5

Bias = 1

X_0	X_1	Z
0	0	0
0	1	1
1	0	1
1	1	1

Algorithm

Step 1

Choose initial weights (may be random, but are usually initialized to 0)

X_0	X_1	b	w_0	w_1	w_b	a	y	Z
0	0	1						0
0	1	1						1
1	0	1						1
1	1	1						1

Algorithm

Step 1

Choose initial weights (may be random, but are usually initialized to 0)

X_0	X_1	b	w_0	w_1	w_b	a	y	Z
0	0	1	0	0	0			0
0	1	1						1
1	0	1						1
1	1	1						1

Algorithm

Step 2

For each individual feature vector (row)

a. Compute perceptron value, a

$$a = \sum_{i=0}^{n}(x_i w_i) + b w_b$$

b. Determine classification, y

$$y = (a \geq threshold)\,?\,1:0$$

c. Adjust weights

$$w_a = w_c + r x_p (z - y)$$

where, w_a is the adjusted weight, w_c is the most recent/current value of the weight, and x_p pertains to the x-value of the previous feature vector

X_0	X_1	b	w_0	w_1	w_b	a	y	Z
0	0	1	0	0	0			0
0	1	1						1
1	0	1						1
1	1	1						1

Start of our iteration

Algorithm

Step 2

For each individual feature vector (row)

a. Compute perceptron value, a

$$a = \sum_{i=0}^{n}(x_i w_i) + b w_b$$

b. Determine classification, y

$$y = (a \geq threshold) ? 1 : 0$$

c. Adjust weights

$$w_a = w_c + r x_p (z - y)$$

where, w_a is the adjusted weight, w_c is the most recent/current value of the weight, and x_p pertains to the x-value of the previous feature vector

X_0	X_1	b	w_0	w_1	w_b	a	y	Z
0	0	1	0	0	0			0
0	1	1						1
1	0	1						1
1	1	1						1

a = 0(0) + 0(0) + 1(0)

a = 0

Algorithm

Step 2

For each individual feature vector (row)

a. Compute perceptron value, a

$$a = \sum_{i=0}^{n}(x_i w_i) + b w_b$$

b. Determine classification, y

$$y = (a \geq threshold)\,?\,1:0$$

c. Adjust weights

$$w_a = w_c + r x_p(z - y)$$

where, w_a is the adjusted weight, w_c is the most recent/current value of the weight, and x_p pertains to the x-value of the previous feature vector

X_0	X_1	b	w_0	w_1	w_b	a	y	Z
0	0	1	0	0	0	0		0
0	1	1						1
1	0	1						1
1	1	1						1

a = 0(0) + 0(0) + 1(0)

a = 0

Algorithm

Step 2

For each individual feature vector (row)

a. Compute perceptron value, a

$$a = \sum_{i=0}^{n}(x_i w_i) + b w_b$$

b. Determine classification, y

$$y = (a \geq threshold)\,?\,1:0$$

c. Adjust weights

$$w_a = w_c + r x_p (z - y)$$

where, w_a is the adjusted weight, w_c is the most recent/current value of the weight, and x_p pertains to the x-value of the previous feature vector

X_0	X_1	b	w_0	w_1	w_b	a	y	Z
0	0	1	0	0	0	0		0
0	1	1						1
1	0	1						1
1	1	1						1

a = 0 t = 0.5

0 > 0.5? False

Therefore, y = 0

Algorithm

Step 2

For each individual feature vector (row)

a. Compute perceptron value, a

$$a = \sum_{i=0}^{n}(x_i w_i) + b w_b$$

b. Determine classification, y

$$y = (a \geq threshold)\,?\,1:0$$

c. Adjust weights

$$w_a = w_c + r x_p(z - y)$$

where, w_a is the adjusted weight, w_c is the most recent/current value of the weight, and x_p pertains to the x-value of the previous feature vector

X_0	X_1	b	w_0	w_1	w_b	a	y	Z
0	0	1	0	0	0	0	0	0
0	1	1						1
1	0	1						1
1	1	1						1

a = 0 t = 0.5

0 > 0.5? False

Therefore, y = 0

Algorithm

Step 2

For each individual feature vector (row)

a. Compute perceptron value, a

$$a = \sum_{i=0}^{n}(x_i w_i) + bw_b$$

b. Determine classification, y

$$y = (a \geq threshold)?1:0$$

c. Adjust weights

$$w_a = w_c + rx_p(z - y)$$

where, w_a is the adjusted weight, w_c is the most recent/current value of the weight, and x_p pertains to the x-value of the previous feature vector

X_0	X_1	b	w_0	w_1	w_b	a	y	Z
0	0	1	0	0	0	0	0	0
0	1	1						1
1	0	1						1
1	1	1						1

$W_0 = 0 + (0.1)(0)(0\text{-}0) = 0$

$W_1 = 0 + (0.1)(0)(0\text{-}0) = 0$

$W_b = 0 + (0.1)(1)(0\text{-}0) = 0$

Algorithm

Step 2

For each individual feature vector (row)

a. Compute perceptron value, a

$$a = \sum_{i=0}^{n}(x_i w_i) + b w_b$$

b. Determine classification, y

$$y = (a \geq threshold)\,?\,1:0$$

c. Adjust weights

$$w_a = w_c + r x_p (z - y)$$

where, w_a is the adjusted weight, w_c is the most recent/current value of the weight, and x_p pertains to the x-value of the previous feature vector

X_0	X_1	b	w_0	w_1	w_b	a	y	Z
0	0	1	0	0	0	0	0	0
0	1	1	0	0	0			1
1	0	1						1
1	1	1						1

$W_0 = 0 + (0.1)(0)(0\text{-}0) = 0$

$W_1 = 0 + (0.1)(0)(0\text{-}0) = 0$

$W_b = 0 + (0.1)(1)(0\text{-}0) = 0$

Algorithm

Step 2

For each individual feature vector (row)

a. Compute perceptron value, a

$$a = \sum_{i=0}^{n}(x_i w_i) + b w_b$$

b. Determine classification, y

$$y = (a \geq threshold)\,?\,1:0$$

c. Adjust weights

$$w_a = w_c + r x_p (z - y)$$

where, w_a is the adjusted weight, w_c is the most recent/current value of the weight, and x_p pertains to the x-value of the previous feature vector

X_0	X_1	b	w_0	w_1	w_b	a	y	Z
0	0	1	0	0	0	0	0	0
0	1	1	0	0	0			1
1	0	1						1
1	1	1						1

a = 0(0) + 1(0) + 1(0)

a = 0

Algorithm

Step 2

For each individual feature vector (row)

a. Compute perceptron value, a

$$a = \sum_{i=0}^{n}(x_i w_i) + b w_b$$

b. Determine classification, y

$$y = (a \geq threshold)\,?\,1:0$$

c. Adjust weights

$$w_a = w_c + r x_p (z - y)$$

where, w_a is the adjusted weight, w_c is the most recent/current value of the weight, and x_p pertains to the x-value of the previous feature vector

X_0	X_1	b	w_0	w_1	w_b	a	y	Z
0	0	1	0	0	0	0	0	0
0	1	1	0	0	0	0		1
1	0	1						1
1	1	1						1

a = 0(0) + 1(0) + 1(0)

a = 0

Algorithm

Step 2

For each individual feature vector (row)

a. Compute perceptron value, a

$$a = \sum_{i=0}^{n}(x_i w_i) + bw_b$$

b. Determine classification, y

$$y = (a \geq threshold)?1:0$$

c. Adjust weights

$$w_a = w_c + rx_p(z - y)$$

where, w_a is the adjusted weight, w_c is the most recent/current value of the weight, and x_p pertains to the x-value of the previous feature vector

X_0	X_1	b	w_0	w_1	w_b	a	y	Z
0	0	1	0	0	0	0	0	0
0	1	1	0	0	0	0		1
1	0	1						1
1	1	1						1

a = 0 t = 0.5

0 > 0.5? False

Therefore, y = 0

Algorithm

Step 2

For each individual feature vector (row)

a. Compute perceptron value, a

$$a = \sum_{i=0}^{n}(x_i w_i) + b w_b$$

b. Determine classification, y

$$y = (a \geq threshold)\,?\,1:0$$

c. Adjust weights

$$w_a = w_c + r x_p (z - y)$$

where, w_a is the adjusted weight, w_c is the most recent/current value of the weight, and x_p pertains to the x-value of the previous feature vector

X_0	X_1	b	w_0	w_1	w_b	a	y	Z
0	0	1	0	0	0	0	0	0
0	1	1	0	0	0	0	0	1
1	0	1						1
1	1	1						1

a = 0 t = 0.5

0 > 0.5? False

Therefore, y = 0

Algorithm

Step 2

For each individual feature vector (row)

a. Compute perceptron value, a

$$a = \sum_{i=0}^{n}(x_i w_i) + bw_b$$

b. Determine classification, y

$$y = (a > threshold)?1:0$$

c. Adjust weights

$$w_a = w_c + rx_p(z - y)$$

where, w_a is the adjusted weight, w_c is the most recent/current value of the weight, and x_p pertains to the x-value of the previous feature vector

X_0	X_1	b	w_0	w_1	w_b	a	y	Z
0	0	1	0	0	0	0	0	0
0	1	1	0	0	0	0	0	1
1	0	1						1
1	1	1						1

$W_0 = 0 + (0.1)(0)(1\text{-}0) = 0$

$W_1 = 0 + (0.1)(1)(1\text{-}0) = 0.1$

$W_b = 0 + (0.1)(1)(1\text{-}0) = 0.1$

Algorithm

Step 2

For each individual feature vector (row)

a. Compute perceptron value, a

$$a = \sum_{i=0}^{n}(x_i w_i) + b w_b$$

b. Determine classification, y

$$y = (a > threshold)\,?\,1:0$$

c. Adjust weights

$$w_a = w_c + r x_p (z - y)$$

where, w_a is the adjusted weight, w_c is the most recent/current value of the weight, and x_p pertains to the x-value of the previous feature vector

X_0	X_1	b	w_0	w_1	w_b	a	y	Z
0	0	1	0	0	0	0	0	0
0	1	1	0	0	0	0	0	1
1	0	1	0	0.1	0.1			1
1	1	1						1

$W_0 = 0 + (0.1)(0)(1\text{-}0) = 0$

$W_1 = 0 + (0.1)(1)(1\text{-}0) = 0.1$

$W_b = 0 + (0.1)(1)(1\text{-}0) = 0.1$

Algorithm

Step 2

For each individual feature vector (row)

a. Compute perceptron value, a

$$a = \sum_{i=0}^{n}(x_i w_i) + b w_b$$

b. Determine classification, y

$$y = (a \geq threshold)\,?\,1:0$$

c. Adjust weights

$$w_a = w_c + r x_p (z - y)$$

where, w_a is the adjusted weight, w_c is the most recent/current value of the weight, and x_p pertains to the x-value of the previous feature vector

X_0	X_1	b	w_0	w_1	w_b	a	y	Z
0	0	1	0	0	0	0	0	0
0	1	1	0	0	0	0	0	1
1	0	1	0	0.1	0.1			1
1	1	1						1

a = 1(0) + 0(0.1) + 1(0.1)

a = 0.1

Algorithm

Step 2

For each individual feature vector (row)

a. Compute perceptron value, a

$$a = \sum_{i=0}^{n}(x_i w_i) + b w_b$$

b. Determine classification, y

$$y = (a \geq threshold)\,?\,1:0$$

c. Adjust weights

$$w_a = w_c + r x_p (z - y)$$

where, w_a is the adjusted weight, w_c is the most recent/current value of the weight, and x_p pertains to the x-value of the previous feature vector

X_0	X_1	b	w_0	w_1	w_b	a	y	Z
0	0	1	0	0	0	0	0	0
0	1	1	0	0	0	0	0	1
1	0	1	0	0.1	0.1	0.1		1
1	1	1						1

a = 1(0) + 0(0.1) + 1(0.1)

a = 0.1

Algorithm

Step 2

For each individual feature vector (row)

a. Compute perceptron value, a

$$a = \sum_{i=0}^{n}(x_i w_i) + b w_b$$

b. Determine classification, y

$$y = (a \geq threshold)\,?\,1:0$$

c. Adjust weights

$$w_a = w_c + r x_p (z - y)$$

where, w_a is the adjusted weight, w_c is the most recent/current value of the weight, and x_p pertains to the x-value of the previous feature vector

X_0	X_1	b	w_0	w_1	w_b	a	y	Z
0	0	1	0	0	0	0	0	0
0	1	1	0	0	0	0	0	1
1	0	1	0	0.1	0.1	0.1		1
1	1	1						1

a = 0.1 t = 0.5

0.1 > 0.5? False

Therefore, y = 0

Algorithm

Step 2

For each individual feature vector (row)

a. Compute perceptron value, a

$$a = \sum_{i=0}^{n}(x_i w_i) + bw_b$$

b. Determine classification, y

$$y = (a \geq threshold)\,?\,1:0$$

c. Adjust weights

$$w_a = w_c + rx_p(z - y)$$

where, w_a is the adjusted weight, w_c is the most recent/current value of the weight, and x_p pertains to the x-value of the previous feature vector

X_0	X_1	b	w_0	w_1	w_b	a	y	Z
0	0	1	0	0	0	0	0	0
0	1	1	0	0	0	0	0	1
1	0	1	0	0.1	0.1	0.1	0	1
1	1	1						1

a = 0.1 t = 0.5

0.1 > 0.5? False

Therefore, y = 0

Algorithm

Step 2

For each individual feature vector (row)

a. Compute perceptron value, a

$$a = \sum_{i=0}^{n}(x_i w_i) + b w_b$$

b. Determine classification, y

$$y = (a \geq threshold)?1:0$$

c. Adjust weights

$$w_a = w_c + r x_p (z - y)$$

where, w_a is the adjusted weight, w_c is the most recent/current value of the weight, and x_p pertains to the x-value of the previous feature vector

X_0	X_1	b	w_0	w_1	w_b	a	y	Z
0	0	1	0	0	0	0	0	0
0	1	1	0	0	0	0	0	1
1	0	1	0	0.1	0.1	0.1	0	1
1	1	1						1

$W_0 = 0 + (0.1)(1)(1\text{-}0) = 0.1$

$W_1 = 0.1 + (0.1)(0)(1\text{-}0) = 0.1$

$W_b = 0.1 + (0.1)(1)(1\text{-}0) = 0.2$

Algorithm

Step 2

For each individual feature vector (row)

a. Compute perceptron value, a

$$a = \sum_{i=0}^{n}(x_i w_i) + b w_b$$

b. Determine classification, y

$$y = (a \geq threshold) ? 1 : 0$$

c. Adjust weights

$$w_a = w_c + r x_p (z - y)$$

where, w_a is the adjusted weight, w_c is the most recent/current value of the weight, and x_p pertains to the x-value of the previous feature vector

X_0	X_1	b	w_0	w_1	w_b	a	y	Z
0	0	1	0	0	0	0	0	0
0	1	1	0	0	0	0	0	1
1	0	1	0	0.1	0.1	0.1	0	1
1	1	1	0.1	0.1	0.2			1

$W_0 = 0 + (0.1)(1)(1\text{-}0) = 0.1$

$W_1 = 0.1 + (0.1)(0)(1\text{-}0) = 0.1$

$W_b = 0.1 + (0.1)(1)(1\text{-}0) = 0.2$

Algorithm

Step 2

For each individual feature vector (row)

a. Compute perceptron value, a

$$a = \sum_{i=0}^{n}(x_i w_i) + bw_b$$

b. Determine classification, y

$$y = (a \geq threshold)\,?\,1:0$$

c. Adjust weights

$$w_a = w_c + rx_p(z - y)$$

where, w_a is the adjusted weight, w_c is the most recent/current value of the weight, and x_p pertains to the x-value of the previous feature vector

X_0	X_1	b	w_0	w_1	w_b	a	y	Z
0	0	1	0	0	0	0	0	0
0	1	1	0	0	0	0	0	1
1	0	1	0	0.1	0.1	0.1	0	1
1	1	1	0.1	0.1	0.2			1

a = 1(0.1) + 1(0.1) + 1(0.2)

a = 0.4

Algorithm

Step 2

For each individual feature vector (row)

a. Compute perceptron value, a

$$a = \sum_{i=0}^{n}(x_i w_i) + b w_b$$

b. Determine classification, y

$$y = (a \geq threshold) ? 1 : 0$$

c. Adjust weights

$$w_a = w_c + r x_p (z - y)$$

where, w_a is the adjusted weight, w_c is the most recent/current value of the weight, and x_p pertains to the x-value of the previous feature vector

X_0	X_1	b	w_0	w_1	w_b	a	y	Z
0	0	1	0	0	0	0	0	0
0	1	1	0	0	0	0	0	1
1	0	1	0	0.1	0.1	0.1	0	1
1	1	1	0.1	0.1	0.2	0.4		1

a = 1(0.1) + 1(0.1) + 1(0.2)

a = 0.4

Algorithm

Step 2

For each individual feature vector (row)

a. Compute perceptron value, a

$$a = \sum_{i=0}^{n}(x_i w_i) + b w_b$$

b. Determine classification, y

$$y = (a \geq threshold)\,?\,1:0$$

c. Adjust weights

$$w_a = w_c + r x_p (z - y)$$

where, w_a is the adjusted weight, w_c is the most recent/current value of the weight, and x_p pertains to the x-value of the previous feature vector

X_0	X_1	b	w_0	w_1	w_b	a	y	Z
0	0	1	0	0	0	0	0	0
0	1	1	0	0	0	0	0	1
1	0	1	0	0.1	0.1	0.1	0	1
1	1	1	0.1	0.1	0.2	0.4		1

a = 0.4 t = 0.5

0.4 > 0.5? False

Therefore, y = 0

Algorithm

Step 2

For each individual feature vector (row)

a. Compute perceptron value, a

$$a = \sum_{i=0}^{n}(x_i w_i) + b w_b$$

b. Determine classification, y

$$y = (a \geq threshold)\,?\,1:0$$

c. Adjust weights

$$w_a = w_c + r x_p (z - y)$$

where, w_a is the adjusted weight, w_c is the most recent/current value of the weight, and x_p pertains to the x-value of the previous feature vector

X_0	X_1	b	w_0	w_1	w_b	a	y	Z
0	0	1	0	0	0	0	0	0
0	1	1	0	0	0	0	0	1
1	0	1	0	0.1	0.1	0.1	0	1
1	1	1	0.1	0.1	0.2	0.4	0	1

a = 0.4 t = 0.5

0.4 > 0.5? False

Therefore, y = 0

Algorithm

Step 2

For each individual feature vector (row)

a. Compute perceptron value, a

$$a = \sum_{i=0}^{n}(x_i w_i) + b w_b$$

b. Determine classification, y

$$y = (a \geq threshold)\,?\,1:0$$

c. Adjust weights

$$w_a = w_c + r x_p (z - y)$$

where, w_a is the adjusted weight, w_c is the most recent/current value of the weight, and x_p pertains to the x-value of the previous feature vector

X_0	X_1	b	w_0	w_1	w_b	a	y	Z
0	0	1	0	0	0	0	0	0
0	1	1	0	0	0	0	0	1
1	0	1	0	0.1	0.1	0.1	0	1
1	1	1	0.1	0.1	0.2	0.4	0	1

a = 0.4 t = 0.5

0.4 > 0.5? False

Therefore, y = 0

Our iteration is not finished yet

Algorithm

Step 2

For each individual feature vector (row)

a. Compute perceptron value, a

$$a = \sum_{i=0}^{n}(x_i w_i) + b w_b$$

b. Determine classification, y

$$y = (a \geq threshold) ? 1 : 0$$

c. Adjust weights

$$w_a = w_c + r x_p (z - y)$$

where, w_a is the adjusted weight, w_c is the most recent/current value of the weight, and x_p pertains to the x-value of the previous feature vector

X_0	X_1	b	w_0	w_1	w_b	a	y	Z
0	0	1	0	0	0	0	0	0
0	1	1	0	0	0	0	0	1
1	0	1	0	0.1	0.1	0.1	0	1
1	1	1	0.1	0.1	0.2	0.4	0	1

$W_0 = 0.1 + (0.1)(1)(1\text{-}0) = 0.2$

$W_1 = 0.1 + (0.1)(1)(1\text{-}0) = 0.2$

$W_b = 0.2 + (0.1)(1)(1\text{-}0) = 0.3$

Final adjusted weights

Algorithm

Step 3

if (weights **converge**):

stop learning

else:

repeat step 2 for the next feature vector.

The weights have converged if they stay the **same** throughout all feature vectors in the training data set.

X_0	X_1	b	w_0	w_1	w_b	a	y	Z
0	0	1	0	0	0	0	0	0
0	1	1	0	0	0	0	0	1
1	0	1	0	0.1	0.1	0.1	0	1
1	1	1	0.1	0.1	0.2	0.4	0	1
			0.2	0.2	0.3			

Final adjusted weights

Algorithm

Step 3

if (weights **converge**):

stop learning

else:

repeat step 2 for the next feature vector.

The weights have converged if they stay the **same** throughout all feature vectors in the training data set.

X_0	X_1	b	w_0	w_1	w_b	a	y	Z
0	0	1	0	0	0	0	0	0
0	1	1	0	0	0	0	0	1
1	0	1	0	0.1	0.1	0.1	0	1
1	1	1	0.1	0.1	0.2	0.4	0	1
			0.2	0.2	0.3			

Do these weights converge?

No. Repeat step 2.

Algorithm

Iteration 1

X_0	X_1	b	w_0	w_1	w_b	a	y	Z
0	0	1	0	0	0	0	0	0
0	1	1	0	0	0	0	0	1
1	0	1	0	0.1	0.1	0.1	0	1
1	1	1	0.1	0.1	0.2	0.4	0	1
			0.2	0.2	0.3			

Iteration 2

X_0	X_1	b	w_0	w_1	w_b	a	y	Z
0	0	1	0.2	0.2	0.3			0
0	1	1						1
1	0	1						1
1	1	1						1

Bring final adjusted weights as initial weights for the next iteration. Then, repeat step 2

Algorithm

Step 2

For each individual feature vector (row)

a. Compute perceptron value, a

$$a = \sum_{i=0}^{n}(x_i w_i) + b w_b$$

b. Determine classification, y

$$y = (a \geq threshold) ? 1 : 0$$

c. Adjust weights

$$w_a = w_c + r x_p (z - y)$$

where, w_a is the adjusted weight, w_c is the most recent/current value of the weight, and x_p pertains to the x-value of the previous feature vector

X_0	X_1	b	w_0	w_1	w_b	a	y	Z
0	0	1	0.2	0.2	0.3			0
0	1	1						1
1	0	1						1
1	1	1						1

Start of our iteration

Algorithm

Step 2

For each individual feature vector (row)

a. Compute perceptron value, a

$$a = \sum_{i=0}^{n} (x_i w_i) + b w_b$$

b. Determine classification, y

$$y = (a \geq threshold) ? 1 : 0$$

c. Adjust weights

$$w_a = w_c + r x_p (z - y)$$

where, w_a is the adjusted weight, w_c is the most recent/current value of the weight, and x_p pertains to the x-value of the previous feature vector

X_0	X_1	b	w_0	w_1	w_b	a	y	Z
0	0	1	0.2	0.2	0.3			0
0	1	1						1
1	0	1						1
1	1	1						1

a = 0(0.2) + 0(0.2) + 1(0.3)

a = 0.3

Algorithm

Step 2

For each individual feature vector (row)

a. Compute perceptron value, a

$$a = \sum_{i=0}^{n}(x_i w_i) + bw_b$$

b. Determine classification, y

$$y = (a \geq threshold)\,?\,1:0$$

c. Adjust weights

$$w_a = w_c + rx_p(z - y)$$

where, w_a is the adjusted weight, w_c is the most recent/current value of the weight, and x_p pertains to the x-value of the previous feature vector

X_0	X_1	b	w_0	w_1	w_b	a	y	Z
0	0	1	0.2	0.2	0.3	0.3		0
0	1	1						1
1	0	1						1
1	1	1						1

a = 0(0.2) + 0(0.2) + 1(0.3)

a = 0.3

Algorithm

Step 2

For each individual feature vector (row)

a. Compute perceptron value, a

$$a = \sum_{i=0}^{n}(x_i w_i) + b w_b$$

b. Determine classification, y

$$\boxed{y = (a \geq threshold)\,?\,1:0}$$

c. Adjust weights

$$w_a = w_c + r x_p (z - y)$$

where, w_a is the adjusted weight, w_c is the most recent/current value of the weight, and x_p pertains to the x-value of the previous feature vector

X_0	X_1	b	w_0	w_1	w_b	a	y	Z
0	0	1	0.2	0.2	0.3	0.3		0
0	1	1						1
1	0	1						1
1	1	1						1

a = 0.3   t = 0.5

0.3 > 0.5? False

Therefore, y = 0

Algorithm

Step 2

For each individual feature vector (row)

a. Compute perceptron value, a

$$a = \sum_{i=0}^{n}(x_i w_i) + b w_b$$

b. Determine classification, y

$$y = (a \geq threshold)\,?\,1:0$$

c. Adjust weights

$$w_a = w_c + r x_p (z - y)$$

where, w_a is the adjusted weight, w_c is the most recent/current value of the weight, and x_p pertains to the x-value of the previous feature vector

X_0	X_1	b	w_0	w_1	w_b	a	y	Z
0	0	1	0.2	0.2	0.3	0.3	0	0
0	1	1						1
1	0	1						1
1	1	1						1

a = 0.3 t = 0.5

0.3 > 0.5? False

Therefore, y = 0

Algorithm

Step 2

For each individual feature vector (row)

a. Compute perceptron value, a

$$a = \sum_{i=0}^{n}(x_i w_i) + b w_b$$

b. Determine classification, y

$$y = (a \geq threshold)\,?\,1:0$$

c. Adjust weights

$$w_a = w_c + r x_p (z - y)$$

where, w_a is the adjusted weight, w_c is the most recent/current value of the weight, and x_p pertains to the x-value of the previous feature vector

X_0	X_1	b	w_0	w_1	w_b	a	y	Z
0	0	1	0.2	0.2	0.3	0.3	0	0
0	1	1						1
1	0	1						1
1	1	1						1

W_0 = 0.2 + (0.1)(0)(0-0) = 0.2

W_1 = 0.2 + (0.1)(0)(0-0) = 0.2

W_b = 0.3 + (0.1)(1)(0-0) = 0.3

Algorithm

Step 2

For each individual feature vector (row)

a. Compute perceptron value, a

$$a = \sum_{i=0}^{n} (x_i w_i) + b w_b$$

b. Determine classification, y

$$y = (a \geq threshold) ? 1 : 0$$

c. Adjust weights

$$w_a = w_c + r x_p (z - y)$$

where, w_a is the adjusted weight, w_c is the most recent/current value of the weight, and x_p pertains to the x-value of the previous feature vector

X_0	X_1	b	w_0	w_1	w_b	a	y	Z
0	0	1	0.2	0.2	0.3	0.3	0	0
0	1	1	0.2	0.2	0.3			1
1	0	1						1
1	1	1						1

W_0 = 0.2 + (0.1)(0)(0-0) = 0.2

W_1 = 0.2 + (0.1)(0)(0-0) = 0.2

W_b = 0.3 + (0.1)(1)(0-0) = 0.3

Algorithm

Step 2

For each individual feature vector (row)

a. Compute perceptron value, a

$$a = \sum_{i=0}^{n}(x_i w_i) + bw_b$$

b. Determine classification, y

$$y = (a \geq threshold)\,?\,1:0$$

c. Adjust weights

$$w_a = w_c + rx_p(z - y)$$

where, w_a is the adjusted weight, w_c is the most recent/current value of the weight, and x_p pertains to the x-value of the previous feature vector

X_0	X_1	b	w_0	w_1	w_b	a	y	Z
0	0	1	0.2	0.2	0.3	0.3	0	0
0	1	1	0.2	0.2	0.3			1
1	0	1						1
1	1	1						1

Continue to do this until the weights converge.

Algorithm

Iteration 2

X_0	X_1	b	w_0	w_1	w_b	a	y	Z
0	0	1	0.2	0.2	0.3	0.3	0	0
0	1	1	0.2	0.2	0.3	0.5	0	1
1	0	1	0.2	0.3	0.4	0.6	1	1
1	1	1	0.2	0.3	0.4	0.9	1	1
			0.2	0.3	0.4	0.4		

Iteration 3

X_0	X_1	b	w_0	w_1	w_b	A	y	Z
0	0	1						0
0	1	1						1
1	0	1						1
1	1	1						1

Algorithm

Iteration 2

X_0	X_1	b	w_0	w_1	w_b	a	y	Z
0	0	1	0.2	0.2	0.3	0.3	0	0
0	1	1	0.2	0.2	0.3	0.5	0	1
1	0	1	0.2	0.3	0.4	0.6	1	1
1	1	1	0.2	0.3	0.4	0.9	1	1
			0.2	0.3	0.4	0.4		

Iteration 3

X_0	X_1	b	w_0	w_1	w_b	A	y	Z
0	0	1	0.2	0.3	0.4			0
0	1	1						1
1	0	1						1
1	1	1						1

Bring final adjusted weights as initial weights for the next iteration. Then, repeat step 2

Algorithm

Iteration 2

X_0	X_1	b	w_0	w_1	w_b	a	y	Z
0	0	1	0.2	0.2	0.3	0.3	0	0
0	1	1	0.2	0.2	0.3	0.5	0	1
1	0	1	0.2	0.3	0.4	0.6	1	1
1	1	1	0.2	0.3	0.4	0.9	1	1
			0.2	0.3	0.4	0.4		

Iteration 3

X_0	X_1	b	w_0	w_1	w_b	A	y	Z
0	0	1	0.2	0.3	0.4	0.4	0	0
0	1	1	0.2	0.3	0.4	0.7	1	1
1	0	1	0.2	0.3	0.4	0.6	1	1
1	1	1	0.2	0.3	0.4	0.9	1	1
			0.2	0.3	0.4			

Did the weights converge?

YES. Stop computation

Algorithm

Are there instances where the weights do not converge?

Yes. This happens when the data is **not linearly separable**.

Content

I. Background
II. Implementation of Linear Classification Using Perceptrons

Keep safe!