Neural Network

Practical Work

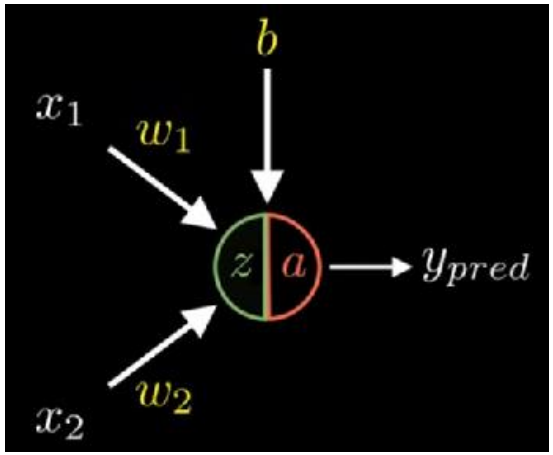
N. Houmani and S. Garcia

10/01/2024

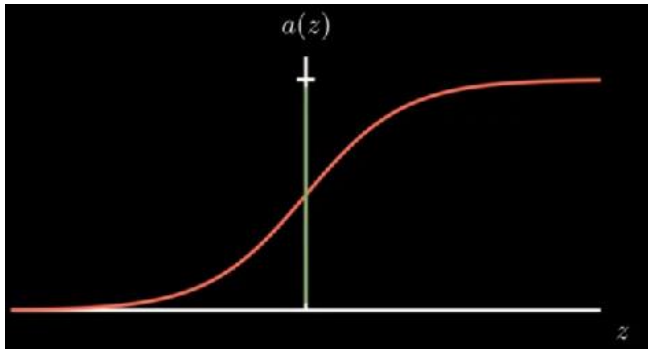
LINEAR PERCEPTRON

Linear_Perceptron.ipynb

- Generate an artificial dataset :
- Two variables : x1, x2
- 2 classes : y=0, y=1



$$z = w_1 x_1 + w_2 x_2 + b$$



$$a = \frac{1}{1 + e^{-z}}$$

- Log Loss

$$L = -\frac{1}{m} \sum_{i=1}^{m} y_i log(a_i) + (1 - y_i) log(1 - a_i)$$

- *Analyze the performance of the model and the loss function*
- *Observe the parameters of the model*
- *Predict the value of different samples*
- *Observe the decision boundary*

LINEAR PERCEPTRON

Linear_Perceptron_DogCat.ipynb

- Copy : dataset folder and utility.py in the current folder
- Load the dataset splitted into training and test sets

  - *Analyze the architecture of the model*
  - *Apply the model with learning_rate=0.1 and n_iter=200. Observation.*
  - *Change the value of the "learning_rate" and "n_iter"*
  - *Analyze the performance and loss function on the training and test sets. Conclusion.*

- Note : data should be normalized and flatten

Linear_Perceptron_Circle.ipynb

- Generate the dataset
  - *Apply the model with learning_rate=0.1 and n_iter=200.*
  - *Analyze the performance and the loss function.*
  - *Observe the decision boundary. Conclusion.*

PERCEPTRON WITH TWO LAYERS

Neural Network_Circle.ipynb

- Generate the dataset
  - Apply the model with n1=2, 4, 8 in the hidden layer.
  - *Analyze the performance and the loss function. Conclusion.*

Neural Network_DogCat.ipynb

  - Apply the model with n1= 4, 8, 32 in the hidden layer, and change the number of iterations. *Observation.*