MachLe - Résumé Olivier D'Ancona

Evaluation Metrics

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$
$$Precision = \frac{TP}{TP + FP}$$
$$Recall = \frac{TP}{TP + FN}$$
$$Specificity = \frac{TP}{TN + FP}$$
$$Fscore = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall}$$

Activation Functions

Sigmoid : $\frac{1}{1+e^{-x}}$

Hyperbolic tangent : $\frac{e^x - e^{-x}}{e^x + e^{-x}}$

Relu : $\begin{cases} 0 & \text{si } x < 0 \\ x & \text{si } x \geq 0 \end{cases}$

Gaussian : e^{-x^2}

Softmax : $\frac{e^{z_j}}{\sum_{k=1}^{K} e^{z_k}}$

Neural Network

Structure

Biais : b, An extra weight that can be learned using a learning algorithm. The purpose is to replace threshold.
Input : I, Input vector **Weights** : W, Vector of weights

Learning algorithm

1. Randomly initialize weights
2. Compute the neuron's output for a fiven input vector X
3. Update weights : $W_j(t+1) = W_j(t) + \eta\,(\hat{y_i} - y)\,x$ with η the learning rate and $\hat{y_i}$ the desired output.
4. Repeat steps 2 and 3 for the number of epochs you need or until the error is smaller than a threshold.