***Récapitulatif du cours***

Lesson 2 : Introduction to Neural Networks

*HIGHER DIMENSIONS*

Explication sur les opérations de matrice : quelles dimensions de matrice seraient adéquates pour le poids, le biais et le feature.

Une image contenant texte, tableau blanc

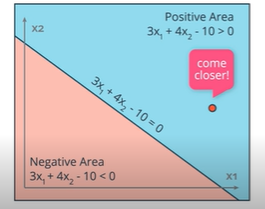
Description générée automatiquement

*PERCEPTRONS AS LOGICAL OPERATORS*

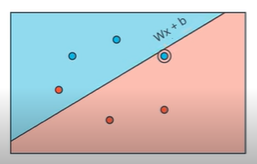
Pour passer d’un AND Perceptron à un OR Perceptron :

* Augmenter les poids
* Diminuer la magnitude du biais

*PERCEPTRON TRICK*



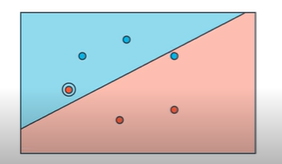
*PERCEPTRON ALGORITHM*



Si prediction = 0 : pour i allant de 1 à n

* Changer wi par wi + alpha\*xi

Changer b par b + alpha



Si prediction = 1 : pout i allant de 1 à n

* Changer wi par wi - alpha\*xi

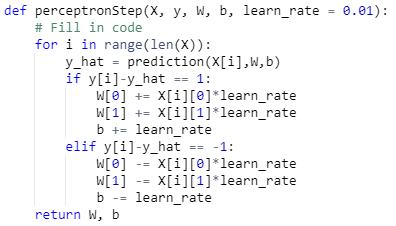
Changer b par b – alpha

La fonction permet de modifier la frontière pour rapprocher le point qui se trouverait mal classifié.

Y[i] est la classe réelle du point (celle que l’on trouve sur le graphe)

Y\_hat est la classe prédite

Ensuite, c’est le même raisonnement qu’en haut.

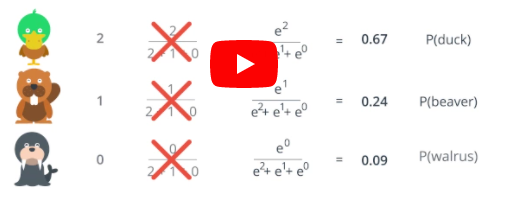


*LOGG-LOSS ERROR FUNCTION*

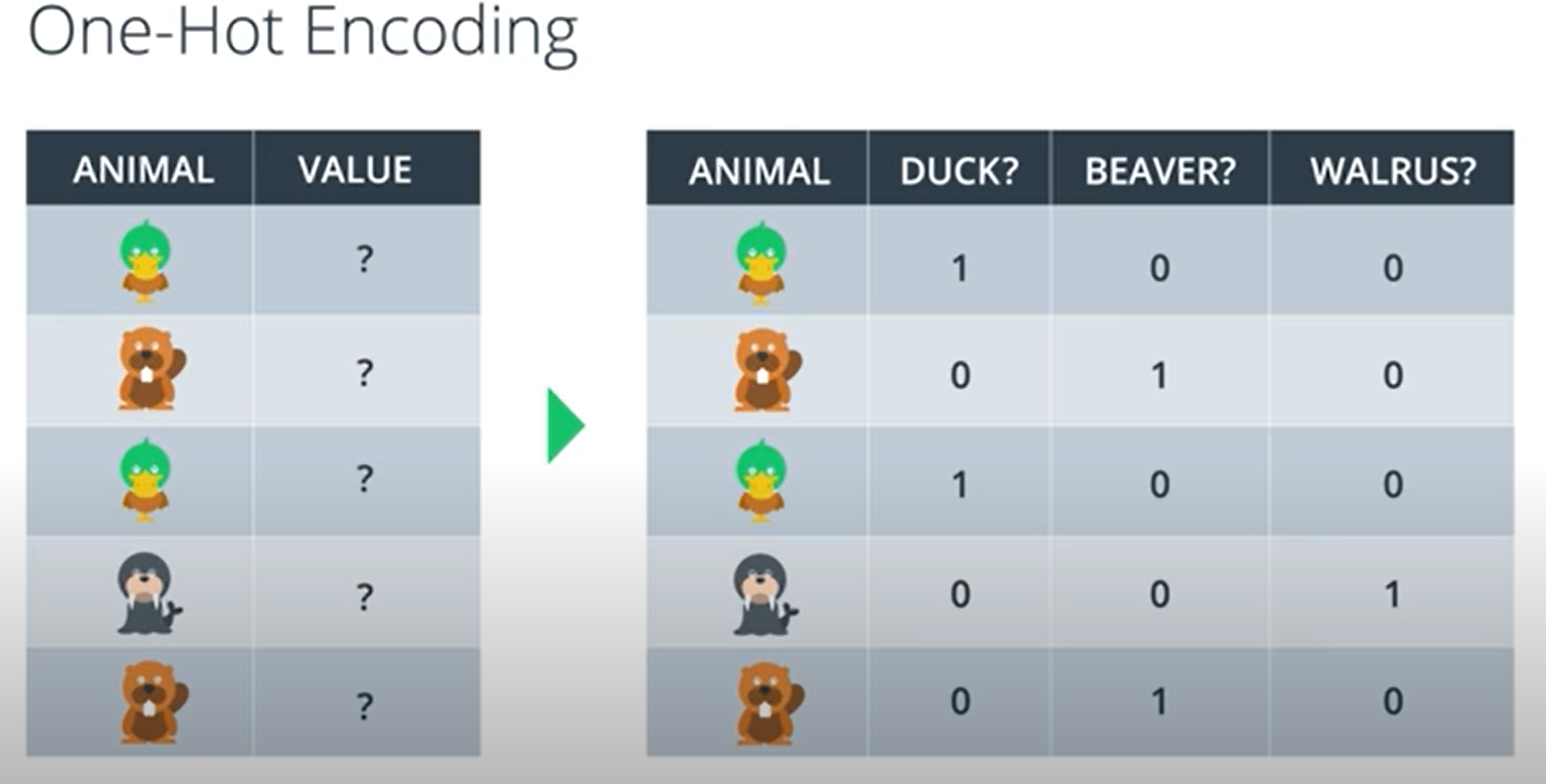
La fonction erreur doit être différentiable et continue pour être appliquer à la descente de Gradient.

*SOFTMAX*

Cette fonction permet d’arriver à une prédiction continue avec 3 classes ou plus. (Exemple : 3 animaux avec une prédiction 0, 1, 2 qui est transformé en 0.67, 0.59, 0.14)



*ONE-HOT ENCODING*



CROSS-ENTROPY

Permet de calculer l’accuracy du modèle. Plus elle est élevée, moins le modèle est précis.

