Modèle Linéaire pour classification

Classification linéaire: **Introduction**

Question: Comment adopter le modèle linéaire
à la classification?

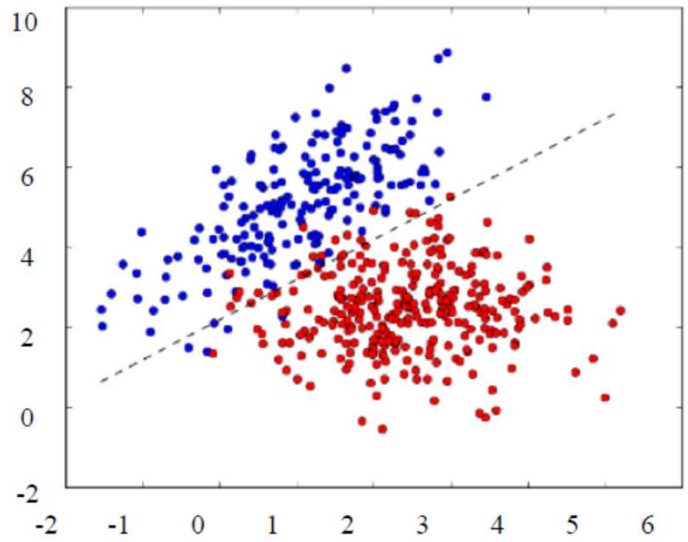
Classification linéaire: **Classification binaire**

Une classification **binaire** a **2 classes**

Dans ce genre de problème, on aura un Dataset contenant une variable target $\boldsymbol{y}$ pouvant prendre **2** valeurs seulement,

$(y' = \{-1{,}1\})$:

- $g(x) = sign(y - w^T x)$
- $d + 1\ paramètres$



Exemple:

Pour une Dataset d'emails

- si y = -1, alors l'email n'est pas un spam
- si y = 1, alors l'email est un spam

Classification linéaire: **Classification binaire**

Classification binaire: $(y' = \{-1,1\})$:

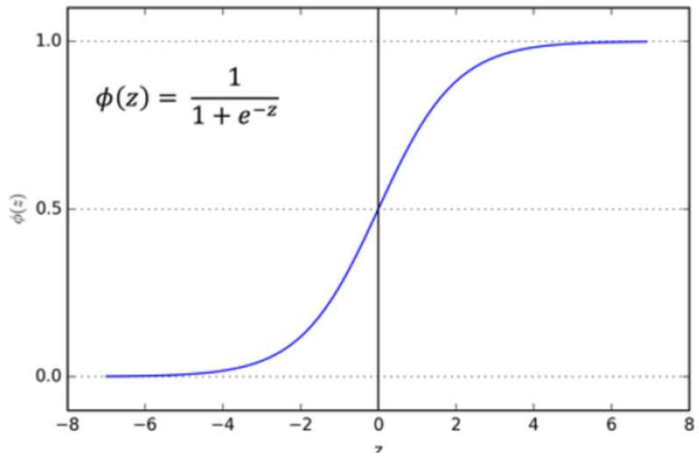
- $g(x) = sign(y - w^T x)$
- $d + 1\ paramètres$

Exemple:

Pour une Dataset d'emails

- si y = -1, alors l'email n'est pas un spam
- si y = 1, alors l'email est un spam

Classification binaire: Logistic Regression(1)

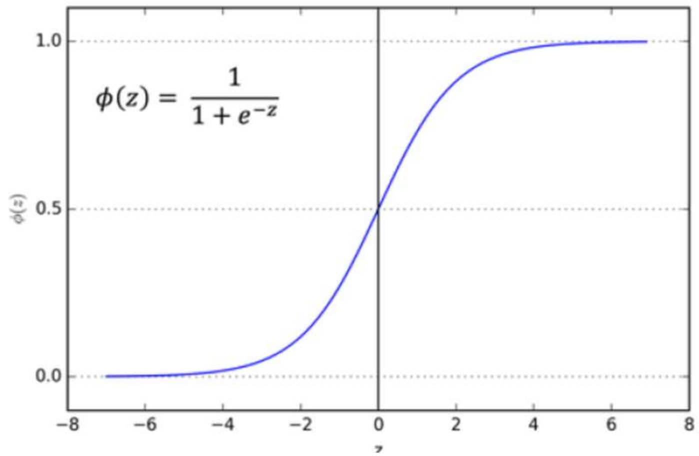


- le modèle de régression logistique est utilisé pour résoudre les problèmes **de classification binaire,**
- Le modèle est linéaire $g(x) = w^T x$
- La fonction d'activation: on applique la fonction logistique (aussi appelé fonction sigmoïde ou tout simplement sigma $\boldsymbol{\sigma}$) sur la sortie linéaire

$$\sigma(x) = \frac{1}{1 + e^{-w^T x}}$$

- Cette fonction permet de transformer le modèle linéaire en une fonction non linéaire toujours comprise en 0 et 1.

Classification binaire: **Logistic Regression(2)**



- A partir de cette fonction, il est possible de définir un **seuil,**
- Typiquement, on définit le seuil à **0.5** comme ceci :

$$\begin{cases} y = -1 \ si \ \sigma(X.\theta) < 0.5 \\ y = 1 \ si \ \sigma(X.\theta) \geq 0.5 \end{cases}$$

Classification linéaire: **Classification multiple**

- Logistic regression applies only to binary classification problems. For multi-class classification problems, use the Softmax function.

Classification linéaire: **Classification multi-classes (K classes)**

- Classification multi-classes ($y \in \{1,,,,,K\}$):

$$g(x) = \arg \max_{k \in \{1,\dots,K\}} \left(w_k^T x\right)$$

- Nombre de paramètres: $K \times d \ \left(w_k \in \mathbb{R}^d\right)$

- Exemple:

$$z = (7, -7{,}5{,}10) \quad scores$$
$$g(x) = 3$$

Classification multi- classes: **SoftMax function**

- Calculer des **scores positifs, compris entre 0 et 1 sous forme d'une distribution de probabilités** (logits) à partir du modèle linéaire:

- On applique **SoftMax transform**

$$z = (w_1^T x, \dots, w_K^T x)$$

$$\downarrow$$

$$(z_1, z_2, \dots, z_K)$$

$$\downarrow$$

$$(e^{z_1}, \dots \dots, e^{z_K})$$

$$\sigma(z) = \left(\frac{e^{z_1}}{\sum_{k=1}^{K} e^{z_k}}, \dots, \frac{e^{z_K}}{\sum_{k=1}^{K} e^{z_k}}\right)$$

SoftMax function

Classification linéaire: **softmax function**

$$\sigma(z) = \left(\frac{e^{z_1}}{\sum_{k=1}^{K} e^{z_k}}, \dots, \frac{e^{z_K}}{\sum_{k=1}^{K} e^{z_k}}\right)$$

Exemple

$z = (7, -7.5, 10)$

$\sigma(z) = (0.05, 0, 0.95)$

Classification linéaire: **Loss function?**

- output du modèle: prédiction des probabilités des classes

$$\sigma(z) = \left(\frac{e^{z_1}}{\sum_{k=1}^{K} e^{z_k}}, \dots, \frac{e^{z_K}}{\sum_{k=1}^{K} e^{z_k}}\right)$$

- La valeur des targets pour la probabilité des classes

p= $([y = 1], \dots, [y = K])$

[a]: Iverson brackets:

$$[a] = \begin{cases} 1 & \text{Si a est true} \\ 0 & \text{Si a est false} \end{cases}$$

- La similarité entre z et p peut être mesurée par **l'entropie croisée (Cross entropy**

$$-\sum_{k=1}^{K} [y = k] \log \frac{e^{z_k}}{\sum_{k=1}^{K} e^{z_k}} = -\log \frac{e^{z_y}}{\sum_{j=1}^{K} e^{z_j}}$$

Classification linéaire: **cross entropy exemple**

Suppose $K = 3$ and $y = 1$:

- $-1 * \log 1 - 0 * \log 0 - 0 * \log 0 = 0$
- $-1 * \log 0.5 - 0 * \log 0.25 - 0 * \log 0.25 \approx 0.693$
- $-1 * \log 0 - 0 * \log 1 - 0 * \log 0 = +\infty$

Classification linéaire: **cross entropy pour classification**

La fonction **cross-entropy** est différentiable et peut être utilisée comme **Loss Function**

•

$$L(w) = -\sum_{i=1}^{l}\sum_{k=1}^{K}[y_i = k] log \frac{e^{w_k^T x_i}}{\sum_{j=1}^{K} e^{w_j^T x_i}}$$

$$-\sum_{i=1}^{l} log \frac{e^{w_{y_i}^T x_i}}{\sum_{j=1}^{K} e^{w_j^T x_i}} \rightarrow \min_{w} L(w)$$

Classification linéaire: **Récapitulatif**

- Les modèles linéaires peuvent facilement généralisés pour les tâches de classification
- Régression----------→ Loss Function : MSE (Mean Squared Error)
- Classification-------→ Loss Function: Cross Entropy