RAPPORT DE PROJET

Réalisés par:

Aymane Maghouti
Abdelghafor Elgharbaoui
Ossama Outmani

Encadré par :

Pr.Asmae BOUFASSIL

Sommaire :

Partie 1 : le fondamentale théorique de la régression logistique (aspect mathématique et algorithmique)

1.0-Régression logistique (Algorithme de classification)

Le problème de la classification est un problème courant dans de nombreux domaines où l'on cherche à prédire une variable de sortie discrète en fonction d'un ensemble de variables d'entrée. Cependant, résoudre ce problème peut être complexe en raison de la nature des données d'entrée et de la complexité des interactions entre elles. Heureusement, des méthodes statistiques telles que la régression logistique ont été développées pour résoudre ce problème. La régression logistique est une technique de modélisation statistique utilisée pour prédire une variable de sortie binaire (variable cible) en fonction d'un ensemble de variables d'entrée continues ou catégorielles. Elle est souvent utilisée pour résoudre des problèmes de classification dans des domaines tels que la médecine, la finance, le marketing, etc. La régression logistique est particulièrement utile pour prédire la classe en se basant sur des données.

La régression logistique est une technique d'apprentissage supervisé, ce qui signifie qu'elle nécessite des données étiquetées pour entraîner le modèle. Le modèle est entraîné à prédire une variable de sortie binaire en fonction d'un ensemble de variables d'entrée. Les données d'entraînement pour la régression logistique comprennent des exemples de valeurs d'entrée et de sortie binaire correspondantes.

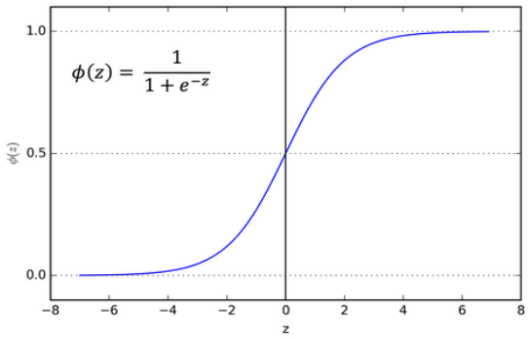
1.1-Interprétation du modèle

1.1.1- Fonction logistique

- Dans un problème de classification binaire y∈{0,1}, si nous utilisons un modèle de régression linéaire $h_\theta(x)=\theta^T x$, il est logique que $h_\theta(x)$ prenne des valeurs supérieures à 1 ou inférieures à 0.
- Le modèle de régression logistique est défini de sorte que $0 \leq h_\theta(x) \leq 1$, où $h_\theta(x)=g(\theta^T x)$.
- g(.) est la fonction logistique, également appelée fonction sigmoïde, est une courbe en forme de S qui peut prendre n'importe quel nombre à valeur réelle et le mapper en une valeur comprise entre 0 et 1, mais jamais

$$z = \theta^T x \quad ; \quad g(z) = \frac{1}{1+e^{-z}}$$

exactement à ces limites.



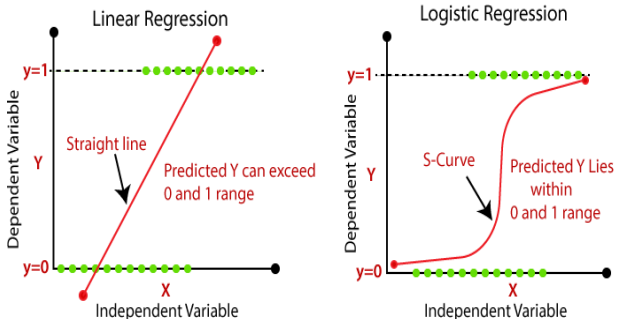
Cette fonction a la particularité d'être toujours comprise en 0 et 1 (Probabilité).

Pour coller la fonction logistique sur un Dataset $(\boldsymbol{X}, \boldsymbol{y})$ on y fait passer le produit matriciel $\boldsymbol{X}.\boldsymbol{\theta}$ ce qui nous donne le **modèle** de *Logistic Regression* :

$$\sigma(\boldsymbol{X}.\boldsymbol{\theta}) = \frac{1}{1+e^{-\boldsymbol{X}.\boldsymbol{\theta}}}$$

- $h_\theta(x)$ est la probabilité estimée que y=1 pour l'entrée x.
- $h_\theta(x)$=0,7 ➔ une probabilité de 70 % que notre sortie soit 1.

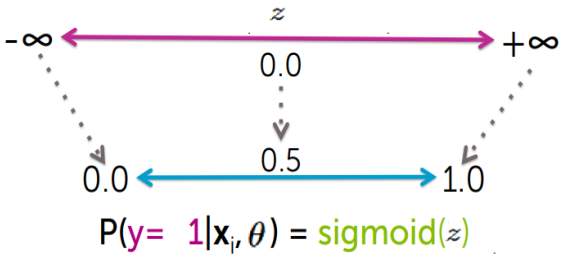
$$h_\theta(x) = P(y=1|x;\theta) = 1 - P(y=0|x;\theta)$$
$$P(y=0|x;\theta) + P(y=1|x;\theta) = 1$$



1.1.2- borne de décision

- La frontière de décision (Decision Boundary) est la ligne qui sépare la zone où y = 0 et où y = 1. Elle est créée par notre fonction d'hypothèse.
- Pour prédire une valeur discrète 0 ou 1, la sortie de la fonction d'hypothèse est traduite comme suit :

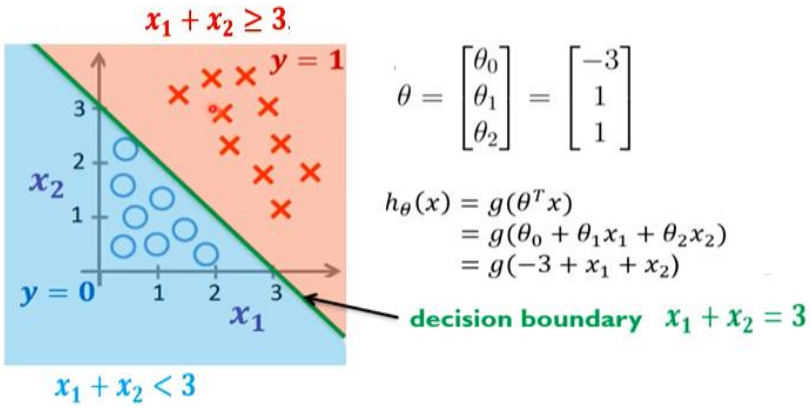
$$h_\theta(x) \geq 0.5 \rightarrow y = 1$$
$$h_\theta(x) < 0.5 \rightarrow y = 0$$



- Si notre entrée à g est $\theta^T x$, alors cela signifie :

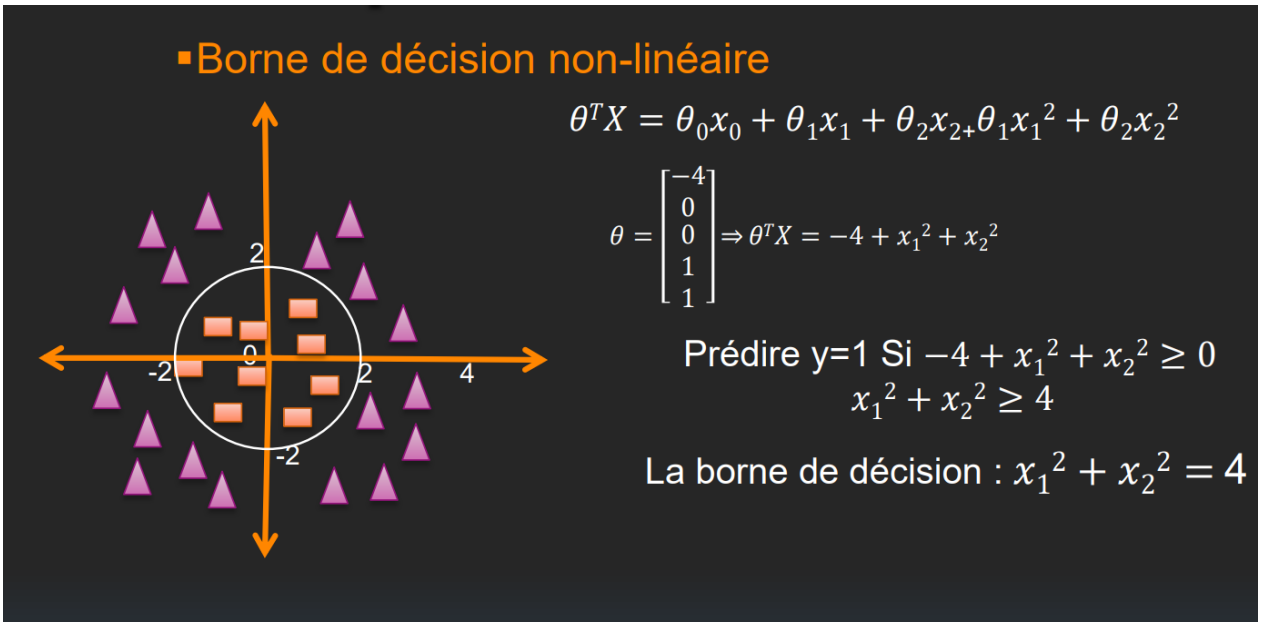
$$h_\theta(x) = g(\theta^T x) \geq 0.5 \quad when\ \theta^T x \geq 0 \quad \Rightarrow \quad \theta^T x \geq 0 \Rightarrow y = 1 \quad \theta^T x < 0 \Rightarrow y = 0$$

Exemple:



Non linear decision Boundary:

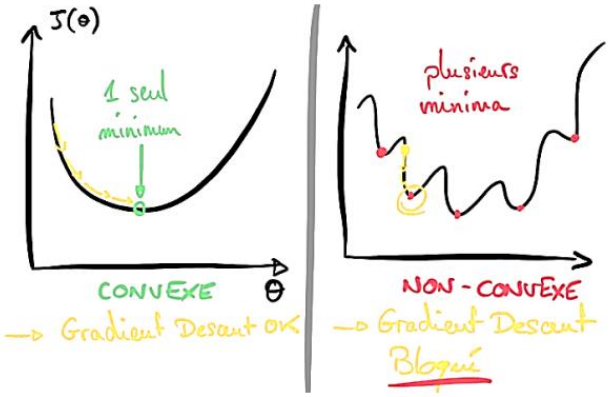
L'entrée de la fonction sigmoïde g(z) (c'est-à-dire $\theta^T x$) n'a pas besoin d'être linéaire et peut être une fonction qui décrit un cercle (par exemple $z=\theta_0+\theta_1 x_1^2+\theta_2 x_2^2$) ou n'importe quelle forme pour s'adapter à nos données.



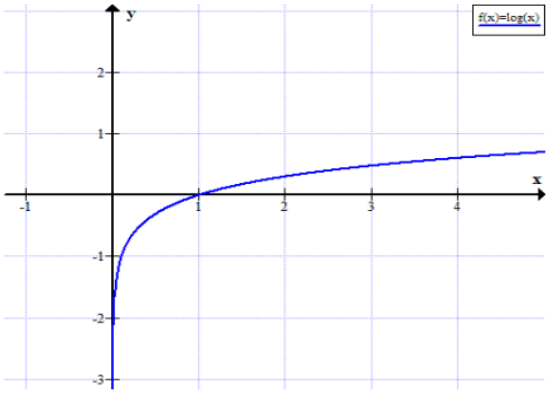
1.2-Fonction cout

1.2.1 – fonction cout pour la régression logistique

Pour la régression linéaire, la Fonction $J(\theta) = \frac{1}{2m}\sum(X.\theta - Y)^2$ Coût donnait une courbe convexe (qui présente un unique minima). C'est ce qui fait que l'algorithme de Gradient Descent fonctionne. En revanche, utiliser cette fonction pour le modèle Logistique ne donnera pas de courbe convexe (dû à la non-linéarité) et l'algorithme de Gradient Descent se bloque



Il faut donc développer une nouvelle Fonction Coût spécialement pour la régression logistique. On utilise alors la fonction logarithme pour transformer la fonction sigma en fonction convexe en séparant les cas où y = 1 des cas où y = 0.



Fonction Coût dans les cas où y = 1

Voici la Fonction Coût que l'on utilise dans les cas où y = 1 :

$$\boldsymbol{J(\theta)} = -\,\mathbf{log}\,(\boldsymbol{\sigma(X.\theta)})$$

Explications :

Si notre modèle prédit $\sigma(x)$ = 0 alors que y = 1, on doit pénaliser la machine par une grande erreur (un grand coût). La fonction logarithme permet de tracer cette courbe avec une propriété convexe, ce qui poussera le Gradient Descent à trouver les paramètres $\boldsymbol{\theta}$ pour un coût qui tend vers 0

Dans le cas ou y=1 → $Cout(h\theta\,(x), y) = -\,\mathbf{log}\,(\boldsymbol{\sigma(X.\theta)})$

Fonction Coût dans les cas où y = 0

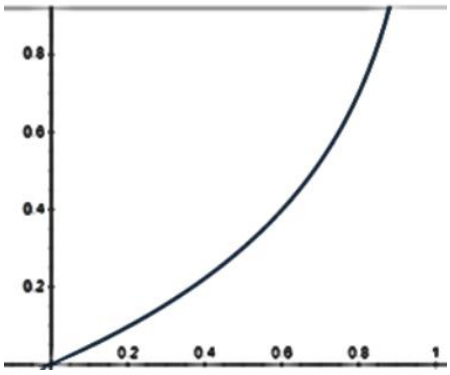
Cette fois la Fonction Coût devient :

$$\boldsymbol{J(\theta)} = -\,\mathbf{log}\,(\mathbf{1} - \boldsymbol{\sigma(X.\theta)})$$

Explications :

Si notre modèle prédit $\sigma(x)$ = 1 alors que y = 0, on doit pénaliser la machine par une grande erreur (un grand coût). Cette fois $-\,\mathbf{log}\,(\mathbf{1} - \mathbf{0})$ donne la même courbe, inversée sur l'axe vertical.

Dans le cas ou y=0 → $Cout(\mathrm{J}) = -\,\mathbf{log}\,(1\text{-}\boldsymbol{\sigma(X.\theta)})$



Fonction Coût complète

Pour écrire la Fonction Coût en une seule équation, on utilise l'astuce de séparer les cas $y = 0$ *et* $y = 1$ avec une annulation :

$$\boldsymbol{J(\theta) = \frac{-1}{m} \sum y \times \log(\sigma(X.\theta)) + (1-y) \times \log(1 - \sigma(X.\theta))}$$

Dans le cas où $y = 0$, il nous reste :

$$J(\theta) = \frac{-1}{m} \sum \cancel{0 \times \log(\sigma(X.\theta))} + 1 \times \log(1 - \sigma(X.\theta))$$

Et dans le cas où $y = 1$:

$$J(\theta) = \frac{-1}{m} \sum 1 \times \log(\sigma(X.\theta)) \cancel{+ 0 \times \log(1 - \sigma(X.\theta))}$$

1.2.2 - Descente de gradient pour la minimisation de la fonction cout

L'algorithme de Gradient Descent s'applique exactement de la même manière que pour la régression linéaire. En plus, la dérivée de la Fonction Coût est la même aussi ! On a :

$$Gradient: \frac{\partial J(\theta)}{\partial \theta} = \frac{1}{m} \sum (\sigma(X.\theta) - y).X$$

$$Gradient\ Descent: \theta = \theta - \alpha \times \frac{\partial J(\theta)}{\partial \theta}$$

Alors pour minimiser la fonction cout on fait appliquer cet algorithme :

Répéter **{** (mettre à jour les Θ_j simultanément)

$\theta_j := \theta_j - \alpha$ *Gradient;

}

C'est-à-dire :

Répéter **{** (mettre à jour les Θ_j simultanément)

$$\theta_j := \theta_j - \alpha * \frac{1}{m} \sum (\sigma(X.\theta) - y).X$$

}

Résumé de la Régression Logistique

$$\textbf{Modèle}: \sigma(X.\theta) = \frac{1}{1+e^{-X.\theta}}$$

$$\textbf{Fonction Coût}: J(\theta) = \frac{-1}{m}\sum y \times \log\big(\sigma(X.\theta)\big) + (1-y) \times \log\big(1-\sigma(X.\theta)\big)$$

$$\textbf{Gradient}: \frac{\partial J(\theta)}{\partial \theta} = \frac{1}{m} X^T.(\sigma(X.\theta) - y)$$

$$\textbf{Gradient Descent}: \theta = \theta - \alpha \times \frac{\partial J(\theta)}{\partial \theta}$$

Partie 2 : Simulation de l'algorithme

2.0- Implémentation de la régression logistique (from scratch)

Voir le code source

2.1- la construction et l'application du modèle

2.1.1- présentation des données.

Petit Description des données (la source du dataset)

(Exploration des donnes : nb_row , nb_col ….)

2.1.2 l'application de L'ACP (réduction de dimension)

L'Analyse en Composantes Principales (ACP) est une technique d'analyse statistique utilisée pour réduire la dimensionnalité d'un ensemble de données en transformant un grand nombre de variables corrélées en un petit nombre de variables non corrélées, appelées composantes principales. L'ACP est souvent utilisée pour explorer et visualiser des structures de données complexes en identifiant des modèles et des relations cachés entre les variables, et dans notre cas on vas appliquer cette algorithme pour déterminer les variables qu'ils faut prendre en considération afin de réduire la dimension (2D) et aussi pour bien visualiser les classes (Plan).

2.1.3 L'application de modelé

1- Collecte et préparation des données.

2-Division des données en ensembles d'entraînement et de test

3-Entraînement du modèle

4-Évaluation du modèle

5-Utilisation du modèle pour les prédictions

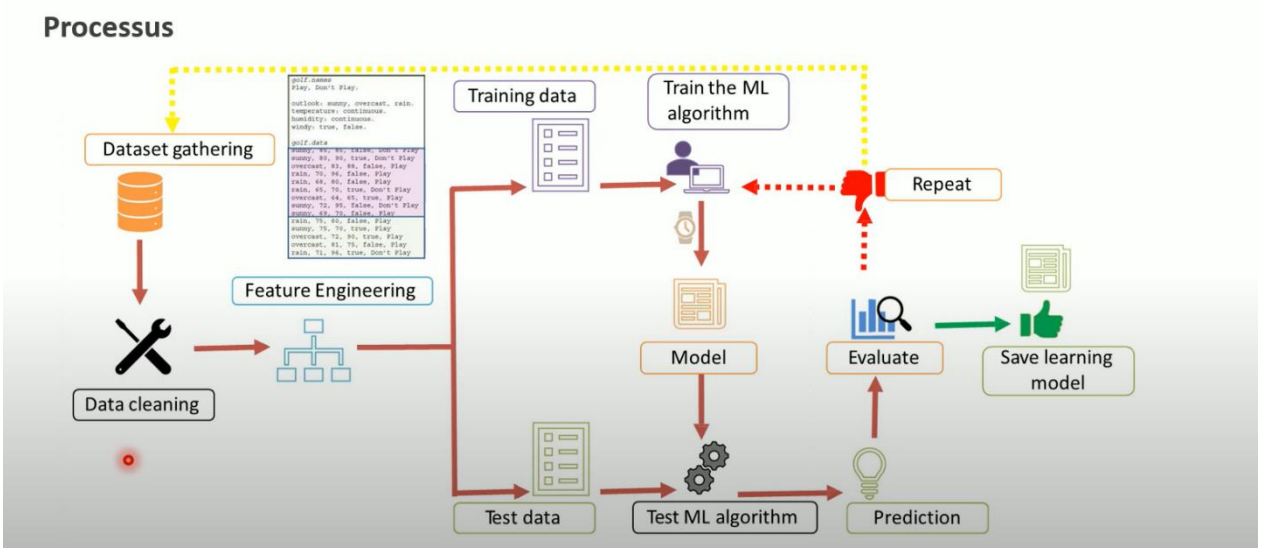
Voir le code source.

2.1.4 Évaluation de modèle (accourcie)

-matrice de confusion

- La précision

2.1.5 – Utilisation de modelé (streamlit)



2.2- Comparaison de performance de prédiction (enter sklearn et notre algorithme)

Voire le code

3-les cas Pratique de la régression logistique

La classification des données médicales.

La prédiction des risques financiers.

La détection des fraudes.

La classification des médias sociaux.