

---

## RAPPORT DE TP

### TP 4 : Machine Learning

---

Auteurs :

Y. Josué  
Japhet

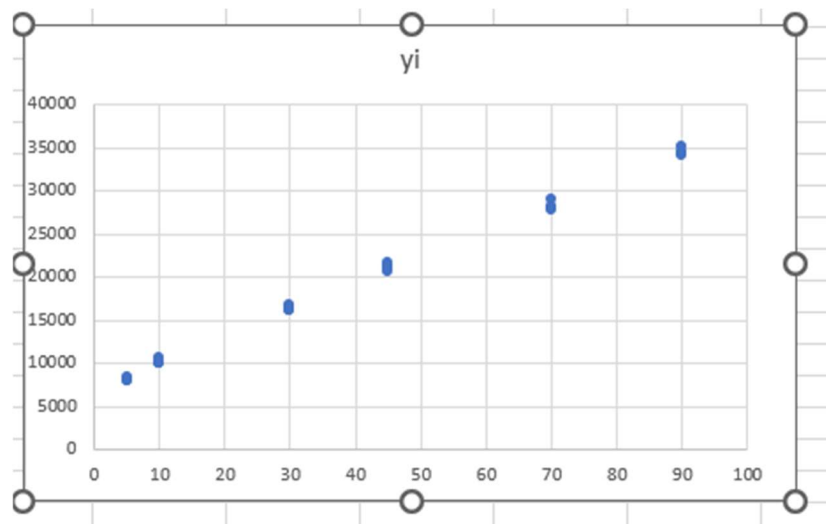
MIENGUE  
ADABADJI

Enseignant :

M. RUKUNDO Jean-  
Paul

#### - I. Régression Linéaire-

T1. Représentation graphique :



**Explication :** Le choix de la régression Linéaire est la meilleur pour prédire la relation entre deux variables  $x_i$ ,  $y_i$  dans notre cas, afin de prédire une quantité continue. De plus l'hypothèse de départ est que le phénomène a un comportement linéaire.

T2. -Critère J

$$J(\theta_0, \theta_1) = \frac{1}{48} \sum_{i=1}^{24} (H_{\theta}(x)^{(i)} - y^{(i)})^2 \text{ avec } H_{\theta}(x) = \theta_0 + \theta_1 x$$

### Algorithme

#### ● Initialisation

```
n ← 0;
θ₀⁰ ← 0;
θ₁⁰ ← 0;
m := taille de vecteur data;
erreur := 0;
lire vecteur x(1:m);
lire vecteur y(1:m);
```

#### ● Entraînement

```
do
{
  θ₀ⁿ⁺¹ := θ₀ⁿ - α ∑_{i=1}^m (h_{θ}(x^{(i)}) - y^{(i)});
  θ₁ⁿ⁺¹ := θ₁ⁿ - α ∑_{i=1}^m (h_{θ}(x^{(i)}) - y^{(i)}) x^{(i)};
  n := n + 1;
  erreur = erreur + (h_{θ}(x^{(i)}) - y^{(i)})²;
}
while (erreur > 1e - 6)
θ₀ := θ₀ⁿ⁺¹;
θ₁ := θ₁ⁿ⁺¹;
```

#### ● Prédiction

```
lire x(m+1);
y(m+1) := θ₁x(m+1) + θ₀
```

$$H_3(x) = 2xH_2(x) - 4H_1(x) = 8x^3 - 12x$$

E1. A Partir du code

E2. Résultats :

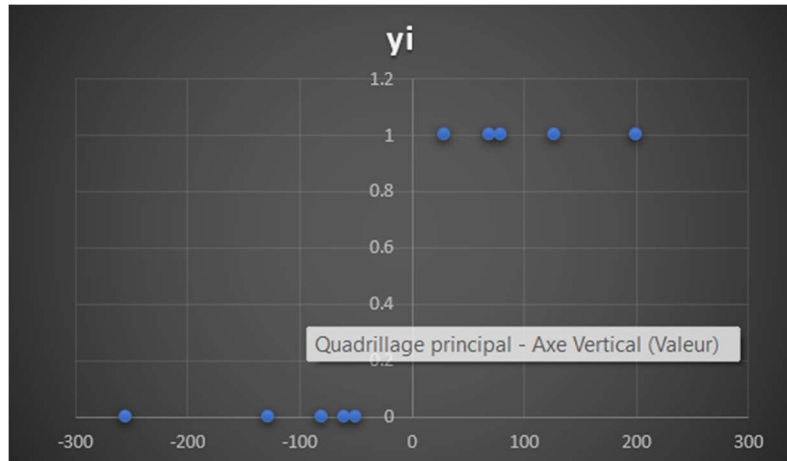
E3. Analyse des résultat après application de diffèrent taux d'apprentissage

Conclusion :

NB : Notre Carte Arduino DUE ou le câble ne marche pas pour qu'on puisse afficher les résultats.

## - II. Régression Logistique-

T3. Graphique :



**Explication** : Nous avons une variable d'entrée (x) où la variable de sortie (y) est une valeur discrète (0 ou 1).

T4. Critère J :

$$J(\theta_0, \theta_1) = \frac{1}{48} \sum_{l=1}^{24} (H_{\theta}(x)^{(i)} - y^{(i)})^2 \text{ avec } H_{\theta}(x) = \frac{1}{1 + e^{-(\theta_0 + \theta_1 x)}}$$

### Algorithme

#### ● Initialisation

```
n ← 0;
θ₀⁰ ← 0;
θ₁⁰ ← 0;
m := taille de vecteur data;
erreur := 0;
lire vecteur x(1:m);
lire vecteur y(1:m);
```

#### ● Entraînement

```
do
{
  θ₀ⁿ⁺¹ := θ₀ⁿ - (α/m) ∑_{i=1}^m (h_{θ}(x^{(i)}) - y^{(i)}) h_{θ}(x^{(i)}) (1 - h_{θ}(x^{(i)}));
  θ₁ⁿ⁺¹ := θ₁ⁿ - (α/m) ∑_{i=1}^m (h_{θ}(x^{(i)}) - y^{(i)}) h_{θ}(x^{(i)}) (1 - h_{θ}(x^{(i)})) x^{(i)};
  n := n + 1;
  erreur = erreur + (h_{θ}(x^{(i)}) - y^{(i)})²;
}
while (erreur > 1e-6)
θ₀ := θ₀ⁿ⁺¹;
θ₁ := θ₁ⁿ⁺¹;
```

#### ● Prédiction

```
lire x(m+1);
y(m+1) := h_{θ}(x(m+1))
```

E4. A Partir du code

E5. Résultats :

E6. Analyse des résultats après application de différent taux d'apprentissage et L'Objectif Erreur

Conclusion :

NB : Notre Carte Arduino DUE ou le câble ne marche pas pour qu'on puisse afficher les résultats.

Nous attestons que ce travail est original, qu'il est le fruit d'un travail commun au binôme et qu'il a été rédigé de manière autonome.

Lyon, le 23/03/2024