Rapport TP2 - Partie 2: Approximation

YMERAJ Eldis

11/2022

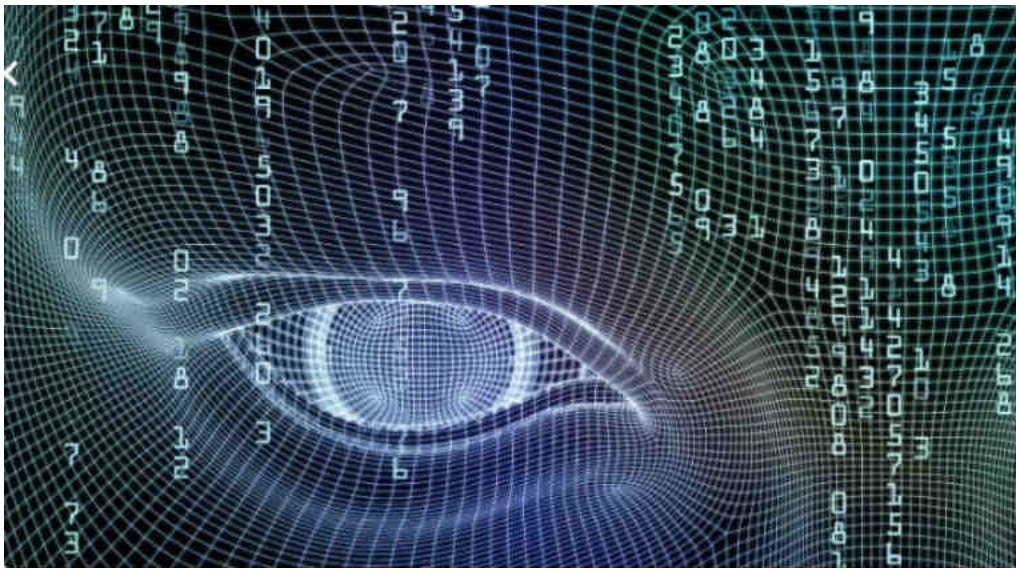

Figure 1 – Algorithmique Numérique

Table des matières

Introduction

Lorsqu'on détient un nombre fini de couples de données, et qu'on cherche à déterminer s'il y a une corrélation entre notre variable explicative (en x) et une variable expliquée (en y), on peut opter pour un modèle de régression linéaire. Ce projet explore plusieurs aspect de l'ajustement affine, résultant donc en une équation de droite de type y = ax + b. Spécifiquement, on exploite ici la méthode des moindres carrés ; par laquelle on trouve a et b tels que la somme des carrés des distances, entre les y mesurés et les y estimés, est minimisée. L'ajustement affine peut aussi être une étape dans la recherche d'une équation de type $y = b * x^a$, pour l'ajustement puissance.
Nous allons donc exposer des programmes qui mènent à ces résultats, puis nous les testeront avec des jeux d'essais prédéterminés. Nous envisagerons ensuite quelques limites de nos programmes, et quelques pistes possibles d'amélioration.

Rappel rapide - Regression linéaire
La régression linéaire par la méthode des moindres carrés nous permet de trouver une fonction affine qui décrit une relation entre une variable explicative et une variable expliquée.

$$\begin{cases} a = \dfrac{\sum_{i=0}^{N-1}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=0}^{N-1}(x_i - \bar{x})} \\ b = \bar{y} - a\bar{x} \end{cases} \quad \text{ou} \quad \bar{x} = \frac{\sum_{i=0}^{N-1} x_i}{N}, \qquad \bar{y} = \frac{\sum_{i=0}^{N-1} y_i}{N}$$

Figure 2 – Formule de regression linéaire

Un des problèmes les plus fréquents qui se posent à un expérimentateur est de trouver une formule analytique pour décrire une fonction dont il a mesuré en un certains nombres d'abscisses des valeurs expérimentales f[i] = f (xi) où f est inconnue. On peut songer à utiliser l'interpolation polynomiale mais en général les résultats sont peu satisfaisants. L'interpolation dans ce cas consiste à approcher la courbe liant deux séries de mesures par un polynôme. Les coefficients optimaux de ce polynôme sont ceux qui minimisent la variance de l'erreur d'interpolation. Ce principe est connu sous le nom de la méthode des moindres carrés.

Rappel rapide - Ajustement puissance L'ajustement puissance utilise également la régression linéaire et la méthode des moindres carrés. En effet, l'équation d'une droite peut se rapporter à l'équation d'une courbe puissance après un changement d'échelle : on prend une courbe de type ln(y) = a ln(x) + ln(b), qui donne ensuite $y = b * x^a$.

1 Implémentation des fonctions en langage C

1.1 Fonctions

linaire(...) :

- Fonction qui permet de nous aider a trouver la droite de regression lineaire.
- Type de retour de type void –> la fonction renverra rien.
- Entrée : un tableau x et un tableau y de dimension 1 qu'ils prennent les valeurs de points testees, deux pointeures a, b de type double et n le nombre de points testees.
- En sortie : La fonction va nous retourner rien car il est de type void.

Notre fonctions utilise deux tableaux de type double x et y qui représenten une coordonnée (xi, yi).
Signature de fonction :

```
void linaire (double *x ,double *y ,double* a,double* b, int n)
```

La fonction lineaire(...) modifie des nombres a et b par effet de bord grâce à leurs pointeurs.
On commence par calculer la moyenne des x et la moyenne des y, qu'on désigne par xB et yB respectivement.

```
for(int i = 0;i < n; i++)
    {
        xB+= x[i]; //calcul de X barre
        yB+= y[i]; //calcul de Y barre
    }
```

Ensuite on calcule le nominateur et le dominateyr de a.

```
for(int i=0;i<n;i++)
    {
        as1+= (x[i] - xB/n)*(y[i] - yB/n); //calcul du nominateur de a
        as2+= (x[i]- xB/n)*(x[i] - xB/n); //calcul du denominateur de a
    }
    as = as1/as2;
    bs = yB/n - as*(xB/n);
```

Il ne nous reste enfin plus qu'à trouver l'intersection de notre droite grâce à la pente a et aux moyennes xB et yB.

```
    as = as1/as2;
    bs = yB/n - as*(xB/n);

    *a = as;
    *b = bs;
```

puis(...) :

- Fonction qui realise l'ajustement puissance de type $y = a * x^b$.
- Type de retour de type void.
- Entrée : un tableau x et un tableau y de dimension 1 qu'ils prennent les valeurs de points testees, deux pointeures a, b de type double et n le nombre de points testees.
- En sortie : La fonction va nous retourner rien car il est de type void.

Notre fonction puis(...) modifie des nombres a et b par effet de bord grâce à leurs pointeurs.
Signature de fonction :

```
1  void puis(double *x, double *y, double *a, double* b, int n )
```

Au debut on remplace toutes les valeurs de x par ln(x) et toutes les valeurs de y par ln(y) et ensuite on calcule la moyenne de x et y qu'on les designe par xB et yB.

```
1  for(int i=0;i<n;i++)
2      {
3          xB += log(x[i]); //somme de ln(x[i])
4          yB += log(y[i]); // somme de ln(y[i])
5      }
```

Ensuite on fait les meme calcules qu'on a fait dans la fonction lineaire(...) pour calculer le nominateur et dominateur de a.

```
1  for(int i=0;i<n;i++)
2      {
3          as1+= (log(x[i]) - xB/n)*(log(y[i]) - yB/n);
4          as2+= (log(x[i]) - xB/n)*(log(x[i]) - xB/n);
5      }
```

Enfin, après avoir transformé b en son exponentielle, les a et b obtenus complètent l'équation $y = b * x^a$.

```
1      as = as1/as2;
2      bs = yB/n-as*(xB/n);
3      bs = exp(bs); //on remplace b par e^b
4      *a = bs;
5      *b = as;
```

La fonction affCoord(...) :

Cette fonction permet de nous afficher toutes les points qu'on teste.
Signature :

```
1     void affCoord(double *x, double *y, int n)
```

1.1.1 La fonction Main

Dans la fonction main on a fait toutes les jeux d'essais et aussi le programme va nous afficher toutes les résultats de jeux d'essais.

2 Jeux d'essais

2.1 Jeu d'essai 4.1

Densité (D) de l'eau en fonction de la température (T)
Pour le jeu d'essai 4.1 on a obtenu la droite de regression suivante : f(x) = -0.000186*X+ 1.001280

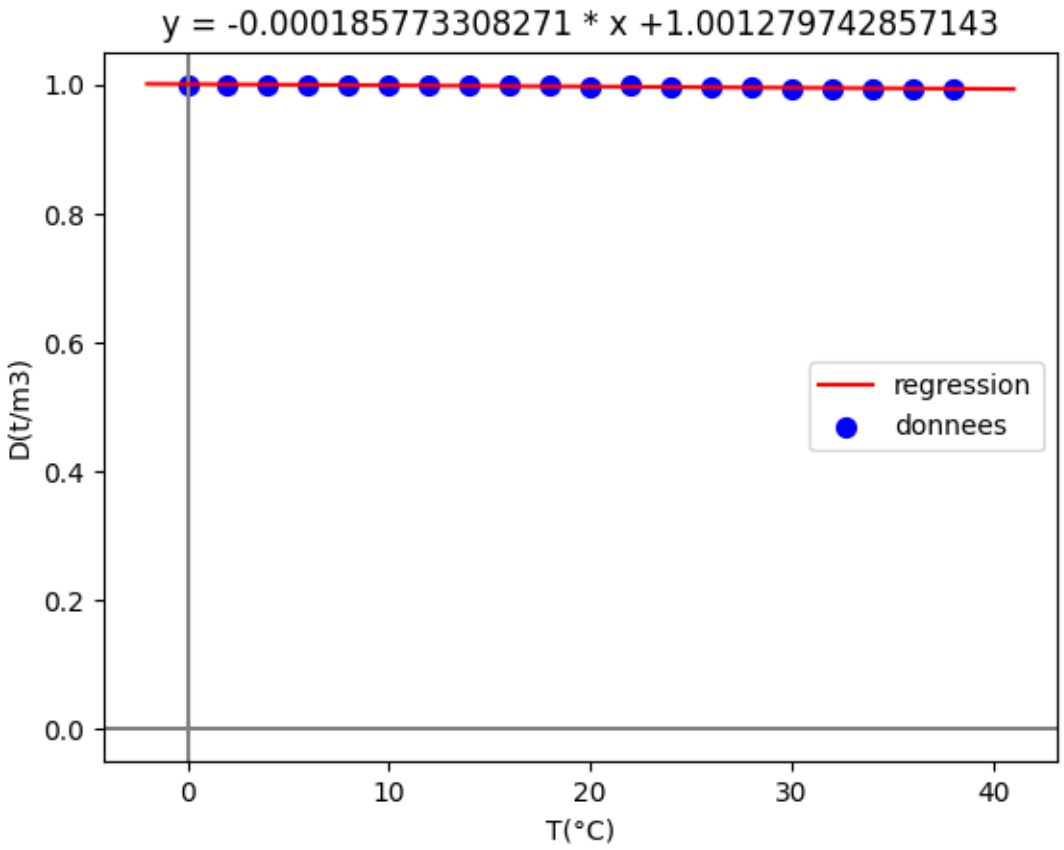

Figure 3 – f(x) = -0.000186*X+ 1.001280(s = 0.00091 ; jeu d'essai 4.1)

On observe, avec les données en 4.1, qu'un regression linéaire semble idéal , et et si on calcule l'écart type sur l'echantillon, on voit qu'il est raisonnablement petit. L'écart type = 0.00091.

2.2 Jeu d'essai 4.2

Dépenses mensuelles et revenus
Pour le jeu d'essai 4.1 on a obtenu la droite de regression suivante : f(x) = 0.324356*x - 112.658491

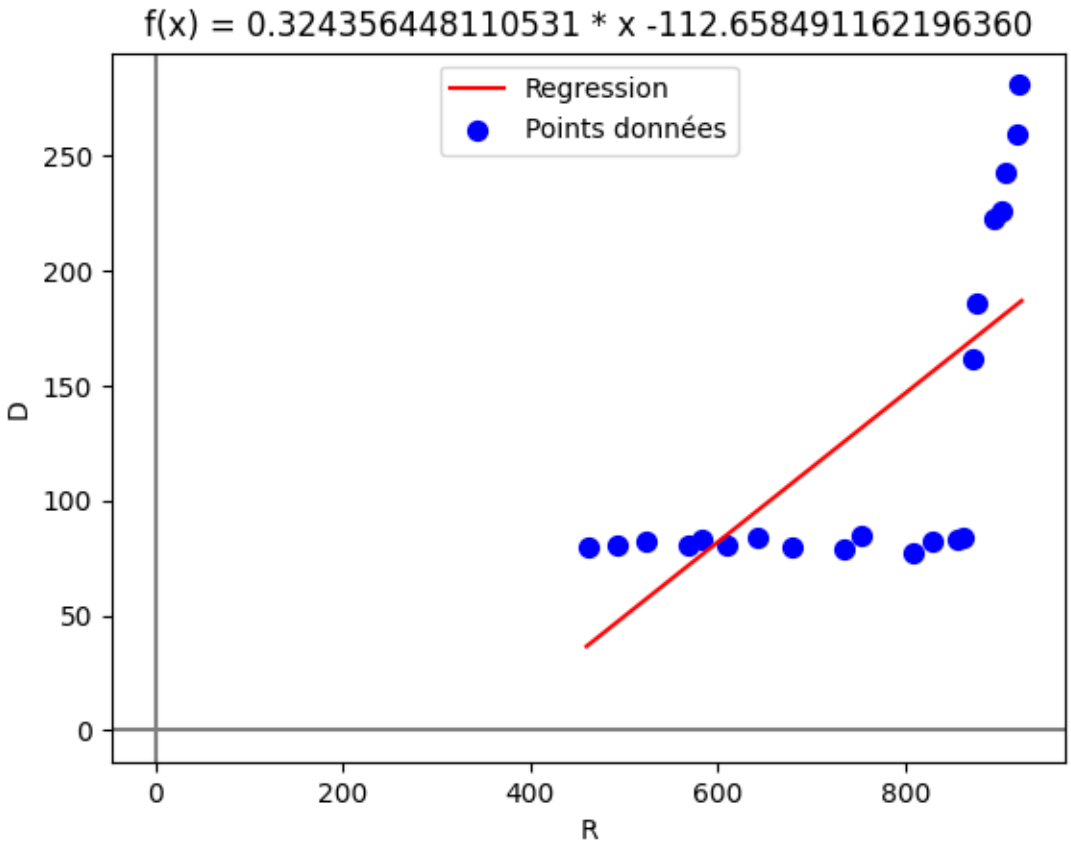

Figure 4 – f(x) = 0.324356*x - 112.658491 (s = 52.98 ; jeu d'essai 4.2)

Visuellement, le jeu de donnée en 4.2 suggère une relation exponentielle plutôt que linéaire, ou des erreurs dans le prélèvement des données. On a tenté de ramener le jeu à un nuage de point qui pourrait correspondre à une droite de type y = exp(ax+b), en soustrayant min(y) à chaque y avant d'appliquer un ajustement exponentiel au tableau, et mon idée s'est avérée mal-fondée, avec un écart-type = 52.98242. Sans être idéale, la régression linéaire offre une meilleure approximation. Il semblerait qu'il y ait une relation linéaire positive entre les revenus et les dépenses de loisirs.

2.3 Jeu d'essai 4.3

Série S dûe à Anscombe
Pour le jeu d'essai 4.1 on a obtenu la droite de regression suivante : f(x) = 0.500091*x + 3000091

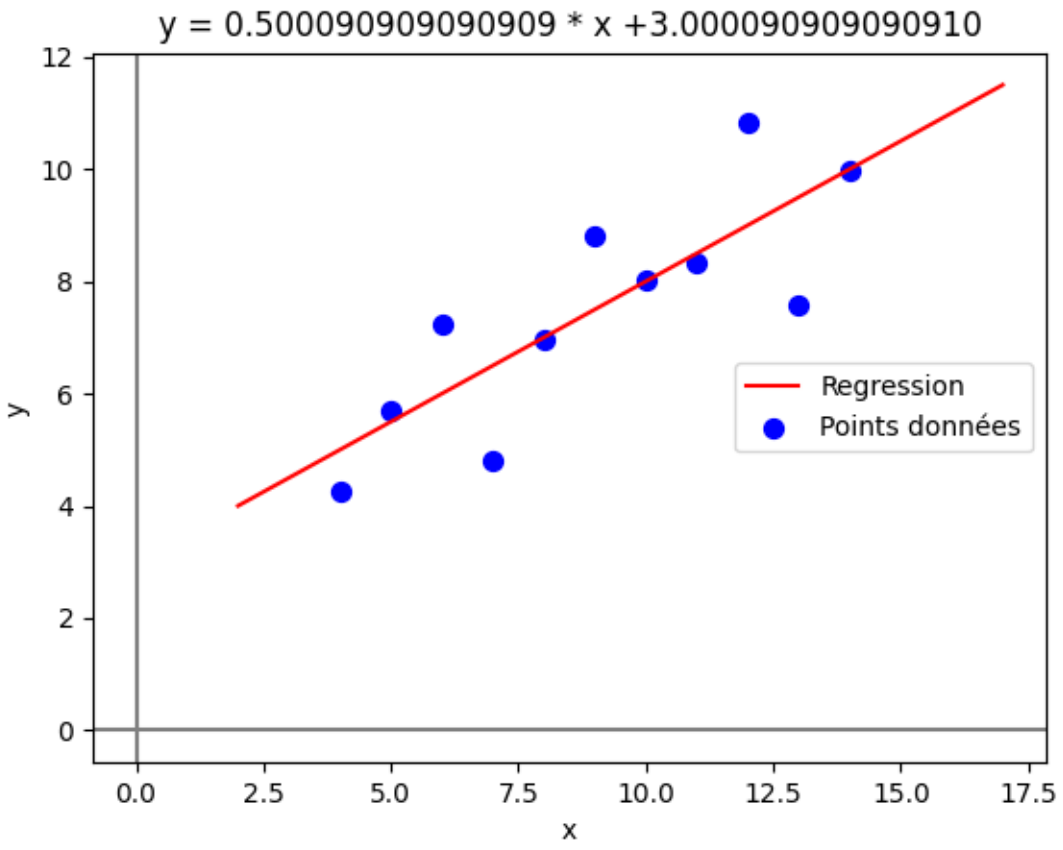

Figure 5 – f(x) = 0.500091*x + 3000091 (s = 1.7814 ; jeu d'essai 4.3)

Les données de 4.3 ressemblent au nuage de points typique qui montre une relation positive linéaire moyennement forte entre des x et y. En considérant l'amplitude des y, un écart type d'échantillon de 1.17814 (Cf. s dans chaque sous-titre)suggère que le programme a réussi à trouver une équation de droite satisfaisante.

2.4 Jeu d'essai 4.4

Vérification de la loi de Pareto
Le jeu de données en 4.4 nous invite à vérifier la loi de Pareto avec un ajustement puissance. Si $y = A/x^a$, alors A = 696600.11656 et a = 2.51719 pour la période et la région considérée, avec x les revenus et y le nombre de personnes ayant un revenu supérieur à x. Vu l'amplitude des y (351.9), un écart-type de 2.86079 est satisfaisant.

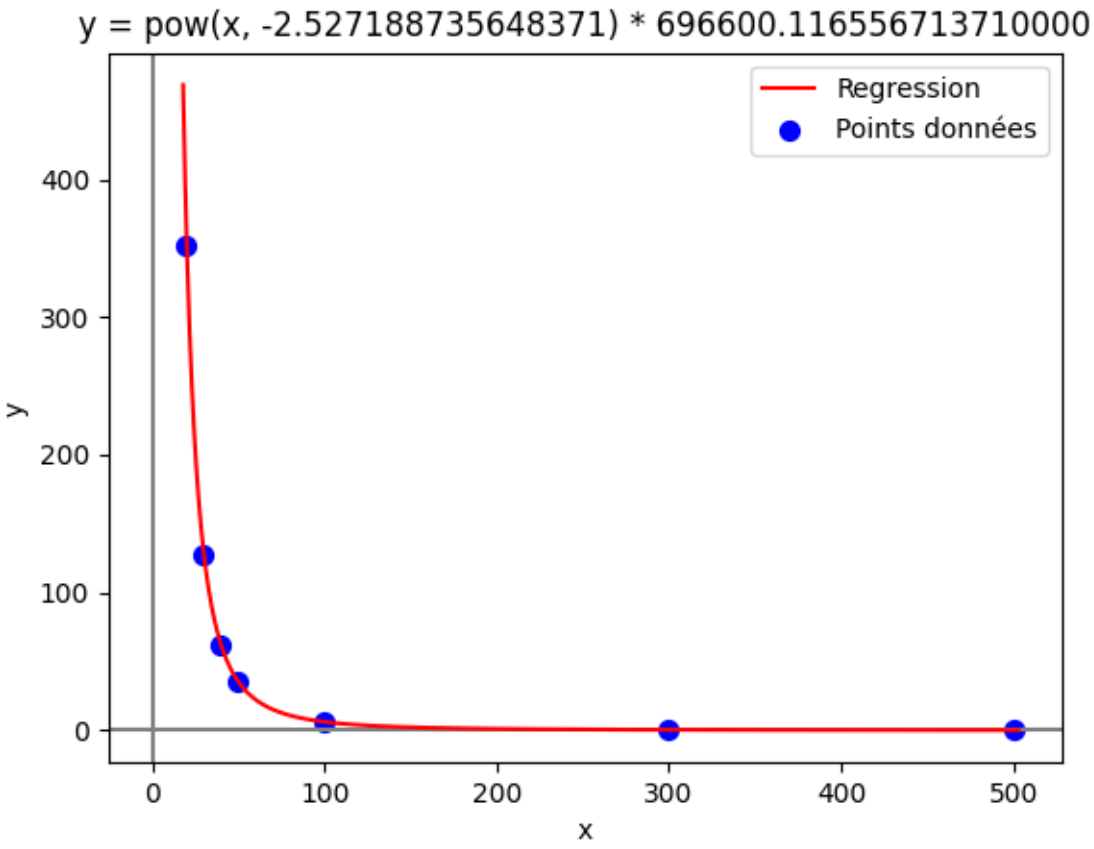

Figure 6 – $y = x^{(-2.52719)} * 69600.117$ (s = 2.86079 ; jeu d'essai 4.4)

Conclusion

On utilise la régression linéaire pour identifier une tendance dans des données, et prédire une valeur de y pour un certain x. Visuellement, c'est typiquement représenté par une courbe passant par un nuage de points. L'équation de cette courbe se veut une approximation du lien entre une variable explicative (par exemple, le temps ; x) et une variable expliquée (par exemple, le nombre d'individus d'une espèce spécifique dans un écosystème : y).
Si l'équation est destinée à être utilisée pour prédire des résultats, on peut considérer l'écart type (entre les y approximés et les y connus pour un même x) pour déterminer si la courbe est bien adaptée à nos données ; plus généralement, il convient aussi de calculer un coefficient de corrélation entre les x et les y pour confirmer que la démarche de régression est même justifiée.
Enfin, on doit considérer, dans le cadre de la régression linéaire, qu'il est moins risqué de tenter de prédire un y dans l'intervalle des y connus, que si ce y est en dehors de cet intervalle (Cf. figs 4, 5).

Table des figures