Modèle IA

Créé par :

Pauline Sanchez

Ewen Laxalde

26/01/2022

Sommaire :

1. Un Rappel sur la régression linéaire simple, multiple et polynomiale

2. Explication de chaque fonction en présentant l’équation matricielle qui lui correspond (par la méthode normale)

3. Présentation des résultats des modèles sous forme de graphiques et évaluation des modèles

4. Présentation des résultats avec le module Scikit-Learn et comparaison avec la méthode normale

5. Conclusion :
    - Qu’avez-vous appris ?
    - Comment ?
    - Des difficultés ?
    - Comment vous vous sentez après ce projet ?

1. Un Rappel sur la régression linéaire simple, multiple et polynomiale

- Régression linéaire simple

Un modèle de régression linéaire simple est un modèle de régression qui cherche à établir une relation linéaire entre deux variables. La régression linéaire simple se base sur l'utilisation d'une seule variable (X) pour définir une target (y).

- Régression multiple

Un modèle de régression linéaire multiple est un modèle de régression qui cherche à établir une relation linéaire entre une variable, dite expliquée, et plusieurs variables, dite explicatives. La régression linéaire multiple se base sur l'utilisation de plusieurs variables (X1, X2, X3…) pour définir une target (y).

- Régression polynomiale

Un modèle de régression linéaire polynomiale est un modèle de régression qui cherche à établir une relation entre deux variables avec un ordre de régression supérieur à 1. La régression polynomiale est comparable aux deux autres régressions à l'exception que celle-ci ne forme pas une droite et qu'il faut utiliser des polynômes.

2. Explication de chaque fonction en présentant l'équation matricielle qui lui correspond (par la méthode normale)

- Régression linéaire simple

$$y = ax + b$$

On met cette équation sous sa forme matricielle.

On considère x et y comme des matrices que l'on notera X et Y.

Pour faire une équation matricielle, il faut que toutes les valeurs soient des matrices ou multipliées par des matrices comme ici a avec X.

Pour b, on multiplie cette valeur par une matrice unitaire U (cette matrice ne contient que des 1 pour ne par changer la valeur de b).

Ce qui donne l'équation :

$$Y = aX + bU$$

On peut changer la forme de cette équation

$$X = X + U = [X \quad U] \; et \; \theta = \begin{bmatrix} a \\ b \end{bmatrix}$$

Donc :

$$Y = X.\theta$$

- Régression multiple

$$y = b + \sum_{i=0}^{N} \sum_{j=0}^{M} a_i x_{i,j}$$

Forme matricielle :

On met la matrice unitaire U avec la valeur de b, ce qui donne :

$$Y = bU + aX$$

Avec a les inconnues pour les a_n et X les valeurs de tous les x_n.

Ce qui donne comme pour la première :

$$Y = X.\theta$$

Avec $\theta = \begin{bmatrix} a_0 \\ \vdots \\ a_n \\ b \end{bmatrix}$ $et\ X = \begin{bmatrix} x_{0,0} & \cdots & x_{i,0} & 1 \\ \vdots & \ddots & \vdots & \vdots \\ x_{0,j} & \cdots & x_{i,j} & 1 \end{bmatrix}$

- Régression polynomiale

$$y = \sum_{j=0}^{N} a_j x^j$$

On met sous forme matricielle :

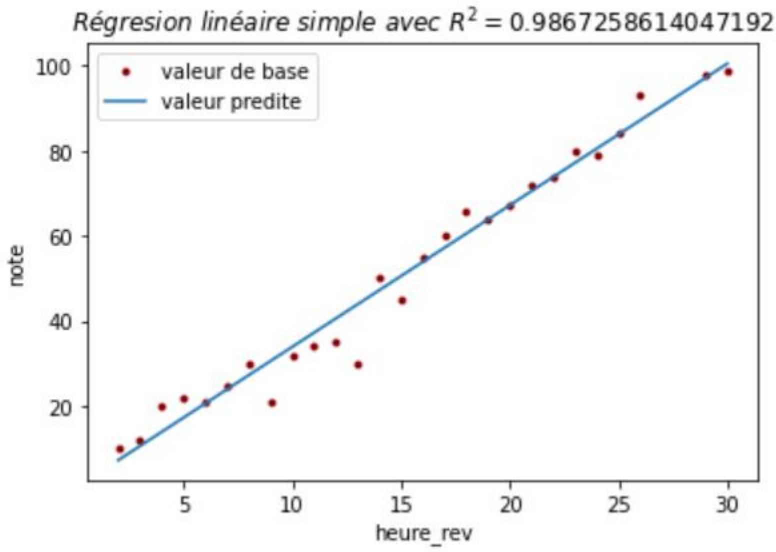
$$X = \begin{bmatrix} x^0 = 1 & x^1 & x^2 \\ \vdots & \vdots & \vdots \\ x^0 = 1 & x^1 & x^2 \end{bmatrix} \ et\ \theta = \begin{bmatrix} a_0 \\ a_1 \\ a_2 \end{bmatrix}$$

Donc on obtient :

$$Y = X.\theta$$

3. Présentation des résultats des modèles sous forme de graphiques et évaluation des modèles

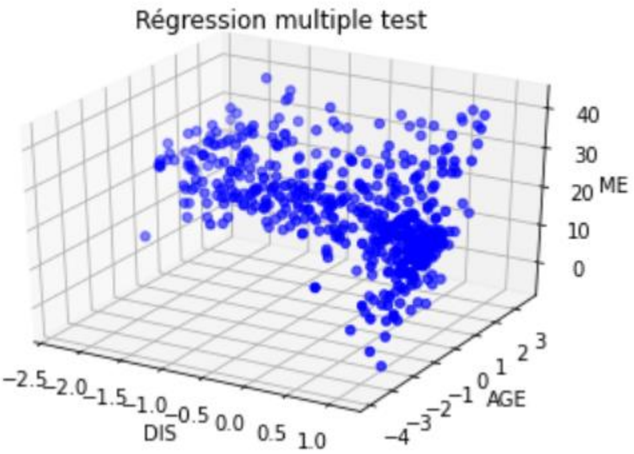
- Régression linéaire simple

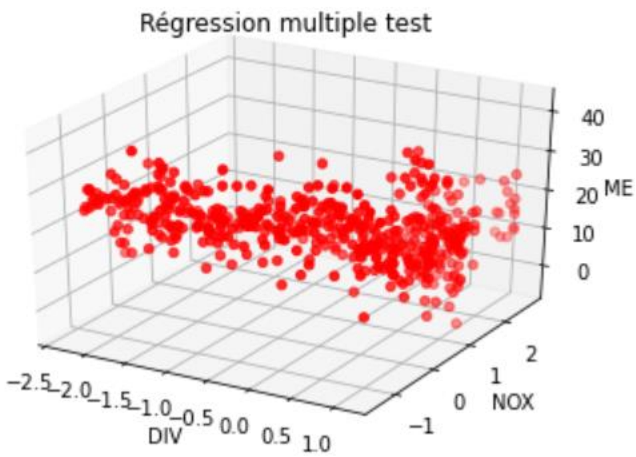


On fait la régression linéaire simple de la colonne note par rapport aux heures de révision.

Le coefficient de détermination est R^2=0.987 donc notre modèle est bon.

- ## Régression multiple

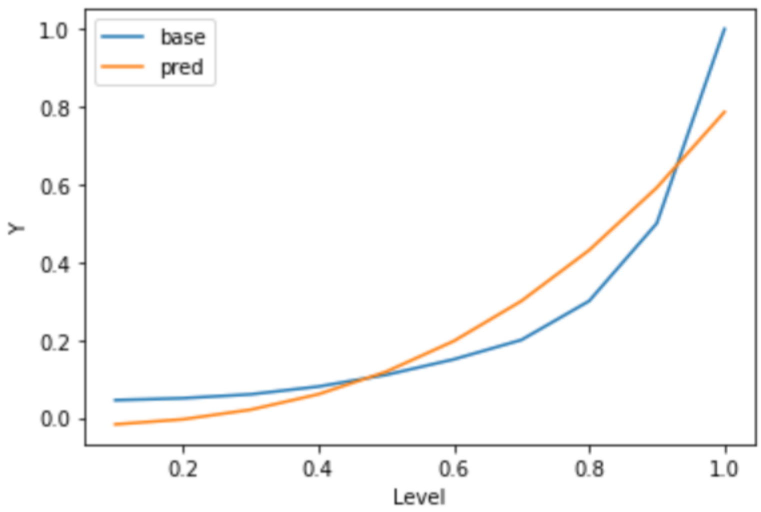




On fait cette régression multiple avec les colonnes qui ont les meilleures corrélations avec la colonne de Target MED qui est ici sur l'axe z.

Ici notre coefficient de détermination est de R^2=0.726, ce qui signifie que notre modèle manque de qualité au niveau de la prédiction.

- ## Régression polynomiale

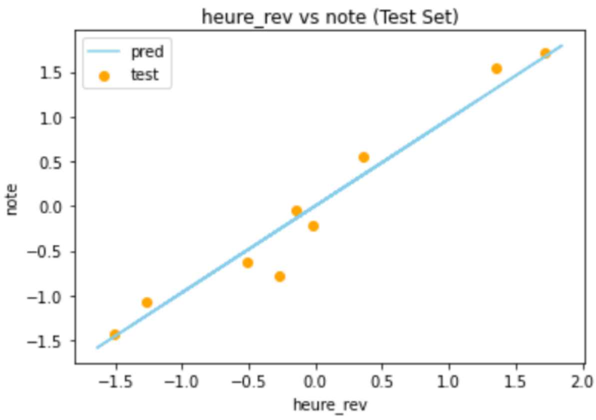


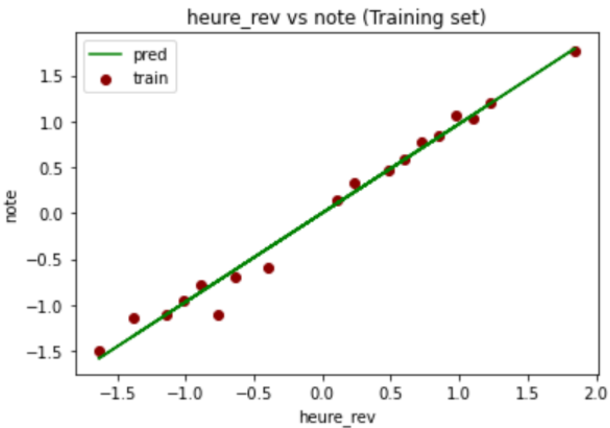
La régression polynomiale permet de trouver l'ordre de l'équation prédite qui correspond au mieux à nos données de base.

Ici nous avons fait l'erreur moyenne quadratique qui est égale à 0.0092. Notre modèle est bien prédit.

4. Présentation des résultats avec le module Scikit-Learn et comparaison avec la méthode normale

- Régression linéaire simple

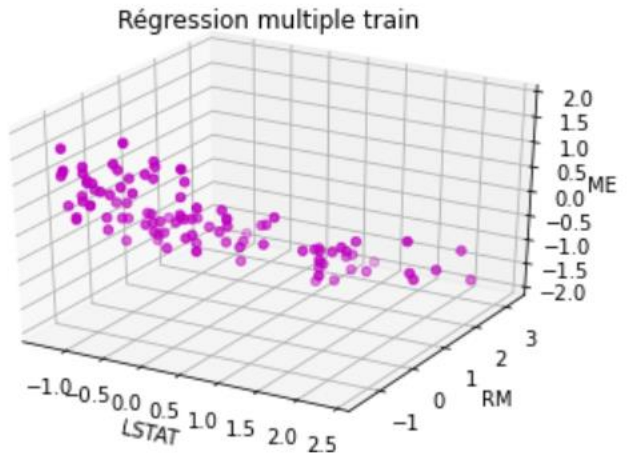


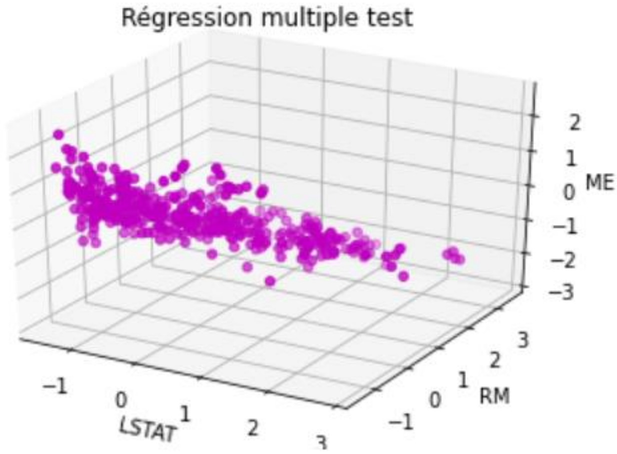


Nous avons fait des représentations des données d'entrainement et de test de notre modèle.

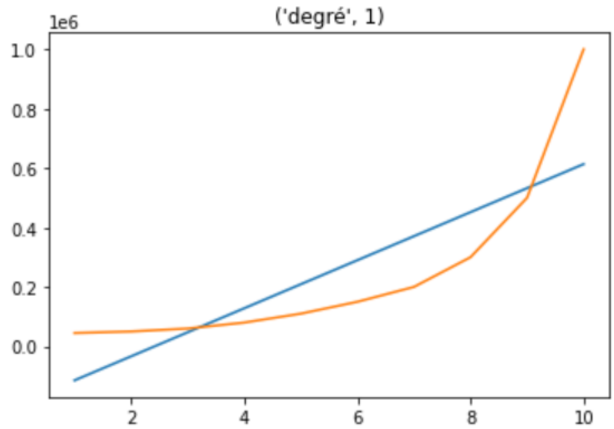
Notre modèle Scikit-Learn ne diffère pas énormément de notre modèle normal.

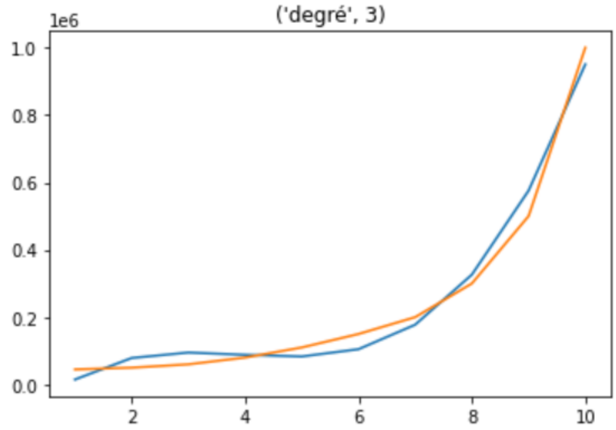
- Régression multiple

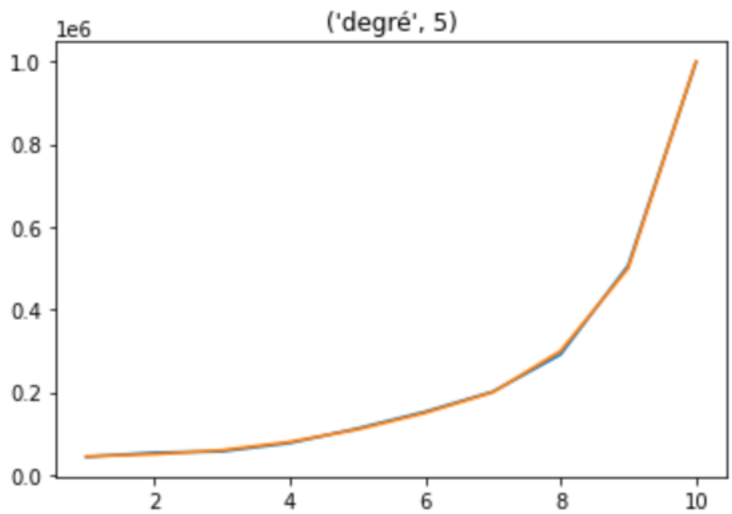




- Régression polynomiale







Notre modèle permet d’analyser l’ordre de notre polynôme pas à pas. On peut voire que plus l’ordre se rapproche de 5 plus notre modèle prédit ressemble à la courbe des données de base.

Ce modèle permet une meilleure vision sur l’ordre à utiliser que la méthode normale.

5. Conclusion

Grâce à ce brief nous avons appris à créer des modèles de régressions linéaires avec et sans Scikit-Learn. L’étape la plus compliquée ne fut pas l’apprentissage de l’utilisation de Scikit-Learn mais plutôt la mise en place des différentes régressions sans Scikit-Learn. Il fut également compliqué de savoir si ce que nous faisions était juste étant donné que nous n’avions pas la moindre idée de ce à quoi le résultat devait ressembler (notamment pour les mean square error, absolute error …). Par ailleurs, l’énoncé demandant de faire une régression polynomiale avec le dataset qualite-vin-rouge.csv nous a paru erroné étant donné que faire une régression polynomiale sur ce dataset est impossible. De manière générale, nous avons hâte de voir la suite afin de pouvoir réellement commencer à travailler sur des modèles. Et enfin, nous espérons obtenir une marche à suivre claire et précise sur les étapes à suivre afin de créer un modèle fonctionnel.