

# 

| RAPPORT SAE 2.04 |
| --- |

# 

| * IUT Lannion - 2024/2025 * Département Informatique | * [LE CHEVERE Yannis](https://yannislechevere.github.io/portfolio/) * [LE SECH Marceau](https://tx-diloxi.github.io/Portfolio/) |
| --- | --- |

SOMMAIRE (lien cliquable)

| [**I - Présentation des données**](#_b35v6rosxgw1) |
| --- |

1. [Présentation des données](#_3g7gqf2hzu9e)
2. [Problématique](#_1wa7s2z91qc)

| [**II - Import des données, mise en forme**](#_e35cy5lptakl) |
| --- |

1. [Importer les données en Python](#_3hkr68v3rjt)
2. [Mise en forme](#_p3rms2bawznm)
3. [Normalisation](#_pwcoic6fvibu)

| [**III - Représentations graphiques**](#_oler3rcu6uig) |
| --- |

| [**IV - Matrice des coefficients de corrélation**](#_ar9xv0xm4g0a) |
| --- |

1. [Démarche](#_cueqtf3lehv3)
2. [Matrice des corrélations](#_vfxlspe5bcl1)

| [**V - Régression linéaire multiple**](#_pq2zqbxui20a) |
| --- |

1. [Utilisation de la Régression linéaire multiple : comment ?](#_417pbgdfqfh)
2. [Variables explicatives les plus pertinentes](#_vcp46b2a4ngp)
3. [Lien avec la problématique](#_z712pkj3z49l)
4. [Régression Linéaire Multiple en Python](#_bjnh2q5tq5cn)
5. [Paramètres obtenus et interprétation détaillée](#_yysx5vk44ber)
6. [Coefficient de corrélation multiple, interprétation](#_zdrl6xld2f7y)

| [**VI - Conclusion**](#_xnmvyp5n1vb7) |
| --- |

1. [Réponse à la problématique](#_h0qpfslcxhpj)
2. [Argumentation depuis les résultats de la régression linéaire](#_i9oi96ggtxpw)
3. [Interprétations personnelles](#_ibjflfoj7aw7)

# 

| I - Présentation des données |
| --- |

## Présentation des données

Population étudiée : Étudiant ayant fourni leur prénom, leur moyenne générale, leur code postal, leur niveau d'études et leur mention au bac.

Description des variables :

* *prenom* : Prénom de l’étudiant (ex : Maurice)
* *moyenne* : Moyenne générale obtenue de l'étudiant (ex : 12.75)
* *code\_postal* : Code postal de résidence de l’étudiant (ex : 22300)
* *niveau\_etude* : Niveau d’étude de l’étudiant (ex : Terminale)
* *mention\_bac* : Mention au bac de l’étudiant (ex : TB)

## Problématique

En utilisant ces données, on va essayer de répondre à la problématique suivante :

*Existe-t-il une influence de la première lettre du prénom sur le niveau d'études, la mention obtenue au bac, le code postal et la moyenne générale des étudiants ?*

| II - Import des données, mise en forme |
| --- |

## Importer les données en Python

Les données sont importées en Python sous forme de DataFrame à l’aide de la commande suivante :

| cheminFichier = "./vue.csv" VueDf = pd.read\_csv(cheminFichier, sep=";") |
| --- |

## Mise en forme

On a besoin de supprimer les cases vides (qui contiennent nan en Pythons), puis on transforme notre DataFrame en Array :

| Voyelle\_df = Voyelle\_df.dropna() Voyelle\_ar = Voyelle\_df.to\_numpy() |
| --- |

## Normalisation

Pour cela on prend toutes les variables qui étaient des chaînes de caractère pour les transformer en variable quantitatif.

* *prenom* devient *initiale*
* *mention\_bac* devient *mention\_bac\_num*
* *niveau\_etude* devient *niveau\_etude\_num*

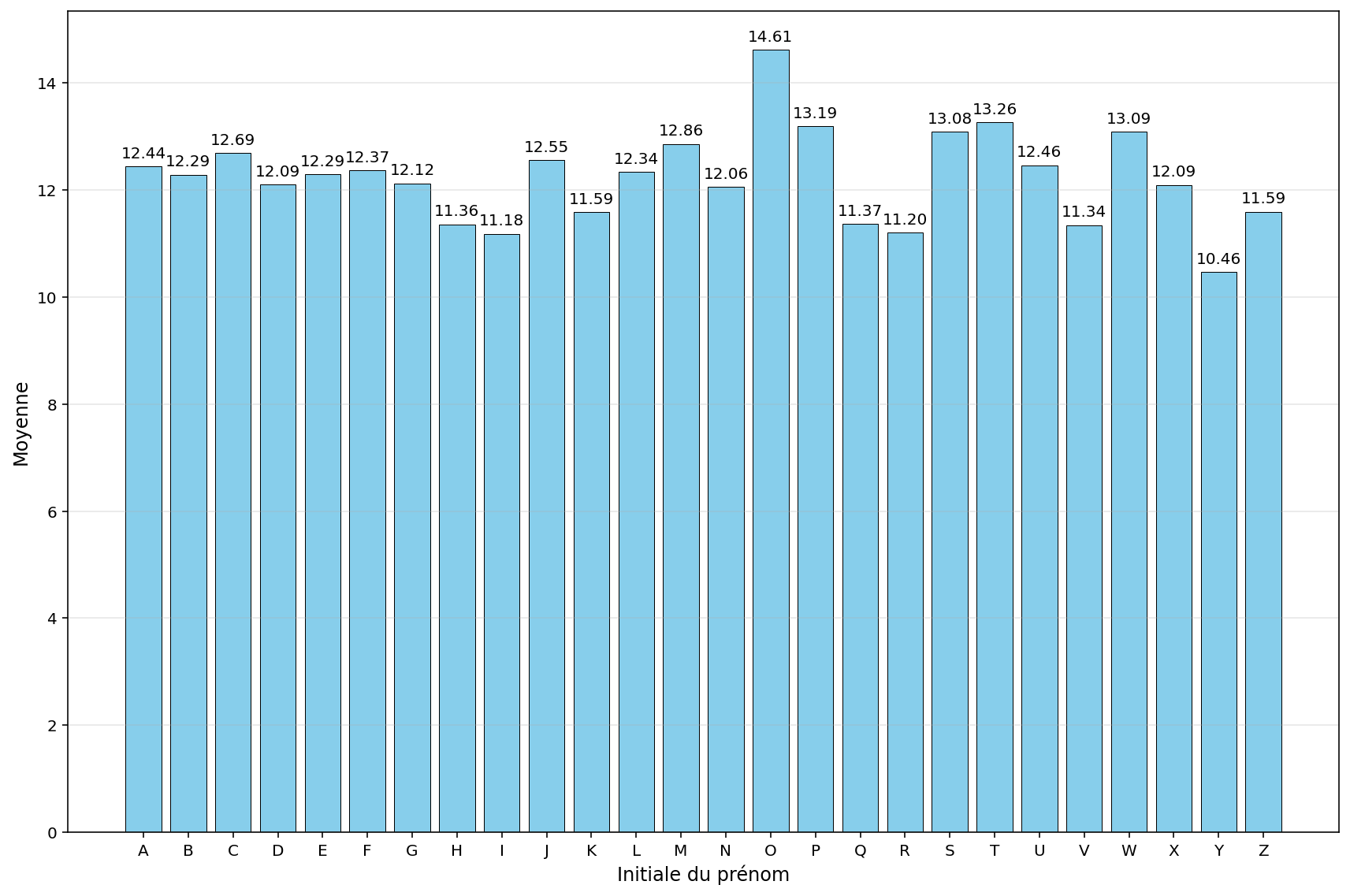
| VueDf['initiale'] = VueDf['prenom'].str[0]  alphabet = list("ABCDEFGHIJKLMNOPQRSTUVWXYZ") mapping\_initiale = {lettre: i+1 for i, lettre in enumerate(alphabet)} VueDf['initiale\_num'] = VueDf['initiale'].map(mapping\_initiale)  VueDf['mention\_bac\_num'] = VueDf['mention\_bac'].map({  'P': 1, 'AB': 2, 'B': 3, 'TB': 4 })  VueDf['niveau\_etude\_num'] = VueDf['niveau\_etude'].map({  "Terminale": 1, "Année préparatoire aux études supérieures": 2, "1ère année d'études supérieures": 3, "2nd année d'études supérieures": 4 }) |
| --- |

# 

| III - Représentations graphiques |
| --- |

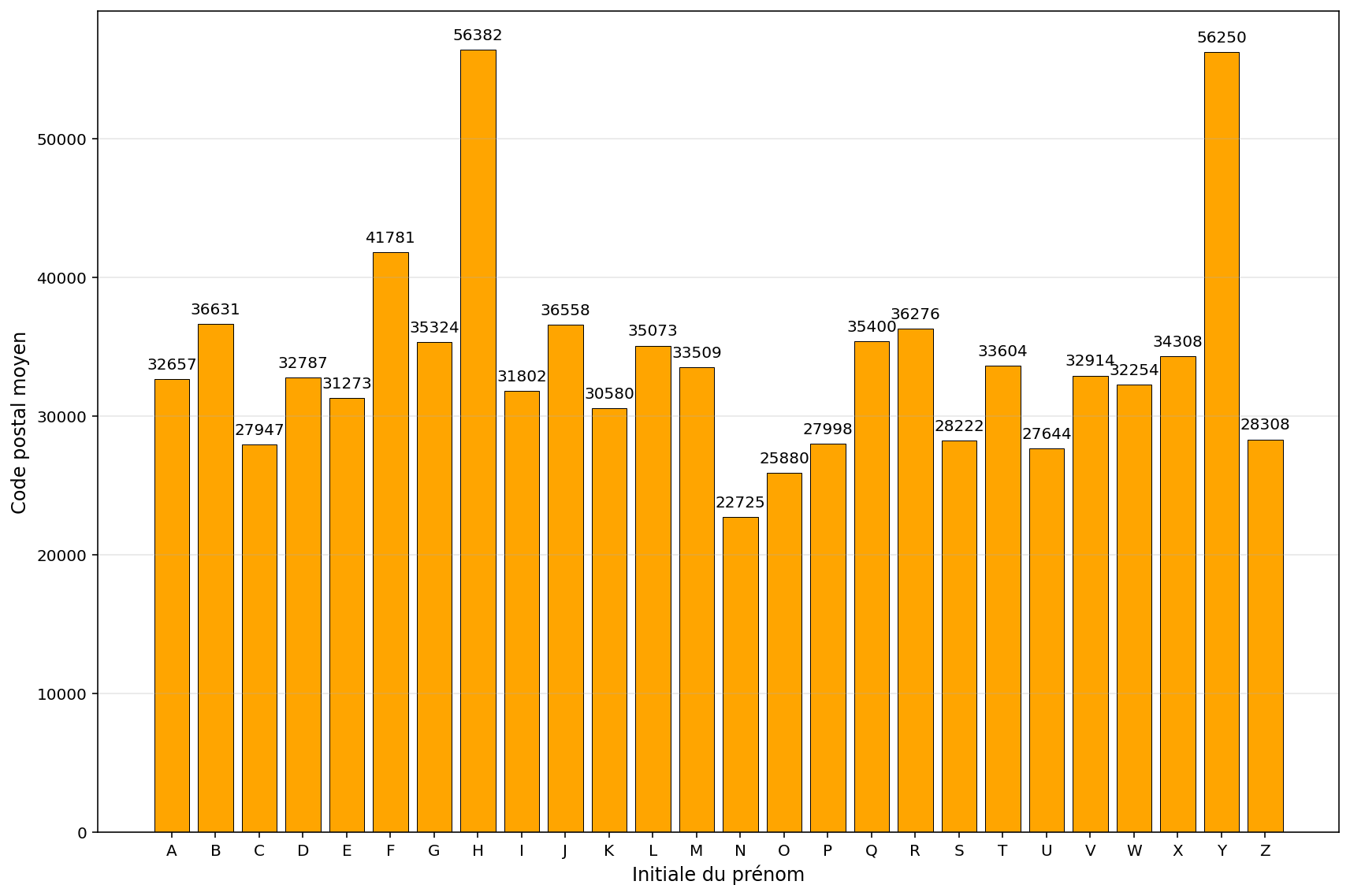
Pour les diagrammes suivants, on a choisi d'utiliser des diagrammes en bâtons car on avait seulement le choix entre diagramme en bâtons et diagramme en moustaches, et donc on a fait la moyenne de chaque variable en fonction de l'initiale du prénom. Seulement, dans la plupart des cas, ce n'est absolument pas adapté de faire un diagramme en bâtons. Il aurait été plus efficace de faire un diagramme de points (avec scatterplot) ou autre puisque’on aurait put réélemnt voir, mais ce n'est pas autorisé dans les consignes.

* Diagramme en bâtons : Moyenne générale en fonction de l'initiale du prénom



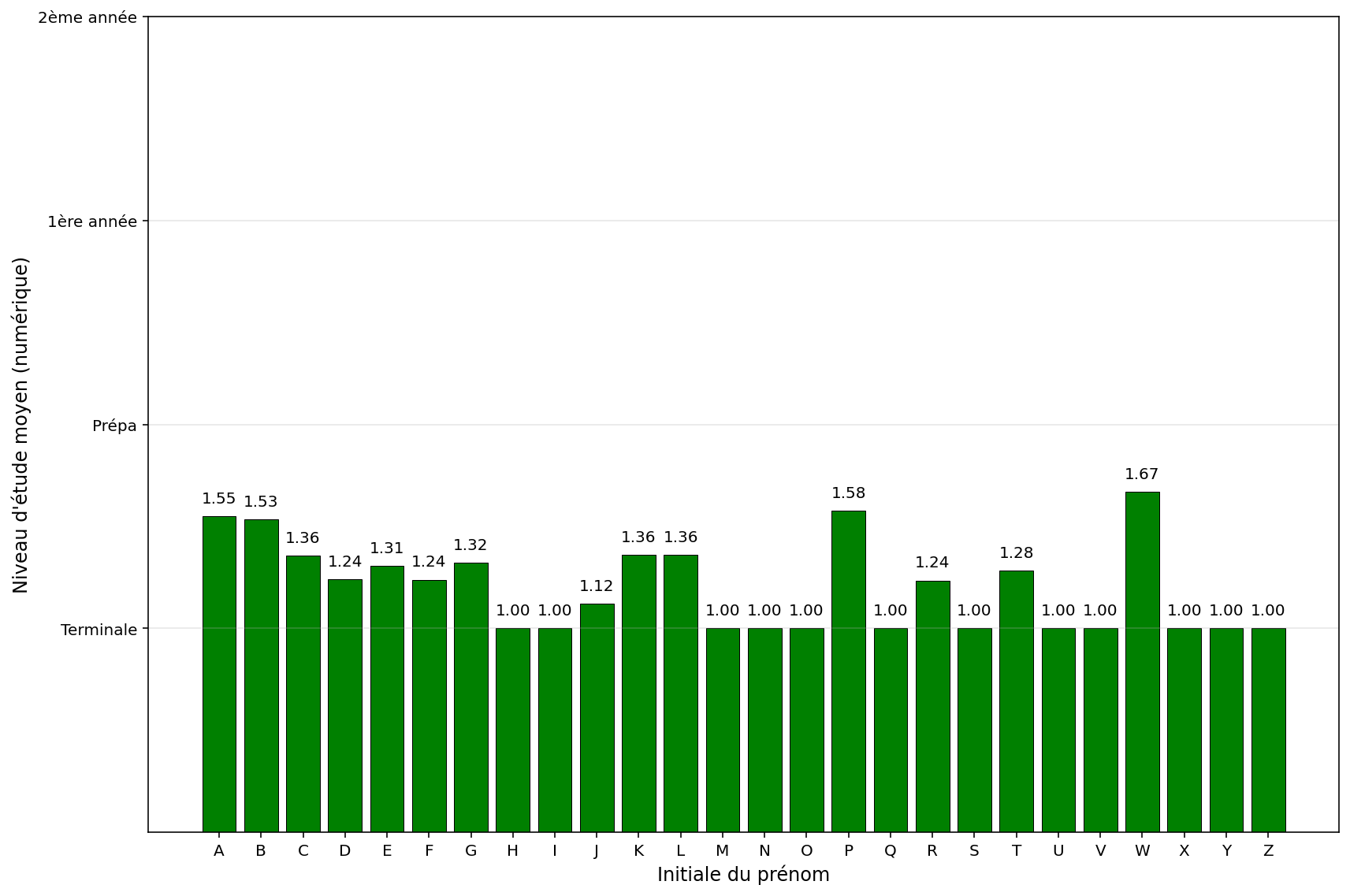
On remarque que les notes moyennes sont très homogènes. Seuls les prénoms avec O comme initiale sont bien au-dessus du reste.

* Diagramme en bâtons : Code postal moyen en fonction de l'initiale du prénom



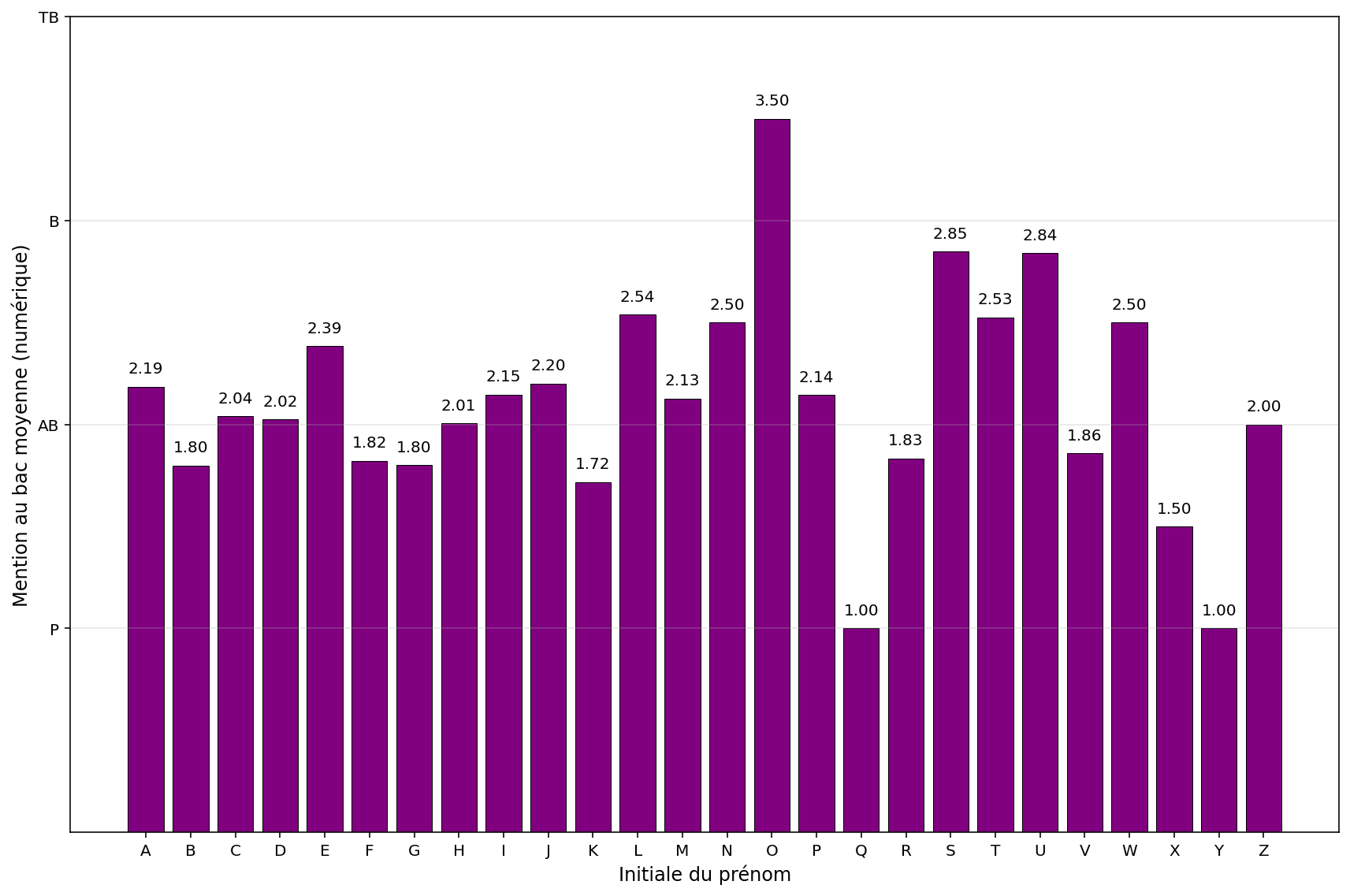
On remarque qu'il y a deux valeurs bien au-dessus des autres. La représentation en diagramme en bâtons n'est pas adaptée car on est obligé de faire la moyenne de codes postaux, ce qui est peu utile pour se donner une représentation des codes postaux en fonction des initiales du prénom.

* Diagramme en bâtons : Niveau d'étude moyen en fonction de l'initiale du prénom

****

Les valeurs cette fois-ci sont très homogènes. Certains niveaux d'étude ne sont pas atteints alors que certaines personnes ont ce niveau d'étude. C'est tout simplement parce qu'on fait une moyenne du niveau d'étude, ce qui nous permet de représenter les données en diagramme en bâtons.

* Diagramme en bâtons : Mention au bac en fonction de l'initiale du prénom

****

On remarque qu'il y a une valeur bien au-dessus (lettre O) et deux valeurs en dessous du reste (lettres Q et Y).

| IV - Matrice des coefficients de corrélation |
| --- |

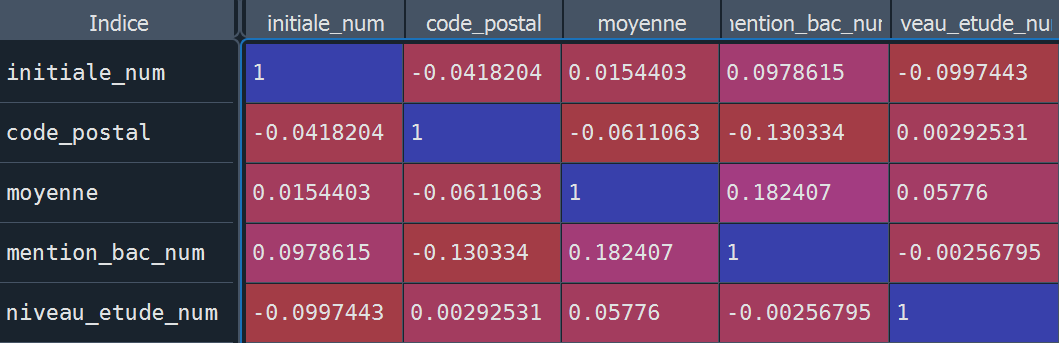
## Démarche

Dans cette partie, on calcule la matrice des corrélations afin de mesurer les relations linéaires entre nos différentes variables numériques.

| df\_corr = VueDf[['initiale\_num', 'code\_postal', 'moyenne', 'mention\_bac\_num', 'niveau\_etude\_num']]  corr\_matrix = df\_corr.corr() |
| --- |

## Matrice des corrélations

On obtient la matrice suivante :

****

Les corrélations qui nous intéressent sont celles de *initiale\_num* avec le reste de variables. On remarque donc que pour :

* *code\_postal* : Corrélation très faible et négative (pratiquement aucun lien)
* *moyenne* : Corrélation très proche de 0 donc très faible positive (encore moin de lien)
* *mention\_bac\_num* : Corrélation faible et positive (un petit lien)
* *niveau\_etude\_num* : Corrélation faible et négative (un petit lien)

| V - Régression linéaire multiple |
| --- |

## Utilisation de la Régression linéaire multiple : comment ?

Dans cette analyse, nous choisissons comme variable endogène la variable *initiale\_num*, c’est-à-dire la première lettre du prénom des étudiants, convertie en nombre selon l’ordre alphabétique (A = 1, ..., Z = 26).

Ce choix est atypique car l’initiale d’un prénom n’est généralement pas influencée par des caractéristiques telles que la moyenne ou le niveau d’étude.

Les variables explicatives choisies sont :

* *moyenne* (la moyenne générale de l’étudiant),
* *mention\_bac\_num* (mention obtenue au bac, de 1 à 4),
* *niveau\_etude\_num* (niveau d’étude, de 1 à 4),
* *code\_postal* (code postal de provenance).

## Variables explicatives les plus pertinentes

Les variables explicatives sont toutes numériques car précédemment normalisées. Elles couvrent à la fois des aspects académiques (*moyenne*, *mention\_bac\_num*, *niveau\_etude\_num*) et géographiques (*code\_postal*). Ces données sont disponibles pour tous les étudiant de notre vue, ce qui rend leur utilisation pratique pour modéliser une relation, même hypothétique.

## Lien avec la problématique

La problématique étant :

*Existe-t-il une influence de la première lettre du prénom sur le niveau d'études, la mention obtenue au bac, le code postal et la moyenne générale des étudiants ?*

Cette problématique est donc discutable car il n’y a pas de corrélation entre les variables comme vue avant. Cette démarche est volontairement absurde.

## Régression Linéaire Multiple en Python

On fait maintenant la régression linéaire multiple avec Python :

| # Fonctions fournies en tp  def coefficients\_regression\_lineaire(X, y):  """  Calcule les coefficients de l'hyperplan pour une régression linéaire multiple.  X : ndarray de shape (n, m)  y : ndarray de shape (n, 1) ou (n,)  Retourne : theta (ndarray de shape (m+1,) avec b à l'indice 0)  """  n\_samples = X.shape[0]  X\_aug = np.hstack((np.ones((n\_samples, 1)), X))  theta = np.linalg.inv(X\_aug.T @ X\_aug) @ X\_aug.T @ y  return theta.flatten()   def predire\_y(X, theta):  """  Calcule y\_pred à partir de X et theta.  X : ndarray de shape (n, m)  theta : ndarray de shape (m+1,) -- inclut l'intercept  Retourne : y\_pred (ndarray de shape (n,))  """  n\_samples = X.shape[0]  X\_aug = np.hstack((np.ones((n\_samples, 1)), X))  y\_pred = X\_aug @ theta  return y\_pred   def coefficient\_correlation\_multiple(y\_true, y\_pred):  """  Calcule le coefficient de corrélation multiple (R^2)  y\_true : valeurs réelles (shape: (n,))  y\_pred : valeurs prédites (shape: (n,))  Retourne : R² (float)  """  y\_true = np.ravel(y\_true)  y\_pred = np.ravel(y\_pred)  ss\_res = np.sum((y\_true - y\_pred)\*\*2)  ss\_tot = np.sum((y\_true - np.mean(y\_true))\*\*2)  r\_squared = 1 - ss\_res / ss\_tot  return r\_squared  # --- Régression avec pour cible l'ordre alphabétique des prénoms --- X = VueDf[['moyenne', 'mention\_bac\_num', 'niveau\_etude\_num', 'code\_postal']].to\_numpy() y = VueDf['initiale\_num'].to\_numpy()  theta = coefficients\_regression\_lineaire(X, y) y\_pred = predire\_y(X, theta) r2 = coefficient\_correlation\_multiple(y, y\_pred)  # --- Résultats --- print("Coefficients du modèle (theta) :", theta) print("Coefficient de corrélation multiple R² :", r2) |
| --- |

## Paramètres obtenus et interprétation détaillée

**L'exécution du code de régression nous donne les coefficients suivants :**

* a₀ (ordonnée à l'origine) : 10.661
* a₁ (coefficient de la moyenne) : 0.0048
* a₂ (coefficient de mention\_bac\_num) : 0.739
* a₃ (coefficient de niveau\_etude\_num) : -0.936
* a₄ (coefficient de code\_postal) : -0.0000144

**Modèle mathématique obtenu :**

Le modèle s'écrit sous la forme : *initiale\_num* = a₀ + a₁ × *moyenne* + a₂ × *mention\_bac\_num* + a₃ × *niveau\_etude\_num* + a₄ × *code\_postal*

**Interprétation des coefficients :**

* a₀ (ordonnée à l'origine) : La valeur 10.661 représente la valeur prédite de l'initiale numérique lorsque toutes les variables explicatives sont nulles. Cette valeur correspond approximativement à la lettre "K" (11ème lettre), ce qui n'a aucun sens pratique dans notre contexte.
* a₁ (coefficient de *moyenne*) : Le coefficient 0.0048 est extrêmement faible, indiquant qu'une augmentation de 1 point de moyenne "prédirait" une augmentation de seulement 0.0048 dans l'ordre alphabétique de l'initiale. Cette valeur proche de zéro confirme l'absence de relation entre moyenne et initiale du prénom.
* a₂ (coefficient de *mention\_bac\_num*) : Le coefficient 0.739 est le plus élevé en valeur absolue. Il suggère que passer d'une mention à la suivante (ex: AB → B) "prédit" une augmentation de 0.739 dans l'initiale numérique. Cette relation est causalement impossible.
* a₃ (coefficient de *niveau\_etude\_num*) : Le coefficient -0.936 suggérerait qu'avancer d'un niveau d'étude "prédit" une diminution de l'initiale numérique. Cette relation négative n'a aucun fondement logique.
* a₄ (coefficient de *code\_postal*) : Le coefficient -0.0000144 est négligeable, confirmant l'absence totale de relation entre code postal et initiale du prénom.

**Analyse critique des résultats :**

Tous ces coefficients, bien qu'ils puissent être calculés mathématiquement, n'ont aucun sens causal réel. Ils représentent des corrélations purement fortuites dans notre échantillon, confirmant ainsi que :

* La régression peut toujours produire des coefficients, même en l'absence de relation logique
* L'existence de coefficients non nuls ≠ existence d'une relation causale
* Les très faibles corrélations observées dans la matrice (partie IV) se retrouvent dans les coefficients obtenus (a₁ et a₄ quasi-nuls)

## Coefficient de corrélation multiple, interprétation

Avec la fonction de sklearn :

| from sklearn.linear\_model import LinearRegression  modele\_sk = LinearRegression() modele\_sk.fit(X\_Ar, y\_Ar) r2\_sklearn = modele\_sk.score(X\_Ar, y\_Ar)  print("R² avec sklearn :", r2\_sklearn) |
| --- |

Le coefficient de corrélation multiple avec la formule vue dans le Cours 1 des Statistiques pour la SAE

| import numpy as np  def coefficients\_regression\_lineaire(X, y):  """  Calcule les coefficients de l'hyperplan pour une régression linéaire multiple.  X : ndarray de shape (n, m)  y : ndarray de shape (n, 1) ou (n,)  Retourne : theta (ndarray de shape (m+1,) avec b à l'indice 0)  """  n\_samples = X.shape[0]  X\_aug = np.hstack((np.ones((n\_samples, 1)), X))  theta = np.linalg.inv(X\_aug.T @ X\_aug) @ X\_aug.T @ y  return theta.flatten()  def coefficient\_correlation\_multiple(y\_true, y\_pred):  """  Calcule le coefficient de corrélation multiple (R^2)  y\_true : valeurs réelles (shape: (n,))  y\_pred : valeurs prédites (shape: (n,))  Retourne : R² (float)  """  y\_true = np.ravel(y\_true)  y\_pred = np.ravel(y\_pred)  ss\_res = np.sum((y\_true - y\_pred)\*\*2)  ss\_tot = np.sum((y\_true - np.mean(y\_true))\*\*2)  r\_squared = 1 - ss\_res / ss\_tot  return r\_squared  def predire\_y(X, theta):   """   Calcule y\_pred à partir de X et theta.   """   n\_samples = X.shape[0]   X\_aug = np.hstack((np.ones((n\_samples, 1)), X))  y\_pred = X\_aug @ theta   return y\_pred  X\_Ar = VueDf[['moyenne', 'mention\_bac\_num', 'niveau\_etude\_num','code\_postal']].to\_numpy()  y\_Ar = VueDf['initiale\_num'].to\_numpy()   # Calcul des coefficients  theta = coefficients\_regression\_lineaire(X\_Ar, y\_Ar)   # Prédiction des valeurs de y (version optimisée)  y\_pred = predire\_y(X\_Ar, theta)  # Calcul du R²  r2 = coefficient\_correlation\_multiple(y\_Ar, y\_pred)   print("R² (calcul manuel sans sklearn) :", r2)  print("Coefficients du modèle :", theta) |
| --- |

| VI - Conclusion |
| --- |

## Réponse à la problématique

La problématique que nous avons choisie est volontairement absurde :

*Existe-t-il une influence de la première lettre du prénom sur le niveau d'études, la mention obtenue au bac, le code postal et la moyenne générale des étudiants ?*

La réponse est clairement non. Nos analyses statistiques montrent qu’il n’existe aucune relation significative entre ces variables.

## Argumentation depuis les résultats de la régression linéaire

Nos résultats quantitatifs soutiennent cette conclusion :

Les coefficients de la régression linéaire multiple sont très faibles :

* *moyenne* → effet ≈ 0.0048
* *niveau\_etude\_num* → effet ≈ -0.936
* *code\_postal* → effet ≈ -0.0000144
* Seule mention au bac présente un léger effet : ≈ 0.739, mais sans signification causale.

Le coefficient de corrélation multiple R2R^2 est de 0.0203, soit 2.03% de la variance expliquée.

Cela signifie que 97.97% de la variation de l’initiale du prénom ne peut pas être expliquée par nos variables. C’est un très mauvais modèle prédictif, ce qui valide le caractère volontairement absurde de notre question.

## Interprétations personnelles

Sérieuse

Ce travail illustre que l’on peut **calculer une régression même lorsqu’il n’existe aucun lien logique** entre les variables. La pertinence d’un modèle ne dépend pas seulement de sa validité mathématique, mais aussi de la **pertinence de la question posée** et du **bon sens statistique**.

Absurde

On pourrait conclure que :

* Les étudiants dont le prénom commence par **O** sont de futurs génies, car ils ont une moyenne supérieure à la normale.
* Le code postal 22300 dévalorise le prénom vers des lettres plus basses de l’alphabet.
* Avoir une mention **Très Bien** augmente les chances que ton prénom commence par un **M** ou un **N**.