Statistique pour la SAé 2.04 - TP3
Analyse de données de la SAé

Dans ce TP, on va reprendre ce qu'on a appris en Statistique et sur Régression Linéaire Multiple dans les TP sur les sangliers, et faire un travail d'analyse de données à partir d'une vue issue de la base de données des Collèges.

La démarche suivie dans ce TP sera un modèle pour analyser les données que vous choisirez pour votre projet de SAé.

1 Présentation des données Colleges.csv

Le fichier Colleges.csv contient plusieurs séries statistiques sur l'ensemble de toutes les collèges répertoriés dans votre base de données :

1. Dans ces données, quelle est la population ?
2. Quelle(s) colonne(s) du fichier .csv ser(ven)t à identifier les individus de cette population ?

Les variables statistiques considérées sont les suivantes :

- La 1e variable statistique sur cette population est la note moyenne à l'écrit du brevet en 2023 pour chaque collège.
- La 2e est la valeur ajoutée estimée sur la note au brevet moyenne pour ce collège.
- La 3e est le pourcentage de collégiens hors segpa et ulis dans chaque collège
- La 4e est le pourcentage de collégiens en LV1 allemand an 3e.
- La 5e est le pourcentage de collégiennes en 3e.
- La 6e est l'indice de position sociale (IPS) moyen du collège.
- La 7e est l'écart-type de l'IPS dans le collège.
- La 8e donne le classement REP/REP+ ou hors Education Prioritaire (EP) du collège.
- La dernière est l'académie dans laquelle se trouve le collège.

On a choisi ces données pour se faire une idée de ce qui favorise de bonnes notes au brevet dans les différents collèges.

2 Import des données, mise en forme

On commence par importer dans Python et centrer-réduire les données du fichier Colleges1.csv. On doit pour cela résoudre quelques problèmes de mise en forme.

3. Importer `Colleges.csv` sous forme d'un DataFrame `CollegesDF`.
4. Transformer ce DataFrame en np.array `CollegesAr`.
5. Séparer ce tableau `CollegesAr` en 2 tableaux np.array : un tableau `CollegesAr0` qui contient les colonnes avec des valeurs numériques, et un tableau `CollegesAr1` contenant les autres colonnes.
6. On va maintenant centrer-réduire les données numériques. Exécuter la fonction `Centreduire` sur `CollegesAr0`. Un problème apparait, lequel ? On ne demande pas d'expliquer pourquoi.
7. Dans votre tableau `CollegesAr0`, il y avait des cases contenant `nan` (signifie *Not A Number*). A quoi correspondent-elles dans le fichier csv de départ ?
8. Pour résoudre le problème apparu dans la question précédente, appliquer la commande suivante **juste après** l'import du fichiers csv :

   ```
   CollegesDF=pd.read_csv("Colleges.csv")
   CollegesDF = CollegesDF.dropna()
   ```

   Cette commande supprime les lignes ayant des cases vides, sans changer le nom des lignes conservées.

3 Exploration des données

Quand on a un grand jeu de données comme c'est le cas ici, il est difficile de se faire une idée de ce qu'on peut en faire. La première chose à faire est d'essayer de s'en faire une meilleure idée, c'est ce qu'on appelle **l'exploration de données** (en anglais *data mining* : "forage" de données).

On va le faire sur nos données avec 2 approches :

a. Exploration à l'aide de réprésentations graphiques

Une première approche est de faire des représentations graphiques. On en fait quelques unes dans les questions ci-dessous.

On va commencer par faire des diagrammes en batons représentant chacune de nos colonnes de données :

9. Pour rappel, le tracé de diagramme en bâtons en Python a été vu dans le TP1 de Statistique dans l'exercice 4.

   Créer une fonction DiagBatons, sur le modèle suivant (à compléter) :

```
def DiagBatons(Colonne)
    m=     # m contient la valeur minimale de la colonne
    M=     # M contient la valeur maximale de la colonne
    inter=[]   # liste de 21 valeurs allant de m à M. On
               # peut utiliser la fonction np.linspace
    plt.figure()
    # tracé du diagramme pour les intervalles inter
```

10. Représenter les diagrammes en bâtons de chacune des variables statistiques de `CollegesAr0` (de préférence avec les données non centrées-réduites, elles seront plus parlantes).

...Questions 11 à 15 : pour les + rapides...

On va maintenant représenter côte à côte plusieurs boites à moustaches donnant les IPS, chaque boite représentant une académie.

11. Rentrer dans Python la liste `L_Acad` des académies sous forme de chaines de caractères (on peut copier-coller à partir du fichier csv par exemple).
12. A l'aide des tableaux `CollegesAr0` et `CollegesAr1`, créer une liste `IPS_Est` contenant les IPS des collèges de l'académie Grand-Est.

    On rappelle que les indices des lignes de `CollegesAr0` *et* `CollegesAr1` *sont les mêmes.*
13. Créer une liste de liste `IPS_Acad`, contenant les 16 listes des IPS de tous les collèges de chacune des 16 académies.

14. Pour représenter les boites à moustaches des IPS des 16 académies, il vous suffit de rentrer la commande :

```
plt.boxplot(IPS_Acad,labels=L_Acad)
```

15. Faire la même chose pour les notes moyennes au brevet.

b. Exploration à l'aide de la matrice de covariance

Dans cette partie, on apprend ce qu'est la matrice de covariance, qui permet d'obtenir en un seul calcul les corrélations 2 à 2 de toutes nos variables.

Matrice de Covariance

Si on a p variables statistiques $X_1, X_2, \ldots, X_p$ (par exemple les $p = 7$ colonnes du tableau `CollegesAr0`), la **matrice de covariance** de ces p variables sera la matrice :

$$\begin{pmatrix} \text{var}(X_1) & \text{cov}(X_1,X_2) & \ldots & \text{cov}(X_1,X_j) & \ldots & \text{cov}(X_1,X_p) \\ \text{cov}(X_2,X_1) & \text{var}(X_2) & \ldots & \text{cov}(X_2,X_j) & \ldots & \text{cov}(X_2,X_p) \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ \text{cov}(X_j,X_1) & \text{cov}(X_j,X_2) & \ldots & \text{var}(X_j) & \ldots & \text{cov}(X_j,X_p) \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ \text{cov}(X_p,X_1) & \text{cov}(X_p,X_2) & \ldots & \text{cov}(X_p,X_j) & \ldots & \text{var}(X_p) \end{pmatrix}$$

Si les variables $X_1, X_2, \ldots, X_p$ sont centrées-réduites, alors pour toutes les séries X_i et X_j, on a de plus :

$$\text{Var}(X_i) = 1, \quad \text{et } \text{Cov}(X_i, X_j) = \text{Cor}(X_i, X_j).$$

La commande suivante permet de calculer la matrice de covariance d'un tableau `Array` contenant chaque série statistique dans une colonne :

```
np.cov(Array,rowvar=False)
```

16. Calculer la matrice de covariance associée à l'ensemble de nos 7 variables.
17. Quelles sont les 4 variables statistiques les plus corrélées avec la note moyenne au brevet ?

4 Analyse des données avec une régression linéaire multiple

On rappelle que la question qui avait motivé l'étude de ces données était de *se faire une idée de ce qui favorise de bonnes notes au brevet dans les différents collèges* (voir p.1 de ce TP).

A la question 17, on a repéré, parmi nos variables, celles qui avaient les + fortes corrélations avec les notes au brevet. On va maintenant aller plus loin en réalisant une régression linéaire multiple avec :

- les notes au brevet comme variable endogène
- les 4 variables de la question 17 comme variables explicatives.

18. Créer un tableau `Y` à une colonne, contenant la variable endogène, et un tableau `X` à 4 colonnes contenant les variables explicatives.
19. Calculer les paramètres de cette régression linéaire multiple en utilisant la fonction LinearRegression de sklearn.
20. Interpréter les paramètres obtenus.
21. Calculer le coefficient de corrélation multiple avec la fonction LinearRegression.
22. La corrélation est-elle forte ? *Attention, on rappelle que la commande* `linear_regression.score` *donne le carré du coefficient de corrélation.*

...pour les + rapides...

Influence du nombre de variables explicatives

23. On souhaite comparer les résultats de la régression linéaire multiple ci-dessus avec ceux de la régression linéaire suivante :
    - les notes au brevet comme variable endogène
    - l'IPS comme (seule) variable explicative.

    (a) Calculer, comme ci-dessus, les paramètres et le coefficient de corrélation de cette régression linéaire.

(b) On aurait pu deviner à l'avance ces résultats en utilisant la matrice de covariance : comment ?

24. Calculer les paramètres et coefficients de corrélation des régressions linéaires multiples, avec toujours les notes au brevet comme variable endogène, et avec comme variables explicatives :
    (a) l'IPS et l'écart-type de l'IPS,
    (b) l'IPS, l'écart-type de l'IPS et la Valeur Ajoutée du collège.

25. Que fait le coefficient de corrélation à chaque ajout d'une nouvelle variable explicative ?

Influence du classement REP/REP+

26. Créer une liste ListeREP contenant les numéros de ligne des collèges classés REP, et une liste ListeREPP contenant les numéros de ligne des collèges classés REP+.

27. On rappelle que pour un numpy.array `Tab` et une liste `Liste`, la commande `Tab[Liste,:]` renvoie le tableau Tab mais seulement avec les lignes dont le numéro est dans `Liste`.

    Créer les tableaux `CollegesOAr_CR_REP` et `CollegesOAr_CR_REPP` contenant les données numériques respectivement pour les collèges REP et REP+.

28. Comparer la valeur ajoutée des collèges REP et REP+, en faisant la moyenne sur l'ensemble des collèges, puis avec des diagrammes en batons.

29. Calculer les matrices de covariance pour ces 2 types de collèges. Les variables les + corrélées avec la note au brevet ne sont plus les mêmes qu'avec la liste complète des collèges... Expliquez.