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

Dans ce TP, on va reprendre tout ce qu'on a appris en Statistique, et faire un travail d'analyse de données à partir d'une vue issue de la base de données sur les étudiant.es du DUT Info de Lannion.

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

0 Présentation des données Vue.csv

Télécharger le fichier `Vue.csv` sur Moodle.

1. Dans le fichier `Vue.csv`, quelle est la population ? 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 catégorie socioprofessionnelle d'un parent de l'étudiant.e.
- La 2e est la série du bac obtenu par l'étudiant.e : S pour le bac général scientifique, ES pour le bac général économique et social, ou STI2D, et "21" pour une autre situation.
- La 3e variable est le classement obtenu sur Parcoursup avant l'admission en 1e année.
- On a ensuite 13 variables qui correspondent au notes dans les modules du 1e semestre, dont les intitulés sont rappelés dans la Table 1 de la page suivante.
- La 17e variable est la moyenne obtenue dans l'UE11 : moyenne des notes dans les ressources informatiques au 1e semestre.

- La 18e est la moyenne obtenue dans l'UE12 : moyenne des notes dans les ressources non informatiques au 1e semestre.
- La 19e est la moyenne obtenue dans l'UE11 : moyenne des notes dans les ressources informatiques au 2e semestre.
- La 20e est la moyenne obtenue dans l'UE12 : moyenne des notes dans les ressources non informatiques au 2e semestre.

Programme du semestre 1			
UE	ID Module	Libellé module	Coefficient
UE11	M1101	Introduction aux systèmes informatiques	3.5
	M1102	Introduction à l'algorithmique et à la programmation	3.5
	M1103	Structures de données et algorithmes fondamentaux	2.5
	M1104	Introduction aux bases de données	3.5
	M1105	Conception de documents et d'interfaces numériques	2.5
	M1106	Projet tutoré - Découverte	1.5
UE12	M2101	Mathématiques discrètes	2.5
	M2102	Algèbre linéaire	2
	M2103	Environnement économique	1.5
	M2104	Fonctionnement des organisations	2.5
	M2105	Expression-Communication - Fondamentaux de la communication	2
	M2106	Anglais et Informatique	1.5
	M2107	PPP - Connaître le monde professionnel	1

TABLE 1 – Programme du semestre 1 du DUT Informatique - Version 2013

On a choisi ces données pour se faire une idée des informations qui peuvent permettre de prévoir la réussite d'un.e étudiant.e au 2e semestre.

1 Import des données, mise en forme

On commence par importer dans Python les données de `Vue.csv`. On voit ensuite comment résoudre quelques problèmes de mise en forme.

2. Importer dans Python le fichier `Vue.csv` sous forme d'un DataFrame `VueDF`.
3. Transformer ce DataFrame en un Ndarray `Vue`.
4. Séparer ce tableau `Vue` en 2 Ndarray : un tableau `VueStr` qui contient les colonnes avec des chaines de caractères, et un tableau `VueNum` contenant les autres colonnes.
5. Pour les 2 tableaux `VueStr` et `VueNum`, afficher leur attribut `dtype`, qui correspond au type de données attribué à leurs contenus.

Le type `'O'`, autrement appelé type `object`, est le type "'fourre-tout"' d'un tableau Numpy, qui prend donc le maximum d'espace de stockage. Il est utilisé par défaut quand un tableau contient différents types d'objects : ici c'était le cas du tableau `Vue`, qui contenait des chaines de caractère et des `float`, et donc par héritage c'est le cas des tableaux `VueStr` et `VueNum`.

6. Changer le type de vos tableaux `VueStr` et `VueNum` pour des types plus appropriés à l'aide de la méthode `astype` : `numpy.org/doc/stable/reference/generated/numpy.ndarray.astype.html`

2 Recherche de corrélations

La commande suivante permet de calculer la matrice des corrélations `Correls` pour nos données :

```
Correls=np.corrcoef(VueNum,rowvar=False)
```

L'option `rowvar=False` sert à préciser que nos variables ne correspondent pas aux lignes mais bien aux colonnes de notre tableau.

Voici les premières lignes de la matrice `Correls` :

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
0	1	-0.014	-0.29	-0.19	-0.18	-0.25	-0.25	-0.08	-0.056	-0.14	-0.22	-0.052	-0.068	-0.055	-0.23	-0.15	-0.15	0.062
1	-0.014	1	0.68	0.69	0.7	0.64	0.33	0.55	0.38	0.75	0.57	0.61	0.67	0.018	0.88	0.77	0.55	0.7
2	-0.29	0.68	1	0.8	0.64	0.67	0.29	0.52	0.38	0.7	0.61	0.71	0.58	0.13	0.9	0.77	0.51	0.56
3	-0.19	0.69	0.8	1	0.63	0.57	0.26	0.48	0.31	0.74	0.68	0.68	0.61	0.17	0.88	0.79	0.55	0.61
4	-0.18	0.7	0.64	0.63	1	0.45	0.29	0.52	0.24	0.61	0.61	0.57	0.72	0.089	0.83	0.73	0.64	0.6
5	-0.25	0.64	0.67	0.57	0.45	1	0.31	0.18	0.2	0.58	0.35	0.5	0.35	0.034	0.73	0.5	0.26	0.29

Cette matrice a 18 lignes et 18 colonnes, où 18 est le nombre de colonnes de notre tableau de données. La case située à la i-eme ligne de la j-eme colonne contient le coefficient de corrélation de la i-eme variable avec la j-eme variable.
Par exemple : la 1e variable de notre tableau de données est le rang sur Parcoursup. La 1e ligne du tableau `Correls` contient les coefficients de corrélation entre le rang et chacune des notes de notre tableau de données. Le coefficient de corrélation entre le rang et la ressource m1102 est donc de -0.29.

7. Quelles sont les 7 variables statistiques les plus corrélées avec le rang d'admission sur Parcoursup ?
8. Avec les intitulés des matières (Table 1 page précédente), quelle analyse pouvez-vous faire de la réponse à la question précédente ?
9. A l'aide de la matrice `Correls` (visuellement, sans taper un code) : quelles sont les 4 matières du 1e semestre qui sont le plus corrélées avec la moyenne en informatique du 2e semestre (UE12) ?

   Que conseilleriez-vous alors à un.e futur.e étudiant.e de 1e année pour réussir son 2e semestre ?

3 Régression linéaire multiple

A la question 20, on a repéré, parmi nos variables, celles qui avaient les + fortes corrélations avec la moyenne en informatique au 2e semestre. On va maintenant aller plus loin en réalisant une régression linéaire multiple avec :

- les notes en UE21 comme variable endogène
- les 4 variables de la question 9 comme variables explicatives.

10. Créer un tableau `Y` à une colonne, contenant la variable endogène, et un tableau `X` à 4 colonnes contenant les variables explicatives.
11. Créer des versions normalisées `Yn` et `Xn` de ces 2 tableaux, comme fait dans le TP1 des Statistiques pour la Saé.
12. Calculer les paramètres de cette régression linéaire multiple en utilisant la fonction LinearRegression de sklearn (voir TP précédent) sur `Yn` et `Xn`.
13. Interpréter les paramètres obtenus.
14. Calculer le coefficient de corrélation multiple à l'aide la fonction LinearRegression.

    Attention, on rappelle que la commande `linear_regression.score` *donne le carré du coefficient de corrélation.*
15. La corrélation est-elle forte ?
16. Calculez une estimation de votre future moyenne au semestre 2 en fonction de vos notes du 1e semestre.

4 Comparaison de diagrammes en batons

Dans cette partie, on utilise le fichier `Vue2.csv`, à télécharger sur Moodle. Il contient moins de variables que précédemment dans `Vue.csv`, mais pour plus d'étudiant.es car le classement attribué aux étudiant.es sur Parcoursup n'était disponible que pour l'année 2016.

17. Reprendre la démarche des questions 2, 3, 4 et 6 sur le fichier `Vue2.csv`, afin d'obtenir les Ndarray `Vue2Str` de chaines de caractères, et `Vue2Num` de `float`.

Dans cette partie, on souhaite observer si les résultats au S1 sont différents selon l'origine socio-professionnelles des parents des étudiant.es.
Comme l'origine socio-professionnelle n'est pas une donnée numérique, on ne peut pas a priori chercher de corrélation. On va donc s'y prendre autrement, en comparant visuellement les résultats selon les situations.

18. Grâce à la colonne `cat_socio_parent` de votre tableau `VueStr`, créer un tableau `Lignes_ma_categorie` contenant les indices des lignes des étudiant.es ayant un parent de la même catégorie socio-professionnelle que vous. On pourra utiliser la fonction `np.argwhere` : `numpy.org/doc/stable/reference/generated/numpy.argwhere.html`
19. A l'aide de `Lignes_ma_categorie`, créer des tableaux `UE11_ma_categorie` et `UE21_ma_categorie` contenant respectivement les notes dans l'UE11 et l'UE21 des étudiant.es ayant un parent de la même catégorie socio-professionnelle que vous.
20. Tracer un diagramme en bâtons des notes dans l'UE11 de l'ensemble des étudiant.es. On imposera que les intervalles des bâtons soient `np.arange(21)`.
21. Tracer maintenant le diagramme en bâtons des notes contenues dans `UE11_ma_categorie`. Si vous mettez la commande juste à la suite de celle de la question précédente, les 2 diagrammes se superposent en 2 couleurs.

22. Faire la même superposition de 2 diagrammes pour les notes en UE21. Pour qu'ils ne se superposent pas aux précédents diagrammes, intercaler la commande `plt.figure()` avant de créer vos nouveaux diagrammes.
23. Analyser les diagrammes obtenus.

5 Comparaison de boites à moustaches

On va maintenant représenter côte à côte plusieurs boites à moustaches représentant plusieurs notes, et selon la catégorie sociale, afin de pouvoir les comparer.

24. Représenter la boite à moustaches des notes en UE11 de l'ensemble des étudiant.es du fichier.
25. Représenter maintenant la boite à moustaches des notes contenues dans `UE11_ma_categorie`.

    Afin que les boites à moustaches ne se superposent pas, vous pouvez indidiquer la position (abscisse) souhaitée de la boite. L'exemple suivant place la boite en l'abscisse x=2, et indique en légende ce qu'elle contient :

    ```
    plt.boxplot(UE21_ma_categorie,positions=[2],labels=['UE21'])
    ```
26. Placer à la suite les mêmes boites à moustaches mais pour les notes de l'UE21.
27. Analyser les diagrammes obtenus.