Probabilités discrètes et statistiques descriptives

L2 Université Lyon 1 2022-2023

TP noté numéro 89965 à rendre avant le mercredi 21 décembre 2022 à 23h59

Il faut charger sur Tomuss une feuille de calcul python et une présentation des réponses aux questions de cette feuille dans les deux colonnes Tomuss prévues à cet effet.

Instructions pour la synthèse de présentation des résultats du TP

Le projet est à rédiger avec Word ou LibreOffice (ou LaTeX si vous savez déjà l'utiliser). Mais vous devrez convertir votre fichier au format **pdf**. Attention ! Quand on exporte un fichier en pdf, il manque parfois les formules mathématiques ou des graphes. Il faut soigneusement vérifier le fichier pdf et si nécessaire, essayer d'autres convertisseurs pdf (il en existe de nombreux gratuits sur le web, comme cutepdf).

Le nom des fichiers devra contenir le nom de famille des deux membres du groupe (sans accents) sous la forme Fisher_Pearson.pdf et Fisher_Pearson.py.

Rédigez soigneusement. Commentez à chaque fois vos graphes et vos résultats.

La première page devra contenir : nom, prénom, formation, année universitaire, nom de l'UE, date, numéro du sujet.

- Ajoutez des numéros de page, s'ils n'y sont pas.
- Vérifiez les graphiques : il faut qu'ils soient centrés, que leurs titres soient en français.
- Harmonisez la mise en forme des titres. Le style des paragraphes doit être justifié, pas aligné à gauche.
- Relisez pour l'orthographe (au moins deux fois), relisez pour la ponctuation.
- Écrivez le code python dans un fichier à part (colonne à part sur Tomuss). Si vous copiez du code python dans votre présentation, utilisez une police adéquate (avec un interlettrage fixe, comme `Lucida Console`).

  Dans tous les cas, nettoyez le code des lignes inutiles, commentez-le de manière raisonnable. 3 points seront accordés à la présentation de la synthèse et à la lisibilité du code !

Dans la suite on suppose avoir chargé les librairies suivantes :

```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pan
import scipy.stats as st
import statsmodels.api as sm
```

Exercice 1. Statistiques
Chargez le jeu de données :

```
Air=pan.read_csv('http://tinyurl.com/y39an7ef/Data89965.csv',sep='\t',na_values='-')
```

Ce jeu de données contient des informations issues de relevés des 4 principaux polluants du trafic routier mesurés à 4 stations de la région Auvergne-Rhône-Alpes en 2015 et 2016. Tous les polluants sont mesurés en microgrammes par mètre cube d'air.

a. Le dioxyde d'azote, monoxyde d'azote et les particules de taille 10 micromètres (PM10) sont relevés aux stations suivantes : au sud de Lyon sur l'A7, plus au sud sur l'A7 au niveau du nord de l'Isère, sur la rocade de Grenoble et sur l'A71 à Clermont-Ferrand.

b. Les particules de taille 2,5 micromètres (PM2,5) sont relevées à toutes les stations sauf celle de Clermont-Ferrand.

Les valeurs indiquées "-" dans le fichier, ou NA ou nan après importation par `pandas`, sont les valeurs non disponibles, correspondant à des jours de non-observation.

On donnera les réponses numériques arrondies avec TROIS décimales.

A. Informations générales

1. Quel est le type de variable statistique de chacune des variables (nominale, ordinale, quantitative discrète ou continue) ?
2. Quel est le nombre de jours d'observation de l'échantillon ? Quel est le nombre de jours où les particules PM10 sont mesurées à toutes les stations ?(Indication : vous pouvez utiliser la fonction `isna`).
3. Créez deux nouvelles variables booléennes PMObs, AzoteObs ayant, chaque jour, pour valeur « VRAI » respectivement si toutes les mesures de particules ont été observées ce jour-ci ou si tous les autres polluants ont été observés. Donnez la table de contingence de ces deux variables. Quel est le nombre de jours où tous les polluants sont observés ?
4. En ignorant les jours de non observation des particules (en utilisant par exemple la fonction `dropna`), trouvez la moyenne empirique, variance empirique, variance empirique non-biaisée et le quartile à 25 % de la variable mesurant les particules PM10 sur l'A7 à Lyon.

B. Intervalles de confiance pour les probabilités de dépasser un seuil de Pollution aux particules

1. Créer deux variables vallant 0 ou 1 telles que :
   — la première vaut 1 si et seulement si le seuil d'information pour les particules PM10 de 50 microgrammes par mètre cube d'air est dépassé sur l'A7 à Lyon,
   — la seconde vaut 1 si et seulement si le seuil d'alerte pour les particules PM10 de 80 microgrammes par mètre cube d'air est dépassé sur l'A7 à Lyon.

   Quel est la loi théorique de ces variables ?
2. Avec la fonction `st.binomtest().proportion_ci()`, trouver un intervalle de confiance exact pour les probabilités p que le seuil d'information pour les particules PM10 soit dépassé sur l'A7 à Lyon, séparément pour un jour typique des années 2015 et 2016. Même question pour le seuil d'alerte.
3. Depuis le premier janvier 2005, le seuil d'information pour les particules PM10 ne doit pas être dépassé plus de 35 jours par an, soit une probabilité journalière maximum de dépassement de $p_0 = \frac{35}{365}$. Un riverain de l'A7 veut tester si il y a un risque que ce seuil ne soit pas respecter. Tester pour lui l'hypothèse alternative que ce seuil a été respecté, soit $p < p_0$, séparément pour les années 2015 et 2016.

4. Devant l'inquiétude du riverain, la municipalité veut tester si ce seuil a été significativement dépassé. Quel test réalise-t-elle ? Que concluez-vous pour les années 2015 et 2016 (discuter la significativité statistique du résultat).

C. Une Régression Linéaire

1. Dans la suite, on considère ensemble toutes les particules (on ajoute les chiffres pour les particules PM10 et PM2.5) et tous les oxydes d'azote ensemble (on ajoute les chiffres du dioxyde et du monoxyde). Créez un data.frame `dfs` contenant les 7 mesures obtenues ainsi (on ne garde les mesures que si les deux mesures sont disponibles) et oubliant tous les jours où il manque une des mesures concernées.

   Dans la suite, on travaille sur les données du data.frame `dfs`.

   Calculer la matrice de covariance de `dfs`.

2. On cherche à savoir si la pollution en Oxydes d'Azote à Grenoble y est significativement corrélée à celle sur l'A7 à Lyon x. Effectuez un test de corrélation entre $y' = \ln(y)$ et $x' = \ln(x)$. Quelle hypothèse doit-être vérifiée pour effectuer ce test de corrélation (de Pearson) ? Que concluez-vous ?

Exercice 2. Simulation

On rappelle que la fonction `st.uniform.rvs` permet de simuler une loi uniforme continue en obtenant un vecteur dont les nombres sont uniformément répartis dans un intervalle.

1. On donne les commandes suivantes :

```
U=st.uniform.rvs(0,1,size=10000);
S=(-np.log(U)*10); R=np.ceil(S)
```

   En comparant un histogramme de S et une densité bien choisie, émettez une hypothèse sur la loi de S.

2. Calculez la probabilité empirique que $R = 2$ puis la probabilité que $R = 3$ sachant $R \geqslant 1$.

3. On veut vérifier que R est un échantillon de loi geométrique $\mathcal{G}(p)$. En comparant le diagramme en bâton de R avec les valeurs théoriques d'une loi géométrique pour p bien choisie, émettez une hypothèse sur p. (Il n'est pas demandé de tester statistiquement cette hypothèse en détail)

4. On considère le programme suivant :

```
x=0; y= [0]; j= 1
for i in range(len(R)):
    x= x + R[i]
    if(x>8):
        x = 0
        y = np.concatenate((y,[i+1-j]))
        j = i+2
T = y[1:len(y)]
```

   On veut vérifier que T représente l'échantillon d'une variable de loi binomiale $\mathcal{B}(n, p)$. En remplaçant R par une suite de 1 dans le programme précédent, trouver la valeur maximale n que T peut théoriquement prendre.

   On revient au T de l'énoncé (calculé avec le R initial). Calculez la moyenne empirique de T. En déduire un candidat pour les paramètres de la loi de T. En comparant le diagramme en bâton de T avec le diagramme d'une loi discrète bien choisie, vérifiez graphiquement que l'hypothèse sur la loi de T est réaliste.

5. Réalisez un test du χ^2 d'adéquation à une loi binomiale que vous avez trouvé pour tester votre hypothèse. (On calculera en particulier la p-valeur du test pour discuter de la significativité statistique du résultat). Si vous répétez la simulation aléatoire 100 fois (avec une boucle), compter combien de fois vous conservez l'hypothèse nulle (au niveau de risque 5%) ? Qu'en concluez-vous sur votre hypothèse concernant la simulation ?

6. Expliquez ce que fait le programme ci-dessus pour obtenir T. (Si nécessaire, on pourra représenter des tracés intermédiaires pour comprendre les étapes). Comparer le temps de simulation à celui obtenu avec `st.binom.rvs`.