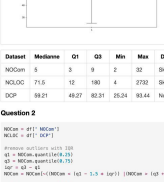
IFT3913 TP3

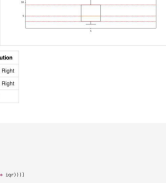
Ethan Mai 20146661

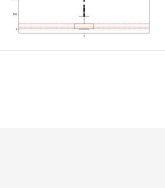
https://github.com/AmiEthanMai/IFT3913_TP3.git

Question N0.1

```
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd

df = pd.read_csv('/Users/ethanmai/Downloads/jfreechart-stats.csv')
data = df['DCP']
#creating plot
plt.boxplot(data)
#show the median of the data
plt.axhline(data.median(), color='r', linestyle='dashed', linewidth=1)
#show the first quartile of the data
plt.axhline(data.quantile(0.25), color='r', linestyle='dashed', linewidth=1)
#show the third quartile of the data
plt.axhline(data.quantile(0.75), color='r', linestyle='dashed', linewidth=1)
```





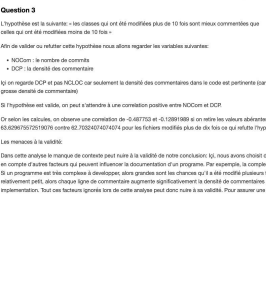


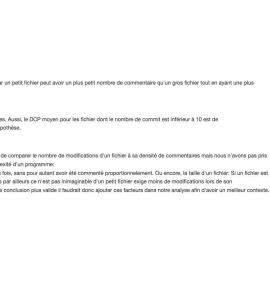
Dataset	Medianne	Q1	Q3	Min	Max	Distribution
NOCom	5	3	9	2	32	Skewed Right
NCLOC	71.5	12	180	4	2732	Skewed Right
DCP	59.21	49.27	82.31	25.24	93.44	Normal

Question 2

```
NOCom = df['NOCom']
NCLOC = df['NCLOC']

#remove outliers with IQR
q1 = NOCom.quantile(0.25)
q3 = NOCom.quantile(0.75)
iqr = q3 - q1
NOCom = NOCom[(~(NOCom < (q1 - 1.5 * iqr)) |(NOCom > (q3 + 1.5 * iqr)))]

q1 = NCLOC.quantile(0.25)
q3 = NCLOC.quantile(0.75)
iqr = q3 - q1

NCLOC = NCLOC[(~(NCLOC < (q1 - 1.5 * iqr)) |(NCLOC > (q3 + 1.5 * iqr)))]

#remove the missing values
if len(NCLOC) > len(NOCom):
    NCLOC = NCLOC[:len(NOCom)]
else:
    NOCom = NOCom[:len(NCLOC)]

diff = len(NCLOC) - len(NOCom)

#plot the data
plt.scatter(NOCom, NCLOC, color='red', marker='o', s=10)
#calculate the correlation coefficient
corr = np.corrcoef(NOCom, NCLOC)
print(corr)
#plot the regression line
m, b = np.polyfit(NOCom, NCLOC, 1)
plt.plot(NOCom, m*NOCom + b, color='blue')

#add title
plt.title("NCLOC vs DCP without outliers")
#add labels
plt.xlabel("NOCom")
plt.ylabel("DCP")
plt.show()
```

Dataset	correlation	m	b
NOCom vs DCP	-0.4827753	-1.8300216935332569	75.80959171110902
NOCom vs DCP without outliers	-0.12891969	-0.6289842618132071	67.53760267576802
NOCom vs NCLOC	0.71457166	26.04699832348385	-87.99240691592912
NOCom vs NCLOC without outliers	0.24695678	7.000198878430157	55.954043540046627





Question 3

L'hypothèse est la suivante: « les classes qui ont été modifiées plus de 10 fois sont mieux commentées que celles qui ont été modifiées moins de 10 fois »

Afin de valider ou réfuter cette hypothèse nous allons regarder les variables suivantes:

- NOCom : le nombre de commits
- DCP : la densité des commentaire

Ici on regarde DCP et pas NCLOC car seulement la densité des commentaires dans le code est pertinente (car un petit fichier peut avoir un plus petit nombre de commentaire qu'un gros fichier tout en ayant une plus grosse densité de commentaire)

Si l'hypothèse est valide, on peut s'attendre à une correlation positive entre NOCom et DCP.

Or selon les calculs, on observe une correlation de -0.4827753 et -0.12891989 si on retire les valeurs abérantes. Aussi, le DCP moyen pour les fichier dont le nombre de commit est inférieur à 10 est de 63.82967557251907 contre 62.703240740740740 pour les fichiers modifiés plus de dix fois ce qui refutte l'hypothèse.

Les menaces à la validité:

Dans cette analyse le manque de contexte peut nuire à la validité de notre conclusion; Ici, nous avons choisit de comparer le nombre de modifications d'un fichier à sa densité de commentaires mais nous n'avons pas pris en compte d'autres facteurs qui peuvent influencer la documentation d'un programe. Par exemple, la complexité d'un programme:
Si un programmes est très complexe à développer, alors grandes sont les chances qu'il a été modifié plusieurs fois, sans pour autant avoir été commenté proportionnellement. Ou encore, la taille d'un fichier: Si un fichier est relativement petit, alors chaque ligne de commentaire augmente significativement la densité de commentaires par ailleurs ce n'est pas intraignable d'un petit fichier exige moins de modifications lors de son implementation. Tout ces facteurs ignorés lors de cette analyse peut donc nuire à sa validité. Pour assurer une conclusion plus valide il faudrait donc ajouter ces facteurs dans notre analyse afin d'avoir un meilleur contexte.