

Travail Pratique de SDD4142-Statistiques

TEAM 6 : MATHIAM FAYE

BAYE THILOR SENE

FATOU DIARRA

Question 3 : Reponses aux questions

- 1- Enregistrer les données dans un format adapté pour une lecture par la suite avec Python sachant que la première ligne du fichier smp.csv correspond au noms des variables. Vérifier si vous avez une structure de 799 observations et 26 variables.

Home Page - Select or creat STAT EXERCICE 2 - Jupy X _main.pdf Enonce-projet_2020.pdf + v

localhost:8888/notebooks/STAT%20EXERCICE%202.ipynb

jupyter STAT EXERCICE 2 (auto-sauvegardé) Logout

File Edit View Insert Cell Kernel Widgets Help Fiable Python 3

Exécuter Code

```
liste_prisomme.statypes
```

```
dimension: (799, 26)
nombre de ligne: 799
nombre de colonne: 26
```

```
Out[87]: age          float64
         prof         category
         duree        category
         discip        category
         n.enfant      float64
         n.fratricie   int64
         ecole         category
         separation    category
         juge.enfant   category
         place         category
         abus          category
         grav.cons     category
```

Taper ici pour rechercher

12:45 01/04/2020

2- Changer les types des variables. Vous devez obtenir le résultat suivant :

Home Page - Select or creat STAT EXERCICE 2 - Jupy X _main.pdf Enonce-projet_2020.pdf + v

localhost:8888/notebooks/STAT%20EXERCICE%202.ipynb

jupyter STAT EXERCICE 2 (auto-sauvegardé) Logout

File Edit View Insert Cell Kernel Widgets Help Fiable Python 3

Exécuter Code

```
liste_prisomme.statypes
```

```
dimension: (799, 26)
nombre de ligne: 799
nombre de colonne: 26
```

```
Out[87]: age          float64
         prof         category
         duree        category
         discip        category
         n.enfant      float64
         n.fratricie   int64
         ecole         category
         separation    category
         juge.enfant   category
         place         category
         abus          category
         grav.cons     category
```

Taper ici pour rechercher

12:45 01/04/2020

The screenshot shows a Jupyter Notebook titled "STAT EXERCICE 2 (modifié)". The code cell contains a list of variables and their corresponding data types:

Variable	Type
juge.enfant	category
place	category
abus	category
grav.cons	category
dep.cons	category
ago.cons	category
ptsd.cons	category
alc.cons	category
subst.cons	category
scz.cons	category
char	category
rs	category
ed	category
dr	category
suicide.s	float64
suicide.hr	category
suicide.past	category
dur.interv	float64
dtype:	object

3- Calculer la moyenne, la variance, et l'écart type pour chacune des variables suivantes:
La méthode `describe()`, nous retourne des statistiques descriptives sur l'ensemble des colonnes numériques de notre DataFrame.

The screenshot shows the same Jupyter Notebook with the code `print(liste_prisonniers.describe())` executed. The output displays the dimensions of the DataFrame and a summary of its numerical columns:

```
dimension: (799, 26)
nombre de ligne: 799
nombre de colonne: 26
```

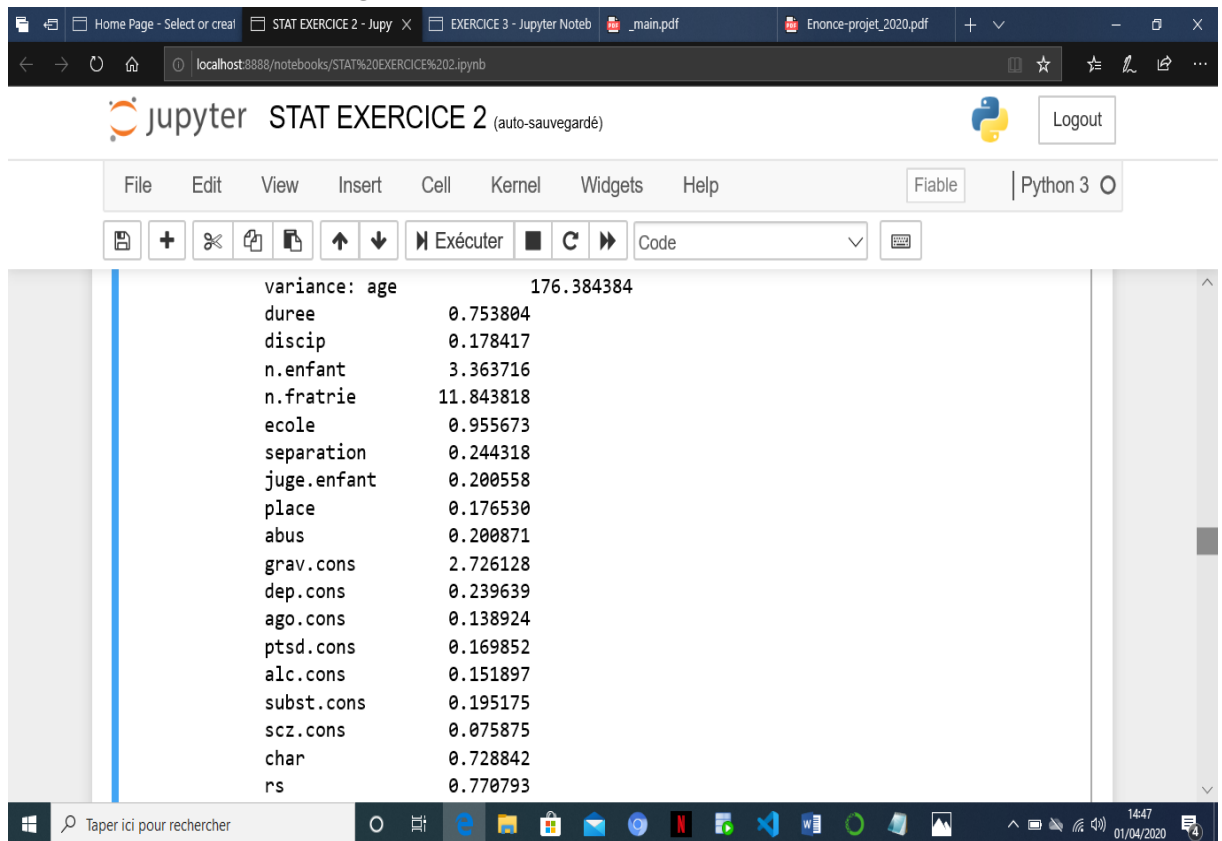
	age	n.enfant	n.fratrerie	suicide.s	dur.interv
count	797.000000	773.000000	799.000000	758.000000	749.000000
mean	38.899624	1.755498	4.286608	0.794195	61.891856
std	13.280978	1.834044	3.441485	1.435488	19.669605
min	19.000000	0.000000	0.000000	0.000000	0.000000
25%	28.000000	0.000000	2.000000	0.000000	48.000000
50%	37.000000	1.000000	3.000000	0.000000	60.000000
75%	48.000000	3.000000	6.000000	1.000000	75.000000
max	83.000000	13.000000	21.000000	5.000000	120.000000

Below the output, the code cell shows the imports for pandas and numpy:

```
Entrée [51]: import pandas as pd
import numpy as np
```

Pour les variance :La méthode `<< liste_prisonniers.var()`
`>>` nous renvoie la variance de tous les colonnes numériques du DataFrame `liste_prisonniers`. Et pour un

seul colonne(exemple age) du DataFrame, on utilise << liste_prisonniers.age.var()>>

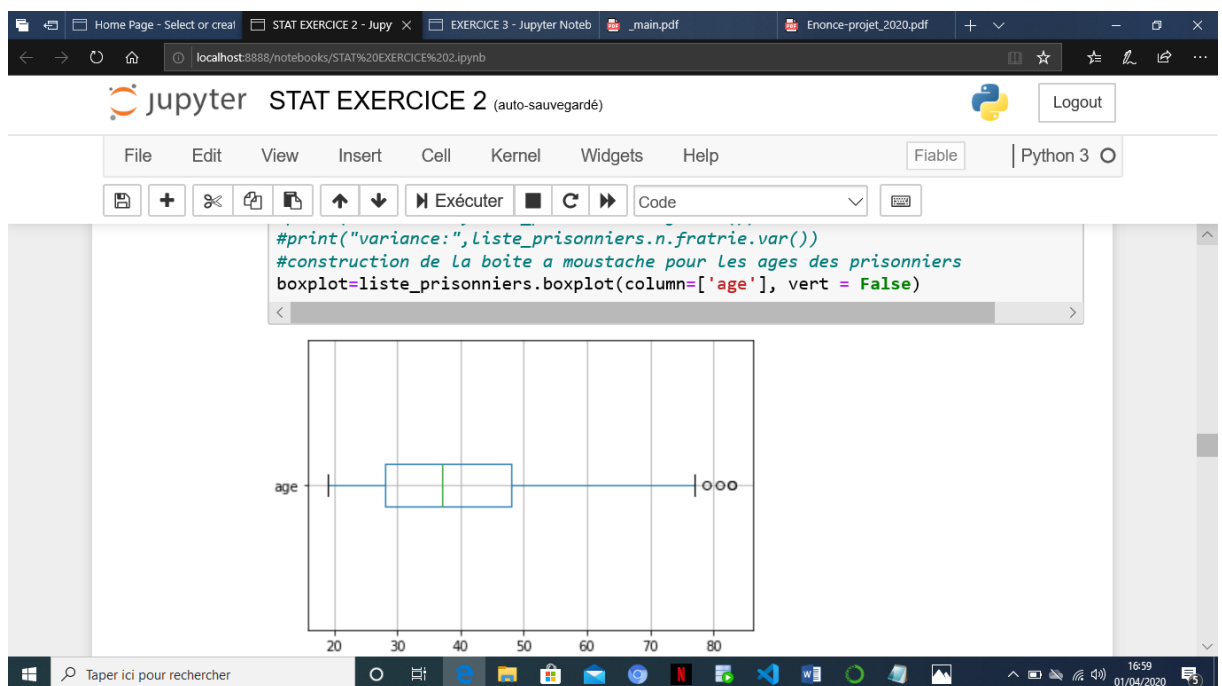


	MOYENNE	VARIANCE	ECART TYPE
age	38.899624	176.384384	13.280978
n.enfant	1.755498	3.363716	1.834044
n.fratrerie	4.286608	11.843818	3.441485
dur.interv	61.891856	386.893369	19.669605
Les quantiles des ages	1 ^{er} quantiles	2ieme quantiles	3iem quantiles
	28.000000	37.000000	48.000000

4- Tracer le boxplot pour la variable age. Quelles conclusions en tirez-vous?

- Plus de $\frac{3}{4}$ des prisonniers ont un âge inférieure a 50ans.
- De plus, la dispersion des âges des prisonniers dans :
L'âge min et le premier quartile (première moustache)
Le premier quartile et la médiane (partie inférieure de la boite a moustache)
La médiane et le troisième quartile (partie droite de la boite a moustache) est sensiblement égale.
- La dispersion des âges des prisonniers est très grande entre le troisième quartile et l'âge max. en plus il y a trois valeurs aberrantes c'est-à-dire trois prisonniers dont l'âge est supérieur à $\min(\max, Q3+1,5(Q3-Q1))$.

Conclusion : Une observation plus détaillée des données est nécessaire.



5- Afficher les données pour les agriculteurs qui ont plus de 2 enfants.

La methode :

```
print(liste_prisonniers.loc[(liste_prisonniers['prof']=="agriculteur") & (liste_prisonniers['n.enfant'] > 2),:])
```

Permet d'avoir le resultat suivant :

```

age      prof      duree  discip  n.enfant  n.fratrerie  ecole  separation \
14      64.0  agriculteur  NaN    0.0      3.0      2    1.0      0.0
311     42.0  agriculteur   4.0    0.0      3.0      6    1.0      0.0
390     36.0  agriculteur   4.0    1.0      3.0      4    3.0      1.0
441     79.0  agriculteur   5.0    0.0      5.0      6    2.0      0.0

juge.enfant  place  ...  subst.cons  scz.cons  char  rs  ed  dr  suicide.s \
14           0.0   0.0  ...           0           0  1.0  1.0  1.0  3.0      0.0
311          0.0   0.0  ...           0           0  2.0  1.0  3.0  2.0      3.0
390          1.0   1.0  ...           1           0  1.0  NaN  3.0  1.0      0.0
441          0.0   0.0  ...           0           0  1.0  2.0  1.0  1.0      0.0

suicide.hr  suicide.past  dur.interv
14           0.0           0.0      80.0
311          1.0           0.0      NaN
390          0.0           0.0      NaN
441          0.0           0.0      85.0

[4 rows x 26 columns]

```

6- Calculer les fréquences des modalités de la variable prof.

```

ouvrier      227
sans emploi  222
employe      135
artisan       90
prof.intermediaire  58
autre        31
cadre        24
agriculteur   6
Name: prof, dtype: int64
{'ouvrier': 227, 'sans emploi': 222, 'employe': 135, 'artisan': 90, 'prof.intermediaire': 58, 'autre': 31, 'cadre': 24, 'agriculteur': 6}
dict_keys(['ouvrier', 'sans emploi', 'employe', 'artisan', 'prof.intermediaire', 'autre', 'cadre', 'agriculteur'])
ouvrier  sans emploi  employe  artisan  prof.intermediaire  autre \
0  28.625473   27.994956  17.02396  11.349306           7.313997  3.909206

cadre  agriculteur  total
0  3.026482    0.75662  100.0

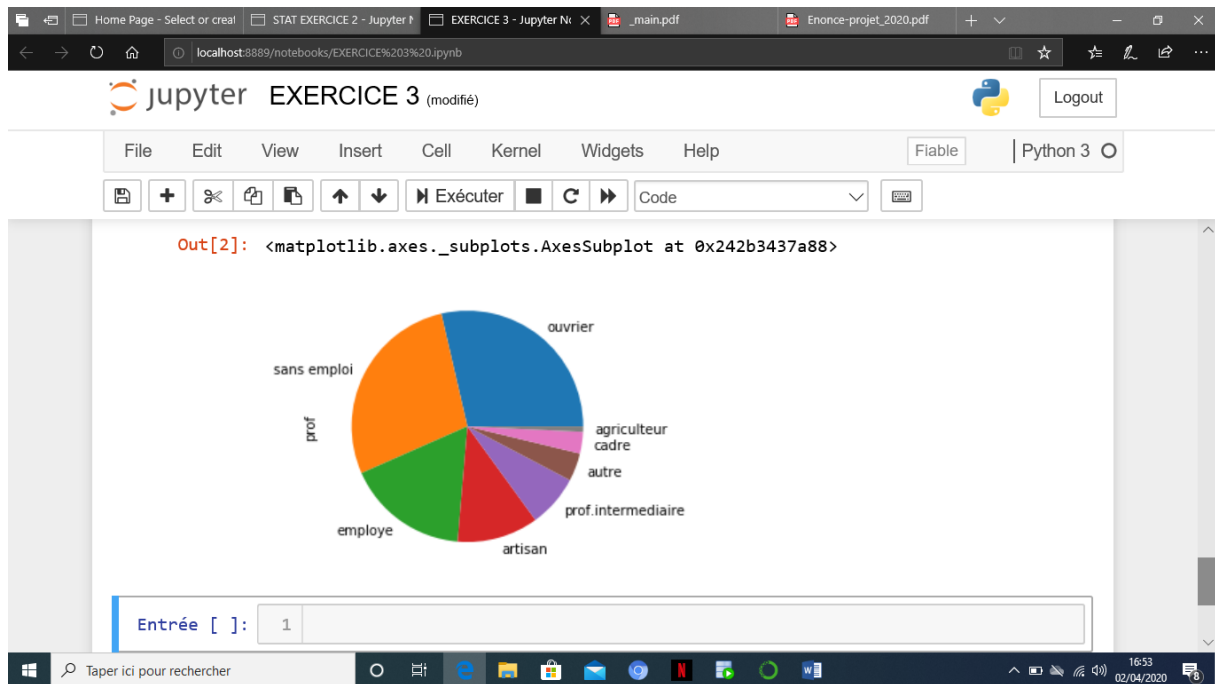
```

Quelle est la catégorie modale? : **la catégorie modale est : « ouvrier »** ☹️

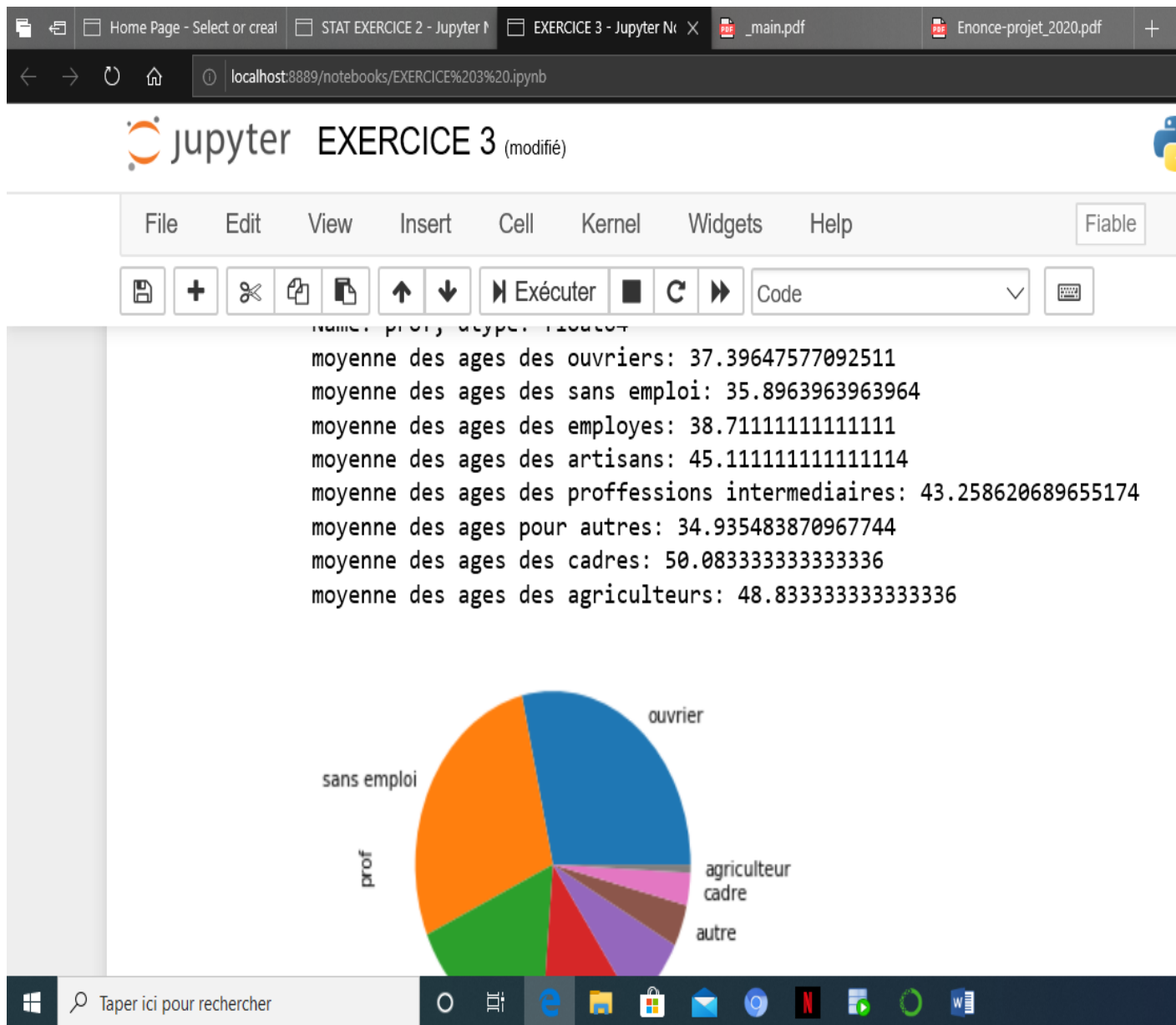
7. Tracer le diagramme circulaire de la variable profession

#diagramme circulaire de la variable profession

resultat.plot.pie()



8. Donner les moyennes des âges par profession



9- Donner la table des effectifs pour les variables prof incluant les "NaN".

The screenshot shows a Jupyter Notebook interface in a web browser. The browser tabs include 'Nous ne pouvons', 'Enonce-projet_202', 'presentation_donn', 'fr_Tanagra_Data_M', 'Home Page - Selec', 'EXERCICE 3 - Jupyter', and 'STAT EXERCICE'. The address bar shows 'localhost:8888/notebooks/STAT%20EXERCICE%202.ipynb'. The notebook title is 'STAT EXERCICE 2 (modifié)'. The menu bar includes 'File', 'Edit', 'View', 'Insert', 'Cell', 'Kernel', 'Widgets', and 'Help'. The toolbar has icons for saving, adding, deleting, copying, pasting, and running code. The code cell contains three print statements:

```
102 print('Valnull.suicide.hr:', 799-760 )
103 print('Valnull.suicide.past:', 799- 785 )
104 print('Valnull.dur.interv:', 799-749 )
```

The output of the code cell shows a list of professions and their corresponding counts:

```
ouvrier          227
sans emploi      222
employe          135
artisan           90
prof.intermediaire 58
autre            31
cadre            24
agriculteur       6
Name: prof, dtype: int64
Valnull: 6
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 799 entries, 0 to 798
Data columns (total 1 columns):
prof    793 non-null category
```

10. Donner le nombre de "Nan" pour chaque variable.

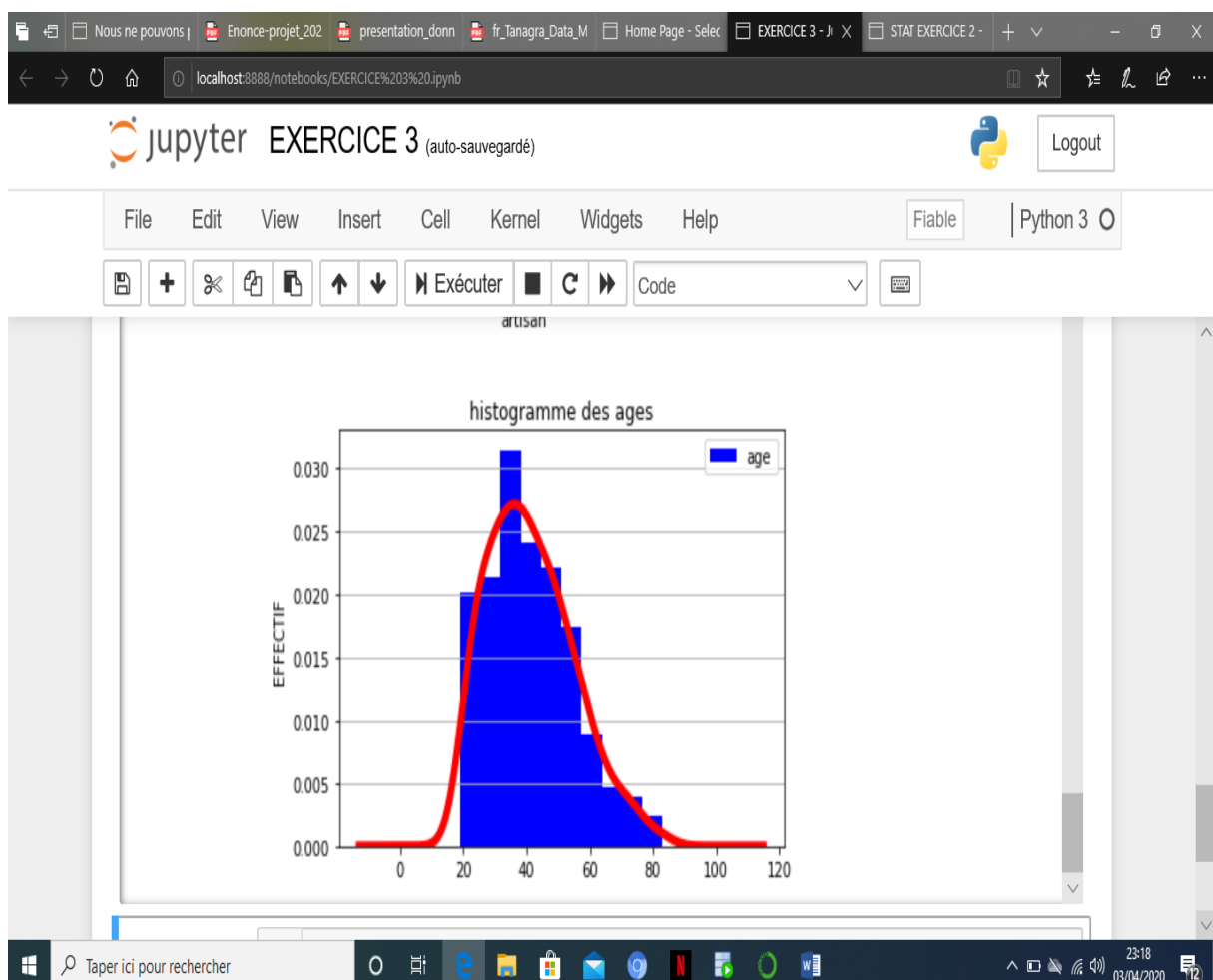
En faisant une itération sur les colonnes du DataFrame voici le résultat obtenu

```
age 2
prof 6
duree 223
discip 6
n.enfant 26
ecole 5
separation 11
juge.enfant 5
place 7
abus 7
grav.cons 4
char 96
rs 103
ed 107
dr 111
suicide.s 41
suicide.hr 39
suicide.past 14
dur.interv 50
```

11. Supprimer toutes les lignes contenant des "Nan".

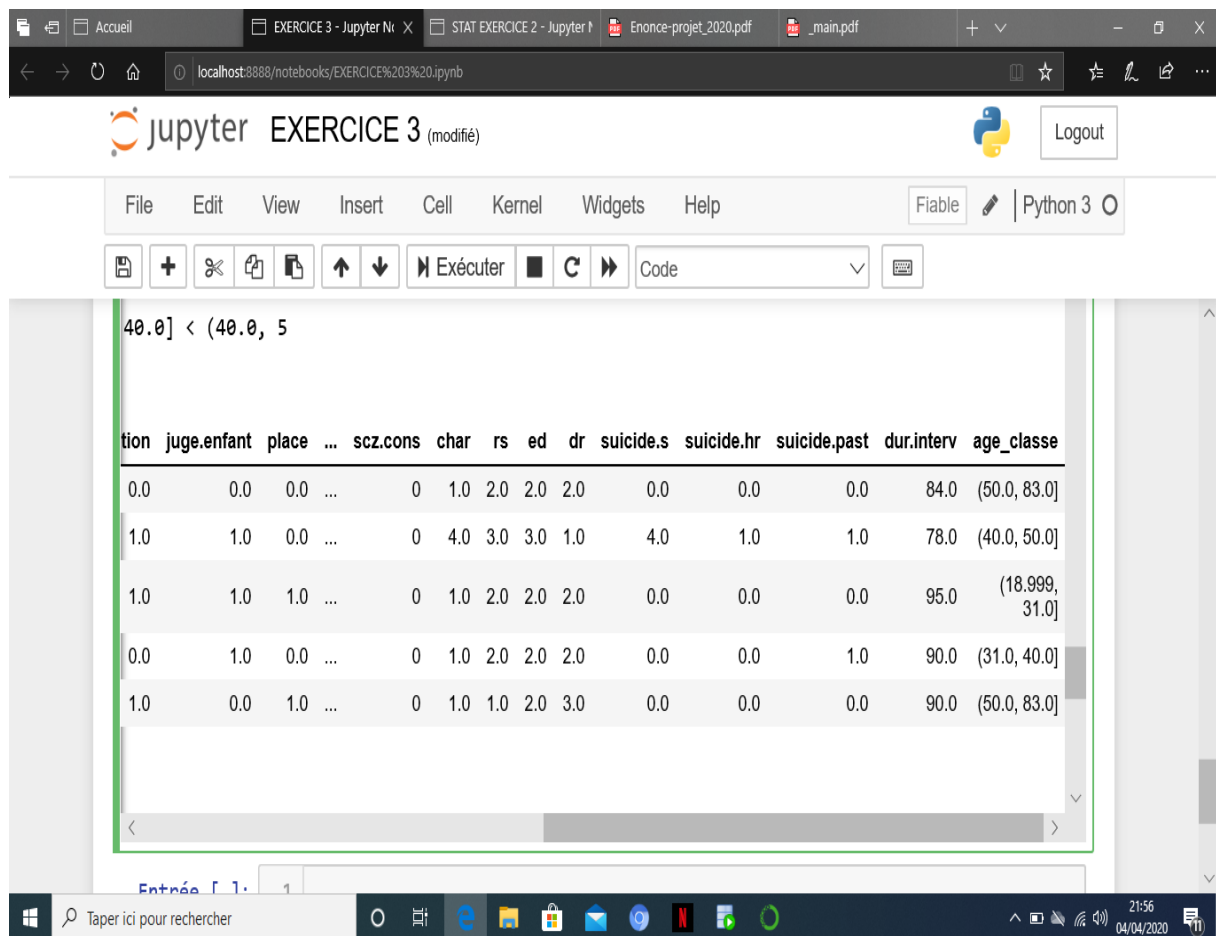
Deux méthodes peuvent être utilisées : Voir script

12. Tracer l'histogramme et la densité de la variable age sur la même figure.



13. Discrétisez la variable age. Pour ce faire on ajoutera une variable dans le DataFrame des données une nouvelle variable nommée `age_classe`. Cette variable aura 4 classes : $[\min(\text{age}), Q1]$, $]Q1, Q2]$, $]Q2, Q3]$, $]Q3, \max(\text{age})]$. ou $Q1, Q2, Q3$ sont respectivement les 3 premiers quantiles de la variable age, $\min(\text{age})$ et $\max(\text{age})$ respectivement la plus petite et la plus grande valeur de la variable age.

Après utilisation de la méthode `qcut` on peut afficher le tableau de données ci-dessous :



tion	juge.enfant	place	...	scz.cons	char	rs	ed	dr	suicide.s	suicide.hr	suicide.past	dur.interv	age_classe
0.0	0.0	0.0	...	0	1.0	2.0	2.0	2.0	0.0	0.0	0.0	84.0	(50.0, 83.0]
1.0	1.0	0.0	...	0	4.0	3.0	3.0	1.0	4.0	1.0	1.0	78.0	(40.0, 50.0]
1.0	1.0	1.0	...	0	1.0	2.0	2.0	2.0	0.0	0.0	0.0	95.0	(18.999, 31.0]
0.0	1.0	0.0	...	0	1.0	2.0	2.0	2.0	0.0	0.0	1.0	90.0	(31.0, 40.0]
1.0	0.0	1.0	...	0	1.0	1.0	2.0	3.0	0.0	0.0	0.0	90.0	(50.0, 83.0]

14. Donner les fréquences des modalités de la nouvelle variable `age_classe`.

Home Page - Select or creat EXERCICE 3 - Jupyter Nt X STAT EXERCICE 2 - Jupyter MPRA_paper_76653.pdf Enonce-projet_2020.pdf

localhost:8888/notebooks/EXERCICE%203%20.ipynb

jupyter EXERCICE 3 (modifié) Logout

File Edit View Insert Cell Kernel Widgets Help Fiable Python 3

Exécuter Code

```
...
793 (18.999, 31.0]
795 (40.0, 50.0]
796 (18.999, 31.0]
797 (31.0, 40.0]
798 (50.0, 83.0]
Name: age_classe, Length: 403, dtype: category
Categories (4, interval[float64]): [(18.999, 31.0] < (31.0, 40.0] < (40.0, 50.0] < (50.0, 83.0]]
(18.999, 31.0] 107
(31.0, 40.0] 104
(50.0, 83.0] 96
(40.0, 50.0] 96
Name: age_classe, dtype: int64
(18.999, 31.0] 0.265509
(31.0, 40.0] 0.258065
(50.0, 83.0] 0.238213
(40.0, 50.0] 0.238213
Name: age_classe, dtype: float64
```

Taper ici pour rechercher 06:32 05/04/2020