J1 – Statistiques

Satistiques descriptives

Exercice 1 – Quel pourcentile ? (*)

Dérire cette série de nombres avec un maximum de statistiques descriptives.

Hint: Utilisez `numpy` et `scipy`

Entrée [1]:

```
import numpy as np
import pandas as pd
```

Entrée [2]:

```
test_array = np.array([ 29.06114022,  26.17437913,  36.4492889 ,  54.90292031,
        49.44535829,  70.72485829,  67.46168782,  77.42488633,
         5.19166198,  46.1153849 , 102.91120315,  37.63296073,
        38.20406491,  71.23979213,  50.67913182,  71.61930794,
        36.13388738,  27.3738083 ,  80.05990108,  64.42082913,
        85.96083068,  38.48042099,  39.96362245,  62.00757552,
        83.12837179,  50.02396422,  73.35132353,  55.20438011,
        45.0256195 ,  18.21004262,  46.61125507,  29.79881717,
        19.16614263,  36.87047247,  34.71334354,  85.11379842,
        66.79951584,  61.00108181,  46.82147047,   4.4950038 ,
        59.64403006,  17.01467171,  40.00601743,  35.13955427,
        38.31776797,  54.1456781 ,  34.30530359, 103.841159  ,
        83.36621903,  43.18991314,  46.98826925,  66.62157158,
        41.79001612,  40.60759538,  65.20520983,  21.43783658,
        69.52452364,  86.3687045 ,  71.41994957,  47.33225797,
        21.115204  ,  55.05271646,  41.89208457,  55.22620396,
        51.83378269,  46.00827601,  44.26225881,  83.07739312,
        48.96878562,  64.82302883,  25.87666904,  48.55161088,
        48.19525418,  47.60694118,  42.81430297,  15.83639471,
        73.88628351,  59.81470386,  36.10382006,  54.88516162,
        63.63872644,  26.40355033,  81.54731183,  26.72902021,
        73.58336019,  29.31653704,  59.08846558,  47.91728695,
        20.65932672,  67.37507865,  29.69230719,  35.60901864,
        75.98322683,  74.13652542,  42.1707353 ,  36.14038798,
        63.02800873,  39.74962657,  23.85164459,  93.0516192 ])
```

Entrée [3]:

```
from scipy import stats

statistics = stats.describe(test_array)
statistics.mean
```

Out[3]:

```
50.677423945600005
```

Entrée [4]:

```
df = pd.DataFrame(data=test_array)
```

Entrée [5]:

```
df.describe()
```

Out[5]:

	0
count	100.000000
mean	50.677424
std	21.192488
min	4.495004
25%	36.138763
50%	47.762114
75%	66.666058
max	103.841159

Entrée [6]:

```
stats.skew(test_array)
```

Out[6]:

```
0.2419004544946985
```

Entrée [7]:

```
print(f"Nombre d'observations : {test_array.shape[0]}")
print(f"Moyenne de l'échantillon : {test_array.mean():.3f}")
print(f"Moyenne de l'échantillon : {np.mean(test_array):.3f}")
print(f"Ecart-type de l'échantillon : {test_array.std():.10f}")
print(f"Mediane de l'échantillon : {np.median(test_array):.3f}")
print(f"Max de l'échantillon : {np.max(test_array):.3f}")
```

```
Nombre d'observations : 100
Moyenne de l'échantillon : 50.677
Moyenne de l'échantillon : 50.677
Ecart-type de l'échantillon : 21.0862591103
Mediane de l'échantillon : 47.762
Max de l'échantillon : 103.841
```

Exercice 2 - Le football est-il relié à l'économie ? (*)

Trouver le coefficient de correlation entre les valeurs du CAC40 et les prix des actions du Groupe OL

Hint: Il existe une fonction de correlation sur `numpy`

Entrée [8]:

```
cac_values = np.array([5197.79, 5011.98, 5022.38, 4858.97, 4762.78, 4695.44, 4771.39,
       4688.74, 4606.24, 4539.91, 4444.56, 4445.45, 4496.98, 4458.16,
       4498.34, 4277.63, 4273.13, 4344.95, 4472.5 , 4490.22, 4549.64,
       4501.44, 4433.38, 4483.13])

olg_values = np.array([2.34, 2.41, 2.41, 2.32, 2.27, 2.24, 2.16, 2.1 , 2.07, 2.07, 2.08,
       2.1 , 2.1 , 2.08, 2.08, 2.07, 2.06, 2.08, 2.08, 2.1 , 2.12, 2.15,
       2.14, 2.17])
```

Entrée [9]:

```
np.corrcoef(cac_values, olg_values)
```

Out[9]:

```
array([[1.        , 0.87975468],
       [0.87975468, 1.        ]])
```

Entrée [10]:

```
print(f"La corrélation entre le CAC et OLG est de {np.corrcoef(cac_values, olg_values)[0
```

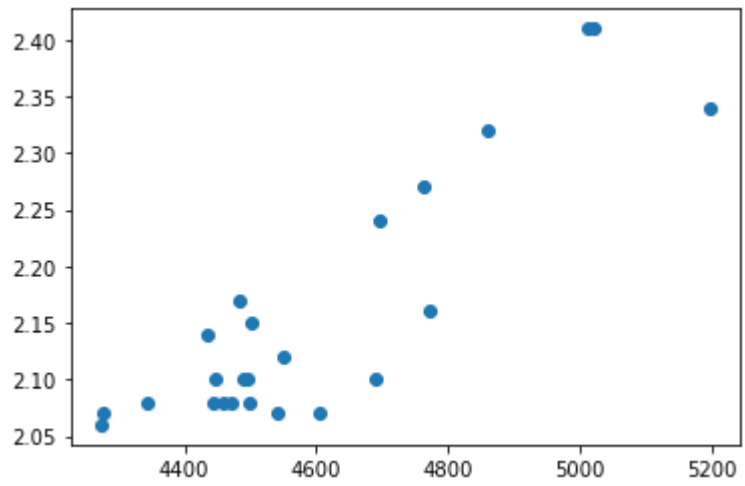
```
La corrélation entre le CAC et OLG est de 88.0%
```

Entrée [11]:

```
import matplotlib.pyplot as plt

plt.scatter(cac_values,olg_values)
```

Out[11]:

```
<matplotlib.collections.PathCollection at 0x7f88b0ff3fd0>
```



Entrée [64]:

```
df_airbnb_filtered['has_reviews'] = ~df_airbnb_filtered['last_review'].isna()
```

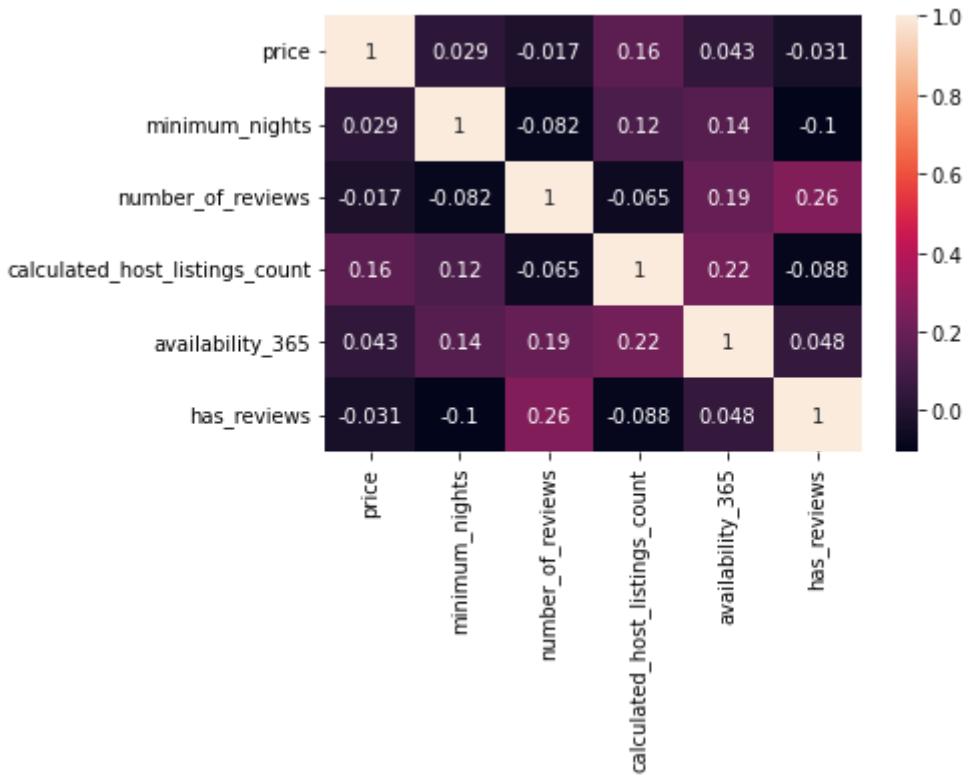
```
C:\Users\antoi\AppData\Local\Temp/ipykernel_22804/3179642567.py:1: Setting
WithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-doc
s/stable/user_guide/indexing.html#returning-a-view-versus-a-copy (https://
pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-
view-versus-a-copy)
  df_airbnb_filtered['has_reviews'] = ~df_airbnb_filtered['last_review'].i
sna()
```

Entrée [65]:

```
cols_to_remove = ['id', 'host_id', 'latitude', 'longitude', 'reviews_per_month']
df_corr = df_airbnb_filtered.drop(columns=cols_to_remove)
corr_matrix = df_corr.corr()
sns.heatmap(data=corr_matrix, annot=True)
```

Out[65]:

```
<AxesSubplot:>
```

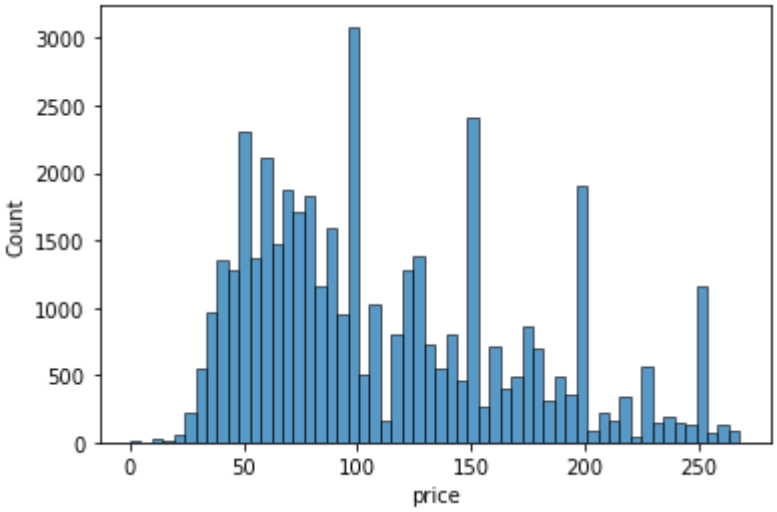


Entrée [66]:

```
sns.histplot(data=df_airbnb_filtered, x='price')
```

Out[66]:

```
<AxesSubplot:xlabel='price', ylabel='Count'>
```



BONUS: Tests d'hypothèse

Exercise 6 – La taille des cerveaux? (***)

Grâce au dataset `brain_size.csv` , répondre à la question: est-ce que la moyenne de la "VIQ size" est significativement différente entre les homme et les femmes?

Hint: pour selectionner des données par genre il faut slicer la dataframe grâce à une condition (`pandas`).

Source: [https://scipy-lectures.org/packages/statistics/index.html (https://scipy-lectures.org/packages/statistics/index.html)](https://scipy-lectures.org/packages/statistics/index.html)

Entrée [67]:

```
import pandas as pd
data = pd.read_csv('brain_size.csv', sep=';', index_col=0)
data
```

Out[67]:

	Gender	FSIQ	VIQ	PIQ	Weight	Height	MRI_Count
1	Female	133	132	124	118	64.5	816932
2	Male	140	150	124	.	72.5	1001121
3	Male	139	123	150	143	73.3	1038437
4	Male	133	129	128	172	68.8	965353
5	Female	137	132	134	147	65.0	951545
6	Female	99	90	110	146	69.0	928799
7	Female	138	136	131	138	64.5	991305
8	Female	92	90	98	175	66.0	854258
9	Male	89	93	84	134	66.3	904858
10	Male	133	114	147	172	68.8	955466
11	Female	132	129	124	118	64.5	833868

Entrée [68]:

```
data['Gender'] == 'Male'
```

Out[68]:

```
1     False
2      True
3      True
4      True
5     False
6     False
7     False
8     False
9      True
10     True
11    False
12     True
13     True
14    False
15    False
16    False
17    False
18     True
19    False
20     True
21     True
22     True
23    False
24     True
25    False
26     True
27    False
28     True
29    False
30    False
31    False
32     True
33     True
34     True
35    False
36    False
37     True
38    False
39     True
40     True
Name: Gender, dtype: bool
```

Entrée [69]:

```
data[ data['Gender'] == 'Male' ]['VIQ']
```

Out[69]:

```
2     150
3     123
4     129
9      93
10    114
12    150
13    129
18     96
20     77
21     83
22    107
24    145
26    145
28     96
32    145
33     96
34     96
37    150
39     90
40     91
Name: VIQ, dtype: int64
```

Entrée [70]:

```
female_viq = data[data['Gender'] == 'Female']['VIQ']
male_viq = data[data['Gender'] == 'Male']['VIQ']
```

Entrée [71]:

```
from scipy import stats

my_test = stats.ttest_ind(female_viq, male_viq)
print(my_test.pvalue)
```

```
0.44452876778583217
```

Entrée [72]:

```
data.shape
```

Out[72]:

```
(40, 7)
```