RT B
SG01

09.10

TP 01 /Initiation et prise en main du langage Python (NumPy, matplotlib, Pandas, fichiers CSV, etc...)

KESSAB IKRAM

PROGRAMMATION ORIENTÉE OBJETS

Objectif du TP

Ce TP vise à nous familiariser avec le langage de programmation Python. Pendant cette session, nous couvrirons l'installation de Python, l'intégration de bibliothèques, l'importation de modules, ainsi que la manipulation de fichiers.

Manipulation

1°

```
[1]: import pandas as pd
     df= pd.read_csv(r"C:/Users/DELL/Downloads/titanic-passengers.csv", sep=";")
     df
```

[1]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	343	No	2	Collander, Mr. Erik Gustaf	male	28.0	0	0	248740	13.0000	NaN	S
1	76	No	3	Moen, Mr. Sigurd Hansen	male	25.0	0	0	348123	7.6500	F G73	S
2	641	No	3	Jensen, Mr. Hans Peder	male	20.0	0	0	350050	7.8542	NaN	S
3	568	No	3	Palsson, Mrs. Nils (Alma Cornelia Berglund)	female	29.0	0	4	349909	21.0750	NaN	S
4	672	No	1	Davidson, Mr. Thornton	male	31.0	1	0	F.C. 12750	52.0000	B71	S
...	...	...	...	...	...	...	...	...	...	...	...	...
886	10	Yes	2	Nasser, Mrs. Nicholas (Adele Achem)	female	14.0	1	0	237736	30.0708	NaN	C
887	61	No	3	Sirayanian, Mr. Orsen	male	22.0	0	0	2669	7.2292	NaN	C
888	535	No	3	Cacic, Miss. Marija	female	30.0	0	0	315084	8.6625	NaN	S
889	102	No	3	Petroff, Mr. Pastcho ("Pentcho")	male	NaN	0	0	349215	7.8958	NaN	S
890	428	Yes	2	Phillips, Miss. Kate Florence ("Mrs Kate Louis...	female	19.0	0	0	250655	26.0000	NaN	S

891 rows × 12 columns

Import pandas as pd : Ce premier morceau de code effectue l'importation d'une bibliothèque Python appelée "pandas". Pandas est couramment employé pour effectuer des opérations de manipulation et d'analyse de données

df = pd.read_csv('votre_fichier.csv') : Cette instruction lit un fichier de données au format CSV en utilisant la fonction read_csv fournie par pandas et le stocke dans une structure de données appelée “DataFrame” (abrégé en “df”).

2°

```
[2]: df.isnull()
```

[2]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	False	False	False	False	False	False	False	False	False	False	True	False
1	False	False	False	False	False	False	False	False	False	False	False	False
2	False	False	False	False	False	False	False	False	False	False	True	False
3	False	False	False	False	False	False	False	False	False	False	True	False
4	False	False	False	False	False	False	False	False	False	False	False	False
...	...	...	...	...	...	...	...	...	...	...	...	...
886	False	False	False	False	False	False	False	False	False	False	True	False
887	False	False	False	False	False	False	False	False	False	False	True	False
888	False	False	False	False	False	False	False	False	False	False	True	False
889	False	False	False	False	False	True	False	False	False	False	True	False
890	False	False	False	False	False	False	False	False	False	False	True	False

891 rows × 12 columns

df.isnull() : Cela permet de repérer les valeurs manquantes (ou nulles) dans un nouveau DataFrame créé, où chaque cellule est définie comme True si la valeur correspondante dans le DataFrame "df" est manquante, et False dans le cas contraire.

3°

```
[5]: number_of_elements= len(df['Cabin'])
     print('Number of elements:',number_of_elements)

     Number of elements: 891
```

number_of_elements = len(df["Cabin"]) : Cette instruction de code détermine la taille (c'est-à-dire le nombre d'éléments) de la colonne "Cabin" dans le DataFrame "df" en utilisant la fonction len(). Le résultat est ensuite stocké dans la variable number_of_elements.

4°

```
[9]: print(df['Cabin'].head())
     print(df['Cabin'].head().isnull())

     0      NaN
     1    F G73
     2      NaN
     3      NaN
     4      B71
     Name: Cabin, dtype: object
     0     True
     1    False
     2     True
     3     True
     4    False
     Name: Cabin, dtype: bool
```

print(df['Cabin'].head()) : Cette ligne affiche les cinq premières lignes de la colonne “Cabin” de votre DataFrame df à l’aide de la méthode head (). Cela vous montrera les premières cinq valeurs de cette colonne.

print(df['Cabin'].head().isnull()) : méthode isnull() est utilisée pour vérifier si chaque valeur dans la colonne “Cabin” est nulle (c’est-à-dire s’il s’agit de NaN ou de données manquantes).

5°

```
[10]: print(df['Cabin'].value_counts())

      G6             4
      B96 B98        4
      C23 C25 C27    4
      F33            3
      D              3
                    ..
      C91            1
      D45            1
      F G63          1
      A34            1
      E63            1
      Name: Cabin, Length: 147, dtype: int64
```

value_counts() : Cette méthode compte combien de fois chaque valeur unique apparaît dans la colonne "Cabin" et renvoie les résultats sous forme d'une série, une structure de données semblable à une liste. Les valeurs uniques deviennent les index de la série, tandis que le nombre d'occurrences correspond aux valeurs."

6°

```
[13]: df.drop('Ticket',axis=1)
```

[13]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Fare	Cabin	Embarked
0	343	No	2	Collander, Mr. Erik Gustaf	male	28.0	0	0	13.0000	NaN	S
1	76	No	3	Moen, Mr. Sigurd Hansen	male	25.0	0	0	7.6500	F G73	S
2	641	No	3	Jensen, Mr. Hans Peder	male	20.0	0	0	7.8542	NaN	S
3	568	No	3	Palsson, Mrs. Nils (Alma Cornelia Berglund)	female	29.0	0	4	21.0750	NaN	S
4	672	No	1	Davidson, Mr. Thornton	male	31.0	1	0	52.0000	B71	S
...	...	...	...	...	...	...	...	...	...	...	...
886	10	Yes	2	Nasser, Mrs. Nicholas (Adele Achem)	female	14.0	1	0	30.0708	NaN	C
887	61	No	3	Sirayanian, Mr. Orsen	male	22.0	0	0	7.2292	NaN	C
888	535	No	3	Cacic, Miss. Marija	female	30.0	0	0	8.6625	NaN	S
889	102	No	3	Petroff, Mr. Pastcho ("Pentcho")	male	NaN	0	0	7.8958	NaN	S
890	428	Yes	2	Phillips, Miss. Kate Florence ("Mrs Kate Louis...	female	19.0	0	0	26.0000	NaN	S

891 rows × 11 columns

df.drop('Ticket', axis=1) : Suppression de la colonne “Ticket” de votre DataFrame df.

7°

```
[18]: df['Cabin'].fillna('G6',inplace=True)
      df.head()
```

[18]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	343	No	2	Collander, Mr. Erik Gustaf	male	28.0	0	0	248740	13.0000	G6	S
1	76	No	3	Moen, Mr. Sigurd Hansen	male	25.0	0	0	348123	7.6500	F G73	S
2	641	No	3	Jensen, Mr. Hans Peder	male	20.0	0	0	350050	7.8542	G6	S
3	568	No	3	Palsson, Mrs. Nils (Alma Cornelia Berglund)	female	29.0	0	4	349909	21.0750	G6	S
4	672	No	1	Davidson, Mr. Thornton	male	31.0	1	0	F.C. 12750	52.0000	B71	S

df['Cabin'].fillna('G6', inplace=True) : Cette instruction remplace les valeurs manquantes (c'est-à-dire les NaN) dans la colonne "Cabin" par la valeur 'G6'. L'argument "inplace=True" modifie directement le DataFrame "df" en place.

8°

```
[19]: df['Age'].fillna(df['Age'].mean(), inplace=True)
      df.tail()
```

[19]:

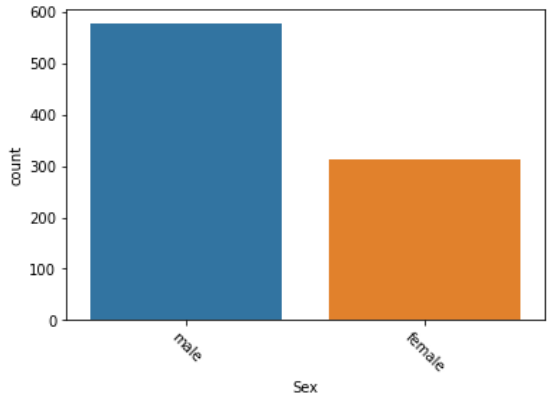
	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
886	10	Yes	2	Nasser, Mrs. Nicholas (Adele Achem)	female	14.000000	1	0	237736	30.0708	G6	C
887	61	No	3	Sirayanian, Mr. Orsen	male	22.000000	0	0	2669	7.2292	G6	C
888	535	No	3	Cacic, Miss. Marija	female	30.000000	0	0	315084	8.6625	G6	S
889	102	No	3	Petroff, Mr. Pastcho ("Pentcho")	male	29.699118	0	0	349215	7.8958	G6	S
890	428	Yes	2	Phillips, Miss. Kate Florence ("Mrs Kate Louis...	female	19.000000	0	0	250655	26.0000	G6	S

df['Age'].fillna(df['Age'].mean(), inplace=True) : "Cette instruction remplace les valeurs manquantes (NaN) dans la colonne "Age" en utilisant la moyenne des valeurs présentes dans cette même colonne qui ne sont pas manquantes."

9°

```
[41]: import matplotlib as plt
      import pandas as pd
      import seaborn as sns
      df= pd.read_csv(r"C:/Users/DELL/Downloads/titanic-passengers.csv", delimiter=";")
      sns.countplot(x='Sex', data=df)
      plt.pyplot.xticks(rotation=-45)
```

```
[41]: (array([0, 1]), [Text(0, 0, 'male'), Text(1, 0, 'female')])
```



import seaborn as sns : Importer la bibliothèque Seaborn sous l'alias sns, qui est couramment utilisée pour créer des graphiques statistiques.

sns.countplot(x='Sex', data=df) : créer un graphique à barres avec Seaborn en spécifiant que vous voulez afficher la répartition par sexe.

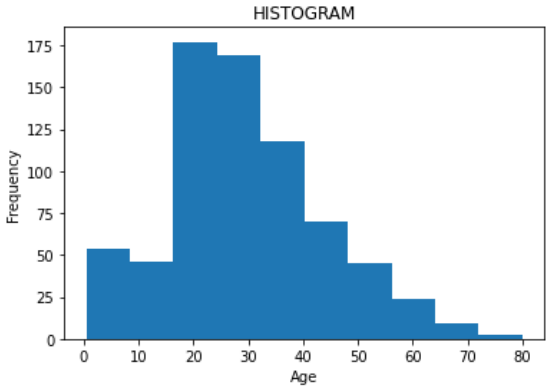
import matplotlib.pyplot as plt : Importer la bibliothèque Matplotlib sous l'alias plt, ce qui vous permettra de personnaliser et d'afficher le graphique.

df = pd.read_csv("titanic-passengers.csv", delimiter=';') : Lire les données du fichier CSV "titanic-passengers.csv" en utilisant pandas, en spécifiant que le délimiteur est un point-virgule (;)

10°

```
[48]: plt.pyplot.title('HISTOGRAM')
      plt.pyplot.xlabel('Age')
      df['Age'].plot.hist()
```

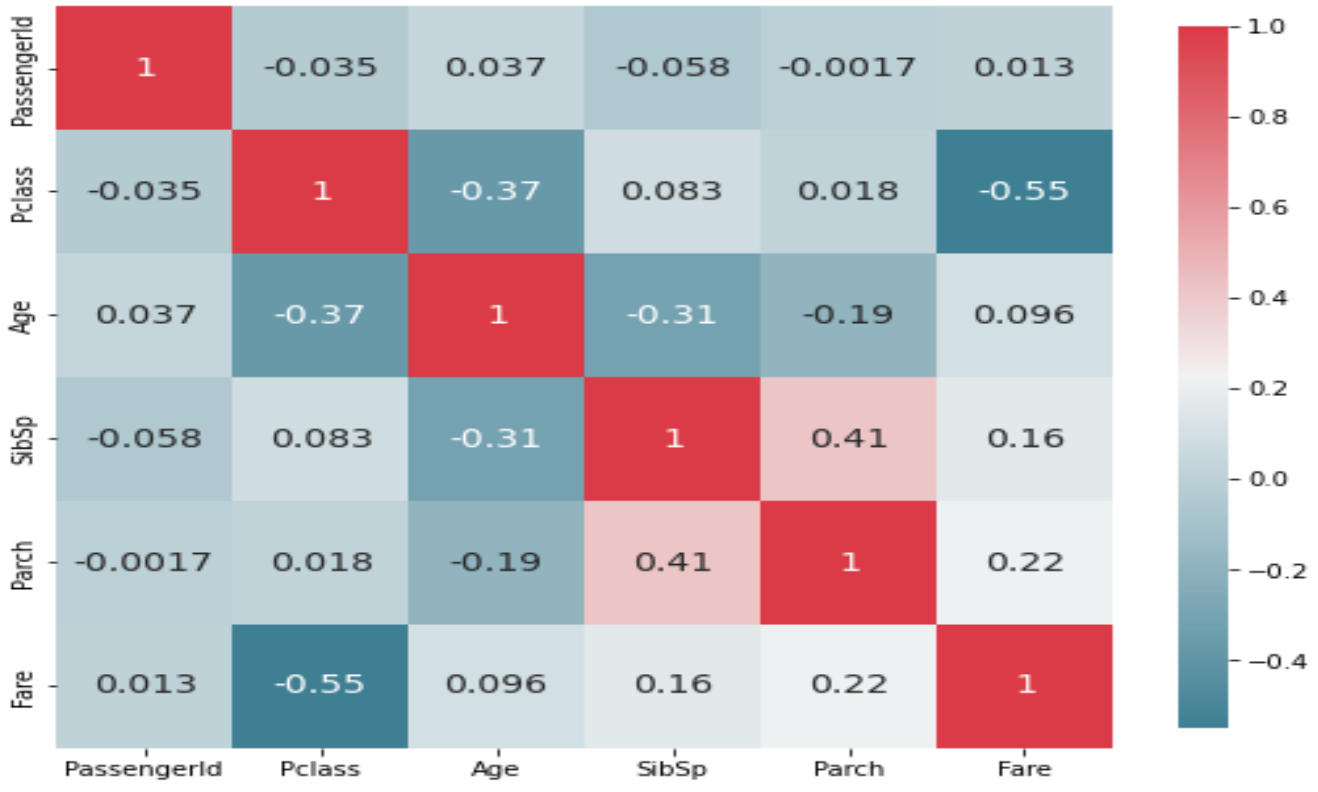
```
[48]: <AxesSubplot:title={'center':'HISTOGRAM'}, xlabel='Age', ylabel='Frequency'>
```



df['Age'].plot.hist() : Cela tracera un histogramme des valeurs de la colonne “Age”.

11°

```
[23]: import matplotlib.pyplot as plt
      import seaborn as sns
      def plot_correlation_map(df):
          numeric_columns=df.select_dtypes(include=['number'])
          corr =numeric_columns.corr()
          plt.figure(figsize=(9,10))
          cmap=sns.diverging_palette(220, 10, as_cmap=True)
          sns.heatmap(corr, cmap=cmap, square=True, cbar_kws={'shrink':0.7}, annot=True, annot_kws={'fontsize':14}  )
          plt.show()
      plot_correlation_map(df)
```



plot_correlation_map : La principale utilité de cette fonction est de fournir une représentation graphique de la structure de corrélation au sein d'un ensemble de données. Elle aide les analystes de données et les scientifiques à identifier et à interpréter rapidement les relations entre les variables numériques. Ces informations sont essentielles pour diverses tâches liées aux données, telles que la sélection des caractéristiques, la détection de la multicollinéarité et l'obtention d'une compréhension plus approfondie de l'ensemble de données.

Conclusion

Dans ce travail pratique, nous avons appliqué les compétences que nous avons acquises à partir de divers modules, notamment NumPy, matplotlib, Pandas, et la manipulation de fichiers CSV. Notre mission consistait principalement à effectuer le prétraitement des données et à créer des visualisations.

Nous avons initié le processus en utilisant Pandas pour importer le jeu de données, examiner les premières lignes pour comprendre sa structure, et obtenir une vue d'ensemble de ses colonnes et de leurs valeurs. En approfondissant le prétraitement des données, nous avons identifié les données manquantes et les avons remplacées par des valeurs appropriées afin de préparer les données pour une analyse ultérieure.

Dans la phase suivante, nous avons créé une visualisation pour explorer la corrélation entre le sexe et l'âge. Cette représentation graphique visait à offrir des perspectives claires sur l'importance de l'âge et du sexe en relation avec la survie des individus.

Un moment significatif de notre travail a été l'utilisation d'une fonction nommée "plot_correlation_map". Cette fonction a été employée pour calculer et présenter graphiquement la corrélation entre les différentes variables numériques de l'ensemble de données. Grâce à cette fonction, nous avons pu déceler la force et la direction des corrélations entre diverses variables, ce qui nous a permis de découvrir des relations précieuses au sein des données.