LA SOUSCRIPTION DES CONTRATS D'ASSURANCES VIE

Mohamed Mahfoudh Bouna

PLAN

01
L'ENVIRONNEMENT DE TRAVAIL

Dans cette partie, nous choisissons le sujet d'analyse et les datasets, et leur ressources , En plus les étapes de préparation et de chargement des données sélectionnées et les outils de travail et des bibliothèques...

02
ANALYSE DE DONNÈES

Dans cette partie, nous analyserons les données pour savoir ce qu'elles contiennent des statistiques importantes qui peuvent nous aider à extraire des connaissance.

03
EXTRACTION DE CONNAISSANCE

Dans cette partie, nous allons construire un ensemble des modéles de prediction de les variables.

04
DÈPLOIEMENT

La mise en place d'une application permettant l'utilisation du modèle obtenu pour la prédiction des nouvelles observations

01 L'ENVIRONNEMENT DE TRAVAIL

Dans cette partie, nous choisissons le sujet d'analyse et les datasets, et leur ressources ,
En plus les étapes de préparation et de chargement des données sélectionnées et les outils de travail et des bibliothèques...

01

PRÈPARATION DE L'ENVIRONNEMENT DU TRAVAIL

LE SUJET D'ANALYSE

La souscription des contrats d'assurances vie

LES DATASETS

https://www.kaggle.com

LES OUTILS DE TRAVAIL

Sklearn

Est une bibliothèque libre en Python destinée à l'apprentissage automatique.

Jupyter

Pour Faire Le Codage En Python et L'analyse de données.....

Pandas

Est une bibliothèque en Python permettant la manipulation et l'analyse des données.

Matplotlib

Est une bibliothèque en Python destinée à tracer et visualiser des données sous formes de graphiques

NumPy

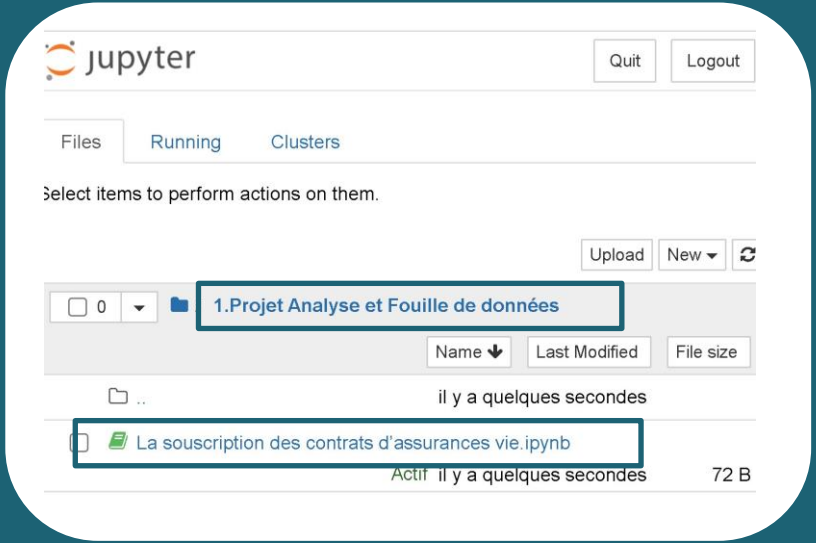
Est une bibliothèque en Python destinée à manipuler des matrices ou tableaux multidimensionnels ainsi que des fonctions mathématiques opérant sur ces tableaux.

01

PRÈPARATION DE L'ENVIRONNEMENT DU TRAVAIL

JUPYTER

1. Dans Jupyter, nous créons un nouveau fichier, à l'intérieur de ce fichier, nous créons un nouveau fichier ipynb



2.L'importation des Bibliothéques

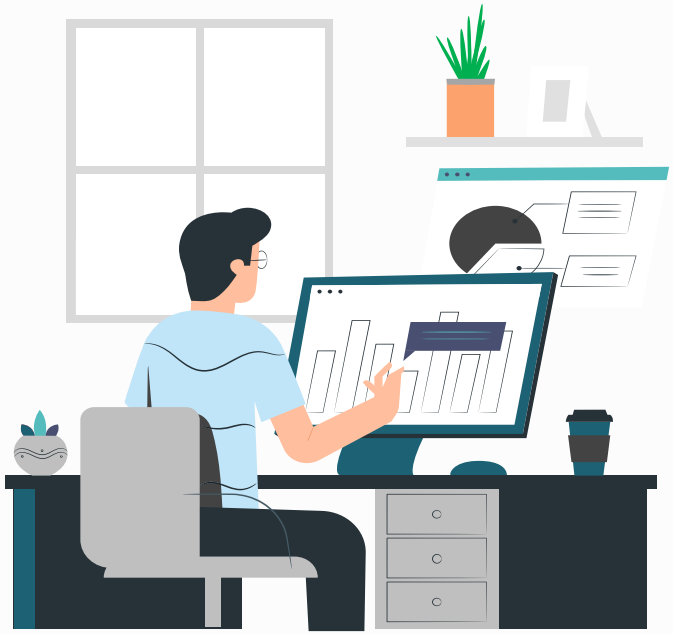
```
Entrée [ ]: #Importation des Bibliotheque
            import numpy as np
            import pandas as pd
            import matplotlib.pyplot as plt
```

3.L'importation des Données

```
Entrée [6]: #L'importation des datasets
            data_set=pd.read_csv("insurance.csv")
```

02 ANALYSE DE DONNÈES

Dans cette partie, nous analyserons les données dont nous disposons pour savoir ce qu'elles contiennent des statistiques importantes qui peuvent nous aider à extraire des connaissance.



02

ANALYSE DE DONNÈES

- head(): Pour Afficher Les 5 premières lignes(par défaut 5)

```
Entrée [7]: data_set.head()
```

Out[7]:

	age	sex	bmi	children	smoker	region	expenses
0	19	female	27.9	0	yes	southwest	16884.92
1	18	male	33.8	1	no	southeast	1725.55
2	28	male	33.0	3	no	southeast	4449.46
3	33	male	22.7	0	no	northwest	21984.47
4	32	male	28.9	0	no	northwest	3866.86

- shape: Pour Afficher Le Volume de données (lignes, colonnes)

```
Entrée [8]: data_set.shape
```

Out[8]: (1338, 7)

- describe():Pour afficher toutes les statistiques importantes(le Moyenne, L'ecart Type.....)

```
data_set.describe()
```

	age	bmi	children	expenses
count	1338.000000	1338.000000	1338.000000	1338.000000
mean	39.207025	30.665471	1.094918	13270.422414
std	14.049960	6.098382	1.205493	12110.011240
min	18.000000	16.000000	0.000000	1121.870000
25%	27.000000	26.300000	0.000000	4740.287500
50%	39.000000	30.400000	1.000000	9382.030000
75%	51.000000	34.700000	2.000000	16639.915000
max	64.000000	53.100000	5.000000	63770.430000

02

- convertir des données catégorielles en données numériques (pour les fumeurs: oui avec 0, et non avec 1) (pour le sexe: male avec 1,et femelle avec 0)

Entrée [11]:

```
data_set['smoker']=data_set['smoker'].map({'yes':0,'no':1})
data_set['sex']=data_set['sex'].map({'male':1,'female':0})
data_set.head()
```

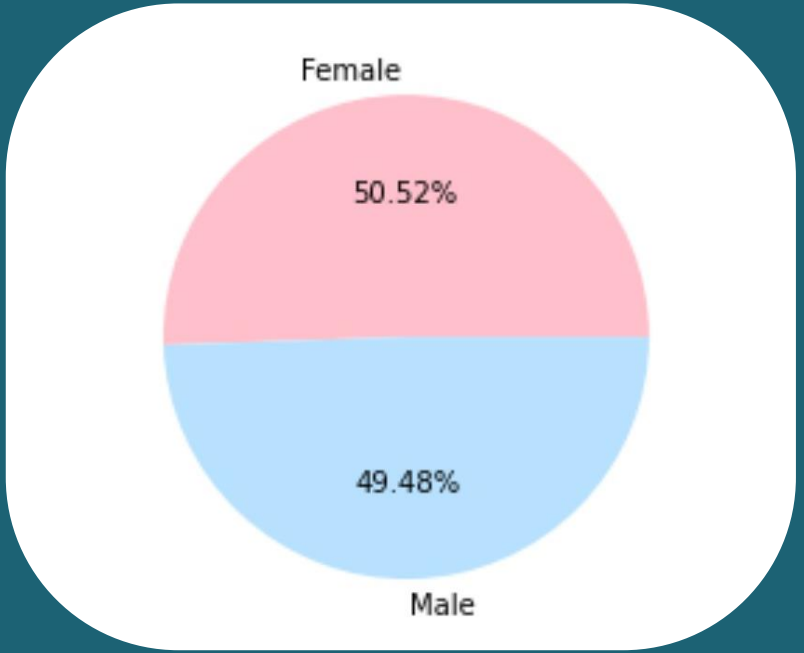
Out[11]:

	age	sex	bmi	children	smoker	region	expenses
0	19	1	27.9	0	0	southwest	16884.92
1	18	0	33.8	1	1	southeast	1725.55
2	28	0	33.0	3	1	southeast	4449.46
3	33	0	22.7	0	1	northwest	21984.47
4	32	0	28.9	0	1	northwest	3866.86

02

- Le pourcentage d'hommes et de femmes

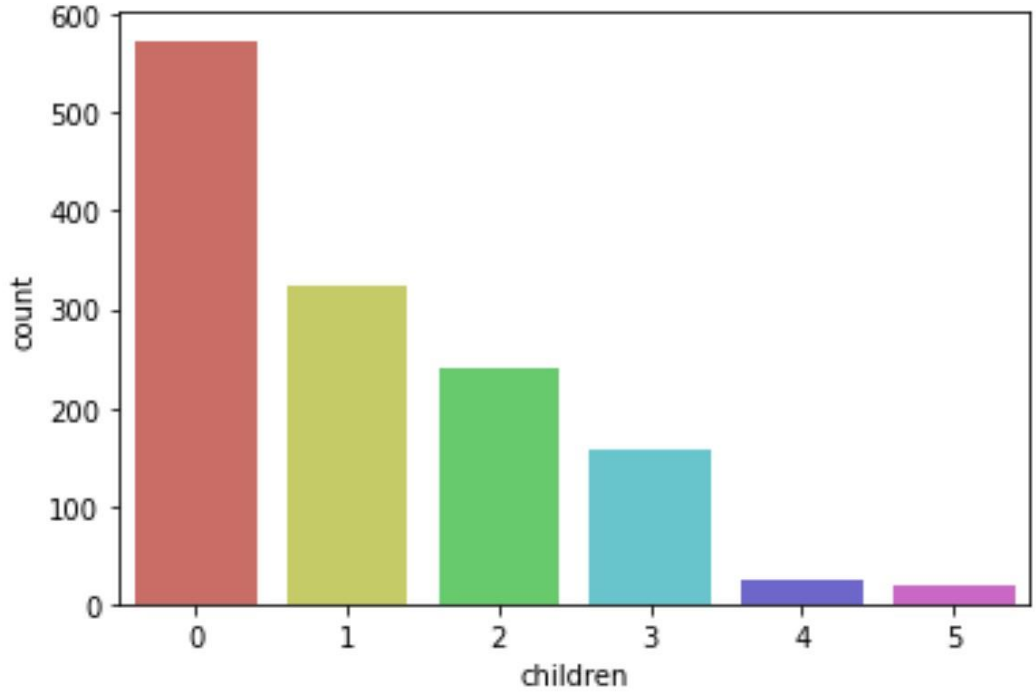
```
plt.pie(x=data_set["sex"].value_counts(),colors=["pink","#B8E1FF"],labels=["Female","Male"],autopct="%1.2f%%")
plt.show()
```



02

- Nombre d'enfants par rapport à l'age

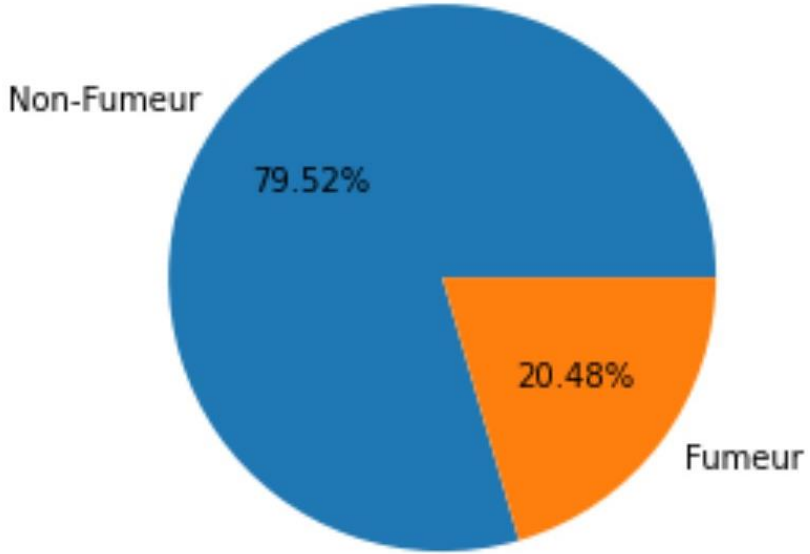
```
Entrée [8]: sb.countplot(data_set["children"], palette="hls")
```



02

- Pourcentage de fumeurs

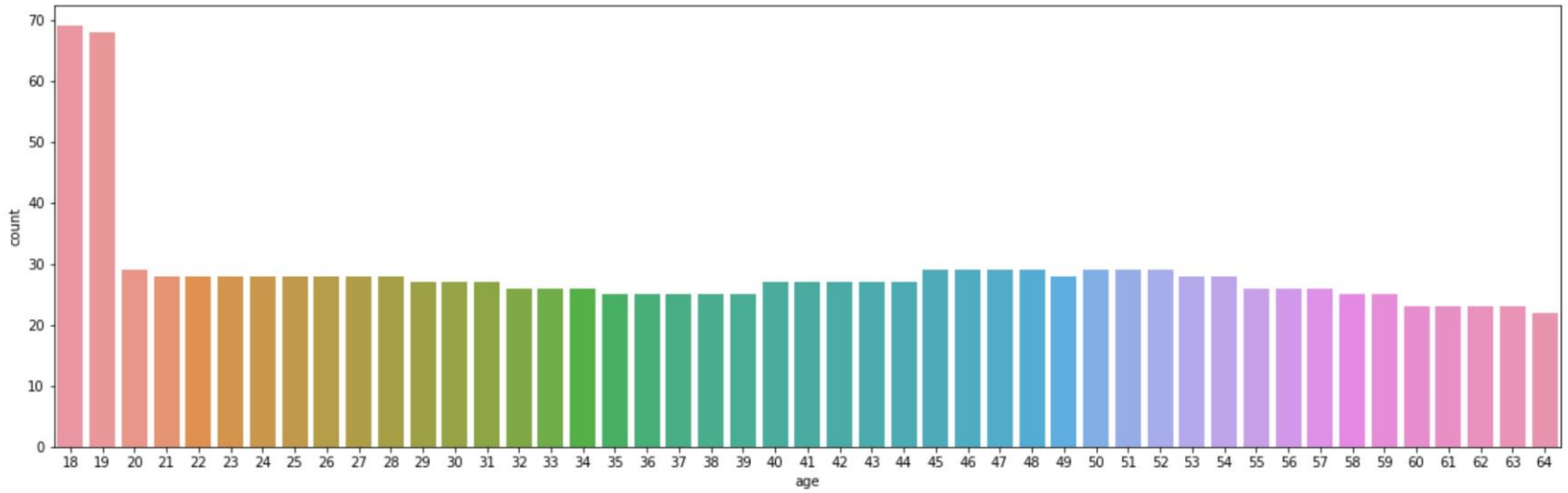
```
plt.pie(x=data_set["smoker"].value_counts(),labels=["Non-Fumeur","Fumeur"],autopct="%1.2f%%")
plt.show()
```



02

- Les Nombres de personnes par rapport à l'age

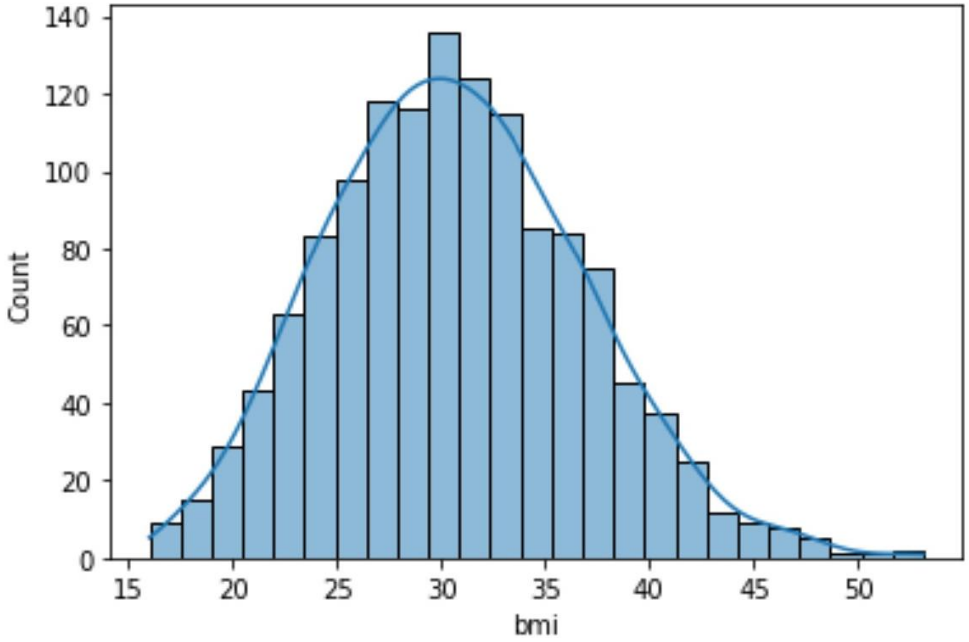
```
Entrée [12]: fig, ax = plt.subplots(figsize=(20,6))
             sb.countplot(data_set["age"])
```



02

- Stastiques de L'indice Masse Corporelle (IMC)

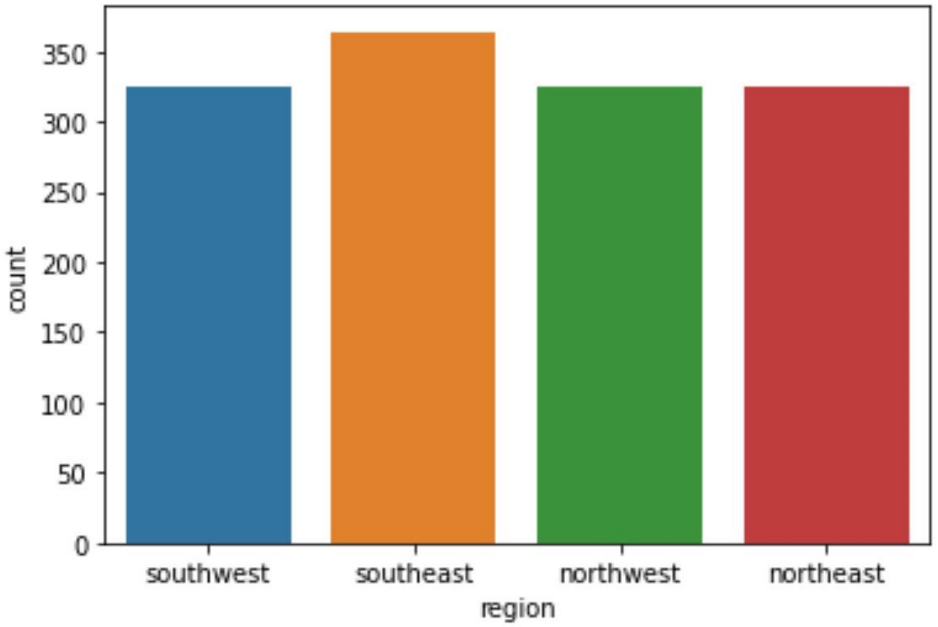
```
sb.histplot(x = data_set["bmi"], kde=True);
```



02

- Stastiques de Personnes Par Région

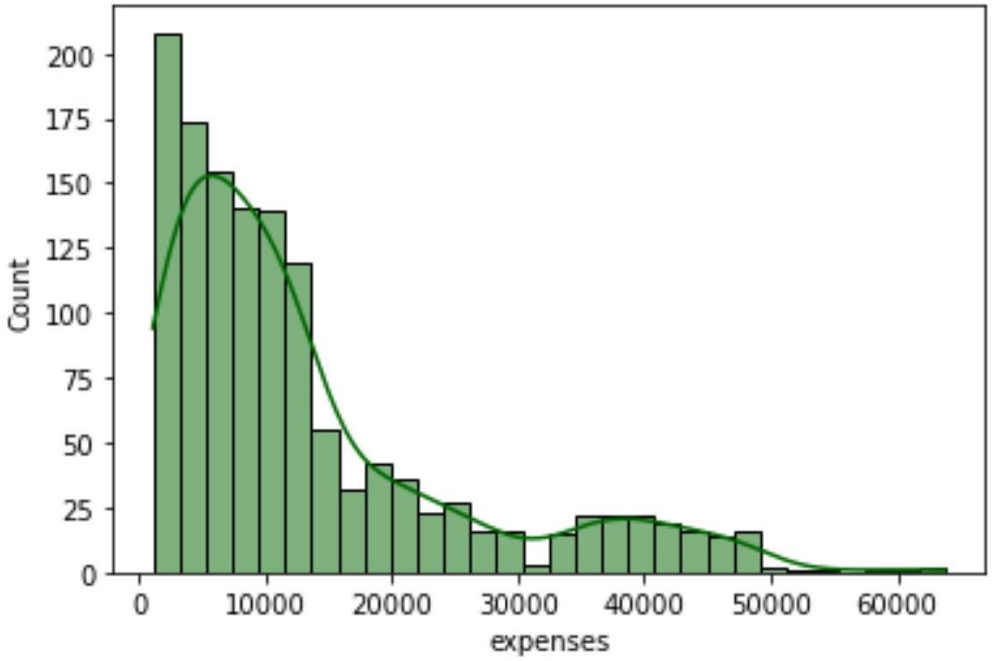
```
Entrée [14]: sb.countplot(data_set["region"])
```



02

- Les Dépenses Par Rapport à L'age

```
Entrée [15]: sb.histplot(x = data_set["expenses"], color="darkgreen", kde=True)
```



03 EXTRACTION DE CONNAISSANCE

Dans cette partie, nous allons construire un ensemble des modéles de prediction de les variables.

02

- Tout d'abord, nous supprimons les colonnes qui ne contiennent pas de données de type numérique(Region).

```
data_set.drop('region', axis=1, inplace=True)
```

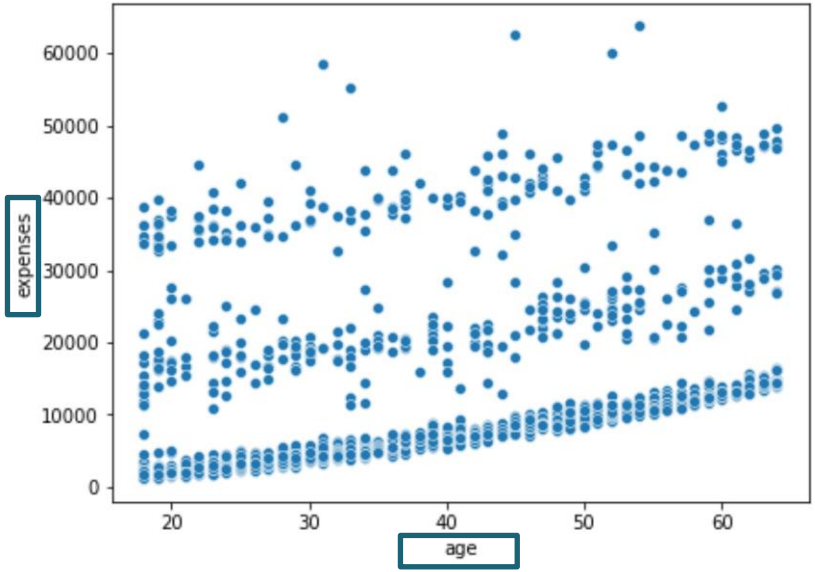
```
data_set.head()
```

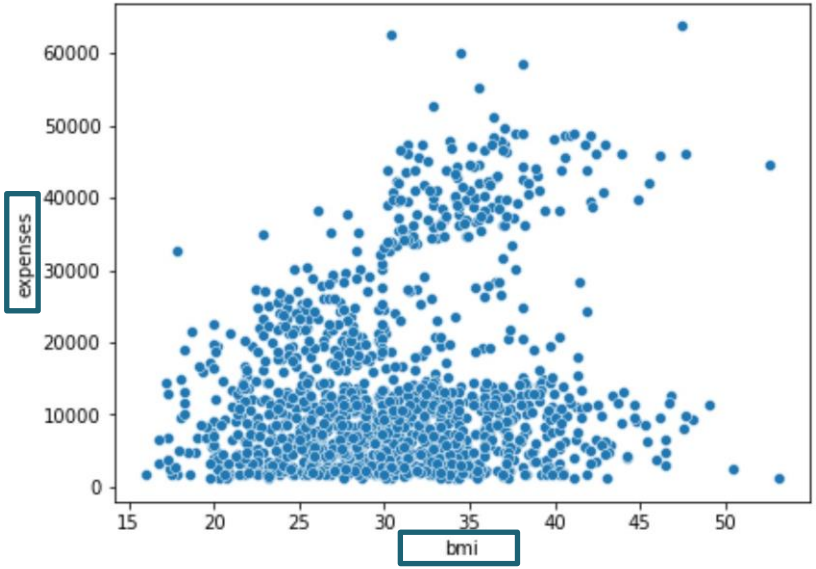
	age	sex	bmi	children	smoker	expenses
0	19	1	27.9	0	0	16884.92
1	18	0	33.8	1	1	1725.55
2	28	0	33.0	3	1	4449.46
3	33	0	22.7	0	1	21984.47
4	32	0	28.9	0	1	3866.86

02

- Le Nuage des points (Age-dépenses et BMI – dépenses)

```
Entrée [20]: fig, ax =plt.subplots(1,2, figsize=(15,5))
             sb.scatterplot(x = data_set['age'], y = data_set['expenses'],ax=ax[0])
             sb.scatterplot(x = data_set['bmi'], y = data_set['expenses'],ax=ax[1])
```





02

- Division de données (x= age-sex-bmi..... et y = expenses)

```
x= data_set.drop("expenses",axis=1)
x.head()
```

```
y=data_set['expenses']
y.head()
```

	age	sex	bmi	children	smoker
0	19	1	27.9	0	0
1	18	0	33.8	1	1
2	28	0	33.0	3	1
3	33	0	22.7	0	1
4	32	0	28.9	0	1

```
0    16884.92
1     1725.55
2     4449.46
3    21984.47
4     3866.86
```

02

- construire le modèle de prédiction.
- l'importation de sklearn…….
- Faire Le Test.

```
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.2,random_state=0)
```

```
x_train
```

	age	sex	bmi	children	smoker
621	37	0	34.1	4	0
194	18	0	34.4	0	1
240	23	1	36.7	2	0
1168	32	0	35.2	2	1
1192	58	1	32.4	1	1
...	...	...	...	...	...
763	27	0	26.0	0	1
835	42	0	36.0	2	1
1216	40	0	25.1	0	1
559	19	0	35.5	0	1
684	33	1	18.5	1	1

```
y_train
```

```
621      40182.25
194       1137.47
240      38511.63
1168      4670.64
1192     13019.16
           ...
763       3070.81
835       7160.33
1216      5415.66
559       1646.43
684       4766.02
```

04 DÈPLOIEMENT

La mise en place d'une application permettant l'utilisation du modèle obtenu pour la prédiction des nouvelles observations

02

- Evaluer les performances d'un classiffieur (Par le fonction TreeRegressor() Dans sklearn.
La fonction fit() qui apprend un modèle à partir des données

```
from sklearn.tree import DecisionTreeRegressor
tree = DecisionTreeRegressor()
tree.fit(x_train,y_train)
```

```
DecisionTreeRegressorScore = tree.score(x_test, y_test)
DecisionTreeRegressorScore
```

```
0.727546618432371
```

MERCI!