Charger les bibliothèques

In [1]:
```python
import pandas as pd
import numpy as np
import seaborn as sns
from sklearn import preprocessing
from pandas import read_csv
from statsmodels.formula.api import ols
import statsmodels.api as sm
import matplotlib.pyplot as plt
from scipy import stats
```

Partie I : Questions de compréhension

1- Familles de description des variables quantitatives:

- Caractéristiques de tendances centrales
- Caractéristiques de dispersion
- Caractéristiques de formes

2- Le coefficient d''asymétrie est < 0 $\implies$ la distribution de la variable est décalée vers la droite et la moyenne est inférieure à la médiane

3- Modalités | n_jl |f_j

- Pomme | 30 | 0.3
- Banane | 20 | 0.2
- Orange | 25| 0.25
- Raison | 10| 0.10
- Fraise | 15| 0.15
- Total | 100| 1

4- Il n'est pas possible de calculer l'effectif cumulé puisque la variable n'est pas ordinale

5- Il s'agit d'un diagramme bidimensionnelle conçue pour visualiser et comparer les paramètres Ss, Sp et F1-score pour 5 classificateurs.

- Les classificateurs ont des performances variables en termes Ss, Sp et F1-score
- Le K-plus proche voisin performe le mieux en terme de F-score
- La machine à support vectoriel est la meilleure en terme de spécificité.

Partie II: Analyse de données

Étape 1- Lecture de données et analyse préliminaire

1 - Téléchargement des données

In [2]:
```python
df = pd.read_excel('USA_cars_dataset.xlsx')
display(df)
```

	Unnamed: 0	price	brand	model	year	title_status	mileage	color	vin	lot	state	country	condition
0	0	6300	toyota	cruiser	2008	clean vehicle	274117.0	black	jtezu11f88k007763	159348797	new jersey	usa	10 days left
1	1	2899	ford	se	2011	clean vehicle	190552.0	silver	2fmdk3gc4bbb02217	166951262	tennessee	usa	6 days left
2	2	5350	dodge	mpv	2018	clean vehicle	39590.0	silver	3c4pdcgg5jt346413	167655728	georgia	usa	2 days left
3	3	25000	ford	door	2014	clean vehicle	64146.0	blue	1ftfw1et4efc23745	167753855	virginia	usa	22 hours left
4	4	27700	chevrolet	1500	2018	clean vehicle	6654.0	red	3gcpcrec2jg473991	167763266	florida	usa	22 hours left
...	...	...	...	...	...	...	...	...	...	...	...	...	...
2494	2494	7800	nissan	versa	2019	clean vehicle	23609.0	red	3n1cn7ap9kl880319	167722715	california	usa	1 days left
2495	2495	9200	nissan	versa	2018	clean vehicle	34553.0	silver	3n1cn7ap5jl884088	167762225	florida	usa	21 hours left
2496	2496	9200	nissan	versa	2018	clean vehicle	31594.0	silver	3n1cn7ap9jl884191	167762226	florida	usa	21 hours left
2497	2497	9200	nissan	versa	2018	clean vehicle	32557.0	black	3n1cn7ap3jl883263	167762227	florida	usa	2 days left
2498	2498	9200	nissan	versa	2018	clean vehicle	31371.0	silver	3n1cn7ap4jl884311	167762228	florida	usa	21 hours left

2499 rows × 13 columns

In [3]:
```python
dimension1 = df.shape
print("USA_cars_dataset.xlsx:",dimension1)
```

```
USA_cars_dataset.xlsx: (2499, 13)
```

2- Identification des variables et de leurs types

In [4]:
```python
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2499 entries, 0 to 2498
Data columns (total 13 columns):
 #   Column        Non-Null Count  Dtype  
---  ------        --------------  -----  
 0   Unnamed: 0    2499 non-null   int64  
 1   price         2499 non-null   int64  
 2   brand         2499 non-null   object 
 3   model         2499 non-null   object 
 4   year          2499 non-null   int64  
 5   title_status  2499 non-null   object 
 6   mileage       2499 non-null   float64
 7   color         2499 non-null   object 
 8   vin           2499 non-null   object 
 9   lot           2499 non-null   int64  
 10  state         2499 non-null   object 
 11  country       2499 non-null   object 
 12  condition     2499 non-null   object 
dtypes: float64(1), int64(4), object(8)
memory usage: 253.9+ KB
```

La base de données contient 10 variables

- Variables quantitatives: price, mileage, lot
- Variables qualitatives : brand, model, title_status
- Variable quantitative ou qualitative ordinale: year

3- Détermination de nombre de voiture vendues annuellement

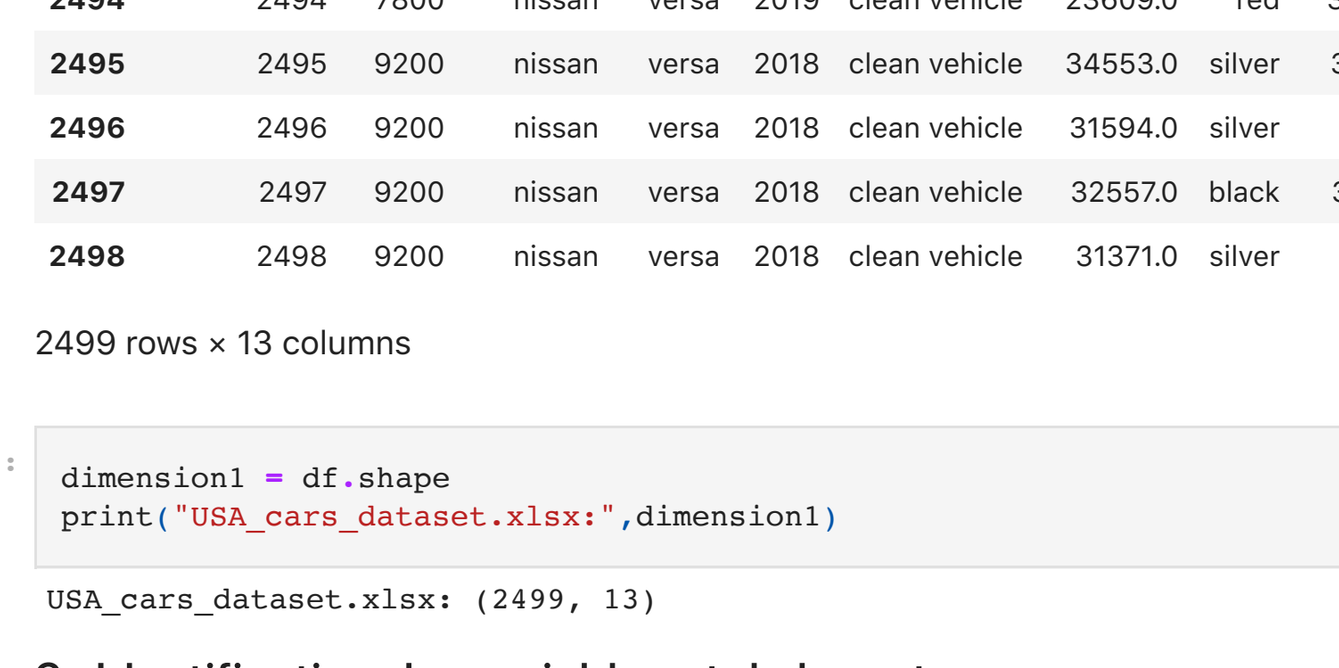
In [7]:
```python
E1 = df.year.value_counts().sort_index()
display(E1)
```

```
1973      1
1984      1
1993      1
1994      2
1995      1
1996      2
1997      2
1998      4
1999      1
2000      4
2001      5
2002      2
2003      9
2004      6
2005      6
2006      8
2007      6
2008     18
2009     11
2010     13
2011     23
2012     72
2013     86
2014    104
2015    196
2016    203
2017    377
2018    395
2019    892
2020     48
Name: year, dtype: int64
```

La première colonne de la liste précédente indique l'année et la deuxième colonne le nombre de voiture vendues. On remarque que ce nombre est relativement faible jusqu'en 2012. Le maximum est atteint en 2019. On peut également obtenir ces informations à partir du diagramme en bar

In [8]:
```python
E1.plot.bar(figsize=(8,4),fontsize=13)
```

Out[8]: <AxesSubplot:>



Étape 2: Analyse des prix des voitures

1- Création de df2010

In [9]:
```python
df2010=df[(df.year>=2010) & (df.year<2020)]
df2010.describe().style.format('{:0.2f}')
```

Out[9]:

	Unnamed: 0	price	year	mileage	lot
count	2361.00	2361.00	2361.00	2361.00	2361.00
mean	1280.61	19269.02	2017.14	48329.43	167699671.55
std	710.45	11812.30	2.11	50816.03	111619.35
min	1.00	0.00	2010.00	0.00	166951262.00
25%	692.00	10840.00	2016.00	22013.00	167629013.00
50%	1290.00	17200.00	2018.00	35065.00	167745308.00
75%	1891.00	25900.00	2019.00	58576.00	167780189.00
max	2498.00	84900.00	2019.00	1017936.00	167805500.00

2 - indicateurs descriptifs de tendances centrales, deux de dispersions et deux de formes de cette variable.

In [25]:
```python
print('La moyenne est',df2010['price'].mean())
print('Le min est',df2010['price'].min())
print('La variation standard est',df2010['price'].std())
print('L'étendue est',df2010['price'].max()-df2010['price'].min())
print('Le coefficient d'' assymétrie', df2010['price'].skew())
print('Le coefficient d'' applatissement', df2010['price'].kurtosis())
```

```
La moyenne est 19269.02414231258
Le min est 0
La variation standard est 11812.29617594079
Létendue est 84900
Le coefficient d assymétrie 0.9733826097690635
Le coefficient d applatissement 1.4458861111065073
```

Tendence centrale

- La moyenne est 19269.02
- Le min est 0

Dispersion

- La varaition standard est 11812.29
- L étendue est 84900

Forme

- Le coefficient d assymétrie 0.97
- Le coefficient d applatissement 1.44

3 -Visualisation graphique

In [11]:
```python
df2010.boxplot(column='price', figsize=(8,4),vert = False, fontsize=15)
None
```

In [12]:
```python
df2010.hist(column='price', bins =20, figsize=(8,4))
plt.xlabel('Year',fontsize=14)
plt.ylabel('count',fontsize=14)
None
```

4- Lien entre les indicateurs numériques et le diagramme

- Le diagramme en boite montre que la valeur min = 0 et que la valeur max est d'environ 85000.
- L'histogramme de la variable price montre que la distribution est décalée vers la gauche. Ceci est confirmé par la valeur du Kurtosis = 0.92>0.

Étape 3: Analyse de vente de voiture

Effectifs de ventes annuelles des voitures

In [13]:
```python
df1 = df2010.groupby(['year','brand']).size().to_frame('count').reset_index()
display(df1)
```

	year	brand	count
0	2010	chevrolet	5
1	2010	ford	5
2	2010	heartland	1
3	2010	honda	1
4	2010	peterbilt	1
...	...	...	...
97	2019	jeep	26
98	2019	kia	1
99	2019	lincoln	1
100	2019	mercedes-benz	1
101	2019	nissan	74

102 rows × 3 columns

Les 6 voitures les plus vendues

In [14]:
```python
df1pt = df2010.brand.value_counts().reset_index().rename(columns={'index':'brand','brand':'count'}).head(6)

display(df1pt)
```

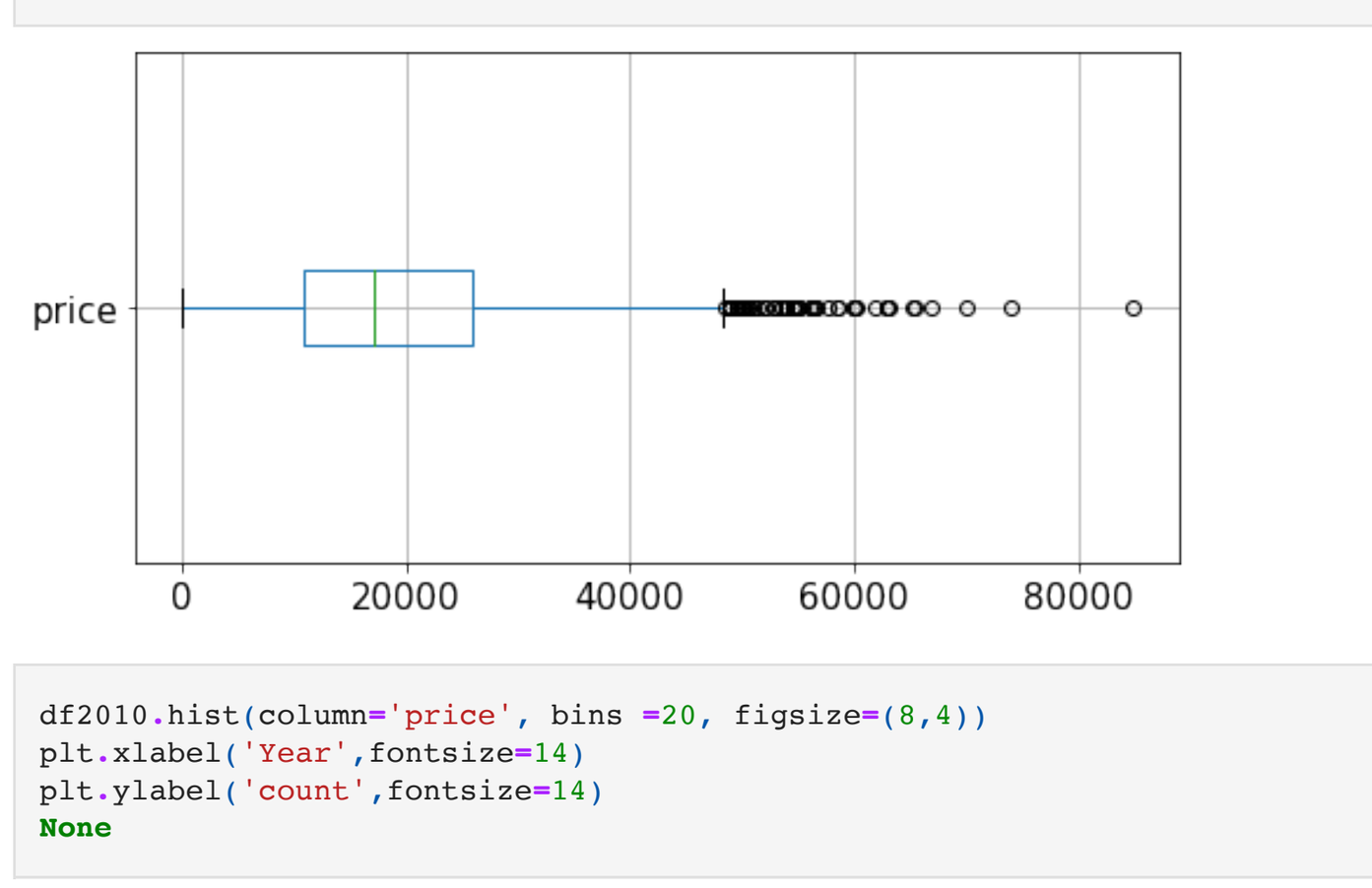
	brand	count
0	ford	1187
1	dodge	423
2	nissan	298
3	chevrolet	260
4	gmc	37
5	jeep	28

- On note que les 6 voitures les plus vendues sont la ford, la dodge, la nissan, la chevrolet, la gmc et la Jeep (avec n=28).

1- Répartition des effectifs de ventes des 6 marques

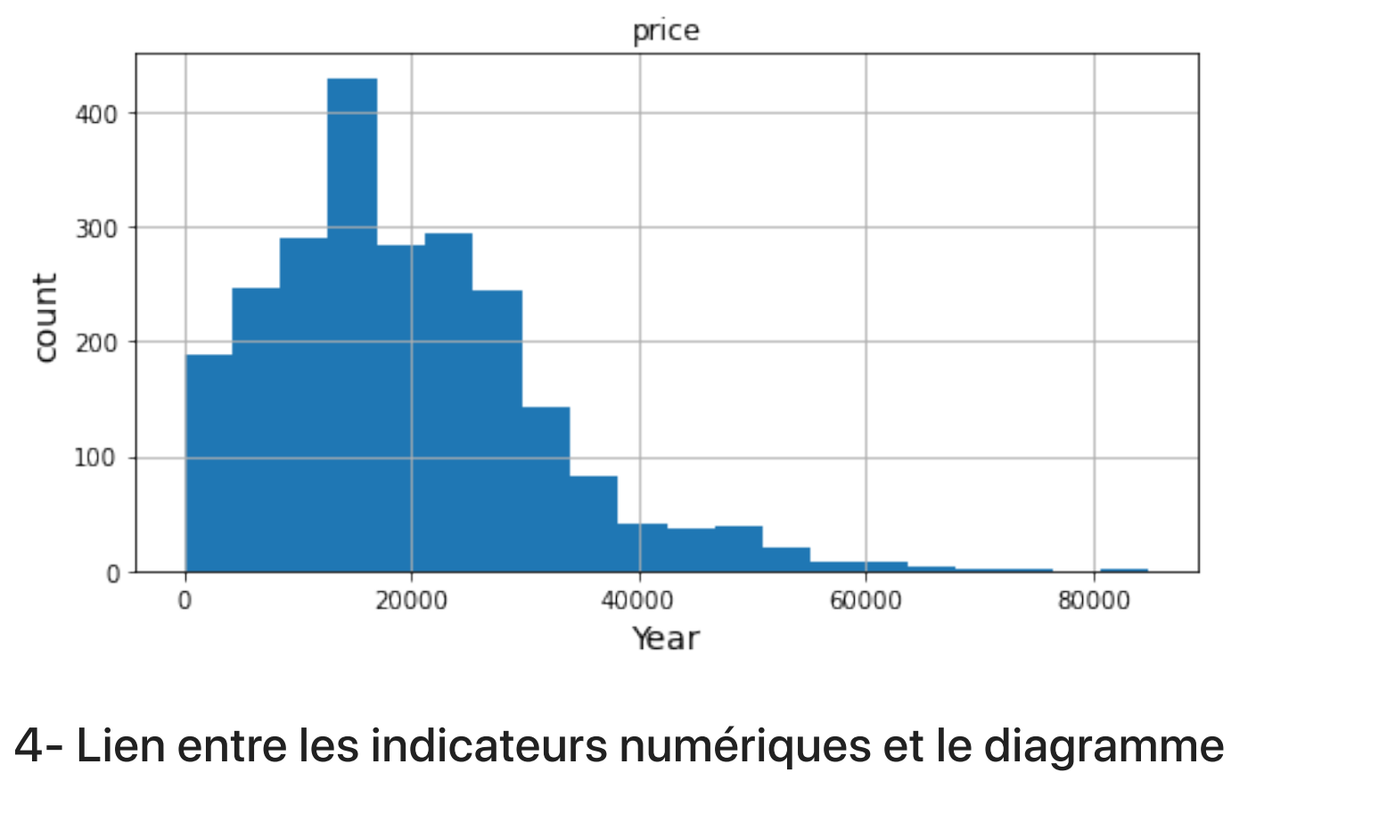
In [15]:
```python
df2010R = df2010[df2010["brand"].str.contains('(ford|dodge|nissan|chevrolet|gmc|jeep)', regex=True)]
df2= df2010R.groupby('year').brand.value_counts().unstack()
plt.rcParams["figure.figsize"]=12,6
df2.plot.bar().grid()
```

```
/Users/mmezghani/opt/anaconda3/lib/python3.8/site-packages/pandas/core/strings/accessor.py:101: UserWarning: This pattern has match groups. To actual
ly get the groups, use str.extract.
  return func(self, *args, **kwargs)
```



2- Variation de vente annuelle de chacune des 6 marques

In [16]:
```python
df2.plot.line().grid()
```



3- Interprétation

- On note que les voitures la majorité des marques suivent une allure de croissance au cours des années
- La chevrolet est la marque la plus vendue annuellement

Étape 4: Analyse des relations

1- Analyse de la relation entre le price et le mileage

In [17]:
```python
model1 = ols('price ~ mileage', data=df2010).fit()
print(model1.summary())
```

```
                            OLS Regression Results                            
==============================================================================
Dep. Variable:                  price   R-squared:                       0.129
Model:                            OLS   Adj. R-squared:                  0.129
Method:                 Least Squares   F-statistic:                     349.1
Date:                Tue, 30 Mar 2021   Prob (F-statistic):           9.38e-73
Time:                        16:28:50   Log-Likelihood:                -25326.
No. Observations:                2361   AIC:                         5.066e+04
Df Residuals:                    2359   BIC:                         5.067e+04
Df Model:                           1                                         
Covariance Type:            nonrobust                                         
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Intercept    2.33e+04    313.221     74.396      0.000    2.27e+04    2.39e+04
mileage       -0.0835      0.004    -18.683      0.000      -0.092      -0.075
==============================================================================
Omnibus:                      543.752   Durbin-Watson:                   1.680
Prob(Omnibus):                  0.000   Jarque-Bera (JB):             1327.196
Skew:                           1.253   Prob(JB):                    6.35e-289
Kurtosis:                       5.686   Cond. No.                     9.68e+04
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 9.68e+04. This might indicate that there are
strong multicollinearity or other numerical problems.
```

- Nous disposons de deux varaibles quantitatives --> L'analyse de la relation peut se faire à l'aide d'une analyse de régression linéaire.
- La valeur du coefficient de détermination R^2 étant faible (0.129) --> On peut conclure qu'il y a une faible relation entre le price et le mileage.

2- Analyse de variance : Variable quantitative X et varaible qualitative Y

In [18]:
```python
model1 = ols('price ~ title_status', data=df2010).fit()
print(model1.summary())
```

```
                            OLS Regression Results                            
==============================================================================
Dep. Variable:                  price   R-squared:                       0.077
Model:                            OLS   Adj. R-squared:                  0.077
Method:                 Least Squares   F-statistic:                     197.4
Date:                Tue, 30 Mar 2021   Prob (F-statistic):           4.00e-43
Time:                        16:28:54   Log-Likelihood:                -25394.
No. Observations:                2361   AIC:                         5.079e+04
Df Residuals:                    2359   BIC:                         5.080e+04
Df Model:                           1                                         
Covariance Type:            nonrobust                                         
=====================================================================================================
                                        coef    std err          t      P>|t|      [0.025      0.975]
-----------------------------------------------------------------------------------------------------
Intercept                          1.996e+04    238.685     83.622      0.000    1.95e+04    2.04e+04
title_status[T.salvage insurance]  -1.63e+04   1159.772    -14.050      0.000   -1.86e+04    -1.4e+04
==============================================================================
Omnibus:                      399.300   Durbin-Watson:                   1.642
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              732.626
Skew:                           1.051   Prob(JB):                    8.17e-160
Kurtosis:                       4.741   Cond. No.                         4.97
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
```

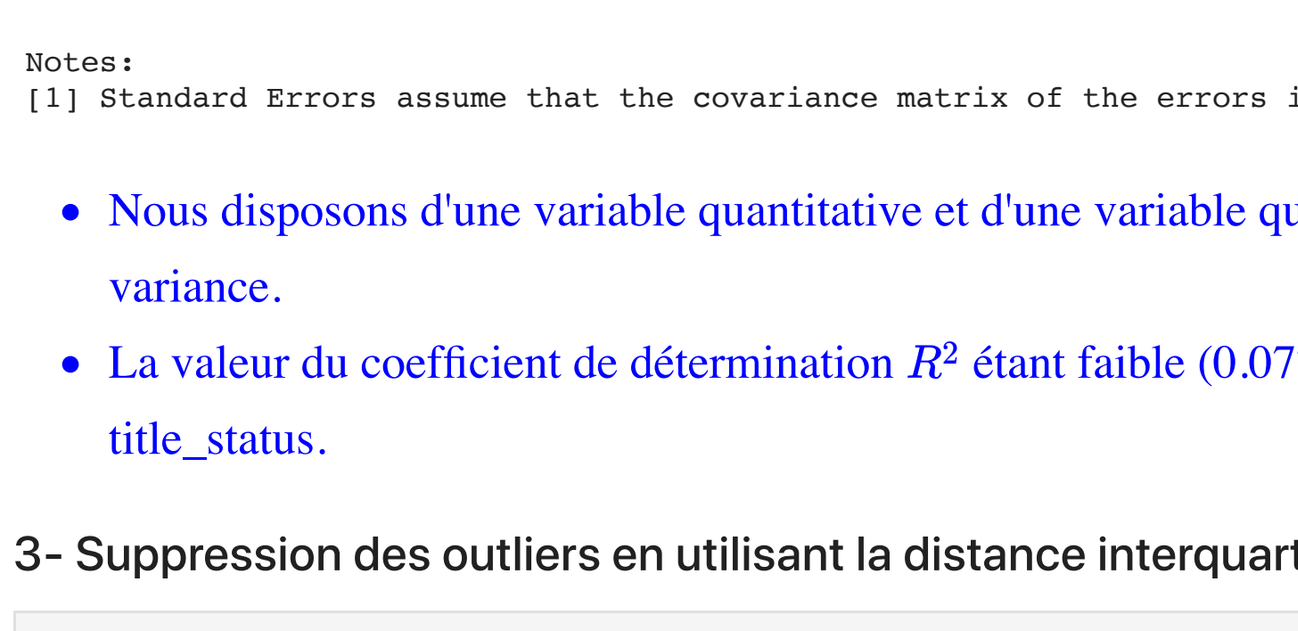
- Nous disposons d'une variable quantitative et d'une variable qualitative --> L'analyse de la relation peut se faire à l'aide d'une analyse de variance.
- La valeur du coefficient de détermination R^2 étant faible (0.077) --> On peut conclure qu'il y a une faible relation entre le price et title_status.

3- Suppression des outliers en utilisant la distance interquartile

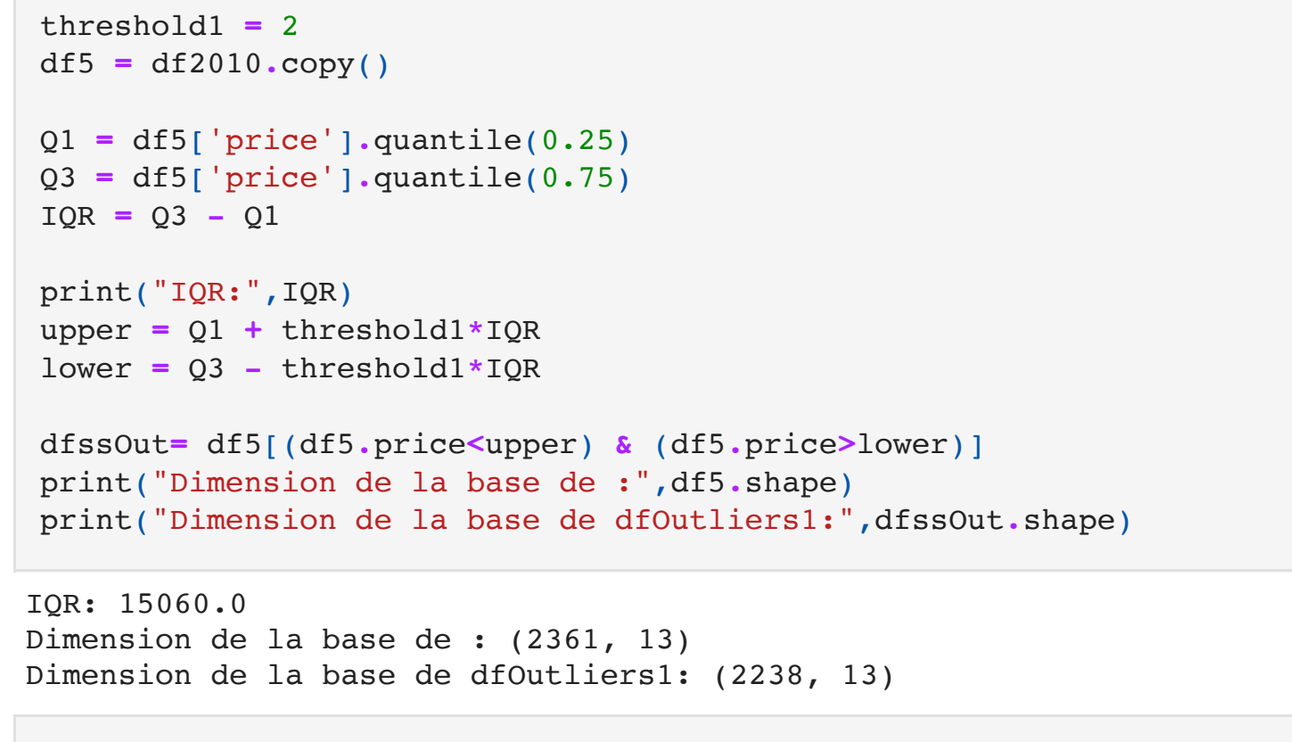
In [19]:
```python
threshold1 = 2
df5 = df2010.copy()

Q1 = df5['price'].quantile(0.25)
Q3 = df5['price'].quantile(0.75)
IQR = Q3 - Q1

print("IQR:",IQR)
upper = Q1 + threshold1*IQR
lower = Q3 - threshold1*IQR

dfssOut= df5[(df5.price<upper) & (df5.price>lower)]
print("Dimension de la base de :",df5.shape)
print("Dimension de la base de dfOutliers1:",dfssOut.shape)
```

```
IQR: 15060.0
Dimension de la base de : (2361, 13)
Dimension de la base de dfOutliers1: (2238, 13)
```

In [20]:
```python
df5.boxplot(column='price', figsize=(8,4),vert = False)
None
```



In [21]:
```python
dfssOut.boxplot(column='price', figsize=(8,4),vert = False)
None
```



In [22]:
```python
model2 = ols('price ~ mileage', data=dfssOut).fit()
print(model2.summary())
```

```
                            OLS Regression Results                            
==============================================================================
Dep. Variable:                  price   R-squared:                       0.146
Model:                            OLS   Adj. R-squared:                  0.146
Method:                 Least Squares   F-statistic:                     382.0
Date:                Tue, 30 Mar 2021   Prob (F-statistic):           1.13e-78
Time:                        16:29:01   Log-Likelihood:                -23474.
No. Observations:                2238   AIC:                         4.695e+04
Df Residuals:                    2236   BIC:                         4.696e+04
Df Model:                           1                                         
Covariance Type:            nonrobust                                         
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Intercept     2.1e+04    254.425     82.551      0.000    2.05e+04    2.15e+04
mileage       -0.0694      0.004    -19.546      0.000      -0.076      -0.062
==============================================================================
Omnibus:                      124.839   Durbin-Watson:                   1.709
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              154.808
Skew:                           0.549   Prob(JB):                     2.42e-34
Kurtosis:                       3.674   Cond. No.                     9.91e+04
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 9.91e+04. This might indicate that there are
strong multicollinearity or other numerical problems.
```

- Bien que les valeurs aberrantes soient supprimées, la corrélation reste faible (R^2 étant égal à 0.146) --> On peut conclure qu'il y a une faible relation entre le price et le mileage.

Étape 5: Inférence stat

1- Test d'hypothèse

- Il semble que la moyenne de prix des voitures est statistiquement égale à 20000.
- Dans ce cas l'hypothèse nulle est $H_0 : \mu = 20000$ et l'hypothèse alternative $H_1 : \mu \neq 20000$.

2 - Réalisation du test d'hypothèse

In [23]:
```python
results = stats.ttest_1samp(df2010['price'], 20000)
print('p_valeur:' , results[1])
```

```
p_valeur: 0.002667250812289089
```

In [24]:
```python
df2010['price'].mean()
```

Out[24]: 19269.02414231258

3- Interprétation des résultats

- p = 0.002 < 0.05 --> On rejete l hyptoèse nulle --> la moyenne de prix des voitures est statistiquement différente de 20000

In [ ]: