```
    to_return.insert(4, "species", df_modified.species, True)
    return to_return

def find_outliers(data, column):
    df = data.copy()
    q1 = df[column].quantile(0.25)
    q3 = df[column].quantile(0.75)
    iqr = q3 - q1
    return df[(df[column] > (q3 + 1.5 * iqr)) | (df[column] < (q1 - 1.5 * iqr))]
```

0.0.1 Charger les bibliothèques

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

0.0.2 Analyse de la bases de données Brain

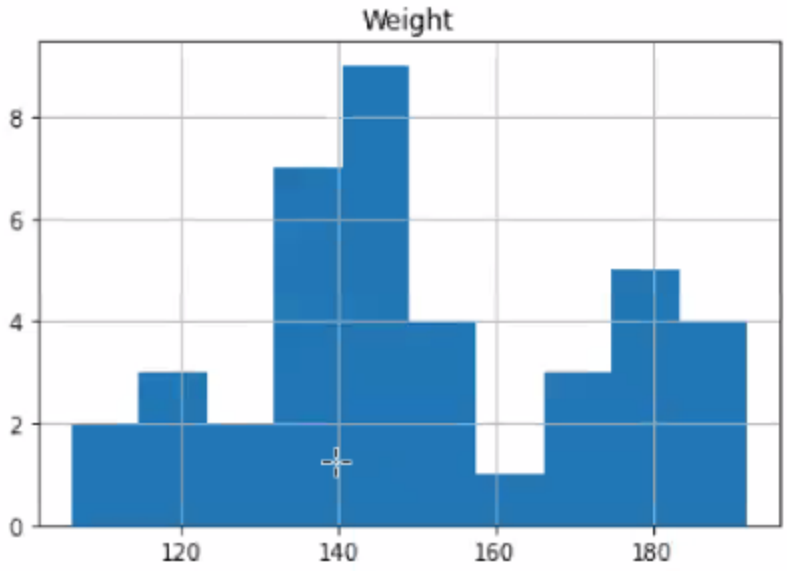
[2]:
```
data = pd.read_excel('brain_size_Mod.xlsx')
```

[3]:
```
print(data.dtypes)
```

```
Unnamed: 0      int64
Gender         object
FSIQ            int64
VIQ             int64
PIQ             int64
Weight          int64
Height        float64
MRI_Count       int64
Activity       object
dtype: object
```

[11]:
```
data.hist(column='Weight')
```

[11]:
```
array([[<AxesSubplot:title={'center':'Weight'}>]], dtype=object)
```




```
dimension1 = data.shape
print("Dimension de la base de données brain",dimension1)
```

```
Dimension de la base de données brain (40, 9)
```

0.1 Stat unvarié

0.1.1 Variable qualitative

```
E1 = data['Gender'].value_counts()
print("Effectif :\n",E1)
```

```
Effectif :
 Female    20
Male       20
Name: Gender, dtype: int64
```

```
E2 = data['Activity'].value_counts()
print("Effectif :\n",E2)
```

```
Effectif :
 doctor       12
professor     11
```

Calculer mayen median quarties.. etc

0.1.2 Variable quantitative

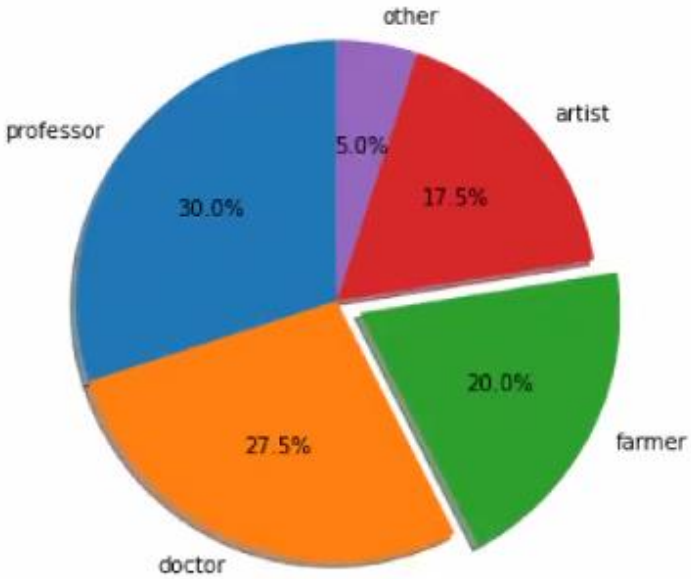
10]:
```
data.describe()
```

10]:

	Unnamed: 0	FSIQ	VIQ	PIQ	Weight	Height	\
count	40.000000	40.000000	40.000000	40.00000	40.00000	40.000000	
mean	20.500000	113.450000	112.350000	111.02500	150.80000	68.662500	
std	11.690452	24.082071	23.616107	22.47105	22.90023	4.036988	
min	1.000000	77.000000	71.000000	72.00000	106.00000	62.000000	
25%	10.750000	89.750000	90.000000	88.25000	135.75000	66.000000	
50%	20.500000	116.500000	113.000000	115.00000	146.50000	68.250000	
75%	30.250000	135.500000	129.750000	128.00000	172.00000	70.875000	
max	40.000000	144.000000	150.000000	150.00000	192.00000	77.000000	

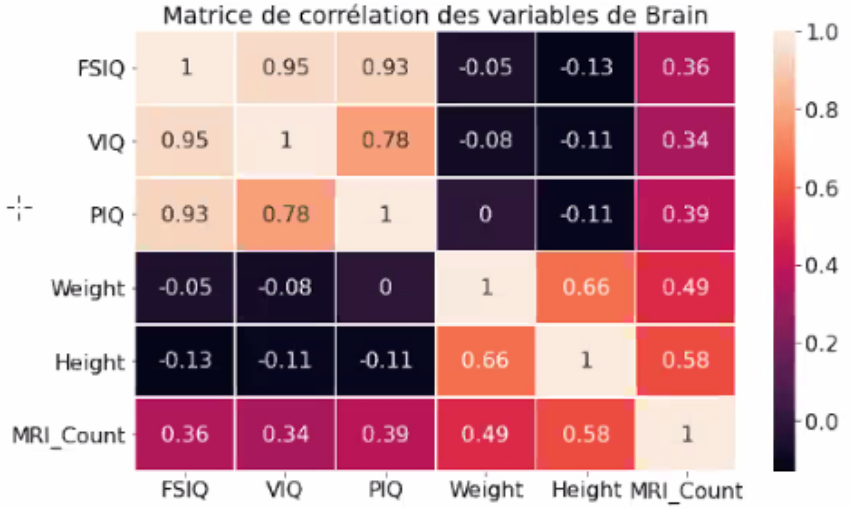
	MRI_Count
count	4.000000e+01
mean	9.087550e+05
std	7.228205e+04
min	7.906190e+05
25%	8.559185e+05
50%	9.053990e+05
75%	9.500780e+05

```
ax1.pie(E2, explode=explode, labels=labels, autopct='%1.1f%%',
        shadow=True, startangle=90)
ax1.axis('equal')
plt.tight_layout()
plt.show()
```

3]:
```
figsize=(6,4)
explode = (0, 0, 0.1, 0, 0)
fig1, ax1 = plt.subplots(figsize=(6,4))
ax1.pie(E2, explode=explode, labels=labels, autopct='%1.1f%%',
        shadow=True, startangle=90)
ax1.axis('equal')
plt.tight_layout()
plt.show()
```

[21]:
```
plt.rcParams.update({'font.size': 16})
fig = plt.figure(figsize=(10,6))
sns.heatmap(data=matrice_corr, annot=True,linewidths=.5)
plt.title('Matrice de corrélation des variables de Brain', fontsize = 18, color␣
↪= 'black')
None
```



[20]:
```
matrice_corr = data.iloc[:,1:9 ].corr().round(2)
print(matrice_corr)
```

```
             FSIQ   VIQ   PIQ  Weight  Height  MRI_Count
FSIQ         1.00  0.95  0.93   -0.05   -0.13       0.36
VIQ          0.95  1.00  0.78   -0.08   -0.11       0.34
PIQ          0.93  0.78  1.00    0.00   -0.11       0.39
Weight      -0.05 -0.08  0.00    1.00    0.66       0.49
Height      -0.13 -0.11 -0.11    0.66    1.00       0.58
MRI_Count    0.36  0.34  0.39    0.49    0.58       1.00
```

0.1.4 Visualisation de la corrélation

0.1.5 Corrélation de pearson (2)

[22]:
```
from scipy.stats import pearsonr
pd.DataFrame(pearsonr(data["PIQ"], data["VIQ"]),␣
↪index=['pearson_coeff','p-value'], columns=['Results'])
```

[22]:

	Results
pearson_coeff	7.781351e-01
p-value	3.438186e-09

0.1.6 Analyse de regression (1)

[23]:
```
lm = LinearRegression()
x = np.array(data["PIQ"]).reshape((-1, 1))
y = data["VIQ"]
```

[24]:
```
model1 = lm.fit(x,y)
```

[25]:
```
from math import *
r_sq = model1.score(x, y)
print('coefficient of determination:', r_sq)
print('racine carrée de coefficient of determination:', sqrt(r_sq))
```

```
coefficient of determination: 0.6054942551579...
```

0.1.6 Analyse de regression (1)

[23]:
```
lm = LinearRegression()
x = np.array(data["PIQ"]).reshape((-1, 1))
y = data["VIQ"]
```

[24]:
```
model1 = lm.fit(x,y)
```

[25]:
```
from math import *
r_sq = model1.score(x, y)
print('coefficient of determination:', r_sq)
print('racine carrée de coefficient of determination:', sqrt(r_sq))
```

```
coefficient of determination: 0.605494255157936
racine carrée de coefficient of determination: 0.778135113690377
```

0.1.7 Analyse de régression (2)

[26]:
```
model1 = ols('PIQ ~ VIQ', data=data).fit()
print(model1.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                    PIQ   R-squared:                       0.605
Model:                            OLS   Adj. R-squared:                  0.595
Method:                 Least Squares   F-statistic:                     58.32
Date:                Sat, 20 Feb 2021   Prob (F-statistic):           3.44e-09
Time:                        21:12:15   Log-Likelihood:                -162.14
No. Observations:                  40   AIC:                             328.3
Df Residuals:                      38   BIC:                             331.7
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
```

[27]:
```
import statsmodels.api
result = ols('data["VIQ"] ~ data["Gender"]', data=data).fit()
table = statsmodels.api.stats.anova_lm(result)
table
```

[27]:

	df	sum_sq	mean_sq	F	PR(>F)
data["Gender"]	1.0	336.4	336.400000	0.596936	0.444529
Residual	38.0	21414.7	563.544737	NaN	NaN

[28]:
```
X = data["Gender"]
Y = np.array(data["VIQ"]).reshape((-1, 1))
```

[ ]:

[ ]:

[ ]:

[29]:
```
model1 = ols('data["VIQ"] ~ data["Gender"]', data=data).fit()
print(model1.summary())
```

```
                            OLS Regression Results
==============================================================================

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

[30]:
```
def eta_squared(x,y):
    moyenne_y = y.mean()
    classes = []
    for classe in x.unique():
        yi_classe = y[x==classe]
        classes.append({'ni': len(yi_classe),
                        'moyenne_classe': yi_classe.mean()})
    SCT = sum([(yj-moyenne_y)**2 for yj in y])
    print ('Somme carré total', SCT)
    SCE = sum([c['ni']*(c['moyenne_classe']-moyenne_y)**2 for c in classes])
    print ('Somme carré total', SCE)
    return SCE/SCT
eta_squared(X,Y)
```

```
Somme carré total [21751.1]
Somme carré total 336.39999999999964
```

[30]:
```
array([0.01546588])
```

[31]:
```
plt.rcParams.update({'font.size': 14})
data.boxplot(column='VIQ',by='Gender', figsize=(8,6))
plt.xlabel("Genre")
plt.ylabel("VIQ")
```

[31]:
```
Text(0, 0.5, 'VIQ')
```

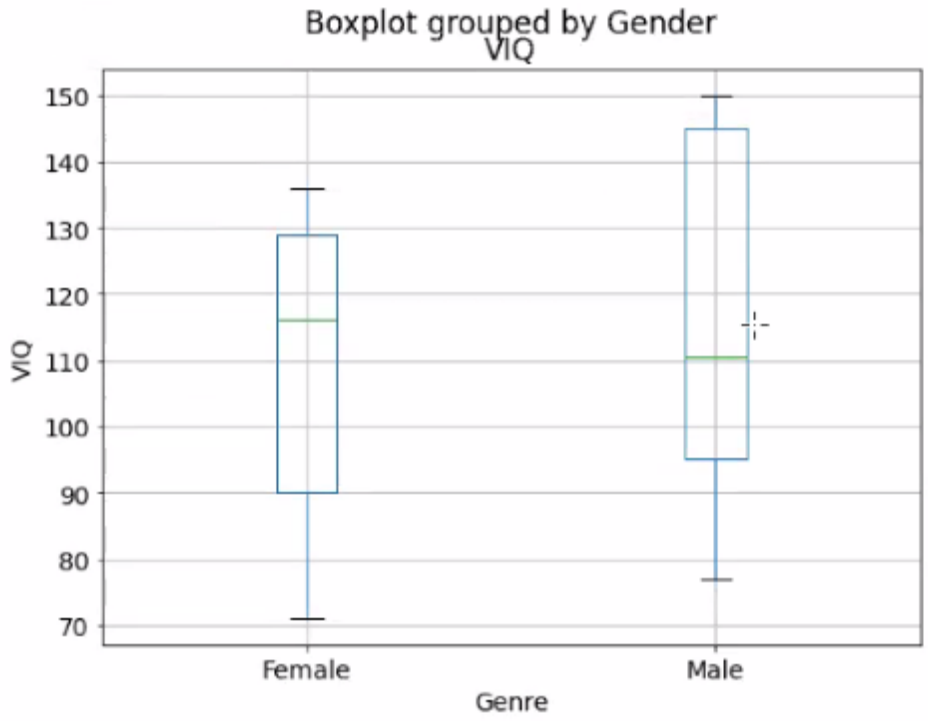
[ ]:

[29]:
```
model1 = ols('data["VIQ"] ~ data["Gender"]', data=data).fit()
print(model1.summary())
```

OLS Regression Results

Dep. Variable:	data["VIQ"]	R-squared:	0.015
Model:	OLS	Adj. R-squared:	-0.010
Method:	Least Squares	F-statistic:	0.5969
Date:	Sat, 20 Feb 2021	Prob (F-statistic):	0.445
Time:	21:12:28	Log-Likelihood:	-182.42
No. Observations:	40	AIC:	368.8
Df Residuals:	38	BIC:	372.2
Df Model:	1		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	109.4500	5.308	20.619	0.000	98.704	120.196

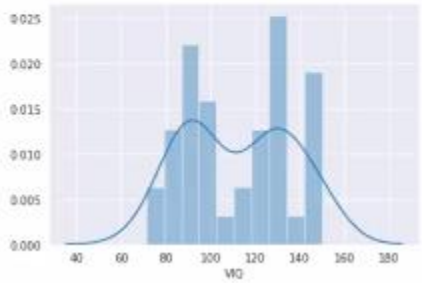



```
ax = sns.violinplot(x=data['Gender'], y=data['VIQ'], data=data)
```

Il n'y a pas de valeurs aberrantes dans le jeu de donnée. On observe de nouveau que la m
droite.

In [203]:

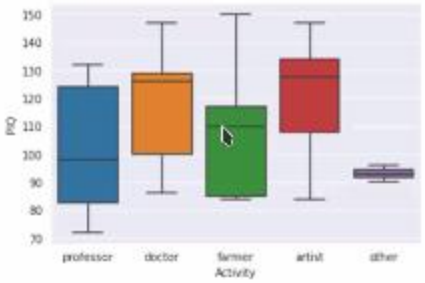
```
1 sns.set_style("darkgrid")
2 sns.distplot(VIQ, kde=True, bins=10)
3
4 None
```



???

R^2 est faible, donc la relation entre les deux variables est très pauvre.

In [191]:
```
1 sns.boxplot(x="Activity", y = "PIQ", data = df)
2 None
```



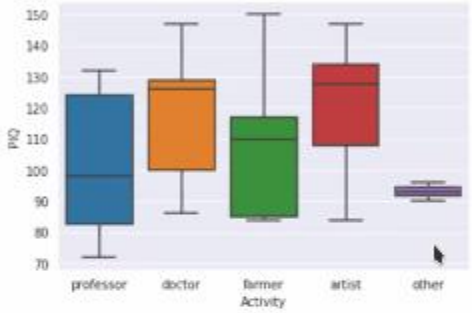
Il y a un bon chevauchement pour ce qui est du résultat au test `PIQ`, ce qui veut dire que la profession n'a pas

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

???

R^2 est faible, donc la relation entre les deux variables est très pauvre.

In [191]:
```
1 sns.boxplot(x="Activity", y = "PIQ", data = df)
2 None
```



Il y a un bon chevauchement pour ce qui est du résultat au test `PIQ`, ce qui veut dire que la profession n'a pas de lien fort avec le score au test, à l'exception de `other`, mais seulement dans un sens: sachant `other`, on a une bonne idée du résultat, mais sachant le résultat, on ne peut pas dire à quelle profession il appartient.

4. Pensez-vous que les variables Activity et Gender sont reliées ? Faites une analyse numérique. Représentez la table de contingence. Interprétez et commentez les résultats.

In [196]:
```
1 table = pd.crosstab(df.Activity, df.Gender)
2 table
```

Out[196]:

Gender	Female	Male
Activity		
artist	8	0

3. Pensez-vous que les variables Activity et FSIQ sont reliées ? Faites une analyse numérique et une analyse graphique. Interprétez et commentez les résultats.

Il faut faire une analyse de variance (ANOVA).

[194]:
```
model2 = ols("FSIQ ~ Activity", data= df).fit()
rapport_correlation = anova_lm(model2)
display(rapport_correlation)
print(model2.summary())
```

	df	sum_sq	mean_sq	F	PR(>F)
Activity	4.0	3127.507684	781.876921	1.404061	0.2529
Residual	35.0	19490.392316	556.868352	NaN	NaN

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                   FSIQ   R-squared:                       0.138
Model:                           OLS   Adj. R-squared:                  0.040
Method:                Least Squares   F-statistic:                     1.404
Date:               Tue, 23 Feb 2021   Prob (F-statistic):              0.253
Time:                       16:50:46   Log-Likelihood:                -180.53
No. Observations:                 40   AIC:                             371.1
Df Residuals:                     35   BIC:                             379.5
Df Model:                          4
Covariance Type:           nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.825      0.975]
------------------------------------------------------------------------------
Intercept    119.3750      8.343     14.308      0.000     102.437     136.313
Activity[T.doctor]     3.2917     10.771      0.306      0.762     -18.575      25.158
Activity[T.farmer]   -12.8036     12.213     -1.048      0.302     -37.598      11.990
Activity[T.other]    -29.8750     18.656     -1.601      0.118     -67.748       7.998
Activity[T.professor]   -11.5568     10.965     -1.054      0.299     -33.817      10.703
==============================================================================
Omnibus:                       20.660   Durbin-Watson:                   1.515
Prob(Omnibus):                  0.000   Jarque-Bera (JB):                3.940
Skew:                          -0.289   Prob(JB):                        0.139
Kurtosis:                       1.575   Cond. No.                         6.44
==============================================================================
```

avec le score au test, à l'exception de `other`, mais seulement dans un sens: sachant `other`, on a une bonne idée du résultat, mais sachant le résultat, on ne peut pas dire à quelle profession il appartient.

4. Pensez-vous que les variables Activity et Gender sont reliées ? Faites une analyse numérique. Représentez la table de contingence. Interprétez et commentez les résultats.

96]:
```
table = pd.crosstab(df.Activity, df.Gender)
table
```

96]:

Activity	Female	Male
artist	8	0
doctor	1	11
farmer	0	7
other	2	0
professor	9	2

Pour voir si les variables sont reliées, on va procéder au test du χ^2, mais de prime abord, il semble que oui (la table de contingence est très débalancée).

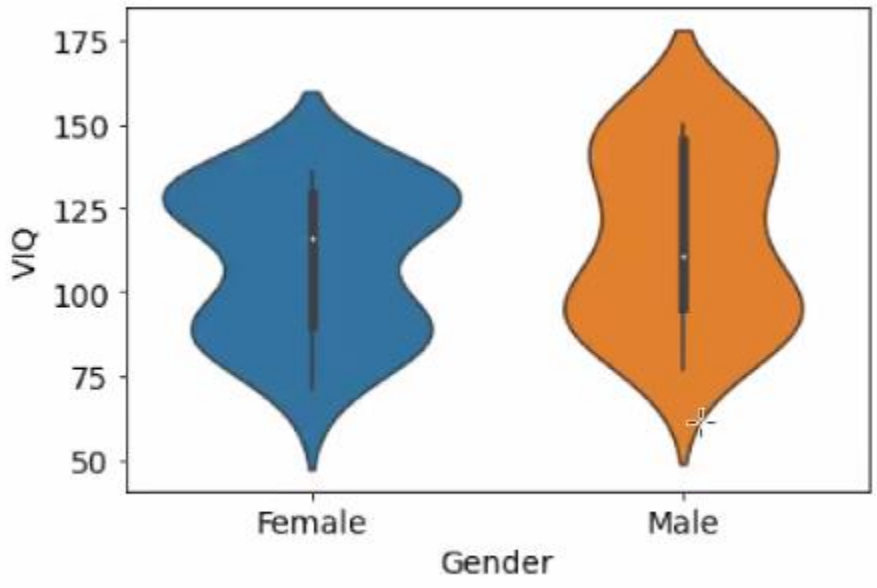
01]:
```
results = chi2_contingency(table)

print("Chi square:", results[0])
print("p-value:", results[1])
# print("Degrees of freedom:", results[2]) -- intérêt?
```

```
Chi square: 29.78787878787879
p-value: 5.405997743214392e-06
```

La valeur de χ^2 est plutôt élevée, donc on ne peut pas dire qu'il n'y a pas de lien entre les deux variables. Du reste, nous avons une $p-value$ qui est de l'ordre de 5 millionnième, ce qui veut dire que le lien est statistiquement significatif.

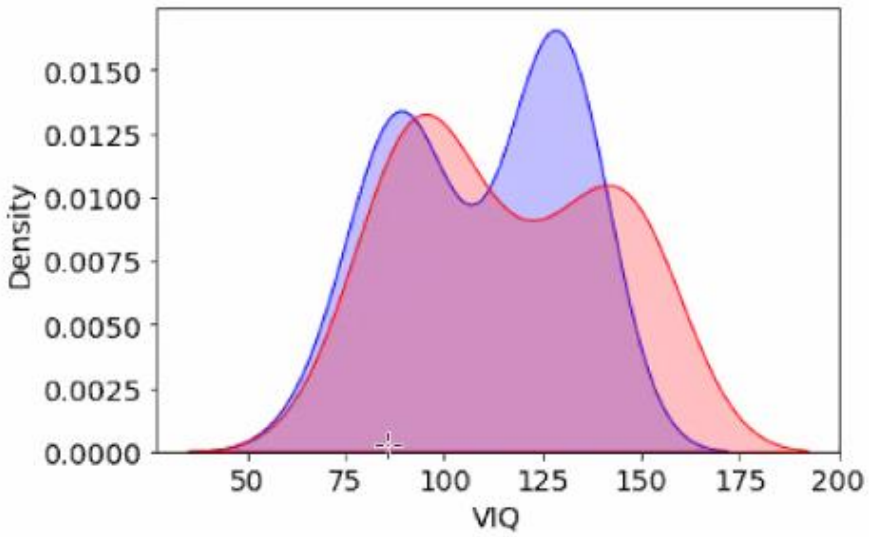
Exercice 3 : Inférence statistique



[33]:
```
sns.kdeplot(data.loc[(data['Gender']=='Female'),
            'VIQ'], color='b', shade=True, Label='Female')
sns.kdeplot(data.loc[(data['Gender']=='Male'),
            'VIQ'], color='r', shade=True, Label='Male')
None
```

```
/Users/nmezghani/opt/anaconda3/lib/python3.8/site-
packages/seaborn/distributions.py:949: MatplotlibDeprecationWarning: Case-
insensitive properties were deprecated in 3.3 and support will be removed two
minor releases later
  scout = self.ax.fill_between([], [], **plot_kws)
/Users/nmezghani/opt/anaconda3/lib/python3.8/site-
packages/seaborn/distributions.py:992: MatplotlibDeprecationWarning: Case-
```

```
minor releases later
  artist = ax.fill_between(
```



[34]:
```
from scipy.stats import chi2_contingency
table= pd.crosstab(data['Gender'], data['Activity'])
table
```

[34]:

Activity	artist	doctor	farmer	other	professor
Gender					
Female	8	1	0	2	9

3. Pensez-vous que les variables Activity et FSIQ sont reliées ? Faites une analyse numérique et une analyse graphique. Interprétez et commentez les résultats. ¶

Il faut faire une analyse de variance (ANOVA).

[194]:
```
model2 = ols("FSIQ ~ Activity", data= df).fit()
rapport_correlation = anova_lm(model2)
display(rapport_correlation)
print(model2.summary())
```

	df	sum_sq	mean_sq	F	PR(>F)
Activity	4.0	3127.507684	781.876921	1.404061	0.2529
Residual	35.0	19490.392316	556.868352	NaN	NaN

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                   FSIQ   R-squared:                       0.138
Model:                            OLS   Adj. R-squared:                  0.040
Method:                 Least Squares   F-statistic:                     1.404
Date:                Tue, 23 Feb 2021   Prob (F-statistic):              0.253
Time:                        16:50:46   Log-Likelihood:                -180.53
No. Observations:                  40   AIC:                             371.1
Df Residuals:                      35   BIC:                             379.5
Df Model:                           4
Covariance Type:            nonrobust
==========================================================================================
                             coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------------------
Intercept                119.3750      8.343     14.308      0.000     102.437     136.313
Activity[T.doctor]         3.2917     10.771      0.306      0.762     -18.575      25.158
Activity[T.farmer]       -12.8036     12.213     -1.048      0.302     -37.598      11.990
Activity[T.other]        -29.8750     18.656     -1.601      0.118     -67.748       7.998
Activity[T.professor]    -11.5568     10.965     -1.054      0.299     -33.817      10.703
==============================================================================
Omnibus:                       20.660   Durbin-Watson:                   1.515
Prob(Omnibus):                  0.000   Jarque-Bera (JB):                3.940
Skew:                          -0.289   Prob(JB):                        0.139
Kurtosis:                       1.575   Cond. No.                         6.44
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specifie
```

Exercice 3 : Inférence statistique

L'objectif d'un projet de recherche est déterminer la valeur de FSIQ au sein de la population canadienne. Pour cela, un échantillon de 40 observations a été collecté. Deux chercheurs émettent deux hypothèses différentes.

Le premier pense que la moyenne de la variable FSIQ la population est égale à 100. Le deuxième pense que la moyenne de la variable FSIQ la population est inférieure à 160. Que pensez-vous de ces deux hypothèses ?

In [85]:
```
#Test d'hypothèses

# Cas 1 Le premier cas montre que
#l'hypothèse nulle est que le FSIQ moyen
#de la population est égal à 100. Nous allons ensuite tester
#si avec l'échantillon de 40 cas nous acceptons ou rejetons cette hypothèse

#H_0: μ = 100
#H_1: μ # 100

results = stats.ttest_1samp(df['FSIQ'],100,0)
print('statistics: ',results[0])
print('p_value: ', results[1])
```

```
statistics:  3.532307014238269
p_value:  0.0010766792736967715
```

La valeur p est inférieure à 0,05, ce qui indique que nous rejetons l'hypothèse nulle que la moyenne de la variable FSIQ est égale à 100

In [ ]:
```
# Cas 2

# Le cas 2 ne peut pas être traité dans un test d'hypothèse car nous n'avons pas de paramètre fixe à tester.
```