In [2]:
```
import numpy as np
import pandas as pd
import matplotlib as mp
import statsmodels.api as sm
```

In [3]:
```
mu, sigma = 0, 5 # mean and standard deviation of normal distribution for the
x = np.random.uniform(40,80,100)
epsilon = np.random.normal(mu,sigma,100)
y = 3 + 4*x + epsilon
```

In [5]:
```
model_reg = sm.OLS(y,x).fit()
model_reg.summary()
```

Out[5]:

OLS Regression Results

Dep. Variable:	y	**R-squared (uncentered):**	1.000
Model:	OLS	**Adj. R-squared (uncentered):**	1.000
Method:	Least Squares	**F-statistic:**	2.157e+05
Date:	Sun, 25 Sep 2022	**Prob (F-statistic):**	4.45e-167
Time:	10:31:54	**Log-Likelihood:**	-308.07
No. Observations:	100	**AIC:**	618.1
Df Residuals:	99	**BIC:**	620.7
Df Model:	1		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
x1	4.0379	0.009	464.475	0.000	4.021	4.055

Omnibus:	0.299	**Durbin-Watson:**	1.771
Prob(Omnibus):	0.861	**Jarque-Bera (JB):**	0.102
Skew:	-0.072	**Prob(JB):**	0.950
Kurtosis:	3.058	**Cond. No.**	1.00

Notes:

[1] R^2 is computed without centering (uncentered) since the model does not contain a constant.

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

In [9]:
```
x_updated = sm.add_constant(x)
model_updated = sm.OLS(y,x_updated).fit()
model_updated.summary()
```

Out[9]:

OLS Regression Results

Dep. Variable:	y	**R-squared:**	0.987
Model:	OLS	**Adj. R-squared:**	0.987
Method:	Least Squares	**F-statistic:**	7643.
Date:	Sun, 25 Sep 2022	**Prob (F-statistic):**	8.45e-95
Time:	10:38:20	**Log-Likelihood:**	-307.73
No. Observations:	100	**AIC:**	619.5
Df Residuals:	98	**BIC:**	624.7
Df Model:	1		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	2.2556	2.788	0.809	0.420	-3.277	7.788
x1	4.0015	0.046	87.425	0.000	3.911	4.092

Omnibus:	0.306	**Durbin-Watson:**	1.801
Prob(Omnibus):	0.858	**Jarque-Bera (JB):**	0.085
Skew:	-0.060	**Prob(JB):**	0.958
Kurtosis:	3.078	**Cond. No.**	320.

Notes:

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

In [16]:
```
# We now generate autocorrelated error terms
epsilon[0] = np.random.normal(mu,sigma,1)
for i in range(0,99):
    epsilon[i+1]=0.4*epsilon[i]+0.6*np.random.normal(mu,sigma,1)
```

In [17]:
```
y = 3 + 4*x + epsilon
```

In [18]:
```
x_updated = sm.add_constant(x)
model_OLS = sm.OLS(y,x_updated).fit()
model_OLS.summary()
```

Out[18]:

OLS Regression Results

Dep. Variable:	y	**R-squared:**	0.994

Model:	OLS	**Adj. R-squared:**	0.994
Method:	Least Squares	**F-statistic:**	1.668e+04
Date:	Sun, 25 Sep 2022	**Prob (F-statistic):**	2.92e-111
Time:	10:52:07	**Log-Likelihood:**	-268.82
No. Observations:	100	**AIC:**	541.6
Df Residuals:	98	**BIC:**	546.8
Df Model:	1		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	3.1394	1.889	1.662	0.100	-0.609	6.888
x1	4.0055	0.031	129.146	0.000	3.944	4.067

Omnibus:	1.295	**Durbin-Watson:**	0.928
Prob(Omnibus):	0.523	**Jarque-Bera (JB):**	1.155
Skew:	0.087	**Prob(JB):**	0.561
Kurtosis:	2.503	**Cond. No.**	320.

Notes:

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

In [26]:
```
from scipy.linalg import toeplitz
toeplitz(np.array([1,0.5,0,0,0,0,0,0]))
```

Out[26]:
```
array([[1. , 0.5, 0. , 0. , 0. , 0. , 0. , 0. ],
       [0.5, 1. , 0.5, 0. , 0. , 0. , 0. , 0. ],
       [0. , 0.5, 1. , 0.5, 0. , 0. , 0. , 0. ],
       [0. , 0. , 0.5, 1. , 0.5, 0. , 0. , 0. ],
       [0. , 0. , 0. , 0.5, 1. , 0.5, 0. , 0. ],
       [0. , 0. , 0. , 0. , 0.5, 1. , 0.5, 0. ],
       [0. , 0. , 0. , 0. , 0. , 0.5, 1. , 0.5],
       [0. , 0. , 0. , 0. , 0. , 0. , 0.5, 1. ]])
```

In [33]:
```
rho = 0.4
cov_matrix = sigma**2*toeplitz(np.append([1,rho],np.zeros(98)))
sm.GLS(y,x_updated,cov_matrix).fit().summary()
```

Out[33]:

GLS Regression Results

Dep. Variable:	y	**R-squared:**	0.997
Model:	GLS	**Adj. R-squared:**	0.997
Method:	Least Squares	**F-statistic:**	3.188e+04
Date:	Sun, 25 Sep 2022	**Prob (F-statistic):**	5.51e-125

Time:	11:12:50	**Log-Likelihood:**	-253.63
No. Observations:	100	**AIC:**	511.3
Df Residuals:	98	**BIC:**	516.5
Df Model:	1		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	5.1123	1.409	3.628	0.000	2.316	7.909
x1	3.9723	0.022	178.538	0.000	3.928	4.016

Omnibus:	0.333	**Durbin-Watson:**	1.649
Prob(Omnibus):	0.847	**Jarque-Bera (JB):**	0.503
Skew:	0.078	**Prob(JB):**	0.778
Kurtosis:	2.689	**Cond. No.**	194.

Notes:

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

In [30]:

```
np.append([1,rho],np.zeros(98))
```

Out[30]:

```
array([1. , 0.4, 0. , 0. , 0. , 0. , 0. , 0. , 0. , 0. , 0. , 0. , 0. ,
       0. , 0. , 0. , 0. , 0. , 0. , 0. , 0. , 0. , 0. , 0. , 0. , 0. ,
       0. , 0. , 0. , 0. , 0. , 0. , 0. , 0. , 0. , 0. , 0. , 0. , 0. ,
       0. , 0. , 0. , 0. , 0. , 0. , 0. , 0. , 0. , 0. , 0. , 0. , 0. ,
       0. , 0. , 0. , 0. , 0. , 0. , 0. , 0. , 0. , 0. , 0. , 0. , 0. ,
       0. , 0. , 0. , 0. , 0. , 0. , 0. , 0. , 0. , 0. , 0. , 0. , 0. ,
       0. , 0. , 0. , 0. , 0. , 0. , 0. , 0. , 0. , 0. , 0. , 0. , 0. ,
       0. , 0. , 0. , 0. , 0. , 0. , 0. , 0. , 0. ])
```

In [ ]: