Multiple linear regression and adjusted R-squared

Import the relevant libraries

In [3]:

```python
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import statsmodels.api as sm
import seaborn
seaborn.set()
```

Load the data

In [4]:

```python
# Load the data from a .csv in the same folder
data = pd.read_csv('1.02.Multiple_linear_regression.csv')
```

In [5]:

```python
# Let's check what's inside this data frame
data
```

Out[5]:

	SAT	GPA	Rand 1,2,3
0	1714	2.40	1
1	1664	2.52	3
2	1760	2.54	3
3	1685	2.74	3
4	1693	2.83	2
...	...	...	...
79	1936	3.71	3
80	1810	3.71	1
81	1987	3.73	3
82	1962	3.76	1
83	2050	3.81	2

84 rows × 3 columns

In [6]:

```python
# This method gives us very nice descriptive statistics. We don't need this as of now, but will later on!
data.describe()
```

Out[6]:

	SAT	GPA	Rand 1,2,3
count	84.000000	84.000000	84.000000
mean	1845.273810	3.330238	2.059524
std	104.530661	0.271617	0.855192
min	1634.000000	2.400000	1.000000
25%	1772.000000	3.190000	1.000000
50%	1846.000000	3.380000	2.000000
75%	1934.000000	3.502500	3.000000
max	2050.000000	3.810000	3.000000

Create your first multiple regression

In [7]:

```python
# Following the regression equation, our dependent variable (y) is the GPA
y = data ['GPA']
# Similarly, our independent variable (x) is the SAT score
x1 = data [['SAT','Rand 1,2,3']]
```

In [8]:

```python
# Add a constant. Esentially, we are adding a new column (equal in lenght to x), which consists only of 1s
x = sm.add_constant(x1)
# Fit the model, according to the OLS (ordinary least squares) method with a dependent variable y and an idepen
results = sm.OLS(y,x).fit()
```

In [9]:

```python
# Print a nice summary of the regression.
results.summary()
```

Out[9]:

OLS Regression Results

Dep. Variable:	GPA	**R-squared:**	0.407
Model:	OLS	**Adj. R-squared:**	0.392
Method:	Least Squares	**F-statistic:**	27.76
Date:	Thu, 22 Sep 2022	**Prob (F-statistic):**	6.58e-10
Time:	20:19:38	**Log-Likelihood:**	12.720
No. Observations:	84	**AIC:**	-19.44
Df Residuals:	81	**BIC:**	-12.15
Df Model:	2		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	0.2960	0.417	0.710	0.480	-0.533	1.125
SAT	0.0017	0.000	7.432	0.000	0.001	0.002
Rand 1,2,3	-0.0083	0.027	-0.304	0.762	-0.062	0.046

Omnibus:	12.992	**Durbin-Watson:**	0.948
Prob(Omnibus):	0.002	**Jarque-Bera (JB):**	16.364
Skew:	-0.731	**Prob(JB):**	0.000280
Kurtosis:	4.594	**Cond. No.**	3.33e+04

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 3.33e+04. This might indicate that there are
strong multicollinearity or other numerical problems.