In [13]:

```
Alex Beckwith
Math 839 - Fall 21
HW 2

#Importing the packages used at the top of a script is standard practice.

import pandas as pd
import statsmodels.api as sm
```

In [14]:

```
#Problem 1
#Here, I load and display a small section of the data.

data = read_csv("HW2-data-table-B5.csv")
data.head()
```

Out[14]:

	y	x1	x2	x3	x4	x5	x6	x7
0	36.98	5.1	400	51.37	4.24	1484.83	2227.25	2.06
1	13.74	26.4	400	72.33	30.87	289.94	434.90	1.33
2	10.08	23.8	400	71.44	33.01	320.79	481.19	0.97
3	8.53	46.4	400	79.15	44.61	164.76	247.14	0.62
4	36.42	7.0	450	80.47	33.84	1097.26	1645.89	0.22

In [16]:

```
#Here, I'm organizing the data into a "regressor" DataFrame,
#adding a constant term, segmenting the dependent series,
#and fitting the model.

X = data.loc[:,["x6","x7"]]
X = sm.add_constant(X)
y = data.loc[:,"y"]
est = sm.OLS(y,X).fit()

#The output of this regression is used to analyze the following questions.
est.summary()
```

Out[16]:

OLS Regression Results

Dep. Variable:	y	**R-squared:**	0.700
Model:	OLS	**Adj. R-squared:**	0.675
Method:	Least Squares	**F-statistic:**	27.95
Date:	Tue, 05 Oct 2021	**Prob (F-statistic):**	5.39e-07
Time:	19:37:33	**Log-Likelihood:**	-98.686
No. Observations:	27	**AIC:**	203.4
Df Residuals:	24	**BIC:**	207.3
Df Model:	2		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	2.5265	3.610	0.700	0.491	-4.924	9.977
x6	0.0185	0.003	6.742	0.000	0.013	0.024

x7	2.1858	0.973	2.247	0.034	0.178	4.193

Omnibus:	1.544	**Durbin-Watson:**	2.332
Prob(Omnibus):	0.462	**Jarque-Bera (JB):**	0.466
Skew:	0.060	**Prob(JB):**	0.792
Kurtosis:	3.632	**Cond. No.**	2.26e+03

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 2.26e+03. This might indicate that there are
strong multicollinearity or other numerical problems.

In [17]:

```
#a
"""
The estimated model is given by:
y(x6,x7) = 2.5265 + 0.0185(x6) + 2.1858(x7)
"""

#b
"""
H0: The predictiveness of the regression is not statistically significant.
Ha: The predictiveness of the regression is statistically significant.

Due to the p value of the F-statistic being below a significance level of 0.05 (p = 5.39e-7),
it is reasonable to reject the null hypothesis in favor of the alternative,
that the predictiveness of the regression is statistically significant.
"""

#c
"""
R-squared = 0.700
R-squared adj = 0.675
The x6 and x7 factors can be said to be 70% correlated with y, or 67.5% when adjusted for the
number of factors.
"""

#d
"""
H0: The x6 and x7 factors do not have a significant relationship with y.
Ha: The x6 and x7 factors do have a significant relationship with y.

Due to the p values associated with x6 and x7 both being below a significance value of 0.05,
(x6:0.000, x7:0.034)
It is reasonable to reject the null hypothesis in favor of the alternative,
that x6 and x7 do exhibit a significant impact on y.
"""

#e
"""
Here are the 95% confidence intervals.
x6: 0.013       0.024
x7: 0.178       4.193

The are significant because neither interval passes through zero.
This indicates that we are at least 95% certain that the slopes have some impact.
A zero slope would indicate zero impact, which our analysis indicates is unlikely.
"""
```

Out[17]:
In [18]:

```
#I'm selecting inputs and refitting the regression here.

X = data.loc[:,"x6"]
X = sm.add_constant(X)
est = sm.OLS(y,X).fit()

est.summary()
```

Out[18]:

OLS Regression Results

Dep. Variable:	y	**R-squared:**	0.636
Model:	OLS	**Adj. R-squared:**	0.622
Method:	Least Squares	**F-statistic:**	43.77
Date:	Tue, 05 Oct 2021	**Prob (F-statistic):**	6.24e-07
Time:	19:37:53	**Log-Likelihood:**	-101.26
No. Observations:	27	**AIC:**	206.5
Df Residuals:	25	**BIC:**	209.1
Df Model:	1		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	6.1442	3.483	1.764	0.090	-1.029	13.318
x6	0.0194	0.003	6.616	0.000	0.013	0.025

Omnibus:	3.431	**Durbin-Watson:**	1.978
Prob(Omnibus):	0.180	**Jarque-Bera (JB):**	2.267
Skew:	-0.017	**Prob(JB):**	0.322
Kurtosis:	4.419	**Cond. No.**	2.01e+03

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 2.01e+03. This might indicate that there are
strong multicollinearity or other numerical problems.

In [19]:

```
#f
"""
y(x6) = 6.1442 + 0.0194(x6)

H0: The predictiveness of the regression is not statistically significant.
Ha: The predictiveness of the regression is statistically significant.

Due to the p value of the F-statistic being below a significance level of 0.05 (p = 6.24e-7),
it is reasonable to reject the null hypothesis in favor of the alternative,
that the predictiveness of the regression is statistically significant.
"""

#g1
"""
R-squared = 0.636
R-squared adj = 0.622
```

```
The x6 factor can be said to be 63.6% correlated with y, or 62.2% when adjusted for the
number of factors.
"""

#g2
"""
95% Confidence Interval
just x6-    x6:0.013    0.025
with x7-    x6: 0.013   0.024

There is a hair more certainty when x7 is applied to the model.
"""

#h
"""
The f-statistic is proportional to our MS-Res.
The F-statistic rose from 27.95 to 43.77 when we removed x7 from the model.
This indicates that x7 contributed positively to the accuracy of the model.
"""
```

Out[19]:

In [22]:

```
#Problem 2
#Time to load a different dataset.

wines = read_csv("HW2-data-table-B11.csv")
wines.head()
```

Out[22]:

	Clarity	Aroma	Body	Flavor	Oakiness	Quality	Region
0	1.0	3.3	2.8	3.1	4.1	9.8	1
1	1.0	4.4	4.9	3.5	3.9	12.6	1
2	1.0	3.9	5.3	4.8	4.7	11.9	1
3	1.0	3.9	2.6	3.1	3.6	11.1	1
4	1.0	5.6	5.1	5.5	5.1	13.3	1

In [24]:

```
#Here I'm pulling the quality columns out and adding a constant term.
q = "Quality"
y = wines.loc[:,q]
X = wines[wines.columns.difference([q])]
X = sm.add_constant(X)

#Here's where we build our model.
sommelier = sm.OLS(y,X).fit()
sommelier.summary()
```

Out[24]:

OLS Regression Results

Dep. Variable:	Quality	**R-squared:**	0.721
Model:	OLS	**Adj. R-squared:**	0.667
Method:	Least Squares	**F-statistic:**	13.33
Date:	Tue, 05 Oct 2021	**Prob (F-statistic):**	2.04e-07
Time:	19:48:25	**Log-Likelihood:**	-56.370
No. Observations:	38	**AIC:**	126.7
Df Residuals:	31	**BIC:**	138.2

Df Model:	6
Covariance Type:	nonrobust

	coef	std err	t	P>|t|	[0.025	0.975]
const	3.9843	2.270	1.755	0.089	-0.645	8.613
Aroma	0.4973	0.305	1.629	0.114	-0.125	1.120
Body	0.2784	0.341	0.817	0.420	-0.417	0.974
Clarity	2.3475	1.764	1.331	0.193	-1.249	5.944
Flavor	1.1699	0.310	3.779	0.001	0.538	1.801
Oakiness	-0.6923	0.285	-2.431	0.021	-1.273	-0.111
Region	-0.0338	0.296	-0.114	0.910	-0.637	0.569

Omnibus:	1.020	**Durbin-Watson:**	0.845
Prob(Omnibus):	0.601	**Jarque-Bera (JB):**	0.911
Skew:	-0.357	**Prob(JB):**	0.634
Kurtosis:	2.745	**Cond. No.**	137.

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

In [ ]:

```
#a
"""
y(Aroma,Body,Clarity,Flavor,Oakiness,Region) = 3.9843 + 0.4973(Aroma) + 0.2784(Body)
                                               + 2.3475(Clarity) + 1.1699(Flavor)
                                               - 0.6923(Oakiness) - 0.0338(Region)
"""

#b
"""
H0: The predictiveness of the regression is not statistically significant.
Ha: The predictiveness of the regression is statistically significant.

Due to the p value of the F-statistic being below a significance level of 0.05 (p = 2.04e-7),
it is reasonable to reject the null hypothesis in favor of the alternative,
that the predictiveness of the regression is statistically significant.
"""

#c
"""
regressors = (Aroma,Body,Clarity,Flavor,Oakiness,Region)
For each regressor:
    H0 = The regressor does not contribute significantly to the model
    Ha = The regressor does contribute significantly to the model

Aroma   0.114
Body    0.420
Clarity 0.193
Flavor  0.001
Oakiness        0.021
Region  0.910

Using a significance level of 0.05,
```

```
Flavor and Oakiness are the only regressors meeting the threshold to reject the null
hypothesis.
The others are assumed to not contribute significantly to the model.
The responsible thing to do would be to rerun the model iteratively,
removing regressors one at a time until all are sufficiently significant.
"""
```

In [ ]:

```
#d
"""
With all:
R-sq = 0.721
R-sq-a = 0.667

With just Flavor & Aroma:
R-sq = 0.659
R-sq-a = 0.639

With fewer variables, the model is less predictive.
I'd be curious to see how R-sq would change if the less predictive regressors were
programatically removed.
"""

#e
"""
first | const:-0.645    8.613
second | const:2.298    6.395
The aggregate error of regressors seems to contribute to the wideness of CIs.
Less predictive regressors likely lead to wider confidence intervals.
"""
```

In [26]:

```
#Slicing and refitting

X = wines.loc[:,["Flavor","Aroma"]]
X = sm.add_constant(X)
sommelier = sm.OLS(y,X).fit()
sommelier.summary()
```

Out[26]:

OLS Regression Results

Dep. Variable:	Quality	**R-squared:**	0.659
Model:	OLS	**Adj. R-squared:**	0.639
Method:	Least Squares	**F-statistic:**	33.75
Date:	Tue, 05 Oct 2021	**Prob (F-statistic):**	6.81e-09
Time:	20:40:33	**Log-Likelihood:**	-60.188
No. Observations:	38	**AIC:**	126.4
Df Residuals:	35	**BIC:**	131.3
Df Model:	2		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	4.3462	1.009	4.307	0.000	2.298	6.395
Flavor	1.1702	0.291	4.027	0.000	0.580	1.760
Aroma	0.5180	0.276	1.877	0.069	-0.042	1.078

Omnibus:	0.321	**Durbin-Watson:**	0.869

Prob(Omnibus):	0.852	**Jarque-Bera (JB):**	0.499
Skew:	0.076	**Prob(JB):**	0.779
Kurtosis:	2.460	**Cond. No.**	35.8

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

In [30]:

```
#I followed the process to find a different final regression model.

X = wines.loc[:,["Aroma","Flavor","Oakiness"]]
X = sm.add_constant(X)
sommelier = sm.OLS(y,X).fit()
sommelier.summary()
```

Out[30]:

OLS Regression Results

Dep. Variable:	Quality	**R-squared:**	0.704
Model:	OLS	**Adj. R-squared:**	0.678
Method:	Least Squares	**F-statistic:**	26.92
Date:	Tue, 05 Oct 2021	**Prob (F-statistic):**	4.20e-09
Time:	20:46:19	**Log-Likelihood:**	-57.489
No. Observations:	38	**AIC:**	123.0
Df Residuals:	34	**BIC:**	129.5
Df Model:	3		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	6.4672	1.333	4.852	0.000	3.759	9.176
Aroma	0.5801	0.262	2.213	0.034	0.047	1.113
Flavor	1.1997	0.275	4.364	0.000	0.641	1.758
Oakiness	-0.6023	0.264	-2.278	0.029	-1.140	-0.065

Omnibus:	0.955	**Durbin-Watson:**	0.837
Prob(Omnibus):	0.620	**Jarque-Bera (JB):**	0.964
Skew:	-0.338	**Prob(JB):**	0.618
Kurtosis:	2.611	**Cond. No.**	58.6

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

In [ ]: