1. **(35%) Linear Regression Analysis for Wine Quality**

(a) (10%) Show the results of regression analysis as follows

MDS_Assignment1 > hw1_q1_sm.txt

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                  grade   R-squared:                       0.495
Model:                            OLS   Adj. R-squared:                  0.472
Method:                 Least Squares   F-statistic:                     21.52
Date:                Tue, 04 Oct 2022   Prob (F-statistic):           1.16e-70
Time:                        22:54:25   Log-Likelihood:                -381.52
No. Observations:                 620   AIC:                             819.0
Df Residuals:                     592   BIC:                             943.1
Df Model:                          27
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Intercept      2.0339      0.018    110.528      0.000       1.998       2.070
f0            -0.0055      0.025     -0.224      0.823      -0.054       0.043
f1             0.0440      0.028      1.571      0.117      -0.011       0.099
f2             0.3140      0.035      9.028      0.000       0.246       0.382
f3             0.0186      0.042      0.447      0.655      -0.063       0.100
f4            -0.0035      0.038     -0.091      0.928      -0.078       0.072
f5            -0.0740      0.030     -2.483      0.013      -0.133      -0.015
f6            -0.0710      0.026     -2.742      0.006      -0.122      -0.020
f7             0.0235      0.028      0.853      0.394      -0.031       0.078
f8             0.0410      0.019      2.170      0.030       0.004       0.078
f9          2.877e-17   3.57e-17      0.807      0.420   -4.13e-17    9.88e-17
f10           -0.0446      0.021     -2.119      0.035      -0.086      -0.003
f11           -0.0292      0.020     -1.438      0.151      -0.069       0.011
f12           -0.0006      0.022     -0.027      0.979      -0.044       0.043
f13            0.0336      0.024      1.412      0.159      -0.013       0.080
f14           -0.1832      0.021     -8.898      0.000      -0.224      -0.143
f15           -0.1061      0.019     -5.565      0.000      -0.144      -0.069
f16           -0.0358      0.020     -1.756      0.080      -0.076       0.004
f17            0.0633      0.019      3.409      0.001       0.027       0.100
f18           -0.1904      0.021     -9.194      0.000      -0.231      -0.150
f19            0.0278      0.026      1.051      0.294      -0.024       0.080
f20            0.0126      0.020      0.644      0.520      -0.026       0.051
f21           -0.0357      0.028     -1.263      0.207      -0.091       0.020
f22            0.0747      0.021      3.533      0.000       0.033       0.116
f23           -0.0088      0.020     -0.442      0.659      -0.048       0.030
f24            0.0193      0.024      0.800      0.424      -0.028       0.067
f25           -0.0679      0.020     -3.406      0.001      -0.107      -0.029
f26           -0.0360      0.022     -1.625      0.105      -0.080       0.008
f27           -0.0062      0.019     -0.324      0.746      -0.044       0.031
==============================================================================
Omnibus:                       39.669   Durbin-Watson:                   2.004
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              147.525
Skew:                           0.090   Prob(JB):                     9.23e-33
Kurtosis:                       5.383   Cond. No.                     1.34e+16
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is 1.13e-29. This might indicate that there are
strong multicollinearity problems or that the design matrix is singular.
```

(b) (5%) The fitting of the linear regression is a good idea? If yes, why? If no, why? What's the possible reason of poor fitting?

Ans:

使用以下兩指標解讀線性分析的結果。

R-squared: the measurement of how much of the independent variable is explained by changes in our dependent variables.

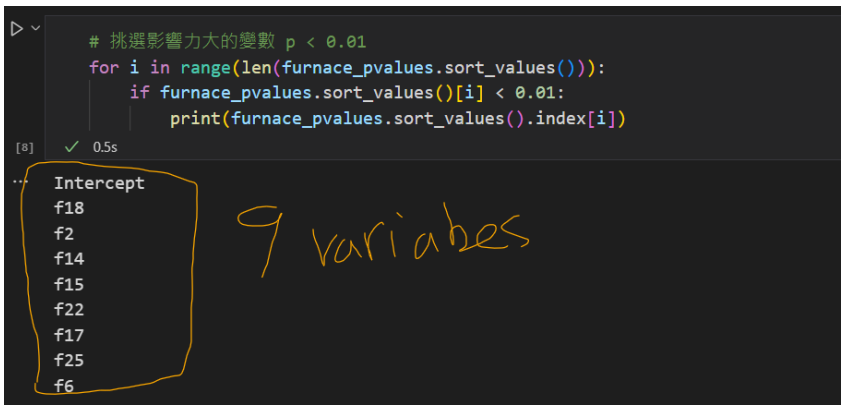
- 統計結果的 R-squared 是 0.495，表示模型能解釋變數們 49.5% 的變量。R-squared < 0.5 指出此模型表現不佳。
- Adj. R-squared 的結果雷同。

Prob (F-Statistic): this number to tell you the accuracy of the null hypothesis(H0), or whether it is accurate that your variables' effect is 0.

- H0 是殘差正常的假設。
- 統計結果的 `Prob (F-Statistic)` 很小，為 1.16e-70，因此我們拒絕此假設，也表示此數據不太適合做 linear regression。

(c) Based on the results, rank the independent variables by p values and which one are statistically significant variables with p values<0.01?

Ans: p 越小影響力越大，如下程式運行的結果，依照小至大排序為 f18 < f2 < … < f6。



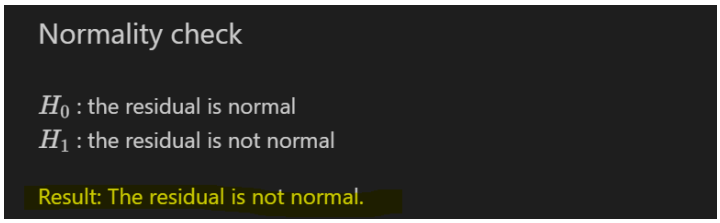
(d) (15%) Testify the underlying assumptions of regression (1) Normality, (2) Independence, and (3) Homogeneity of Variance with respect to residual

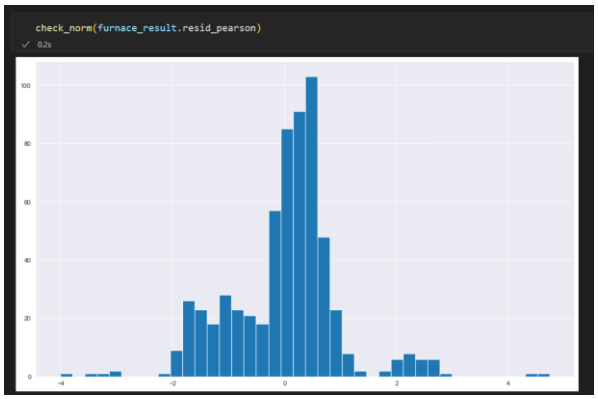
Ans:

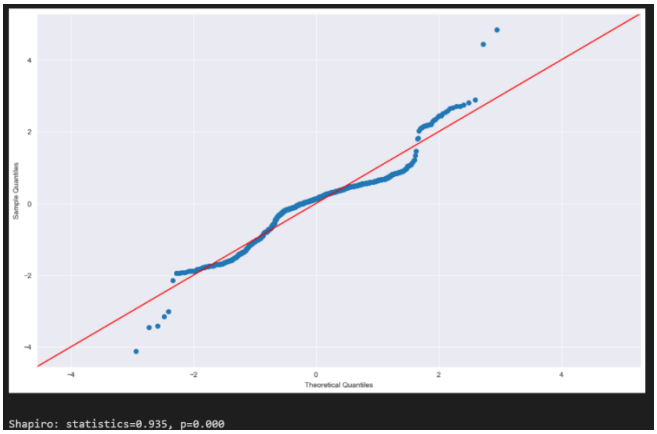
- Normality: Shapiro-Wilk test 常態性檢定 => Not normal
- Independence: Durbin-Watson test 獨立性檢定 => Independent, features 間關聯性低
- Homogeneity: Breusch-Pagan test 異質變異數檢定 => No Homogeneity

(一) Normality Check

The residual is not normal. Because `Shapiro: statistics=0.935, p=0.000`. Result p-value < 0.05 so we reject H0.







(二) Independence Check (aka check multicollinearity)

None of the VIF (Variance Inflation Factor) value of the features > 10, therefore, no significant multicolinearity within the variables. No features should be removed.

```
from statsmodels.stats.outliers_influence import variance_inflation_factor

vif = [variance_inflation_factor(furnace_X_const.values, i) for i in range(furnace_X_const.shape[1])]
pd.DataFrame({'vif': vif[1:]}, index=furnace_X.columns).T
```

0.6s Python

```
c:\Users\Weber\.conda\envs\testAI\lib\site-packages\statsmodels\regression\linear_model.py:1736: RuntimeWarning: invalid value
encountered in double_scalars
  return 1 - self.ssr/self.centered_tss
```

	f0	f1	f2	f3	f4	f5	f6	f7	f8	f9	...	f18	f19	f20	f21	f22
vif	1.81784	2.314364	3.571219	5.095433	4.30931	2.623037	1.983188	2.24362	1.054478	NaN	...	1.266249	2.060656	1.12521	2.356358	1.319547

1 rows × 28 columns

(三) Homogeneity Check (aka Homoscedasticity)

B-P test shows hetroscedasticity, G-Q test shows homoscedasticity.

B-P test reject null hypothesis with p-value = $5.3e^{-20} < 0.5$. Which means residual doesn't show homoscedasticity. G-Q test result a p-value = 0.42. Although it less than 0.5 a little bit. It is more acceptable than the result of B-P test.

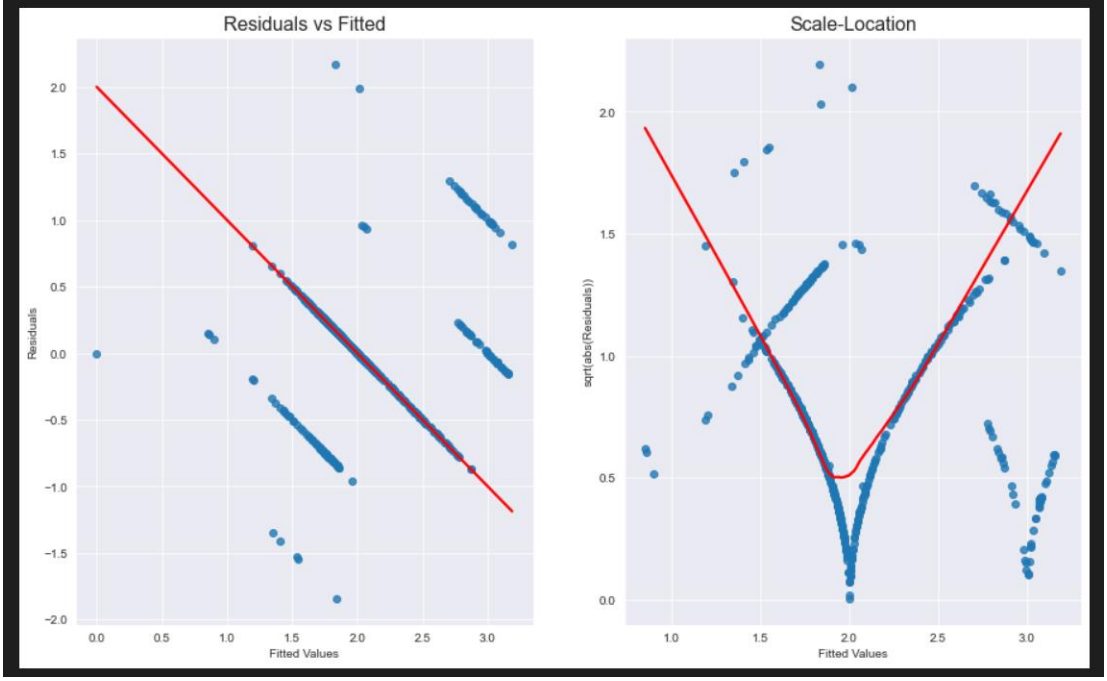
Check Homoscedasticity assumption

H_0 : Homoscedasticity
H_1 : Hetroscedasticity

```
 Breusch-Pagan test ----
                                    value
Lagrange multiplier statistic  1.406203e+02
p-value                        5.871843e-17
f-value                        6.431711e+00
f p-value                      5.311816e-20

 Goldfeld-Quandt test ----
                 value
F statistic  1.024212
p-value      0.420429
```

```
Residuals plots ----
```



2. (35%) Association Rule Market Basket Analysis

- ✓ **Set the minimum support to 0.001**
- ✓ **Set the minimum confidence of 0.15**

(1) (10%) How to handle the raw dataset via data preprocessing?

Ans: Make each transaction record in to Boolean vector, so the dataset is a Boolean matrix. Item bought indicates the corresponding column is True, otherwise False.

```
from mlxtend.preprocessing import TransactionEncoder
te = TransactionEncoder()
te_ary = te.fit(trans).transform(trans)
trans_bool = pd.DataFrame(te_ary, columns=te.columns_)
trans_bool
```

products

	Instant food products	UHT-milk	abrasive cleaner	artif. sweetener	baby cosmetics	baby food	bags	baking powder	bathroom cleaner	beef	...	turkey	vinegar	waffles	whipped/sour cream	whisky
0	False	False	False	False	False	False	False	False	False	False	...	False	False	False	False	False
1	False	False	False	False	False	False	False	False	False	False	...	False	False	False	False	False
2	False	False	False	False	False	False	False	False	False	False	...	False	False	False	False	False
3	False	False	False	False	False	False	False	False	False	False	...	False	False	False	False	False
4	False	False	False	False	False	False	False	False	False	False	...	False	False	False	False	False
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
9830	False	False	False	False	False	False	False	False	False	True	...	False	False	False	True	False
9831	False	False	False	False	False	False	False	False	False	False	...	False	False	False	False	False
9832	False	False	False	False	False	False	False	False	False	False	...	False	False	False	False	False
9833	False	False	False	False	False	False	False	False	False	False	...	False	False	False	False	False
9834	False	False	False	False	False	False	False	False	False	False	...	False	True	False	False	False

9835 rows × 171 columns

(2) (10%) What's the top 5 association rules? Show the support, confidence, and lift to each specific rule, respectively?

Ans:

All the 5 rules show that "whole milk" and "other vegetables" is a great consequent product. The most frequent rule is {root vegetables, flour, whipped/sour cream → whole milk}. Its

support value is 0.001729, which means this itemset buying ratio.

All the confidence is 1.0 which means these are strong causal relationship.
Lift > 1 means these rules appear more frequent than normal case of consequent.

Pick top 5 rules, pick by confidence in order.

```
conf_rules = rules[rules['confidence'] >= 0.15].sort_values(by=['confidence', 'antecedent support'], ascending=False)
conf_rules[:5]
```

✓ 0.6s Python

{A→ B}

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction
73399	(root vegetables, flour, whipped/sour cream)	(whole milk)	0.001729	0.255516	0.001729	1.0	3.913649	0.001287	inf
99065	(oil, root vegetables, other vegetables, yogurt)	(whole milk)	0.001423	0.255516	0.001423	1.0	3.913649	0.001060	inf
44771	(rice, sugar)	(whole milk)	0.001220	0.255516	0.001220	1.0	3.913649	0.000908	inf
92855	(whipped/sour cream, other vegetables, butter,...)	(whole milk)	0.001220	0.255516	0.001220	1.0	3.913649	0.000908	inf
95133	(tropical fruit, root vegetables, whipped/sour...)	(other vegetables)	0.001220	0.193493	0.001220	1.0	5.168156	0.000984	inf

(3) (5%) Please provide/guess the "story" to interpret one of top 5 rules you are interested in.

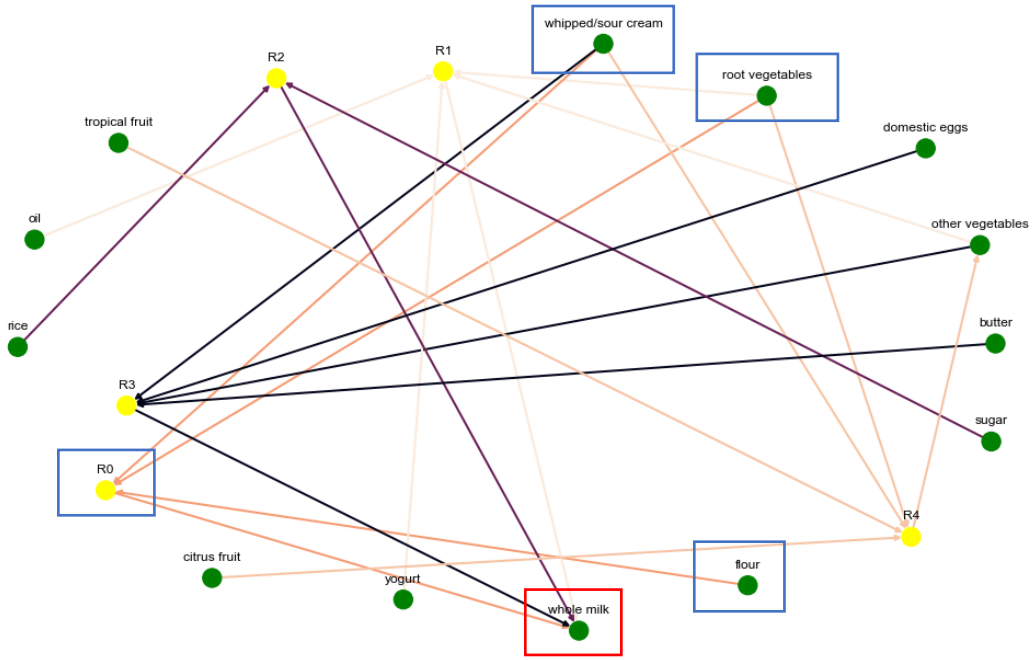
Ans:

The first rule is {root vegetables, flour, whipped/sour cream → whole milk} may tell us that people tend to cook bisque soup. As well as the vegetables and flour, milk is a necessary ingredient for bisque soup.

(4) (10%) Give a visualization graph of your association rules.

Ans:

For the clearness, I show the graph from top 5 rules. We can find an interesting causal relationship: most edges point from "Rule node" to "whole milk" and "other vegetables".



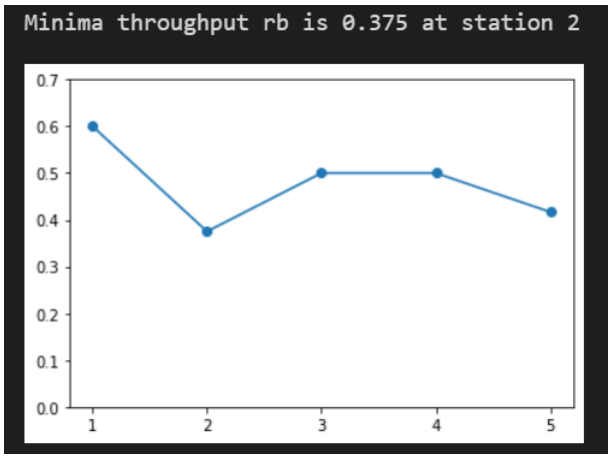
3. (30%) Manufacturing System Analysis

(a) (10%)根據 Little's Law 試 計算各工作站的產出率 TH 於下表，試問瓶頸站的產出率 rb、最小生產週期時間 (總加工時間， $T0$)、關鍵在製品水準 ($W0$)各為多少？

Ans: $rb = 0.375$, $T0 = 41$, $W0 = 15.375$.

($W0 = rb * T0$)

工作站編號	機台數	加工時間 (小時)	工作站的產能 TH (個/小時)
1	3	5	
2	3	8	
3	6	12	
4	2	4	
5	5	12	



(b) (10%)試給出最佳績效 (best case)下，最大的產出率 (THbest)與最小生產週期時間 (CTbest)的計算公式 (提示講義 22~29 頁)

Ans: 如下方程式碼。

```
# critical WIP's param
rb = 0.375
T0 = 41
W0 = 15.375

def THbest(w):
    '''最大產出率'''
    if w <= W0:
        return w/T0 # 投料過少
    return rb # 投料適中or過多，TH為瓶頸站產量

def CTbest(w):
    ''' 最小生產週期時間 '''
    if w <= W0:
        return T0 # 最短周期
    return w/rb # 投量過多，TH為瓶頸站產量，所需時間遞增
```

$$\mathbf{CT(w)} = \begin{cases} T_0 & , w \le W_0 \\ \dfrac{w}{r_b} & , w > W_0 \end{cases}$$

$$\mathbf{TH(w)} = \begin{cases} \dfrac{w}{T_0} & , w \le W_0 \\ r_b & , w > W_0 \end{cases}$$

(c) (10%)根據該問題的產線，試程式撰寫建立一模擬模型(或用套裝軟體、數值分析)來驗證，當在製品 WIP 數量超過工廠產能時，其生產週期將嚴重惡化。也就是當產線的投

料速度(投產量)大於產線的產出率，此時生產系統將處於非穩態的狀態(non steady state)。

試用圖表呈現 WIP、 CT 與 TH 之間惡化的關係。(提示講義 22~29 頁)

Ans: 如下圖，W0 時 WIP = 15.375 是 critical wip，為圖中的轉折點。