# Table of Contents



# Your Turn -- Activity II: Partial F-Test

In [2]:
```
import pandas as pd
shear_df = pd.read_excel('data/lect02-lin-reg.xlsx', sheet_name='Strength')
shear_df.head()
```

Out[2]:

| | Force | Power | Temperature | Time | Strength |
|---|---|---|---|---|---|
| **0** | 30 | 60 | 175 | 15 | 26.2 |
| **1** | 40 | 60 | 175 | 15 | 26.3 |
| **2** | 30 | 90 | 175 | 15 | 39.8 |
| **3** | 40 | 90 | 175 | 15 | 39.7 |
| **4** | 30 | 60 | 225 | 15 | 38.6 |

In [3]:
```
import statsmodels.api as sm

Y = shear_df.Strength
X = shear_df.drop('Strength', axis=1)
Xreg = sm.add_constant(X)
```

```
C:\Users\2543b\anaconda3\lib\site-packages\statsmodels\tsa\tsatools.py:142: FutureWarning: In a future version
of pandas all arguments of concat except for the argument 'objs' will be keyword-only
  x = pd.concat(x[::order], 1)
```

In [4]:
```
shear_lm = sm.OLS(Y, Xreg).fit()
```

In [5]:
```
# Fit by using formula
from statsmodels.formula.api import ols

formula = 'Strength ~ Temperature + Power'
#formula = 'Viscosity ~ Temperature + Catalyst - 1'

shear_lm = ols(formula, data=shear_df).fit()
```

In [6]:
```
shear_lm.summary2()
```

Out[6]:

| | | | |
|---|---|---|---|
| Model: | OLS | Adj. R-squared: | 0.662 |
| Dependent Variable: | Strength | AIC: | 186.9755 |
| Date: | 2021-09-02 11:13 | BIC: | 191.1791 |
| No. Observations: | 30 | Log-Likelihood: | -90.488 |
| Df Model: | 2 | F-statistic: | 29.38 |
| Df Residuals: | 27 | Prob (F-statistic): | 1.67e-07 |
| R-squared: | 0.685 | Scale: | 27.113 |

| | Coef. | Std.Err. | t | P>\|t\| | [0.025 | 0.975] |
|---|---|---|---|---|---|---|
| **Intercept** | -24.9017 | 10.0721 | -2.4723 | 0.0200 | -45.5678 | -4.2355 |
| **Temperature** | 0.1297 | 0.0425 | 3.0499 | 0.0051 | 0.0424 | 0.2169 |
| **Power** | 0.4983 | 0.0709 | 7.0328 | 0.0000 | 0.3529 | 0.6437 |

| | | | |
|---|---|---|---|
| Omnibus: | 0.289 | Durbin-Watson: | 2.351 |
| Prob(Omnibus): | 0.866 | Jarque-Bera (JB): | 0.086 |
| Skew: | -0.127 | Prob(JB): | 0.958 |
| Kurtosis: | 2.935 | Condition No.: | 2275 |

## จาก P-Value Temperature และ Power มีความ significant

In [7]:
```
# Fit by using formula
from statsmodels.formula.api import ols

formula2 = 'Strength ~ Temperature + Power + Force + Time '
#formula = 'Viscosity ~ Temperature + Catalyst - 1'

shear_lm2 = ols(formula2, data=shear_df).fit()
```

In [8]:
```
shear_lm2.summary2()
```

Out[8]:

| | | | |
|---|---|---|---|
| Model: | OLS | Adj. R-squared: | 0.668 |
| Dependent Variable: | Strength | AIC: | 188.0994 |
| Date: | 2021-09-02 11:13 | BIC: | 195.1053 |
| No. Observations: | 30 | Log-Likelihood: | -89.050 |
| Df Model: | 4 | F-statistic: | 15.60 |
| Df Residuals: | 25 | Prob (F-statistic): | 1.59e-06 |
| R-squared: | 0.714 | Scale: | 26.605 |

| | Coef. | Std.Err. | t | P>\|t\| | [0.025 | 0.975] |
|---|---|---|---|---|---|---|
| **Intercept** | -37.4767 | 13.0996 | -2.8609 | 0.0084 | -64.4559 | -10.4974 |
| **Temperature** | 0.1297 | 0.0421 | 3.0789 | 0.0050 | 0.0429 | 0.2164 |
| **Power** | 0.4983 | 0.0702 | 7.0997 | 0.0000 | 0.3538 | 0.6429 |
| **Force** | 0.2117 | 0.2106 | 1.0052 | 0.3244 | -0.2220 | 0.6454 |
| **Time** | 0.2583 | 0.2106 | 1.2268 | 0.2313 | -0.1754 | 0.6920 |

| | | | |
|---|---|---|---|
| Omnibus: | 1.712 | Durbin-Watson: | 2.261 |
| Prob(Omnibus): | 0.425 | Jarque-Bera (JB): | 1.156 |
| Skew: | -0.480 | Prob(JB): | 0.561 |
| Kurtosis: | 2.965 | Condition No.: | 3038 |

## จากผลจะเห็นได้ว่าเมื่อตัวแปรเพิ่มขึ้นจะทำให้ F-Statistic ลดลง ส่งผลให้ Predict ได้ดียิ่งขึ้น

## จาก P-Value Temperature และ Power มีความ significant

In [9]:
```
sse_k = shear_lm.ssr
sse = shear_lm2.ssr
mse = shear_lm2.mse_resid
```

In [10]:
```
f0 = ((sse_k - sse ) /2 )/mse
```

In [11]:
```
print(f0)
```

```
1.2577329559236514
```

In [12]:
```
from scipy import stats
stats.f.sf(f0, 2, 25)
```

Out[12]:
```
0.301674930195559
```

## Coefficient ของตัวแปรอื่นๆ ไม่ต่างจาก 0 ไม่ช่วยในการ Predict ค่า Y ได้ดียิ่งขึ้น

In [ ]:
```

```