Analysis of Covariance

In [17]:
```
data = pd.read_excel('lect06-ancova.xlsx', sheet_name='Filament')
data
```

Out[17]:

	Strength	Diameter	Machine
0	36	20	M1
1	41	25	M1
2	39	24	M1
3	42	25	M1
4	49	32	M1
5	40	22	M2
6	48	28	M2
7	39	22	M2
8	45	30	M2
9	44	28	M2
10	35	21	M3
11	37	23	M3
12	42	26	M3
13	34	21	M3
14	32	15	M3

In [24]:
```
def codingfunc(x):
    if x == 'M1':
        return (1,0)
    elif x == 'M2':
        return (0,1)
    else:
        return (-1,-1)

data[['I1', 'I2']] = pd.DataFrame(data['Machine'].apply(codingfunc).tolist(),
                                  index=data.index)
data.Diameter -= data.Diameter.mean()
```

In [25]:
```
data
```

Out[25]:

	Strength	Diameter	Machine	I1	I2
0	36	-4.133333	M1	1	0
1	41	0.866667	M1	1	0

	Strength	Diameter	Machine	I1	I2
2	39	-0.133333	M1	1	0
3	42	0.866667	M1	1	0
4	49	7.866667	M1	1	0
5	40	-2.133333	M2	0	1
6	48	3.866667	M2	0	1
7	39	-2.133333	M2	0	1
8	45	5.866667	M2	0	1
9	44	3.866667	M2	0	1
10	35	-3.133333	M3	-1	-1
11	37	-1.133333	M3	-1	-1
12	42	1.866667	M3	-1	-1
13	34	-3.133333	M3	-1	-1
14	32	-9.133333	M3	-1	-1

In [26]:
```
from statsmodels.formula.api import ols
from statsmodels.stats.anova import anova_lm


formula = 'Strength ~ C(Machine)'


lm = ols(formula, data=data).fit()
aov_table = anova_lm(lm)
aov_table
```

Out[26]:

	df	sum_sq	mean_sq	F	PR(>F)
C(Machine)	2.0	140.4	70.200000	4.08932	0.044232
Residual	12.0	206.0	17.166667	NaN	NaN

ค่า p-value = 0.044 ซึ่งมีค่าน้อยกว่า 0.05 สรุปได้ว่า Machine ส่งผลต่อ Strength

In [27]:
```
from statsmodels.formula.api import ols

formula = 'Strength ~ I1 + I2 + Diameter'
lm = ols(formula, data=data).fit()
lm.summary2()
```

Out[27]:

Model:	OLS	Adj. R-squared:	0.897
Dependent Variable:	Strength	AIC:	59.9229
Date:	2021-10-14 18:58	BIC:	62.7551

No. Observations:	15	Log-Likelihood:	-25.961
Df Model:	3	F-statistic:	41.72
Df Residuals:	11	Prob (F-statistic):	2.66e-06
R-squared:	0.919	Scale:	2.5442

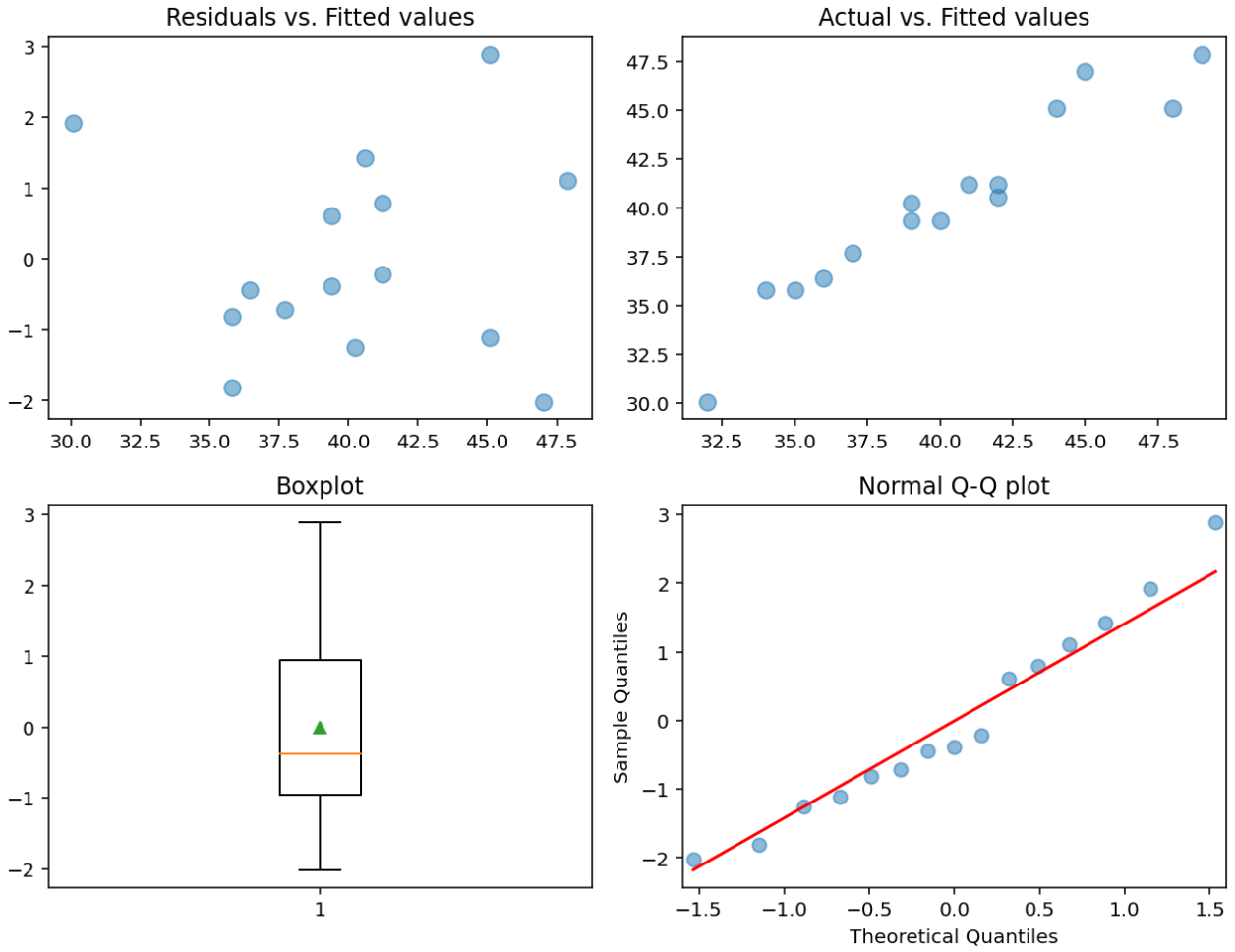
	Coef.	Std.Err.	t	P>|t|	[0.025	0.975]
Intercept	40.2000	0.4118	97.6109	0.0000	39.2935	41.1065
I1	0.1824	0.5950	0.3066	0.7649	-1.1272	1.4920
I2	1.2192	0.6201	1.9661	0.0750	-0.1456	2.5841
Diameter	0.9540	0.1140	8.3648	0.0000	0.7030	1.2050

Omnibus:	0.909	Durbin-Watson:	1.931
Prob(Omnibus):	0.635	Jarque-Bera (JB):	0.818
Skew:	0.463	Prob(JB):	0.664
Kurtosis:	2.327	Condition No.:	7

ค่า p-value ของ I1 and I2 ไม่significant หมายความว่า เมื่อเพิ่ม covariate ทำให้ factor machines ไม่ significant ทำให้ machine ไม่ส่งผลต่อ Strength

In [30]:
```
def check_linreg_residuals(model):

    kws = dict(color='blue', marker='o', markersize=7, alpha=0.5)
    sns1_kws = dict(marker='o', s=70, alpha=0.5)
    sns2_kws = dict(marker='o', s=7, alpha=0.5)

    x = model.resid
    yhat = model.fittedvalues

    fig = plt.figure(figsize=(9, 7))

    ax1 = fig.add_subplot(221)
    ax2 = fig.add_subplot(222)
    ax3 = fig.add_subplot(223)
    ax4 = fig.add_subplot(224)

    ax1.scatter(yhat, x, **sns1_kws)
    ax1.set_title('Residuals vs. Fitted values')

    n=len(x)
    #sns.residplot(np.linspace(1,n,n), x, ax=ax2, scatter_kws=sns1_kws)
    #ax2.set_title('Sequence plot')
    ax2.scatter(yhat+x, yhat, **sns1_kws)
    ax2.set_title('Actual vs. Fitted values')

    # Box plot
    ax3.boxplot(x,showmeans=True)
    ax3.set_title('Boxplot')
```

```
    # qq plot
    sm.graphics.qqplot(x,line='q', ax=ax4, **kws)
    ax4.set_title('Normal Q-Q plot')
    plt.tight_layout()
check_linreg_residuals(lm)
```



In [36]:

```
from statsmodels.formula.api import ols

formula = 'Strength ~ Diameter'
fila_reduce1_lm = ols(formula, data=data).fit()
fila_reduce1_lm.summary2()
```

Out[36]:

Model:	OLS	Adj. R-squared:	0.872
Dependent Variable:	Strength	AIC:	61.7493
Date:	2021-10-14 19:09	BIC:	63.1654
No. Observations:	15	Log-Likelihood:	-28.875
Df Model:	1	F-statistic:	96.12
Df Residuals:	13	Prob (F-statistic):	2.26e-07
R-squared:	0.881	Scale:	3.1746

	Coef.	Std.Err.	t	P>|t|	[0.025	0.975]

Intercept	40.2000	0.4600	87.3831	0.0000	39.2061	41.1939
Diameter	1.0797	0.1101	9.8039	0.0000	0.8418	1.3177

Omnibus:	0.495	Durbin-Watson:	1.387
Prob(Omnibus):	0.781	Jarque-Bera (JB):	0.362
Skew:	0.331	Prob(JB):	0.834
Kurtosis:	2.623	Condition No.:	4

In [37]:
```
print('SSE_k (reduced model): {:.2f} '.format(fila_reduce1_lm.ssr))
print('SSE (full model): {:.2f} '.format(fila_reduce1_lm.ssr))
print('MSE (full model): {:.2f} '.format(fila_reduce1_lm.mse_resid))
```

```
SSE_k (reduced model): 41.27
SSE (full model): 41.27
MSE (full model): 3.17
```

In [38]:
```
from scipy import stats
k=2
partial_F0 = ((fila_reduce1_lm.ssr-lm.ssr)/k)/lm.mse_resid
print('Test statistic: {:.2f}, P-value: {:.4f}'.format(partial_F0,
                                                       stats.f.sf(partial_F0, k, lm.df_
print('Critical value at 0.05: {:.2f} '.format(stats.f.isf(0.05, k, lm.df_resid)))
```

```
Test statistic: 2.61, P-value: 0.1181
Critical value at 0.05: 3.98
```

In [40]:
```
formula = 'Strength ~ I1+I2'
fila_reduce2_lm = ols(formula, data=data).fit()
fila_reduce2_lm.summary2()
```

Out[40]:

Model:	OLS	Adj. R-squared:	0.306
Dependent Variable:	Strength	AIC:	87.8655
Date:	2021-10-14 19:10	BIC:	89.9897
No. Observations:	15	Log-Likelihood:	-40.933
Df Model:	2	F-statistic:	4.089
Df Residuals:	12	Prob (F-statistic):	0.0442
R-squared:	0.405	Scale:	17.167

	Coef.	**Std.Err.**	**t**	**P>|t|**	**[0.025**	**0.975]**
Intercept	40.2000	1.0698	37.5776	0.0000	37.8691	42.5309
I1	1.2000	1.5129	0.7932	0.4431	-2.0963	4.4963
I2	3.0000	1.5129	1.9829	0.0707	-0.2963	6.2963

Omnibus:	1.225	Durbin-Watson:	2.359
Prob(Omnibus):	0.542	Jarque-Bera (JB):	1.000

Skew:	0.553	Prob(JB):	0.607
Kurtosis:	2.385	Condition No.:	2

In [41]:
```
print('SSE_k (reduced model): {:.2f} '.format(fila_reduce2_lm.ssr))
print('SSE (full model): {:.2f} '.format(fila_reduce2_lm.ssr))
print('MSE (full model): {:.2f} '.format(fila_reduce2_lm.mse_resid))
```

```
SSE_k (reduced model): 206.00
SSE (full model): 206.00
MSE (full model): 17.17
```

In [42]:
```
k=2
partial_F0 = ((fila_reduce2_lm.ssr-lm.ssr)/k)/lm.mse_resid
print('Test statistic: {:.2f}, P-value: {:.4f}'.format(partial_F0,
                                                       stats.f.sf(partial_F0, k, lm.df_
print('Critical value at 0.05: {:.2f} '.format(stats.f.isf(0.05, k, lm.df_resid)))
```

```
Test statistic: 34.98, P-value: 0.0000
Critical value at 0.05: 3.98
```

จาก Patial F-test, P-value ของ Diameter = 0.11 ซึ่งสรุปได้ว่า Diameter ไม่ significant ได้ว่า Diameter ไม่ส่งผลต่อ Strength

In [ ]:
```

```