# Chapter 5

# Simple Linear regression (solutions to exercises)

# Contents

**5 Simple Linear regression (solutions to exercises) 1**

5.1 Plastic film folding machine . . . . . . . . . . . . . . . . . . . . . . . 4

5.2 Linear regression life time model . . . . . . . . . . . . . . . . . . . 6

5.3 Yield of chemical process . . . . . . . . . . . . . . . . . . . . . . . . 9

5.4 Plastic material . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 13

5.5 Water polution . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 16

5.6 Membrane pressure drop . . . . . . . . . . . . . . . . . . . . . . . 20

5.7 Membrane pressure drop (matrix form) . . . . . . . . . . . . . . . 24

5.8 Independence and correlation . . . . . . . . . . . . . . . . . . . . . 27

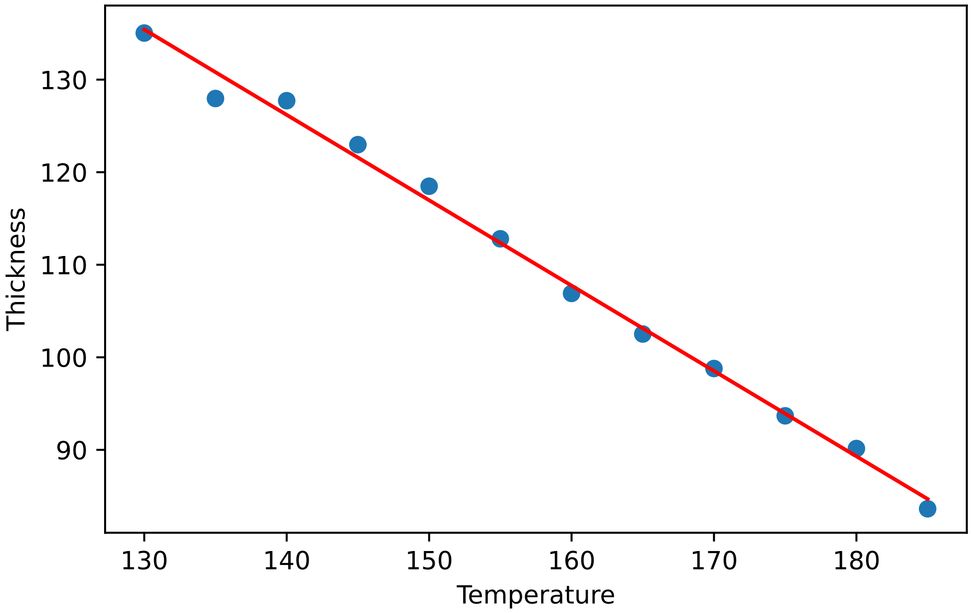
# Import Python packages

```
# Import all needed python packages
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
import scipy.stats as stats
import statsmodels.formula.api as smf
import statsmodels.api as sm
```

## 5.1 Plastic film folding machine

### |||| Exercise 5.1 Plastic film folding machine

On a machine that folds plastic film the temperature may be varied in the range of 130-185 °C. For obtaining, if possible, a model for the influence of temperature on the folding thickness, $n = 12$ related set of values of temperature and the fold thickness were measured that is illustrated in the following figure:



a) Determine by looking at the figure, which of the following sets of estimates for the parameters in the usual regression model is correct:

1) $\hat{\beta}_0 = 0$, $\hat{\beta}_1 = -0.9$, $\hat{\sigma} = 36$

2) $\hat{\beta}_0 = 0$, $\hat{\beta}_1 = 0.9$, $\hat{\sigma} = 3.6$

3) $\hat{\beta}_0 = 252$, $\hat{\beta}_1 = -0.9$, $\hat{\sigma} = 3.6$

4) $\hat{\beta}_0 = -252$, $\hat{\beta}_1 = -0.9$, $\hat{\sigma} = 36$

5) $\hat{\beta}_0 = 252$, $\hat{\beta}_1 = -0.9$, $\hat{\sigma} = 36$

### |||| Solution

First of all, the only possible intercept ($\hat{\beta}_0$) among the ones given in the answers is 252. And then the slope estimate of -0.9 in these two options looks reasonable. We just need to decide on whether the estimated standard deviation of the error $s_e = \hat{\sigma}$ is 3.6 or 36. From the figure it is clear that the points are NOT having an average vertical distance to the line in the size of 36, so 3.6 must be the correct number and hence the correct answer is:

3 ) $\hat{\beta}_0 = 252, \hat{\beta}_1 = -0.9, \hat{\sigma} = 3.6$

b) What is the only possible correct answer:

1) The proportion of explained variation is 50% and the correlation is 0.98
2) The proportion of explained variation is 0% and the correlation is $-0.98$
3) The proportion of explained variation is 96% and the correlation is $-1$
4) The proportion of explained variation is 96% and the correlation is 0.98
5) The proportion of explained variation is 96% and the correlation is $-0.98$

### |||| Solution

The proportion of variation explained must be pretty high, so 0 can be ruled out. Answer 1 and 4 is also ruled out since the correlation clearly is negative. This also narrows the possibilities down to answer 3 and 5. And since the correlation is NOT exactly -1 (in which case the observations would be exactly on the line), the correct answer is:

5) The proportion of explained variation is 96% and the correlation is $-0.98$

## 5.2 Linear regression life time model

### |||| Exercise 5.2 Linear regression life time model

A company manufactures an electronic device to be used in a very wide temperature range. The company knows that increased temperature shortens the life time of the device, and a study is therefore performed in which the life time is determined as a function of temperature. The following data is found:

| Temperature in Celcius (t) | 10 | 20 | 30 | 40 | 50 | 60 | 70 | 80 | 90 |
|---|---|---|---|---|---|---|---|---|---|
| Life time in hours (y) | 420 | 365 | 285 | 220 | 176 | 117 | 69 | 34 | 5 |

a) Calculate the 95% confidence interval for the slope in the usual linear regression model, which expresses the life time as a linear function of the temperature.

### |||| Solution

Either one could do all the regression computations to find the $\hat{\beta}_1 = -5.3133$ and then subsequently use the formula for the confidence interval for $\beta_1$ in Method 5.15

$$\hat{\beta}_1 \pm t_{1-\alpha/2} \cdot \hat{\sigma}_{\beta_1} = \hat{\beta}_1 \pm t_{\alpha/2} \cdot \hat{\sigma} \sqrt{\frac{1}{\sum_{i=1}^{n}(x_i - \bar{x})^2}},$$

or just run `smf.ols() in` Python to find:

```
df = pd.DataFrame({
    ’t’: [10,20,30,40,50,60,70,80,90],
    ’y’: [420,365,285,220,176,117,69,34,5]
})
fit = smf.ols(’y ~ t’, data=df).fit()
print(fit.summary(slim=True))


                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.984
Model:                            OLS   Adj. R-squared:                  0.982
No. Observations:                   9   F-statistic:                     431.5
Covariance Type:            nonrobust   Prob (F-statistic):           1.51e-07
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Intercept    453.5556     14.394     31.511      0.000     419.520     487.591
t             -5.3133      0.256    -20.773      0.000      -5.918      -4.709
==============================================================================
```

and use the knowledge of the information in the Python output that what is know as the "standard error for the slope” can be directly read off as

$$\hat{\sigma}_{\beta_1} = \hat{\sigma}\sqrt{\frac{1}{\sum_{i=1}^{n}(x_i - \bar{x})^2}} = 0.2558,$$

and $t_{0.025}(7) = 2.364$ - in Python:

```
print(stats.t.ppf(0.975, 7))


2.3646242515927844
```

to get $-5.31 \pm 2.365 \cdot 0.2558$, or in Python:

```
print(-5.31 + np.array([-1, 1]) * stats.t.ppf(0.975, 7) * 0.256)


[-5.915 -4.705]
```

Notice that the `fit.summary()` output rounds the p-values to zero. To print more specific p-values in Python we can use `print(fit.pvalues)`.

b) Can a relation between temperature and life time be documented on level 5%?

### |||| Solution

Since the confidence interval does not include 0, it can be documented that there is a relationship between life time and temperature, also the $p$-value is $1.5 \cdot 10^{-7} < 0.05 = \alpha$, which also give strong evidence against the null-hypothesis.

## 5.3 Yield of chemical process

### |||| Exercise 5.3 Yield of chemical process

The yield $y$ of a chemical process is a random variable whose value is considered to be a linear function of the temperature $x$. The following data of corresponding values of $x$ and $y$ is found:

| Temperature in °C ($x$) | 0 | 25 | 50 | 75 | 100 |
|---|---|---|---|---|---|
| Yield in grams ($y$) | 14 | 38 | 54 | 76 | 95 |

The average and standard deviation of temperature and yield are

$$\bar{x} = 50,\ s_x = 39.52847,\ \bar{y} = 55.4,\ s_y = 31.66702,$$

In the exercise the usual linear regression model is used

$$Y_i = \beta_0 + \beta_1 x_i + \varepsilon_i, \quad \varepsilon_i \sim N(0, \sigma_\varepsilon^2), \quad i = 1, \ldots, 5$$

a) Can a significant relationship between yield and temperature be documented on the usual significance level $\alpha = 0.05$?

### Solution

It could most easily be solved by running the regression in Python as:

```
df = pd.DataFrame({
    ’x’: [0,25,50,75,100],
    ’y’: [14,38,54,76,95]})
fit = smf.ols(’y ~ x’, data=df).fit()
print(fit.summary(slim=True))
```

```
/home/jkmo/.local/lib/python3.10/site-packages/statsmodels/stats/stattools.py:74: Value
  warn("omni_normtest is not valid with less than 8 observations; %i "
                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.997
Model:                            OLS   Adj. R-squared:                  0.996
No. Observations:                   5   F-statistic:                     1071.
Covariance Type:            nonrobust   Prob (F-statistic):           6.27e-05
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Intercept     15.4000      1.497     10.290      0.002      10.637      20.163
x              0.8000      0.024     32.733      0.000       0.722       0.878
==============================================================================
```

Alternatively one could use hand calculations and use the formula in Theorem 5.12 for the $t$-test of the null hypothesis: $H_0 : \beta_1 = 0$.

The relevant test statistic and $p$-value can be read off in the Python output as 32.7 and 0.000063. So the answer is:

Yes, as the relevant test statistic and $p$-value are resp. 32.7 and $0.00006 < 0.05 = \alpha$.

b) Give the 95% confidence interval of the expected yield at a temperature of $x_{\text{new}} = 80$ °C.

### Solution

We use the formula in Equation (5-59) for the confidence limit of the line (the expected value of $Y_i$ for a value $x_{\text{new}}$):

$$\hat{\beta}_0 + \hat{\beta}_1 x_{\text{new}} \pm t_{1-\alpha/2}\hat{\sigma}\sqrt{\frac{1}{n} + \frac{(x_{\text{new}} - \bar{x})^2}{S_{xx}}},$$

and we have to compute $\hat{\beta}_0$, $\hat{\beta}_1$ and $s_e$ either by hand OR in Python as above:

$$\hat{\beta}_0 = 15.4,\ \hat{\beta}_1 = 0.8,\ \hat{\sigma} = 1.932.$$

So the confidence interval becomes

$$(15.4 + 0.8 \cdot 80) \pm 3.182 \cdot 1.932\sqrt{\frac{1}{5} + \frac{(80-50)^2}{6250}},$$

since

$$s_x^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2 = \frac{1}{n-1}S_{xx} \Leftrightarrow$$
$$S_{xx} = (n-1)s_x^2 = 4 \cdot 39.528^2 = 6250.$$

Thus the answer is

$$79.40 \pm 3.61 = [75.79,\ 83.01].$$

In Python this could be done by:

```
print(fit.get_prediction(pd.DataFrame({’x’: [80]})).summary_frame(alpha=0.05))


   mean   mean_se  mean_ci_lower  mean_ci_upper  obs_ci_lower  obs_ci_upper
0  79.4  1.133255      75.793477      83.006523      72.27132      86.52868
```

c) What is the upper quartile of the residuals?

### Solution

The five residuals become: -1.4, 2.6, -1.4, 0.6 og -0.4.

We use the basic definition of finding a quantile (from Definition 1.7) and the upper quartile is $q_{0.75}$ (see Definition 1.8). We set $n = 5$, $p = 0.75$, so

$$np = 3.75$$

So the upper quartile is the 4th observation in the ordered sequence:

$$-1.4, -1.4, -0.4, 0.6, 2.6.$$

The residuals can be shown using the `fit.resid` from the regression. They are:

```
Residuals:
   1   2    3   4    5
-1.4 2.6 -1.4 0.6 -0.4
```

So the answer is: 0.6.

## 5.4 Plastic material

### Exercise 5.4 Plastic material

In the manufacturing of a plastic material, it is believed that the cooling time has an influence on the impact strength. Therefore a study is carried out in which plastic material impact strength is determined for 4 different cooling times. The results of this experiment are shown in the following table:

| Cooling times in seconds (x) | 15 | 25 | 35 | 40 |
|---|---|---|---|---|
| Impact strength in kJ/m$^2$ (y) | 42.1 | 36.0 | 31.8 | 28.7 |

The following statistics may be used:

$$\bar{x} = 28.75,\ \bar{y} = 34.65,\ S_{xx} = 368.75.$$

a) What is the 95% confidence interval for the slope of the regression model, expressing the impact strength as a linear function of the cooling time?

## Solution

The easiest way to get to the confidence interval is to use the standard error for the slope ($\hat{\sigma}_{\beta_1}$ or denoted with $SE_{\beta_1}$) given in the Python output:

```
x = [15, 25, 35, 40]
y = [42.1, 36.0, 31.8, 28.7]
df = pd.DataFrame({'x': x, 'y': y})
fit = smf.ols('y ~ x', data=df).fit()
print(fit.summary(slim=True))
```

```
/home/jkmo/.local/lib/python3.10/site-packages/statsmodels/stats/stattools.py:74: Value
  warn("omni_normtest is not valid with less than 8 observations; %i "
                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.994
Model:                            OLS   Adj. R-squared:                  0.991
No. Observations:                   4   F-statistic:                     323.7
Covariance Type:            nonrobust   Prob (F-statistic):            0.00308
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Intercept     49.6390      0.878     56.513      0.000      45.860      53.418
x             -0.5214      0.029    -17.991      0.003      -0.646      -0.397
==============================================================================
```

the standard error for the slope is $\hat{\sigma}_{\beta_1} = 0.029$ (also known as the sampling distribution standard deviation for $\hat{\beta}_1$). Finding the relevant $t$-quantile (with $\nu = 2$ degrees of freedom (either of):

```
print(stats.t.ppf(0.025, 2), stats.t.ppf(0.975, 2))
```

```
-4.302652729696144 4.302652729696142
```

$|t_{0.025}| = 4.303$, which using Theorem 5.15 gives

$$-0.521 \pm 4.303 \cdot 0.029,$$

giving

$$-0.521 \pm 0.125,$$

or, that we say with high confidence that the true parameter value is in the interval, i.e.

$$-0.646 \leq \beta_1 \leq -0.396.$$

b) Can you conclude that there is a relation between the impact strength and the cooling time at significance level $\alpha = 5\%$?

### |||| Solution

The relevant $p$-value can be read off directly from the `summary` output: 0.00308, and we can conclude: *Yes, as the relevant p-value is 0.00308, which is smaller than 0.05.* Again, 0.003 can be read of the summary, but use `fit.pvalues` for more specific p-values.

c) For a similar plastic material the tabulated value for the linear relation between temperature and impact strength (i.e the slope) is $-0.30$. If the following hypothesis is tested (at level $\alpha = 0.05$)

$$H_0 : \beta_1 = -0.30$$
$$H_1 : \beta_1 \neq -0.30$$

with the usual $t$-test statistic for such a test, what is the range (for $t$) within which the hypothesis is accepted?

### |||| Solution

The so-called critical values for the $t$-statistic with $\nu = 2$ degrees of freedom is found as (or at least the negative one of the two): $t_{0.025} = -4.303$ - in Python: `stats.t.ppf(0.025,2)`). So the answer becomes:

$$[-4.303, 4.303].$$

## 5.5 Water polution

### Exercise 5.5 Water polution

In a study of pollution in a water stream, the concentration of pollution is measured at 5 different locations. The locations are at different distances to the pollution source. In the table below, these distances and the average pollution are given:

| | | | | | |
|---|---|---|---|---|---|
| Distance to the pollution source (in km) | 2 | 4 | 6 | 8 | 10 |
| Average concentration | 11.5 | 10.2 | 10.3 | 9.68 | 9.32 |

a) What are the parameter estimates for the three unknown parameters in the usual linear regression model: 1) The intercept ($\beta_0$), 2) the slope ($\beta_1$) and 3) error standard deviation ($\sigma$)?

### Solution

The question is solved by considering the following Python output:

```
df = pd.DataFrame({
    'concentration': [11.5, 10.2, 10.3, 9.68, 9.32],
    'distance': [2, 4, 6, 8, 10]
})
fit = smf.ols('concentration ~ distance', data=df).fit()
print(fit.summary(slim=True))


/home/jkmo/.local/lib/python3.10/site-packages/statsmodels/stats/stattools.py:74: Value
  warn("omni_normtest is not valid with less than 8 observations; %i "
                            OLS Regression Results
==============================================================================
Dep. Variable:          concentration   R-squared:                       0.868
Model:                            OLS   Adj. R-squared:                  0.823
No. Observations:                   5   F-statistic:                     19.66
Covariance Type:            nonrobust   Prob (F-statistic):             0.0213
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Intercept     11.6640      0.365     31.955      0.000      10.502      12.826
distance      -0.2440      0.055     -4.434      0.021      -0.419      -0.069
==============================================================================
```

Given the knowledge of the Python-output structure, the first two values can be read directly from the output. $\sigma$ can be found from either the Mean Squared error of the residuals `np.sqrt(fit.mse_resid)` or the scale `np.sqrt(fit.scale)`.
So the correct answer is: $\hat{\beta}_0 = 11.7$, $\hat{\beta}_1 = -0.244$ and $SE_{\hat{\sigma}} = \hat{\sigma} = 0.348$.

b) How large a part of the variation in concentration can be explained by the distance?

### Solution

The amount of variation in the model output ($Y$) explained by the variable input ($x$) can be found from the squared correlation, that can be read off directly from the output as `"R-squared"`. So the correct answer is: $R^2 = 86.8\%$ (it is actually an estimate of the variation in concentration which can be explained by distance, since it is what we found with the particular data at hand. If the sample was taken again, then this value would vary. We should actually calculate a confidence interval for $R^2$ to understand how accurate this estimate is!).

c) What is a 95%-confidence interval for the expected pollution concentration 7 km from the pollution source?

### Solution

The wanted number is estimated by the point on the line (using $x_{\text{new}} = 7$)

$$-0.244 \cdot 7 + 11.664 = 9.96,$$

and the confidence interval is given by

$$9.96 \pm t_{0.025}(3) \cdot \hat{\sigma} \sqrt{\frac{1}{5} + \frac{(7-6)^2}{S_{xx}}},$$

where $S_{xx} = 4^2 + 2^2 + 0^2 + 2^2 + 4^2 = 40$ and $t_{0.025}(3) = 3.182$ (in Python: `stats.t.ppf(0.975, 3)`). we have that

$$3.182 \cdot 0.348 \sqrt{\frac{1}{5} + \frac{1}{40}} = 0.525,$$

where $s_x$ is:

```
print(df['distance'].std(ddof=1))


3.1622776601683795
```

and thus

$$S_{xx} = (n-1) \cdot s_x^2 = 4 \cdot 3.162^2 = 40.$$

This could also have been found by

```
print(fit.get_prediction(pd.DataFrame({'distance': [7]})).summary_frame(alpha=0.05))


    mean   mean_se  mean_ci_lower  mean_ci_upper  obs_ci_lower  obs_ci_upper
0  9.956  0.165082       9.430636      10.481364      8.730151     11.181849
```

So the correct answer is:

$$9.96 \pm 0.525 = [9.43, 10.5].$$

## 5.6 Membrane pressure drop

### |||| Exercise 5.6 Membrane pressure drop

When purifying drinking water you can use a so-called membrane filtration. In an experiment one wishes to examine the relationship between the pressure drop across a membrane and the flux (flow per area) through the membrane. We observe the following 10 related values of pressure ($x$) and flux ($y$):

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|
| Pressure ($x$) | 1.02 | 2.08 | 2.89 | 4.01 | 5.32 | 5.83 | 7.26 | 7.96 | 9.11 | 9.99 |
| Flux ($y$) | 1.15 | 0.85 | 1.56 | 1.72 | 4.32 | 5.07 | 5.00 | 5.31 | 6.17 | 7.04 |

Copy this into Python to avoid typing in the data:

```
df = pd.DataFrame({
    'pressure': [1.02,2.08,2.89,4.01,5.32,5.83,7.26,7.96,9.11,9.99],
    'flux': [1.15,0.85,1.56,1.72,4.32,5.07,5.00,5.31,6.17,7.04]
})
```

a) What is the empirical correlation between pressure and flux estimated to? Give also an interpretation of the correlation.

## |||| Solution

The questions are most easily solved by using `smf.ols()` in Python

```
df = pd.DataFrame({
    ’pressure’: [1.02,2.08,2.89,4.01,5.32,5.83,7.26,7.96,9.11,9.99],
    ’flux’: [1.15,0.85,1.56,1.72,4.32,5.07,5.00,5.31,6.17,7.04]
})
fit = smf.ols(’flux ~ pressure’, data=df).fit()
print(fit.summary(slim=True))


                            OLS Regression Results
==============================================================================
Dep. Variable:                   flux   R-squared:                       0.929
Model:                            OLS   Adj. R-squared:                  0.920
No. Observations:                  10   F-statistic:                     104.6
Covariance Type:            nonrobust   Prob (F-statistic):           7.18e-06
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Intercept     -0.1886      0.442     -0.427      0.681      -1.207       0.830
pressure       0.7225      0.071     10.227      0.000       0.560       0.885
==============================================================================
```

The found coefficient of determination (see Theorem 5.25) can be read off the Python output to be 0.929. The sign of the correlation is the same as the sign of the slope, which can be read off to be positive ($\hat{\beta}_1 = 0.7225$), so the correlation is

$$\hat{\rho} = r = \sqrt{0.929} = 0.964.$$

So the empirical correlation is 0.964, and thus flux is found to increase with increasing pressure.

b) What is a 90% confidence interval for the slope $\beta_1$ in the usual regression model?

### |||| Solution

We use the formula for the slope ($\beta_1$, see Method 5.15) confidence interval, and can actually just realize that the correct $t$-quantile to use is the $t_{1-0.05}(8) = 1.860$ (in Python: `stats.t.ppf(0.95,8)`), and the other values we read of the `summary` output. So the confidence interval is: $0.7225 \pm 1.860 \cdot 0.0706$.

c) How large a part of the flux-variation ($\sum_{i=1}^{10}(y_i - \bar{y})^2$) is not explained by pressure differences?

### |||| Solution

The squared correlation, $r^2 = 0.929$ express the explained variation, this means that $1 - 0.929 = 0.071$ express the unexplained variation by the model.

d) Can you at significance level $\alpha = 0.05$ reject the hypothesis that the line passes through $(0, 0)$?

### |||| Solution

The hypothesis is the same as:

$$H_0 : \beta_0 = 0$$

which is the hypothesis results provided in the output in the `"intercept"` row of the `summary`, so: *No, since the relevant $p$-value is 0.68, which is larger than $\alpha$.*

e) A confidence interval for the line at three different pressure levels: $x_{\text{new}}^{\text{A}} = 3.5$, $x_{\text{new}}^{\text{B}} = 5.0$ and $x_{\text{new}}^{\text{C}} = 9.5$ will look as follows:

$$\hat{\beta}_0 + \hat{\beta}_1 \cdot x_{\text{new}}^{U} \pm C_U$$

where $U$ then is either A, B or C. Write the constants $C_U$ in increasing order.

## |||| Solution

The formula for the Confidence limits of $\alpha + \beta x_{\text{new}}$ includes the following term:

$$\frac{(x_{\text{new}} - \bar{x})^2}{S_{xx}}$$

and this is the ONLY term in $C_U$ that makes $C_U$ different between the three $U$s. And since $\bar{x} = 5.547$ it is clear that

$$(5.0 - 5.547)^2 < (3.5 - 5.547)^2 < (9.5 - 5.547)^2$$

and hence

$$(x^{\text{B}}_{\text{new}} - 5.547)^2 < (x^{\text{A}}_{\text{new}} - 5.547)^2 < (x^{\text{C}}_{\text{new}} - 5.547)^2$$

So $C_{\text{B}} < C_{\text{A}} < C_{\text{C}}$

## 5.7 Membrane pressure drop (matrix form)

### |||| Exercise 5.7 Membrane pressure drop (matrix form)

This exercise uses the data presented in Exercise 6 above.

a) Find parameters values, standard errors, *t*-test statistics, and *p*-values for the standard hypotheses tests.

Copy this into Python to avoid typing in the data:

```
df = pd.DataFrame({
    ’pressure’: [1.02,2.08,2.89,4.01,5.32,5.83,7.26,7.96,9.11,9.99],
    ’flux’: [1.15,0.85,1.56,1.72,4.32,5.07,5.00,5.31,6.17,7.04]
})
```

### |||| Solution

```
df = pd.DataFrame({
    ’pressure’: [1.02,2.08,2.89,4.01,5.32,5.83,7.26,7.96,9.11,9.99],
    ’flux’: [1.15,0.85,1.56,1.72,4.32,5.07,5.00,5.31,6.17,7.04]
})
fit = smf.ols(’flux ˜ pressure’, data=df).fit()
print(fit.summary(slim=True))


                            OLS Regression Results
==============================================================================
Dep. Variable:                   flux   R-squared:                       0.929
Model:                            OLS   Adj. R-squared:                  0.920
No. Observations:                  10   F-statistic:                     104.6
Covariance Type:            nonrobust   Prob (F-statistic):           7.18e-06
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Intercept     -0.1886      0.442     -0.427      0.681      -1.207       0.830
pressure       0.7225      0.071     10.227      0.000       0.560       0.885
==============================================================================
```

> The parameter estimates are given in the first column, the standard errors in the second column, the t-test statistics are given in the third column and the $p$-values of the standard hypothesis are given in the fourth column. (followed by the confidence interval)

b) Reproduce the above numbers by matrix vector calculations. You will need some matrix notation in Python:
   - Matrix multiplication ($XY$): `np.dot(X,Y) or X@Y`
   - Matrix transpose ($X^T$): `X.T`
   - Matrix inverse ($X^{-1}$): `np.linalg.inv(X)`
   - Make a matrix from vectors ($X = [x_1^T; x_2^T]$): `np.column_stack((x1,x2))`

   See also Example 5.24.

## |||| Solution

```
X = np.column_stack((np.ones(len(df[’pressure’])), df[’pressure’]))
y = df[’flux’]
n = len(y)

# Beta calculation
beta = np.linalg.inv(X.T @ X) @ X.T @ y

# Error term and standard error calculation
e = y - X @ beta
s = np.sqrt(np.sum(e**2) / (n - 2))
Vbeta = s**2 * np.linalg.inv(X.T @ X)
se_beta = np.sqrt(np.diag(Vbeta))

# t-statistic and p-value
t_obs = beta / se_beta
p_value = 2 * (1 - stats.t.cdf(np.abs(t_obs), df=n-2))

# Organizing results into a table
analysis_table = np.column_stack((beta, se_beta, t_obs, p_value))

# Assigning column and row names
col_names = ["Estimates", "Std.Error", "t.obs", "p.value"]
row_names = ["beta1", "beta2"]

# Display table
result_table = pd.DataFrame(analysis_table, index=row_names, columns=col_names)
print(result_table)
```

```
       Estimates  Std.Error      t.obs   p.value
beta1  -0.188574   0.441712  -0.426917  0.680697
beta2   0.722476   0.070644  10.226945  0.000007
```

## 5.8 Independence and correlation

### |||| Exercise 5.8 Independence and correlation

Consider the layout of independent variable in Example 5.11,

a) Show that $S_{xx} = \frac{n\cdot(n+1)}{12\cdot(n-1)}$.

Hint: you can use the following relations

$$\sum_{i=1}^{n} i = \frac{n(n+1)}{2},$$
$$\sum_{i=1}^{n} i^2 = \frac{n(n+1)(2n+1)}{6}.$$

### |||| Solution

$\bar{x}$ becomes

$$\begin{aligned}\bar{x} &= \frac{1}{n}\sum_{i=1}^{n}\frac{i-1}{n-1} = \frac{1}{n(n-1)}\sum_{i=1}^{n}(i-1)\\ &= \frac{1}{n(n-1)}\left(\frac{n(n+1)}{2}-n\right) = \frac{1}{2},\end{aligned}$$

and $S_{xx}$ becomes

$$\begin{aligned}S_{xx} &= \sum_{i=1}^{n}\left(\frac{i-1}{n-1}-\frac{1}{2}\right)^2\\ &= -\frac{n}{4}+\frac{1}{(n-1)^2}\sum_{i=1}^{n}(i^2+1-2i)\\ &= -\frac{n}{4}+\frac{1}{(n-1)^2}\left(\frac{n(n+1)(2n+1)-6n^2}{6}\right)\\ &= \frac{n}{(n-1)^2}\left(\frac{4n^2+6n+2-12n-3(n-1)^2}{12}\right)\\ &= \frac{n}{(n-1)^2}\left(\frac{n^2-1}{12}\right) = \frac{n(n+1)}{12(n-1)}.\end{aligned}$$

b) Show that the asymptotic correlation between $\hat{\beta}_0$ and $\hat{\beta}_1$ is

$$\lim_{n\to\infty} \rho_n(\hat{\beta}_0, \hat{\beta}_1) = -\frac{\sqrt{3}}{2}.$$

### |||| Solution

The correlation between $\hat{\beta}_0$ and $\hat{\beta}_0$ is

$$\begin{aligned}
\rho_n(\hat{\beta}_0, \hat{\beta}_1) &= \frac{\mathrm{Cov}(\hat{\beta}_0, \hat{\beta}_1)}{\sqrt{\mathrm{V}(\hat{\beta}_0)\,\mathrm{V}(\hat{\beta}_1)}} \\
&= -\frac{\sigma^2 \bar{x}/S_{xx}}{\sqrt{\sigma^4\left(\frac{1}{n}+\frac{\bar{x}^2}{S_{xx}}\right)\frac{1}{S_{xx}}}} \\
&= -\frac{\bar{x}/S_{xx}}{\frac{1}{S_{xx}}\sqrt{\left(\frac{S_{xx}}{n}+\bar{x}^2\right)}} \\
&= -\frac{\bar{x}}{\sqrt{\frac{S_{xx}}{n}+\bar{x}^2}}.
\end{aligned}$$

Notice that the correlation is not a function of the variance ($\sigma^2$), but only a function of the independent variables. Now insert the values of $\bar{x}$ and $S_{xx}$

$$\begin{aligned}
\rho_n(\hat{\beta}_0, \hat{\beta}_1) &= -\frac{1}{2\sqrt{\frac{n+1}{12(n-1)}+\frac{1}{4}}} = -\frac{1}{2\sqrt{\frac{n+1+3(n-1)}{12(n-1)}}} \\
&= -\frac{1}{2\sqrt{\frac{2n-1}{6(n-1)}}} = -\frac{\sqrt{6(n-1)}}{2\sqrt{2n-1}} \\
&= -\frac{1}{2}\sqrt{\frac{6(n-1)}{2(n-1/2)}} = -\frac{\sqrt{3}}{2}\sqrt{\frac{n-1}{n-1/2}}
\end{aligned}$$

.

which converges to $-\frac{\sqrt{3}}{2}$ for $n \to \infty$.

Consider a layout of the independent variable where $n = 2k$ and $x_i = 0$ for $i \leq k$ and $x_i = 1$ for $k < i \leq n$.

c) Find $S_{xx}$ for the new layout of $x$.

### |||| Solution

$$\bar{x} = \frac{1}{2},$$

and

$$\begin{aligned} S_{xx}^{\text{new}} &= \sum_{i=1}^{k} \left(0 - \frac{1}{2}\right)^2 + \sum_{i=k+1}^{2k} \left(1 - \frac{1}{2}\right)^2 \\ &= \frac{k}{4} + \frac{k}{4} = \frac{k}{2} = \frac{n}{4}. \end{aligned}$$

d) Compare $S_{xx}$ for the two layouts of $x$.

### |||| Solution

$$\frac{S_{xx}}{S_{xx}^{\text{new}}} = \frac{n(n+1)}{12(n-1)} \frac{4}{n} = \frac{(n+1)}{3(n-1)} < 1; \quad for \quad n > 2$$

which imply that $S_{xx}^{\text{new}} > S_{xx}$ for all $n > 2$.

e) What is the consequence for the parameter variance in the two layouts?

### |||| Solution

The larger $S_{xx}$ for the new layout imply that the parameter variance is smaller for the new layout (given that data comes from the same model).

f) Discuss pro's and cons for the two layouts.

## |||| Solution

The smaller parameter variance for the new layout would suggest that we should use this layout. However, we would not be able to check that data is in fact generated by a linear model. Consider e.g. data generated by the model

$$y_i = \beta_0 + \beta_1 x_i^2 + \varepsilon_i, \quad \varepsilon_i \sim N(0, \sigma^2),$$

if we only look at $x_i = 0$ or $x_i = 1$ we will not be able to detect that the relationship is in fact non-linear.