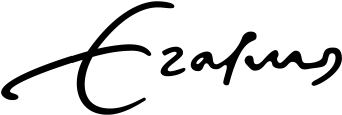
MOOC Econometrics

# Test Exercise 1

**Notes:**

* See website for how to submit your answers and how feedback is organized.
* This exercise uses the datafile TestExer1 and requires a computer.
* The dataset TestExer1 is available on the website.

**Goals and skills being used:**

* Get hands-on experience with performing simple regressions.
* Get feeling for consequences of violations of regression assumptions.
* Obtain some experience with how to diagnose that an assumption is violated.

**Questions**

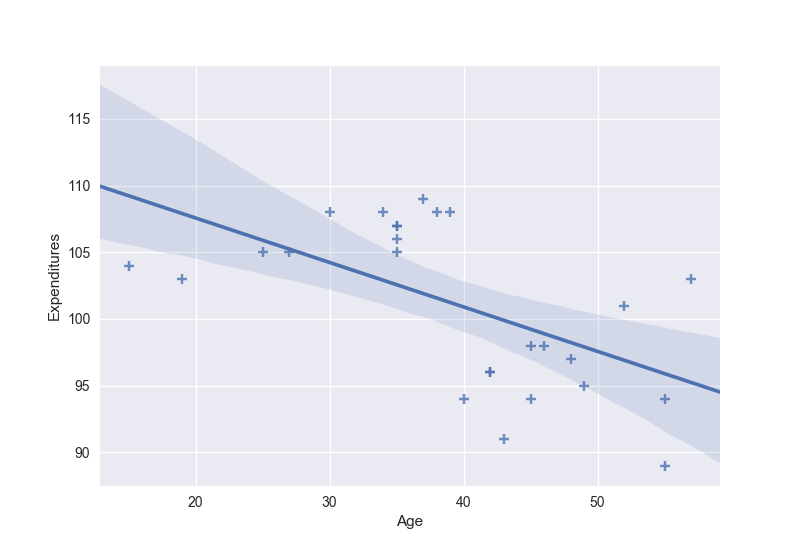
This exercise considers an example of data that do not satisfy all the standard assumptions of simple regression. In the considered case, assumption A6 that the coefficients *α* and *β* are the same for all observations is violated. The dataset contains survey outcomes of a travel agency that wishes to improve recommendation strategies for its clients. The dataset contains 26 observations on age and average daily expenditures during holidays.

1. Use all data to estimate the coefficients *a* and *b* in a simple regression model, where expenditures is the dependent variable and age is the explanatory factor. Also compute the standard error and the *t*-value of *b*.
2. Make the scatter diagram of expenditures against age and add the regression line *y* = *a* + *bx* of part (a) in this diagram. What conclusion do you draw from this diagram?
3. It seems there are two sets of observations in the scatter diagram, one for clients aged 40 or higher and another for clients aged below 40. Divide the sample into these two clusters, and for each cluster estimate the coefficients *a* and *b* and determine the standard error and *t*-value of *b*.
4. Discuss and explain the main differences between the outcomes in parts (a) and (c). Describe in words what you have learned from these results.

HW1 Answer. The code is in appendix.

**Q1:** (using formula in lecture 1.4, slide 9)

a = 114.2411, b = -0.3336  
standard error = 5.0733, tb = -7.4918E-16  
standard error of b = 0.0954, t-value of b = -3.4979  
  
**Q2:**Figure 1: Scatter Diagram of Expenditure against Age.

  
As shown in the above scatter diagram, the average daily expenditure of older (higher age) client is usually lower. On the scatter diagram, the regression line is downward sloping,

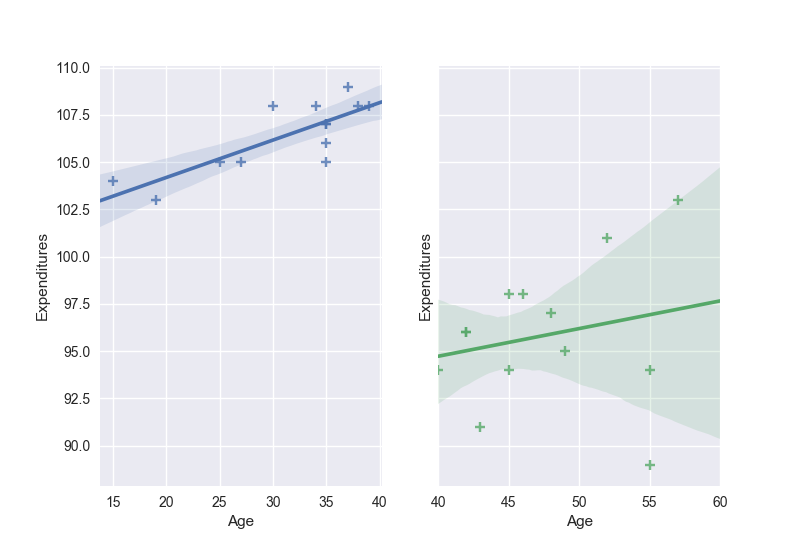
**Q3:**

For Age < 40:  
  
a = 100.2323, b = 0.1980  
standard error = 1.1531, tb = 1.0015E-16  
standard error of b = 0.0444, t-value of b = 4.4605  
  
For Age >= 40:

a = 88.8719, b = 0.1465  
standard error = 3.8329, tb = 2.1093E-16  
standard error of b = 0.1974, t-value of b = 0.7421

**Q4:**

* 1. The main difference between Q1 and Q3 is the sign of b value. In part(1), the outcome tells that older clients(higher age) have lower expenditure, or the expenditure is trending down as age increases. In part(3), the outcome tells that the expenditure is trending up in two separate groups.
  2. As shown in Figure 1, the expenditure trends down as the age increases in part(1). In part(3), the original data are split into two sets. For clients of age < 40, the expenditure is slightly trending up with slope b = 0.198; for clients of age >= 40, the expenditure is also slightly trending up with slope b = 0. 1465.  
       
     Besides, the standard error in part(1) is much higher than the standard errors in part(3).  
       
     Finally, the average expenditure of the clients with age < 40 is higher than the average expenditure of the clients with age >= 40.   
       
       
     Figure 2: Two sets of observation.



===Python code====

**import** pandas **as** pd  
**import** numpy **as** np  
**import** statsmodels.api **as** sm  
**import** seaborn **as** sns;  
**import** matplotlib.pyplot **as** plt  
  
df = pd.read\_excel(**'TestExer1-holiday expenditures-round2.xls'**)  
  
print(**"Input Data:"**)  
print(df)  
*# data = df.values***def** calc\_stats(x, y):  
  
 n = len(x)  
  
 y1 = y - np.mean(y) *# de-meaned y* x1 = x - np.mean(x) *# de-meaned x* b = np.inner(y1, x1)/np.inner(x1, x1)  
 a = np.mean(y) - b\*np.mean(x)  
  
 e = y - a - b\*x  
 s2 = np.inner(e, e)/(n-2)  
 s = np.sqrt(s2)  
 R2 = 1 - np.inner(e, e)/np.inner(y1, y1)  
  
 *# tb = (b-beta)/sb = sum(c\*e)/sb # lecture 1.4, slide 9* c = x1/np.inner(x1, x1)  
 beta = b - np.inner(c, e)  
 sb2 = s2/np.inner(x1, x1)  
 sb = np.sqrt(sb2)  
 *# tb = (b - beta)/sb* tb = np.inner(c, e)/sb  
 tb1 = b/sb  
  
 print(**'a = %.4f'** % a)  
 print(**'b = %.4f'** % b)  
 print(**'standard error = %.4f'** % s)  
 print(**'tb = %.4E'** % tb)  
 print(**'standard error of b = %.4f'** % sb)  
 print(**'t-value of b = %.4f'** % tb1)  
 *# print(sb2, sb, b, beta, np.inner(c, e), sb)* X = sm.add\_constant(x)  
 model = sm.OLS(y, X)  
 results = model.fit()  
 print(results.summary())  
  
print(**"\nQ1:"**)  
  
y = np.array(df[**'Expenditures'**])  
x = np.array(df[**'Age'**])  
calc\_stats(x, y)  
  
print(**'\nQ2:'**)  
  
sns.set(color\_codes=**True**)  
ax = sns.regplot(data=df, x=**'Age'**, y=**'Expenditures'**, marker=**'+'**)  
plt.show()  
  
print(**"\nQ3:"**)  
  
print(**'For Age >= 40:'**)  
df\_GE40 = df.copy()  
df\_GE40 = df\_GE40[df\_GE40[**'Age'**]>=40]  
y = np.array(df\_GE40[**'Expenditures'**])  
x = np.array(df\_GE40[**'Age'**])  
calc\_stats(x, y)  
  
print(**'\nFor Age < 40:'**)  
df\_L40 = df.copy()  
df\_L40 = df\_L40[df\_L40[**'Age'**]<40]  
y = np.array(df\_L40[**'Expenditures'**])  
x = np.array(df\_L40[**'Age'**])  
calc\_stats(x, y)  
  
print(**"\nQ4:"**)  
  
fig, (ax1, ax2) = plt.subplots(ncols=2, sharey=**True**)  
ax2.set(xlim=(40,60))  
sns.regplot(data=df\_L40, x=**'Age'**, y=**'Expenditures'**, marker=**'+'**, ax=ax1)  
sns.regplot(data=df\_GE40, x=**'Age'**, y=**'Expenditures'**, marker=**'+'**, ax=ax2)  
plt.show()