BS2280 - Econometrics
R Workshop 3 - Predictions, Residuals and Multiple Regression Analysis

In this computer lab we will have to achieve the following tasks/learning outcomes:

- import .Rdata data and view the data
- run simple regression model
- undertake hypothesis tests and predictions
- create the histogram of the residuals
- calculate the arithmetic mean of the residuals
- run multiple regression model

Preparing your workspace

Before you do each task, you need to prepare your workspace first.

Step 1. Create a folder called RWorkshop3

Step 2. Go to Blackboard week 6 R Workshop 3 and download datafile: crime.Rdata, save it in the RWorkshop3 folder you created in step 1

Step 3. Open Rstudio and set working directory

Menu bar → Click Session → Set Working Directory → Choose Directory → Select RWorkshop3 folder you created in step 1

Step 4. Create an R script

!! If you forget how to prepare your workspace, please review R Workshop 0 first.

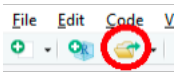
Task 1. Open the data set in R.

This is what we have done in R Workshop 2. In this Workshop 3, we will use crime.Rdata again, so repeat what you have done in R Workshop 2 task 1.

Before you start working on the crime dataset, ensure that you have prepared the workspace. From R Workshop 1, we know the native data format of R is .Rdata. In this Workshop 3, we are going to use crime.Rdata so we do not need to activate library readxl this time.

To open the R datafile crime.Rdata,

Option 1. Click on



and select the crime.Rdata dataset.

Option 2. Use the command line:

```
load("~/Desktop/Aston/Econometrics/Rworkshop3/crime.Rdata")
# this is my working path, you should change it to your working path
```

```
summary(crime)        # get summary descriptive statistics for dataset crime
```

Task 2. Run a simple regression model of officers on crimes and discuss the statistical significance of the intercept, the coefficient of crimes and of R-squared (R^2).

Running regression in R is simple.

```
officersfit <- lm(XXXX~XXXX,data=crime)
officersfit
```

Look at the codes first.

officersfit: name this regression model as officersfit

lm(): lm() activates the regression function for linear model

officers: the dependent variable

crimes: the independent variable

data=crime: specify variable "officers" and "crimes" are from dataset "crime"

We can avoid repetitively including the crime$ prefix by using the argument 'data = NAME OF DATASET'. The tilde (~) is telling R that we regress officers on crimes.

The estimated regression model is:

$$\widehat{officers} = -5.4183 + 0.0238 crimes$$

The output is produced and we should be careful with the interpretation. The intercept states that if we have zero crimes within a city, the number of police officers would be -5.4. In this case, the intercept is meaningless. Always be careful when interpreting an intercept when it does not lie within the data range.

We do not have any city that has actually a crime rate of zero. The slope coefficient states that for every additional crime, we observe on average 0.24 more police officers. To use more user-friendly numbers, we can also infer that for every 1,000 additional crimes committed within a city, 24 more police officers are employed.

R^2 is the measure that provides information on the overall goodness of fit of the model. The R^2 value can be found in the summary output table of the regression object officersfit:

```
summary(officersfit)
```

In this case it is 0.83. This means that 83% of the variation in police officers can be explained with the variation in number of crimes committed. Our estimated model has a good degree of explanatory power. Running regression in R is simple.

Task 3. Using the regression results from above, check the model's prediction for the number of police officers based on the number of crimes committed within a city.

$$Y_i = \hat{Y}_i + \hat{u}_i = \hat{\beta}_1 + \hat{\beta}_2 X_i + \hat{u}_i$$

$$officers_i = \widehat{officers}_i + \hat{u}_i = \hat{\beta}_1 + \hat{\beta}_2 crimes_i + \hat{u}_i$$

Therefore, predicted officers is calculated by

$$\widehat{officers}_i = \hat{\beta}_1 + \hat{\beta}_2 crimes_i$$

We can use regression results to predict the number of police officers, based on the number of crimes committed. We will use the **predict** command to solve this task:

```
crime$XXXX <- XXXX(officersfit, newdata=data.frame(crimes=crime$XXXX))
```

I start with explaining the right hand side first: The first argument of the predict command requires you to mention the model that we use for our predictions. In our case, it is officersfit.

newdata=: The next argument newdata, requires information on the values we want to base our predictions on.

data.frame(crimes=crime$crimes): We use the actual number of crimes in each city: crimes=crime$crimes. If you wanted to predict the number of police officers for a city with 30,000 crimes committed, use: crimes=30000. Please do not forget to add the data.frame() command, otherwise you will get an error message.

crime$officershat: We want to add our predictions as a variable called officershat or yhat to our crime dataset. This explains the left hand side of the command.

Open the data viewer to take a look at the predictions.

Task 4. Calculate the residuals $\hat{u}_i$ for each observation to identify how far off the predicted values are away from the actual values.

$$Y_i = \hat{Y}_i + \hat{u}_i = \hat{\beta}_1 + \hat{\beta}_2 X_i + \hat{u}_i$$

$$officers_i = \widehat{officers}_i + \hat{u}_i = \hat{\beta}_1 + \hat{\beta}_2 crimes_i + \hat{u}_i$$

Therefore, residuals $\hat{u}_i$ is calculated by

$$\hat{u}_i = officers_i - \widehat{officers}_i$$

The difference between the actual value of officers minus the predicted value of officers is the residual $\hat{u}_i$ - the unexplained part of our estimation. This 'error' we try to minimise when we use OLS:

```
crime$residuals <- crime$XXXX-crime$XXXX
```

Now we look at the codes:

crime$officers-crime$officershat: pick actual value of officers from crime dataset minus the predicted value of officers from crime dataset

crime$residuals <-: assign the the results of crime$officers-crime$officershat to a variable called residuals to our crime dataset.

Open the data viewer to see the results.

Task 5. Construct the histogram of the residuals.

A histogram plot helps to identify the distribution of the residuals. Remembers, that we assume that the residuals are approximately normally distributed.

```
hist(crime$XXXX, main = "Histogram of model residuals",
xlab = "Residuals")
```

Now we look at the codes:

hist(): this function creates a histogram plot

crime$residuals: pick variable residuals from dataset crime and make use of it to create a histogram

main = : give this histogram a title "Relationship between number of police officers and crime"

xlab = : name the x axis as "Residuals" for this histogram

Task 6. Calculate the arithmetic mean of the residuals.

We simple select the variable of interest and apply the mean command:

```
XXXX(crime$residuals)
```

mean(crime$residuals): mean function will calculate the average value for the variable residuals from crime dataset

OLS Assumption 3. Disturbance term has zero expectation.

This value is extremely small (0.00000000000002). According to Assumption 3, the value should be 0, but due to the limit in the number of decimal places software packages use when storing numbers were are a tiny bit off.

Task 7. Add popdens to the regression model, run the model and comment on the regression results. Identify differences between the results of the simple regression model and the multiple regression model:

You can follow similar steps you have taken when undertaking the simple regression model:

```
officersfit2 <- lm(officers~XXXX+XXXX,data=crime)
summary(officersfit2)
```

Look at the codes.

officersfit2: name this regression model as officersfit2

lm(): lm() activates the regression function for linear model

officers: the dependent variable

crimes+popdens: we have two independent variable this time so it means we are running the multiple regression

data=crime: specify variable "officers", "crimes" and "popdens" are from dataset "crime"

summary(): this function shows the output table of the regression object officersfit2

To add another independent variable to our model use + new variable after the first independent variable. The display format of the regression coefficients is not very convenient, as R used the scientific notation heavily. To 'punish' R for using scientific notation, we can introduce a penalty with the scipen option. The higher the penalty, the less likely R uses scientific notation, i.e. if the penalty is 999, R will hardly use scientific notation.

For example:

```
options(scipen=4) # Set scipen = 0 to get back to default
summary(officersfit2)
```

Always, we can ask Dr.Google ("R option scipen") or use the help function (?options) to figure out what the optional input into the option function.

$$\widehat{officers} = -182.2418 + 0.0232crimes + 0.0400popdens$$

Now carefully compare and contrast the results from the simple and the multiple regression model.

Task 8. Comment on the R-squared (R^2) values across both the simple regression model and the multiple regression model.

Simple regression model (officersfit) output:

```
##
## Call:
## lm(formula = officers ~ crimes, data = crime)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -756.64 -153.71  -25.75   89.64 1000.97
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -5.418291  75.587257  -0.072    0.943
## crimes       0.023804   0.001611  14.777   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 298.9 on 44 degrees of freedom
## Multiple R-squared:  0.8323, Adjusted R-squared:  0.8285
## F-statistic: 218.4 on 1 and 44 DF,  p-value: < 2.2e-16
```

Multiple regression model (crimefit2) output:

```
##
## Call:
## lm(formula = officers ~ crimes + popdens, data = crime)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -630.09 -119.91  -33.32  119.86  934.67
##
## Coefficients:
##                Estimate  Std. Error t value Pr(>|t|)
## (Intercept) -182.241869   89.566586  -2.035  0.04808 *
## crimes         0.023226    0.001485  15.645  < 2e-16 ***
## popdens        0.040032    0.012898   3.104  0.00337 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 273.3 on 43 degrees of freedom
## Multiple R-squared:  0.863,  Adjusted R-squared:  0.8566
## F-statistic: 135.4 on 2 and 43 DF,  p-value: < 2.2e-16
```

The R-squared value is higher for the multiple regression model than for the simple regression model. Adding more variables to a model will always increase the explanatory power of a model.

The R-squared for the multiple regression model reveals that 86 % of the variation in officers can be explained by the variations in the independent variables. The explanatory power of the estimated model is high.

Further Exercise

We move away from crimes to CEO salaries. Download the Stata data set called ceosal.dta from the module page on Blackboard and save it. The dataset records information on CEOs (their salary, age, etc.) across

177 companies and some financial information on these companies. (Source: http://fmwww.bc.edu/ec-p/data/wooldridge/datasets.list.html) In the following exercises you can practice your skills you have gained in the previous three workshops:

1. Open the data set in R
2. Label the variables using the following definitions:

salary	...	CEO compensation in 1990, $
age	...	age in years
bach	...	college =1 if attended college
grad	...	grad =1 if attended graduate school
comten	...	years with company
ceoten	...	years as CEO with company
sales	...	Firm sales in 1990, $
profit	...	Firm profits in 1990, $
mktval	...	Firm market value in 1990, $

3. Provide summary statistics for the variables salary, sales, mktval and comment on them.
4. Run a simple regression model of CEO salary on firm sales and interpret the results.

$$salary_i = \beta_1 + \beta_2 sales_i + u_i$$

5. Rescale the variables by converting salary in $ 000 and sales and mktval in $ million. Hint: create a new variable.
6. Using the rescaled variable, re-run the model in question 4 and interpret the estimated coefficients.
7. What can you say about the goodness of fit of the model? Is it a good model to explain variation in CEO salary?
8. Add the firm market value to the above regression model, run the model and comment on the regression results. Identify differences between the results of the simple regression model undertaken in question 4.

$$salary_i = \beta_1 + \beta_2 sales_i + \beta_3 mktval_i + u_i$$

9. Comment on the R-squared values across both the simple regression model and the multiple regression model in question 7.
10. Construct the predicted values of CEO salary from the regression model run in Question 7, i.e.:

$$\widehat{salary_i} = \hat{\beta}_1 + \hat{\beta}_2 sales_i + \hat{\beta}_3 mktval_i$$

11. Calculate the predicted residuals
12. Construct the histogram of the predicted residuals

The following questions are optional and aim at students who want to deepen their understanding in calculating regression coefficients and improve their R skills:

13. Consider the following regression model of CEO salary on firm sales.

$$salary_i = \beta_1 + \beta_2 sales_i + u_i$$

We aim to estimate the values of the regression coefficients, which in a simple regression model can be estimated using the following formulae:

$$\hat{\beta}_2 = \frac{\sum_{i=1}^{n}(X_i - \overline{X})(Y_i - \overline{Y})}{\sum_{i=1}^{n}\left(X_i - \overline{X}\right)^2}$$

$$\hat{\beta}_1 = \overline{Y} - \hat{\beta}_2 \overline{X}$$

Calculate the values of the regression coefficients $\hat{\beta}_1$ and $\hat{\beta}_2$ in R as per the above formulae.