BS2280 - Econometrics
R Workshop 5 - Nonlinear Regression

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

- import .Rdata data and view the data
- run different types multiple regression model
- interpret different types multiple regression model

Preparing your workspace

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

Step 1. Create a folder called RWorkshop5

Step 2. Go to Blackboard week 11 R Workshop 5 and download datafile: wages.Rdata, save it in the RWorkshop5 folder you created in step 1

Step 3. Open Rstudio and set working directory

Menu bar → Click Session → Set Working Directory → Choose Directory → Select RWorkshop5 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.

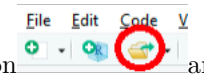
Task 1. Open the data set in R.

In workshop 5 we will introduce the estimation of nonlinear regression models in R. Download the R data set called **wages.Rdata**. This is the data set on earnings and other characteristics for 526 individuals we used in Workshop 4, but in different format:

wage	...	average hourly earnings ($)
educ	...	years of education
exper	...	years of potential experience
tenure	...	years with current employer
numdep	...	number of dependents
gender	...	gender of individual
sector	...	industry of occupation
location	...	region where the individual works

Before you start working on the wages dataset, ensure that you have prepared the workspace. Look at the name of the wages data set. From R Workshop 1, we know the native data format of R is .Rdata. In this Workshop 5, we are going to use wages.Rdata so we do not need to use read.excel() or read.csv() this time.

To open the R datafile wages.Rdata in your folder R Workshop 5,

Option 1. Click on 

and select the wages.Rdata dataset.

Option 2. Use the command line:

```
load("~/Desktop/BS2280/R Workshop5/wages.Rdata")
# this is my working path, you should change it to your working path
```

Task 2. Generate variables that are the natural logarithm of wage, educ, exper and tenure.

R has a built in log-function, allowing us to create the log transformed variables swiftly:

```
wages$lnwage <- XXXX(wages$XXXX)        # log transformation of wage
```

wages$lnwage: generate a new variable called lnwage in data set wages

log(): log transformation function

wages$wage: pick variable wage from dataset wages and make use of it to do log transformation

```
XXXX(XXXX(wages$lnwage))
# Does the log transformation for wage result in any lost observations?
```

is.infinite(wages$lnwage): pick variable lnwage from dataset wages and check if it has any infinity values

sum(): the sum of observations

sum(is.infinite(wages$lnwage)): sum of infinity values for variable lnwage

```
wages$XXXX <- log(wages$XXXX)       # log transformation of exper to get lnexper
sum(is.infinite(wages$XXXX))
# Does the log transformation for exper result in any lost observations?
```

```
wages$XXXX <- log(wages$XXXX)       # log transformation of educ to get lneduc
sum(is.infinite(wages$XXXX))
# Does the log transformation for educ result in any lost observations?
```

```
wages$XXXX <- log(wages$XXXX)      # log transformation of tenure to get lntenure
sum(is.infinite(wages$XXXX))
# Does the log transformation for tenure result in any lost observations?
```

You will have realised that the logarithmic transformation of educ and tenure lead to the generation of -Inf values: negative infinity.

We cannot use infinity values for our estimations, therefore the logarithmic transformation led to a loss of data if the variable of interest has observations with values of smaller and equal to 0. Logarithmic transformation is not always feasible.

In this example, the smallest value that we can observe is 0. You will also come across data, e.g. FDI data, that contains negative values. If you undertake a logarithmic transformation with negative numbers, then R will inform you that you have generated a NaN - 'Not a Number'. The is.infinite() command will not count those. I recommend to use the is.infinite() and is.nan() to count both -Inf and NaN values:

```
sum(XXXX(wages$lnwage) | XXXX(wages$lnwage))
```

The vertical bar '|' stands for 'or', and commands R to count the observations that are either infinite or not a number.

Task 3. Estimate the log-model below and interpret each coefficient:

$$log\, wage_i = \beta_1 + \beta_2 log\, exper_i + \beta_3 log\, educ_i + \delta male_i + u_i$$

We use the variables we created in Task 2 to run the log regression.

```
lnwagefit1 <- lm(XXXX~XXXX+XXXX+gender, data=wages)
```

lnwagefit1: name this regression model as lnwagefit1

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

lnwage: the dependent variable

lnexper+lneduc+gender: the independent variables

data=wages: specify variable lnwage, lnexper, lneduc and gender are from dataset wages

Following up the discussion about logarithmic transformation in Task 3: If we add our log education and log work experiences from Task 2, R will give you an error message: **Error in lm.fit(x, y, offset = offset, singular.ok = singular.ok, ...) : NA/NaN/Inf in 'x'**.

The reason is that we cannot estimate any coefficients when we have observations with a value of minus infinity.

While R can 'ignores' NA (missing) values, it does not drop infinity values automatically. To avoid these problematic observations, we add the subset argument to the lm function.

The last argument commands R to only run the regression on the subset of observation that have positive values for education and work experience.

```
lnwagefit2 <- lm(lnwage~lnexper+lneduc+gender, data=wages, subset = (XXXX > 0) & (XXXX > 0) )
summary(lnwagefit2)
```

Look at the interpretations. First, the p-values indicate that lnexper, lneduc and gender are all statistical significant at 1% significance level. Therefore, using a Log-Log model allows us to interpret coefficients as elasticities, i.e. A 1% increase in work experience will increase wages on average by 0.15 %, ceteris paribus; A 1% increase in education will increase wages on average by 0.95 %, ceteris paribus.

Task 4. Estimate the semi-log model below and interpret each coefficient:

$$log\, wage_i = \beta_1 + \beta_2 exper_i + \beta_3 educ_i + \delta male_i + u_i$$

The model we run is also call a log-lin model. Only the dependent variable is in logs, the independent variables are measured in levels:

```
slnwagefit1 <- lm(XXXX~XXXX+XXXX+gender, data=wages)
summary(slnwagefit1)
```

slnwagefit1: name this regression model as slnwagefit1, represents semi-log model

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

lnwage: the dependent variable

exper+educ+gender: the independent variables

data=wages: specify variable lnwage, exper, educ and gender are from dataset wages

Look at the interpretations.

First, the p-values indicate that exper, educ and gender are all statistical significant at 1% significance level.

The coefficients of a semi-log model are not elasticities!

One more year of work experience will increase wages on average by $100 \times 0.009 = 0.9$ %, ceteris paribus.
One more year of education will increase wages on average by $100 \times 0.091 = 9.1$ %, ceteris paribus.

Don't forget to multiply the coefficient by 100 to get the result measured in percentages.

Task 5. Generate variables that are the square of educ and exper.

We use the command line again to generate squared variables.

```
wages$expersq <- wages$XXXX^XXXX
wages$educsq <- wages$XXXX^XXXX
```

wages$expersq: generate a new variable called expersq in data set wages, to represent squared exper

wages$exper^2: calculate squared exper

wages$educsq: generate a new variable called educsq in data set wages, to represent squared educ

wages$educ^2: calculate squared educ

Task 6. Estimate the quadratic model below and interpret each coefficient

$$wage_i = \beta_1 + \beta_2 exper_i + \beta_3 {exper_i}^2 + \beta_4 educ_i + \beta_5 {educ_i}^2 + \delta male_i + u_i$$

We run now a quadratic model. Note that the dependent variable is now also measured in levels:

```
sqwagefit1 <- lm(wage~exper+XXXX+educ+XXXX+gender, data=wages)
summary(sqwagefit1)
```

sqwagefit1: name this regression model as sqwagefit1, represents quadratic model

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

wage: the dependent variable

exper+expersq+educ+educsq+gender: the independent variables

data=wages: specify variable wage, exper, expersq, educ, educsq and gender are from dataset wages

The marginal impact of work experience and education is not constant anymore. It will depend on the level of work experience and education an individual has got.

To calculate the marginal effect we have to differentiate the regression equation with respect to educ and exper.

$$\widehat{wage_i} = 0.7759 + 0.2630exper_i - 0.0047{exper_i}^2 - 0.3720educ_i + 0.0393{educ_i}^2 + 1.9716male_i$$

E.g., the marginal effect of work experience is:

$$\frac{d\,\widehat{wage}}{d\,exper} = \beta_2 + 2\beta_3 exper = 0.263 + 2 \times (-0.0047)exper = 0.263 - 2 \times 0.0047exper$$

$\beta_3 = -0.0047 < 0$, the marginal effect of work experience is diminishing with respect to higher levels of work experience.

Task 7. Estimate the model with the interaction term between gender and education and interpret each coefficient

$$wage_i = \beta_1 + \beta_2 exper_i + \beta_3 educ_i + \delta male_i + \lambda\, educmale_i + u_i$$

The λ in above's equation, the coefficient of the interaction term, is a slope dummy, that allows us to identify difference in the marginal effect of education on wages of men and women.

To add an interaction effect, we use a ':' for the variables (educ and gender) we want to interact:

```
intwagefit1 <- lm(wage~exper+educ+gender+XXXX:XXXX, data=wages)
summary(intwagefit1)
```

intwagefit1: name this regression model as intwagefit1, represents interactive explanatory variables model

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

wage: the dependent variable

exper+educ+gender+educ:gender: the independent variables, educ:gender represents the interaction term between gender and education

data=wages: specify variable wage, exper, educ and gender are from dataset wages

Note that the male dummy and the interaction term are statistically insignificant. I recommend to undertake an F-test to see if those variable have joint statistically significance or not.

Task 8. Use stargazer() to get the regression output

```
install.packages("stargazer")
#Install stargazer if you did not install in R workshop 4
#If you installed, just activate stargazer using library function
library(stargazer)
```

You can also use starfazer() to summarise the output for single model and multiple models.

```
stargazer(lnwagefit2, slnwagefit1, sqwagefit1, intwagefit1, type = "text")
```

More applications of stargazer() can be accessed on Blackboard under Week 9 in the R Workshop 4 section.

Final comments

We have created all our logarithmic variables, interaction terms and polynomials manually, luckily, there is a neat short-cut in R that lets you run the regression with skipping the step of variable creation. For example:

- Logarithmic model

```
lnwagefit3 <- lm(log(wage)~log(exper)+log(educ)+gender, data=wages, subset = (educ > 0) & (exper >0))
```

we applied log function directly.

- Quadratic model

```
sqwagefit2 <- lm(wage~exper+I(exper^2)+educ+I(educ^2)+gender, data=wages)
```

We have to use the I function to inhibit the interpretation of operators such as +, -, * and ^ as formula operators, so they are used as arithmetical operators.

- Interaction dummies model

```
intwagefit2 <- lm(wage~exper+educ*gender, data=wages)
```

The * between educ and gender can be interpreted as "include an intercept, all main effects and the interaction".

Further Exercise

For our final exercises, we will use a variety of different R datasets to practice the application of non-linear regression models.

1. Use R's mtcars dataset (type ?mtcars for more information) to estimate the following model:

$$mpg_i = \beta_1 + \beta_2 disp_i + \beta_3 {disp_i}^2 + u_i$$

where mpg_i stand for miles per gallon and $disp_i$ for engine displacement (in cubic inch) of car i.

To access the dataset use the data(mtcars) command. Interpret carefully all its coefficients.

2. Use the same dataset to estimate the following model:

$$log\, mpg_i = \beta_1 + \beta_2 log\, hp_i + \beta_3 am_i + u_i$$

where log stand for the natural Logarithm, mpg_i for miles per gallon, hp_i for horse power of car i, and am_i is a dummy indicating if the car has an automatic or manual gear. Interpret carefully all its coefficients.

3. Install and load the 'faraway' packages so that you can use the diabetes dataset (type ?diabetes for more information). Now estimate the following model:

$$chol_i = \beta_1 + \beta_2 age_i + \delta_1 frame_i{}^{med} + \delta_2 frame_i{}^{large} + \lambda_1 age_i \times frame_i{}^{med} + \lambda_2 age_i \times frame_i{}^{large} + u_i$$

where $chol_i$ is total Cholesterol, age_i is age in years, $frame_i$ is a factor with levels small, medium and large. Carefully interpret all the coefficients.