Chapter 8 - Linear regression

Dr. Alessandro Ruggieri

Contents

Univariate linear regression

Let's analyze the relation between log of GDP per capita and hours worked across countries.

- We first upload data on GDP per capita across countries:

```
# import data on GDP per capita across countries
data_gdp <- read.csv("GDP_205U_NOC_NB_A-filtered-2020-12-22.csv",  header = TRUE)
# rename columns
names(data_gdp) <- c("country", "label", "source", "year", "gdp")
```

- Then we upload data on weekly hours worked across countries:

```
# import new data:hours worked rate across countries
data_hours <- read.csv("HOW_TEMP_SEX_ECO_GEO_NB_A-filtered-2021-01-12.csv",  header = TRUE)
# rename columns
names(data_hours) <- c("country", "label", "source",
                       "gender","occupations","area",
                       "year", "weeklyhours", "status",
                       "note1", "note2")
# subset matrix according to gender groups: Sex=all gender
data_hours<-data_hours[data_hours$gender == "Sex: Total",]
# subset matrix according to occupations groups: occupations=Aggregate
data_hours<-data_hours[data_hours$occupations == "Economic activity (Aggregate): Total", ]
# subset matrix according to area groups: area=Aggregate
data_hours<-data_hours[data_hours$area == "Area type: National", ]
```

Finally we merge the two datasets:

```
# merge two data frames by country name and year
data_univariate <- merge(data_gdp,data_hours,by=c("country","year"))
```

We first compute the correlation coefficient between log real GDP per capita and hours worked:

```
# Correlation coefficient
cor(log(data_univariate$gdp),data_univariate$weeklyhours)
```

```
## [1] -0.4044041
```

The simple linear regression tries to find the best line to predict poverty rate on the basis of log GDP per capita. In this case, the univariate linear model equation can be written as follow:

$\text{hours-worked}_{it} = \beta_0 + \beta_1 \log \text{realGDPxcapita}_{it} + \epsilon_{it}$

We can estimate β_0 and β_1 using the R function function lm( )

```
# Univariate linear regression
bols <- lm(weeklyhours ~ log(gdp), data = data_univariate)
# display estimates
bols
```

```
##
## Call:
## lm(formula = weeklyhours ~ log(gdp), data = data_univariate)
##
## Coefficients:
## (Intercept)     log(gdp)
##      49.437       -1.025
```

We can display the statistical summary of the model using the R function summary( ):

```
# display model summary
model<-summary(bols)
model
```

```
##
## Call:
## lm(formula = weeklyhours ~ log(gdp), data = data_univariate)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -17.1843  -1.4336  -0.0302   1.5259   9.8537
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 49.43653    0.93873   52.66   <2e-16 ***
## log(gdp)    -1.02525    0.09194  -11.15   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.979 on 636 degrees of freedom
## Multiple R-squared:  0.1635, Adjusted R-squared:  0.1622
## F-statistic: 124.3 on 1 and 636 DF,  p-value: < 2.2e-16
```

To test the significance of the estimate for β_1, we can construct the appropriate t-statistics:

```
# Recover estimate coefficient
bols_1 <-model$coefficients["log(gdp)", "Estimate"]
# Recover estimate standard error
se_1 <-model$coefficients["log(gdp)", "Std. Error"]
# Construct t-stats
t_1 <- bols_1/se_1
print(t_1)
```

```
## [1] -11.15122
```

Multivariate linear regression

It is often the case that one explanatory variable is insufficient to explain the variation in the dependent variable. The multiple linear regression model allows for more than one explanatory variable.

Suppose we think that the poverty rate might affect aggregate labor supply. First we upload data on poverty rate across countries:

```r
# import new data:poverty rate across countries
data_pov <- read.csv("SDG_0111_SEX_AGE_RT_A-filtered-2021-01-09.csv",  header = TRUE)
# rename columns
names(data_pov) <- c("country", "label", "source","gender","age", "year", "povertyrate")
# subset matrix according to gender groups: Sex=all gender
data_pov<-data_pov[data_pov$gender == "Sex: Total",]
# subset matrix according to age groups: age=15+
data_pov<-data_pov[data_pov$age == "Age (Youth, adults): 15+", ]
```

Finally we merge the new data with previous dataset:

```r
# merge with previous data frame by country name and year
data_multivariate <- merge(data_univariate,data_pov,by=c("country","year"))
```

We can specify our multivariate linear model as follow:

hours-worked$_{it}$ = $\beta_0 + \beta_1$ log realGDPxcapita$_{it}$ + β_2 povertyrate$_{it}$ + ϵ_{it}

We can estimate β_0, β_1 and β_2 using the R function function lm( )

```r
# Multivariate linear regression
bols_multi <- lm(weeklyhours ~ log(gdp) + povertyrate, data = data_multivariate)
# display estimates
bols_multi
```

```
##
## Call:
## lm(formula = weeklyhours ~ log(gdp) + povertyrate, data = data_multivariate)
##
## Coefficients:
## (Intercept)     log(gdp)  povertyrate
##    46.02610     -0.51756     -0.08337
```

We can again display the statistical summary of the model using the R function summary( ):

```r
# display model summary
model_multi<-summary(bols_multi)
model_multi
```

```
##
## Call:
## lm(formula = weeklyhours ~ log(gdp) + povertyrate, data = data_multivariate)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -14.5081  -2.4569   0.0344   2.0924   8.7682
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 46.02610    2.73239  16.845  < 2e-16 ***
## log(gdp)    -0.51756    0.28629  -1.808   0.0716 .
## povertyrate -0.08337    0.01808  -4.610 5.88e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.431 on 311 degrees of freedom
```

```
## Multiple R-squared:  0.07523,    Adjusted R-squared:  0.06928
## F-statistic: 12.65 on 2 and 311 DF,  p-value: 5.228e-06
```

Finally, we can test the significance of the estimate for β_2, by constructing the appropriate t-statistics:

```r
# Recover estimate coefficient
bols_multi_3 <-model_multi$coefficients["povertyrate", "Estimate"]
# Recover estimate standard error
se_multi_3 <-model_multi$coefficients["povertyrate", "Std. Error"]
# Construct t-stats
t_multi_3 <- bols_multi_3/se_multi_3
print(t_multi_3)
```

```
## [1] -4.610234
```