

‘From Statistics to Data Mining’

Computer Lab Session n° 7:


Linear Regression (1/2)

Master 1 COSI / CPS²
Saint-Étienne, France

Fabrice Muhlenbach

Laboratoire Hubert Curien, UMR CNRS 5516
Université Jean Monnet de Saint-Étienne
18 rue du Professeur Benoît Luras
42000 SAINT-ÉTIENNE, FRANCE
<https://perso.univ-st-etienne.fr/muhlfabr/>

Outcome

The objective of this lab is to become familiar with  functions for working with linear regression.


1 Introduction

Regression analysis is used for explaining or modeling the relationship between a single variable Y , called the *response*, *output* or *dependent* variable, and one or more *predictor*, *input*, *independent* or *explanatory* variables, X_1, X_2, \dots, X_p . The case of one explanatory variable ($p = 1$) is called simple linear regression. For more than one explanatory variable ($p > 1$), it is called multiple linear regression.

We use regression to estimate the unknown effect of changing one variable over another. When running a regression we are making two assumptions:

1. there is a linear relationship between two variables (i.e., X and Y)
2. this relationship is additive (i.e., $Y = x_1 + x_2 + \dots + x_N$).

A regression with two or more explanatory variables is called a multiple regression. Rather than modeling the mean response as a straight line, as in simple regression, it is now modeled as a function of several explanatory variables.

The function `lm` can be used to perform multiple linear regression in  and much of the syntax is the same as that used for fitting simple linear regression models. To perform multiple linear regression with p explanatory variables use the command:

```
lm(response ~ explanatory_1 + explanatory_2 + ... + explanatory_p)
```

Here the terms `response` and `explanatory_i` in the function should be replaced by the names of the response and explanatory variables, respectively, used in the analysis.

2 Simple Linear Regression on Eucalyptus Dataset

Consider the following example: we are trying to estimate the height of eucalyptus trees based on their circumference (it is easier to measure the trunk of a tree than its height).

Import the eucalyptus dataset, print the summary of the dataset and plot the height (variable `ht` in meters) in function of the circumference (variable `circ` in cm).

```
euc <- read.table("eucalyptus.txt", header=T)
summary(euc)
plot(ht~circ, data=euc)
```

We compute then the linear regression, i.e., the estimation phase, by using the `lm` function. This function is used to fit linear models.

```
regeuc <- lm(ht~circ, data=euc)
summary(regeuc)
regeuc
```

In output, we have the information matrix on the coefficients with 4 columns and there are as many rows as coefficients.

The model $ht = \beta_0 + \beta_1 \times circ$ is then:

$$ht = 9.037476 + 0.257138 \times circ$$

With the function `attributes`, we can know the different attributes of the “lm object” `regeuc`.

```
attributes(regeuc)
```

The coefficient are stored in the attribute `coefficients` and we can access to them with `regeuc$coefficients` or more simply with `coef(regeuc)`.

To examine the quality of the model and observations, we plot the fitted line and observations. As there is an uncertainty in the estimates, we also draw a confidence interval of 95% for the prediction. First, we plot the line obtained by the linear regression model (function `abline`). Then we define the scale the X-axis. After that, we will use the `predict` function for obtaining the predictions of the model with a confidence interval of 95%. Finally we can plot the columns of the circumference matrix against the columns of the confidence interval for the prediction.

```
abline(regeuc, col="red")
circ=seq(min(euc[, "circ"]), max(euc[, "circ"]), length=100)
grid<-data.frame(circ)
CIpred<-predict(regeuc, new=grid, interval="pred", level=0.95)
matlines(grid$circ, cbind(CIpred), lty=c(1, 2, 2), col=2)
```

We will now try a multiple linear regression model $ht = \beta_0 + \beta_1 \times circ + \beta_2 \times \sqrt{circ}$

```

multreg<-lm(ht~circ+I(sqrt(circ)),data=euc)
summary(multreg)
plot(ht~circ,data=euc)
circ=seq(min(euc[, "circ"]),max(euc[, "circ"]),length=100)
grid2<-data.frame(circ)
CIpred2<-predict(multreg,new=grid2,interval="pred",level=0.95)
matlines(grid2$circ,cbind(CIpred2),lty=c(1,2,2),col=2)

```

By plotting directly the model, we can obtain:

- the residuals vs. the fitted values
- the standardized residuals in function of the theoretical quantiles (normal Q-Q)
- the square root of the standardized residuals in function of the fitted values (scale location)
- the residuals vs. the leverage

```

win.graph(800,600,10)
plot(regeuc)
plot(multreg)

```

3 Simple and Multiple Linear Regressions on Ozone Dataset

Ozone is an inorganic compound with the chemical formula O_3 . Ozone precursors are a group of pollutants, predominantly those emitted during the combustion of fossil fuels. Exposure to ozone and the pollutants that produce it is linked to premature death, asthma, bronchitis, heart attack, and other cardiopulmonary problems. Air quality guidelines such as those from the World Health Organization, the United States Environmental Protection Agency (EPA) and the European Union are based on detailed studies designed to identify the levels that can cause measurable ill health effects. According to scientists with the US EPA, susceptible people can be adversely affected by ozone levels as low as 40 nmol/mol. In the EU, the current target value for ozone concentrations is 120 $\mu\text{g}/\text{m}^3$ which is about 60 nmol/mol.

The file `ozone.txt` concerns the maximal concentration of ozone in the air recorded at Rennes (in Brittany, France) for each day during the summer 2001 (“maxO3”) and the day before (“maxO3v”), and different measures of the weather: temperature (T), cloudiness (Ne for « nébulosité ») and wind speed (Vx) at 9:00 AM, 12:00 AM and 3:00 PM (9, 12, 15), maximal wind direction (wind), presence of rain or not (rain).

Import ozone dataset, plot the pairwise representation between the 11 first numerical variables and print the correlation coefficients between the 11 first numerical variables.

```

ozone <- read.table("ozone.txt",header=T)
summary(ozone)

win.graph(800,600,10)
pairs(ozone[1:11])
cor(ozone[1:11])

```

With which variable the maximal ozone concentration (`maxO3`) is the most correlated?

Plot `maxO3` in function of this variable.

Compute the simple linear model of `maxO3` in function of this variable, then trace the line obtained by this model on the graph.

Consider now all numerical variable (except `maxO3`) and compute the multiple linear regression of `maxO3` in function of these variables.

Plot then the results obtained for the simple and the multiple linear regressions on `ozone` dataset.

4 Recommended Readings

The main literature for this section is:

- Adler (2010), “R in a Nutshell – a Desktop Quick Reference,” Chapter 20 “Regression Models”.
- Cohen and Cohen (2008), “Statistics and Data with R,” Chapter 14 “Simple linear regression”.
- Cornillon and Matzner-Løber (2010), « Régression avec R » (in French).
- Dalgaard (2008), “Introductory Statistics with R,” Chapter 6 “Regression and correlation.”

References

Adler, J. (2010). *R in a Nutshell – a Desktop Quick Reference*. O'Reilly.

Cohen, Y. and J. Y. Cohen (2008). *Statistics and Data with R: An Applied Approach Through Examples*. John Wiley & Sons, Ltd.

Cornillon, P.-A. and E. Matzner-Løber (2010). *Régression avec R*. Pratique R. Springer. [Book available at the library of the Faculty of Science and Technology].

Dalgaard, P. (2008). *Introductory Statistics with R* (2nd ed.). Springer.