

Analysing Social Data Using R

Linear regression, part 2

Michał Bojanowski¹, mbojan@icm.edu.pl

Zbigniew Karpiński², zkarpinski@ifispan.waw.pl

¹Interdisciplinary Centre for Mathematical and Computational Modelling

²Institute for Philosophy and Sociology

May 14, 2012

Lists

- List is yet another object in R.
- Lists are “generalised vectors” that can comprise elements of various classes and types: numbers, characters, matrices, data frames, other lists, etc.

List: an example

```
num <- 1:4
mat <- matrix(1:4, ncol = 2)
dst <- data.frame(x = 1:2, y = 3:4)
lst <- list(vector = num, matrix = mat, 'data frame'
= dst)
str(lst)

## List of 3
## $ vector      : int [1:4] 1 2 3 4
## $ matrix      : int [1:2, 1:2] 1 2 3 4
## $ data frame:'data.frame': 2 obs. of  2 variables:
## ..$ x: int [1:2] 1 2
## ..$ y: int [1:2] 3 4
```

List: an example

```
lst

## $vector
## [1] 1 2 3 4
##
## $matrix
##      [,1] [,2]
## [1,]    1    3
## [2,]    2    4
##
## $`data frame`
##   x y
## 1 1 3
## 2 2 4
```



Subsetting a list

```
# using []  
lst[1]  
  
## $vector  
## [1] 1 2 3 4  
##  
  
# using [[]]  
lst[[1]]  
  
## [1] 1 2 3 4
```

Subsetting a list (continued)

```
# using the element name
lst[["matrix"]]

##      [,1] [,2]
## [1,]    1    3
## [2,]    2    4

# using $
lst$vector

## [1] 1 2 3 4
```

lm() object as a list

The output returned by the function `lm()` is a list:

```
## lm(formula = log(do.earn) ~ schooling + age + I(age^2),  
  
## [1] "coefficients" "residuals" "effects"  
## [4] "rank" "fitted.values" "assign"  
## [7] "qr" "df.residual" "na.action"  
## [10] "xlevels" "call" "terms"  
## [13] "model"
```

lm() object as a list

Using `summary(RegModel)` also returns a list:

```
## [1] "call"          "terms"         "residuals"
## [4] "coefficients"  "aliased"       "sigma"
## [7] "df"           "r.squared"     "adj.r.squared"
## [10] "fstatistic"    "cov.unscaled" "na.action"
```


Extracting coefficients from the `lm()` object

```
null[["coefficients"]]
```

```
## (Intercept)      schooling          age      I(age^2)
##    4.3321942    0.1061351    0.0314809   -0.0003412
```

Extracting coefficients from the summary(RegModel) object

```
summary(null)[["coefficients"]]
```

##	Estimate	Std. Error	t value	Pr(> t)
## (Intercept)	4.3321942	1.251e-01	34.639	5.139e-243
## schooling	0.1061351	3.646e-03	29.110	4.606e-176
## age	0.0314809	5.935e-03	5.305	1.163e-07
## I(age^2)	-0.0003412	7.129e-05	-4.786	1.739e-06

Model frame

- Model frame is an auxiliary data frame which contains **all** and **only** the variables and observations that were used in the regression analysis.
- The model null uses only 3 variables: `should.earn`, `schooling`, and `age`.
- The original PGSS dataset contains hundreds of variables. The function `lm()` extracts these 3 variables into a model frame.
- If the variables used in the regression analysis contain missing data, the missing data are removed from the model frame.

Extracting coefficients from the summary(RegModel) object

```
mf <- model.frame(null)
head(mf, 3)

##    log(do.earn) schooling age I(age^2)
## 2          4.605         10  43     1849
## 3          5.298         12  27      729
## 4          4.700         12  36     1296

c(model.frame = nrow(mf), data.frame = nrow(p))

## model.frame  data.frame
##          7084         16234
```