

An introduction to R

R markdown basics

Samy Zitouni

October 2024

About today

Sampling

- Understand the notion of **sampling**, and dealing with **real world data**
- How it links with random variable (**RV**) and why the following elements are RV:
 - **OLS** estimators
 - Basic statistics
- Use our recent knowledge of **plotting** to highlight it

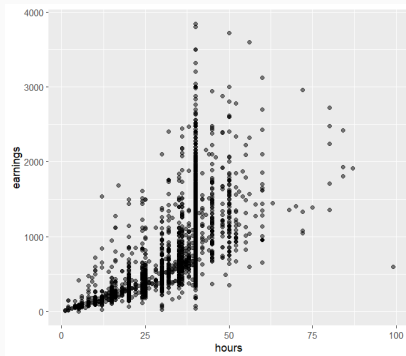
First regressions

- Perform several regressions
- **Interpret** results

Getting started

Data

- Download the data and load it in R
- First, we are interested in the relationship between **hours worked** and **earnings**



Law of large numbers

Theorem

For a series of random variables X_1, \dots, X_n , of the same distribution.
Then,

$$\mu_n = \frac{1}{n} \sum_{i=1}^n X_i \xrightarrow{n \rightarrow \infty} \mathbb{E}[X]$$

Let's show it

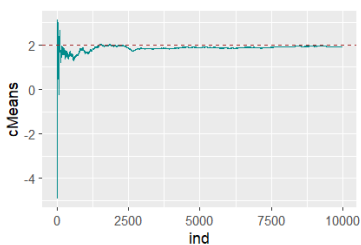
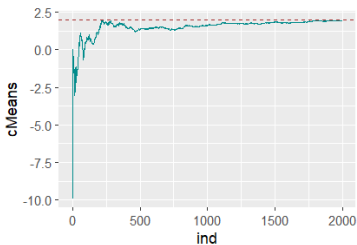
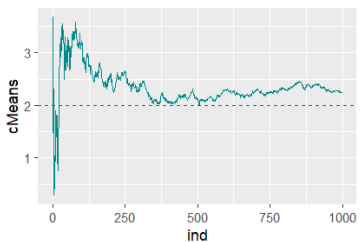
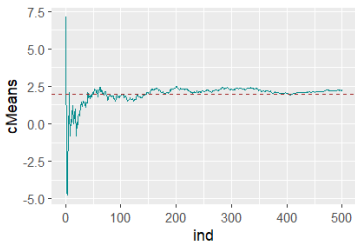
Monte Carlo Method : idea

- Generate a sample X_1, \dots, X_n of the same law
- Compute the mean of sample (X_1) , then the mean of sample (X_1, X_2) , then sample (X_1, \dots, X_i) until the full sample.
- The series of means should converge to $\mathbb{E}[X]$

Bootstrapping

- Generate a sample X_1, \dots, X_n of the same law
- Compute the mean of sample (X_1) , then the mean of sample (X_i, X_j) randomly drawn, then sample $(X_{i_1}, \dots, X_{i_p})$ until the full sample.
- You can also do it with p random sample of a fixed size k
- The series of means should converge to $\mathbb{E}[X]$

Law of large numbers



Regression analysis

Theory and empirics

- We want to estimate the relationship between two (for now) variables, X and Y
- We suppose that they are **linked** with a **theoretical** linear relationship: $Y = \alpha + \beta X + \epsilon$, this relationship is **always true** as ϵ can vary a lot
- **Our goal** is to estimate $\hat{\alpha}$ and $\hat{\beta}$, that are the *estimators* of our line.

Regression analysis

- $\hat{\alpha}$ and $\hat{\beta}$ **draw** our **regression line**
- Said otherwise, the regression line **pass through** all $\hat{y}_i = \hat{\alpha} + \hat{\beta}x_i$, that are predicted ordinates
- There is an **distance** $\hat{\epsilon}_i$ between the predicted value \hat{y}_i and the true data value y_i : $\hat{\epsilon}_i = y_i - \hat{y}_i$

Remember

- The **OLS** estimator is defined by:

$$(\hat{\alpha}, \hat{\beta}) = \underset{a,b}{\operatorname{argmin}} \sum_{i=0}^N Y_i - (a + bX_i)$$

- The resolution of this problem gives:

$$\hat{\beta} = \frac{\widehat{\operatorname{cov}}(X, Y)}{\widehat{V}(X)}$$

$$\hat{\alpha} = \bar{Y} - \hat{\beta}\bar{X}$$

Graphical example

Using previous classes

- Let's plot earnings as a function of hours worked
- Compute $\hat{\alpha}$, $\hat{\beta}$ and add a column `hatEarnings` (\hat{y}_i) to the data
- Add the `hatEarning` dots to the plot, change the color to highlight them
- Add a line
- Plot the same graph, and replace the `geom_line` with `geom_abline` (automatic regression plotting)

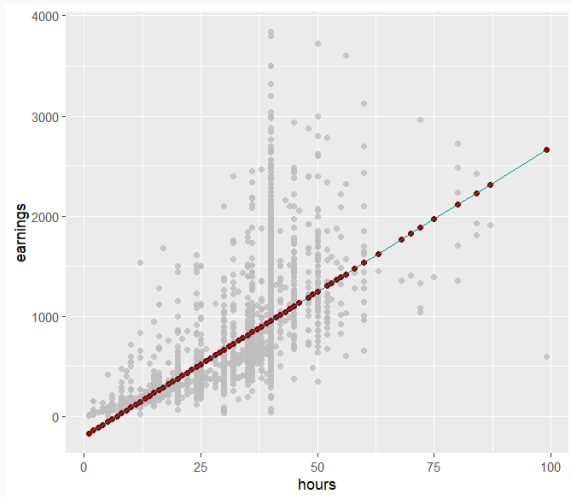
Graphical example

Solution

```
beta <- cov(data$earnings, data$hours)/var(data$hours)
alpha <- mean(data$earnings) - beta*mean(data$hours)

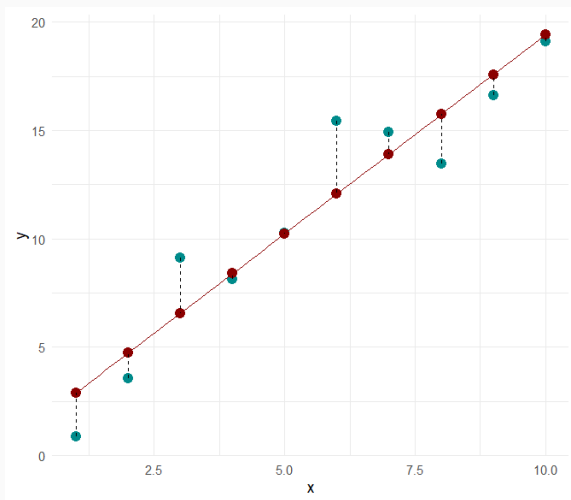
data %>%
  mutate(
    hatEarnings = alpha + beta*hours
  )%>%
  ggplot() +
  geom_point(aes(x = hours, y = earnings), alpha = .8, color = 'grey') +
  geom_point(aes(x=hours, y = hatEarnings), color = "darkred")+
  geom_line(aes(x=hours, y = hatEarnings), color = "darkcyan")
```

Result



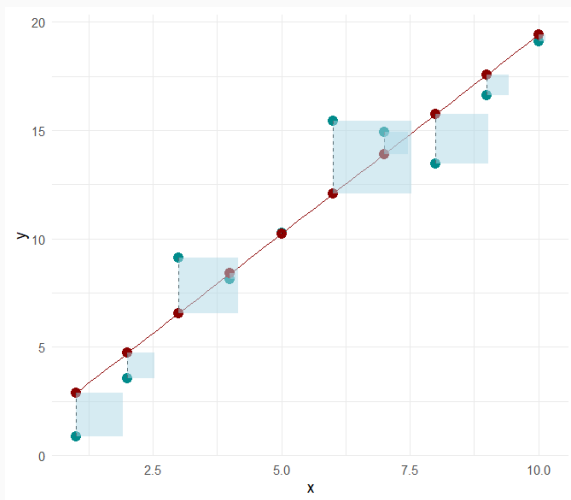
Errors

The goal is to minimize squared errors



Errors

The goal is to minimize squared errors



About sampling

About sampling

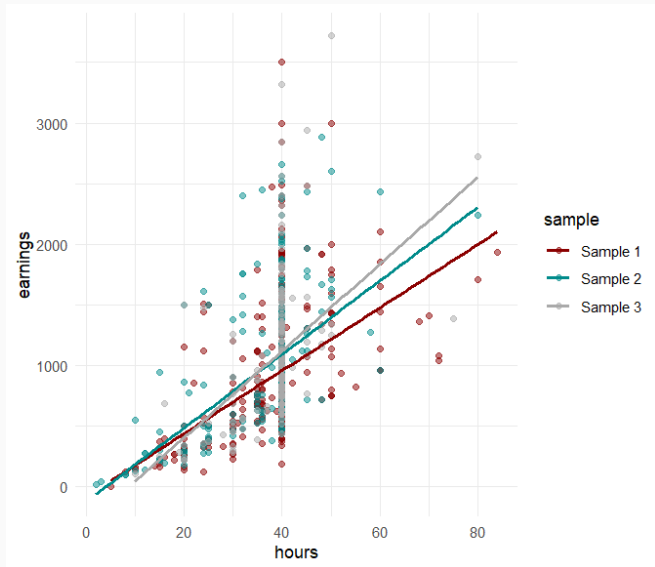
- From three different (not so) random sample, we can get three different regression lines

```
d1 <- sample_n(data, size = 500, weight = hours)
d2 <- sample_n(data, size = 300, weight = hourrt)
d3 <- sample_n(data, size = 200, weight = earnings)
```

```
d1$sample <- "Sample 1"
d2$sample <- "Sample 2"
d3$sample <- "Sample 3"
```

```
d <- bind_rows(d1, d2, d3) #Join data together
```

About sampling



A small exercise

Compute coefficients

- Create a function that takes in entry:
 - A dataframe
 - Two column names (the ones you want to perform the regression on)
- And gives both coefficients $\hat{\alpha}$ and $\hat{\beta}$ for the regression $Y = \alpha + \beta X + \epsilon$

Example

```
computeCoefficients <- function(data, Y, X){  
  #In: The data you will use the Y and X of your regression  
  b <- cov(data[Y], data[X])/var(data[X])  
  a <- mean(data[[Y]]) - b*mean(data[[X]])  
  res <- t(data.frame(c(a, b)))  
  colnames(res) <- c('alpha', 'beta')  
  return(res)  
}  
  
computeCoefficients(d1, "earnings", "hours")
```

Sample	$\hat{\alpha}$	$\hat{\beta}$
d1	-82.9	26.01
d2	-121.3	30.34
d3	-314.02	35.8

About sampling

- All discrete statistics, as \bar{X} , $\hat{\beta}$, $\hat{V}(X)$ vary with the sample $X = (X_1, \dots, X_n)$.
- As we can draw a sample, we indirectly draw estimators
- This is why we are able to compute their Variance, Expectation, etc.
- For example: $\mathbb{E}[\bar{X}] = \mathbb{E}[X]$

About sampling

- That being said, we can **draw conclusions** about our estimators $\hat{\alpha}$ et $\hat{\beta}$, considering that we have our base hypothesis:
 1. $\forall n = 1, \dots, N, \mathbb{E}[\epsilon_n] = 0$
 2. $\forall n = 1, \dots, N, \mathbb{V}[\epsilon_n] = \sigma^2$
 3. $\forall n \neq m, \text{cov}(\epsilon_n, \epsilon_m) = 0$
- For example, β is **unbiased**: $\mathbb{E}[\hat{\beta}] = \beta$
- In the case of the **Gaussian Model** $\epsilon_n \sim \mathcal{N}(0, \sigma^2)$, we can have a *distribution* for $\hat{\beta} \sim \mathcal{N}(\beta, \sigma_{\hat{\beta}}^2)$ and $\hat{\alpha} \sim \mathcal{N}(\alpha, \sigma_{\hat{\alpha}}^2)$, with known values for the variances.
- **Increasing the sample size** means getting closer to the true laws, and that defines convergences as the **Central Limit Theorem**, the **Law of Large Numbers**, etc.

Regressions with R

Regressions with R

In practice

- There are **built functions** that can perform the regression for you
- For a simple regression, we use **lm** (linear model)
- A regression is an object you can store, it has attributes
- **summary()** gives you main informations

Example with the work data

```
reg1 <- lm(fml = earning ~ hours, data)
summary(reg1)
```

Regressions with R

```
call:
lm(formula = y ~ x, data = data)

Residuals:
    Min       1Q   Median       3Q      Max
-2.2695 -1.1248 -0.2785  0.7707  3.3628

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)    1.0509     1.3346   0.787   0.454
x              1.8361     0.2151   8.537 2.73e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.954 on 8 degrees of freedom
Multiple R-squared:  0.9011,    Adjusted R-squared:  0.8887
F-statistic: 72.87 on 1 and 8 DF,  p-value: 2.729e-05
```

Into details

	<i>Dependent variable:</i>
	earnings
hours	28.908*** (0.591)
Constant	−198.809*** (21.915)
Observations	4,609
R ²	0.342
Adjusted R ²	0.342
Residual Std. Error	393.284 (df = 4607)
F Statistic	2,391.645*** (df = 1; 4607)

Note: *p<0.1; **p<0.05; ***p<0.01

- With our previous function we found:

$$\hat{\alpha} = -198.8, \quad \hat{\beta} = 28.9$$

Into details

- **Coefficients:** the estimator values for our data sample
- **StdE, stars:** level of significativity given by a t-test
- **R^2 :** The part of the total variance that is explained by the model
- **Adjusted R^2 :** the same, but weighted by the number of independent variables
- **Residual Std. Error:** The standard error of the model (degrees of freedom: df)
- **F statistic:** a test statistic that is usually used to see if your model explains at least something

Regression, summary attributes

- The regression object `reg1` as well as the summary object `summary(reg1)` both have attributes.
- We can use it to `plot residuals` for example !

List of attributes of the regression object

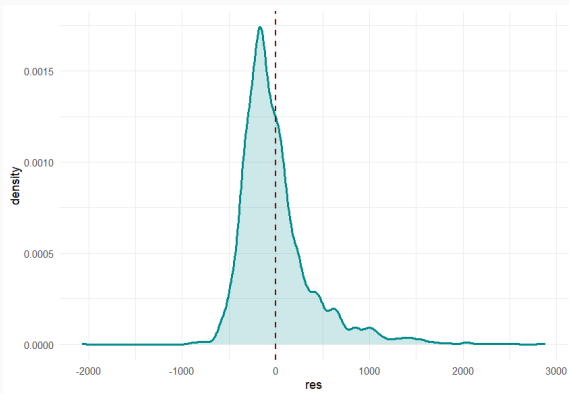
```
> str(reg1, give.attr = F)
List of 12
 $ coefficients : Named num [1:2] 1.05 1.84
 $ residuals    : Named num [1:10] -2.0079 -1.1834 2.5583 -0.2541 0.0274 ...
 $ effects      : Named num [1:10] -35.257 16.677 3.127 0.254 0.474 ...
 $ rank         : int 2
 $ fitted.values: Named num [1:10] 2.89 4.72 6.56 8.4 10.23 ...
 $ assign       : int [1:2] 0 1
 $ qr           :List of 5
 ..$ qr        : num [1:10, 1:2] -3.162 0.316 0.316 0.316 0.316 ...
 ..$ qraux: num [1:2] 1.32 1.27
 ..$ pivot: int [1:2] 1 2
 ..$ tol : num 1e-07
 ..$ rank : int 2
 $ df.residual  : int 8
 $ xlevels      : Named list()
 $ call         : language lm(formula = y ~ x, data = data)
 $ terms        :classes 'terms', 'formula' language y ~ x
 $ model        :'data.frame': 10 obs. of 2 variables:
 ..$ y: num [1:10] 0.879 3.54 9.117 8.141 10.259 ...
 ..$ x: int [1:10] 1 2 3 4 5 6 7 8 9 10
> |
```

List of attributes of the summary object

```
> str(summary(reg1), give.attr = F)
List of 11
 $ call      : language lm(formula = y ~ x, data = data)
 $ terms     :Classes 'terms', 'formula' language y ~ x
 $ residuals : Named num [1:10] -2.0079 -1.1834 2.5583 -0.2541 0.0274 ...
 $ coefficients : num [1:2, 1:4] 1.051 1.836 1.335 0.215 0.787 ...
 $ aliases    : Named logi [1:2] FALSE FALSE
 $ sigma      : num 1.95
 $ df         : int [1:3] 2 8 2
 $ r.squared  : num 0.901
 $ adj.r.squared: num 0.889
 $ fstatistic  : Named num [1:3] 72.9 1 8
 $ cov.unscaled : num [1:2, 1:2] 0.4667 -0.0667 -0.0667 0.0121
> |
```


Residuals

```
data.frame(res = reg1$residuals) %>% # Creates a dataframe with residuals as a single column  
ggplot(aes(x = res)) + # We can now use ggplot on the dataframe using dplyr  
  geom_density(color = 'darkcyan', fill = 'darkcyan', alpha = .2, linewidth = 1)+  
  geom_vline(xintercept = 0, linetype = 'dashed', color = 'darkred', linewidth = .8) +  
  theme_minimal()
```




Coefficient interpretation

Super important

- Depends on the framework you are in (log - log, etc.)
- Being wrong changes completely your results

Choosing the appropriate transformation

- Most of the time, log transformations are used
- They **compress** high values, and can be understood in terms of **utility**. For instance, $\log(\text{salary})$ can mean that workers have a smaller gain in utility going from 5000 to 5500 per month than from 1000 to 1500 (*decreasing marginal utility*)
- They have another interpretation
- You can find keys for interpretation at the end of the slide 

Our results

$$\hat{\alpha} = -198.8, \quad \hat{\beta} = 28.9$$

Coeff.	Value	Interpretation
α	-198.8	
β	28.9	

Our results

$$\hat{\alpha} = -198.8, \quad \hat{\beta} = 28.9$$

Coeff.	Value	Interpretation
α	-198.8	Working 0 hours imply no money (negative salary in fact)
β	28.9	An additional hour worked implies more 28.9\$

- **Beware** about interpretation of α . sometimes, it is **difficult to link it with reality** (would someone really work 0 hour? What about social minimas, etc.)

Binary Variable

Set-up

Let's assume that we want to explain **earnings** by the variable **sex**, 1 for man and 0 for woman.

- A first look at `data$sex` gives us the values "Male" or "Female"
- Even though R is capable of performing regressions on categorical variables, to link it with theory, let's recode the variable **sex**

```
data <- data %>% mutate(sexNum = ifelse(sex=="Male", 1, 0))
```

Binary variables

Let's assume that we want to explain **earnings** by the variable **sex**,
1 for man and 0 for woman.

$$\text{earnings} = a + b\text{sex} + \text{epsilon}$$

A bit of theory

$$\begin{aligned} & E[\text{earnings} \mid \text{Sex} = 1] - E[\text{earnings} \mid \text{Sex} = 0] \\ &= a + b - (a + b \times 0) \\ &= b \end{aligned}$$

- **Interpretation:** In expectation, being a male (sex=1) implies b more \$ of salary than for being a woman (sex = 0), all other things equal.

Binary variables

Let's assume that we want to explain **earnings** by the variable **sex**,
1 for man and 0 for woman.

$$\text{earnings} = a + b\text{sex} + \text{epsilon}$$

A bit of theory

$$E[\text{earnings} \mid \text{Sex} = 0] = a$$

- **Interpretation:** a is the mean of the reference group
- **Therefore:** b is the difference in means

Graphical interpretation

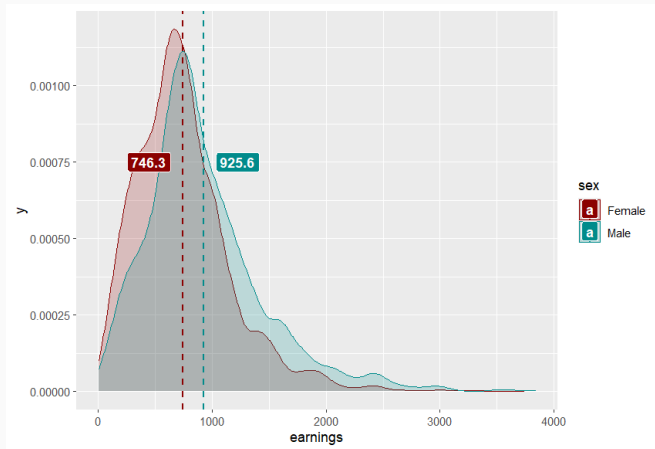
Try to plot the earnings distribution as a function of the sex.

```
sumData <- data %>%
  group_by(sex) %>%
  summarize(m.grp = mean(earnings, na.rm = T))

data %>%
  ggplot(aes(x = earnings, fill = sex, color = sex)) +
  geom_density(alpha = .2) +
  scale_color_manual(values = c('darkred', 'darkcyan')) +
  scale_fill_manual(values = c('darkred', 'darkcyan')) +
  geom_vline(
    data = sumData, aes(xintercept = m.grp, color = sex),
    linewidth = .6, linetype = 'dashed') +
  geom_label(
    data = sumData,
    aes(x = m.grp + (-1)^(sex=="Female")*300,
        y = 0.00075,
        label = as.character(round(m.grp, 1))),
    color = 'white', fontface = 'bold')
```

See that I used the categorical value for sex

Graphical interpretations



Regression

Let's regress earnings on sex, with both the categorical character variable and the binary one `sexnum`

Regress earnings on sex

```
reg2 <- lm(earnings ~ sex, data)
summary(reg2)

reg3 <- lm(earnings ~ sexNum, data)
summary(reg3)
```

Regression

```
> reg2 <- lm(earnings ~ sex, data)
> summary(reg2)

Call:
lm(formula = earnings ~ sex, data = data)

Residuals:
    Min       1Q   Median       3Q      Max
-921.6  -314.3  -75.6   216.4  3053.7

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  746.256     9.862   75.67  <2e-16 ***
sexMale      179.346    14.035   12.78  <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 476.4 on 4607 degrees of freedom
Multiple R-squared:  0.03423,    Adjusted R-squared:  0.03402
F-statistic: 163.3 on 1 and 4607 DF,  p-value: < 2.2e-16

> reg3 <- lm(earnings ~ sexNum, data)
> summary(reg3)

Call:
lm(formula = earnings ~ sexNum, data = data)

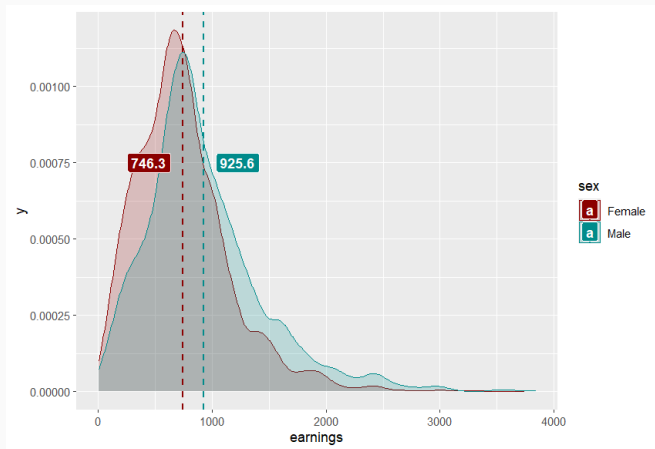
Residuals:
    Min       1Q   Median       3Q      Max
-921.6  -314.3  -75.6   216.4  3053.7

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  746.256     9.862   75.67  <2e-16 ***
sexNum      179.346    14.035   12.78  <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Regression

- Notice how R produces strictly the **same** regression
- It **automatically defines** a reference category.
- Above, the coefficient associated with **sexMale** tells us R defines **Female** as the reference category
- It represents the **difference in means**, all other things being equal.

Graphical interpretations



Our regression gives $\hat{\alpha} = 746.3$ and $\hat{\beta} = 179.3 = 925.6 - 746.3$

Broader categorical variables

More than two values

```
data$education
```

```
> unique(data$educ)
[1] "High school"      "Bachelor's degree" "No high school"    "Associate degree"
```

- A n levels category variable is equivalent to $n - 1$ dummy variables
- Commonly called one hot encoding

educ	Bachelor's degree	High school	No high school
High school	0	1	0
Bachelor's degree	1	0	0
No high school	0	0	1
Associate degree	0	0	0

Integration in R

- Of course, R does not need us to encode every time we want to perform a regression analysis
- It is possible to run regressions directly and the coefficient associated with **variableCategory** will describe the difference in means w.r.t the reference category (the only one not printed)

Let's practice

```
reg4 <- lm(earnings ~ educ, data)
summary(reg4)
```

Integration in R

```
> reg4 <- lm(earnings ~ educ, data)
> summary(reg4)
```

Call:

```
lm(formula = earnings ~ educ, data = data)
```

Residuals:

Min	1Q	Median	3Q	Max
-1006.06	-304.01	-64.01	193.29	2829.94

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)	
(Intercept)	904.01	19.47	46.430	< 2e-16	***
educBachelor's degree	106.06	24.65	4.302	1.72e-05	***
educHigh school	-97.30	21.58	-4.509	6.68e-06	***
educNo high school	-328.85	28.19	-11.665	< 2e-16	***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 468.9 on 4605 degrees of freedom
Multiple R-squared: 0.06466, Adjusted R-squared: 0.06405
F-statistic: 106.1 on 3 and 4605 DF, p-value: < 2.2e-16

Integration in R

Reference category

- See how the only category not printed is **Associate degrees**
- **Key for reading:** In **expectation**, *all other things equal*, a Bachelor's degree implies 106.06\$ more than an Associate degree.

Change the reference category

- Can be **useful** to gain *interpretation power*
- **Factors:** allow us to define implicit orders for variables (levels)
- Levels are **only implicit**, R knows they do not have a specific signification

Integration in R

Re-level all your data

```
data <- data %>%  
  mutate(  
    educf = factor(educ,  
      levels = c("No high school",  
        "High school",  
        "Associate degree",  
        "Bachelor's degree"))  
  )
```

Only set the reference level

```
data <- data %>%  
  mutate(  
    educf = relevel(as.factor(educ), ref = "No high school")  
  )
```

Your column **does not change at all**, but R understand there is an implicit order for some funtions (regression, ggplot, etc.)

Multivariate regressions

Set-up

Interest

- Performing **one** regression at a time is **not productive**
- **Multivariate regressions** allow us to **disentangle** every effect.

Frisch - Waught's theorem, idea :

- In the basic setup $Y = \alpha + \beta X + \epsilon$, the error term ϵ may **encapsulates other effects** than β
- Writing $Y = \alpha + \beta X_1 + \gamma X_2 + \mu$ allows to highlight effect γ and "**strip it** out of ϵ ". Then, if X_1 and X_2 are correlated, there is a chance that the effect for X_1 changes, it would be **cleaned** by X_2

This theorem is one of the solution for the **omitted variable bias problem**

Model and Theorem

Consider a multiple regression model:

$$y = X_1\beta_1 + X_2\beta_2 + \varepsilon,$$

where:

- y is the vector of the dependent variable ($n \times 1$),
- X_1 ($n \times k_1$) and X_2 ($n \times k_2$) are matrices of explanatory variables,
- β_1 and β_2 are vectors of coefficients,
- ε is the error term.

Theorem: The coefficients $\hat{\beta}_1$ can be obtained in three steps:

1. Regress y on X_2 and obtain the residuals r_y .
2. Regress X_1 on X_2 and obtain the residuals R_{X_1} .
3. Regress r_y on R_{X_1} . The coefficients are $\hat{\beta}_1$.

Demonstration

The initial model is:

$$y = X_1\beta_1 + X_2\beta_2 + \varepsilon.$$

1. Project y onto X_2 : $P_2 = X_2(X_2'X_2)^{-1}X_2'$.

$$r_y = (I - P_2)y.$$

2. Project X_1 onto X_2 :

$$R_{X_1} = (I - P_2)X_1.$$

3. Regress r_y on R_{X_1} :

$$\hat{\beta}_1 = (R_{X_1}'R_{X_1})^{-1}R_{X_1}'r_y.$$

This approach is equivalent to directly estimating the full model!

- The residuals r_y represent the part of y that is not explained by X_2 .
- The residuals R_{X_1} represent the part of X_1 that is not explained by X_2 .
- Regressing r_y on R_{X_1} measures the net effect of X_1 on y .

Advantage: This theorem allows for interpreting coefficients in complex contexts.

A Simple multivariate regression

An example

- Let's regress **earnings** on **sex**, **educ**, **hours**
- We **assume** the relationship for individual i is given by:

$$\text{earnings}_i = \beta_0 + \beta_1 \text{sex}_i + \beta_2 \text{educ}_i + \beta_3 \text{hours}_i + \epsilon_i$$

Let's run it

```
multiReg <- lm(earnings ~ sex + educ + hours, data)
summary(multiReg)

data <- data %>%
  mutate(
    logEarnings = log(earnings)
  )

multiRegLog <- lm(logEarnings ~ sex + educ + hours, data)
summary(multiRegLog)
```

A simple multivariate regression (level - level)

```
Call:
lm(formula = earnings ~ sex + educf + hours, data = data)

Residuals:
    Min       1Q   Median       3Q      Max
-1976.4  -232.1   -75.3   140.6  2707.6

Coefficients:
                Estimate Std. Error t value Pr(>|t|)
(Intercept)    -370.0241    24.6203  -15.029 < 2e-16 ***
sexMale         118.6189     11.3734   10.430 < 2e-16 ***
educfAssociate degree 220.7786     22.7767    9.693 < 2e-16 ***
educfBachelor's degree 367.6253     20.5479   17.891 < 2e-16 ***
educfHigh school 114.6405     18.0911    6.337 2.57e-10 ***
hours           27.3701      0.5789   47.280 < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 374.4 on 4603 degrees of freedom
Multiple R-squared:  0.4041,    Adjusted R-squared:  0.4034
F-statistic: 624.2 on 5 and 4603 DF,  p-value: < 2.2e-16
```

A simple multivariate regression (level - level)

- *All other things equal in our model*, men earn in average 118\$ more than women (monthly)
- *All other things equal in our model*, an Associate degree allows 220\$ more monthly than no High School
- *All other things equal in our model*, an Bachelor's degree allows 367\$ more monthly than no High School

A simple multivariate regression (level - level)

A recall on Frisch Waugh's theorem

- In the **simple regression** framework $earnings = \alpha + \beta Male + \epsilon$, we had $\hat{\beta} = 179$
- In the **multivariate regression** framework $earnings = \beta_0 + \beta_1 Male + \beta_2 educf + \beta_3 hours + \epsilon$, we have $\hat{\beta}_1 = 118$, which is **lower**
- Somehow, we have cleaned the coefficient, maybe getting closer to **causality**
- It is the **all other things equal** that changes, meaning that the difference on all people between men and women is lower than the difference on all people that have the same degree and work the same hours

A simple multivariate regression (log - level)

Call:

```
lm(formula = logEarnings ~ sex + educf + hours, data = data)
```

Residuals:

Min	1Q	Median	3Q	Max
-3.8581	-0.2353	-0.0236	0.2299	1.8256

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)	
(Intercept)	4.4561736	0.0269149	165.565	< 2e-16	***
sexMale	0.0848793	0.0124334	6.827	9.82e-12	***
educfAssociate degree	0.2955869	0.0248995	11.871	< 2e-16	***
educfBachelor's degree	0.4502549	0.0224630	20.044	< 2e-16	***
educfHigh school	0.1888843	0.0197772	9.551	< 2e-16	***
hours	0.0506223	0.0006329	79.991	< 2e-16	***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.4093 on 4603 degrees of freedom

Multiple R-squared: 0.6252, Adjusted R-squared: 0.6248

F-statistic: 1535 on 5 and 4603 DF, p-value: < 2.2e-16

Remember...

With X , Y and β being matrices:

$$(XX')\hat{\beta} = X'Y$$

What it means

- The OLS estimator $\hat{\beta}$ is defined **if and only if** the (XX') matrix is invertible.
- This is **equivalent to say that** $rg(X) = N$ with N being the number of regressors (including the constant)
- It means that no variable should be a linear combination of others

Co-linearity: example

Imagine that the columns of X are $X = (1, \text{Male}, \text{Female}, \dots)$

$$X = \begin{bmatrix} 1 & 0 & 1 & \dots \\ 1 & 1 & 0 & \dots \\ \vdots & \vdots & \vdots & \vdots \\ 1 & 1 & 0 & \dots \end{bmatrix}$$

We are **always sure that**, for all individuals $1 = \text{Male} + \text{Female}$, meaning the first column is a combination of the two following.

In practice, R cannot invert computationally expensive inversion : you co-linearity can be approximate, R will return an error.

Interactions and interpretations

The principle of interactions

Purpose

- Sometimes, you want to have access to some precise coefficients
- *Example*: What is the additional earning for men for an additional hour worked.

A look at other specifications

Let's regress earnings over hours, age and Male

$$\text{earnings} = \beta_0 + \beta_1 \text{hours} + \beta_2 \text{age} + \beta_3 \text{Male} + \beta_4 (\text{Male} \times \text{hours}) + \epsilon$$

- To get a feeling about what the coefficients mean, just proceed as for a binary variable interpretation

A look at different specification

- In (1) for $Male = 1$,
 $earnings = \beta_0 + \beta_1 hours + \beta_2 age + \beta_3 + \beta_4 hours + \epsilon$
- For $Male = 0$, $earnings = \beta_0 + \beta_1 hours + \beta_2 age + \mu$
- In expectation we get :

$$\begin{aligned}\mathbb{E}(earnings|Male = 1) - \mathbb{E}(earnings|Male = 0) \\ = \beta_3 + \beta_4 hours\end{aligned}$$

- The effect on earnings of **an additional hour for a man** ($Male = 1$) is β_4 , all other things equal. In our case, β_3 would be the baseline difference in means.
- In this type of model, the impact of time is different whether you are a man or a woman.

Coefficient interpretation

Coefficient interpretation

In the following, I will consider a multivariate regression:

$$Price_i = \beta_0 + \beta_1 Surface + \beta_2 Garden + \epsilon_i$$

Where :

- $Price_i$ is the price of house i
- $Surface_i$ is the Surface of house i
- $Garden_i$ equals 1 if house i has a garden, 0 otherwise

► [Back to course](#)

Traditional framework

$$Price_i = \beta_0 + \beta_1 Surface_i + \beta_2 Garden_i + \epsilon_i$$

In this classic framework:

- $\frac{\partial Price_i}{\partial Surface_i} = \beta_1$, so β_1 is the marginal effect of Surface on the price.
- We can say that β_1 is the variation in **units** induced by an increase of one additional **unit** of Surface, *all other things equal*

Log - log framework

$$\log Price_i = \beta_0 + \beta_1 \log Surface_i + \beta_2 Garden_i + \epsilon_i$$

In this framework:

$$\begin{aligned} Price_i &= e^{\beta_0 + \beta_1 \log(Surface_i) + \beta_2 Garden_i + \epsilon_i} \\ \Rightarrow \frac{\partial Price_i}{\partial Surface_i} &= \frac{\beta_1}{Surface_i} e^{\beta_0 + \beta_1 \log(Surface_i) + \beta_2 Garden_i + \epsilon_i} \\ &= \beta_1 \frac{Price_i}{Surface_i} \end{aligned}$$

So:

$$\beta_1 = \frac{Surface_i}{\partial Surface_i} \frac{\partial Price_i}{Price_i}$$

- We can say that that is an **elasticity**. An change of 1% of $Surface_i$ implies a change of $\beta_1\%$ of the price

Log - level framework

$$\log Price_i = \beta_0 + \beta_1 Surface_i + \beta_2 Garden_i + \epsilon_i$$

In this framework: (I don't show it, same method)

$$100 \times \beta_1 = \frac{100 \frac{\partial Price_i}{Price_i}}{Surface_i}$$

We can say that one additional **unit** of *Surface* implies a change of salary of $100 \times \beta_1 \%$

Level - log framework

$$Price_i = \beta_0 + \beta_1 \log Surface_i + \beta_2 Garden_i + \epsilon_i$$

In this framework: (I don't show it, same method)

$$\frac{\beta_1}{100} = \frac{\partial Price_i}{100 \frac{Surface_i}{Surface_i}}$$

We can say that one additional **percent** of *Surface* implies a change of salary of $\frac{\beta_1}{100}$ **unit**