ECONOMETRICS I

Lecture 2

Basic R: data visualization, transformation, & estimation

Matías Cabello
matias.cabello@wiwi.uni-halle.de

October 16, 2025

Introduction to R

Installation

1. Get Rstudio from `https://rstudio.com/products/rstudio/download/`
2. If you have it already, make sure to upgrade to the latest version (4.5.0).

   It may take a while, **better do it at home**!

Typing `version` in the console you should get `minor = 5.0` (see next page):

Installation

Typing `version` in the console you should get `minor = 5.0` :

```
> version

platform          x86_64-w64-mingw32
arch              x86_64
os                mingw32
...
major             4
minor             5.0
...
```

If this is not the case, try this:

```
install.packages("installr")

library(installr)

updateR()
```

Does it work now?

Some basics

- Basic definitions
  - Many windows:
    - The **console** (and output)
    - The **script**: a set of instructions
    - The **data** in memory
    - Others: viewer, help, etc
  - Working with R is basically about writing scripts.
  - Scripts are typically structured as follows:
    - Preamble: clean memory, call packages, define working directory
    - Call external datasource
    - Statistical analysis (graphs, regressions, etc.)

Some basics

- Commands vs **comments** in the script.
  You can write comments after the **#-symbol**

```
install.packages("installr") # here we are installing
     a package
library(installr) # here we are asking R to call that
     package
updateR() # here we are executing a command (it
    wouldn't work without calling the package in the
    first place)
```

- Select and execute line by line (or all in once) with **CTRL+Enter**

Some basics

Before we start:

Download the files "tech_adoption.xlsx" and "yield_curve.xlsx" save them in a specific working directory (you may create a folder for this lecture)

Now open a script in R. (File → New file → R Script)

Save it in your working directory. (e.g. IEEA Code 1.R)

(remember to save your work regularly!)

Some basics

We usually start R-scripts with a preamble that cleans the workspace, determines the working directory, and calls packages.

```
#-----------------------
# Preamble
#-----------------------
rm(list=ls()) # This command deletes everything from your
     workspace.

setwd("G:/My␣Drive/UniHalle/Courses/Empirical␣Econ␣
    Analysis/")  # Working directory (of YOUR computer!)

#install.packages("openxlsx") # installing the package
    for reading excel files (uncomment the first time)
library(openxlsx) # we are telling R to use it

#install.packages("stargazer") # installing the package (
    uncomment the first time)
library(stargazer) # For regression tables
```

Basic R: data visualization, transformation, & estimation

Loading and Viewing Data

Load the required packages and view the tech adoption data:

```r
# Load packages
library(openxlsx)

# Read Excel file
tech <- read.xlsx("tech_adoption.xlsx")

# View first few rows
head(tech)
```

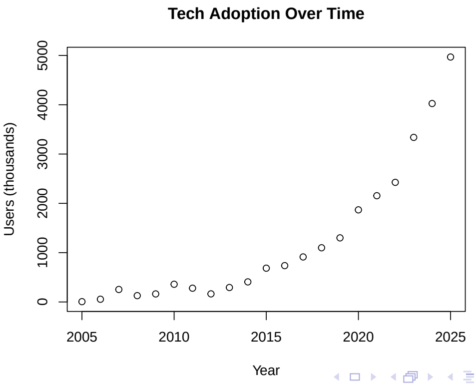
The console will show:

```
> # View first few rows
> head(tech)
  Year Users
1 2005     6
2 2006    55
3 2007   253
4 2008   128
5 2009   163
6 2010   359
```

Basic Plotting

Create a simple scatter plot of tech users over time:

```r
# Basic scatter plot
plot(tech$Year, tech$Users,
     main = "Tech␣Adoption␣Over␣Time",
     xlab = "Year", ylab = "Users")
```

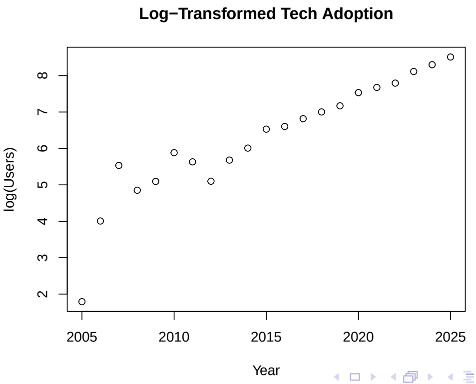


Log Transformation

Slides1

Plotting log-transformed data:

```
# Plot log-transformed users
plot(tech$Year, log(tech$Users),
     main = "Log-Transformed␣Tech␣Adoption",
     xlab = "Year", ylab = "log(Users)")
```



Linear Regression

Run simple linear and log-linear regressions:

```
# Run linear regression
model_linear <- lm(Users ~ Year, data = tech)

# Show results
summary(model_linear)

# Run log-linear model
model_log <- lm(log(Users) ~ Year, data = tech)

# Show results
summary(model_log)
```

Display and compare results:

```
    stargazer(model_linear, model_log,
          type = "text",
          title = "Tech␣Adoption:␣Linear␣vs␣Log-Linear␣
              Models",
          column.labels = c("Linear", "Log-Linear"),
          dep.var.labels = c("Users", "log(Users)"))
```

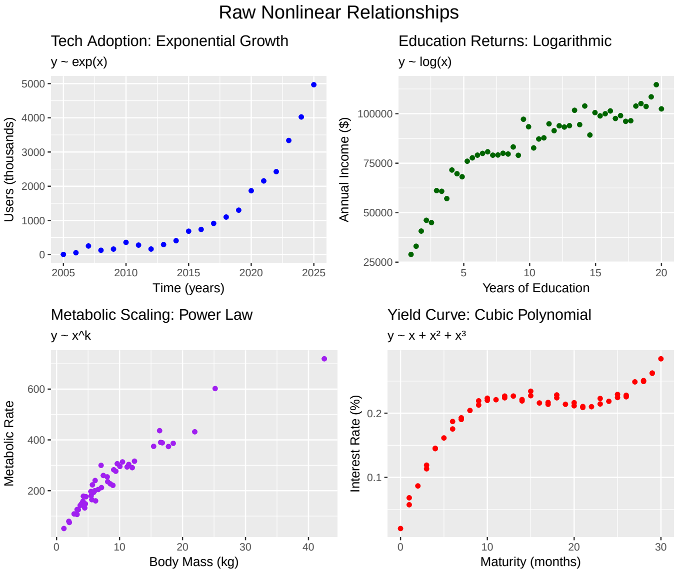
Linear Regression

Console output:

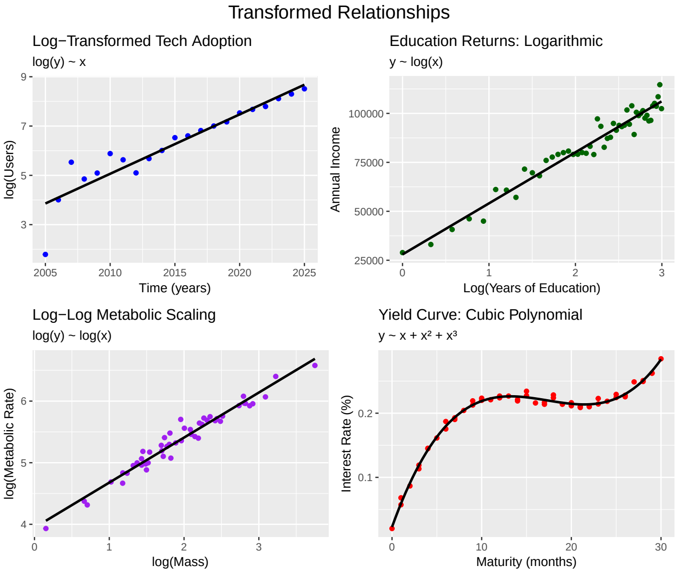
```
    Tech Adoption: Linear vs Log-Linear Models
==========================================================
                               Dependent variable:
                         ----------------------------
                              Users        log(Users)
                              Linear       Log-Linear
                               (1)            (2)
----------------------------------------------------------
Year                       200.094***      0.241***
                           (25.205)        (0.022)

Constant                -401,968.400***  -479.229***
                         (50,787.630)     (44.104)

----------------------------------------------------------
Observations                  21              21
R2                           0.768          0.864
==========================================================
Note:                         *p<0.1; **p<0.05; ***p<0.01
```

Notice difference in R^2.

How would you transform the following data?



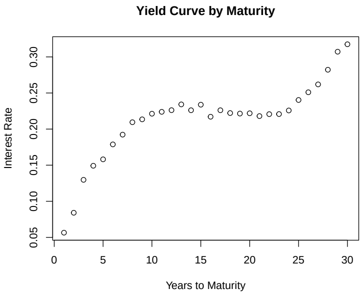
How would you transform the following data?



Polynomial Regression

Working with polynomial relationships:

```
# Read yield curve data
yield <- read.xlsx("yield_curve.xlsx")

# Basic plot
plot(yield$Maturity, yield$Yield,
     main = "Yield Curve by Maturity",
     xlab = "Maturity", ylab = "Yield")
```



Polynomial Regression

Polynomial regression:

```
# Cubic polynomial model
model_poly <- lm(Yield ~ Maturity + I(Maturity^2) + I(
    Maturity^3),
                    data = yield)

# Show results
summary(model_poly)
```

Comparing linear and polynomial models:

```
# Linear model for yield
model_linear_yield <- lm(Yield ~ Maturity, data = yield)

stargazer(yield_lm_linear, yield_lm,
          type = "text",
          title = "Yield␣Curve:␣Linear␣vs␣Polynomial␣
              Models",
          column.labels = c("Linear", "Cubic"),
          dep.var.labels = c("Yield", "Yield"))
```

Polynomial Regression

	Dependent variable: Yield	
	Linear	Cubic
Maturity	0.005***	0.041***
	(0.001)	(0.001)
I(Maturity2)		-0.003***
		(0.0001)
I(Maturity3)		0.0001***
		(0.00000)
Constant	0.131***	0.018***
	(0.011)	(0.004)
Observations	30	30
R2	0.715	0.993
Note:		*p<0.1; **p<0.05; ***p<0.01

Notice difference in R^2: 71.5% vs. 99.3%.