antonio1970 'slides'

c7e411b 24 minutes ago

1 contributor

Raw | Blame | History

338 lines (237 sloc) | 10.3 KB

```
---
title: "Lecture 0: Introduction to the course"
subtitle: "<html><div style='float:left'></div><hr color='#EB811B' size=1px width=796px></h
author: Dr. Antonio Rodriguez Andres| Technical University of Ostrava (VSB-TU)
date: Economic Analysis Using R (EAR) #"`r format(Sys.time(), '%d %B %Y')`"
output:
  xaringan::moon_reader:
    css: [default, metropolis, metropolis-fonts]
    lib_dir: libs
    nature:
      highlightStyle: github
      highlightLines: true
      countIncrementalSlides: false
---

```{r setup, include=FALSE}
options(htmltools.dir.version = FALSE)
library(knitr)
opts_chunk$set(
  fig.align="center",
  fig.height=4, #fig.width=6,
  # out.width="748px", #out.length="520.75px",
  dpi=300, #fig.path='Figs/',
```
```

```
  cache=T#, echo=F, warning=F, message=F
  )
library(tidyverse)
library(hrbrthemes)
```

Table of contents

1. [Prologue](#prologue)

2. [Syllabus description](#syllabus)

3. [Getting started](#started)

4. [R for data science (bookdown)](#r4ds)

5. [Some tips for data R newbies](#https://stackoverflow.com/questions/tagged/r)

---
class: inverse, center, middle
name: syllabus

Syllabus highlights

<html> <hr color='#EB811B' size=1px width=796px></html>

(Read the full document [here](https://github.com/antonio1970/Economic-Analysis-using-R/tre

---

Why this course?

Fill in the gaps left by traditional econometrics and methods classes using less flexible e

- Practical skills that tools that will benefit your dissertation (master or doctoral) and
- Neglected skills like how to actually find datasets in the wild and clean them.

Data science skills are largely distinct from (and complementary to) the core 'metrics oeuv

- Data viz, cleaning and wrangling; programming; cloud computation; relational databases; m

--

> *"In short, we will cover things that I wish someone had taught me when I was starting ou
school."*

```
# You, at the end of this course

<div align="center">
<img src="pics/awesome.jpg">
</div>

---
# Grading

| Component                          | Weight |
|:-|-:|
| 2 × homework assignments (15% each) | 30% |
| 1 × research project                          | 60% |
|       In-class participation                                          | 10% |

- You can swap out one homework assignment for an (approved) final presentation of your own
- We'll get to research project later in the course.

### Research project

- Your final research project will be an analysis in which you apply some techniques covere

- Your research project should look like such as this R markdown `template`: (Read the full

- You can also have a flavour of potential research projects (maybe beyond the course mater
---
# Grading (cont.)
### Research project examples

- Globalization and suicide: (See the full code and data [here](https://www.kaggle.com/tony
- Predicting the Body Mass Index (BMI). (See the full code and data [here](https://www.kagg
- Missing values and Clustering: Knowledge Economy in African countries. This is beyond the

---
# Lecture outline

### Introduction to R basics

- Introduction: Motivation, software installation, and data visualization
- Version control with Git(Hub)
- Learning to love the shell
- R language basics
- Data cleaning, and wrangling with the "Tidyverse"


### Statistical Analysis

- Regression analysis in R
- Binary choice models in R
```

```
- Count data models in R
- Introduction to panel data models in R: the `plm` package
- Generating reports with Rmarkdown

---

# Getting started

<html><div style='float:left'></div><hr color='#EB811B' size=1px width=796px></html>

---

# Software installation and registration

1. Download [R](https://www.r-project.org/).

2. Download [RStudio](https://www.rstudio.com/products/rstudio/download/preview/).

---
class: inverse, center, middle
name: r4ds

# R for Economics analysis and data science
<html><div style='float:left'></div><hr color='#EB811B' size=1px width=796px></html>

---

# Why R and RStudio? (cont.)
```{R, indeeddotcom, echo = F, fig.height = 6, fig.width = 9, dev = "svg"}
# The popularity data
pop_df <-
  data.frame(
  lang = c("SQL", "Python", "R", "SAS", "Matlab", "SPSS", "Stata"),
  n_jobs = c(107130, 66976, 48772, 25644, 11464, 3717, 1624),
  free = c(T, T, T, F, F, F, F)
  )
## Plot it
pop_df %>%
  mutate(lang = lang %>% factor(ordered = T)) %>%
  ggplot(aes(x = lang, y = n_jobs, fill = free)) +
  geom_col() +
  geom_hline(yintercept = 0) +
  aes(x = reorder(lang, -n_jobs), fill = reorder(free, -free)) +
  xlab("Statistical language") +
  scale_y_continuous(label = scales::comma) +
  ylab("Number of jobs") +
  labs(
    title = "Comparing statistical languages",
    subtitle = "Number of job postings on Indeed.com, 2019/01/06"
```
```

```
    ) +
  scale_fill_manual(
    "Free?",
    labels = c("True", "False"),
    values = c("#f92672", "darkslategray")
    ) +
  ggthemes::theme_pander(base_size = 17) +
  # theme_ipsum() +
  theme(legend.position = "bottom")
```

Why R and RStudio? (cont.)

Data science positivism

- Alongside Python, R has become the *de facto* language for data science.
  - See: [*The Impressive Growth of R*](https://stackoverflow.blog/2017/10/10/impressive-gr
- Open-source (free!) with a global user-base spanning academia and industry.
  - "Do you want to be a profit source or a cost center?"

Bridge to applied economics and other tools

- Already has all of the statistics and econometrics support, and is amazingly adaptable as
programming languages and APIs.
- The RStudio IDE and ecosystem allow for further, seemless integration.

Path dependency

- It's also the language that I know best.
--

- (Learning multiple languages is a good idea, though.). The same tools are available in Ma

---

Some R basics

1. Everything is an object.

2. Everything has a name.

3. You do things using functions.

4. Functions come pre-written in packages (i.e. "libraries"), although you can — and should

--

```
</br>

Points 1. and 2. can be summarised as an [object-orientated programming](https://en.wikiped
  - This may sound super abstract now, but we'll see *_lots_* of examples over the coming wee

---

# R vs Stata

If you're coming from Stata, some additional things worth emphasizing:

- Multiple objects (e.g. data frames) can exist happily in the same workspace.
   - No more `keep`, `preserve`, `restore` hackery.
   - This is a direct consequence of the OOP approach.

- You will load packages at the start of every new R session. Make peace with this.
  - "Base" R comes with tons of useful in-built functions. It also provides all the tools n
  - However, many of R's best data science functions and tools come from external packages

- R easily and infinitely parallelizes. For free.
  - Compare the cost of a [Stata/MP](https://www.stata.com/statamp/) license, nevermind the

- You don't need to `tset or xtset` your data. (Although you can too.)

---

# R code example (linear regression)

```{r}
fit = lm(dist ~ 1 + speed, data = cars)
summary(fit)
```

Base R plot

```{r cars_basefig, dev="svg"}
par(mar = c(4, 4, 1, .1)) ## Just for nice plot margins on this slide deck
plot(cars, pch = 19, col = 'darkgray', las = 1)
abline(fit, lwd = 2)
```

Examples using ggplot2

```{r gapm_plot, dev="svg"}
library(ggplot2)
```
```

```
library(gapminder) ## For the gapminder data
ggplot(data = gapminder, mapping = aes(x = gdpPercap, y = lifeExp)) +
  geom_point()
```

Brief aside: The gapminder dataset

Because we're going to be plotting the [gapminder](https://github.com/jennybc/gapminder) da

```{r gapm}
gapminder
```

What else? (cont.)

Elaborate extension: Animation! (See the next slide for the resulting GIF.)
```{r gganim1, eval=FALSE}
# library(gganimate)
ggplot(gapminder, aes(gdpPercap, lifeExp, size = pop, colour = country)) +
  geom_point(alpha = 0.7, show.legend = FALSE) +
  scale_colour_manual(values = country_colors) +
  scale_size(range = c(2, 12)) +
  scale_x_log10() +
  facet_wrap(~continent) +
  # Here comes the gganimate specific bits
  labs(title = 'Year: {frame_time}', x = 'GDP per capita', y = 'life expectancy') +
  transition_time(year) +
  ease_aes('linear')
```

What else? (cont.)

```{r ggamin2, echo=FALSE}
ggplot(gapminder, aes(gdpPercap, lifeExp, size = pop, colour = country)) +
  geom_point(alpha = 0.7, show.legend = FALSE) +
  scale_colour_manual(values = country_colors) +
  scale_size(range = c(2, 12)) +
  scale_x_log10(labels = scales::dollar) +
  facet_wrap(~continent) +
  # Here comes the gganimate specific bits
  labs(title = 'Year: {frame_time}', x = 'Log (GDP per capita)', y = 'Life expectancy') +
  transition_time(year) +
```

```
  ease_aes('linear')
```

What else? (cont.)

There's a lot more to say, but I think we'll stop now for today's lecture.

Rest assured, you will be using ggplot2 throughout the rest of this course and developing y

In the meantime, I want you to do some reading and practice on your own. Pick either of the
- [Chapter 1](https://www.jaredlander.com/r-for-everyone/table-of-contents/) of *R for Ever
- [Chapter 1](https://r4ds.had.co.nz/introduction.html) of *R for Data Science* by G. Grole

---
class: inverse, center, middle

Next lecture: R basics.
<html> <hr color='#EB811B' size=1px width=796px></html>