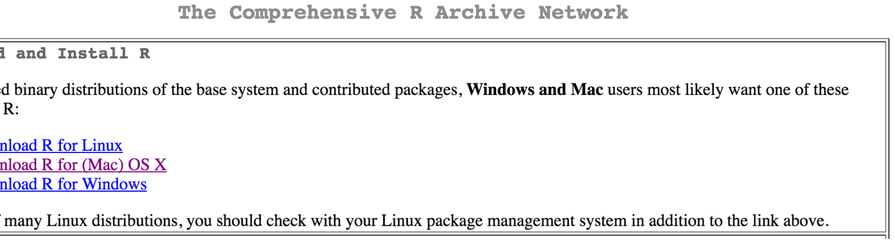
Before we begin…

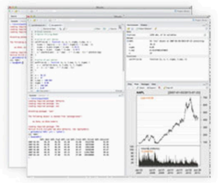
Download R - CRAN R V3.4.1

Download Rstudio - Rstudio.com

Download workshop materials

- goo.gl/J2sgyp





Choose Your Version of RStudio

RStudio is a set of integrated tools designed to help you be more productive with R. It includes a console, syntax-highlighting editor that supports direct code execution, and a variety of robust tools for plotting, viewing history, debugging and managing your workspace. Learn More about RStudio features.

RStudio Desktop Open Source License	RStudio Desktop Commercial License	RStudio Server Open Source License	RStudio Server Pro Commercial License	RStudio Server Pro + RStudio Connect Commercial License
FREE	$995 per year	FREE	$9,995 per year	$29,995 per year
DOWNLOAD	BUY	DOWNLOAD	DOWNLOAD	TALK
Learn More	Learn More	Learn More	Learn More	Learn More

Introduction to R Workshop

Session 3

Sean Nguyen

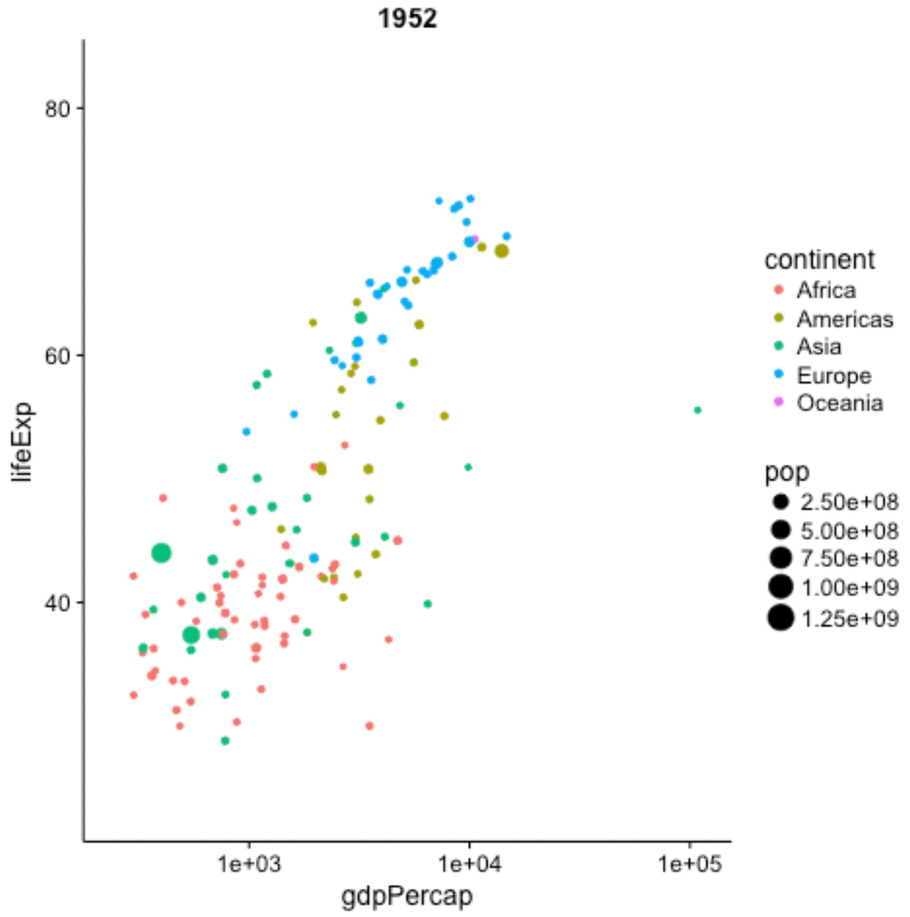
Session 3: Goals

Learn factors and levels

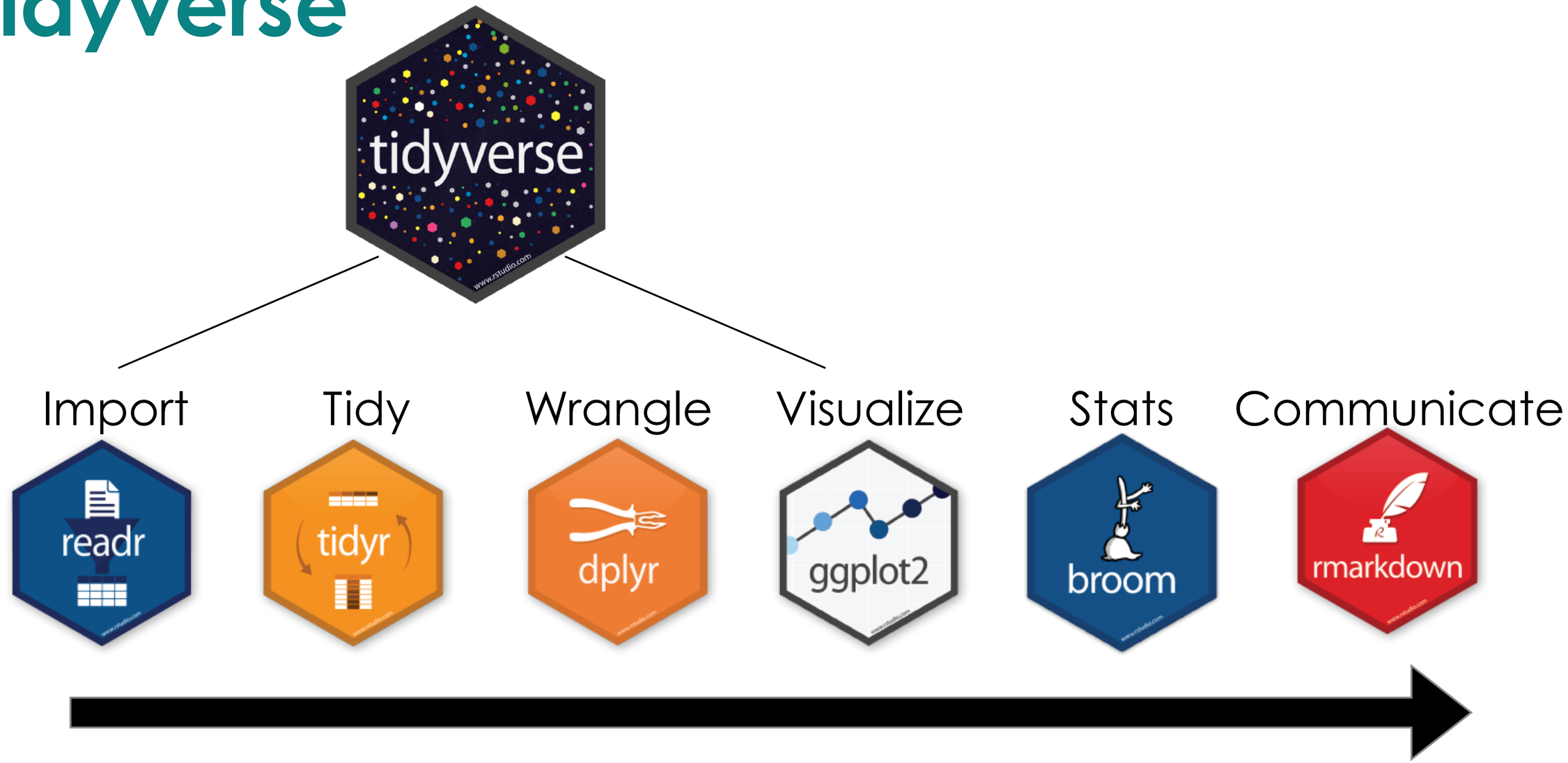
ggplot2

Saving plots

R markdown



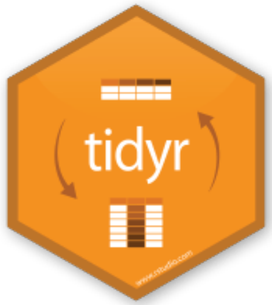
Data Analysis in the **Tidyverse**



Data Analysis in the **Tidyverse**

Import

Tidy



Wrangle

Visualize

ggplot()
geom_line()
geom_bar()
geom_point()
geom_boxplot()

Stats

Communicate

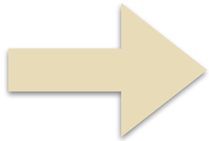
knitr
Rmd
Markdown

factor: categorical variable

Formula: data$column <- **as.factor**(data$column)

data$month <- **as.factor**(data$month)

month (chr)
January
February
March



month (fctr)
January
February
March

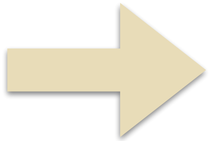
factor: categorical variable

Formula for multiple columns:

mutate_at(**vars**(columnA:columnD), **as.factor**)

mutate_at(**vars**(month:group), **as.factor**)

month (chr)	day (chr)	group (chr)
January	Mon	A
February	Mon	A
March	Tues	B



month (fctr)	day (fctr)	group (fctr)
January	Mon	A
February	Mon	A
March	Tues	B

numeric: numeric variable

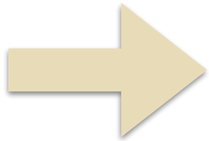
Integers - (~2 billion)

Double - (~1.79e308)

1.15, 4.40, 3.80

Formula: data$column <- **as.numeric**(data$column)

Column (chr)
3.124
5.934
5.600



Column (dbl)
3.124
5.934
5.600

character: strings of text

"MSU R Workshop"

Formula: data$column <- **as.character**(data$column)

Name (fctr)
Debbie
Dylan
Sarah

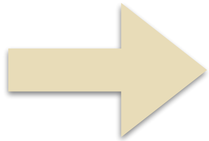
Name (chr)
Debbie
Dylan
Sarah

logical: True/False variable

Formula: data$column <- **as.logical**(data$column)

data$question <- **as.logical**(data$question)

question (chr)
TRUE
FALSE
TRUE



question (lgl)
TRUE
FALSE
TRUE

data types in R:

Factors - categorical variable Monday , Tuesday, Wednesday

Numeric - numbers

Integers (~2 billion)

Double (~1.79e308)

1.15, 4.40, 3.80

Character - strings of characters "Michigan State R Workshop"

Logical - TRUE/FALSE

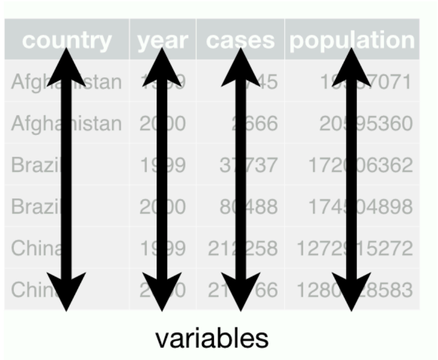
ggplot2 - considerations

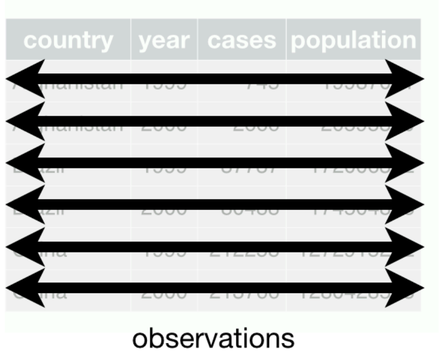
tidy data - variables in columns, observations in rows

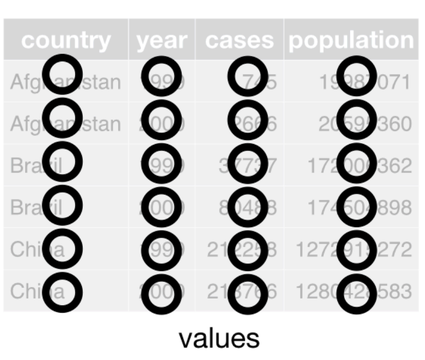
factors - categorical variables

integers/numeric - number variables

levels - order of categorical variables







levels - the **order** of **categorial** variables

Formula: data$column <- **factor**(data$column,
levels = c(“**order**”, “**that**”,”**you**”, “**want**”))

data$month <- **factor**(data$month, **levels** = c(“January”, “February”,
“March”, “April”))

‘April’, ’February’, ’January’, ’March’ → ’January’, ’February’, ’March’, ’April’

ggplot2 - powerful plotting library

General Formula: `ggplot(aes(x = ___, y = ____) ) +`

`geom_point()`

`geom_line()`

`geom_boxplot()`

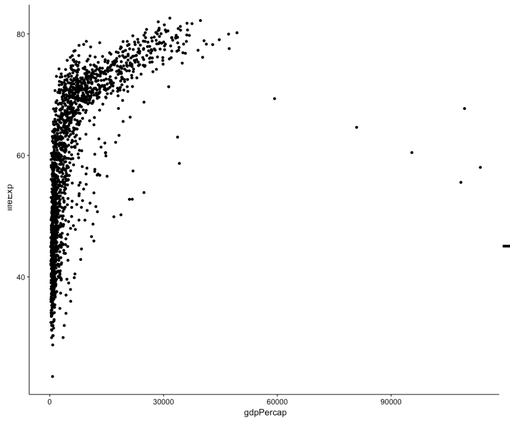
`geom_violin()`

`geom_col()/geom_bar(stat = "identity")`

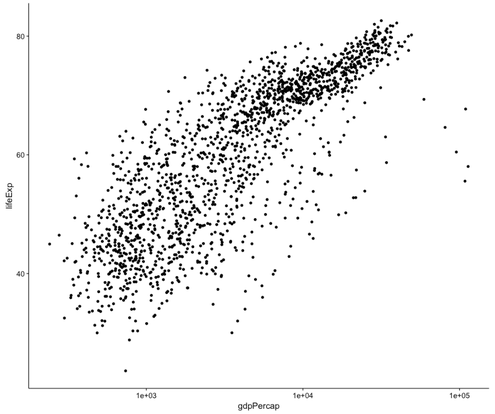
Grammar of Graphics

- Layered
- Iterative
- Customizable

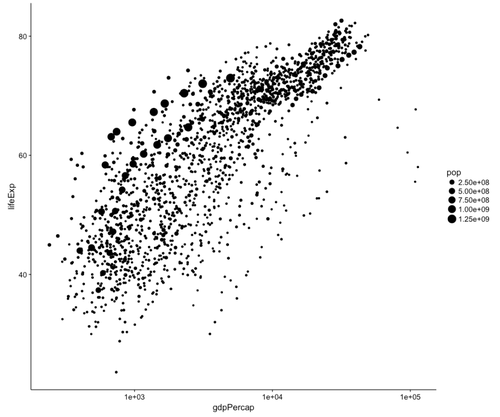
```
# A tibble: 1,704 x 6
       country continent  year lifeExp      pop gdpPercap
        <fctr>    <fctr> <int>   <dbl>    <int>     <dbl>
1  Afghanistan      Asia  1952  28.801  8425333  779.4453
2  Afghanistan      Asia  1957  30.332  9240934  820.8530
3  Afghanistan      Asia  1962  31.997 10267083  853.1007
4  Afghanistan      Asia  1967  34.020 11537966  836.1971
5  Afghanistan      Asia  1972  36.088 13079460  739.9811
```



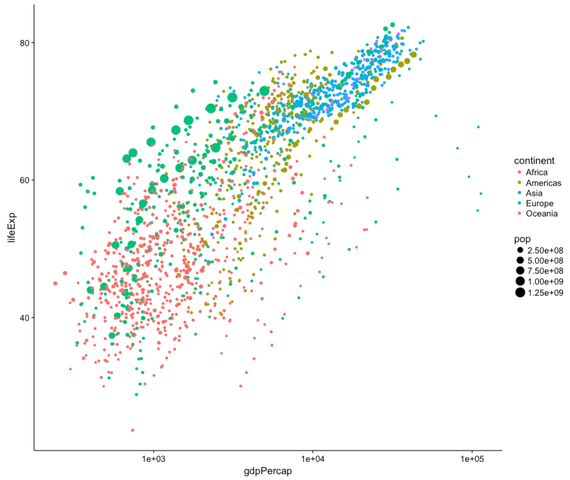

```
data %>%
  ggplot(aes(x = gdpPercap, y = lifeExp))+
  geom_point()
```




```
data %>%
  ggplot(aes(x = gdpPercap, y = lifeExp))+
  geom_point()+
  scale_x_log10()
```




```
data %>%
  ggplot(aes(x = gdpPercap, y = lifeExp, size = pop))+
  geom_point()+
  scale_x_log10()
```




```
data %>%
  ggplot(aes(x = gdpPercap, y = lifeExp,
             size = pop, color = continent))+
  geom_point()+
  scale_x_log10()
```

Demo!

ggplot2 - tweaking your plot

reorder - ggplot(aes (x = **reorder(_____ , ordered_variable)**, y = ____)

color - **+ scale_fill_brewer(palette= “YlOrRd”)**

legends - remove legend **+ guides(color = FALSE)/guides(fill = FALSE)**

annotations - **+ annotate(“text”,x= 2, y=2, label = “your text”)**

Try to plot:

Life expectancy of Asian countries in 1992

Life expectancy of of Africa and Europe in 2007

GDP of Americas and Europe in 2002

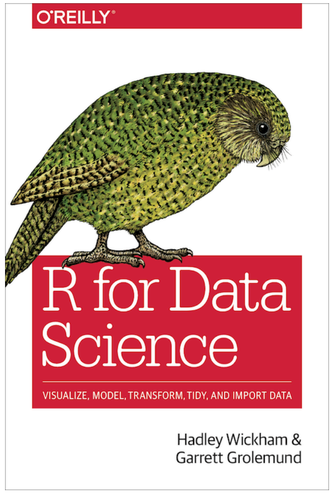
BONUS:

Determine the GDP if each continent in 2007

Additional Resources

R Graphics Cookbook

R for Data Science



Stack Overflow

stackoverflow

R markdown: reproducible documents

Analyze.
Share.
Reproduce.