Advanced R

Logic

Time: 14 minutes

- `isTRUE`: returns TRUE if function == TRUE, FALSE otherwise
- `xor()`: exclusive or; returns TRUE if one argument == TRUE and the other == FALSE
- `which()`: returns indeces of a vector where results == TRUE
- `any()`: returns TRUE if one or more elements == TRUE
- `all()`: returns TRUE if all elements == TRUE

Functions

Time: 19 minutes

- `Sys.Date()`: returns a string of today's date
- `mean()`: returns the average of all numbers in the input vector

lapply and sapply

Time: 14 minutes

- `lapply()`: applies function to each element in a list; returns a list
- `as.list()`: coerces input argument into a list
- `as.character()`: coerces input argument into a character type; strips other attributes (i.e. names)
- `sapply()`: calls `lapply()` method but attempts to simplify output
- `unique()`: returns a vector with all duplicate elements removed (only unique)

vapply and tapply

Time: 10 minutes

- `vapply()`: similar to `sapply()` but allows explicit specification about output format; returns an error if output mismatch
- `tapply()`: applies a function to split-up data (or data entries withina group)

Simulation

Time: 13 minutes

- `rbinom()`: simulates numbers from a binomial distribution
- `rnorm()`: simulates numbers from a random normal distribution
- `rpois()`: simulates numbers from a Poisson distribution
- `replicate()`: repeats trials of an expression; used for simulating groups of random numbers

Dates and Times

Time: 10 minutes

- `Sys.Date()`: returns current system date
- `unclass()`: decodes object; views object internally
- `Sys.Time()`: returns current system date and time
- `as.POSIXlt()`: coerces input to type POSIXlt
- `weekdays()`: returns day of the week
- `months()`: returns month
- `quarters()`: returns quarter (Q1 - Q4)
- `strptime()`: convers character vectors to POSIXlt
- `difftime()`: returns the amount of time passed between two inputted dates

Getting and Cleaning Data

Dates and Times with lubridate

Time: 15 minutes

- `Sys.getlocale()`: returns system local time; needs to be UTF-8 for lubridate to be guarenteed to work properly
- `lubridate::today()`: returns today's date
- `lubridate::year()`: returns year
- `lubridate::month()`: returns month
- `lubridate::day()`: returns day
- `lubridate::wday()`: returns day of week (Sunday = 1)
- `lubridate::now()`: returns current date-time
- `lubridate::hour()`: returns hour
- `lubridate::minute()`: returns minute
- `lubridate::second()`: returns second
- `lubridate::ymd() and other combinations`: parses date-times
- `lubridate::update()`: updates date-time object

- `lubridate::with_tz()`: returns date-time from a different inputted time zone
- `lubridate::interval()`: creates an interval object with start and end dates
- `lubridate::as.period()`: changes an object to a period; if passing an interval, returns time difference

Exploratory Data Analysis

Principles of Analytic Graphs

Time: 6 minutes

No new commands in this lesson

Exploratory Graphs

Time: 23 minutes

- `abline()`: adds straight lines through a plot
- `rug()`: adds 1-D representation of the data to a plot
- `barplot()`: generates a barplot
- `par()`: sets graphical parameters
- `subset()`: returns a subset of vectors, matrices, or data frames if conditions are met
- `with()`: evaluates an R expression in an environment constructed from data

Graphics Devices in R

Time: 10 minutes

- `title()`: adds labels to a plot
- `dev.cur()`: returns the name of the active device
- `pdf()`: starts the graphic device driver for PDF graphics
- `dev.off()`: closes current open device

Plotting Systems

Time: 10 minutes

- `text()`: adds text to a plots

Base Plotting System

Time: 21 minutes

- `range()`: returns min and max values
- `points()`: add points to an already-defined plot
- `legend()`: generates a legend for a plot
- `mtext()`: specifies a main title

Lattice Plotting System

Time: 20 minutes

- `xyplot()`: produces bivariate scatterplot
- `bwplot()`: produces box-and-whisker plots
- `as.factor()`: coerces input into a factor

Working with Colors

Time: 14 minutes

- `colors()`: returns names of 657 predefined colors for use in plotting
- `colorRamp()`: returns a function with argument a vector of values between 0 and 1 mapped to RBG values
- `colorRampPalette()`: returns a function and returns a character vector of RGB colors
- `rgb()`: creates colors corresponding to given intensities
- `RColorBrewer::brewer.pal()`: creates color palettes for thematic maps
- `image()`: displays a color image for 3-D or spatial data use

GGPlot2 (Part 1)

Time: 20 minutes

- `ggplot2::qplot()`: plotting shortcut for ggplot2

GGPlot2 (Part 2)

Time: 22 minutes

- `ggplot2::geom_point()`: adds a scatterplot layer to ggplot2 object
- `ggplot2::geom_smooth()`adds smooth conditional means layer to ggplot2 object
- `ggplot2::facet_grid()`: forms a matrix of panes by row and column by facetting variables

- `ggplot2::ggtitle()`: adds main titles or subtitles
- `ggplot2::labs()`: adds and modifies labels
- `ggplot2::theme_bw()`: adds classic dark-on-light ggplot2 theme (no gray background)
- `ggplot2::geom_path()`: connects data points into a line graph
- `ggplot2::y_lim()`: adds a limit in the graph's y-axis
- `ggplot2::coord_cartesian()`: sets limits to a Cartesian coordinate system

GGPPlot2 (Extras)

Time: 15 minutes

- `ggplot2::geom_boxplot()`: adds a box-and-whiskers layer