

A Short Introduction into Data Science with R

Robert Ladwig
University of Wisconsin-Madison



rladwig2@wisc.edu



@hydrobert

Disclaimer

- material is based on the AWESOME workshop by Rachel Pilla (now postdoc at Oak Ridge National Lab, @rmpilla) “Introduction to R” (*Computer Science in Modern Biology* at Miami University, Ohio)
- with help by Andrew Cannizzarro, Alva Strand and Nicole Berry



About myself

$$\frac{\partial DO}{\partial t} = \frac{1}{A} \frac{\partial}{\partial z} (AK_z^{DO} \frac{\partial DO}{\partial z}) + S_{DO}$$

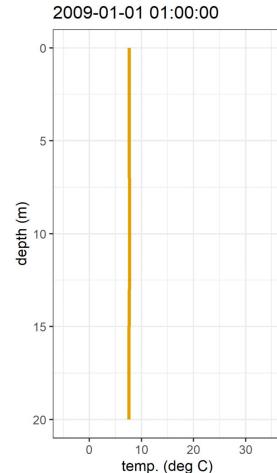


About myself

- Physical limnologist
 - use mathematical models to explore mixing dynamics and water quality in lakes
- developing and applying open-source and open-access software in R, Python and Matlab
- background in civil engineering and geology



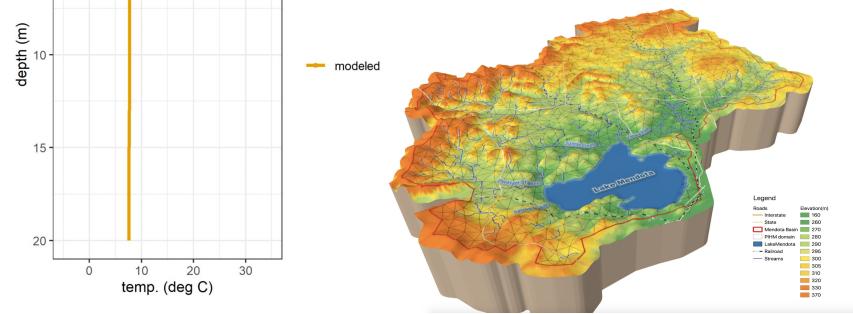
Aquatic ecosystem modeling
mechanistic understanding of cause and effects



$$\frac{\partial T}{\partial t} = \frac{1}{A} \frac{\partial}{\partial z} (A(v_t + v^*) \frac{\partial T}{\partial z}) + \frac{1}{\rho_0 c_p} \frac{\partial H_{sol}}{\partial z} + \frac{dA}{dz} \frac{H_{geo}}{A \rho_0 c_p}$$

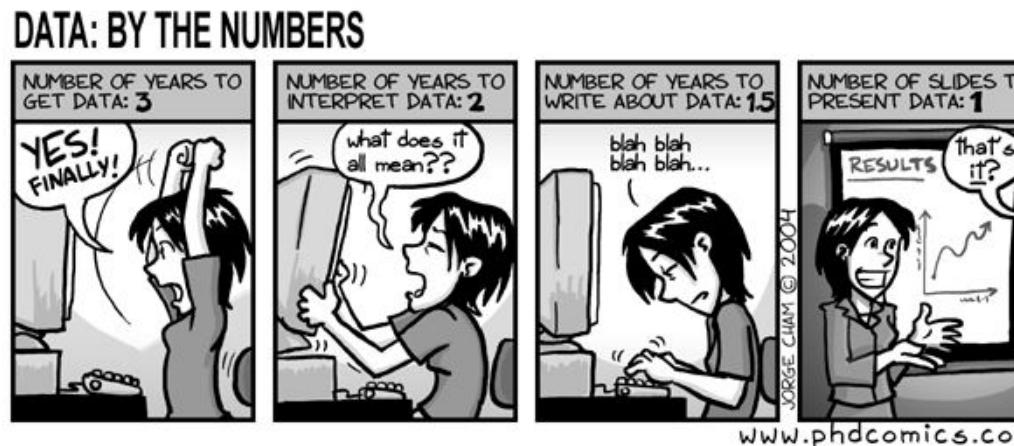
$$\frac{\partial [O_2]}{\partial t} = NEP = GPP - ER \pm D$$

$$\frac{\partial P}{\partial t} = D \frac{\partial^2 P}{\partial z^2} + \lambda P(1 - \frac{P}{K})$$



What's data science?

- combining math & statistics, programming and machine learning to explore any kind of data
- data is often noisy or unstructured → cleaning is big part of the job
- buzzwords like big data



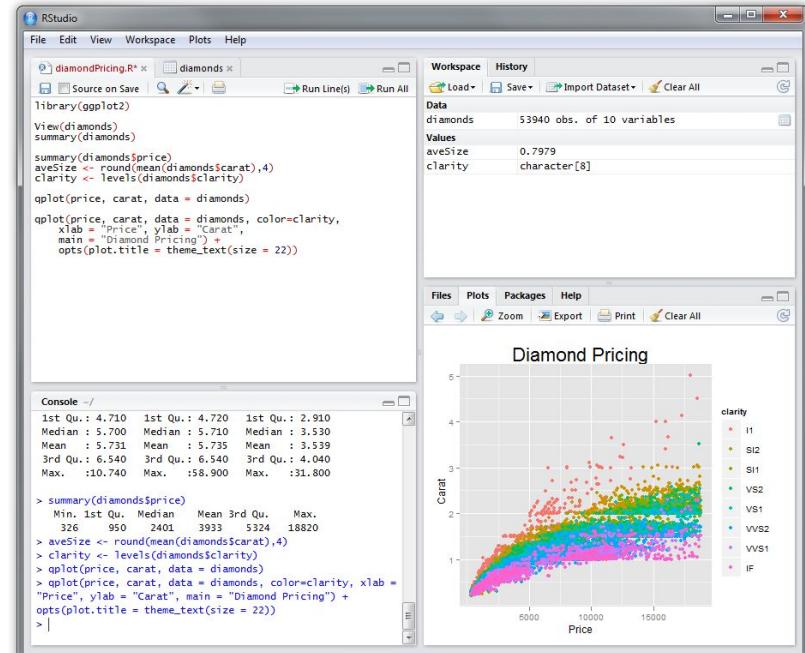
What's the plan for today?

- short introduction into the R scripting language
 - all material on github:
https://github.com/robertladwig/introd_atascience
 - GitHub: uses Git (version control),
fantastic for coding and collaborations
- hands-on coding to introduce you to data analysis
- **at the end:** able to know basic R commands,
load data, analyze data, plot data



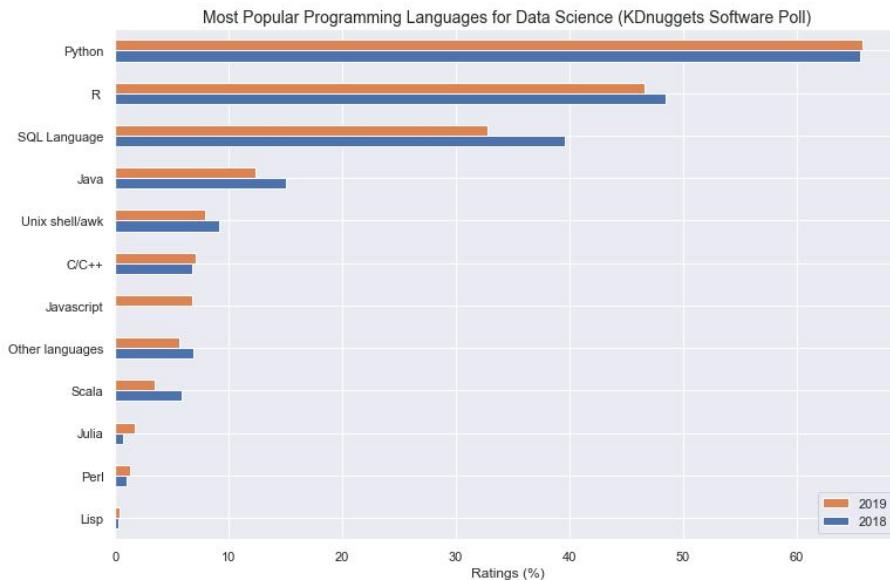
Install R and RStudio

- Install R from here (language):
<https://www.r-project.org/>
- Install RStudio (editor and GUI):
<https://rstudio.com/products/rstudio/download/#download>



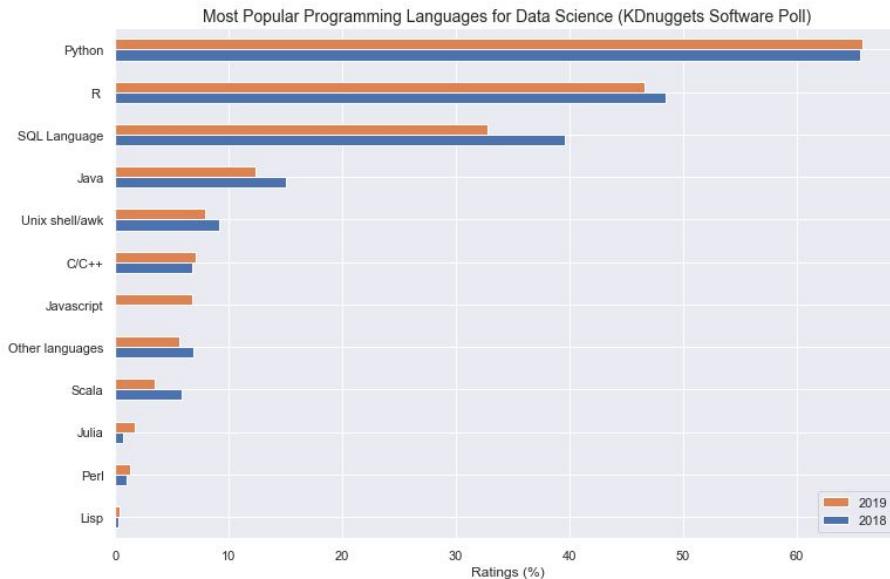
What's R?

- FREE and OPEN SOURCE statistical and computational software



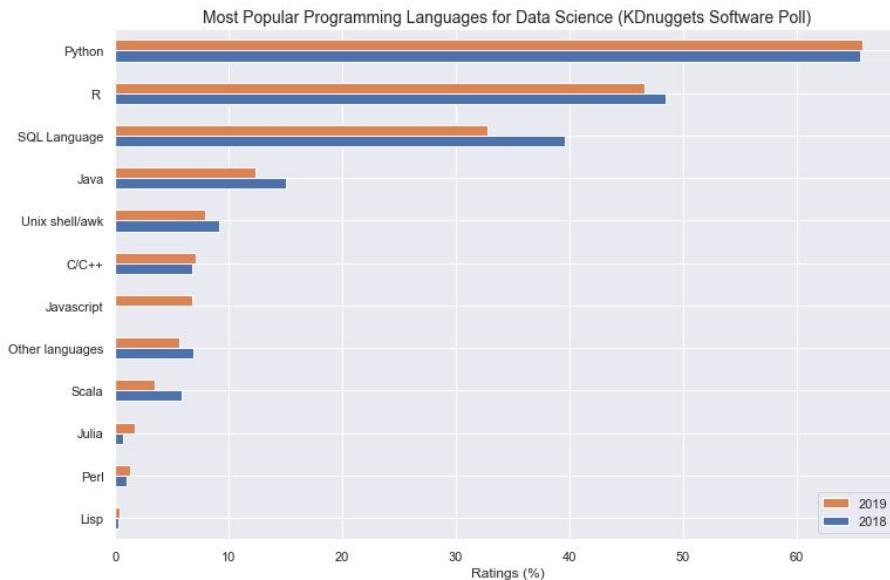
What's R?

- FREE and OPEN SOURCE statistical and computational software
- big community: easy to find solutions and troubleshooting online



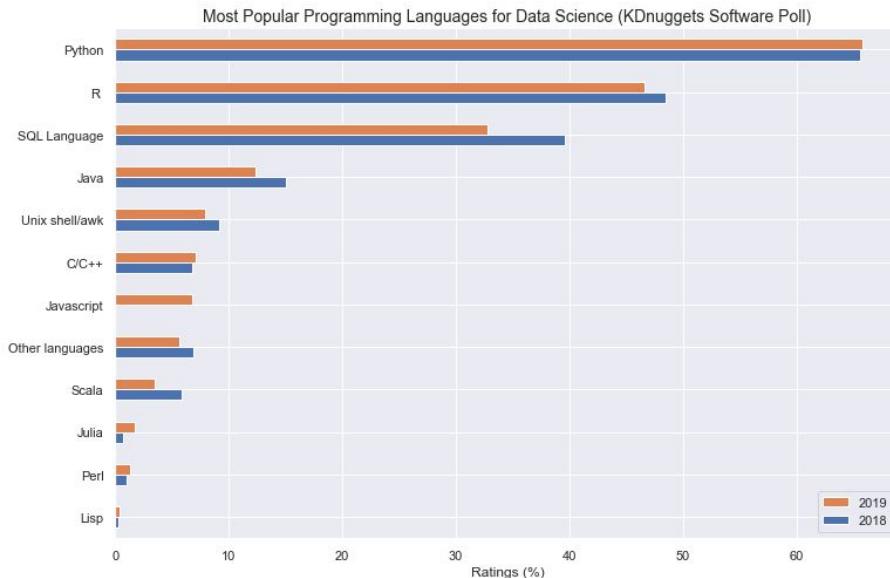
What's R?

- FREE and OPEN SOURCE statistical and computational software
- big community: easy to find solutions and troubleshooting online
- widely used in sciences (esp. statistics and visualization) and rapidly growing in popularity



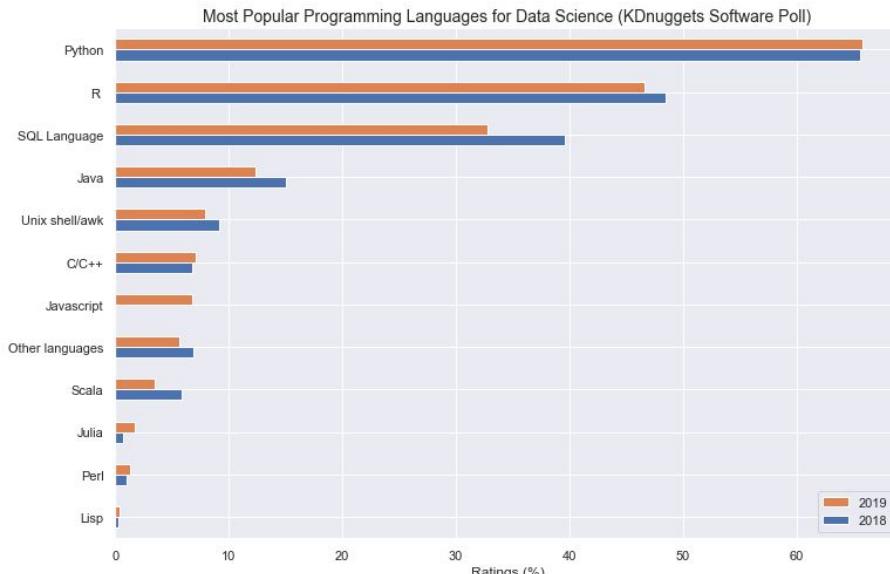
What's R?

- FREE and OPEN SOURCE statistical and computational software
- big community: easy to find solutions and troubleshooting online
- widely used in sciences (esp. statistics and visualization) and rapidly growing in popularity
- can handle more advanced computations, statistical analyses and bigger data files than Excel

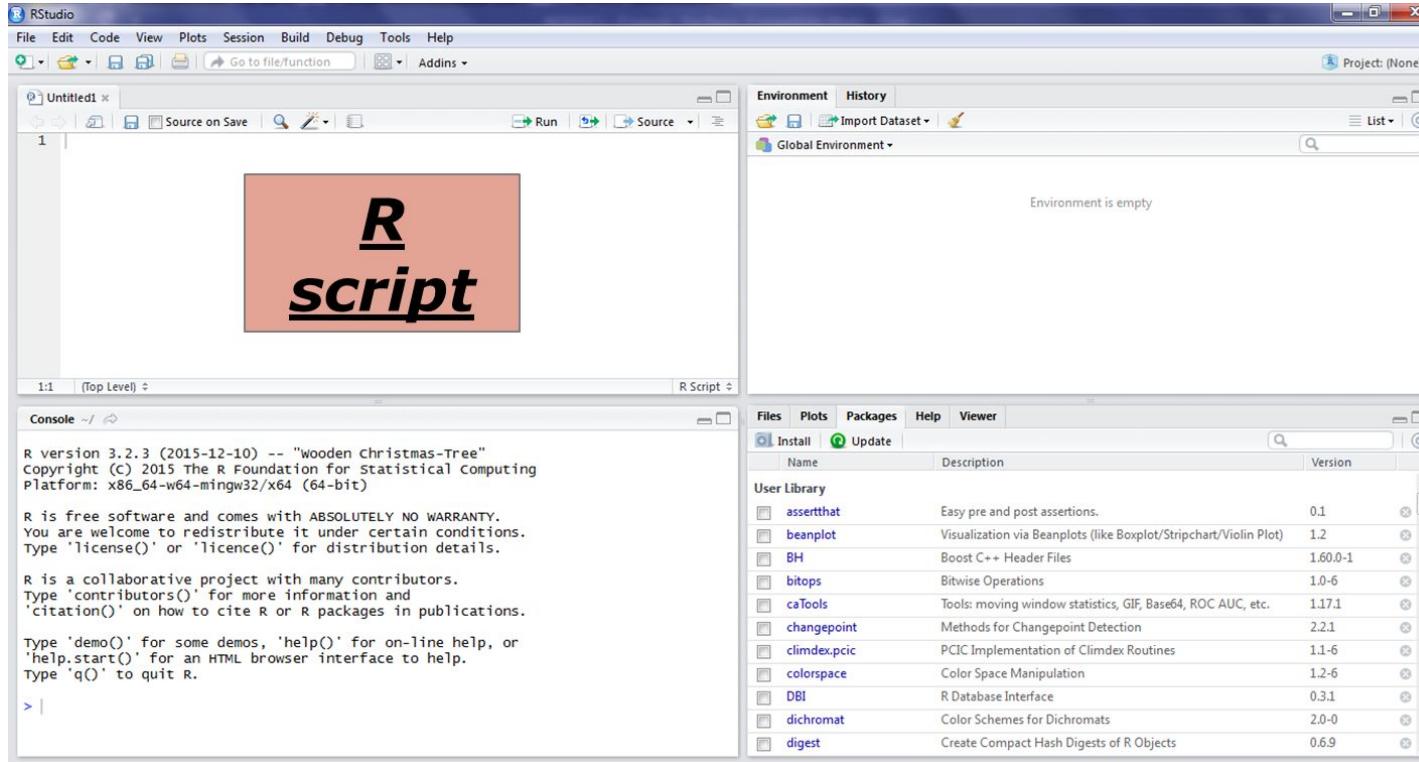


What's R?

- FREE and OPEN SOURCE statistical and computational software
- big community: easy to find solutions and troubleshooting online
- widely used in sciences (esp. statistics and visualization) and rapidly growing in popularity
- can handle more advanced computations, statistical analyses and bigger data files than Excel
- lots of styles for coding



Open RStudio on your laptop now



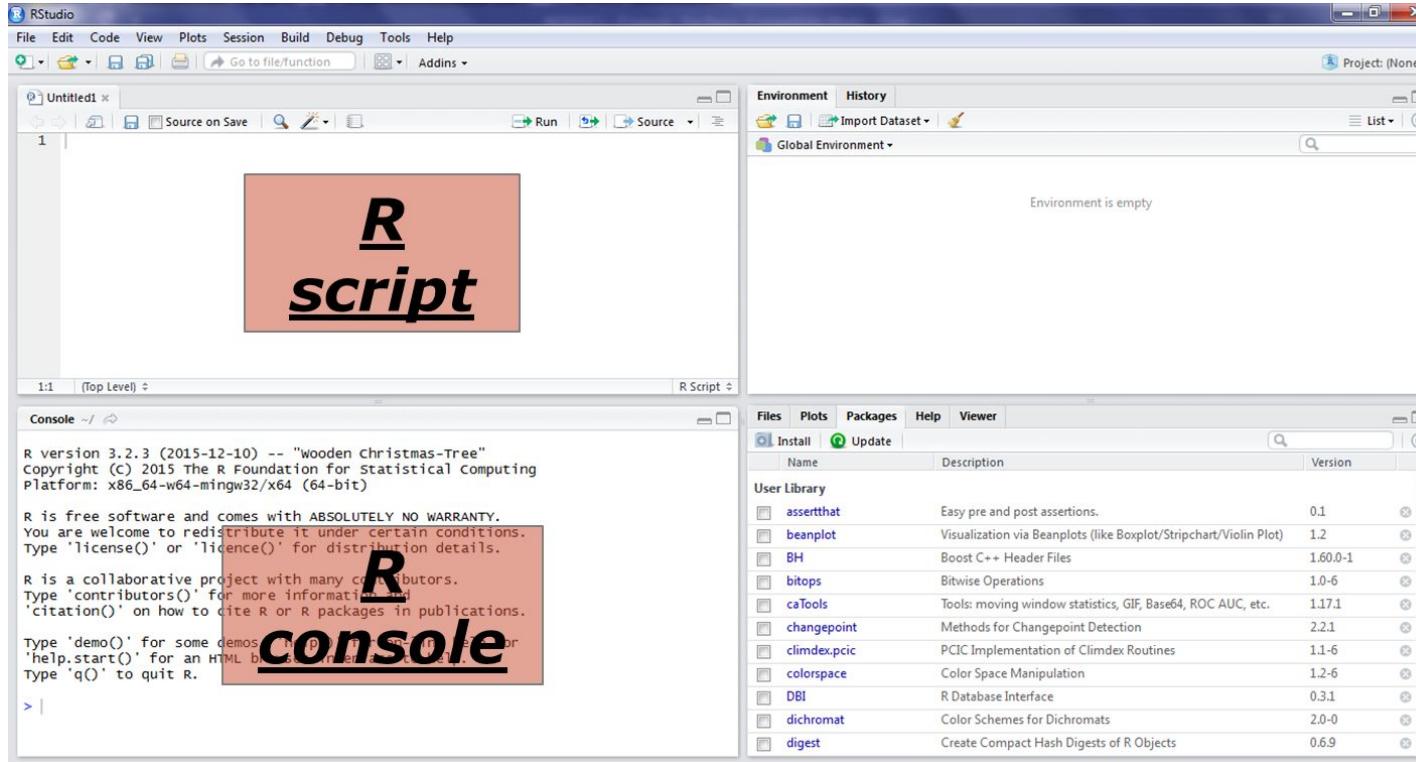
R script:

write and edit code

color-coding for easier identification

automatically fills in parentheses and quotations

Open RStudio on your laptop now

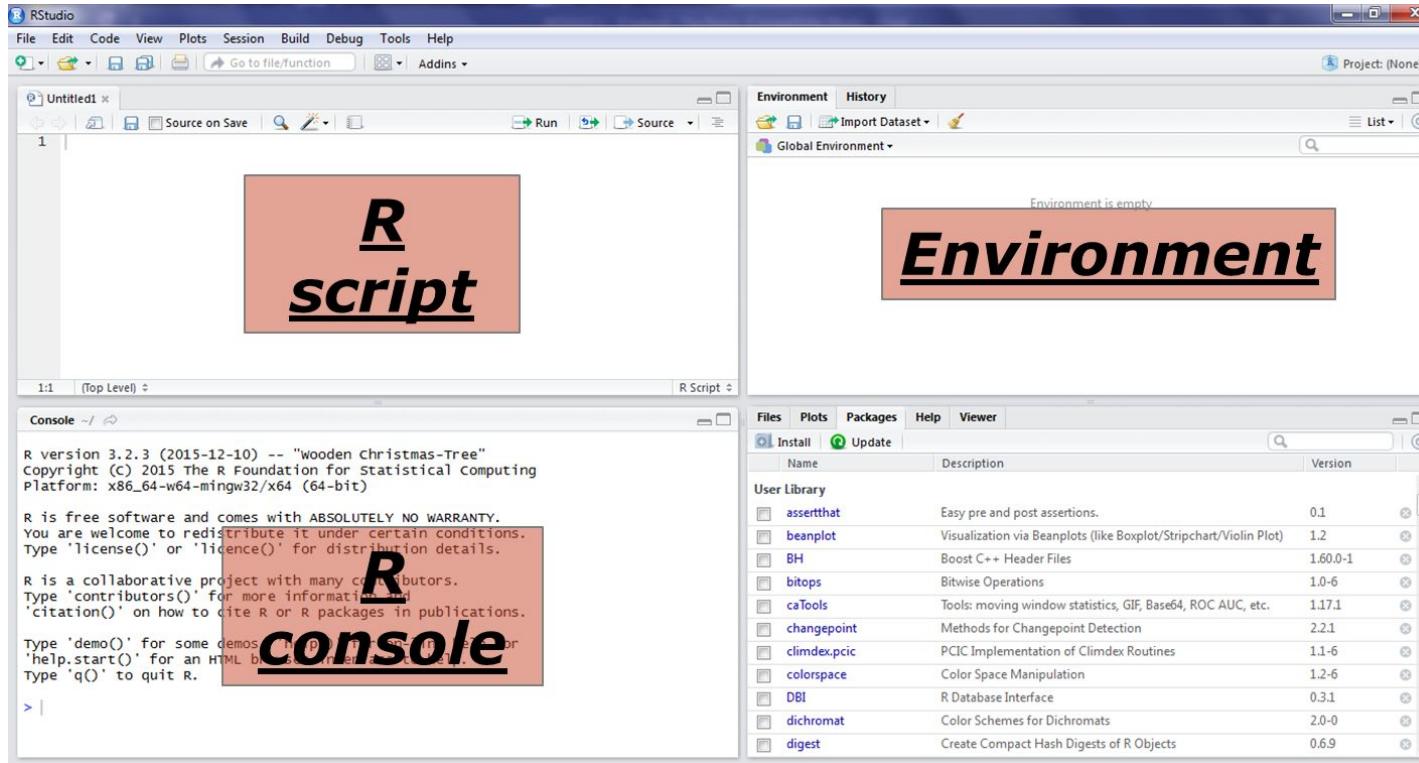


R console:

commands are run and results appear

> means 'ready to work'

Open RStudio on your laptop now

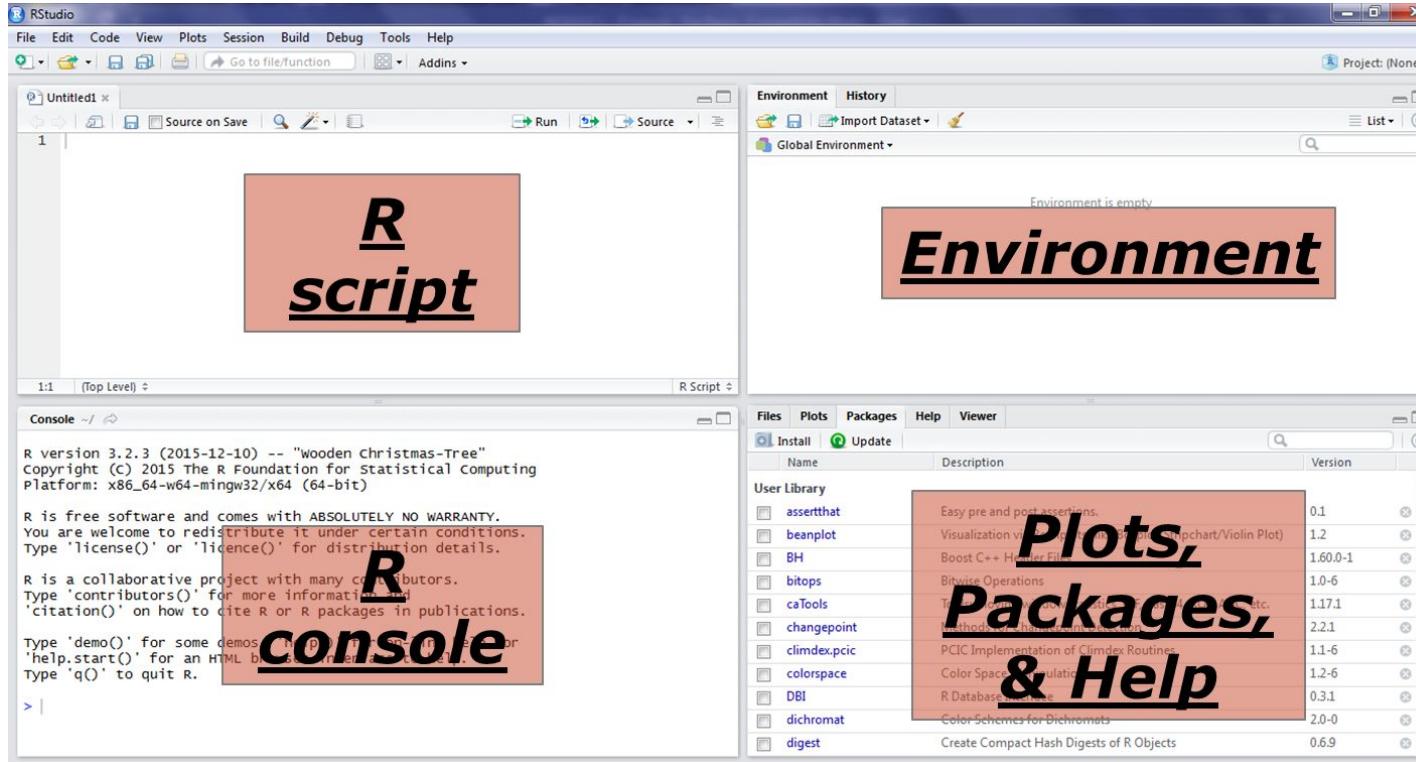


Environment:

see which objects were created

type of objects (double, integers, etc.), size and dimensions

Open RStudio on your laptop now



Plots/Packages/
Help:

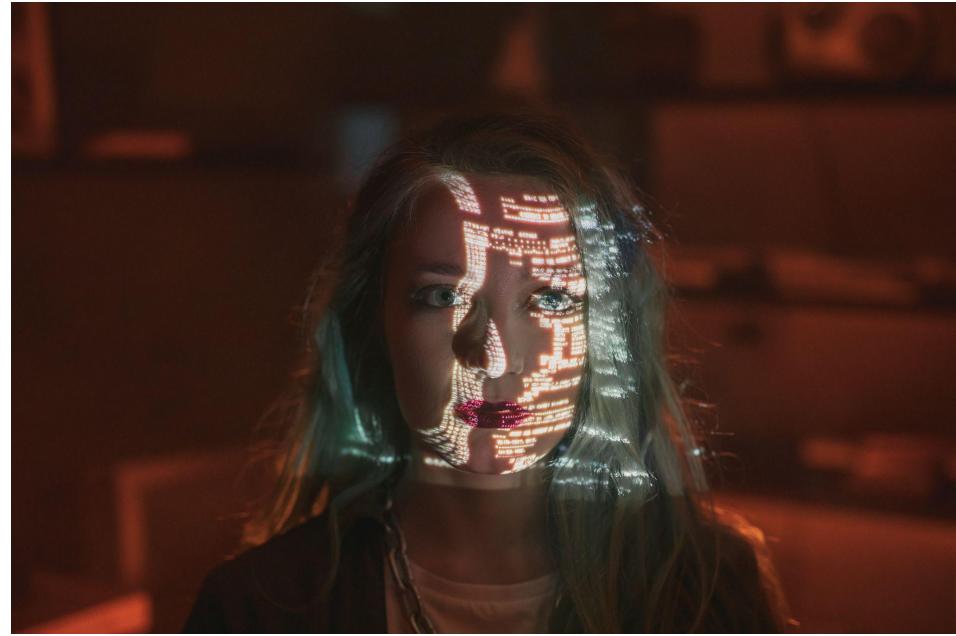
plots will appear

see which
packages are
installed

help functions

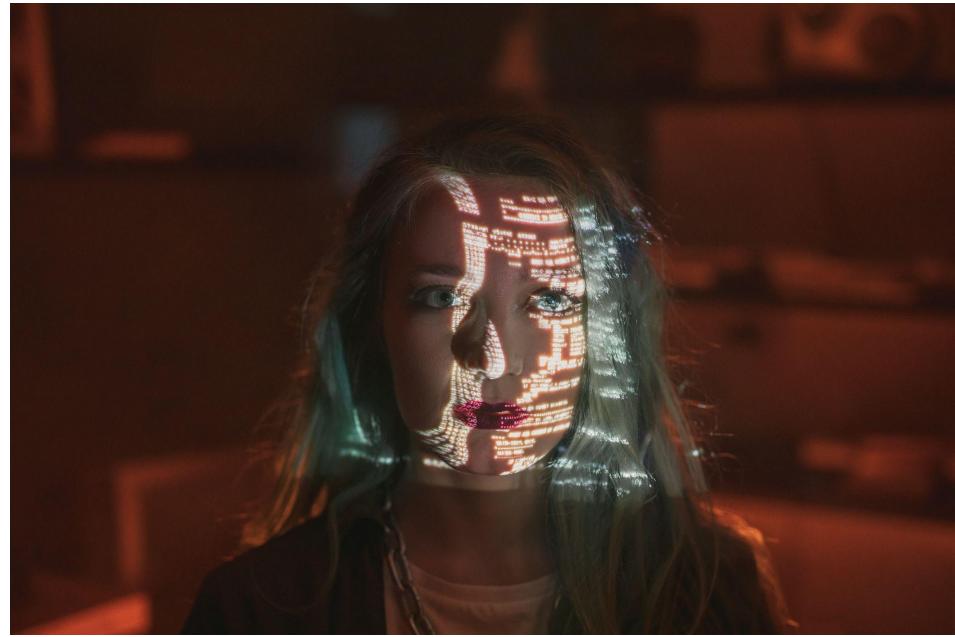
What are packages?

- bundles of tools and functions that others have developed to be used in R



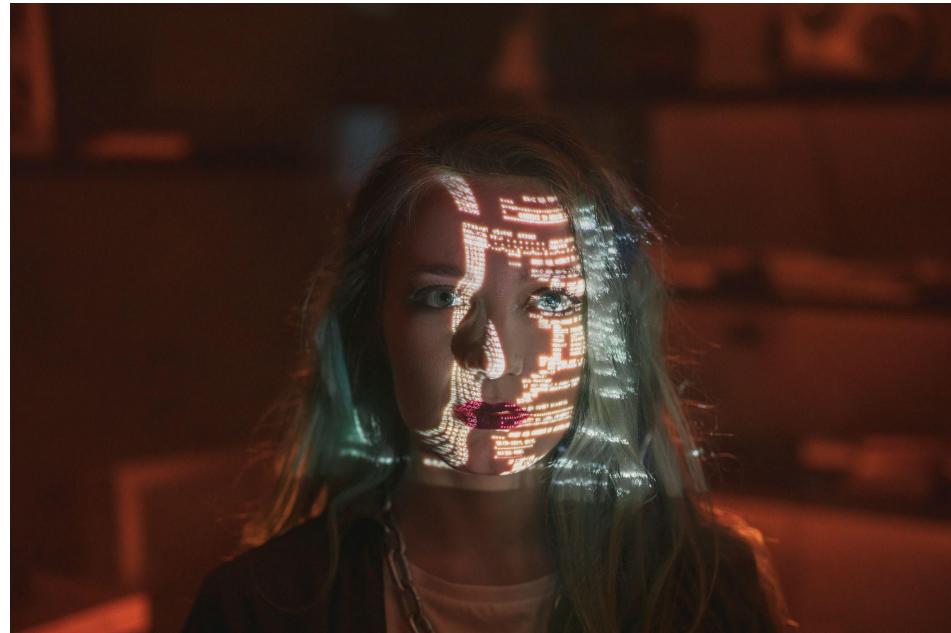
What are packages?

- bundles of tools and functions that others have developed to be used in R
- grouped to specific types of functions, analyses or datasets



What are packages?

- bundles of tools and functions that others have developed to be used in R
- grouped to specific types of functions, analyses or datasets
- you can also create your own packages for specific tasks!



What are packages?

- bundles of tools and functions that others have developed to be used in R
- grouped to specific types of functions, analyses or datasets
- you can also create your own packages for specific tasks!
- currently, above 16,000 packages are available



Install one package

The screenshot shows the RStudio interface. The 'Tools' menu is open, with 'Install Packages...' highlighted. The main workspace shows the 'User Library' with a list of packages like assertthat, beanplot, BH, bitops, caTools, etc. The bottom pane displays the R Console output:

```

~/Miami/R Workshop/ 
> install.packages("rLakeAnalyzer")
Installing package into 'C:/Users/Rachel Pilla/Documents/R/win-library/3.2'
(as 'lib' is unspecified)
trying URL 'https://cran.rstudio.com/bin/windows/contrib/3.2/rLakeAnalyzer_1.7.6.zip'
Content type 'application/zip' length 516163 bytes (504 KB)
downloaded 504 KB

package 'rLakeAnalyzer' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
  C:/Users/Rachel Pilla/AppData/Local/Temp/RtmpmmKDw7/downloaded_packages
>

```

we need:
tidyverse

Install one package

The screenshot shows the RStudio interface with the 'Install' tab selected in the Packages panel. A modal dialog box titled 'Install Packages' is open, prompting the user to enter package names to install. The 'Install from:' dropdown is set to 'Repository (CRAN, CRANExtra)'. The 'Packages' input field contains 'rLake' and 'rLakeAnalyzer'. The 'Install dependencies' checkbox is checked. In the background, the code editor shows R script code for creating bar plots of lake temperature data across different depths.

```
Final-Script_NE-GLEON.R*
```

```
160 barplot(surface$Temp[1:5])
161
162 mean.temp <- aggregate(lake.temp$temp, by=list(lake.temp$Depth), FUN=mean)
163 barplot(mean.temp[,2], names=mean.temp[,1], xlab="Depth (m)", ylab="Mean Temperature (C)", ylim=c(0,25))
164
165 stdev.temp <- aggregate(lake.temp$temp, by=list(lake.temp$Depth), FUN=sd)
166 help(arrows)
167
168 barplot(mean.temp[,2], names=mean.temp[,1], xlab="Depth (m)", ylab="Mean Temperature (C)", ylim=c(0,25))
169 centers<-barplot(mean.temp[,2], names=mean.temp[,1], xlab="Depth (m)", ylab="Mean Temperature (C)", ylim=c(0,25))
170
171 arrows(x0=centers,
172         y0=mean.temp[,2]+stdev.temp[,2],
173         y1=mean.temp[,2]-stdev.temp[,2],
174         code=3, angle=90, length=0.05)
175
176 #####
177 ## PART 3 - Limnology in R ##
178 #####
179
180
181
182
183 < [ ] (Top Level) :
```

```
Console ~/Miami/R Workshop/
> barplot(mean.temp[,2], names=mean.temp[,1], xlab="Depth (m)", ylab="Mean Temperature (C)", ylim=c(0,25))
> centers<-barplot(mean.temp[,2], names=mean.temp[,1], xlab="Depth (m)", ylab="Mean Temperature (C)", ylim=c(0,25))
> arrows(x0=centers,
+         y0=mean.temp[,2]+stdev.temp[,2],
+         y1=mean.temp[,2]-stdev.temp[,2],
+         code=3, angle=90, length=0.05)
>
```

we need:
tidyverse

Install one package

The screenshot shows the RStudio interface. The top menu bar includes File, Edit, Code, View, Plots, Session, Build, Debug, Tools, and Help. A tab labeled "Final_Script_NE-GLEON.R*" is open in the editor. The code in the script is as follows:

```
160 barplot(surface$Temp[1:5])
161
162 mean.temp <- aggregate(lake.temp$temp, by=list(lake.temp$Depth), FUN=mean, na.rm=TRUE)
163 barplot(mean.temp[,2], names=mean.temp[,1], xlab="Depth (m)", ylab="Mean Temp")
164
165 stdev.temp <- aggregate(lake.temp$temp, by=list(lake.temp$Depth), FUN=sd, na.rm=TRUE)
166 help(arrows)
167
168
169 barplot(mean.temp[,2], names=mean.temp[,1], xlab="Depth (m)", ylab="Mean Temp",
170           ylim=c(0,25))
171 centers<-barplot(mean.temp[,2], names=mean.temp[,1], xlab="Depth (m)", ylab="Mean Temp",
172           ylim=c(0,25))
173 arrows(x0=centers,
174         y0=mean.temp[,2]+stdev.temp[,2],
175         y1=mean.temp[,2]-stdev.temp[,2],
176         code=3, angle=90, length=0.05)
177
178 #####
179 ## PART 3 - Limnology in R ##
180
181 <|       !!!
```

The bottom left pane is the Console, showing the command to install the 'rLakeAnalyzer' package:

```
> install.packages("rLakeAnalyzer")
Installing package into 'C:/Users/Rachel Pilla/Documents/R/win-library/3.2'
(as 'lib' is unspecified)
trying URL 'https://cran.rstudio.com/bin/windows/contrib/3.2/rLakeAnalyzer_1.7.6.zip'
content type 'application/zip' length 516163 bytes (504 KB)
downloaded 504 KB

package 'rLakeAnalyzer' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
  C:/Users/Rachel Pilla/Appdata/Local/Temp/RtmpmmKDw*/downloaded_packages
```

The right side of the interface shows the "Packages" tab of the Project Manager, listing various R packages and their versions. The "tidyverse" package is visible in the list.

Name	Description	Version
assertthat	Easy pre and post assertions.	0.1
beanplot	Visualization via Beanplots (like Boxplot/Stripchart/Violin Plot)	1.2
BH	Boost C++ Header Files	1.60.0-1
bitops	Bitwise Operations	1.0-6
caTools	Tools: moving window statistics, GIF, Base64, ROC AUC, etc.	1.17.1
changepoint	Methods for Changepoint Detection	2.2.1
climdex.pcic	PCIC Implementation of Climdex Routines	11.6
colorspace	Color Space Manipulation	1.2-6
DBI	R Database Interface	0.3.1
dichromat	Color Schemes for Dichromats	2.0-0
digest	Create Compact Hash Digests of R Objects	0.6.9
dismo	Species Distribution Modeling	1.0-15
dplyr	A Grammar of Data Manipulation	0.4.3
fields	Tools for Spatial Data	8.3-6
forecast	Forecasting Functions for Time Series and Linear Models	6.2
foreign	Read Data Stored by Minitab, S, SAS, SPSS, Stata, Systat, Weka, dBase, ...	0.8-66
fracdiff	Fractionally differenced ARIMA aka ARFIMA(p,d,q) models	1.4-2
ggplot2	An Implementation of the Grammar of Graphics	2.1.0
gridExtra	Miscellaneous Functions for "Grid" Graphics	2.2.1
gttable	Arrange grobs in tables.	0.1.2
hexbin	Hexagonal Binning Routines	1.27.1
inline	Functions to Inline C, C++, Fortran Function Calls from R	0.3.14
Kendall	Kendall rank correlation and Mann-Kendall trend test	2.2

we need:
tidyverse

Installing a package

- packages only need to be installed ONCE
- but, if you want to use it you'll need to load it:

```
library(package name)
```



Hadley Wickham @hadleywickham

Replying to [@ijlyttle](#)

[@ijlyttle](#) a package is like a book, a library is like a library; you use library() to check a package out of the library [#rsats](#)

8:34 AM · Dec 8, 2014 · Echofon

Package information

The screenshot shows the RStudio interface. On the left, the code editor displays R script code for data manipulation. On the right, the Environment pane is open, showing the 'Packages' tab of the Packages panel. The 'dplyr' package is selected in the list, which is highlighted in the console below.

Code in the Editor:

```
197 # reformatting from wide to long using "gather"
198
199 TempGather <- temp %>%
200   gather(key = "Depth", value = "Temperature", SurfaceWaterTemp, Bot
201
202
203 # grouping data and reformatting data with dplyr
204
205 ?InsectSprays
206 head(InsectSprays)
207 str(InsectSprays)
208
209 InsectSummary <- InsectSprays %>%
210   group_by(spray) %>%
211   summarize(MeanCount = mean(count),
212             StDevCount = sd(count))
213
214
215
216
217
218
219
220 < [REDACTED] >
221 < [REDACTED] >
222 < [REDACTED] >
223 < [REDACTED] >
224 < [REDACTED] >
225 < [REDACTED] >
226 < [REDACTED] >
227 < [REDACTED] >
228 < [REDACTED] >
229 < [REDACTED] >
230 < [REDACTED] >
231 < [REDACTED] >
232 < [REDACTED] >
233 < [REDACTED] >
234 < [REDACTED] >
235 < [REDACTED] >
236 < [REDACTED] >
237 < [REDACTED] >
238 < [REDACTED] >
239 < [REDACTED] >
240 < [REDACTED] >
241 < [REDACTED] >
242 < [REDACTED] >
243 < [REDACTED] >
244 < [REDACTED] >
245 < [REDACTED] >
246 < [REDACTED] >
247 < [REDACTED] >
248 < [REDACTED] >
249 < [REDACTED] >
250 < [REDACTED] >
```

Console Output:

```
6 12 A
> str(InsectSprays)
'data.frame': 72 obs. of 2 variables:
 $ count: num 10 7 20 14 14 12 10 23 17 20 ...
 $ spray: Factor w/ 6 levels "A","B","C","D",...: 1 1 1 1 1 1 1 1 1 1 ...
>
> InsectSummary <- InsectSprays %>%
+   group_by(spray) %>%
+   summarize(MeanCount = mean(count),
+             StDevCount = sd(count))
> |
```

Packages Panel:

Name	Description	Version
backports	Reimplementations of Functions Introduced Since R-3.0.0	1.1.2
base64enc	Tools for base64 encoding	0.1-3
BH	Boost C++ Header Files	1.66.0-1
bindr	Parametrized Active Bindings	0.1.1
bindrcpp	An 'Rcpp' Interface to Active Bindings	0.2.2
broom	Convert Statistical Analysis Objects into Tidy Data Frames	0.4.4
callr	Call R from R	2.0.4
cellranger	Translate Spreadsheet Cell Ranges to Rows and Columns	1.1.0
cli	Helpers for Developing Command Line Interfaces	1.0.0
colorspace	Color Space Manipulation	1.3-2
crayon	Colored Terminal Output	1.3.4
curl	A Modern and Flexible Web Client for R	3.2
DBI	R Database Interface	1.0.0
dplyr	A 'dplyr' Back End for Databases	1.2.1
dichromat	Color Schemes for Dichromats	2.0-0
digest	Create Compact Hash Digests of R Objects	0.6.15
dplyr	A Grammar of Data Manipulation	0.7.5
evaluate	Parsing and Evaluation Tools that Provide More Details than the Default	0.10.1
forcats	Tools for Working with Categorical Variables (Factors)	0.3.0
ggplot2	Create Elegant Data Visualisations Using the Grammar of Graphics	2.2.1
glue	Interpreted String Literals	1.2.0
gridExtra	Miscellaneous Functions for "Grid" Graphics	2.3
gttable	Arrange 'Grob's in Tables	0.2.0
haven	Import and Export 'SPSS', 'Stata' and 'SAS' Files	1.1.1
highr	Syntax Highlighting for R Source Code	0.7
hms	Pretty Time of Day	0.4.2
htmltools	Tools for HTML	0.3.6
lazyeval	Tools for Lazy Evaluation	0.2.1

under packages,
click on **dplyr**

Package information

The screenshot shows the RStudio interface. In the top-left, there's a code editor with a script named "R-Workshop_Lacawac-Summer2018.R". The code demonstrates how to use the `gather` function from the `dplyr` package to reformat data from wide to long format. Below the code editor is a console window showing the results of running the script, including the creation of a data frame and the definition of the `InsectSummary` object.

In the top-right, the "Environment" tab of the workspace is selected. A search bar at the top right contains the text "R: A Grammar of Data Manipulation". A tooltip is displayed over the search bar, showing the title "A Grammar of Data Manipulation" and the R logo.

The main pane displays the "A Grammar of Data Manipulation" documentation for the 'dplyr' package version 0.7.5. The page includes a navigation bar with links to the DESCRIPTION file and user guides. It features a large "Help Pages" section with a table of contents for letters A through U. Under the letter "A", several functions are listed with their descriptions:

Function	Description
add_count	Count/tally observations by group
add_tally	Count/tally observations by group
all_equal.tbl_df	Flexible equality comparison for data frames
all_equal	Flexible equality comparison for data frames
all_vars	Apply predicate to all variables

this lists all available functions

click on any function to see the help file

Coding in R

- write commands in the script (upper left panel)
 - save it, edit it, revisit it later, etc.



Coding in R

- write commands in the script (upper left panel)
 - save it, edit it, revisit it later, etc.
- code NOT automatically run when you hit Enter



Coding in R

- write commands in the script (upper left panel)
 - save it, edit it, revisit it later, etc.
- code NOT automatically run when you hit Enter
- to run it:
 - “Run” button in upper right corner
 - “CTRL + Enter” (Windows)
 - “Command + Enter” (Mac)



Coding in R

```
if(result !== false) {  
    $distArray = array();  
    $row = mysqli_fetch_assoc($result);  
    $correctAnswer = $row['Correct'];  
    $distArray['A'] = $row['Anum'];  
    $distArray['B'] = $row['Bnum'];  
    $distArray['C'] = $row['Cnum'];  
    $distArray['D'] = $row['Dnum'];  
    $distArray['Correct'] = $correctAnswer;  
    $distArray['Answer'] = rtrim($row[$correctAnswer], ".");  
    $distArray['Query'] = "SELECT * FROM TechTerms WHERE Date='".date("Y-m-d")."';  
    return $distArray;  
}  
else {  
    $distArray['Error'] = 'Quiz load query failed';  
    return $distArray;  
}
```



- add comments using #
 - additional information
 - will be ignored

Coding in R



- add comments using #
 - additional information
 - will be ignored
- R is case sensitive
 - “Mean” ≠ “mean”

Coding in R



- add comments using #
 - additional information
 - will be ignored
- R is case sensitive
 - “Mean” ≠ “mean”
- R doesn't care about spaces and tabs

Coding in R



- add comments using #
 - additional information
 - will be ignored
- R is case sensitive
 - “Mean” ≠ “mean”
- R doesn't care about spaces and tabs
 - need to close all parentheses and quotations

Key components

- **functions**: allow you to manipulate data, apply calculations, run statistical analysis, much more!

Key Components of Code

```
x <- seq(1,10)
```

name of the **function**
to create a sequence

Key components

- **functions:** allow you to manipulate data, apply calculations, run statistical analysis, much more!
- **arguments:** defining information for functions, “customize” it

Key Components of Code

```
x <- seq(1,10)
```

the **arguments** to define the function, to create a sequence from 1 through 10

Key components

- **functions**: allow you to manipulate data, apply calculations, run statistical analysis, much more!
- **arguments**: defining information for functions, “customize” it
- **objects**: pieces of data saved in R, can be called up, reused and manipulated

Key Components of Code

```
x <- seq(1,10)
```

name of the **object**
that saves the results
of the function in R

Key components

- **functions**: allow you to manipulate data, apply calculations, run statistical analysis, much more!
- **arguments**: defining information for functions, “customize” it
- **objects**: pieces of data saved in R, can be called up, reused and manipulated

Key Components of Code

x `<-` seq(1,10)

assignment operator
tells R to save the result
of the function as the
named object

Types of data

- **objects (data types):**
 - numeric (2.334, 3.14159)
 - integer (2, 2405, 54)
 - characters ("hello world")
 - logical (TRUE/FALSE)
 - complex (1+4*i*)
- **data structures:**
 - vector (1-D object with same data type)
 - matrix (2-D object with same data type)
 - data frame (2-D object with different data types per column if needed, very useful!)

Live Coding time!

Bird banding data provided by Dr. Dave Russell at Miami University in collaboration with Hueston Woods State Park

