



R-Programming Bootcamp

Vivek Singh

Information Systems Decision Sciences (ISDS)

MUMA College Of Business

vivek4@mail.usf.edu

Web: <http://vivek4.myweb.usf.edu>



About Me

Professional Experience



Technologies



Linux



Teaching @ USF

Sprint 2017

ISM 4930: Applied Data Science (Cloud computing and Real-time Business)

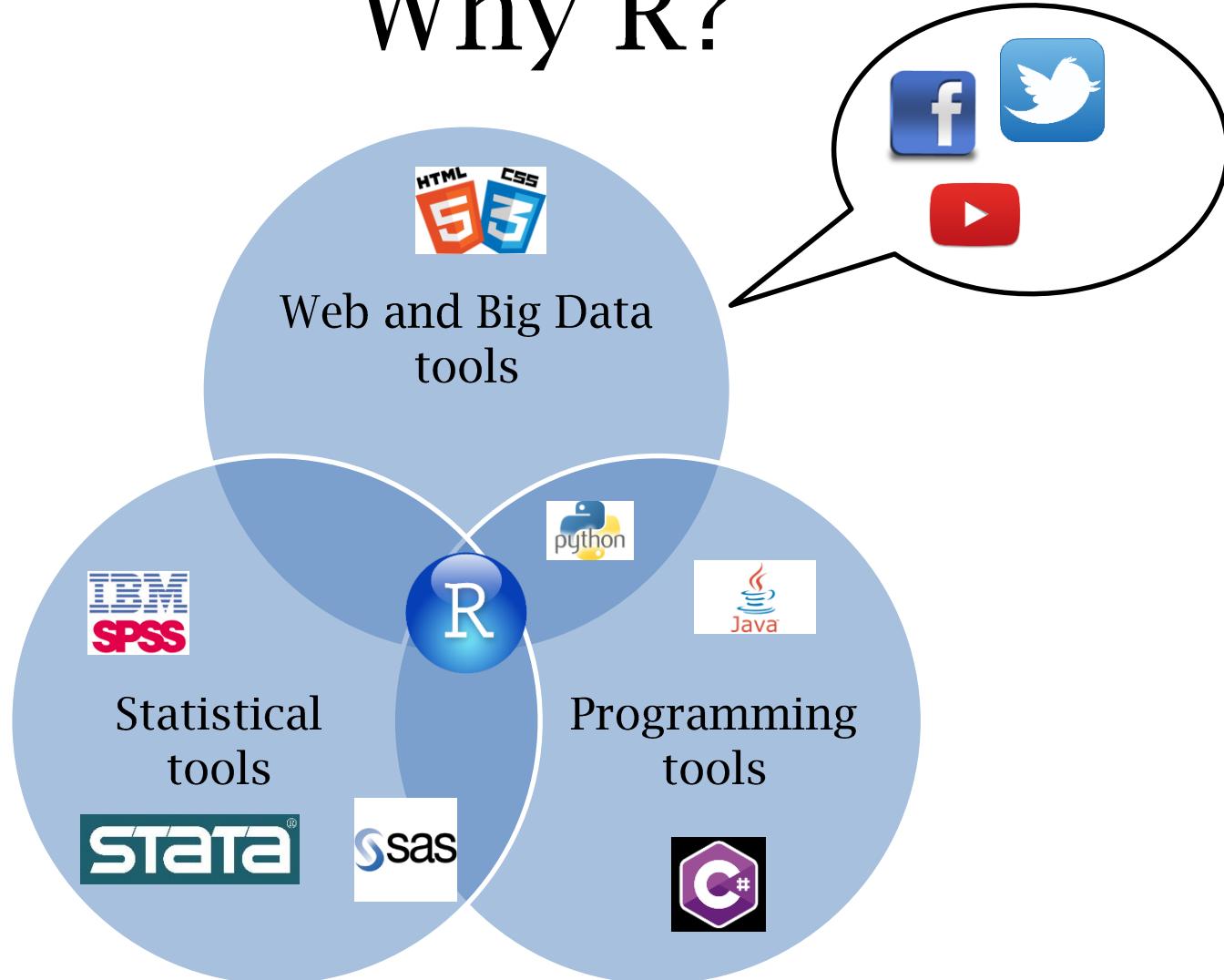


Outline

- Motivation – Why R?
- Programming IDE – R studio, workspace, Console
- Programming – objects, loops, conditionals, function
- File Input/output
- R- Packages
- Data manipulation
- Database connector – MySQL
- Visualization
- Datasets
- Social Media APIs – Twitter, Facebook, Google Trends, Youtube
- CIRCE @USF (Cluster Computing)



Why R?





R Programming

- Programming environment
 - Data manipulation
 - Computation
- Statistical analysis
- Visualization
- Built from S language



Ross Ihaka

Programming Language Designer



George Ross Ihaka is an Associate Professor of Statistics at the University of Auckland who is recognized, along with Robert Gentleman, as one of the originators of the R programming language. [Wikipedia](#)

Born: 1954, Waiuku, New Zealand

Residence: Auckland, New Zealand

Known for: R

Alma mater: University of Auckland, University of California, Berkeley



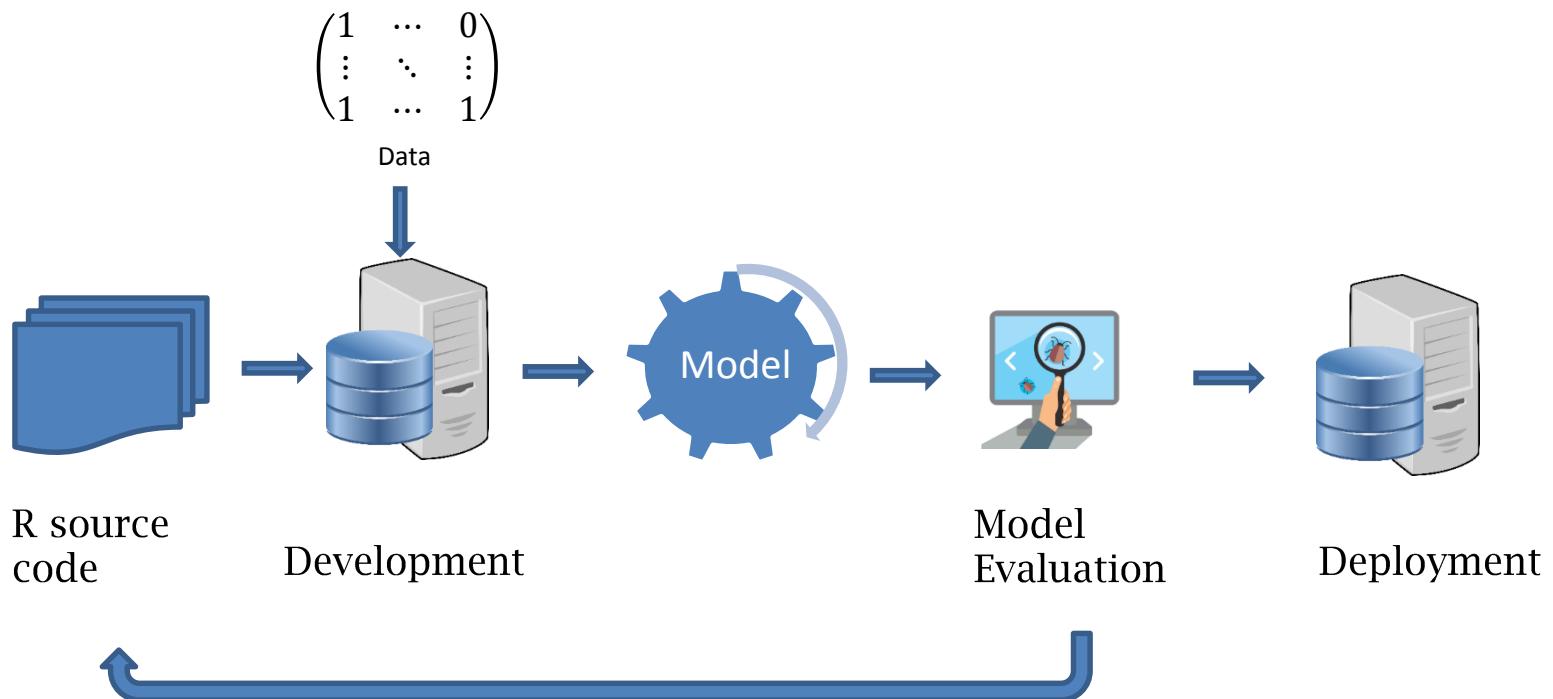
Robert Gentleman

Programming Language Designer





Data Analytics Life Cycle



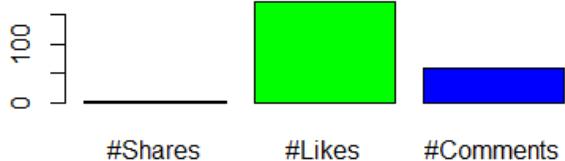
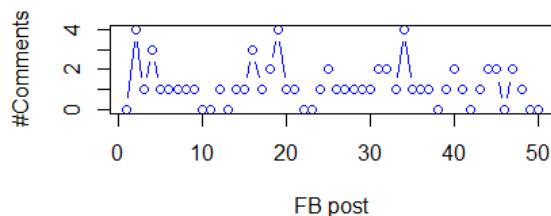
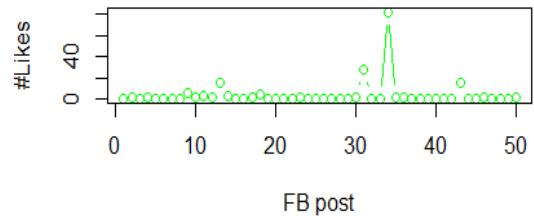
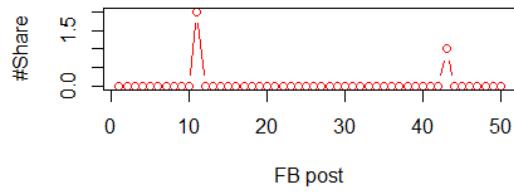


\$ 1,000,000

The screenshot shows the homepage of the Data Science Bowl 2017. At the top, it displays '\$1,000,000 • 668 teams' and the title 'Data Science Bowl 2017'. A timeline bar indicates the competition period from 'Thu 12 Jan 2017' to 'Wed 12 Apr 2017 (2 months to go)', with a 'Merger and Entry Deadline' marked in the middle. Below the title, there's a call-to-action: 'Can you improve lung cancer detection?'. The sidebar on the left includes links for 'Dashboard', 'Home', 'Data', 'Make a submission', 'Information', 'Description', 'Evaluation', 'Rules', 'Prizes', 'About the DSB', and 'Resources'.

<https://www.kaggle.com/c/data-science-bowl-2017>

Social Media Analytics Demo with R



DV: #Share	Std. Estimate	Error	t value	Pr(> t)
(Intercept)	-0.42	0.15	-2.77	0.01 **
#Likes	0.12	0.01	18.88	< 2e-16 ***
#Comments	0.24	0.09	2.59	0.01 *



Run R code on CIRCE @USF

USF Research Computing

[Overview](#) [News](#)

Overview

This is the Research Computing Cluster Web Access System. Below are some of the things you can do here:

- **Documentation.** Research Computing's documentation has been moved to <https://wiki.rc.usf.edu>
- **Read our site news.** It will be updated regularly to provide information on changes to resources, maintenance periods, downtimes, etc.

GPU Hardware

Card Model	Quantity	Memory	Additional Info
NVIDIA Kepler K20	40	6GB	2013 Expansion
NVIDIA Fermi	8	2GB	

Server Hardware

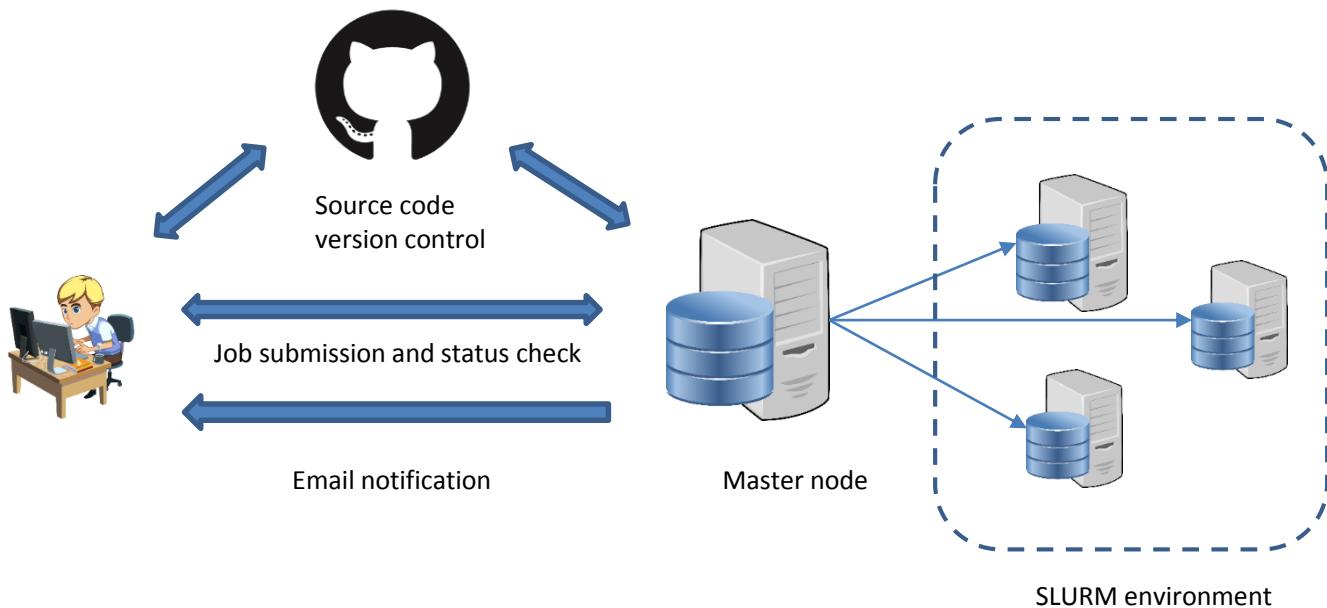
Nodes	Core Count	Processors	Memory per node	Interconnect	Additional Info
138	1656	2 x Intel Xeon E5649 (Six Core)	24GB	QDR InfiniBand	
128	2048	2 x Intel Xeon E5-2670 (Eight Core)	32GB	QDR InfiniBand	2013 Expansion
68	816	2 x Intel Xeon E5-2630 (Six Core)	24GB	QDR InfiniBand	
40	800	2 x Intel Xeon E5-2650 v3 (10-core)	128GB	QDR InfiniBand	hii02 partition
36	288	2 x AMD Opteron 2384 (Quad Core)	16GB	DDR InfiniBand	
34	408	2 x AMD Opteron 2427 (Six Core)	24GB	DDR InfiniBand	
20	320	2 x Intel Xeon E5-2650 v2 (Eight Core)	192GB	QDR InfiniBand	hii01 partition
20	320	2 x Intel Xeon E5-2650 v2 (Eight Core)	64GB	QDR InfiniBand	hii01 partition
16	192	2 x Intel Xeon E5-2620 (Six Core)	64GB	QDR InfiniBand	hii01 partition
4	48	2 x Intel Xeon E5649 (Six Core)	24GB	QDR InfiniBand	Login nodes
4	80	2 x Intel Xeon E5-2650 v3 (10-core)	512GB	QDR InfiniBand	2015 Large-memory nodes
2	32	2 x AMD Opteron 6128 (Eight Core)	192GB	DDR InfiniBand	Large-memory nodes
2	32	2 x AMD Opteron 6128 (Eight Core)	18GB	DDR InfiniBand	
1	16	4 x Intel Xeon E7330 (Quad Core)	132GB	SDR InfiniBand	Large-memory node
1	16	2 x Intel Xeon E5-2650 (Eight Core)	32GB	QDR Infiniband	Chemistry GPU node
Totals		520	7168	24.6TB	

File System Hardware

File System Path	File System Type	Interconnect	Available Size	Backed Up?	Long-Term Storage	Additional Info
/home	GPFS	QDR Infiniband	2.4PB	Daily	Yes	home directory space for secure file storage



Run R code on CIRCE @USF





Reference

- (Petra Kuhnert and Bill Venables) *An Introduction to R Software for Statistical Modelling & Computing*, CSIRO Mathematical and Information Sciences Cleveland, Australia
- Other books and materials
 - CRAN project: <https://cran.r-project.org/other-docs.html>



Outline

- Motivation – Why learn R?
- Programming IDE – R studio, Workspace, Console
- Programming – objects, loops, conditionals, function
- File Input/output
- R- Packages
- Data manipulation
- Database connector – MySQL
- Visualization
- Datasets
- Social Media APIs – Twitter, Facebook, Google Trends, Youtube
- CIRCE @USF (Cluster Computing)



R studio

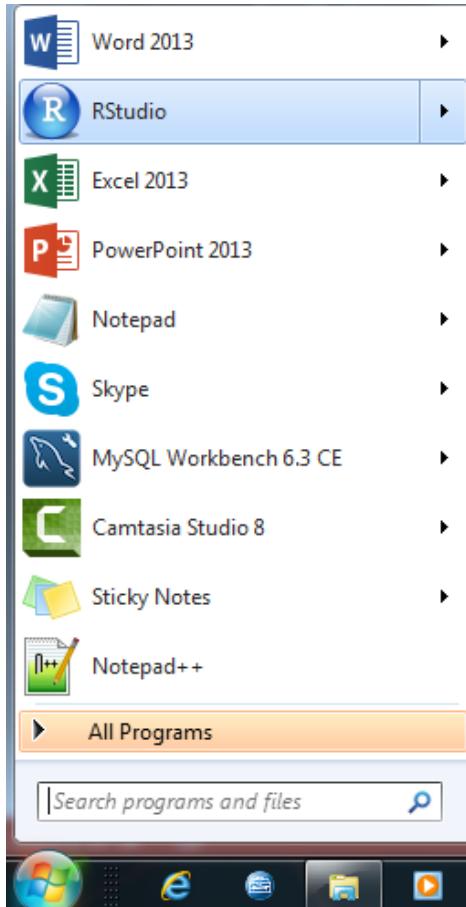
- Download R-studio desktop
 - <https://www.rstudio.com/products/rstudio/download3/>
 - GPL license
 - Windows, Mac OS X, Linux (Ubuntu)



Programming IDE- R Studio

- Coding
- Execution
- Debugging
- Batch mode execution
- R-workspace

R-studio interface



To quit R, close the R studio or use q() function



R-studio interface

The screenshot shows the RStudio interface with four main windows highlighted by hand-drawn speech bubbles:

- Code window**: Located at the top left, showing the R script editor with code like `install.packages("package_name")`.
- Variables window**: Located at the top right, showing the Global Environment pane with various R objects and their details.
- Execution window**: Located at the bottom left, showing the Console pane with a history of R commands and their outputs.
- Help/Visualization window**: Located at the bottom right, showing the Help pane with search results for "arules".

Below the windows, the Windows taskbar is visible with icons for various applications like Internet Explorer, File Explorer, and Microsoft Word.

```
1:33 (Top Level) R Script
Console C:/Users/ThinkPad/Box Sync/ContentSharing/
16 88/4 3629.0/0 0.1534810 82.4
17 10962 4608.660 0.2043140 58.6
18 10743 4787.620 0.2627270 58.6
19 11878 4864.220 0.2000710 58.6
20 9867 4479.410 0.1448100 58.6
21 7838 3428.741 0.1138520 142.0
22 11876 4353.140 0.2910290 142.0
23 12212 4697.650 0.2400770 142.0
24 11875 4353.140 0.1618070 142.0
25 6360 3977.390 0.1794570 740.0
26 4193 1379.350 0.1794550 740.0
27 7416 1916.240 0.1918020 740.0
28 5246 1585.420 0.1320830 740.0
29 6509 1851.210 0.2252140 890.0
30 4895 1239.661 0.3412730 890.0
31 6775 1728.141 0.3116461 890.0
32 7894 1461.061 0.2760161 890.0
33 5980 1426.760 0.1976530 950.0
34 5318 990.388 0.3266350 950.0
35 7392 1350.761 0.1541920 950.0
36 7894 1461.061 0.2760161 950.0
37 3469 1376.701 0.1769691 100.0
38 1468 476.322 0.4387120 100.0
39 3521 1145.690 0.1635350 100.0
40 3267 1644.210 0.2538320 100.0
41 5048 941.543 0.2300810 1300.0
42 1016 308.642 0.2300810 1300.0
43 5605 1145.690 0.4641250 1300.0
44 8793 2280.490 0.4204770 1300.0
45 3475 1174.110 0.2007440 580.0
46 1651 597.808 0.2626510 580.0
47 5514 1455.880 0.1824530 580.0
48 9718 1485.580 0.2004470 580.0
>
```



Hello world!!

Method 1:

Write following code in the execution window

```
>print('Hello World')
```

Clear console: CTRL+l

Method 2:

Write the code in the code window

Save as 'script.R' (optional)

Select the code and click Run button OR (CTRL+ Enter)

Save workspace - `save.image(file='helloWorld.RData')`

Running from Command prompt: `R CMD BATCH script.R`



Running R via command Line

The screenshot shows a Windows 7 desktop environment. In the center, there's a File Explorer window showing the contents of the 'bin' folder of the R-3.3.1 installation. The folder contains several files and subfolders: 'i386', 'x64', 'config.sh', 'R', and 'Rscript'. The 'R' file is selected. Below the File Explorer is a terminal window titled 'Rterm (64-bit)'. The terminal displays the standard R startup message, including the version number (3.3.1), copyright information, and a welcome message about the software being free and redistributable under certain conditions. At the bottom of the terminal window, there's a prompt '>'. The taskbar at the bottom of the screen shows various icons for common Windows applications like Internet Explorer, File Explorer, and the Start button.

Computer > Windows7_OS (C:) > Program Files > R > R-3.3.1 > bin >

Name	Date modified	Type	Size
i386	8/31/2016 10:37 PM	File folder	
x64	8/31/2016 10:37 PM	File folder	
config.sh	6/21/2016 2:32 PM	SH File	10 KB
R	6/21/2016 2:37 PM	Application	87 KB
Rscript	6/21/2016 2:37 PM	Application	87 KB

Rterm (64-bit)

```
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\ThinkPad>cd C:\Program Files\R\R-3.3.1\bin

C:\Program Files\R\R-3.3.1\bin>R

R version 3.3.1 (2016-06-21) -- "Bug in Your Hair"
Copyright (C) 2016 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> 
```

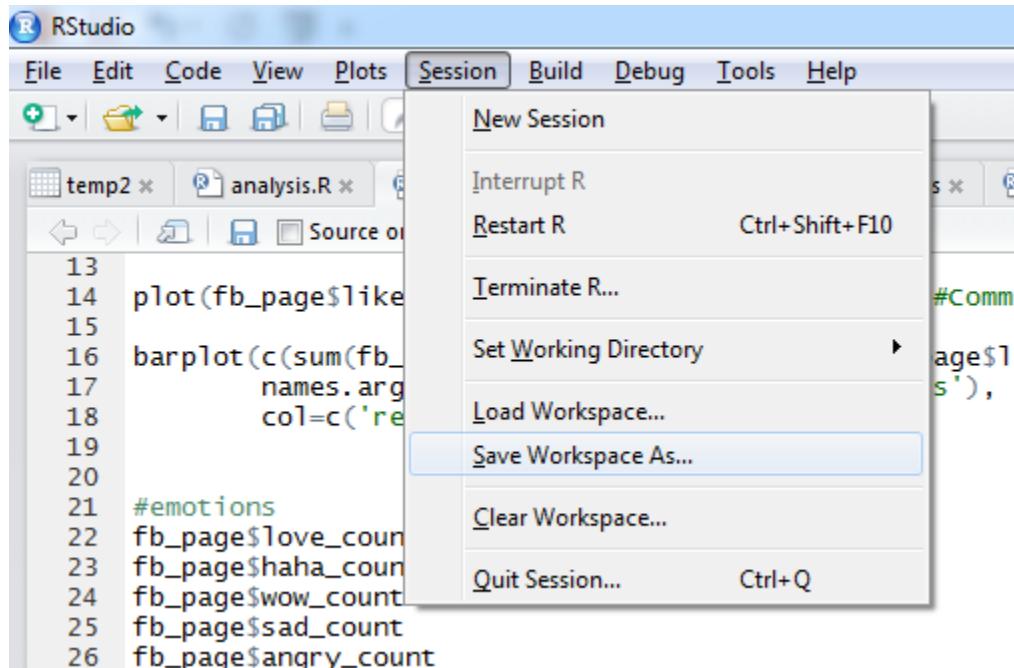


Running R via command Line

- Why?
 - Good for automation and running on Clusters such as USF- CIRCE
[\(http://www.usf.edu/it/research-computing/\)](http://www.usf.edu/it/research-computing/)
- Command
 - "C:\Program Files\R\R-3.3.1\bin\R.exe" CMD BATCH script.R
 - Creates an output file: script.Rout



Save R- workspace

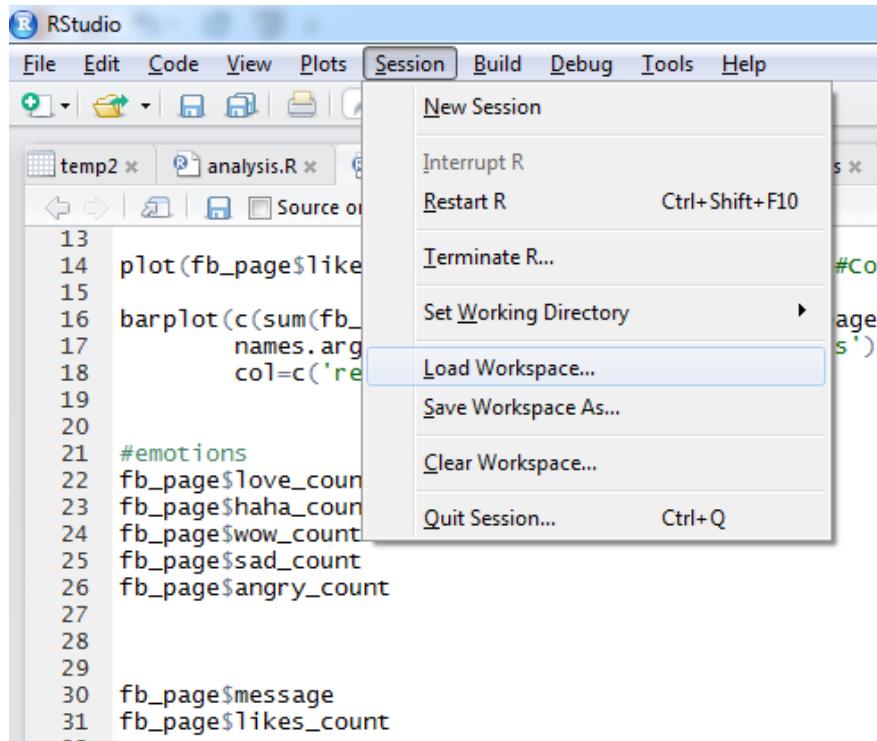


```
save.image("C:/Users/ThinkPad/Box Sync/R workshop/helloworld.RData")
```

*Keep one workspace for each project



Load R- workspace

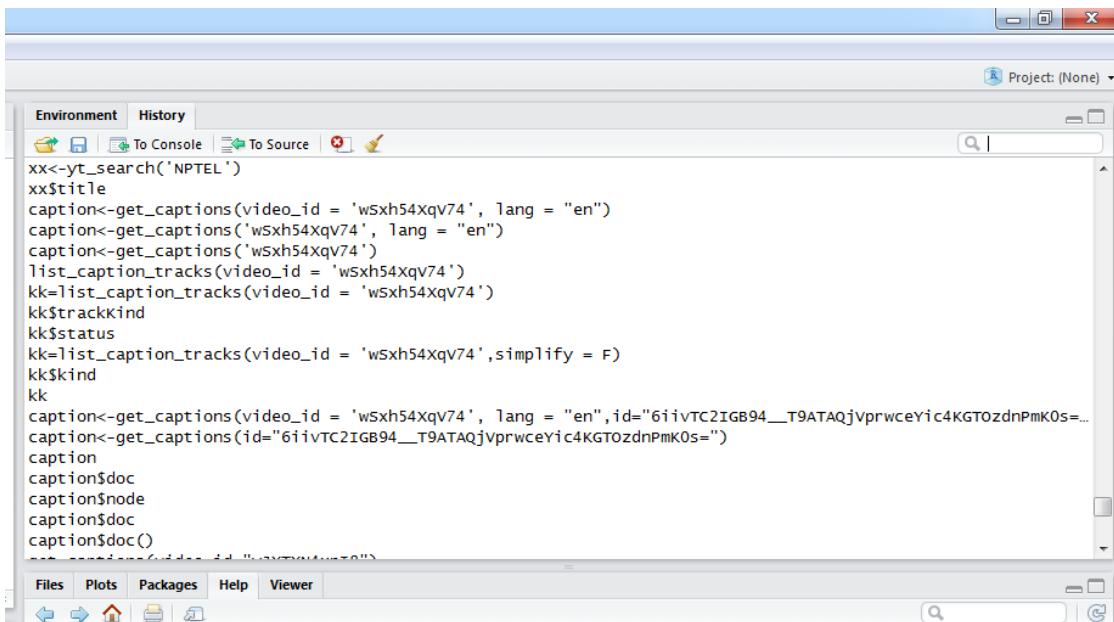


```
load("C:/Users/ThinkPad/Box Sync/R workshop/helloworld.RData")
```

<https://github.com/vivek14632/R-Programming-workshop/tree/master/workspace>

History of commands

- `history()`



The screenshot shows the RStudio interface with the History tab selected in the top navigation bar. The main pane displays a series of R commands run in the console, including function calls like `xx<-yt_search('NPTEL')`, variable assignments like `xx$title`, and other operations like `caption<-get_captions(video_id = 'wsxh54xqv74', lang = "en")`. The code is scrollable, showing multiple lines of history.

```
xx<-yt_search('NPTEL')
xx$title
caption<-get_captions(video_id = 'wsxh54xqv74', lang = "en")
caption<-get_captions('wsxh54xqv74', lang = "en")
caption<-get_captions('wsxh54xqv74')
list_caption_tracks(video_id = 'wsxh54xqv74')
kk=list_caption_tracks(video_id = 'wsxh54xqv74')
kk$trackkind
kk$status
kk=list_caption_tracks(video_id = 'wsxh54xqv74', simplify = F)
kk$kind
kk
caption<-get_captions(video_id = 'wsxh54xqv74', lang = "en", id="6iivTC2IGB94__T9ATAQjvprwceYic4KGTOzdnPmK0s=...")
caption<-get_captions(id="6iivTC2IGB94__T9ATAQjvprwceYic4KGTOzdnPmK0s=")
caption
caption$doc
caption$node
caption$doc
caption$doc()
...
```

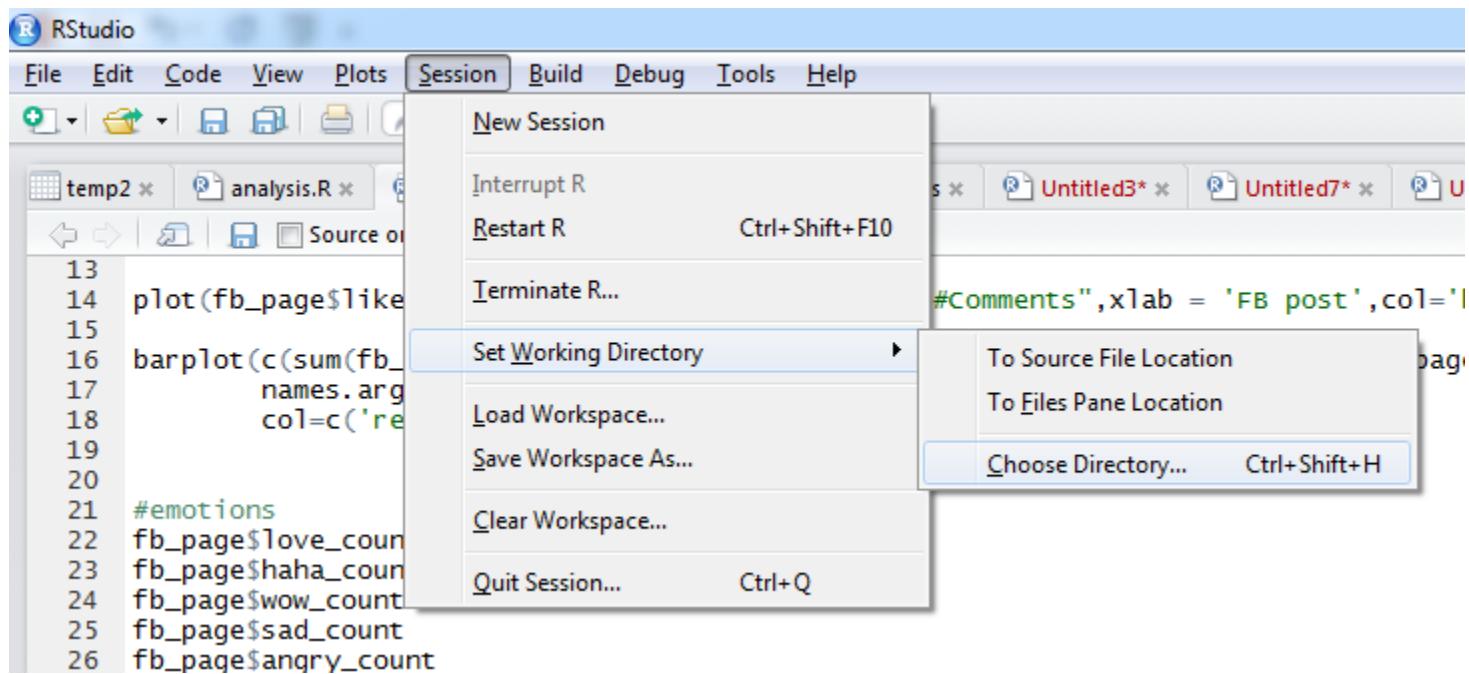


R-working directory

- Check working directory
 - `getwd()`
- Set working directory
 - `setwd('path_to_directory')`



R-working directory



```
setwd("C:/Users/ThinkPad/Box Sync/R workshop/code")
```



Debug R program

- print () function
- Line by line execution
- Breakpoints
 - Using GUI
 - browser() function



Outline

- Motivation – Why learn R?
- Programming IDE – R studio, workspace, Console
- **Programming – objects, loops, conditionals, function**
- File Input/output
- R- Packages
- Data manipulation
- Database connector – MySQL
- Visualization
- Datasets
- Social Media APIs – Twitter, Facebook, Google Trends, Youtube
- CIRCE @USF (Cluster Computing)



R Objects

- Data types: Numeric, Integer, Logical, Complex, Character, Raw
- Different types of Objects
 - Vector
 - Set of elements of same mode: logical, numeric (integer, double), complex, character, list
 - Matrix
 - Rows and columns of same mode: logical, numeric (integer or double), complex or character.
 - Data frame: similar to matrix but the columns can be of different modes
 - List: generalization of vector with a collection of data objects
- Class of an Object
- Example code: <https://github.com/vivek14632/R-Programming-workshop/tree/master/ObjectsInR>



Vectors

- Different types of vectors
 - Numeric vectors
 - Character vectors
 - Logical vector
 - Complex vector

<https://github.com/vivek14632/R-Programming-workshop/blob/master/ObjectsInR/vectors.R>



Vectors

- Creating a vector
 - `c()` function
 - `seq()` function
 - `rep()` function
 - `:` operator
 - Creating vector at run-time
- Even a single value is a vector
- Vector repetition: benefits and challenges

<https://github.com/vivek14632/R-Programming-workshop/blob/master/ObjectsInR/vectors.R>



Matrix

- Converting vector to matrix
 - dim() function
 - Creates matrix by column
 - Can also convert matrix to vector
 - matrix() function
 - matrix (vectorName, #rows,#columns)
 - By row: matrix (vectorName, #rows,#columns, byRow=T)
- rbind() function
- cbind() function

<https://github.com/vivek14632/R-Programming-workshop/blob/master/ObjectsInR/matrix.R>



Data Frame

- Similar to matrix
- Contains data columns with different modes – character, numeric, logical,etc.
- Convert matrix to data frame
 - `data.frame()`
- Columns names: `names()`

<https://github.com/vivek14632/R-Programming-workshop/blob/master/ObjectsInR/dataFrame.R>



List

- Combination of vector, matrix, data frames with different data types
- Used for storing different forms of output and return it from a function
- Display the output

<https://github.com/vivek14632/R-Programming-workshop/blob/master/ObjectsInR/list.R>



Functions

- Using library functions
- User defined functions
- Checking function definitions
- Modifying library functions (optional)

Example code: <https://github.com/vivek14632/R-Programming-workshop/tree/master/functions>



Loops

- Different types of loops
 - For loop
 - Repeat loop
 - While loop
- For and While loop is most widely used.

[https://github.com/vivek14632/R-
Programming-workshop/tree/master/loops](https://github.com/vivek14632/R-Programming-workshop/tree/master/loops)



Conditionals

- If condition
- Else condition
- Else if condition
- Ifelse condition
- [https://github.com/vivek14632/R-
Programming-
workshop/blob/master/conditionals/conditional
s.R](https://github.com/vivek14632/R-Programming-workshop/blob/master/conditionals/conditional.s.R)



Outline

- Motivation – Why learn R?
- Programming IDE – R studio, workspace, Console
- Programming – objects, loops, conditionals, function
- [File Input/output](#)
- R- Packages
- Data manipulation
- Database connector – MySQL
- Visualization
- Datasets
- Social Media APIs – Twitter, Facebook, Google Trends, Youtube
- CIRCE @USF (Cluster Computing)



Different file formats

- Clipboard
- CSV - read and write
- JSON format
- XLSX format
- User inputs via command line
- Text file
- System directory

Example code: <https://github.com/vivek14632/R-Programming-workshop/tree/master/fileIO>



Outline

- Motivation – Why learn R?
- Programming IDE – R studio, workspace, Console
- Programming – objects, loops, conditionals, function
- File Input/output
- R- Packages
- Data manipulation
- Database connector – MySQL
- Visualization
- Datasets
- Social Media APIs – Twitter, Facebook, Google Trends, Youtube
- CIRCE @USF (Cluster Computing)



Packages in R

- Installation
- Loading
- Updating packages
- Uninstall packages
- Example code:
 - <https://github.com/vivek14632/R-Programming-workshop/blob/master/package/packages.R>



Outline

- Motivation – Why learn R?
- Programming IDE – R studio, workspace, Console
- Programming – objects, loops, conditionals, function
- File Input/output
- R- Packages
- **Data manipulation**
- Database connector – MySQL
- Visualization
- Datasets
- Social Media APIs – Twitter, Facebook, Google Trends, Youtube
- CIRCE @USF (Cluster Computing)



Data manipulation functions

- Table
- Subset
- Split
- Sort
- cbind()
- rbind()
- date and time
- apply

Example code: <https://github.com/vivek14632/R-Programming-workshop/tree/master/dataManipulation>



Outline

- Motivation – Why learn R?
- Programming IDE – R studio, workspace, Console
- Programming – objects, loops, conditionals, function
- File Input/output
- R- Packages
- Data manipulation
- [Database connector – MySQL](#)
- Visualization
- Datasets
- Social Media APIs – Twitter, Facebook, Google Trends, Youtube
- CIRCE @USF (Cluster Computing)



DATABASE CONNECTION



MySQL database and R

- Package: RMySQL
- Steps
 - Install package
 - Load package
 - create database connection
 - execute SQL query
- <https://github.com/vivek14632/R-Programming-workshop/blob/master/databaseConnection/mysql.R>



Outline

- Motivation – Why learn R?
- Programming IDE – R studio, workspace, Console
- Programming – objects, loops, conditionals, function
- File Input/output
- R- Packages
- Data manipulation
- Database connector – MySQL
- **Visualization**
- Datasets
- Social Media APIs – Twitter, Facebook, Google Trends, Youtube
- CIRCE @USF (Cluster Computing)



VISUALIZATION



Visualization

- Useful functions for graphs
- `plot()` - frequently used function for plotting
- `xyplot()`
- `legend()` - adding legend
- `points()` - addling points to an existing plot
- `lines()` - adding lines to an existing plot



Plot() basic parameters

- type
 - 'l' → line
 - 'p' → point
 - 'b' → both line and point
- ylab → “label of Y-axis”
- xlab → “label of X-axis”
- xlim → range of X-axis
 - c(lower Value, Upper value)
- ylim → “range of Y-axis”
 - c(lower Value, Upper value)
- main → “title of the plot”



Plot() parameters

- pch → type of character used in the point plots
- lty → line types
- col → color of lines and points
- bty → type of the box to enclose the graph
- lab → change axis scale
- Please check the URL for different forms of points, lines, and colors
 - <http://www.statmethods.net/advgraphs/parameters.html>

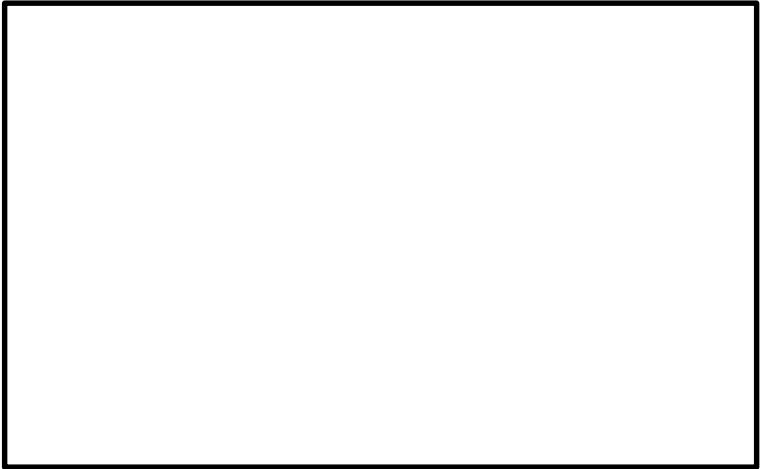
axis() function

- Add axis to an existing plot

side 3

side 2

side 4



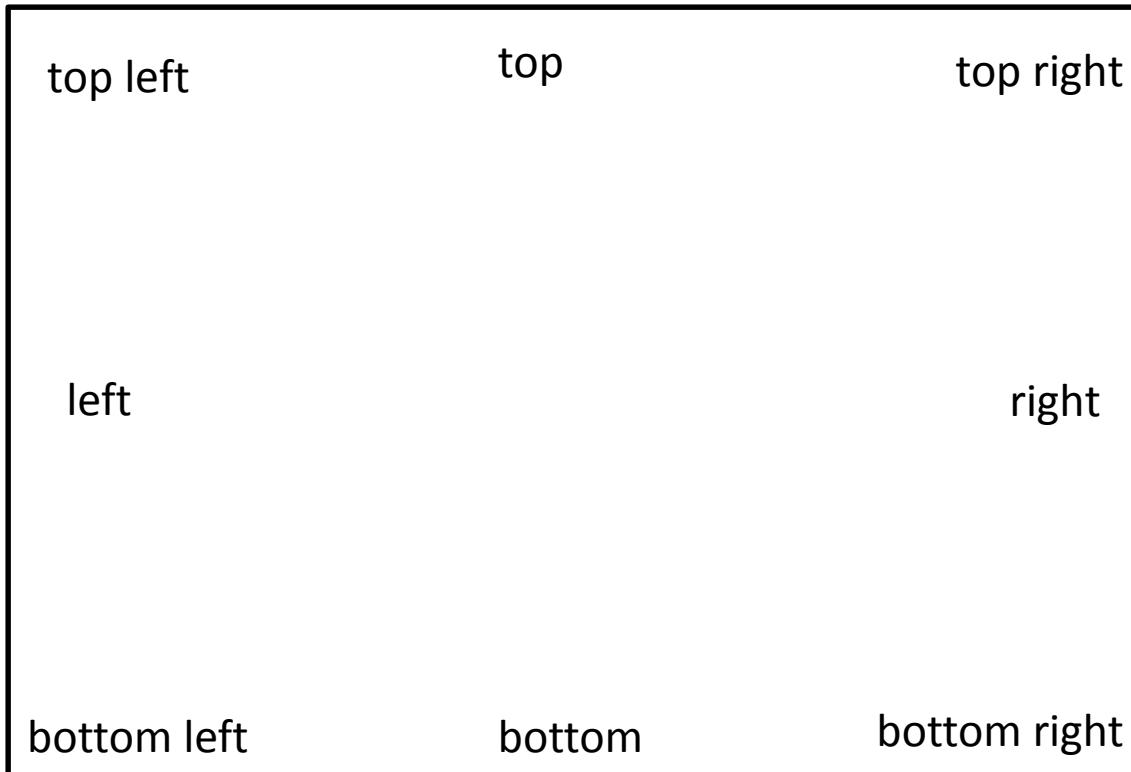
side 1



axis() parameters

- side → side number to add axis
 - select one of the values {1,2,3,4}
- labels → whether to show label on the axis or not
 - select one of the values {T,F}
- tick → whether to add tick to the axis or not
 - select one of the values {T,F}
- line → distance between label and graph
 - any real number preferably between [0,1]
- pos → shift position of the axis

legend () function





legend() function

- `legend('topright',c('Minimum price','Maximum Price'),pch = c(1,0),col=c('black','red'),lty = c(2,3))`
- `c('Minimum price','Maximum Price')` → variable names



Multiple plots

- `par(mfrow=c(number of rows,number of columns))`
 - Example: `par(mfrow=c(2,3))`
 - Appears by row
- `par(mfcol=c(number of rows,number of columns))`
 - Appears by columns



Types of plots

- Generic Plot or Plot
- Density plot
- Histogram
- Geographical Map
- QQ plot
- Time series plots

<https://github.com/vivek14632/R-Programming-workshop/tree/master/Visualization>



Outline

- Motivation – Why learn R?
- Programming IDE – R studio, workspace, Console
- Programming – objects, loops, conditionals, function
- File Input/output
- R- Packages
- Data manipulation
- Database connector – MySQL
- Visualization
- **Datasets**
- Social Media APIs – Twitter, Facebook, Google Trends, Youtube
- CIRCE @USF (Cluster Computing)



DATASETS



Accessing datasets in R

```
>install.packages('MASS')
>library('MASS')
#dataset
>quine
```

Other R dataset packages

(1) datasets

<http://stat.ethz.ch/R-manual/R-devel/library/datasets/html/00Index.html>

<https://github.com/vivek14632/R-Programming-workshop/tree/master/datasets>



Outline

- Motivation – Why learn R?
- Programming IDE – R studio, workspace, Console
- Programming – objects, loops, conditionals, function
- File Input/output
- R- Packages
- Data manipulation
- Database connector – MySQL
- Visualization
- Datasets
- Social Media APIs – Twitter, Facebook, Google Trends, Youtube
- CIRCE @USF (Cluster Computing)



R AND SOCIAL MEDIA



Social Media APIs

- R Packages
 - Facebook API – Rfacebook
 - Twitter API – twitterR
 - Youtube API – tuber
 - Google Trends API – gtrendsR
- Steps
 - Authentication using Oauth
 - Use API functions
- <https://github.com/vivek14632/R-Programming-workshop/tree/master/socialMedia>



Outline

- Motivation – Why learn R?
- Programming IDE – R studio, workspace, Console
- Programming – objects, loops, conditionals, function
- File Input/output
- R- Packages
- Data manipulation
- Database connector – MySQL
- Visualization
- Datasets
- Social Media APIs – Twitter, Facebook, Google Trends, Youtube
- [CIRCE @USF \(Cluster Computing\)](#)



Working with CIRCE

- SLURM Job scheduler
- R script
- Submission script
- Important command



Submission script

```
#!/bin/bash
#
#SBATCH --comment=r-test
#SBATCH --ntasks=4
#SBATCH --job-name=r-test
#SBATCH --output=output.%j.r-test
#SBATCH --time=01:00:00

#### SLURM 4 processor R test to run for 1 hour.

module purge
module add apps/R/3.1.2

mpirun Rmpi test.R
```



Important commands

- sbatch: Submit jobs to SLURM
- squeue: Check your job status
- scancel: cancel your job

<https://github.com/vivek14632/R-Programming-workshop/tree/master/CIRCE>



Optional

- Distribution
 - Working with standard distribution such as normal poisson, and uniform.
- Financial data
 - Quantmod library
 - <https://github.com/vivek14632/R-Programming-workshop/blob/master/qunatitativeTrading/gettingData.R>