



# 3<sup>rd</sup> R-Programming Bootcamp

August 18, 2017

Vivek Singh

Information Systems Decision Sciences (ISDS)

MUMA College Of Business

[vivek4@mail.usf.edu](mailto: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)

1<sup>st</sup> R Bootcamp [January 27, 2017]

2<sup>nd</sup> R Bootcamp [February 17, 2017]

### Fall 2017

ISM 3113: System Analysis Design

3<sup>rd</sup> R Bootcamp [August 18, 2017]

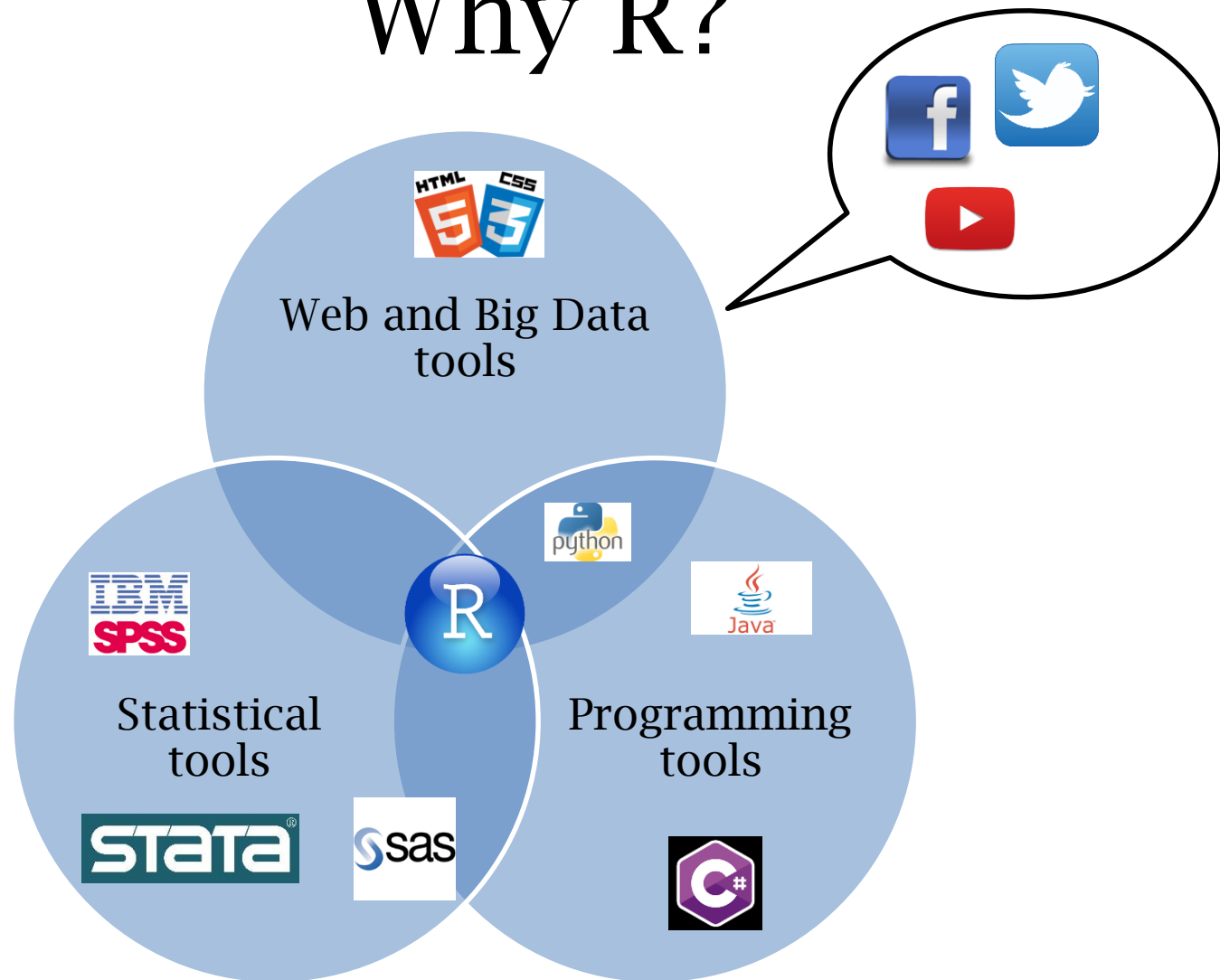


# Outline

- 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 maters:** [University of Auckland](#), [University of California, Berkeley](#)

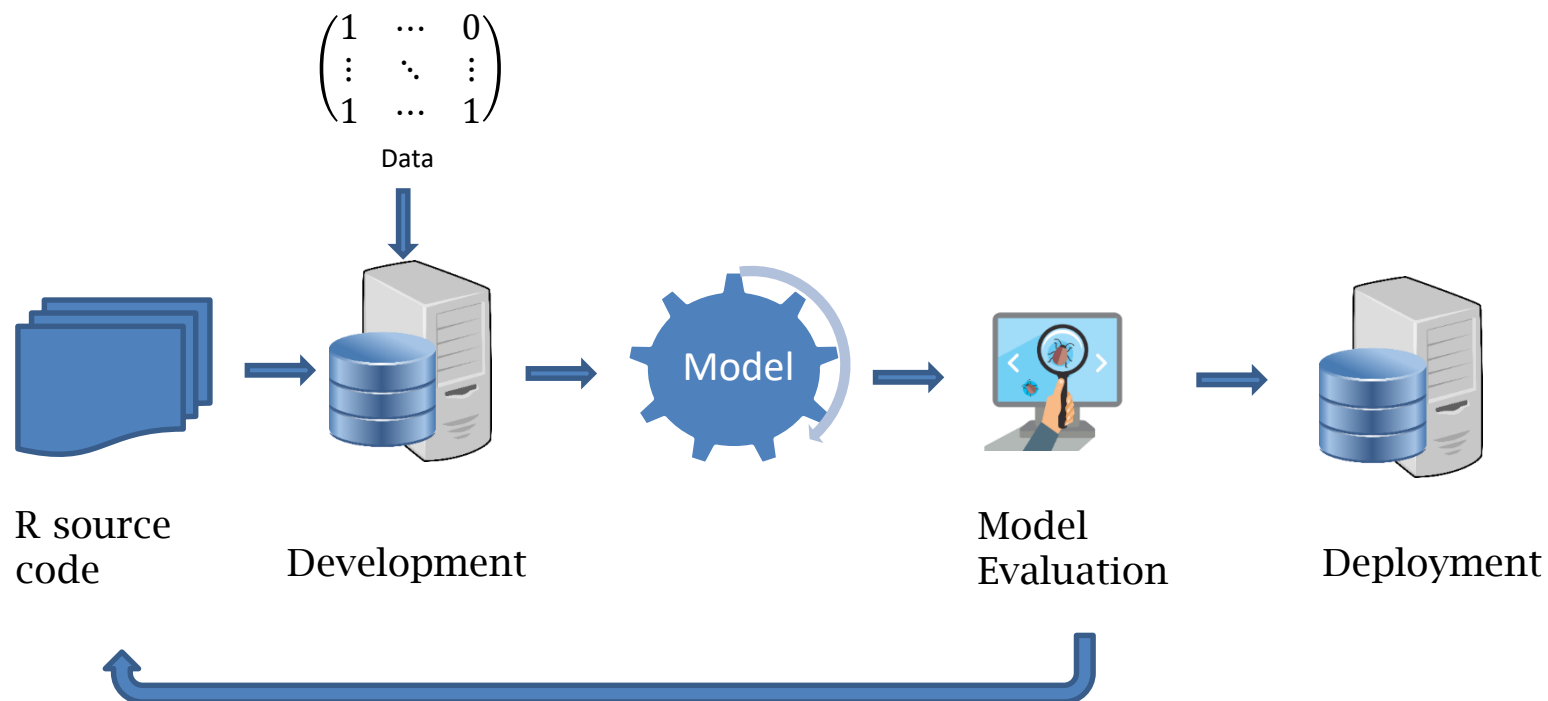
**Robert Gentleman**

Programming Language Designer



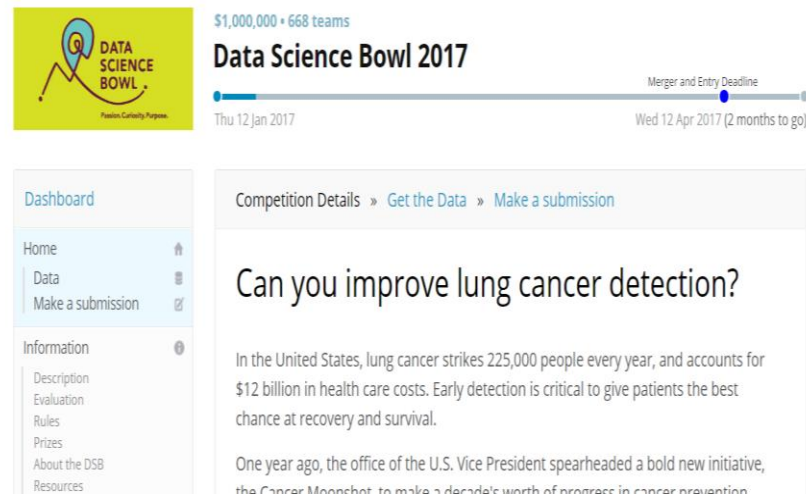


# Data Analytics Life Cycle





# \$ 1,000,000

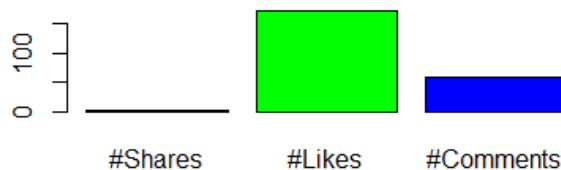
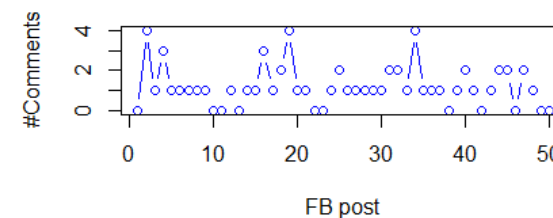
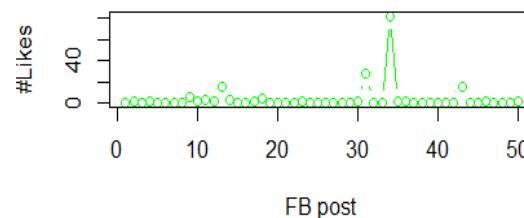
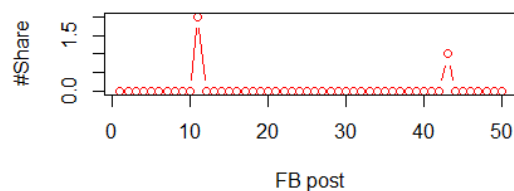
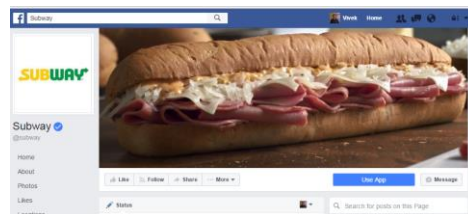


The image shows a screenshot of the Data Science Bowl 2017 competition page. At the top, there is a yellow banner with the competition logo and the text "\$1,000,000 • 668 teams". Below this, the title "Data Science Bowl 2017" is displayed. A progress bar indicates the "Merger and Entry Deadline" is on "Wed 12 Apr 2017 (2 months to go)". The main content area features a sidebar with navigation links: Dashboard, Home, Data, Make a submission, Information, Description, Evaluation, Rules, Prizes, About the DSB, and Resources. The main content area has a heading "Can you improve lung cancer detection?" and a paragraph stating: "In the United States, lung cancer strikes 225,000 people every year, and accounts for \$12 billion in health care costs. Early detection is critical to give patients the best chance at recovery and survival. One year ago, the office of the U.S. Vice President spearheaded a bold new initiative, the Cancer Moonshot, to make a decade's worth of progress in cancer prevention."

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



# Social Media Analytics Demo with R

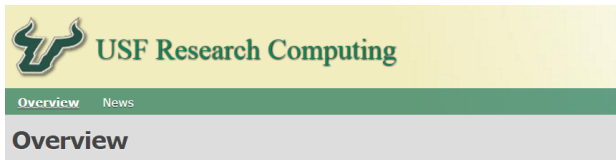


DV: #Share	Std.		t value	Pr(> t )	
	Estimate	Error			
(Intercept)	-0.42	0.15	-2.77	0.01	**
#Likes	0.12	0.01	18.88	< 2e-16	***
#Comments	0.24	0.09	2.59	0.01	*





# Run R code on CIRCE @USF



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.

## 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
<b>Totals</b>	520	7168		24.6TB	

## GPU Hardware

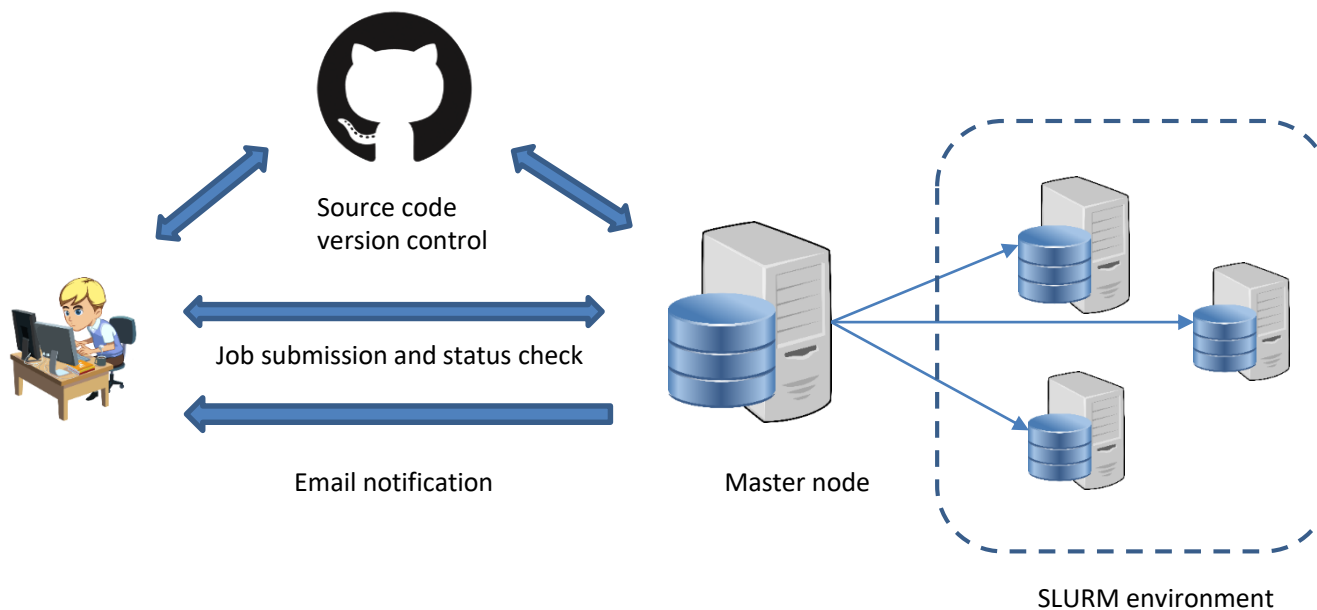
Card Model	Quantity	Memory	Additional Info
NVIDIA Kepler K20	40	6GB	2013 Expansion
NVIDIA Fermi	8	2GB	

## 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/>
  - AGPL license
  - Windows, Mac OS X, Linux (Ubuntu)

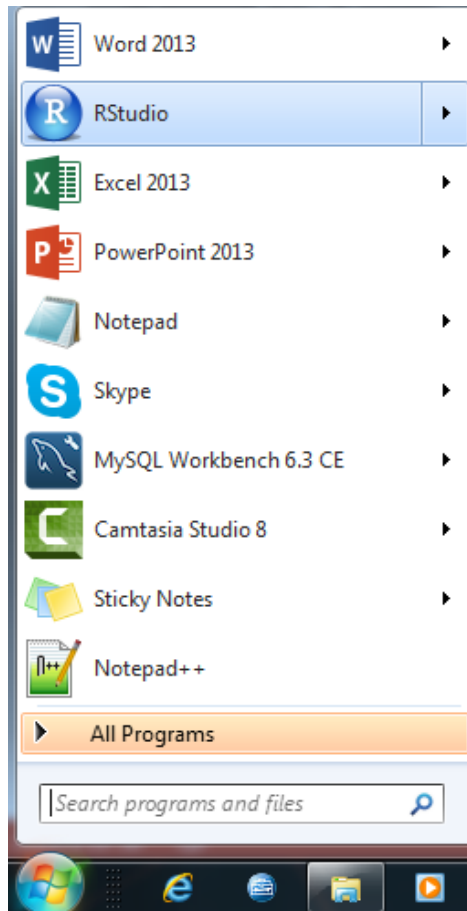


# Programming IDE- R Studio

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



# R-studio interface



To quit R, close the R studio  
or use `q()` function if you are using  
Command line



# R-studio interface

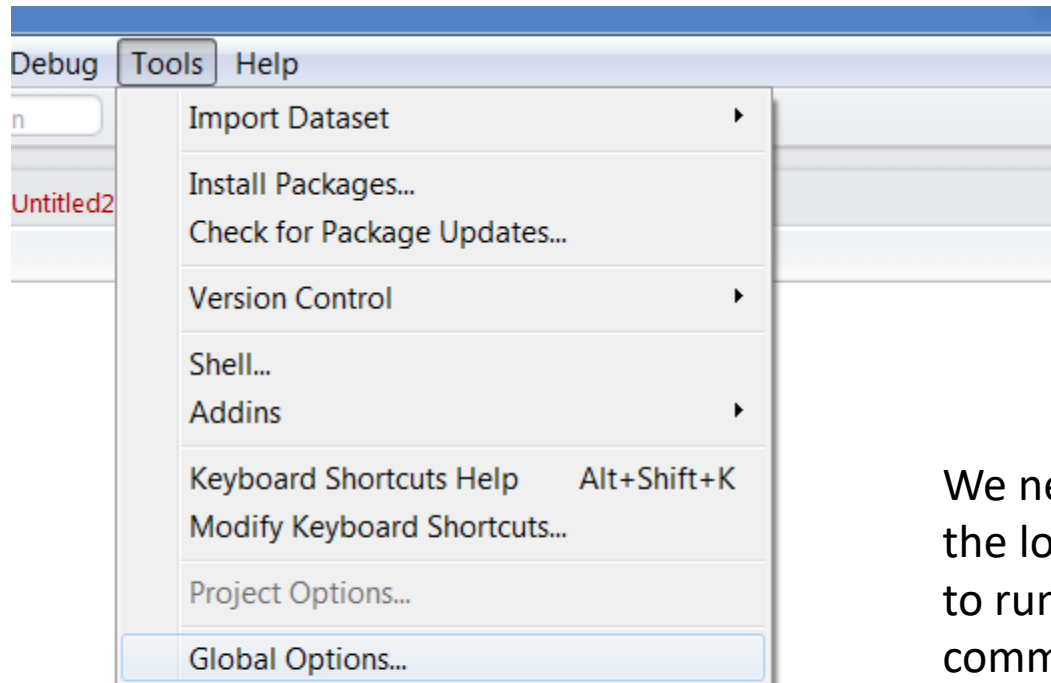
The screenshot shows the RStudio interface with four callouts identifying its main components:

- Code window:** The top-left pane showing the R script editor with the command `install.packages("package_name")`.
- Variables window:** The top-right pane showing the Environment and History tabs. The Environment tab displays loaded objects like `datasetShare` and `fb_page`.
- Execution window:** The bottom-left pane showing the Console output, displaying a large table of numerical data.
- Help/Visualization window:** The bottom-right pane showing the Files, Plots, Packages, Help, and Viewer tabs. The Help tab is active, displaying search results for various datasets.





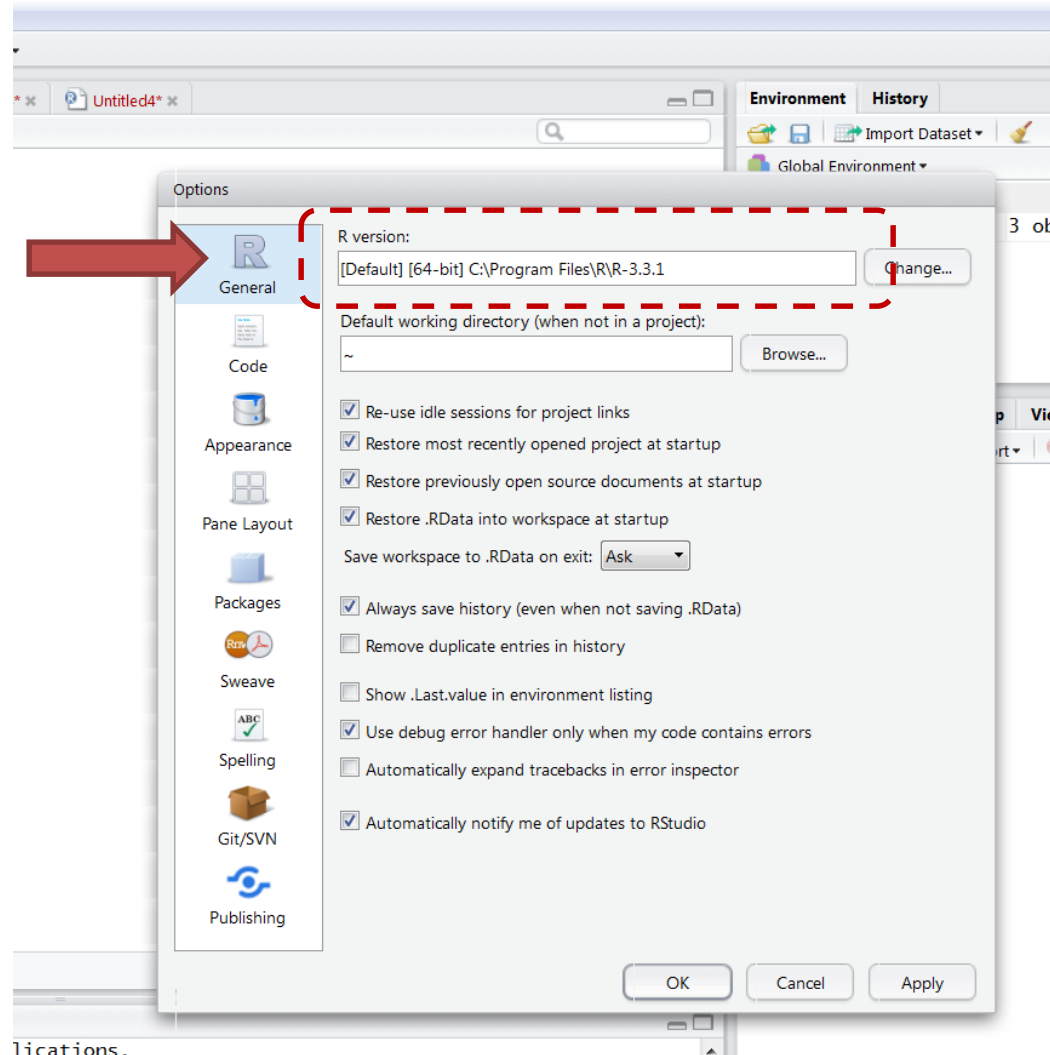
# Location of R program



We need to determine the location of R program to run batch mode or command line executing



# Location of R program





# 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. In the background, a File Explorer window is open to the directory `C:\Program Files\R\R-3.3.1\bin`. The file list includes folders `i386` and `x64`, a file `config.sh`, and applications `R` and `Rscript`. In the foreground, a terminal window titled "Rterm (64-bit)" is open. It displays the output of running `R` from the command line. The terminal shows the Microsoft Windows version (6.1.7601), copyright information, the current directory (`C:\Users\ThinkPad>cd C:\Program Files\R\R-3.3.1\bin`), and the command `C:\Program Files\R\R-3.3.1\bin>R`. The output of `R` includes the version (3.3.1), copyright (2016 The R Foundation), platform (x86\_64-w64-mingw32/x64), and a welcome message with instructions on how to use R.

```
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.

>
```

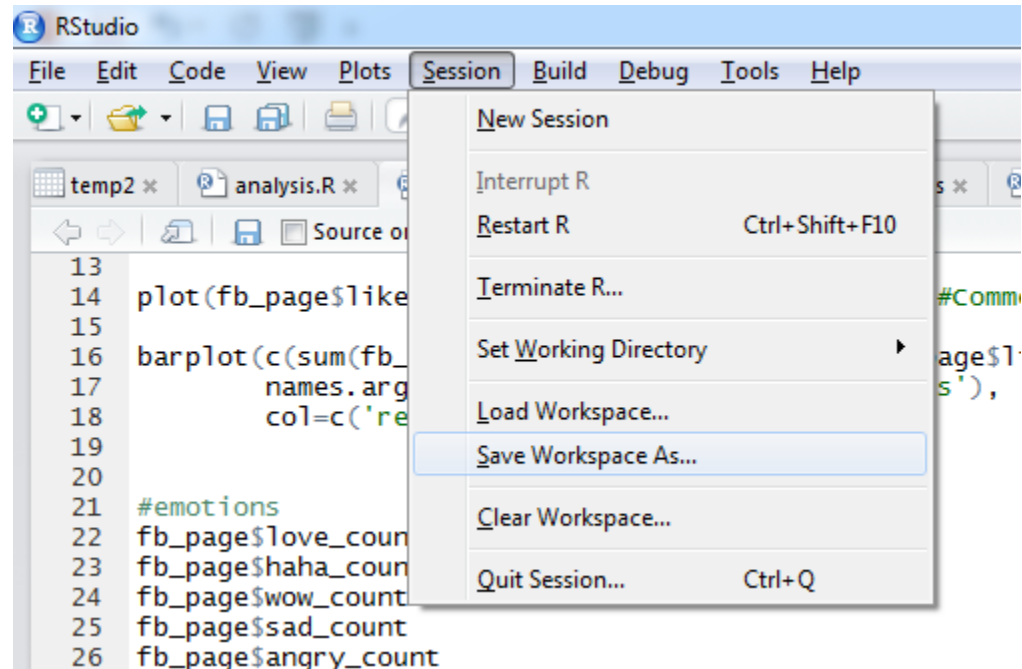


# Running R via command Line

- Why?
  - Good for automation and running on Clusters such as USF- CIRCE  
(<http://www.usf.edu/it/research-computing/>)
- 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

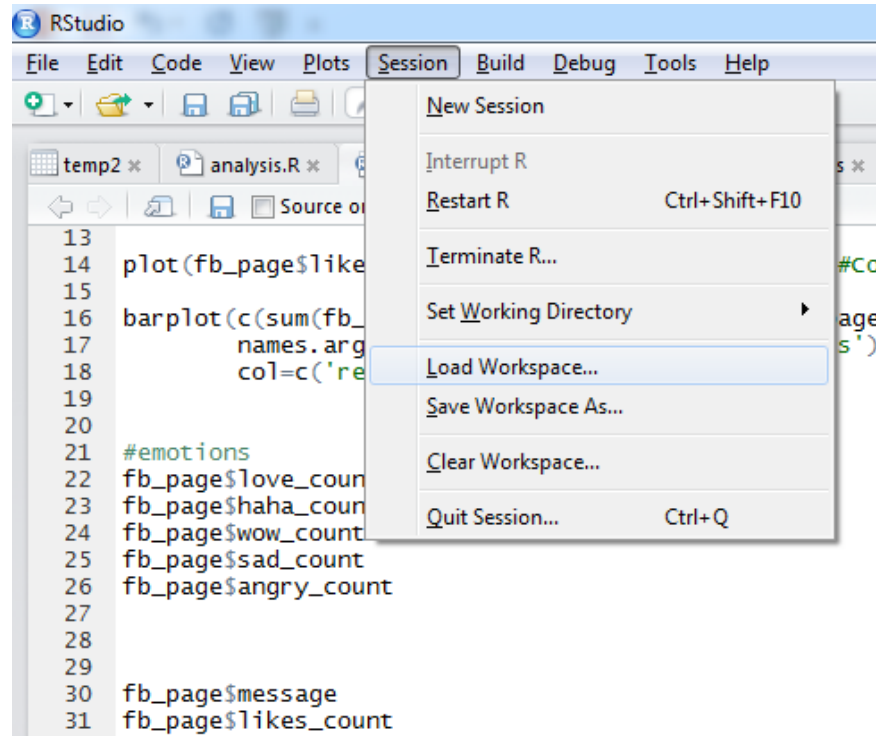


```
save.image("C:/Users/ThinkPad/Box Sync/R workshop/helloworld.RData")
```

\*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/blob/master/workspace/workspace.R>



# Loading Social Media Workspace

- Download the Facebook workspace from github:

<https://github.com/vivek14632/R-Programming-workshop/blob/master/socialMedia/facebook.RData>

- Load the workspace in R studio
- Examine the variables in the workspace 'fb\_page'



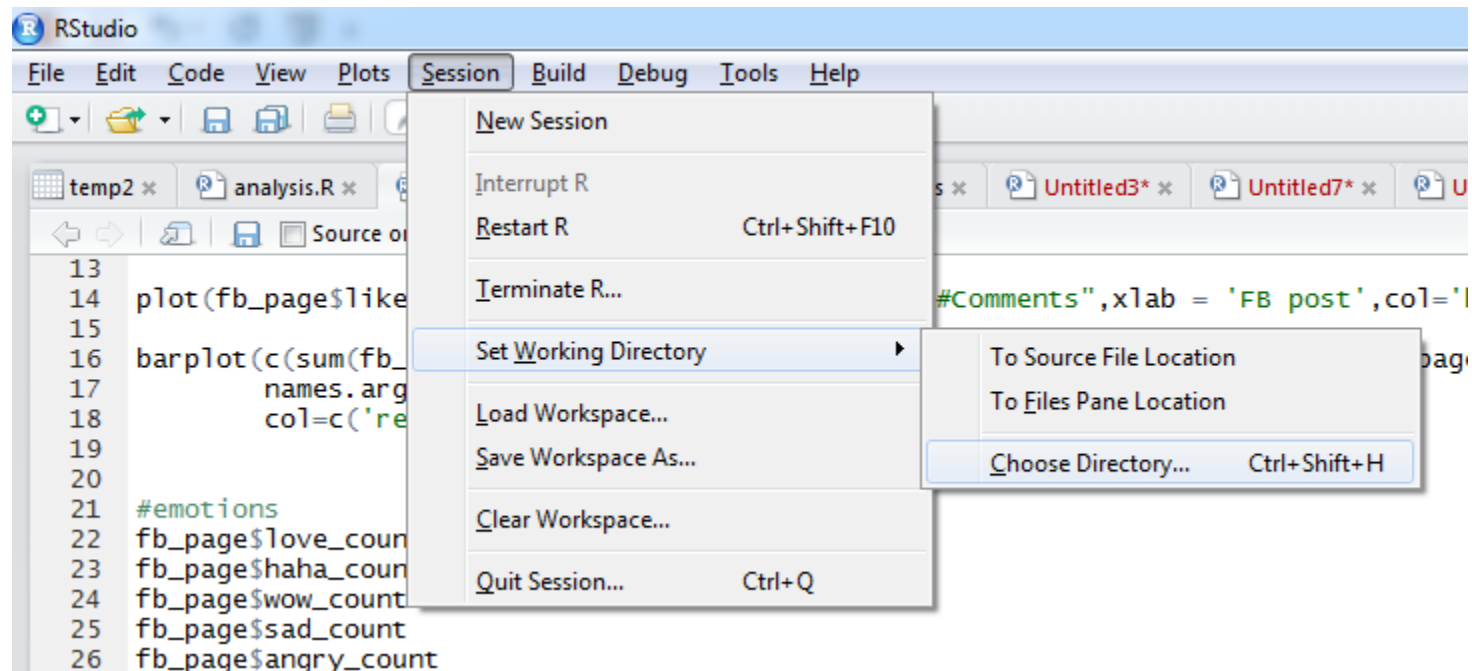


# R-working directory

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



# R-working directory



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



# History of commands

- `history()`

A screenshot of the R Studio interface. The 'Environment' and 'History' tabs are visible at the top. The console window shows a list of commands entered, including `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__T9ATAQjvprwceYic4KGT0zdnPmk0s=...`, `caption<-get_captions(id="6iivTC2IGB94__T9ATAQjvprwceYic4KGT0zdnPmk0s=)`, `caption`, `caption$doc`, `caption$node`, `caption$doc`, and `caption$doc()`. The bottom of the window shows the 'Files', 'Plots', 'Packages', 'Help', and 'Viewer' tabs.

```
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__T9ATAQjvprwceYic4KGT0zdnPmk0s=...
caption<-get_captions(id="6iivTC2IGB94__T9ATAQjvprwceYic4KGT0zdnPmk0s=)
caption
caption$doc
caption$node
caption$doc
caption$doc()
```



# Debug R program

- `print ()` function
- Line by line execution
- Breakpoints
  - Using GUI
  - `browser()` function - <https://github.com/vivek14632/R-Programming-workshop/blob/master/Debug/browser.R>



5 minutes break!!





# 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>



# Conditionals

- If condition
- Else condition
- Else if condition
- Ifelse condition
- <https://github.com/vivek14632/R-Programming-workshop/blob/master/conditionals/conditionals.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>



5 minutes break!!





# 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()` – adding 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”



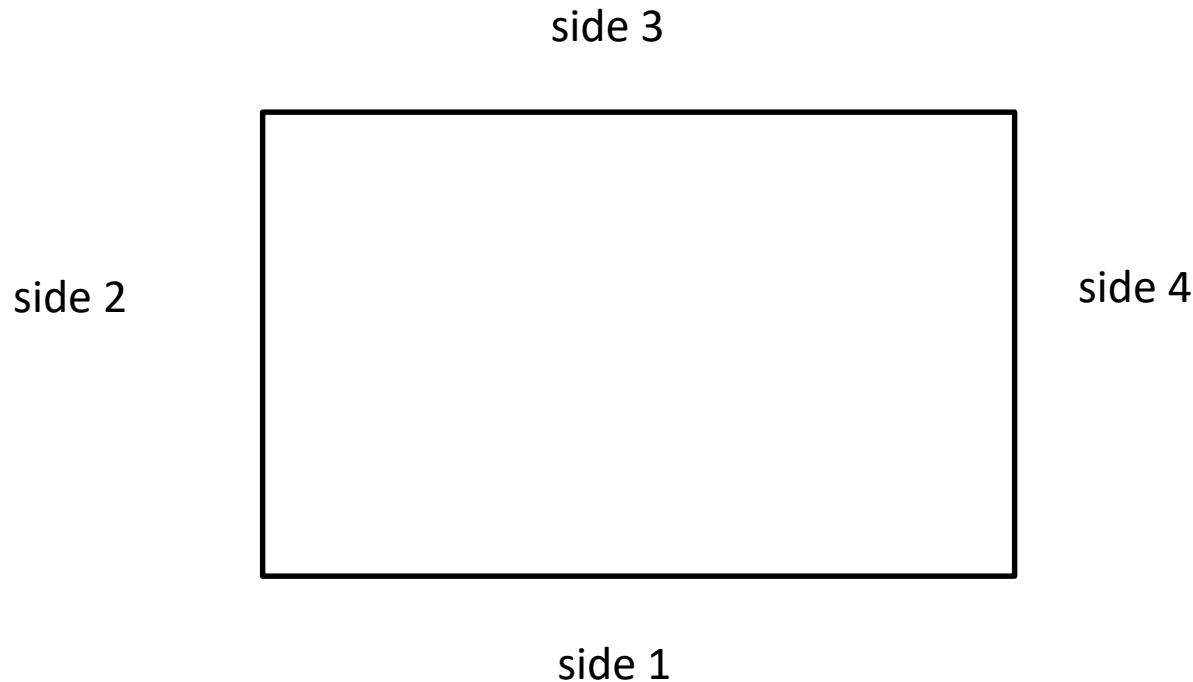
# Plot() parameters

- 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





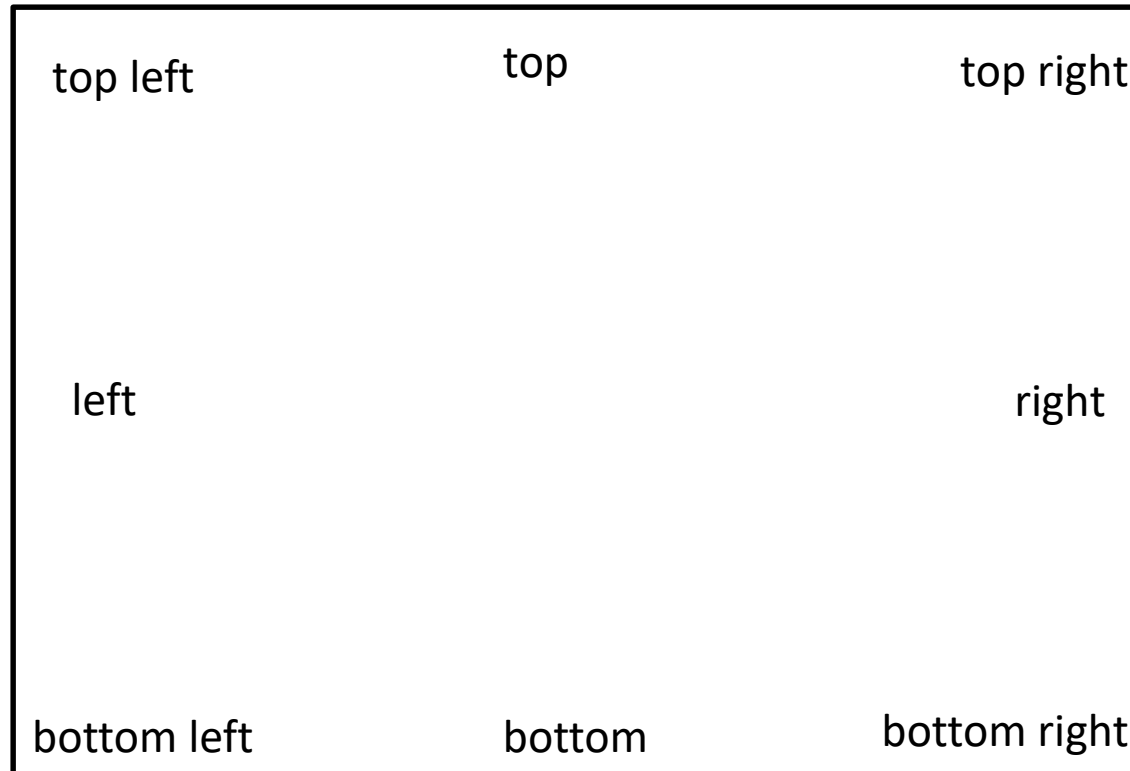
# axis() parameters

- 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>



# Social Media Demo

- Download and load the Facebook workspace from Github:

<https://github.com/vivek14632/R-Programming-workshop/blob/master/socialMedia/facebook.RData>

- Execute the code- facebook.R from line number 10:

<https://github.com/vivek14632/R-Programming-workshop/blob/master/socialMedia/facebook.R>



# 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>