Doing Data Science

Unit 2

Faizan Javed
Data Science @ SMU

Admin notes

HW1 due today

HW2 due 1 hour before live session 3 next Monday

Office hours

Tom Wang (TA): Wednesdays, 6:30 pm EST

Faizan Javed (instructor): Sundays, 8:00 pm EST

Main topics

Basic R programming (functions, control/loop structures)

RStudio

RMarkdown/knitr

Practical tips

Document everything

Ensure compatibility of your software/libraries/packages (troubleshooting assistance)

Comment your code (variable/argument names should not be cryptic)

Source code comment header

Getting started with R R data types: https://www.statmethods.net/input/datatypes.html

Objects (nouns) and Functions (verbs)

#Print R session info -- why is this useful?

```
sessionInfo()
```

# c(combine) function : create vectors (elements have same types)

```
NumVec ← c (2, 3, 4)

CharVec ← c (“doing”, “data”, “science”)
```

```
# data.frame() : create an object with rows and columns

StringNumObj ← data.frame(NumVec, CharVec)
```

```
# cbind()/rbind() : combine vectors side-by-side

StringNumObjCbind ← cbind(NumVec, CharVec)
```

```
# Reassign row.names

row.names(StringNumObj) ← c("First", "Second", "Third")
```

Why use data.frame when we have cbind()/rbind()?

What is the difference between a matrix and a data frame?

$: component selection for data frames

```
NewNumeric ← StringNumObj$NumVec
```

# head()/tail() : select first/last few rows

```
data(mtcars)  #load built in cars dataset

head(mtcars)

tail(mtcars)
```

# [rows,columns] subscript operators, : sequence operator

```
mtcars[3:7, ]
```

# str()

compactly display the structure of an R object

#**what do you get when you apply str() to StringNumObj and StringNumObjCbind?**

# summary()

display summary statistics for analysis (mean, quantiles, etc)

# dim()

retrieve or set the dimensions of an object (array, matrix, dataframe)

Missing/extreme values:

NA = not available

NaN = undefined

Inf = extremely small/large (infinity)

A note on loading packages and functions

```
# load ggplot2

load(ggplot2)
```

```
#only load and use one function from ggplot2

ggplot2::qplot(. . .)
```

Conditional structures and loops

#if / if..else

```
If (test_express) {
    statement(s) if true
} else {
    statement(s) if false
}
```

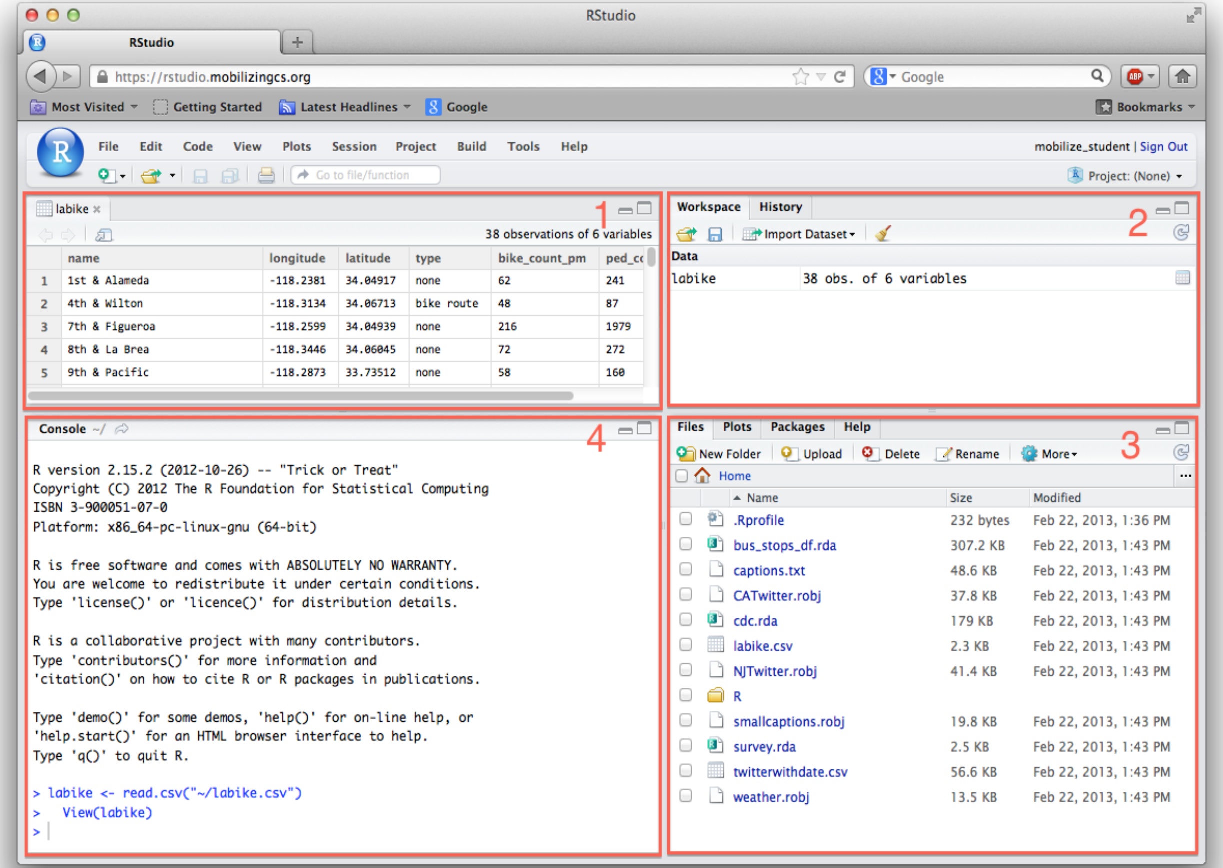
# Example:

```
speed ← 95
If (speed > 65) {
    print("Exceeding speed limit!")
} else {
    print(“Below speed limit")
}
```

```
# for loop

for (val in sequence)

{

statement(s)

}
```

```
# Example

for (month in 1:12)

{

print (month)

}
```

RStudio



38 observations of 6 variables

	name	longitude	latitude	type	bike_count_pm	ped_cc
1	1st & Alameda	-118.2381	34.04917	none	62	241
2	4th & Wilton	-118.3134	34.06713	bike route	48	87
3	7th & Figueroa	-118.2599	34.04939	none	216	1979
4	8th & La Brea	-118.3446	34.06045	none	72	272
5	9th & Pacific	-118.2873	33.73512	none	58	160

Data

labike	38 obs. of 6 variables

```
R version 2.15.2 (2012-10-26) -- "Trick or Treat"
Copyright (C) 2012 The R Foundation for Statistical Computing
ISBN 3-900051-07-0
Platform: x86_64-pc-linux-gnu (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.

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.

> labike <- read.csv("~/labike.csv")
>   View(labike)
> |
```

Name	Size	Modified
.Rprofile	232 bytes	Feb 22, 2013, 1:36 PM
bus_stops_df.rda	307.2 KB	Feb 22, 2013, 1:43 PM
captions.txt	48.6 KB	Feb 22, 2013, 1:43 PM
CATwitter.robj	37.8 KB	Feb 22, 2013, 1:43 PM
cdc.rda	179 KB	Feb 22, 2013, 1:43 PM
labike.csv	2.3 KB	Feb 22, 2013, 1:43 PM
NJTwitter.robj	41.4 KB	Feb 22, 2013, 1:43 PM
R		
smallcaptions.robj	19.8 KB	Feb 22, 2013, 1:43 PM
survey.rda	2.5 KB	Feb 22, 2013, 1:43 PM
twitterwithdate.csv	56.6 KB	Feb 22, 2013, 1:43 PM
weather.robj	13.5 KB	Feb 22, 2013, 1:43 PM

1: View Files & Data

2: See Workspace & History

3: Files, Plots Packages & Help

4: Console

Breakout Session!

Implement the following function in R:

# compute the factorial of a given number

computeFactorial(x)

Example: 6! = 6 X 5 X 4 X 3 X 2 X 1 = 720

There are no factorials of negative numbers

0! = 1

R in the cloud:
Coding ground: http://www.compileonline.com/execute_r_online.php
DataCamp https://cdn.datacamp.com/dcl/standalone-example.html previously R-fiddle: www.r-fiddle.org

One possible solution

```
computeFactorial <- function(x) {
  factorial = 1
  #check edge conditions: negative or zero
  if (x < 0) {
    print("Factorials cant be computed for negative numbers")
  } else if (x == 0) {
    print ("The factorial of 0 is 1")
  } else {
    for (i in 1:x){
      factorial = factorial * i
    }
    print (paste("The factorial of ", x, " is", factorial))
  }
}
```

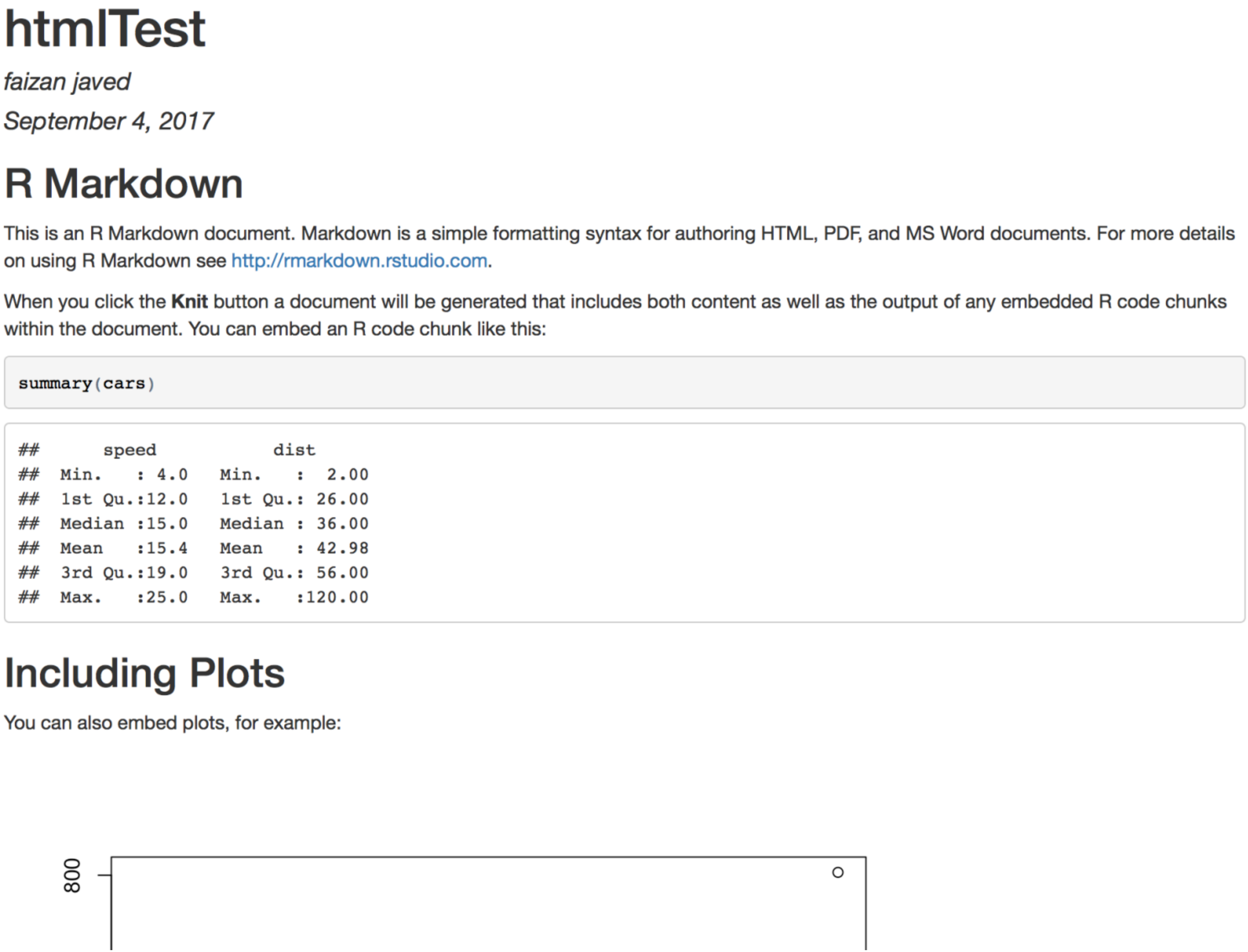
RMarkdown & knitr (http://rmarkdown.rstudio.com/articles_integration.html)

Create **static/interactive documents (code, description, results)**

RMarkdown uses knitr (markup) and Pandoc (document renderer)

There is also R Latex but we will mostly use R Markdown in this session.

See RMarkdown gallery for examples: http://rmarkdown.rstudio.com/gallery.html



htmlTest

faizan javed

September 4, 2017

R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see http://rmarkdown.rstudio.com.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```
summary(cars)
```

```
##      speed           dist       
##  Min.   : 4.0   Min.   :  2.00  
##  1st Qu.:12.0   1st Qu.: 26.00  
##  Median :15.0   Median : 36.00  
##  Mean   :15.4   Mean   : 42.98  
##  3rd Qu.:19.0   3rd Qu.: 56.00  
##  Max.   :25.0   Max.   :120.00
```

Including Plots

You can also embed plots, for example:

Global vs Local chunk options

Global:

~~~
```{r setup, include=FALSE}

knitr::opts_chunk$set(echo = TRUE)

```
~~~

Reads as: chunk label is “setup”, don’t output this code, by default output all other code chunks

Can set any chunk option as an argument to opts_chunk$set(..)

Local:

~~~
```{r pressure, echo=FALSE}
plot(pressure)
```
~~~

Reads as?

What did you learn today?

Questions?