MANGOSOLUTIONS
data analysis that delivers

Introduction to Shiny

LondonR Workshop

March 30th 2015

WiFi

- UCLGuest Wireless Network
- Navigate to a page outside of UCL
- Click on the link to the self service page
- Enter your details and event code (CDRC)
- Click Generate Account
- Make a note of the username and password
- Click on the link to login

Aimee Gott – R Consultant
agott@mango-solutions.com

Workshop Aim

Be able to develop a simple Shiny App with standard inputs and outputs

A subset of Mango's forthcoming Shiny training course!

Aimee Gott – R Consultant
agott@mango-solutions.com

Outline

- A Basic Shiny app
- Defining the User Interface
- Displaying Outputs
- Reactivity
- Beyond the Basics

Aimee Gott – R Consultant
agott@mango-solutions.com

Workshop resources

- R (version 3.1.2)
- RStudio
- Shiny (version 0.11)

Aimee Gott – R Consultant
agott@mango-solutions.com

Workshop structure

- 2 hours
- Presentation format
- Worked examples of creating apps
- Exercises during the workshop

Aimee Gott – R Consultant
agott@mango-solutions.com

What is Shiny?

- R Package for Interactive Web Apps developed by RStudio
- Gives the power of R in a convenient user interface
- Can be written entirely in R

Aimee Gott – R Consultant
agott@mango-solutions.com

A Basic Shiny App

- A basic app requires:
  - A user interface script
  - A "Server" script

- Runs using the `runApp` function

Aimee Gott – R Consultant
agott@mango-solutions.com

The User Interface Script

- Defines the components of the user interface
  - Page titles
  - Input options
  - Outputs

- Defines what the user will see and interact with

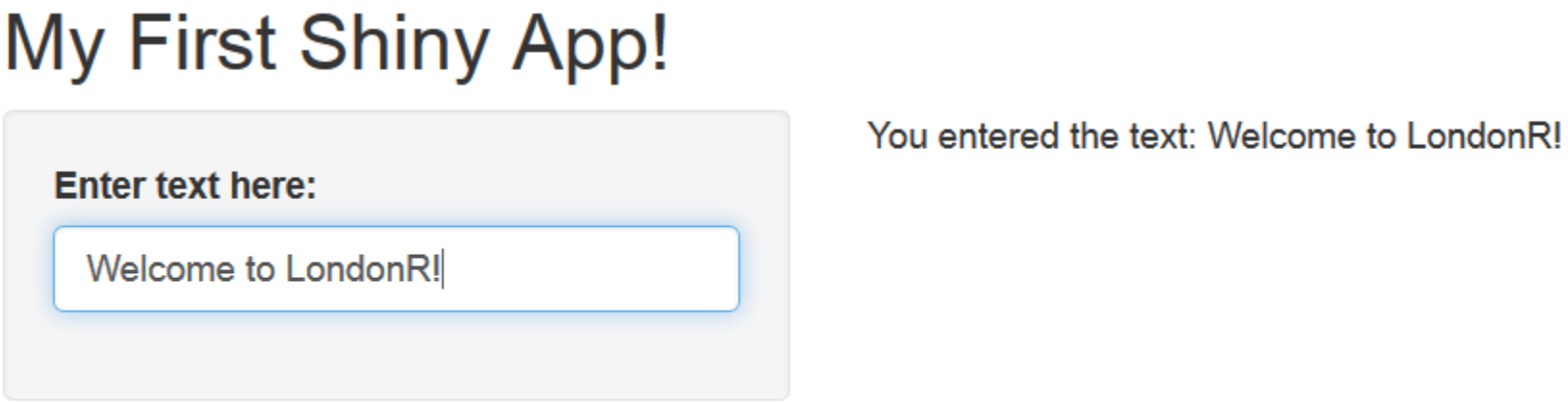
Aimee Gott – R Consultant
agott@mango-solutions.com

The Server Script

- Contains the information to build the app
- Requires a call to the function shinyServer
  - Contains a function with parrmeter input and output
- Defines what happens in R

Aimee Gott – R Consultant
agott@mango-solutions.com

Worked Example 1



Aimee Gott – R Consultant
agott@mango-solutions.com

Worked Example 1 - UI

```
shinyUI(fluidPage(
  titlePanel("My First Shiny App!"),
  sidebarLayout(
      sidebarPanel(
            textInput("myText", "Enter text here:")
       ),
      mainPanel(
                  textOutput("niceTextOutput")
            )
  )
))
```


Aimee Gott – R Consultant
agott@mango-solutions.com

Worked Example 1 - Server

```
shinyServer(function(input, output){

  output$niceTextOutput <-
renderText(paste("You entered the text:\n",
input$myText))

  })
```


Aimee Gott – R Consultant
agott@mango-solutions.com

Layouts

- Example 1 used a sidebarLayout
- There are a number of possible layouts
- In this workshop we will only use the sidebarLayout

Aimee Gott – R Consultant
agott@mango-solutions.com

Sidebar Panel

- Define the contents of the sidebar using the `sidebarPanel` function

- Accepts `*Input` functions that specify the app inputs

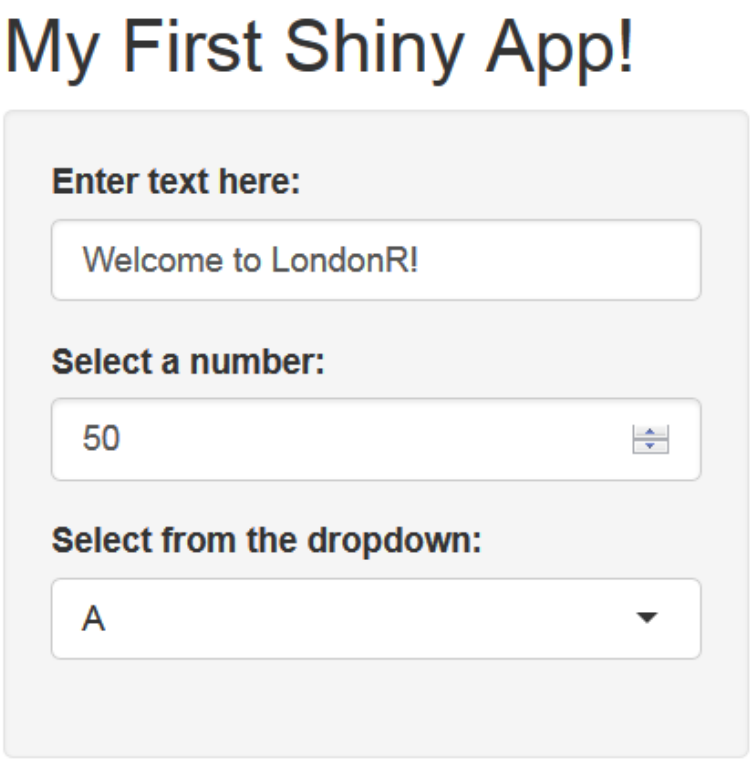
Aimee Gott – R Consultant
agott@mango-solutions.com

Input Controls

Input	Description
textInput	Text string input
numericInput	Numeric value input
selectInput	Select single or multiple values from drop down list
sliderInput	Numeric range “slider” input
radioButtons	Set of radio button inputs
fileInput	File upload control

Aimee Gott – R Consultant
agott@mango-solutions.com

Worked Example 2



Aimee Gott – R Consultant
agott@mango-solutions.com

Worked Example 2 - UI

```
sidebarPanel(
  textInput("myTextInput", "Enter text
here:"),
  numericInput("myNumberInput", "Select a
     number:", value = 50, min = 0, max =
     100, step = 1),
  selectInput("mySelectInput", "Select from
the  dropdown:", choices = LETTERS[1:10])
)
```


Aimee Gott – R Consultant
agott@mango-solutions.com

Main Panel

- Define the contents of the main panel using the function `mainPanel` function
- Can contain outputs using the `*Output` functions
- Can include HTML using a series of functions that replicate the HTML tags

Aimee Gott – R Consultant
agott@mango-solutions.com

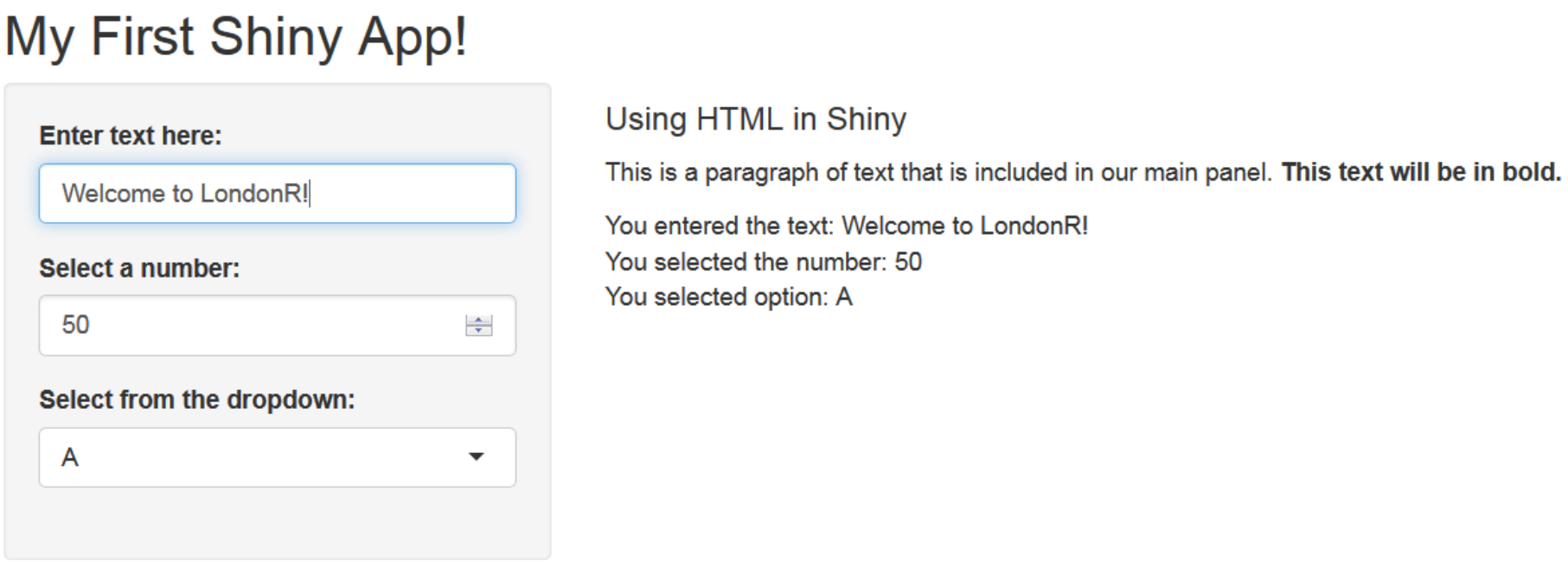
HTML Formatting

- We don't need to use HTML tags
- Shiny includes a series of equivalent functions

Function	Usage
p	A paragraph of text
h*	A level * (1, 2, 3,...) header
code	A block of code
img	An image
strong	Bold text
em	Italic text

Aimee Gott – R Consultant
agott@mango-solutions.com

Worked Example 2



Aimee Gott – R Consultant
agott@mango-solutions.com

Worked Example 2 - UI

```
mainPanel(
    h4("Using HTML in Shiny"),
    p("This is a paragraph of text that is
      included in our main panel.",
      strong("This text will be in bold.")),
       textOutput("niceTextOutput"),
       textOutput("niceNumberOutput"),
       textOutput("niceSelectOutput")
     )
```


Aimee Gott – R Consultant
agott@mango-solutions.com

Exercise 1

Build a simple Shiny application that takes a date string input (e.g. “30-03-2015”) and returns the following text:

- What day of the week is it (e.g. “Wednesday”)
- What month it is (e.g. “December”)
- What year it is

Hint: try using the dateInput and format functions

Aimee Gott – R Consultant
agott@mango-solutions.com

Exercise 1 - UI

```
require(shiny)

shinyUI(fluidPage(

  # Define the header for the page
  titlePanel("Exercise 1"),

  # Set up the page to have a sidebar
  sidebarLayout(
    # Define the contents of the sidebar
    sidebarPanel(
      dateInput("dateInput", "Select date")
    ),

    # Define the contents of the main panel
    mainPanel(
      textOutput("dateOutput")
    )
  )

))
```


Aimee Gott – R Consultant
agott@mango-solutions.com

Exercise 1 - Sever

```
require(shiny)

shinyServer(function(input, output){


  output$dateOutput <- renderText(
      format(input$dateInput, format = "A %A in %B.  The year is %Y")
    )

})
```


Aimee Gott – R Consultant
agott@mango-solutions.com

Defining Outputs

- So far we have just output text
- Shiny also allows us to output graphics, data and images
- We have to define the output in the UI and the Server scripts using different functions

Aimee Gott – R Consultant
agott@mango-solutions.com

Rendering Outputs

Output Type	server.R Function	ui.R Function
Text	renderPrint	textOutput
Data	renderDataTable	dataTableOutput
Plot	renderPlot	plotOutput
Image	renderImage	imageOutput

Aimee Gott – R Consultant
agott@mango-solutions.com

Worked Example 3 - Render Data

- From the user interface select a dataset from a dropdown menu
- Display the data in a dataTable

Aimee Gott – R Consultant
agott@mango-solutions.com

Worked Example 3 - Render Data

Render Data in a Shiny App

Select from the dropdown:

airquality

Show 25 entries Search:

Ozone	Solar.R	Wind	Temp	Month	Da
41	190	7.4	67	5	1
36	118	8	72	5	2
12	149	12.6	74	5	3
18	313	11.5	62	5	4
		14.3	56	5	5
28		14.9	66	5	6
23	299	8.6	65	5	7
19	99	13.8	59	5	8
8	19	20.1	61	5	9
	194	8.6	69	5	10
7		6.9	74	5	11

Worked Example 3 - UI

```
sidebarLayout(
  sidebarPanel(
      selectInput("selectInput", "Select from the
      dropdown:", choices = c("airquality", "iris",
      "mtcars"))
    ),
    mainPanel(
      dataTableOutput("dataOutput")
    )
  )
```


Aimee Gott – R Consultant
agott@mango-solutions.com

Worked Example 3 - Server

```
output$dataOutput <-
  renderDataTable(switch(input$selectInput,
     "airquality" = airquality,
     "iris" = iris,
     "mtcars" = mtcars)
  )
```

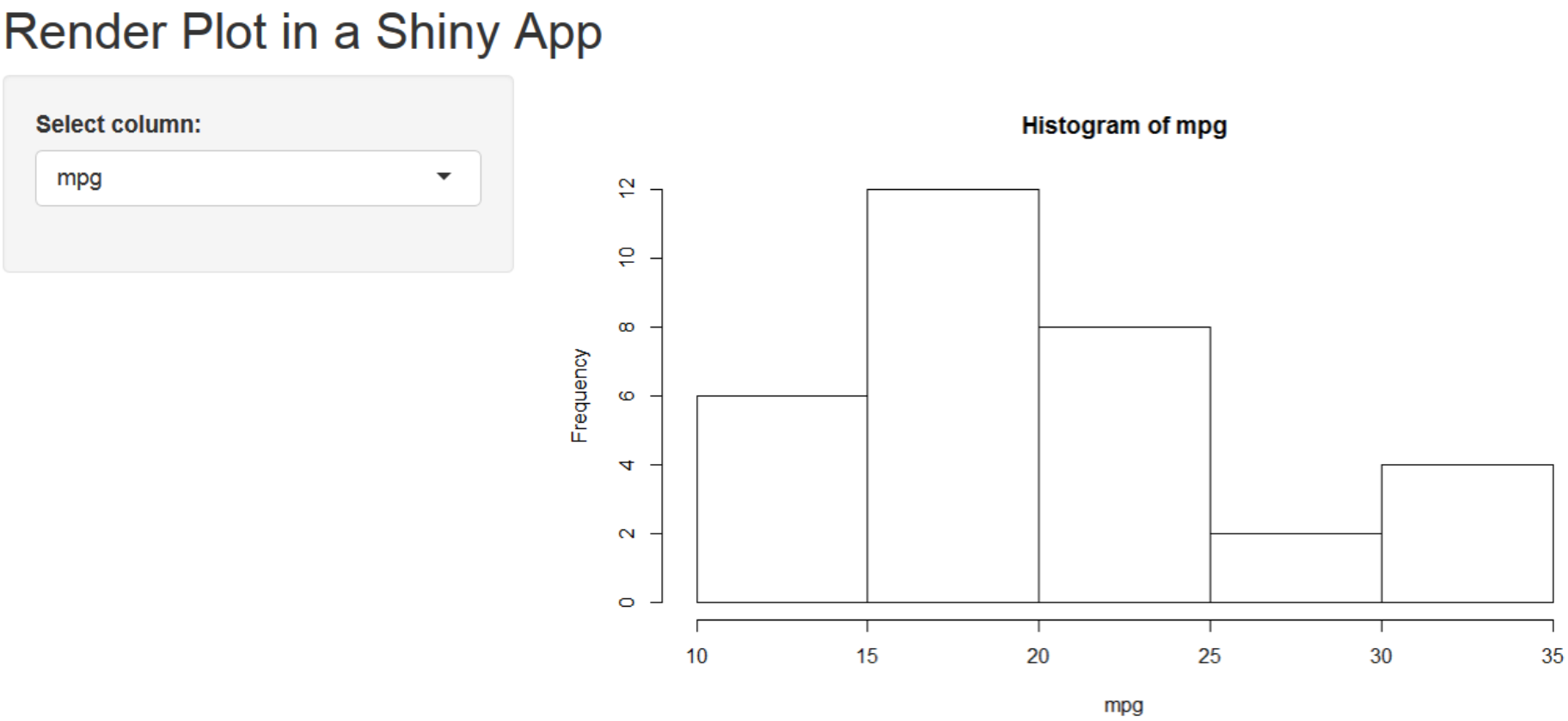

Aimee Gott – R Consultant
agott@mango-solutions.com

Worked Example 4 - Render Plots

- Select a column of the data from a drop down menu
- Plot a histogram of the data

Aimee Gott – R Consultant
agott@mango-solutions.com

Worked Example 4 - Render Plots



Aimee Gott – R Consultant
agott@mango-solutions.com

Worked Example 4 - UI

```
sidebarLayout(
  sidebarPanel(
     selectInput("selectInput", "Select
column:", choices = colnames(mtcars))
    ),
  mainPanel(
     plotOutput("plotOutput")
    )
  )
```


Aimee Gott – R Consultant
agott@mango-solutions.com

Worked Example 4 - Server

```
output$plotOutput <- renderPlot(
      hist(mtcars[,input$selectInput],
           main = paste("Histogram
           of", input$selectInput),
           xlab = input$selectInput)
    )
```


Aimee Gott – R Consultant
agott@mango-solutions.com

Exercise 2

Create a Shiny application that takes:

- A numeric value between 1 and 500
- A colour
- A main title

Use these inputs to create an output histogram of random data from any distribution where n is the numeric input

Aimee Gott – R Consultant
agott@mango-solutions.com

Exercise 2 - UI

```
require(shiny)

shinyUI(fluidPage(

  # Define the header for the page
  titlePanel("Render Plot in a Shiny App"),

  # Set up the page to have a sidebar
  sidebarLayout(
    # Define the contents of the sidebar
    sidebarPanel(
      numericInput("numberInput", "Select size of data:", min = 0, max = 500, value = 100),
      selectInput("colInput", "Select a colour", choices = c("red", "yellow", "blue", "green"))
    ),

    # Define the contents of the main panel
    mainPanel(
      plotOutput("plotOutput")
    )
  )

))
```


Aimee Gott – R Consultant
agott@mango-solutions.com

Exercise 2 - Server

```
require(shiny)

shinyServer(function(input, output){


  output$plotOutput <- renderPlot(
    hist(rnorm(input$numberInput), col = input$colInput)
  )

})
```


Aimee Gott – R Consultant
agott@mango-solutions.com

Reactivity

- Consider the last exercise...
  - Suppose we want to change the colour of the plot, what happens to the data?

Aimee Gott – R Consultant
agott@mango-solutions.com

Reactivity

- Each time we change an option the data is simulated again
- Suppose this was reading in a large dataset, connecting to a database etc.

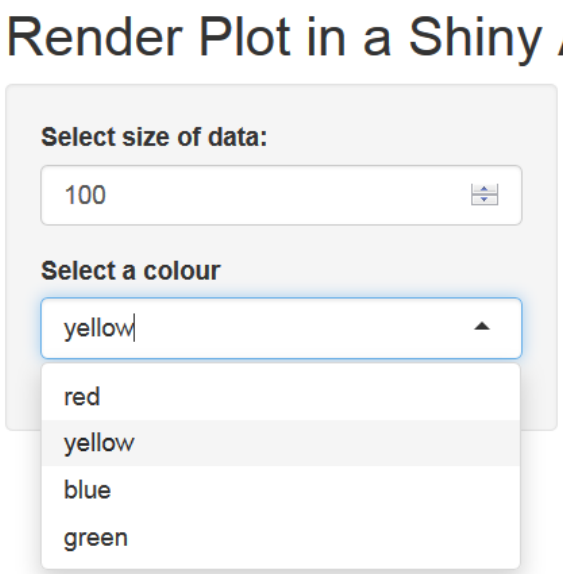
Aimee Gott – R Consultant
agott@mango-solutions.com

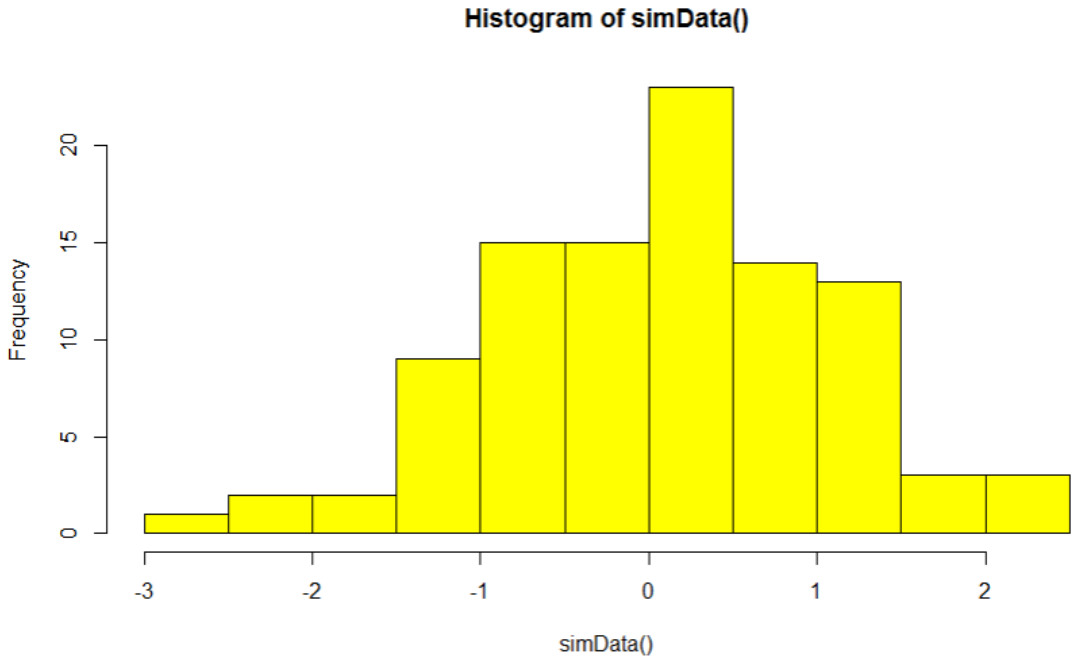
The `reactive` Function

- This lets us create a reactive function
- The function is only called when the relevant inputs are updated
- Our data is only updated when the number of simulations is changed

Aimee Gott – R Consultant
agott@mango-solutions.com

Worked Example 5





Aimee Gott – R Consultant
agott@mango-solutions.com

Worked Example 5 - Server

```
simData <- reactive({
      rnorm(input$numberInput)
    })

  output$plotOutput <- renderPlot(
    hist(simData(), col =
                      input$colInput)
  )
```


Aimee Gott – R Consultant
agott@mango-solutions.com

Beyond the basics

- Changes to layouts
- Including tabbed pages
- Include CSS to style the page
- Incorporate Shiny and Markdown
- Share your app
- ...

All covered on our 1 day Getting Started with Shiny course

Aimee Gott – R Consultant
agott@mango-solutions.com

Shiny Themes

- A new package that allows us to change the bootstrap theme
- Requires Shiny v0.11
- Available on CRAN

http://rstudio.github.io/shinythemes/

Aimee Gott – R Consultant
agott@mango-solutions.com

Worked Example 6

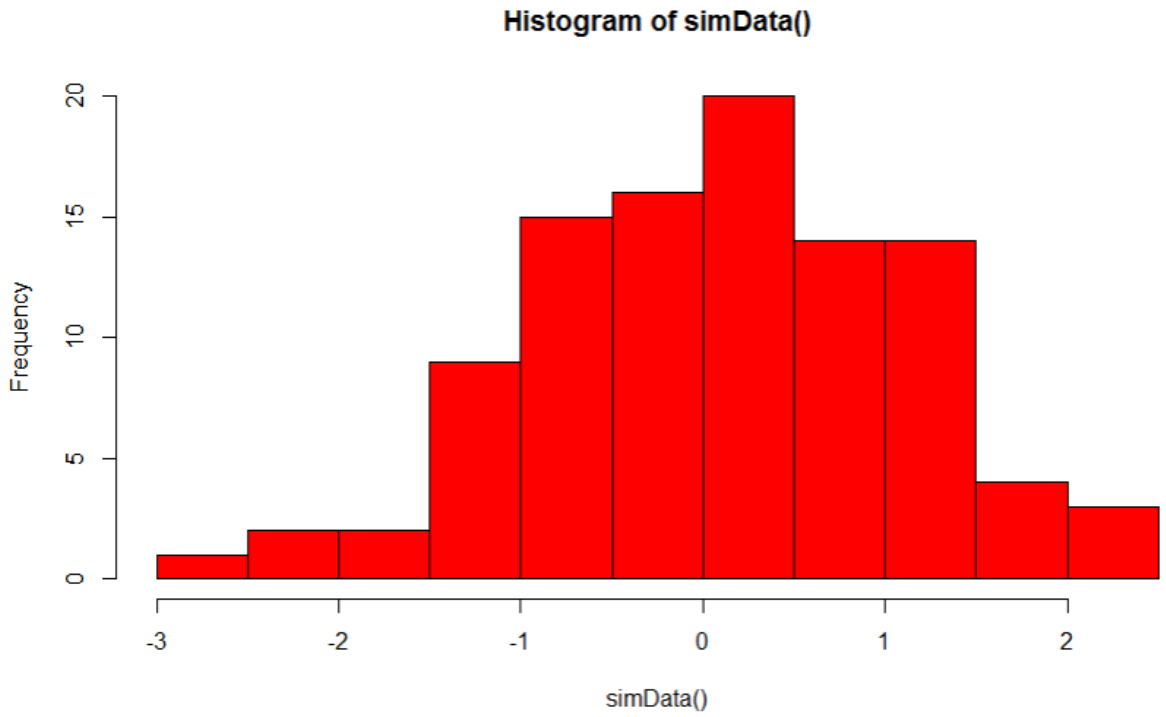
Render Plot in a Shiny App

Select size of data:

100

Select a colour

red



Aimee Gott – R Consultant
agott@mango-solutions.com

Worked Example 6 - UI

```
require(shinythemes)

shinyUI(fluidPage(
          theme = shinytheme("cerulean"),
...
))
```


Aimee Gott – R Consultant
agott@mango-solutions.com

Shiny Dashboard

- Package under development by RStudio for developing Dashboards with Shiny

- Available on github

Aimee Gott – R Consultant
agott@mango-solutions.com

What Next?

- This evenings LondonR meeting!

- Getting Started with Shiny training course
- Data Visualisation in R training course

- EARL 2015

Aimee Gott – R Consultant
agott@mango-solutions.com

LondonR tonight

- **SAS to R Migration** – Rich Pugh, Mango Solutions
- **The Coder and the Designer: Using R for Visualising London's Data** – James Cheshire, UCL (and Co-Author of The Information Capital)
- **How to build a mid-sized analytical application with Shiny** – Enzo Martoglio, Sopra Steria

Aimee Gott – R Consultant
agott@mango-solutions.com

Getting Started with Shiny - May 27th

- 1 Day Training Course
  - Covers this content in more depth + more!
    - Including CSS
    - Shiny + Rmarkdown
- To be held in central London
- See the Mango team for more details!

Aimee Gott – R Consultant
agott@mango-solutions.com

Data Visualisation in R - May 28th

- 1 Day Training Course
  - The Principles of Data Visualisation
  - Implementing these principles in ggplot2
- To be held in central London
- See the Mango team for more details!

Aimee Gott – R Consultant
agott@mango-solutions.com

EARL 2015

LONDON 14 - 16 SEPTEMBER

- Tower Hotel
- 2 day conference
- 1 day of workshops prior to conference
- Capacity for 400+
- Conference evening event in Tower Bridge
- Earlybird deadline ends tomorrow
- Abstract deadline extended to 17th April

Aimee Gott – R Consultant
agott@mango-solutions.com

EARL 2015

LONDON 14 - 16 SEPTEMBER

- **Alex Bellos** (Author of Alex's Adventures in Numberland)
- **Joe Cheng** (Creator of Shiny!)
- **Dirk Eddelbuettel** (Author of Rcpp)
- **Hannah Fry** (Lecturer in Mathematics at the Centre for Advanced Spatial Analysis)

Speak to the Mango team to know more or visit the webpage

http://www.earl-conference.com/

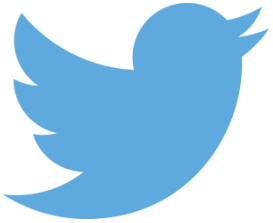
Aimee Gott – R Consultant
agott@mango-solutions.com

EARL Workshops

- 10.00 – 13.00
  - **Integrating R and Python** (Chris Musselle)
  - **Current Best Practices in Formal Package Development** (Aimee Gott, Gregoire Gauriot)
- 14.00 – 17.00
  - **Introduction to Rcpp** (Dirk Eddelbuettel)
  - **Interactive reporting with R Markdown and Shiny** (Joe Cheng)

Aimee Gott – R Consultant
agott@mango-solutions.com

Follow Mango!



@mangothecat

@earlconf

MangoTheCat

Aimee Gott – R Consultant
agott@mango-solutions.com