Combining shiny and RMarkdown

Anthony Davidson from Philip Bowsher (**???**)

2020-02-09

# Prerequisites

This book is build off the *sample* book written in **Markdown**. You can use anything that Pandoc’s Markdown supports, e.g., a math equation .

The **bookdown** package can be installed from CRAN or Github:

# install.packages("bookdown")  
# or the development version  
# devtools::install\_github("rstudio/bookdown")  
library(bookdown)  
library(shiny)

Remember each Rmd file contains one and only one chapter, and a chapter is defined by the first-level heading #.

To compile this example to PDF, you need XeLaTeX. You are recommended to install TinyTeX (which includes XeLaTeX): <https://yihui.name/tinytex/>.

## Additional notes

… check out the website for more resources [here](https://www.ssnhub.com)

# Introduction

R is a powerful programming language for statistical computing with many packages and tools.

The goal of this article is to outline some ways to combine Shiny & R Markdown.

We’ll cover the main approaches below:

1. **Interactive documents**
2. **Including Markdown content in a Shiny app**
3. **Generating downloadable reports from Shiny apps**
4. **Embedding Shiny apps via their URLs in R Markdown output**
5. **Shiny components in rtutor**

## Generating downloadable reports from Shiny app

* It is often required after interacting with a Shiny application to download a report in HTML or PDF format. This can be done with knitr and rmarkdown. The app takes one input value, and passes it as a parameter to an .Rmd document, which is rendered (or knitted) into an HTML output file.
* <https://shiny.rstudio.com/articles/generating-reports.html>
* <http://shiny.rstudio.com/gallery/download-knitr-reports.html>
* Another way to generate a PDF is to pass in a value to output\_format argument of rmarkdown::render(). It will also require pdflatex to be installed on your system.
* <http://shiny.rstudio.com/gallery/download-knitr-reports.html>

## Generating downloadable reports from Shiny app - Example

* <https://github.com/aaronjfisher/interAdapt/tree/master/r_package/inst/interAdapt>
* <https://michaelrosenblum.shinyapps.io/interAdapt/>

# Literature

Here is a review of existing methods. To begin with I have simply forked the rbooks repo and started to add to this along with the shiny library:

R Books [![Awesome](data:image/svg+xml;charset=utf-8;base64,)](https://github.com/sindresorhus/awesome)

A curated list of books about different aspects and applications of [R](https://www.r-project.org/) programming language and an attempt to modernize R Books in a more user friendly way. Inspired by GoBooks.

* [R Books](#r-books)
  + [Beginner Books](#beginner-books)
  + [Advanced Books](#advanced-books)
  + [Data Science](#data-science)
  + [Finance](#finance)
  + [Machine Learning](#machine-learning)
  + [R Development](#r-development)
  + [Reports](#reports)
  + [Visualization](#visualization)

## Beginner Books

#### Learning R [[Amazon]](www.amazon.com/Learning-R-Richard-Cotton/dp/1449357105/)

Learn how to perform data analysis with the R language and software environment, even if you have little or no programming experience. With the tutorials in this hands-on guide, you’ll learn how to use the essential R tools you need to know to analyze data, including data types and programming concepts.

#### Hands-On Programming with R [[Amazon]](http://www.amazon.com/Hands-On-Programming-Write-Functions-Simulations/dp/1449359019)

Learn how to program by diving into the R language, and then use your newfound skills to solve practical data science problems. With this book, you’ ll learn how to load data, assemble and disassemble data objects, navigate R’s environment system, write your own functions, and use all of R’s programming tools.

#### The R Book [[Amazon]](http://www.amazon.com/The-Book-Michael-J-Crawley/dp/0470973927)

Very massive, hugely successful and popular text presenting an extensive and comprehensive guide for all R users. Covers wide variaty of R topics in great detail with very useful examples.

#### R for Everyone: Advanced Analytics and Graphics [[Amazon]](http://www.amazon.com/Everyone-Advanced-Analytics-Graphics-Addison-Wesley/dp/0321888030)

Drawing on his unsurpassed experience teaching new users, professional data scientist Jared P. Lander has written the perfect tutorial for anyone new to statistical programming and modeling. Organized to make learning easy and intuitive, this guide focuses on the 20 percent of R functionality you’ll need to accomplish 80 percent of modern data tasks. Book’s self-contained chapters start with the absolute basics, offering extensive hands-on practice and sample code. You’ll download and install R; navigate and use the R environment; master basic program control, data import, and manipulation; and walk through several essential tests. Then, building on this foundation, you’ll construct several complete models, both linear and nonlinear, and use some data mining techniques.

#### R Cookbook [[Amazon]](http://www.amazon.com/Cookbook-OReilly-Cookbooks-Paul-Teetor/dp/0596809158/)

With more than 200 practical recipes, this book helps you perform data analysis with R quickly and efficiently. The R language provides everything you need to do statistical work, but its structure can be difficult to master. This collection of concise, task-oriented recipes makes you productive with R immediately, with solutions ranging from basic tasks to input and output, general statistics, graphics, and linear regression.

#### Learn R in a Day [[Amazon]](http://www.amazon.com/Learn-R-Day-Steven-Murray-ebook/)

‘Learn R in a Day’ provides the reader with key programming skills through an examples-oriented approach and is ideally suited for academics, scientists, mathematicians and engineers. The book assumes no prior knowledge of computer programming and progressively covers all the essential steps needed to become confident and proficient in using R within a day. Topics include how to input, manipulate, format, iterate (loop), query, perform basic statistics on, and plot data, via a step-by-step technique and demonstrations using in-built datasets which the reader is encouraged to replicate on their computer. Each chapter also includes exercises (with solutions) to practice key skills and empower the reader to build on the essentials gained during this introductory course.

#### R Inferno [[Online]](http://www.burns-stat.com/pages/Tutor/R_inferno.pdf) *Free*

“If you are using R and you think you’re in hell, this is a map for you.”

This is a great introduction to several aspects of R programming. Loading data into R, iterating over data with loops & logic statements, author even touches on creating packages. Nice and pretty thorough book on R.

#### The R Programming Wikiboo [[online]](https://en.wikibooks.org/wiki/R_Programming)

A collaborative book about R on the wikibooks website. Anyone can improve it.

#### Data Munging with R [[Manning]](https://www.manning.com/books/data-munging-with-r)

Data Munging with R shows you how to take raw data and transform it for use in computations, tables, graphs, and more. Whether you already have some programming experience or you’re just a spreadsheet whiz looking for a more powerful data manipulation tool, this book will help you get started.

You’ll discover the ins and outs of using the data-oriented R programming language and its many task-specific packages. With dozens of practical examples to follow, learn to fill in missing values, make predictions, and visualize data as graphs. By the time you’re done, you’ll be a master munger, with a robust, reproducible workflow and the skills to use data to strengthen your conclusions!

#### R for Data Science [[Online]](http://r4ds.had.co.nz/) *Free*

Learn how to use R to turn raw data into insight, knowledge, and understanding. This book introduces you to R, RStudio, and the [*tidyverse*](https://www.tidyverse.org/), a collection of R packages designed to work together to make data science fast, fluent, and fun. Suitable for readers with no previous programming experience, R for Data Science is designed to get you doing data science as quickly as possible.

## Advanced Books

#### Advanced R [[Online]](http://adv-r.had.co.nz/) [[Amazon]](http://www.amazon.com/dp/1466586966/) *Free*

Advanced R presents useful tools and techniques for attacking many types of R programming problems, helping you avoid mistakes and dead ends. With more than ten years of experience programming in R, the author illustrates the elegance, beauty, and flexibility at the heart of R.

#### Data Manipulation with R, Second Edition [[Packt]](https://www.packtpub.com/big-data-and-business-intelligence/data-manipulation-r-second-edition)

This book starts with the installation of R and how to go about using R and its libraries. We then discuss the mode of R objects and its classes and then highlight different R data types with their basic operations.

The primary focus on group-wise data manipulation with the split-apply-combine strategy has been explained with specific examples. The book also contains coverage of some specific libraries such as lubridate, reshape2, plyr, dplyr, stringr, and sqldf. You will not only learn about group-wise data manipulation, but also learn how to efficiently handle date, string, and factor variables along with different layouts of datasets using the reshape2 package.

By the end of this book, you will have learned about text manipulation using stringr, how to extract data from twitter using twitteR library, how to clean raw data, and how to structure your raw data for data mining.

## Data Science

#### Mastering Data Science with R [[Packt]](https://www.packtpub.com/big-data-and-business-intelligence/mastering-data-analysis-r)

This book will give you the guidance you need to build and develop your knowledge and expertise. Bridging the gap between theory and practice, this book will help you to understand and use data for a competitive advantage.

Beginning with taking you through essential data mining and management tasks such as munging, fetching, cleaning, and restructuring, the book then explores different model designs and the core components of effective analysis. You will then discover how to optimize your use of machine learning algorithms for classification and recommendation systems beside the traditional and more recent statistical methods.

#### Practical Data Science with R [[Amazon]](http://www.amazon.com/Practical-Data-Science-Nina-Zumel/dp/1617291560)

Practical Data Science with R shows you how to apply the R programming language and useful statistical techniques to everyday business situations. Using examples from marketing, business intelligence, and decision support, it shows you how to design experiments (such as A/B tests), build predictive models, and present results to audiences of all levels.

This book is accessible to readers without a background in data science. Some familiarity with basic statistics, R, or another scripting language is assumed.

#### R in Action: Data Analysis and Graphics with R [[Amazon]](http://www.amazon.com/gp/product/1617291382)

With more than 200 practical recipes, this book helps you perform data analysis with R quickly and efficiently. The R language provides everything you need to do statistical work, but its structure can be difficult to master. This collection of concise, task-oriented recipes makes you productive with R immediately, with solutions ranging from basic tasks to input and output, general statistics, graphics, and linear regression.

#### R Programming for Data Science [[Leanpub]](https://leanpub.com/rprogramming) *Free*

R in Action, Second Edition teaches you how to use the R language by presenting examples relevant to scientific, technical, and business developers. Focusing on practical solutions, the book offers a crash course in statistics, including elegant methods for dealing with messy and incomplete data. You’ll also master R’s extensive graphical capabilities for exploring and presenting data visually. And this expanded second edition includes new chapters on forecasting, data mining, and dynamic report writing.

#### Exploratory Data Analysis with R [[Leanpub]](https://leanpub.com/exdata) *Free*

This book teaches you to use R to effectively visualize and explore complex datasets. Exploratory data analysis is a key part of the data science process because it allows you to sharpen your question and refine your modeling strategies. This book is based on the industry-leading Johns Hopkins Data Science Specialization, the most widely subscribed data science training program ever created.

## Finance

#### Introduction to R for Quantitative Finance [[Packt]](https://www.packtpub.com/big-data-and-business-intelligence/introduction-r-quantitative-finance)

Introduction to R for Quantitative Finance will show you how to solve real-world quantitative finance problems using the statistical computing language R. The book covers diverse topics ranging from time series analysis to fi nancial networks. Each chapter briefl y presents the theory behind specific concepts and deals with solving a diverse range of problems using R with the help of practical examples.

This book will be your guide on how to use and master R in order to solve quantitative finance problems. This book covers the essentials of quantitative finance, taking you through a number of clear and practical examples in R that will not only help you to understand the theory, but how to effectively deal with your own real-life problems.

Starting with time series analysis, you will also learn how to optimize portfolios and how asset pricing models work. The book then covers fixed income securities and derivatives such as credit risk management.

#### Mastering R for Quantitative Finance [[Packt]](https://www.packtpub.com/big-data-and-business-intelligence/mastering-r-quantitative-finance)

R is a powerful open source functional programming language that provides high level graphics and interfaces to other languages. Its strength lies in data analysis, graphics, visualization, and data manipulation. R is becoming a widely used modeling tool in science, engineering, and business.

The book is organized as a step-by-step practical guide to using R. Starting with time series analysis, you will also learn how to forecast the volume for VWAP Trading. Among other topics, the book covers FX derivatives, interest rate derivatives, and optimal hedging. The last chapters provide an overview on liquidity risk management, risk measures, and more.

The book pragmatically introduces both the quantitative finance concepts and their modeling in R, enabling you to build a tailor-made trading system on your own. By the end of the book, you will be well versed with various financial techniques using R and will be able to place good bets while making financial decisions.

## Machine Learning

#### Machine Learning with R [[Amazon]](http://www.amazon.com/Machine-Learning-R-Brett-Lantz/dp/1782162143) [[O’Reilly]](http://shop.oreilly.com/product/9781782162148.do)

“Machine Learning with R” is a practical tutorial that uses hands-on examples to step through real-world application of machine learning. Without shying away from the technical details, we will explore Machine Learning with R using clear and practical examples. Well-suited to machine learning beginners or those with experience. Explore R to find the answer to all of your questions.

## R Development

#### R Packages [[Online]](http://r-pkgs.had.co.nz/) [[Amazon]](http://www.amazon.com/dp/1491910593/)

Packages are the fundamental units of reproducible R code. They include reusable R functions, the documentation that describes how to use them, and sample data. In this section you’ll learn how to turn your code into packages that others can easily download and use. Writing a package can seem overwhelming at first. So start with the basics and improve it over time. It doesn’t matter if your first version isn’t perfect as long as the next version is better.

## Reports

#### Dynamic Documents with R and knitr, Second Edition [[Amazon]](http://www.amazon.com/dp/1498716962/)

Suitable for both beginners and advanced users, Dynamic Documents with R and knitr, Second Edition makes writing statistical reports easier by integrating computing directly with reporting. Reports range from homework, projects, exams, books, blogs, and web pages to virtually any documents related to statistical graphics, computing, and data analysis. The book covers basic applications for beginners while guiding power users in understanding the extensibility of the knitr package.

#### Report Writing for Data Science in R [[Leanpub]](https://leanpub.com/reportwriting) *Free*

This book teaches the fundamental concepts and tools behind reporting modern data analyses in a reproducible manner. As data analyses become increasingly complex, the need for clear and reproducible report writing is greater than ever. The material for this book was developed as part of the industry-leading Johns Hopkins Data Science Specialization.

## Visualization

#### R Data Visualization Cookbook [[Amazon]](http://www.amazon.com/dp/1466586966/)

This book is packed with practical recipes, designed to provide you with all the guidance needed to get to grips with data visualization with R. It starts off with the basics of R plots and an introduction to heat maps and customizing them, before gradually taking you through creating interactive maps using the googleVis package, generating choropleth maps and contouring maps, bubble plots, and pie charts. You will then learn how to animate 2D and 3D plots in R. By the end of the book, you will be equipped with the key techniques to create impressive data visualizations with professional efficiency and precision.

#### R Graphics Cookbook [[Amazon]](http://www.amazon.com/R-Graphics-Cookbook-Winston-Chang/dp/1449316956)

This practical guide provides more than 150 recipes to help you generate high-quality graphs quickly, without having to comb through all the details of R’s graphing systems. Each recipe tackles a specific problem with a solution you can apply to your own project, and includes a discussion of how and why the recipe works.

Most of the recipes use the ggplot2 package, a powerful and flexible way to make graphs in R. If you have a basic understanding of the R language, you’re ready to get started.

# Methods

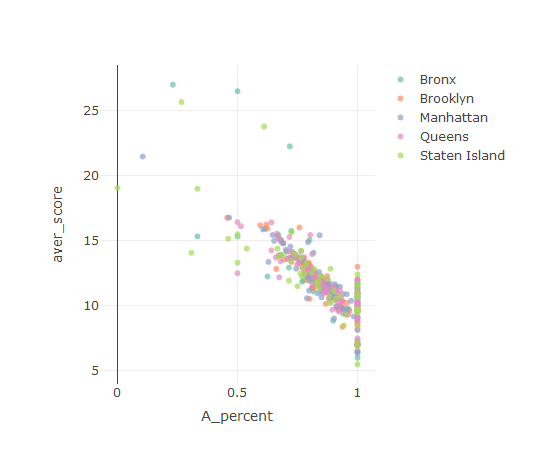
## Flashboards

Flexdashboards in RMarkdown

#read and clean data  
nyc\_inspections <-   
 read\_csv("./data/DOHMH\_New\_York\_City\_Restaurant\_Inspection\_Results.csv.gz",   
 col\_types = cols(building = col\_character()),  
 na = c("NA", "N/A")) %>%  
 clean\_names() %>%  
 filter(grade %in% c("A", "B", "C"),  
 boro != "Missing") %>%   
 mutate(inspection\_num = row\_number(),  
 boro = str\_to\_title(boro)) %>%   
 select(inspection\_num, boro, grade, score, critical\_flag, cuisine\_description)

### Grade A Percentage vs Average Score ScatterPlot

colourCount = length(unique(nyc\_inspections$boro))  
getPalette = colorRampPalette(brewer.pal(9, "Set2"))  
  
nyc\_inspections %>%  
 group\_by(boro, cuisine\_description) %>%  
 summarise(aver\_score = mean(score),  
 num\_inspections = n(),  
 A\_num = sum(grade == "A"),  
 A\_percent = sum(grade == "A")/n(),  
 critaical\_percent = sum(grade == "A")/n()) %>%  
 mutate(text\_label = str\_c("Boro: ", boro, '\nCuisine: ', cuisine\_description,   
 '\nnumber of inspection:', num\_inspections, '\nnumber of A:', A\_num,  
 '\n A%:', paste(round(A\_percent\*100,digits = 2),"%",sep = "")),   
 '\n critical%:', paste(round(critaical\_percent\*100,digits = 2),"%",sep = "")) %>%  
 plot\_ly(x = ~A\_percent, y = ~aver\_score, type = "scatter", mode = "markers",   
 alpha = 0.7, color = ~boro, text = ~text\_label)

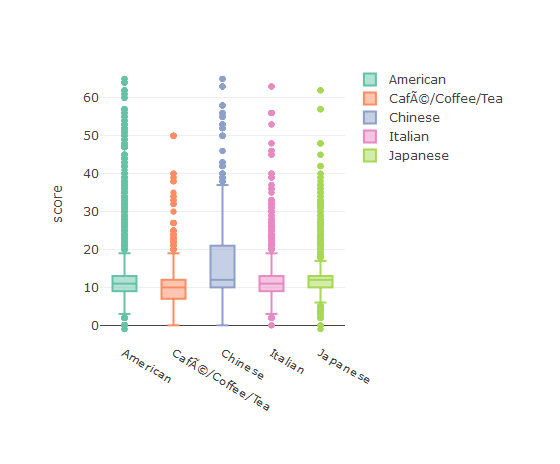


### Boxplot of Scores

colourCount = length(unique(nyc\_inspections$boro))  
getPalette = colorRampPalette(brewer.pal(9, "Set2"))

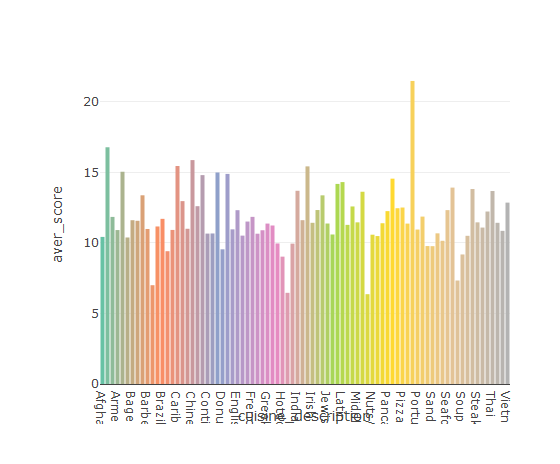
## Warning in brewer.pal(9, "Set2"): n too large, allowed maximum for palette Set2 is 8  
## Returning the palette you asked for with that many colors

cuisine\_boxplot <-  
 nyc\_inspections %>%  
 filter(boro == "Manhattan") %>%  
 group\_by(cuisine\_description) %>%  
 summarise(num\_inspections = n()) %>%  
 top\_n(5, num\_inspections) %>%  
 pull(cuisine\_description)  
  
nyc\_inspections %>%  
 filter(boro == "Manhattan",  
 cuisine\_description %in% cuisine\_boxplot) %>%  
 plot\_ly(y = ~score, color = ~cuisine\_description, type = "box", colors = "Set2")



### Bar-plot for Average Scores by Cuisine

nyc\_inspections %>%  
 group\_by(boro, cuisine\_description) %>%  
 summarise(aver\_score = mean(score),  
 num\_inspections = n(),  
 A\_num = sum(grade == "A"),  
 A\_percent = sum(grade == "A")/n(),  
 critaical\_percent = sum(grade == "A")/n()) %>%  
 mutate(text\_label = str\_c("Boro: ", boro, '\nCuisine: ', cuisine\_description,   
 '\nnumber of inspection:', num\_inspections, '\nnumber of A:', A\_num,  
 '\n A%:', paste(round(A\_percent\*100, digits = 2),"%",sep = "")),   
 '\n critical%:', paste(round(critaical\_percent\*100, digits = 2),"%",sep = "")) %>%  
 filter(boro == "Manhattan") %>%  
 plot\_ly(x = ~cuisine\_description, y = ~aver\_score, color = ~cuisine\_description, type = "bar", text = ~text\_label) %>%  
 layout(showlegend = FALSE)



### Trip planning html-widget

<https://beta.rstudioconnect.com/jjallaire/htmlwidgets-showcase-storyboard/htmlwidgets-showcase-storyboard.html>

<https://rstudio.github.io/leaflet/>

## Shiny

We have briefly introduced Shiny documents in Section @ref(intro-shiny). Shiny is a very powerful framework for building web applications based on R. It is out of the scope of this book to make a comprehensive introduction to Shiny (which is too big a topic). We recommend that readers who are not familiar with Shiny learn more about it from the website <https://shiny.rstudio.com> before reading this chapter.

Unlike the more traditional workflow of creating static reports, you can create documents that allow your readers to change the parameters underlying your analysis and see the results immediately in Shiny R Markdown documents. In the example shown in Figure @ref(fig:shiny), the histogram will be automatically updated to reflect the number of bins selected by the reader.

A picture is worth a thousand words, and a Shiny document can potentially show you a thousand pictures as you interact with it. The readers are no longer tied to the fixed analysis and conclusions in the report. They may explore other possibilities by themselves, and possibly make new discoveries or draw different conclusions.

### Getting started

You can turn any *HTML-oriented* R Markdown documents to Shiny documents by adding runtime: shiny to the YAML metadata as a *top-level* field, e.g.,

---  
title: "Shiny Document"  
output: html\_document  
runtime: shiny  
---

Note that the output format of the R Markdown document must be an HTML format. That is, the document should generate a web page (a \*.html file). Non-HTML formats such as pdf\_document and word\_document will not work with the Shiny runtime. Please also note that some presentation formats are also HTML formats, such as ioslides\_presentation and slidy\_presentation.

You can also create a new Shiny document from the RStudio menu File -> New File -> R Markdown, and choose the document type “Shiny” (see Figure @ref(fig:shiny-new-document)).

Create a new Shiny document in RStudio.

Create a new Shiny document in RStudio.

To run a Shiny document in RStudio, you need to click the button “Run Document” on the editor toolbar (RStudio will automatically replace the “Knit” button with “Run Document” when it detects a Shiny document). If you do not use RStudio, or want to run the document in the R console for troubleshooting, you can call the function rmarkdown::run() and pass the filename to it.

You can embed Shiny inputs and outputs in your document. Outputs are automatically updated whenever inputs change. In the following example, we create a numeric input (numericInput) with the name rows, and then refer to its value via input$rows when generating output:

```{r, echo=FALSE}  
numericInput("rows", "How many cars?", 5)  
  
renderTable({  
 head(cars, input$rows)  
})  
```

Increase the number of rows in the table in a Shiny document.

Increase the number of rows in the table in a Shiny document.

In the above example, the output code was wrapped in a call to renderTable(). There are many other render functions in Shiny that can be used for plots, printed R output, and more. This example uses renderPlot() to create dynamic plot output:

```{r, echo=FALSE}  
sliderInput("bins", "Number of bins:", 30, min = 1, max = 50)  
  
renderPlot({  
 x = faithful[, 2] # Old Faithful Geyser data  
 bins = seq(min(x), max(x), length.out = input$bins + 1)  
  
 # draw the histogram with the specified number of bins  
 hist(x, breaks = bins, col = 'darkgray', border = 'white')  
})  
```

Change the number of bins of a histogram in a Shiny document.

Change the number of bins of a histogram in a Shiny document.

## Deployment

Shiny documents must be supported by a live R session running behind the scenes. When you run a Shiny document locally, it uses your local R session. Usually only you can see and interact with the document. If you want to share the document with people who do not have R installed, or do not want to run the document locally, you will have to deploy the document on a server, and share the URL of the document. Then other people only need a web browser to visit your document.

There are two ways to deploy a Shiny document. You can either use a hosted service provided by RStudio, or set up your own server. The first way is technically easier, but sometimes you may not be allowed to use an external hosted service, so you have to install the required software (Shiny Server or RStudio Connect) on your own server to deploy the Shiny documents.

### ShinyApps.io

You can publish Shiny documents to the ShinyApps (<https://shinyapps.io>) hosted service. To do this you, should ensure that you have:

1. An account on ShinyApps (use the signup form to request an account).
2. A latest version of the **rsconnect** R package. You can install this as follows:

#install.packages("rsconnect")

You can then deploy an interactive Shiny document in the same way that you currently deploy normal Shiny applications. From the working directory containing the document(s), just execute:

#rsconnect::deployApp()

If you are using RStudio, you can also use the Publish button available at the top-right of the window when running a Shiny document (see Figure @ref(fig:shiny-deploy)).

If there is a file named index.Rmd in the directory, it will be served as the default document for that directory, otherwise an explicit path to the Rmd file should be specified in the URL if you want to visit this Rmd document. For example, the URL for index.Rmd deployed to ShinyApps may be of the form https://example.shinyapps.io/appName/, and the URL for test.Rmd may be of the form https://example.shinyapps.io/appName/test.Rmd.

### Shiny Server / RStudio Connect

Both Shiny Server (<https://www.rstudio.com/products/shiny/shiny-server/>) and RStudio Connect (<https://www.rstudio.com/products/connect/>) can be used to publish Shiny documents. They require knowledge about Linux. Installing and configuring them should normally be a task for your system administrator if you are not familiar with Linux or do not have the privilege.

## Embedded Shiny apps

Besides embedding individual Shiny inputs and outputs in R Markdown, it is also possible to embed a standalone Shiny application within a document. There are two ways to do this:

1. Defining the application inline using the shinyApp() function; or
2. Referring to an external application directory using the shinyAppDir() function.

Both functions are available in the **shiny** package (not **rmarkdown**), which will be automatically loaded when runtime: shiny is specified in the YAML metadata of the document, so you do not have to call library(shiny) to load **shiny** (although it does not hurt if you load a package twice).

### Embedding Shiny apps via their URLs in R Markdown output - Overview

* With the knitr package, the functions knitr::include\_url() and include\_app() can be used by inserting an iframe or a screenshot/image in the output. iframes are basically a box on one web page to embed another web page. knitr::include\_app() was designed for embedding Shiny apps via their URLs in RMD output.
* <https://bookdown.org/yihui/bookdown/web-pages-and-shiny-apps.html>

### Embedding Shiny apps via their URLs in R Markdown output - Examples

* RMD HTML Output - <http://r4ds.had.co.nz/r-markdown-formats.html>
* <https://bookdown.org/>
* <http://rmarkdown.rstudio.com/rmarkdown_websites.html>
* <https://github.com/rstudio/blogdown>

### Shiny Components in rtutor - Overview & Examples

* rtutor package makes it easy to turn any R Markdown document into an interactive tutorial
* rtutor package uses runtime: shiny\_prerendered to turn regular R Markdown documents into live tutorials
* <https://rstudio.github.io/rtutor/index.html>
* <https://rstudio.github.io/rtutor/index.html#shiny_components>
* <https://rstudio.github.io/rtutor/examples.html>

### Further Reading

* <http://rmarkdown.rstudio.com/authoring_shiny_prerendered.html>
* <http://colorado.rstudio.com:3939/content/188/> by Edgar Ruiz
* <http://rmarkdown.rstudio.com/authoring_shiny_advanced.html>

# Applications

Some *significant* applications are demonstrated in this chapter.

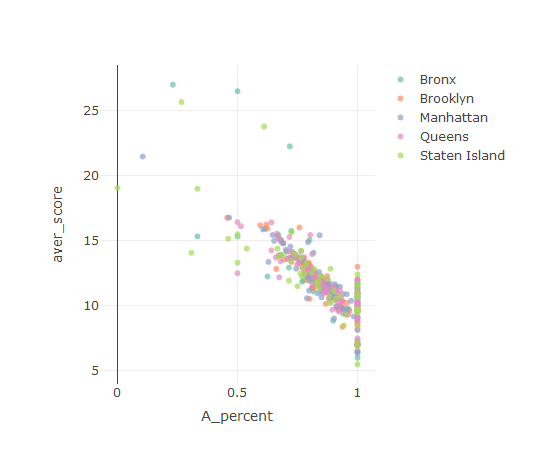
## Example one

Here is an example using flashboards in R (without shiny for now)

#read and clean data  
nyc\_inspections <-   
 read\_csv("C://Code/Combining-Shiny-R-Markdown/data/DOHMH\_New\_York\_City\_Restaurant\_Inspection\_Results.csv.gz",   
 col\_types = cols(building = col\_character()),  
 na = c("NA", "N/A")) %>%  
 clean\_names() %>%  
 filter(grade %in% c("A", "B", "C"),  
 boro != "Missing") %>%   
 mutate(inspection\_num = row\_number(),  
 boro = str\_to\_title(boro)) %>%   
 select(inspection\_num, boro, grade, score, critical\_flag, cuisine\_description)

### Grade A Percentage vs Average Score ScatterPlot

colourCount = length(unique(nyc\_inspections$boro))  
getPalette = colorRampPalette(brewer.pal(9, "Set2"))  
  
nyc\_inspections %>%  
 group\_by(boro, cuisine\_description) %>%  
 summarise(aver\_score = mean(score),  
 num\_inspections = n(),  
 A\_num = sum(grade == "A"),  
 A\_percent = sum(grade == "A")/n(),  
 critaical\_percent = sum(grade == "A")/n()) %>%  
 mutate(text\_label = str\_c("Boro: ", boro, '\nCuisine: ', cuisine\_description,   
 '\nnumber of inspection:', num\_inspections, '\nnumber of A:', A\_num,  
 '\n A%:', paste(round(A\_percent\*100,digits = 2),"%",sep = "")),   
 '\n critical%:', paste(round(critaical\_percent\*100,digits = 2),"%",sep = "")) %>%  
 plot\_ly(x = ~A\_percent, y = ~aver\_score, type = "scatter", mode = "markers",   
 alpha = 0.7, color = ~boro, text = ~text\_label)

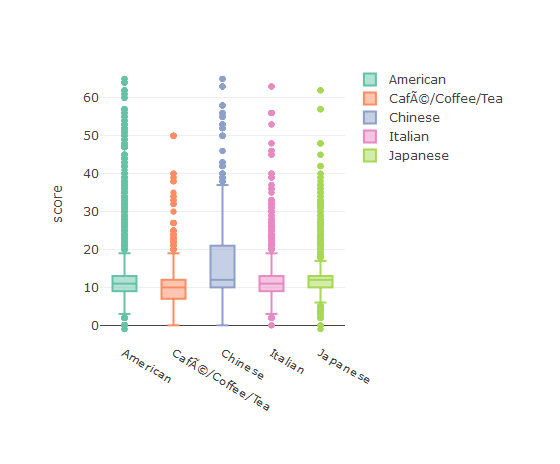


### Boxplot of Scores

colourCount = length(unique(nyc\_inspections$boro))  
getPalette = colorRampPalette(brewer.pal(9, "Set2"))

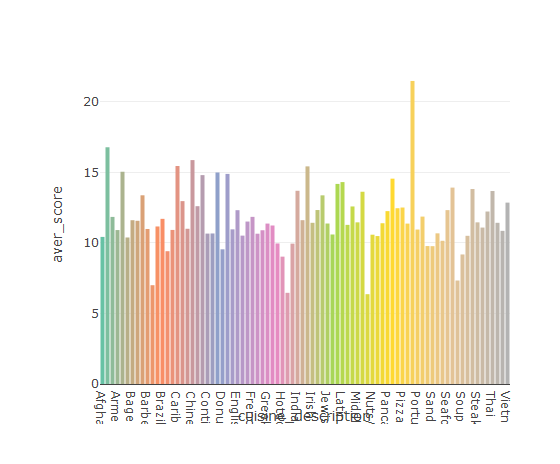
## Warning in brewer.pal(9, "Set2"): n too large, allowed maximum for palette Set2 is 8  
## Returning the palette you asked for with that many colors

cuisine\_boxplot <-  
 nyc\_inspections %>%  
 filter(boro == "Manhattan") %>%  
 group\_by(cuisine\_description) %>%  
 summarise(num\_inspections = n()) %>%  
 top\_n(5, num\_inspections) %>%  
 pull(cuisine\_description)  
  
nyc\_inspections %>%  
 filter(boro == "Manhattan",  
 cuisine\_description %in% cuisine\_boxplot) %>%  
 plot\_ly(y = ~score, color = ~cuisine\_description, type = "box", colors = "Set2")



### Bar-plot for Average Scores by Cuisine

nyc\_inspections %>%  
 group\_by(boro, cuisine\_description) %>%  
 summarise(aver\_score = mean(score),  
 num\_inspections = n(),  
 A\_num = sum(grade == "A"),  
 A\_percent = sum(grade == "A")/n(),  
 critaical\_percent = sum(grade == "A")/n()) %>%  
 mutate(text\_label = str\_c("Boro: ", boro, '\nCuisine: ', cuisine\_description,   
 '\nnumber of inspection:', num\_inspections, '\nnumber of A:', A\_num,  
 '\n A%:', paste(round(A\_percent\*100, digits = 2),"%",sep = "")),   
 '\n critical%:', paste(round(critaical\_percent\*100, digits = 2),"%",sep = "")) %>%  
 filter(boro == "Manhattan") %>%  
 plot\_ly(x = ~cuisine\_description, y = ~aver\_score, color = ~cuisine\_description, type = "bar", text = ~text\_label) %>%  
 layout(showlegend = FALSE)



## Example two

[still coming]

## Flashboard examples

## Shiny showcases

This is a collection of Shiny examples. You can see them in action on <http://gallery.shinyapps.io/example-name> where example-name is the directory name of an example here, e.g. <http://gallery.shinyapps.io/001-hello>

To run the examples locally, you can install the shiny package in R, and use the function runGithub(). For example, to run the example 001-hello:

# if (!require('shiny')) install.packages("shiny")  
# shiny::runGitHub("shiny-examples", "rstudio", subdir = "001-hello")  
# Or you can clone or download this repository, and use run shiny::runApp("001-hello").

Note the examples listed below depend on the development version of shiny to show some new features under development. Please be cautious that such features may or may not end up in the final release, or they may also change according to the feedback.

# Summary

This book is a work in progress for teaching Shiny, Flashboards and RMarkdown. Here are some short notes to summarise:

### Further Reading

* <http://rmarkdown.rstudio.com/authoring_shiny_prerendered.html>
* <http://colorado.rstudio.com:3939/content/188/> by Edgar Ruiz
* <http://rmarkdown.rstudio.com/authoring_shiny_advanced.html>

### License

This work is licensed under a Creative Commons Attribution 4.0 International License.