R Notebook Demo

Introduction

This notebook demonstrates many R Notebook features. Ignore this first chunk for now; we'll come back to it.

```
library(ggplot2)
library(dygraphs)
library(leaflet)
```

```
## Warning: replacing previous import by 'shiny::validateCssUnit' when loading
## 'crosstalk'
```

```
## Warning: replacing previous import by 'shiny::br' when loading 'crosstalk'
```

```
## Warning: replacing previous import by 'shiny::tags' when loading
## 'crosstalk'
```

```
## Warning: replacing previous import by 'shiny::div' when loading 'crosstalk'
```

```
## Warning: replacing previous import by 'shiny::h1' when loading 'crosstalk'
```

```
## Warning: replacing previous import by 'shiny::h2' when loading 'crosstalk'
```

```
## Warning: replacing previous import by 'shiny::h3' when loading 'crosstalk'
```

```
## Warning: replacing previous import by 'shiny::h4' when loading 'crosstalk'
```

```
## Warning: replacing previous import by 'shiny::h5' when loading 'crosstalk'
```

```
## Warning: replacing previous import by 'shiny::h6' when loading 'crosstalk'
```

```
## Warning: replacing previous import by 'shiny::knit_print.html' when loading
## 'crosstalk'
```

```
## Warning: replacing previous import by 'shiny::tagSetChildren' when loading
## 'crosstalk'
```

```
## Warning: replacing previous import by 'shiny::includeScript' when loading
## 'crosstalk'
```

```
## Warning: replacing previous import by 'shiny::em' when loading 'crosstalk'
```

```
## Warning: replacing previous import by 'shiny::tagAppendChild' when loading
## 'crosstalk'
```

```
## Warning: replacing previous import by 'shiny::is.singleton' when loading
## 'crosstalk'
```

```
## Warning: replacing previous import by 'shiny::includeHTML' when loading
## 'crosstalk'
```

```
## Warning: replacing previous import by 'shiny::includeMarkdown' when loading
## 'crosstalk'
```

```
## Warning: replacing previous import by 'shiny::code' when loading
## 'crosstalk'
```

```
## Warning: replacing previous import by 'shiny::tagList' when loading
## 'crosstalk'
## Warning: replacing previous import by 'shiny::a' when loading 'crosstalk'

## Warning: replacing previous import by 'shiny::tagAppendAttributes' when
## loading 'crosstalk'

## Warning: replacing previous import by 'shiny::singleton' when loading
## 'crosstalk'

## Warning: replacing previous import by 'shiny::hr' when loading 'crosstalk'

## Warning: replacing previous import by 'shiny::p' when loading 'crosstalk'

## Warning: replacing previous import by 'shiny::suppressDependencies' when
## loading 'crosstalk'

## Warning: replacing previous import by 'shiny::tagAppendChildren' when
## loading 'crosstalk'

## Warning: replacing previous import by 'shiny::includeText' when loading
## 'crosstalk'

## Warning: replacing previous import by 'shiny::pre' when loading 'crosstalk'

## Warning: replacing previous import by 'shiny::span' when loading
## 'crosstalk'

## Warning: replacing previous import by 'shiny::withTags' when loading
## 'crosstalk'

## Warning: replacing previous import by 'shiny::htmlTemplate' when loading
## 'crosstalk'

## Warning: replacing previous import by 'shiny::img' when loading 'crosstalk'

## Warning: replacing previous import by 'shiny::tag' when loading 'crosstalk'

## Warning: replacing previous import by 'shiny::includeCSS' when loading
## 'crosstalk'

## Warning: replacing previous import by 'shiny::knit_print.shiny.tag' when
## loading 'crosstalk'

## Warning: replacing previous import by 'shiny::knit_print.shiny.tag.list'
## when loading 'crosstalk'

## Warning: replacing previous import by 'shiny::strong' when loading
## 'crosstalk'

## Warning: replacing previous import by 'shiny::HTML' when loading
## 'crosstalk'
```

Options for Chunck

Cache

```r
for (i in 1:5000) {
    lm((i+1)~i)
}
```

Echo

```
## [1] 0.9002541
```

Include

Output Types

Console Output

First, let's create a simple code chunk that prints the numbers from 1 to 10. Run this chunk with *Ctrl+Shift+Enter* (*Cmd+Shift+Enter* on OS X).

```
numbers <- seq_len(15)
numbers
```

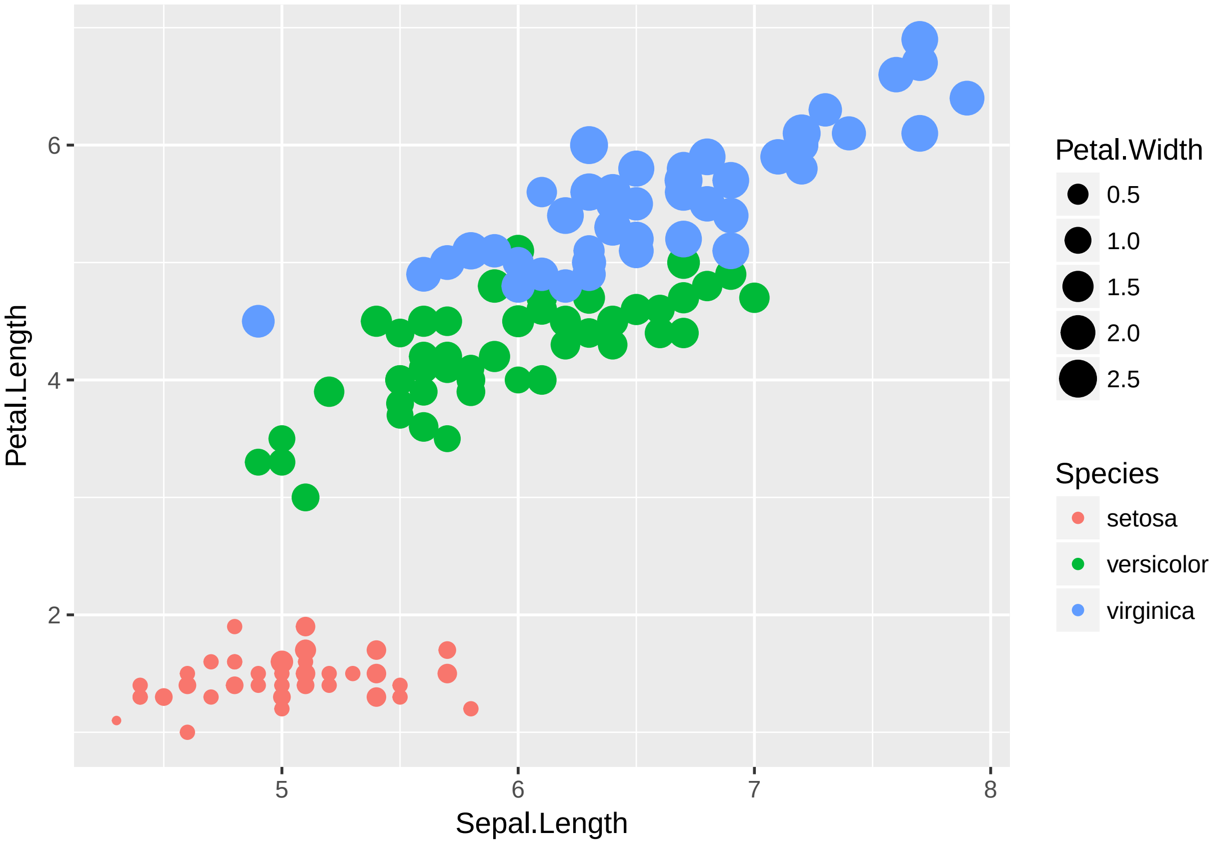
```
##  [1]  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
```

Notice how the output appears right beneath the chunk, as though you'd run the code at the R console. Try changing the sequence length to 20 and re-running the chunk to see the result update.

Graphical Output

Of course, sometimes your R code doesn't just produce text; it produces graphical output such as plots. These are supported, too. Try running this chunk:

```
qplot(Sepal.Length, Petal.Length, data = iris, color = Species,
      size = Petal.Width)
```



Notice that it isn't necessary to prefix `qplot` with `ggplot2::` since there is a `library(ggplot2)` call in the setup chunk. RStudio runs the setup chunk automatically whenever it's needed.

HTML Widgets

If your R analysis involves interactive components, you're probably already familiar with the `htmlwidgets` library. These, too, are supported in the notebook. Run this chunk to see an interactive graph:

```r
dygraph(nhtemp, main = "New Haven Temperatures") %>%
  dyRangeSelector(dateWindow = c("1920-01-01", "1960-01-01"))
```

Running Code

One of the goals of the notebook is to provide a seamless environment for interacting with R – that is, you shouldn't need to reach for the console, even though chunks send code there. To help you see the progress of your chunk – that is, which lines have been executed and which haven't – RStudio draws an indicator in the editor gutter. Try running this chunk:

```r
Sys.sleep(1); runif(3)
```

```
## [1] 0.1516642 0.3888231 0.3864671
```

```r
Sys.sleep(1); runif(3)
```

```
## [1] 0.2357104 0.9469257 0.2918526
```

```
Sys.sleep(1); runif(3)
```

```
## [1] 0.7463086 0.3377681 0.8260968
```

```
Sys.sleep(1); runif(3)
```

```
## [1] 0.3432023 0.6009522 0.5747573
```

```
Sys.sleep(1); runif(3)
```

```
## [1] 0.2296476 0.8276037 0.6403516
```

Alternate Engines

By default, your R notebook chunks will be run using R. However, it's entirely possible to write chunks that use other engines to execute. For instance, you can add some Python to your notebook:

```
def fib(n):
    a, b = 0, 1
    for _ in xrange(n):
        yield a
        a, b = b, a + b
print list(fib(11))
```

```
## [0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55]
```

Try using the Feather package for R and Python to transfer data between them.

You can use a notebook to form an executable, documented workflow that composes several tools into a complete data analysis workflow.

Errors

Sometimes your code will generate errors. Here's an example:

```
# Source a file that doesn't exist
source("missing.R")
```

```
## Warning in file(filename, "r", encoding = encoding): no fue posible abrir
## el archivo 'missing.R': No existe el archivo o el directorio
```

```
## Error in file(filename, "r", encoding = encoding): no se puede abrir la conexión
```

Notice that the line that caused the error is highlighted, and you can see the error's traceback, just as you can in the RStudio console. If an error occurs while you're running chunks, the error will cause the notebook to stop running, and the cursor will scroll to the point where the error occurred.

Tables

```
library(nycflights13)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
flights %>%
  count(origin)
```

```
## # A tibble: 3 x 2
##   origin      n
##    <chr>  <int>
## 1    EWR 120835
## 2    JFK 111279
## 3    LGA 104662
```

```r
flights %>%
  count(origin) %>%
  knitr::kable()
```

origin	n
EWR	120835
JFK	111279
LGA	104662

Saving and Sharing

A notebook's source code is always in an `.Rmd` file. Whenever you save it, a sidecar `.nb.html` file is generated. This file contains a rendered copy of the notebook itself. No special viewer is required.

It also contains a copy of the notebook's source `.Rmd` file.

To look at the `.nb.html` file, click *Preview* in the RStudio editor toolbar. This is a fundamental difference between notebooks and other R Markdown documents; pressing this button doesn't actually cause any of your code to run, it just shows you the HTML file already prepared. It will automatically update whenever you save the `.Rmd` file.

If you open the `.nb.html` file in a web browser, you'll see an option to download the source. You can also open an `.nb.html` file in RStudio; when you do this, RStudio will automatically extract the `.Rmd` file and outputs inside it and open the file in the notebook editor.

Alternate formats

Notebooks as R Markdown Documents

A notebook is also an R Markdown document. Try changing the YAML header in this document so that `html_document` is the first option, then clicking `Knit` (or just pulling down the *Preview*) menu. You could also create a PDF from the notebook, a Word document, or even a dashboard.

R Markdown Documents as Notebooks

By now you've probably realized that any R Markdown document is also a notebook. If you don't like this behavior and prefer to work with the console directly, pull down the gear icon in the editor toolbar and choose Show Chunk Output Inline; there's also a global pref.

Further reading

R Notebook Reference