Rmarkdown (and other things)

rk595@cam.ac.uk

Stats Meeting 26th Nov 2018

1. Package ‘ggplot’
2. Package ‘kableExtra’
3. Rmarkdown

Package ‘ggplot’

Package 'ggplot'

- Plot graphics in R
- More complicated than standard R graphics for simple plots. . .
- . . . but easier for complex graphics / custom formatting.
- Plot formed of overlapping layers
- Requires a data.frame object

Example

Example

The Iris dataset

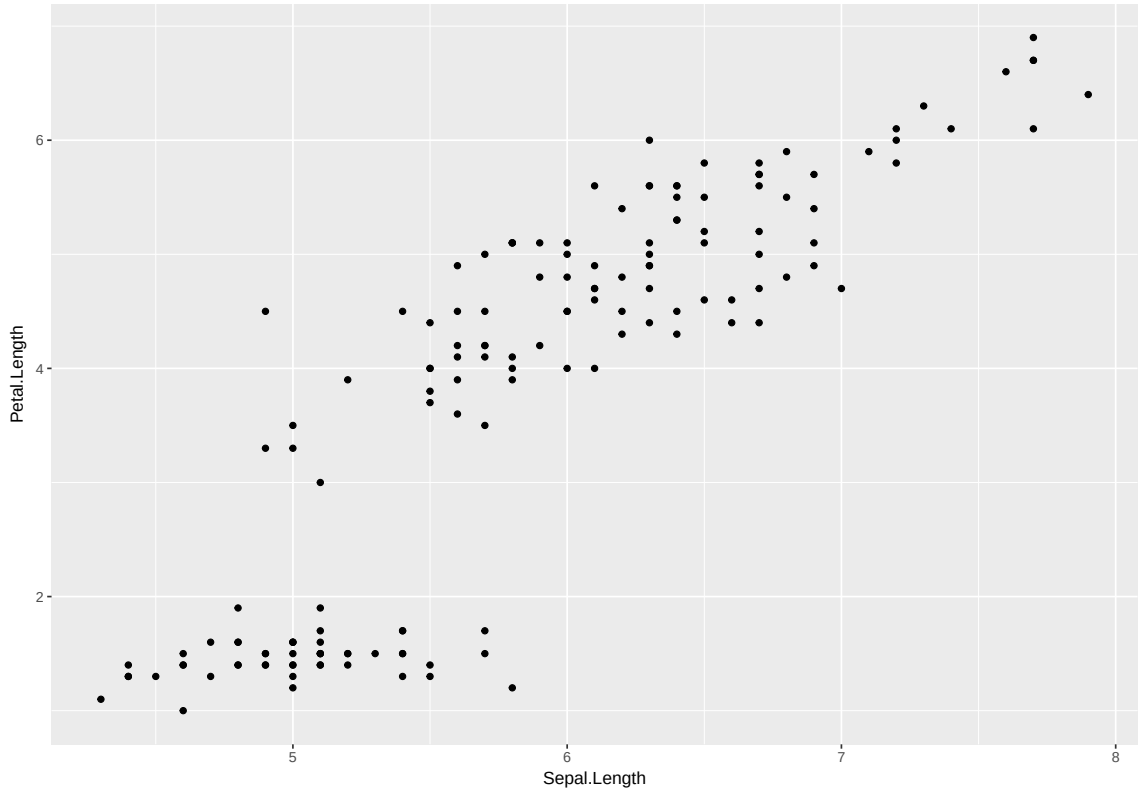
```
##     Sepal.Length Sepal.Width Petal.Length Petal.Width Species
## 1            5.1         3.5          1.4         0.2  setosa
## 2            4.9         3.0          1.4         0.2  setosa
## 3            4.7         3.2          1.3         0.2  setosa
## 4            4.6         3.1          1.5         0.2  setosa
## 5            5.0         3.6          1.4         0.2  setosa
## 6            5.4         3.9          1.7         0.4  setosa
## 7            4.6         3.4          1.4         0.3  setosa
## 8            5.0         3.4          1.5         0.2  setosa
## 9            4.4         2.9          1.4         0.2  setosa
## 10           4.9         3.1          1.5         0.1  setosa
## 11           5.4         3.7          1.5         0.2  setosa
## 12           4.8         3.4          1.6         0.2  setosa
## 13           4.8         3.0          1.4         0.1  setosa
## 14           4.3         3.0          1.1         0.1  setosa
## 15           5.8         4.0          1.2         0.2  setosa
```

Example

```
ggplot(iris)
```

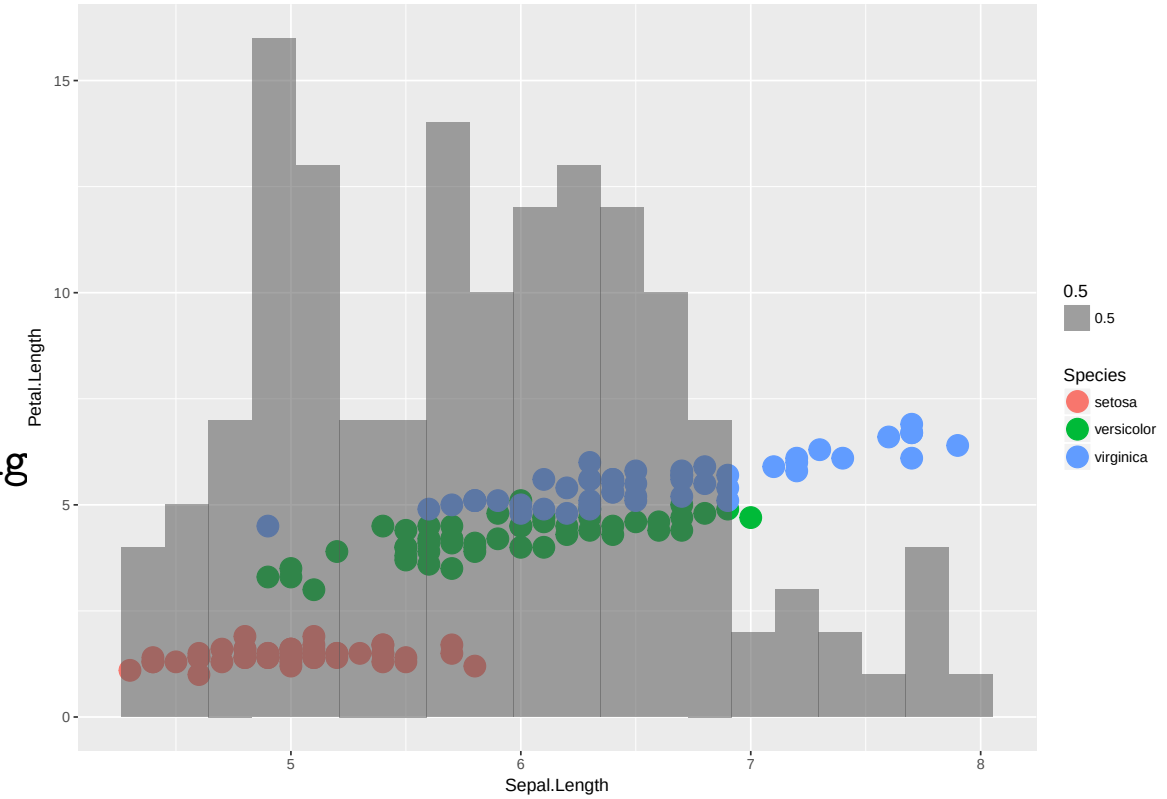
Example

```
ggplot(iris) +
geom_point(aes(x=Sepal.Length,
y=Petal.Length)
```



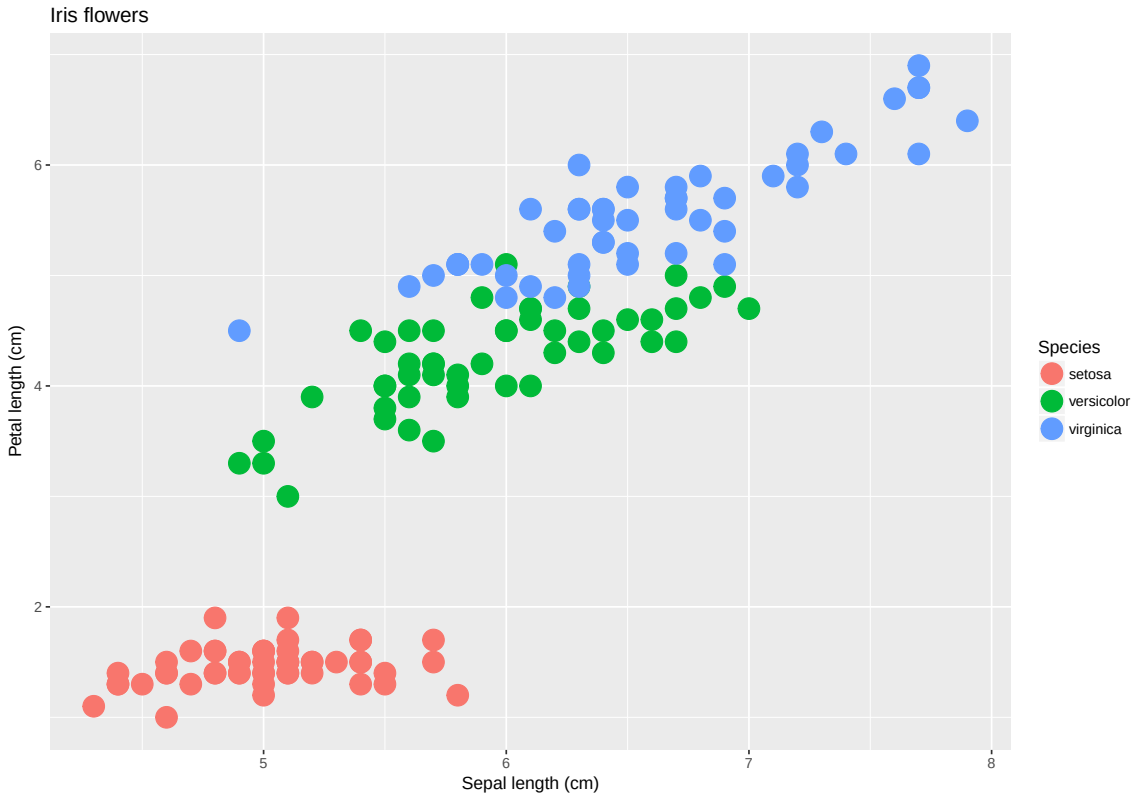
Example

```
ggplot(iris) +
geom_point(aes(x=Sepal.Length,
y=Petal.Length,
col=Species), size=6) +
geom_histogram(aes(x=Sepal.Leng
y=..count.., alpha=0.5),
bins = 20)
```



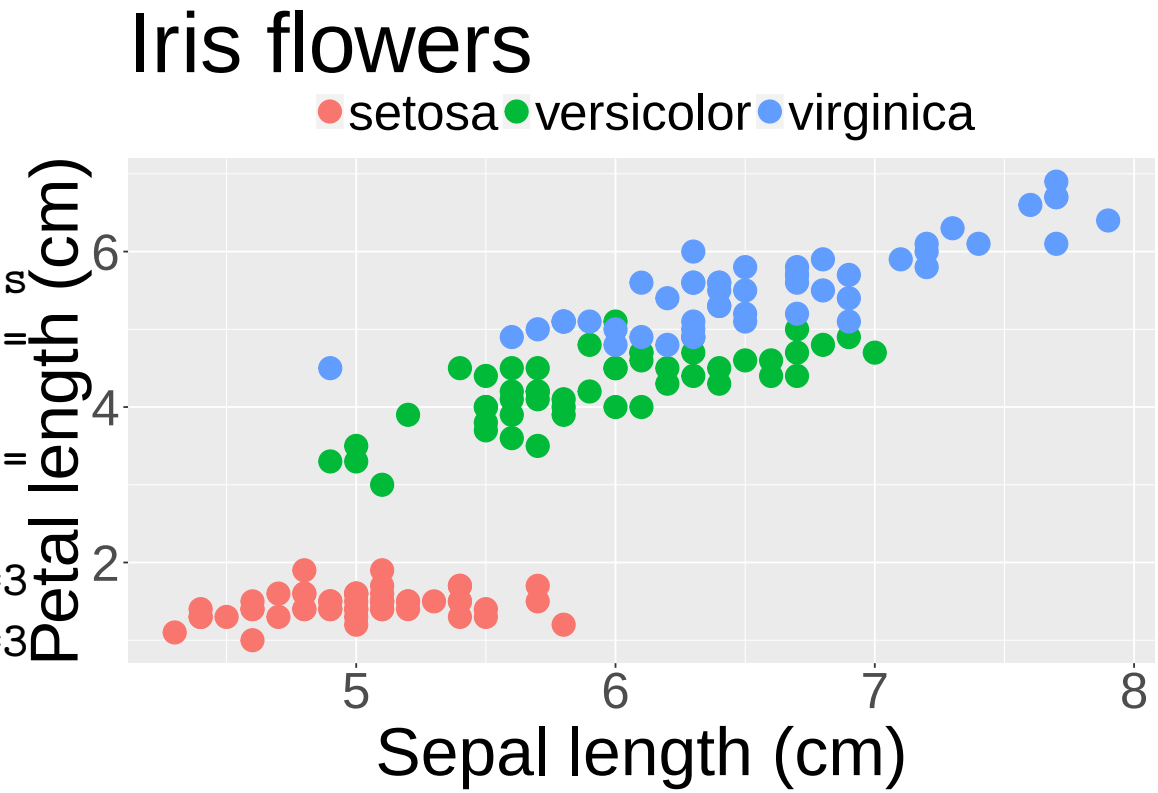
Example

```
ggplot(iris) +
geom_point(aes(x=Sepal.Length,
y=Petal.Length,
col=Species), size=6) +
labs(x="Sepal length (cm)",
y="Petal length (cm)",
title="Iris flowers")
```



Example

```
ggplot(iris) +
geom_point(aes(x=Sepal.Length,
y=Petal.Length,
col=Species), size=6) +
labs(x="Sepal length (cm)",
y="Petal length (cm)",
title="Iris flowers") +
theme(plot.title=element_text(s
axis.title.x=element_text(size=
vjust=-1),
axis.title.y=element_text(size=
vjust=1),
axis.text.x=element_text(size=3
axis.text.y=element_text(size=3
legend.title=element_blank(),
legend.position = "top",
legend.text=element_text(size=30),
plot.margin =
unit(c(1,1,1,1), "cm"))
```



Example

But it will be such a pain to write such a long code for each one of my plots...

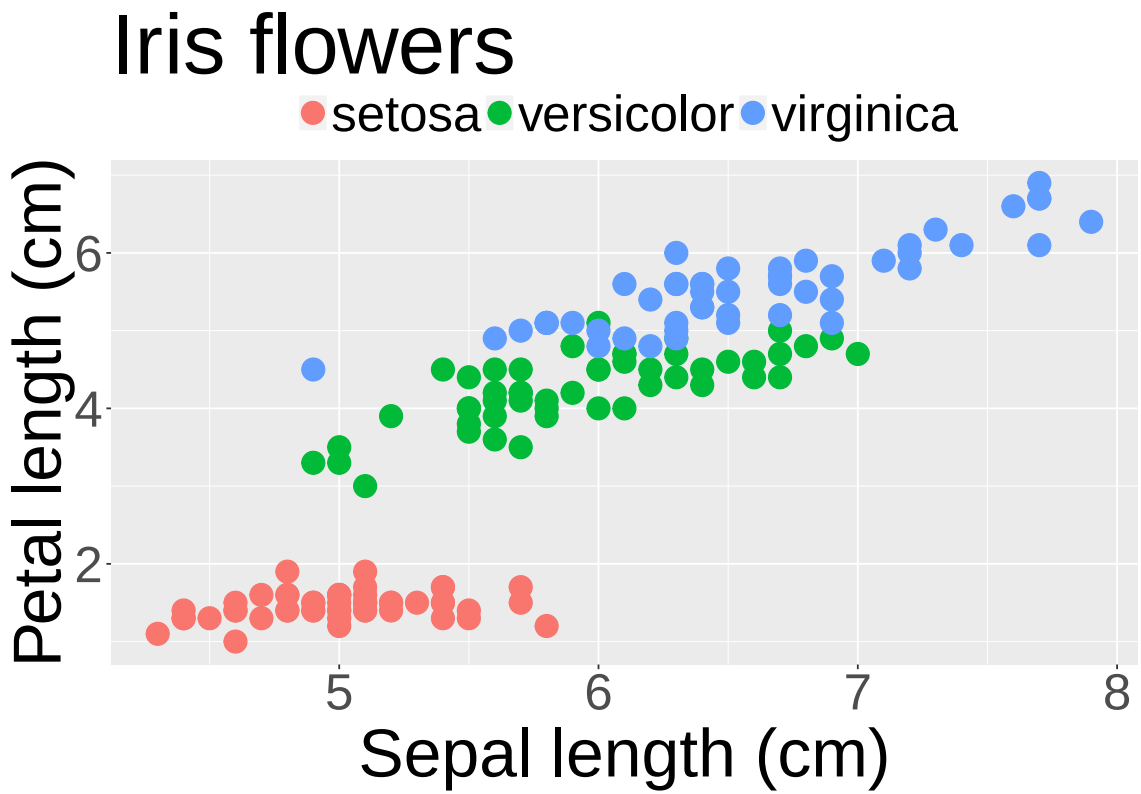
Fear not!

```
myTheme <- theme(plot.title=element_text(size=50),
                 axis.title.x=element_text(size=40, vjust=-1),
                 axis.title.y=element_text(size=40, vjust=1),
                 axis.text.x=element_text(size=30),
                 axis.text.y=element_text(size=30),
                 legend.title=element_blank(),
                 legend.position = "top",
                 legend.text=element_text(size=30),
                 plot.margin = unit(c(1,1,1,1), "cm"))
```

Then all you have to do is add a '+ myTheme' at the end of your ggplot.

Example

```
ggplot(iris) +
geom_point(aes(x=Sepal.Length,
y=Petal.Length,
col=Species), size=6) +
labs(x="Sepal length (cm)",
y="Petal length (cm)",
title="Iris flowers") +
myTheme
```



Resources

- The Tidyverse official guide
- R for Data Science, chapter 3
- Detailed "cheat sheet" with many examples
- Themes and backgrounds
- Colours (and colourblind-friendly palettes)
- Legends
- "Cookbook" for R - ggplot2
- Summary cheat sheet

Package 'kableExtra'

Package 'kableExtra'

- `kable` is a function from the `knitr` package
  - simple table generator
- `kableExtra` turns kable objects into fancier tables
  - exports to HTML and PDF
  - copy/paste from HTML often works to import tables to MS Word / Excel
- Allows for a "layer" syntax (sort of similar to ggplot)

Example

First of all, we need a matrix with the data we want to output:

```
sepalWidthF1 <- data.frame(table(iris$Sepal.Width[iris$Species=="setosa"]))
sepalWidthF2 <- data.frame(table(iris$Sepal.Width[iris$Species=="versicolor"]))
sepalWidthF3 <- data.frame(table(iris$Sepal.Width[iris$Species=="virginica"]))

myTable <- merge(merge(sepalWidthF1, sepalWidthF2, by=1, all=TRUE),
      sepalWidthF3, by=1, all=TRUE)
myTable <- myTable[order(as.numeric(as.character(myTable[,1]))),]
myTable <- cbind(c("Sepal Width (cm)", "Setosa", "Versicolor", "Virginica"), t(myTable))

myTable[is.na(myTable)] <- " "

head(myTable)
```

```
##                         17   18    1     19    20    21    22    23
## Var1   "Sepal Width (cm)" "2"  "2.2" "2.3" "2.4" "2.5" "2.6" "2.7" "2.8"
## Freq.x "Setosa"           " "  " "   " 1"  " "   " "   " "   " "   " "
## Freq.y "Versicolor"       " 1" " 2"  " 3"  " 3"  " 4"  " 3"  " 5"  " 6"
## Freq   "Virginica"        " "  " 1"  " "   " "   " 4"  " 2"  " 4"  " 8"
##        2     3    4     5     6     7     8     9     10    11    12
## Var1   "2.9" "3"  "3.1" "3.2" "3.3" "3.4" "3.5" "3.6" "3.7" "3.8" "3.9"
## Freq.x " 1"  " 6" " 4"  " 5"  " 2"  " 9"  " 6"  " 3"  " 3"  " 4"  " 2"
## Freq.y " 7"  " 8" " 3"  " 3"  " 1"  " 1"  " "   " "   " "   " "   " "
## Freq   " 2"  "12" " 4"  " 5"  " 3"  " 2"  " "   " 1"  " "   " 2"  " "
##        13   14    15    16
## Var1   "4"  "4.1" "4.2" "4.4"
## Freq.x " 1" " 1"  " 1"  " 1"
## Freq.y " "  " "   " "   " "
## Freq   " "  " "   " "   " "
```

Example

```
myKable <- kable(myTable, format="latex", booktabs=TRUE,
                 row.names=FALSE, col.names=NULL)
myKable
```

Sepal Width (cm)	2	2.2	2.3	2.4	2.5	2.6	2.7	2.8	2.9	3	3
Setosa			1						1	6	4
Versicolor	1	2	3	3	4	3	5	6	7	8	3
Virginica		1			4	2	4	8	2	12	4

Example

```
theHeader <- c(1, 9, 10, 4)
names(theHeader) <- c(" ", "Small", "Medium", "Large")

myKable %>%
    kable_styling(full_width=FALSE, position="center",
                  latex_options="scale_down") %>%
    group_rows("Species", 2, 4, latex_gap_space="1em") %>%
    add_header_above(theHeader) %>%
    footnote(general = "Here is a general comment about the table.")
```

	Small									Medium										Large			
Sepal Width (cm)	2	2.2	2.3	2.4	2.5	2.6	2.7	2.8	2.9	3	3.1	3.2	3.3	3.4	3.5	3.6	3.7	3.8	3.9	4	4.1	4.2	4.4
Species																							
Setosa			1						1	6	4	5	2	9	6	3	3	4	2	1	1	1	1
Versicolor	1	2	3	3	4	3	5	6	7	8	3	3	1	1									
Virginica		1			4	2	4	8	2	12	4	5	3	2		1		2					

Note:
Here is a general comment about the table.

Resources

I'm sure there must be lots of excellent guides out there, but these are always my go-to places (and it often has the answers):

- Create awesome HTML tables with knitr:kable
- Create awesome PDF tables with knitr:kable
- The kableExtra complete documentation

Rmarkdown

Rmarkdown

- Very useful for writing up drafts / reports / etc.
- Allows for code "chunks" to be run straight from 
  - If your data change, all of your figures and tables are automatically updated!
- Allows for flexible document structure
  - Tabs, navigation bar, etc.
- This presentation was written using the Rmarkdown/Beamer output!
  - It requires a bit of LaTeX knowledge...
  - ...but there are other interfaces available (e.g., *ioslides*, *slidy*, *xaringan*)

Example: html document

Header:

```
---
title: My first Rmd document
author: My name
date: "`r format(Sys.time(), '%d %b %Y')`" <- this displays current date
output: html_document:
    toc: true      <- four spaces - no tabs!
    toc_depth: 2
    number_sections: true
    theme: united
    highlight: tango
    ...
---
```

Example: html document

Body:

```
After the header, you can just imput normal(ish) text.

- We can make lists by using hyphens
    + then four spaces to add subitems
    + like these

It's possible to make the font _italic_ or **bold** as well.

To change text colour inline, <span style="color:blue"> we need to use the

# It's easy to add section headers

## And subheaders as well {.tabset .tabset-pills}

### We can also make tabs

With something different in each tab

### I'm another tab
```

Example: html document

Body:

```
## But that's not all!

The really cool thing about Rmarkdown are the code chunks:
\```{r, echo=TRUE}
    x <- rnorm(100,0,1)
    y <- rnorm(100,0,1)
    plot(x,y)
\```
```

Other formats / resources

- Rmarkdown cheat sheet
- MS Word document
- Rmarkdown also handles PDF outputs in a similar way, however, you probably need some knowledge of LaTeX to deal with the errors.
- Xaringan - html presentations
- Slidy - html presentations
- Ioslides - html presentations
- Shiny and Rmarkdown - responsive html