Assignment template

James Bond, group 007

1 February 2033

Short introduction to R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. R Markdown files permit you to interweave R code with ordinary text to produce well-formatted data analysis reports that are easy to modify. The R Markdown file itself shows the readers exactly how you got the results in your report. For more details on using R Markdown see http://rmarkdown.rstudio.com.

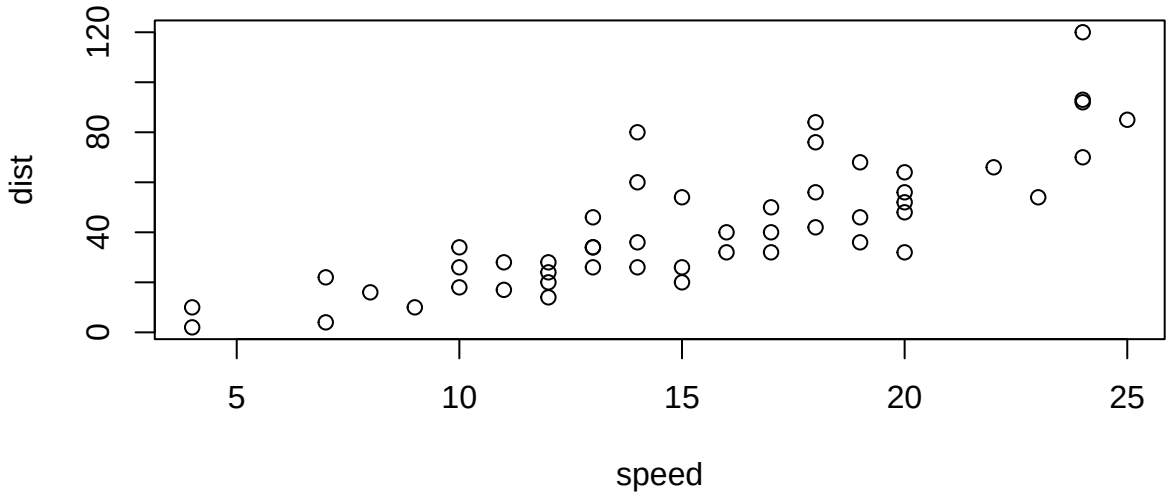
When you click the **Knit** button, a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. For inline R code, surround code with back ticks and r. R replaces inline code with its results. For example, two plus one is 3; for the build-in R dataset `cars`, there were 50 cars studied. 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
```

Figures

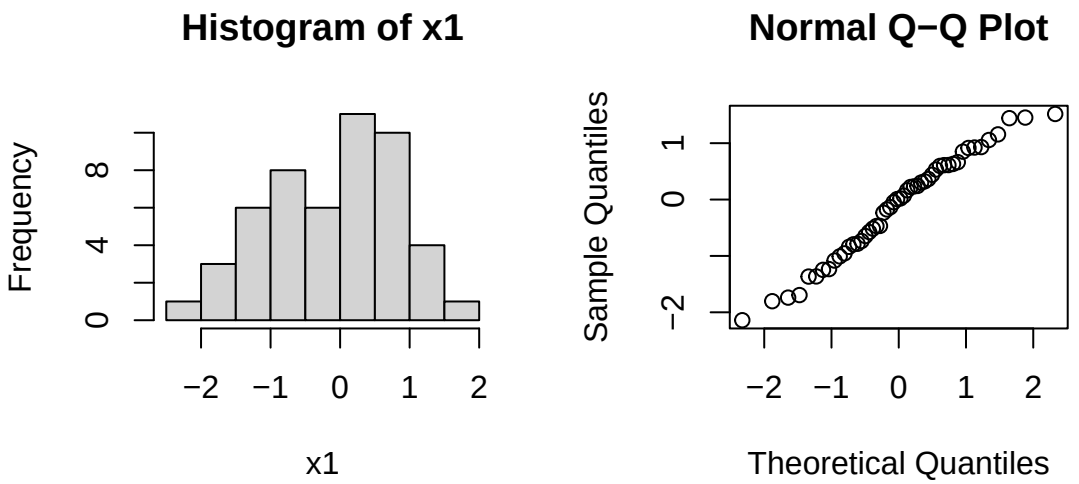
You can also embed plots, for example:



Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot. Use knitr options to style the output of a chunk. Place options in brackets above the chunk. Other options with the defaults are: the `eval=FALSE` option just displays the R code (and does not run it); `warning=TRUE` whether to display warnings; `tidy=TRUE` wraps long code so it does not run off the page.

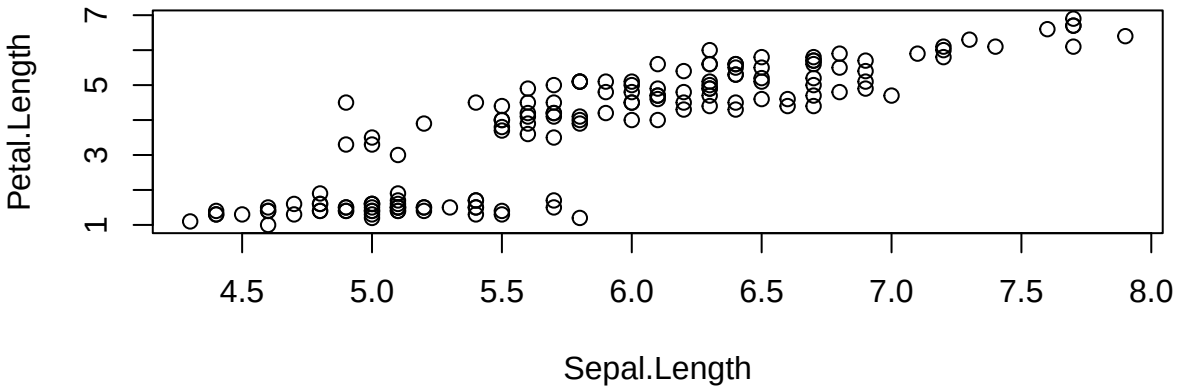
You can control the size and placement of figures. For example, you can put two figures (or more) next to each other. Use `par(mfrow=c(n,m))` to create `n` by `m` plots in one picture in R. You can adjust the proportions of figures by using the `fig.width` and `fig.height` chunk options. These are specified in inches, and will be automatically scaled down to fit within the handout margin. Chunk option `fig.align` takes values `left`, `right`, or `center` (to align figures in the output document).

```
par(mfrow=c(1,2)); x1=rnorm(50); hist(x1); qqnorm(x1)
```



You can arrange for figures to span across the entire page by using the `fig.fullwidth` chunk option.

```
plot(iris$Sepal.Length,iris$Petal.Length,xlab="Sepal.Length",ylab="Petal.Length")
```



More about chunk options can be found at https://yihui.name/knitr/options/.

Equations

To produce mathematical symbols, you can also include LaTeX expessions/equations in your report: inline $\frac{d}{dx}\left(\int_0^x f(u)\,du\right) = f(x)$ and in the display mode: To be able to use this functionality, LaTeX has to be installed.

Footnotes

Here is the use of a footnote[1].

Images

Want an image? This will do it. To depict an image (say, `my_image.jpg` which should be in your current working directory), use this command

Tables

Want a table? This will create one (note that the separators *do not* have to be aligned).

Table Header	Second Header
Table Cell	Cell 2
Cell 3	Cell 4

You can also make table by using knit's `kable` function:

[1]

Exercise 1. Waiting time

A researcher measured (in minutes) how long patients have to wait in the waiting room of a doctor's office: 15.4, 17.9, 19.0, 0.5, 15.9, 2.7, 6.2, 2.5, 4.7, 6.9, 10.8, 24.3, 5.6, 23.0, 10.7. Denote the mean waiting time by /mu.

Table 2: A knit kable.

	mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb
Mazda RX4	21.0	6	160	110	3.90	2.620	16.46	0	1	4	4
Mazda RX4 Wag	21.0	6	160	110	3.90	2.875	17.02	0	1	4	4
Datsun 710	22.8	4	108	93	3.85	2.320	18.61	1	1	4	1
Hornet 4 Drive	21.4	6	258	110	3.08	3.215	19.44	1	0	3	1
Hornet Sportabout	18.7	8	360	175	3.15	3.440	17.02	0	0	3	2

Block quote

> This will create a block quote, if you want one.

Verbatim

```
This text is displayed verbatim/preformatted.
```

Links

Links: http://example.com, in-text link to Google.

This is a hyperlink.

This

is where the hyperlink jumps to.

Itimization, italicized and embolded text

- Single asterisks italicize text *like this*.
- Double asterisks embolden text **like this**.

One more way to italicize and embold: *italic* and **bold**.

Exercise 1

Below is a template for reporting the exercises from the assignments.

a) Here are some consequitive R-commands.

```
x=rep(c("A","B"),each=5); x
```

```
##  [1] "A" "A" "A" "A" "A" "B" "B" "B" "B" "B"
```

```
sample(x)
```

```
##  [1] "B" "B" "A" "A" "A" "B" "A" "B" "A" "B"
```

```
x=rnorm(100)
```

Now the same code chunk but with all the output collapsed into signle block.

```
x=rep(c("A","B"),each=5); x
##  [1] "A" "A" "A" "A" "A" "B" "B" "B" "B" "B"
sample(x)
##  [1] "B" "A" "B" "A" "A" "B" "A" "B" "A" "B"
x=rnorm(100)
```

b) Below we perform a one sample t-test for the artificial data (that we generate ourselves).

```
mu=0.2
x=rnorm(100,mu,1) # creating artificial data
t.test(x,mean=0)   # t.test(x,alternative=c("two.sided"),conf.level=0.95,mu=10)
```

```
##
##  One Sample t-test
##
## data:  x
## t = 4.9421, df = 99, p-value = 3.151e-06
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
##  0.2492243 0.5835948
## sample estimates:
## mean of x
## 0.4164095
```

c) We often do not need to report the whole output of R-commands, only certain values of the output. For example, below we perform a two-sample t-test and report only the (appropriately rounded) values of t-statistics and the p-pavue.

```
mu=0;nu=0.5
x=rnorm(50,mu,1); y=rnorm(50,nu,1) # creating artificial data
ttest=t.test(x,y)
```

The value of t-statistics in the above evaluation is -3.07 and the p-value is 0.0028.

Exercite 1. Waiting time.

A researcher measured (in minutes) how long patients have to wait in the waiting room of a doctor's office: 15.4, 17.9, 19.0, 0.5, 15.9, 2.7, 6.2, 2.5, 4.7, 6.9, 10.8, 24.3, 5.6, 23.0, 10.7. Denote the mean waiting time by μ.

```
x <- as.numeric(list(15.4, 17.9, 19.0, 0.5, 15.9, 2.7, 6.2, 2.5, 4.7, 6.9, 10.8, 24.3, 5.6, 23
```

a) Check normality of the data. Assuming normality (irrespective of your conclusion about normality of the data), construct a 97%-CI for μ. Evaluate the sample size needed to provide that the length of the 97%-CI is at most 2. Compute a bootstrap 97%-CI for μ and compare it to the above CI.

Let's check the normality using Shapiro-Wilk test. H_0 is that sample x came from normally distributed population.

```
shapiro.test(x)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  x
## W = 0.93473, p-value = 0.3207
```

From the output, the p-value > 0.05 implying that the distribution of the data are not significantly different from normal distribution, i.e. the null hypothesis can not be rejected. In other words, we can assume the normality.

Estimated mean value:

```
mu = mean(x)
mu
```

```
## [1] 11.07333
```

Next, we are going to construct a 97%-CI for μ. The standard deviation σ is unknown, therefore, we estimate it by s.

```
s = sd(x)
s
```

```
## [1] 7.727545
```

The confidence interval in such a case is based on a t-distribution and the upper t-quantile.

```
alpha <- 1 - 0.97
n <- length(x)
ta <- qt(1-alpha, df=n-1)
ta
```

```
## [1] 2.046169
```

t-confidence interval of level 3% for μ:

```
CI_97 <- list(mu - ta*s/sqrt(n), mu + ta*s/sqrt(n))
CI_97
```

```
## [[1]]
## [1] 6.990728
##
## [[2]]
## [1] 15.15594
```

Next, we evaluate the sample size needed to provide that the length of the 97%-CI is at most 2. For this, we have to solve $t_{\alpha/2}\frac{s}{\sqrt{n}} \leq E$ for n.

```
E <- 2
n_min <- (ta*s/E)^2
n_min
```

```
## [1] 62.50376
```

To provide the length of the 97%-CI less than 2, we have to collect the sample of at lest 63 objects