Statistics 305: Introduction to Biostatistical Methods for Health Sciences

R Demo for Chapters 2 and 3: Data presentations and summaries

Jinko Graham

2018-09-01

Note on Slides

Demo Slides

- This document shows and explains the R commands used to create the data summaries of the Chapters 2 and 3 lecture slides.
  - For brevity, I omit slides and remarks that do not pertain to computations done in R.
- This document should be read **after** reading the lecture slides.
- You may find the R commands in this demo useful for your homework assignments.

Data Presentation (Chapter 2)

Tables

- Tables can be used to display the frequency distribution of a categorical variable
- Example: Frequency distribution of gender among 21,737 bladder cancer patients. Data from Mungan et al. (2000)

```
uu <- url("http://people.stat.sfu.ca/~jgraham/Teaching/S305_17/Data/mung.csv")
Mungan <- read.csv(uu)
head(Mungan)
```

```
##   Gender Cancer.Stage
## 1   Male            I
## 2   Male            I
## 3   Male            I
## 4   Male            I
## 5   Male            I
## 6   Male            I
```

```
with(Mungan,table(Gender))
```

```
## Gender
## Female   Male
##   5536  16201
```

Software Notes: Reading Data Into R

- `read.csv()` reads comma-separated-value (CSV) files into R.
  - By default it reads files from the "working" directory in which R is running (e.g., the folder of your Jupyter notebook), but it can read files from URLs too.
  - The `url()` function takes a quoted URL as input and returns an object that `read.csv()` can use to fetch the file.
- `read.table()` is a more flexible function for reading tabular data into R; e.g.,

```
uu <- url("http://people.stat.sfu.ca/~jgraham/Teaching/S305_17/Data/mung.csv")
Mungan <- read.table(uu,header=TRUE,sep=",")
```

- `read.table()` options include the following (type `help("read.table")` into R for a full list):
  - `header` (default `FALSE`): Does the first line of the file contain the variable names?
  - `sep` (default "", for blank spaces)

Software Notes: `head()`,`with()` and `table()`

- The `head()` function looks at the first few rows (default is six) of a dataset.
  - In the example, the dataset is called `Mungan`, having variables `Gender` and `Cancer.Stage`.
  - Datasets have as many rows as there are sampled units (e.g., people) and as many columns as there are variables measured on the sampled units.
- The `with()` function takes a dataset as its first argument and the summary to compute as its second.
  - In the above example, the summary is a table of the values of the `Gender` variable in the `Mungan` dataset.
- The `table()` function tabulates the unique values of a variable, or, if given two variables, cross-tabulates the two variables (more on cross-tabulation in Chapter 15).

Tables, cont.

- Joint frequency distribution of two categorical variables:

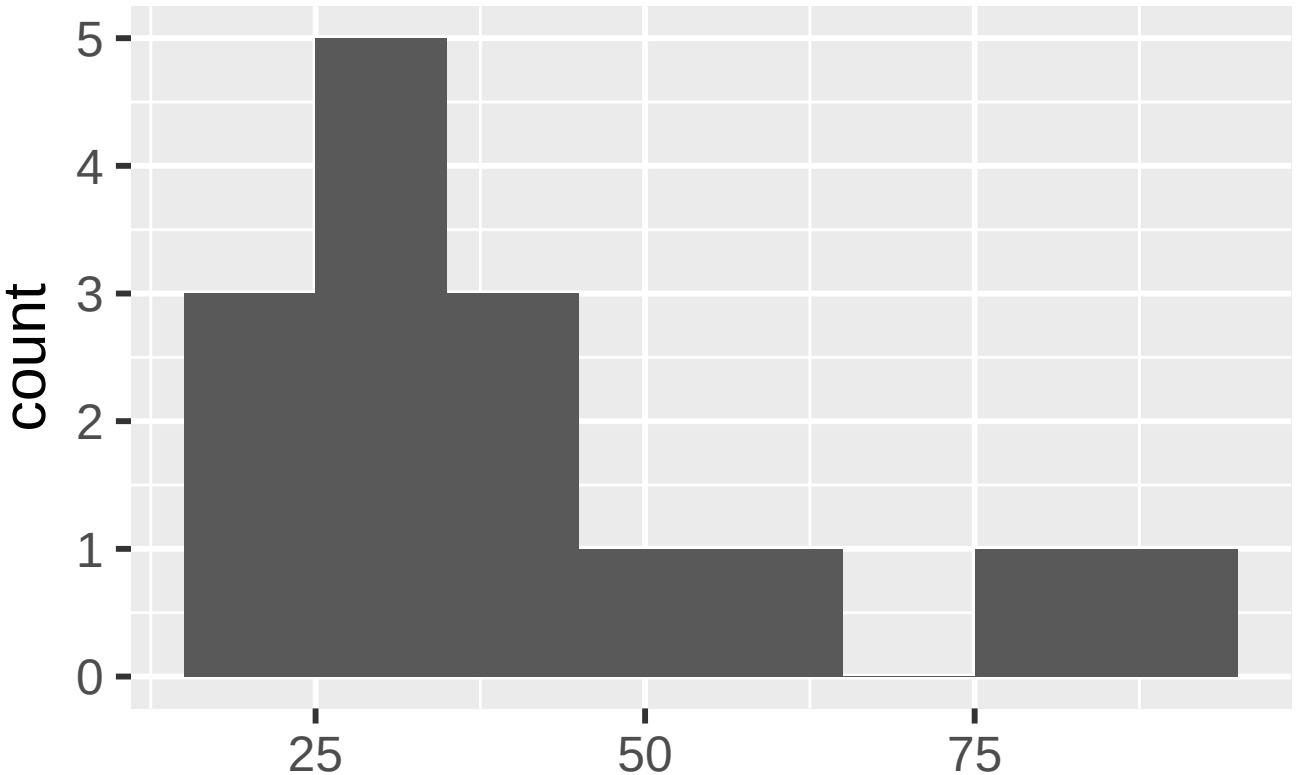
```
with(Mungan,table(Gender,Cancer.Stage))
```

```
##          Cancer.Stage
## Gender          I    II   III    IV
##   Female   3926   402   356   852
##   Male    12418   995   883  1905
```

Example Histogram

- Data on the numbers of hysterectomies performed by 15 male Swiss doctors:
  20 25 25 27 28 31 33 34 36 37 44 50 59 85 86

```
hyst <- data.frame(numHyster = c(20,25,25,27,28,31,33,34,
                                 36,37,44,50,59,85,86))
library(ggplot2)
ggplot(hyst,aes(x=numHyster)) + geom_histogram(binwidth=10)
```



Software Notes: Data Frames

- We used the `data.frame()` function to create a data frame with a single variable `numHyster`.
  - Data frames are objects used to store datasets in R.
  - Typically a data frame consists of multiple variables, such as the `Mungan` data frame with variables `Gender` and `Cancer.Stage`.
  - Use `names()` to find the names of variables in a data frame:

```
names(Mungan)
```

```
## [1] "Gender"       "Cancer.Stage"
```

Software Notes: Add-on Packages

- The code chunk that draws the histogram loads an add-on package for R.
- R consists of a "base" distribution plus many add-on packages that contain useful functions.
  - For example, `ggplot2` is a package that contains the graphics function `ggplot()`.
- To use the functions in a package you must **first** load the package with `library()`.
  - For example, `library(ggplot2)` loads `ggplot2` and gives us access to `ggplot()`.
- If you don't load a package, R can't find its functions.
  - For example, if you haven't yet loaded `ggplot2` and you try to use `ggplot()` you will get an error message:

```
Error: could not find function "ggplot"
```

Software Notes: Installing Add-on Packages

- See the video https://www.youtube.com/watch?v=CtOSryChcGg for more on R packages.
- RStudio users (RStudio Desktop or RStudio Cloud) will need to install the packages they need.
  - RStudio Desktop users should consult the **R Packages** section of the getting started document at https://github.com/SFUStatgen/RforStat2/blob/master/Tutorials/GettingStarted/startR-RStudio.Rmd
  - RStudio Cloud users should consult step 5 of the getting started document at https://github.com/SFUStatgen/RforStat2/blob/master/Tutorials/GettingStarted/startRStudioCloud.Rmd
- Jupyter should have all the packages we will need for this course installed.
  - Thus, Jupyter users don't need to install packages (but **do** need to load them before using).

Software Notes: ggplot()

- ggplot2 is an add-on package for R that implements the graphics function ggplot().
  - We will use ggplot() throughout the course.
- The call was

```
ggplot(hyst,aes(x=numHyster)) + geom_histogram(binwidth=10)
```

- This specifies the dataset (hyst) and the "aesthetic", which is a list of variables to plot as different features of the graph.
  - This example is a histogram of numHyster. We specify that numHyster is the x-axis variable with x=numHyster.
  - The function geom_histogram() adds the histogram; it takes the bin width as an optional argument.

Summary Statistics (Chapter 3)

Centre: The mean

- The mean is the ordinary arithmetic average of the observations.
- Hysterectomy example data:

  20 25 25 27 28 31 33 34 36 37 44 50 59 85 86

  has mean

$$\frac{20 + 25 + \ldots + 86}{n} = 41.3$$

```
library(dplyr)
summarize(hyst,mean(numHyster))
```

```
##   mean(numHyster)
## 1        41.33333
```

Software Note

- `dplyr` is an add-on package for R that includes useful tools for manipulating datasets in R.
  - The `summarize()` function takes the dataset as its first argument, and the summaries to compute as additional arguments.
  - In this example we could have instead used `with(hyst,mean(numHyster))`, but we will eventually want to use `summarize()` together with other tools from `dplyr` to produce data summaries.

Centre: The Median

- The median is the "middle value" of the variable

20, 25, 25, 27, 28, 31, 33, **34**, 36, 37, 44, 50, 59, 85, 86

- The centre observation is the median, $M = 34$.

```
summarize(hyst,median(numHyster))
```

```
## Warning: package 'bindrcpp' was built under R version 3.4.4
```

```
##   median(numHyster)
## 1                34
```

Spread: The Standard Deviation (SD) and Variance

- The SD measures spread about the mean.
- The sample variance, s^2, can be viewed as the average of the squared deviations, and the sample SD, s, as its square root.
- Hysterectomy example: $s = 20.6$

```
summarize(hyst,sd(numHyster))
```

```
##   sd(numHyster)
## 1      20.60744
```

Spread: The Inter-Quartile Range (IQR)

- The first and third quartiles mark the first and third quarters of the observations
  - These are also called the 25th and 75th percentile

```
summarize(hyst,
          Q1=quantile(numHyster,probs=.25),
          Q3=quantile(numHyster,probs=.75))
```

```
##     Q1 Q3
## 1 27.5 47
```

- The middle half of the data lies between.
- The range of the middle half, or IQR, is 47-27.5=19.5.

```
summarize(hyst,IQR(numHyster))
```

```
##   IQR(numHyster)
## 1           19.5
```

Boxplots

- The five number summary is the minimum, maximum, median, 1st and 3rd quartiles.
- Graphed with a boxplot:

```
with(hyst,boxplot(numHyster))
```