itns Chapter 03a

Peter Baumgartner

2019-05-27

Contents


```
### setting up working environment
### for details see: https://yihui.name/knitr/options/
knitr::opts_chunk$set(
        echo = T,
        message = T,
        error = T,
        warning = T,
        comment = '##',
        highlight = T,
        prompt = T,
        strip.white = T,
        tidy = T
        )
```

```
> ### accompanying R package: https://github.com/gitrman/itns
> if (!require("itns")) {
+     remotes::install_github("gitrman/itns", build = TRUE, build_opts = c("--no-resave-data",
+         "--no-manual"))
+     library("itns")
+ }
```

```
## Loading required package: itns
```

```
> ### https://www.tidyverse.org/
> if (!require("tidyverse")) {
+     install.packages("tidyverse", repos = "http://cran.wu.ac.at/")
+     library(tidyverse)
+ }
```

```
## Loading required package: tidyverse
```

```
## Registered S3 methods overwritten by 'ggplot2':
##   method         from
##   [.quosures     rlang
##   c.quosures     rlang
##   print.quosures rlang
```

```
## -- Attaching packages ---------------------------------------------------------------------------- t
```

```
## v ggplot2 3.1.1     v purrr   0.3.2
```

```
## v tibble  2.1.1     v dplyr   0.8.1
## v tidyr   0.8.3     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.4.0
```

```
## -- Conflicts ------------------------------------------------------------------------------- tidyvers
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
> ### above command installed and loaded the core tidyverse packages: ggplot2:
> ### data visualisation tibble: a modern take on data frames tidyr: data
> ### tidying readr: data import (csv, tsv, fwf) purrr: functional R programming
> ### dplyr: data (frame) manipulation stringr: string manipulation forcats:
> ### working with categorial varialbes
>
> # ### to calculate mode: if (!require('modeest'))
> # {install.packages('modeest', repos = 'http://cran.wu.ac.at/')
> # library(modeest)} # I am going to use the `janitor` package for
> # calculating table totals if (!require('janitor'))
> # {install.packages('janitor', repos = 'http://cran.wu.ac.at/')
> # library(janitor)}
```

1.3 Theme adaption for the graphic display with ggplot2

```
> my_theme <- theme_light() + theme(plot.title = element_text(size = 10, face = "bold",
+     hjust = 0.5))
> theme(plot.background = element_rect(color = NA, fill = NA)) + theme(plot.margin = margin(1,
+     0, 0, 0, unit = "cm"))
```

```
## List of 2
##  $ plot.background:List of 5
##   ..$ fill         : logi NA
##   ..$ colour       : logi NA
##   ..$ size         : NULL
##   ..$ linetype     : NULL
##   ..$ inherit.blank: logi FALSE
##   ..- attr(*, "class")= chr [1:2] "element_rect" "element"
##  $ plot.margin    : 'margin' num [1:4] 1cm 0cm 0cm 0cm
##   ..- attr(*, "valid.unit")= int 1
##   ..- attr(*, "unit")= chr "cm"
##  - attr(*, "class")= chr [1:2] "theme" "gg"
##  - attr(*, "complete")= logi FALSE
##  - attr(*, "validate")= logi TRUE
```

2 Following the Book Text

2.1 Get Data

We are going to use the transcription data of the laptop group from the pen_laptop1 data set which is included in the `itns`-package. This package is already installed and loaded. At first we need to get an idea of the structure of this data set.

```
> str(pen_laptop1)
```

```
## 'data.frame':    65 obs. of  2 variables:
##  $ group        : Factor w/ 2 levels "Laptop","Pen": 2 2 2 2 2 2 2 2 2 2 ...
```

```
##  $ transcription: num  12.1 6.5 8.1 7.6 12.2 10.8 1 2.9 14.4 8.4 ...
```

We see that we have 65 observations with 2 variables. With RStudio you can get a more inuitive look at the data with the command `View(pen_laptop1)`. You will get additional information when you hover with your cursor over the column names.

We only need the data from the group "Laptop". To do this we have different possibilities:

- **Subsetting operators:** This is very powerful and fast feature to express complex operations in R. Subsetting is a natural complement of the `str()` command, we have been used above.
- **Subset function:** Another possibility is the subset function which returns a subsets of vectors which meet certain conditions.
- **Using dplyr:** `dplyr` (speak "dipleyer") is a very important and popular package for data manipulation. It is included in the `tidyverse` package collection, which we already have installed and loaded. It is more intuitive than subsetting but it is not basic R and needs therefore to install and to load an extra package.

```
> laptop1_1 <- pen_laptop1[which(pen_laptop1$group == "Laptop"), ]
> ## is the same as:
> laptop1_2 <- subset(pen_laptop1, group == "Laptop")
> laptop1_3 <- dplyr::filter(pen_laptop1, group == "Laptop")
```

2.2 Dot Plots and Frequency Histograms

As far as I know there is no possibility to generate a simple dot plot as in Panel on page 46 of the book. The nearest solution are short lines (instead of dots) generated with `geom_rug` to visualize where each data point is on the x axis.

```
> plot <- ggplot(laptop1_1, aes(x = transcription))  # we will use this line for the following 4 plots
> plot + geom_dotplot()
```

```
## `stat_bindot()` using `bins = 30`. Pick better value with `binwidth`.
```