

Biostatistics & Epidemiological Data Analysis using R

3

Advanced tables & plots

Stefan Konigorski

Health Intervention Analytics Group, HPI

November 11, 2021

Learning objectives

Make

- Advanced tables
- Advanced plots

in R.

Tables

Table 1 with table1::table1()

Nice Table 1's can be easily generated using the `table1::table1()` function.

For example

```
table1::table1(~ Pregnancies + Age + BMI +  
BloodPressure + Glucose | Diabetes, data =  
Pima_diabetes)
```

yields:

Table 1 with `table1::table1()`

	no (n=500)	yes (n=268)	Overall (n=768)
Pregnancies			
Mean (SD)	3.30 (3.02)	4.87 (3.74)	3.85 (3.37)
Median [Min, Max]	2.00 [0.00, 13.0]	4.00 [0.00, 17.0]	3.00 [0.00, 17.0]
Age			
Mean (SD)	31.2 (11.7)	37.1 (11.0)	33.2 (11.8)
Median [Min, Max]	27.0 [21.0, 81.0]	36.0 [21.0, 70.0]	29.0 [21.0, 81.0]
BMI			
Mean (SD)	30.3 (7.69)	35.1 (7.26)	32.0 (7.88)
Median [Min, Max]	30.1 [0.00, 57.3]	34.3 [0.00, 67.1]	32.0 [0.00, 67.1]
BloodPressure			
Mean (SD)	68.2 (18.1)	70.8 (21.5)	69.1 (19.4)
Median [Min, Max]	70.0 [0.00, 122]	74.0 [0.00, 114]	72.0 [0.00, 122]
Glucose			
Mean (SD)	110 (26.1)	141 (31.9)	121 (32.0)
Median [Min, Max]	107 [0.00, 197]	140 [0.00, 199]	117 [0.00, 199]

Comparison of `table1()` and `summary_table()`

There also exist many other functions to create a Table 1, e.g. the `qwraps2::summary_table()` function.

Comparison of `table1()` and `summary_table()`

There also exist many other functions to create a Table 1, e.g. the `qwraps2::summary_table()` function.

Comparison

- `table1::table1()` only allows knitting to html.
- To make a table using `table1::table1()` is generally easier than using `qwraps2::summary_table()`.
- Custom-specification (i.e. to include IQR instead of Min, Max) is easier in `qwraps2::summary_table()`.

Table 1 with `qwraps2::summary_table()`

Overview

- The `summary_table()` function in the `qwraps2` package is a user-friendly wrapper to generate tables with some customization possibilities.
- Vignette with many examples:
<https://cran.r-project.org/web/packages/qwraps2/vignettes/summary-statistics.html>

Table 1 with `qwraps2::summary_table()`

Use in R Markdown

- In header of the R chunk, include `results = "asis"` - otherwise, the table will not be printed as a table but as the commands underlying the formatting of the table:

```
```{r, results = "asis"}  
```
```

- In the R chunk, after loading the `qwraps2` package, generate the table in R Markdown format with

```
> options(qwraps2_markup = "markdown")
```

Table 1 with `qwraps2::summary_table()`

How to use the `summary_table()` function

- 1 Create a structured list, which captures for each variable in the table, which descriptive statistics should be calculated and printed in the table.
- 2 In the list, the function names for computing the descriptive statistics have to be provided. Using the available functions for frequencies, mean & SD, and median & IQR is the most convenient.
- 3 Use the `summary_table()` function with this list and the dataset of interest as arguments to compute the desired table.

Table 1 with `qwraps2::summary_table()`

Example 1

Preparation: Load Pima diabetes dataset "Pima_diabetes" and transform Diabetes variable to factor with levels "yes", "no".

```
• > library(qwraps2)
• > options(qwraps2_markup = "markdown")
• > our_summary <-
+   list("Diabetes" =
+     list("yes" = ~ qwraps2::n_perc0(Diabetes == "yes"),
+       "no" = ~ qwraps2::n_perc0(Diabetes == "no")),
+     "Age" =
+     list("Median (IQR)" = ~ median_iqr(Age)),
+     "BMI" =
+     list("Mean (SD)" = ~ qwraps2::mean_sd(BMI))
+   )
• > summary_table(Pima_diabetes, our_summary)
```

Table 1 with `qwraps2::summary_table()`

Example 1

This produces the following table:

| Pima_diabetes (N = 768) | |
|-------------------------|----------------------|
| Diabetes | |
| yes | 268 (35) |
| no | 500 (65) |
| Age | |
| Median (IQR) | 29.00 (24.00, 41.00) |
| BMI | |
| Mean (SD) | 31.99 \pm 7.88 |

See exercise 1 in `R_3b_exercises.Rmd` for more details and options.

Table 1 with `qwraps2::summary_table()`

Stratified tables

- In order to stratify the descriptive statistics in a table by a variable `groupvariable`, use the `dplyr::group_by()` function:
- `summary_table(dplyr::group_by(Pima_diabetes, groupvariable), our_summary)`

Table 1 with `qwraps2::summary_table()`

Stratified tables

This allows to create tables such as:

| Summary Statistics | Overall (N = 768) | Diabetes (N = 500) | No Diabetes (N = 268) |
|------------------------------|----------------------|----------------------|-----------------------|
| Number of pregnancies | | | |
| 0-1 | 111 (14%) | 73 (15%) | 38 (14%) |
| 2-3 | 178 (23%) | 132 (26%) | 46 (17%) |
| 4-6 | 175 (23%) | 115 (23%) | 60 (22%) |
| 7-17 | 169 (22%) | 74 (15%) | 95 (35%) |
| Age | | | |
| Median (IQR) | 29.00 (24.00, 41.00) | 27.00 (23.00, 37.00) | 36.00 (28.00, 44.00) |
| BMI | | | |
| Mean (SD) | 31.99 (7.88) | 30.30 (7.69) | 35.14 (7.26) |
| Blood Pressure | | | |
| Mean (SD) | 69.11 (19.36) | 68.18 (18.06) | 70.82 (21.49) |
| Glucose | | | |
| Mean (SD) | 120.89 (31.97) | 109.98 (26.14) | 141.26 (31.94) |

Alternative packages to produce tables

- `arsenal::tableby()` function
- `xtables::xtable()` function
- Functions in `tableone` package
- many many others, e.g. also check out the functions in the `kableExtra` package

Exercise 2

Generate tables using the `summary_table()` function:

- Add descriptive statistics for the Insulin variable to the table generated in exercise 1.
- Generate a table including descriptive statistics of diabetes prevalence, age, BMI, blood pressure, glucose levels stratified by number of pregnancy quartiles.
- See exercise 2 in `R_3b_exercises.Rmd`.

Generate nice plots using `ggplot2`

ggplot2 - Overview

- "The" package to produce nice graphics in R.
- Allows to generate many different plots and customize all aspects of the plot.
- Online reference: e.g.
<https://ggplot2.tidyverse.org/reference/>.
- Book: <https://github.com/hadley/ggplot2-book>.
- Cheatsheets:
<https://github.com/rstudio/cheatsheets/blob/master/data-visualization-2.1.pdf>

ggplot2 - Overview

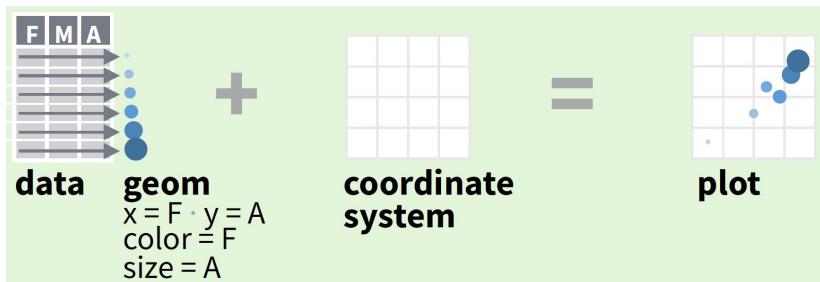
Based on grammar of graphics: any graph can be built from

- a dataset
- a coordinate system
- and a visual representation of the data (in the coordinate system), i.e. a mapping of data to elements in the plot:

ggplot2 - Overview

Based on grammar of graphics: any graph can be built from

- a dataset
- a coordinate system
- and a visual representation of the data (in the coordinate system), i.e. a mapping of data to elements in the plot:



(from <https://github.com/rstudio/cheatsheets/blob/master/data-visualization-2.1.pdf>)

ggplot2 - Structure

- The basis of all plots in ggplot2 is the `ggplot2::ggplot()` function, which initializes a `ggplot` object.
- In this initialization, the data, variables to be used in the plot and in all layers, and the mapping are specified.
- Then, different layers can be specified, which determine which type(s) of plot(s) will be generated, in which coordinate system, if they will be stratified, and how the axes/graphic elements and further settings should be fine-formatted.

ggplot2 - Components

```

ggplot (data = <DATA>) +
<GEOM_FUNCTION> (mapping = aes(<MAPPINGS>),
stat = <STAT>, position = <POSITION>) +
<COORDINATE_FUNCTION> +
<FACET_FUNCTION> +
<SCALE_FUNCTION> +
<THEME_FUNCTION>

```

required

Not required, sensible defaults supplied

(from <https://github.com/rstudio/cheatsheets/blob/master/data-visualization-2.1.pdf>)

ggplot2 - Components

- **Data:** dataset with variables of interest
- **Aesthetics:** x, y, colour, size, shape, ...
- **Geom(etrie)s:** point, line, bar, boxplot, ...
- **Facets:** stratification in panels in rows/columns
- **Statistics:** Transformation of variables: binning, descriptive ...
- **Coordinates:** Cartesian ...
- **Themes:** Fine-formatting

ggplot2 - Components

- **Data:** dataset with variables of interest
- **Aesthetics:** x, y, colour, size, shape, ...
- **Geom(etrie)s:** point, line, bar, boxplot, ...
- **Facets:** stratification in panels in rows/columns
- **Statistics:** Transformation of variables: binning, descriptive ...
- **Coordinates:** Cartesian ...
- **Themes:** Fine-formatting

See examples in the following and exercise 3 in `R_3b_exercises.Rmd`.

ggplot2 - Examples

Bar plot

```
> library(ggplot2)

> ggplot(data = Pima_diabetes, mapping = aes(x =
+ Pregnancies)) + geom_bar()

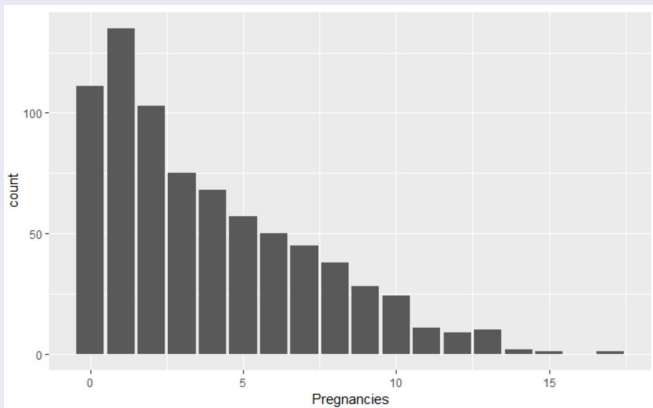
> # in short:

> ggplot(Pima_diabetes, aes(Pregnancies)) + geom_bar()
```

ggplot2 - Examples

Bar plot

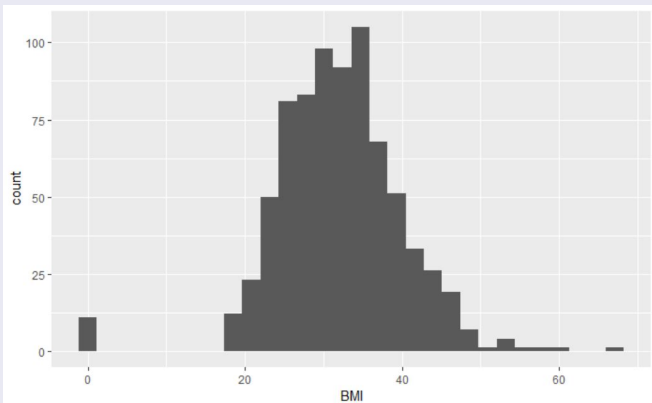
```
> ggplot(Pima_diabetes, aes(Pregnancies)) + geom_bar()
```



ggplot2 - Examples

Histogram

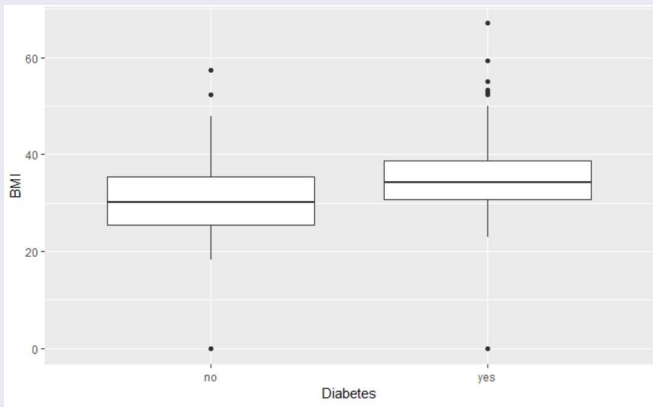
```
> ggplot(Pima_diabetes, aes(BMI)) + geom_histogram()
```



ggplot2 - Examples

Stratified boxplots

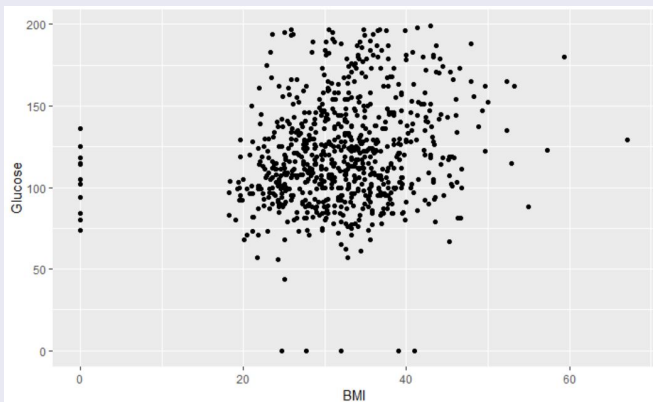
```
> ggplot(Pima_diabetes, aes(Diabetes, BMI)) +  
+ geom_boxplot()
```



ggplot2 - Examples

Scatter plot

```
> ggplot(Pima_diabetes, aes(BMI, Glucose)) +  
+ geom_point()
```



ggplot2 - Examples

Scatter plot using colours for third variable

```
> ggplot(Pima_diabetes, aes(BMI, Glucose, col =  
+ Diabetes)) + geom_point()
```



Exercise 3b

- Produce a boxplot of BMI for each group of number of pregnancies.
- In the scatterplot of BMI and glucose levels, add a regression line which describes the linear association between the two variables.
- See exercise 3b in `R_3b_exercises.Rmd`.

Glimpse at the many more graphics options in R

- Use the patchwork package to stack ggplots:
<https://cran.r-project.org/web/packages/patchwork/vignettes/patchwork.html>
- Check out the ggpubr package to generate publication ready plots: <http://www.sthda.com/english/articles/24-ggpubr-publication-ready-plots/>
- Create maps and visualize your geospatial data in maps using the leaflet package.
- Create interactive ggplot plots using the ggplotly() function in the plotly package.
- See examples in exercise 4 in `R_3b_exercises.Rmd`.

Questions?