

RMarkdown Tables - Data Summary and Presentation

Melinda Higgins

1/23/2024

Why you want Tables

While you can create simple tables with the `table()` function in base R, most of the time you will want to present your results in some kind of table format. This could be for any of the following:

- viewing your data in a table format
- presenting summary statistics of the variables in your dataset
- presenting your models or analysis results in a table format
- and even more...

Get Inspiration

The underlying formatting for making appealing and well organized tables can be sort of an art-form. Getting the code to work along with the formatting for various final formats (like HTML, PDF, DOC, PPT, etc) can be extremely challenging. However, the good news is that this has recently been a hot area of rapid development in the R/RMarkdown world.

In fact, in the past few years there have been contests on the best tables and associated packages and codes for these projects. See:

- [Winners of the 2022 RStudio Tables Contest](#)
- [Winners of the 2021 RStudio Tables contest](#)
- [Winners of the 2020 RStudio Tables contest](#)

Let's try a simple table to get started

Here is an example of basic output to view the “top” of the builtin `mtcars` dataset, using this code: `head(mtcars)` .

```
head(mtcars)
```

##	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
## Valiant	18.1	6	225	105	2.76	3.460	20.22	1	0	3	1

OK, so this is just text on the page - not really a nice table.

To make this a table, let's use the `kable()` function from the `knitr` package. To set this up, we'll also use the `dplyr` package to use the `%>%` pipe coding approach.

```
library(knitr)
library(dplyr)
mtcars %>%
  head() %>%
  knitr::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
Valiant	18.1	6	225	105	2.76	3.460	20.22	1	0	3	1

Let's add a caption for our table.

NOTE: The way the caption shows up will vary depending on whether you “knit” to HTML, DOCX, PDF or other formats...

```
mtcars %>%
  head() %>%
  knitr::kable(caption = "Top 6 rows of the mtcars dataset")
```

Table 2: Top 6 rows of the mtcars dataset

	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
Valiant	18.1	6	225	105	2.76	3.460	20.22	1	0	3	1

Try customization with the `gt` package

You can add headers, footers and more with the `gt` package. See <https://gt.rstudio.com/index.html>.

```
library(gt)
mtcars %>%
  head() %>%
  gt()
```

mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb
-----	-----	------	----	------	----	------	----	----	------	------

21.0	6	160	110	3.90	2.620	16.46	0	1	4	4
21.0	6	160	110	3.90	2.875	17.02	0	1	4	4
22.8	4	108	93	3.85	2.320	18.61	1	1	4	1
21.4	6	258	110	3.08	3.215	19.44	1	0	3	1
18.7	8	360	175	3.15	3.440	17.02	0	0	3	2
18.1	6	225	105	2.76	3.460	20.22	1	0	3	1

Add a header.

```
mtcars %>%
  head() %>%
  gt() %>%
  tab_header(
    title = "The mtcars dataset",
    subtitle = "The top 6 rows are presented"
  )
```

The mtcars dataset
The top 6 rows are presented

mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb
21.0	6	160	110	3.90	2.620	16.46	0	1	4	4
21.0	6	160	110	3.90	2.875	17.02	0	1	4	4
22.8	4	108	93	3.85	2.320	18.61	1	1	4	1
21.4	6	258	110	3.08	3.215	19.44	1	0	3	1
18.7	8	360	175	3.15	3.440	17.02	0	0	3	2
18.1	6	225	105	2.76	3.460	20.22	1	0	3	1

Add a footer.

```
mtcars %>%
  head() %>%
  gt() %>%
  tab_header(
    title = "The mtcars dataset",
    subtitle = "The top 6 rows are presented"
  ) %>%
  tab_source_note(
    source_note = "The data was extracted from the 1974 Motor Trend US magazine, and comprises fuel con."
  )
```

The mtcars dataset
The top 6 rows are presented

mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb
21.0	6	160	110	3.90	2.620	16.46	0	1	4	4
21.0	6	160	110	3.90	2.875	17.02	0	1	4	4
22.8	4	108	93	3.85	2.320	18.61	1	1	4	1
21.4	6	258	110	3.08	3.215	19.44	1	0	3	1

18.7	8	360	175	3.15	3.440	17.02	0	0	3	2
18.1	6	225	105	2.76	3.460	20.22	1	0	3	1

The data was extracted from the 1974 Motor Trend US magazine, and comprises fuel consumption and 10 aspects of automobile design and performance for 32 automobiles (1973–74 models).

What about summary statistics?

A really simple approach is to use the `summary()` function in case R. But the results, while useful, is less than inspiring.

```
mtcars %>%
  summary() %>%
  knitr::kable()
```

mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb
Min.	Min.	Min. :	Min. :	Min.	Min.	Min.	Min.	Min.	Min.	Min.
:10.40	:4.000	71.1	52.0	:2.760	:1.513	:14.50	:0.0000	:0.0000	:3.000	:1.000
1st	1st	1st	1st	1st	1st	1st	1st	1st	1st	1st
Qu.:15.43	Qu.:4.000	Qu.:120.8	Qu.:96.5	Qu.:3.080	Qu.:2.581	Qu.:16.89	Qu.:0.0000	Qu.:0.0000	Qu.:3.000	Qu.:2.000
Median	Median	Median	Median	Median	Median	Median	Median	Median	Median	Median
:19.20	:6.000	:196.3	:123.0	:3.695	:3.325	:17.71	:0.0000	:0.0000	:4.000	:2.000
Mean	Mean	Mean	Mean	Mean	Mean	Mean	Mean	Mean	Mean	Mean
:20.09	:6.188	:230.7	:146.7	:3.597	:3.217	:17.85	:0.4375	:0.4062	:3.688	:2.812
3rd	3rd	3rd	3rd	3rd	3rd	3rd	3rd	3rd	3rd	3rd
Qu.:22.80	Qu.:8.000	Qu.:326.0	Qu.:180.0	Qu.:3.920	Qu.:3.610	Qu.:18.90	Qu.:1.0000	Qu.:1.0000	Qu.:4.000	Qu.:4.000
Max.	Max.	Max.	Max.	Max.	Max.	Max.	Max.	Max.	Max.	Max.
:33.90	:8.000	:472.0	:335.0	:4.930	:5.424	:22.90	:1.0000	:1.0000	:5.000	:8.000

Try the gtsummary package

- Learn more about the `gtsummary` package at: <https://www.danielsjoberg.com/gtsummary/index.html>
- Inspiration Gallery, <https://www.danielsjoberg.com/gtsummary/articles/gallery.html>.

```
library(gtsummary)
mtcars %>%
  tbl_summary()
```

Characteristic	N = 32
mpg	19.2 (15.4, 22.8)
cyl	
4	11 (34%)
6	7 (22%)
8	14 (44%)
disp	196 (121, 326)
hp	123 (97, 180)

Characteristic	N = 32
drat	3.70 (3.08, 3.92)
wt	3.33 (2.58, 3.61)
qsec	17.71 (16.89, 18.90)
vs	14 (44%)
am	13 (41%)
gear	
3	15 (47%)
4	12 (38%)
5	5 (16%)
carb	
1	7 (22%)
2	10 (31%)
3	3 (9.4%)
4	10 (31%)
6	1 (3.1%)
8	1 (3.1%)

Look at statistics by group.

```
mtcars %>%
  tbl_summary(by = cyl)
```

Characteristic	4, N = 11	6, N = 7	8, N = 14
mpg	26.0 (22.8, 30.4)	19.7 (18.7, 21.0)	15.2 (14.4, 16.3)
disp	108 (79, 121)	168 (160, 196)	351 (302, 390)
hp	91 (66, 96)	110 (110, 123)	193 (176, 241)
drat	4.08 (3.81, 4.17)	3.90 (3.35, 3.91)	3.12 (3.07, 3.23)
wt	2.20 (1.89, 2.62)	3.22 (2.82, 3.44)	3.76 (3.53, 4.01)
qsec	18.90 (18.56, 19.95)	18.30 (16.74, 19.17)	17.18 (16.10, 17.56)
vs	10 (91%)	4 (57%)	0 (0%)
am	8 (73%)	3 (43%)	2 (14%)
gear			
3	1 (9.1%)	2 (29%)	12 (86%)
4	8 (73%)	4 (57%)	0 (0%)
5	2 (18%)	1 (14%)	2 (14%)
carb			
1	5 (45%)	2 (29%)	0 (0%)
2	6 (55%)	0 (0%)	4 (29%)
3	0 (0%)	0 (0%)	3 (21%)
4	0 (0%)	4 (57%)	6 (43%)
6	0 (0%)	1 (14%)	0 (0%)
8	0 (0%)	0 (0%)	1 (7.1%)

Add statistical comparison tests.

```
mtcars %>%
  tbl_summary(by = cyl) %>%
  add_p()
```

Characteristic	4, N = 11	6, N = 7	8, N = 14	p-value
mpg	26.0 (22.8, 30.4)	19.7 (18.7, 21.0)	15.2 (14.4, 16.3)	<0.001
disp	108 (79, 121)	168 (160, 196)	351 (302, 390)	<0.001
hp	91 (66, 96)	110 (110, 123)	193 (176, 241)	<0.001
drat	4.08 (3.81, 4.17)	3.90 (3.35, 3.91)	3.12 (3.07, 3.23)	<0.001
wt	2.20 (1.89, 2.62)	3.22 (2.82, 3.44)	3.76 (3.53, 4.01)	<0.001
qsec	18.90 (18.56, 19.95)	18.30 (16.74, 19.17)	17.18 (16.10, 17.56)	0.006
vs	10 (91%)	4 (57%)	0 (0%)	<0.001
am	8 (73%)	3 (43%)	2 (14%)	0.009
gear				<0.001
3	1 (9.1%)	2 (29%)	12 (86%)	
4	8 (73%)	4 (57%)	0 (0%)	
5	2 (18%)	1 (14%)	2 (14%)	
carb				<0.001
1	5 (45%)	2 (29%)	0 (0%)	
2	6 (55%)	0 (0%)	4 (29%)	
3	0 (0%)	0 (0%)	3 (21%)	
4	0 (0%)	4 (57%)	6 (43%)	
6	0 (0%)	1 (14%)	0 (0%)	
8	0 (0%)	0 (0%)	1 (7.1%)	

Also try the arsenal package

Learn more about the **arsenal** package:

- <https://mayoverse.github.io/arsenal/>
- and the `tableby()` function <https://mayoverse.github.io/arsenal/articles/tableby.html>

This time, let's look at the **penguins** dataset from the **palmerpenguins** package.

We'll use the `tableby()` function from the **arsenal** package to get some summary stats.

NOTE: IMPORTANT - when using the **arsenal** package, you need to add `results = "asis"` in your `r-chunk` options so that the table looks correct when you “knit” your Rmarkdown file.

```
library(palmerpenguins)
library(arsenal)

tab1 <- tableby(~ bill_length_mm + bill_depth_mm +
  flipper_length_mm + body_mass_g,
  data = penguins)
summary(tab1)
```

	Overall (N=344)
bill_length_mm	
N-Miss	2
Mean (SD)	43.922 (5.460)
Range	32.100 - 59.600
bill_depth_mm	
N-Miss	2
Mean (SD)	17.151 (1.975)

	Overall (N=344)
Range	13.100 - 21.500
flipper_length_mm	
N-Miss	2
Mean (SD)	200.915 (14.062)
Range	172.000 - 231.000
body_mass_g	
N-Miss	2
Mean (SD)	4201.754 (801.955)
Range	2700.000 - 6300.000

We can also get comparison statistics by group with associated statistical tests. Let's look at these summary stats by the 3 species of penguins.

```
tab1 <- tableby(species ~ bill_length_mm + bill_depth_mm +
  flipper_length_mm + body_mass_g,
  data = penguins)
summary(tab1)
```

	Adelie (N=152)	Chinstrap (N=68)	Gentoo (N=124)	Total (N=344)	p value
bill_length_mm					< 0.001
N-Miss	1	0	1	2	
Mean (SD)	38.791 (2.663)	48.834 (3.339)	47.505 (3.082)	43.922 (5.460)	
Range	32.100 - 46.000	40.900 - 58.000	40.900 - 59.600	32.100 - 59.600	
bill_depth_mm					< 0.001
N-Miss	1	0	1	2	
Mean (SD)	18.346 (1.217)	18.421 (1.135)	14.982 (0.981)	17.151 (1.975)	
Range	15.500 - 21.500	16.400 - 20.800	13.100 - 17.300	13.100 - 21.500	
flipper_length_mm					< 0.001
N-Miss	1	0	1	2	
Mean (SD)	189.954 (6.539)	195.824 (7.132)	217.187 (6.485)	200.915 (14.062)	
Range	172.000 - 210.000	178.000 - 212.000	203.000 - 231.000	172.000 - 231.000	
body_mass_g					< 0.001
N-Miss	1	0	1	2	
Mean (SD)	3700.662 (458.566)	3733.088 (384.335)	5076.016 (504.116)	4201.754 (801.955)	
Range	2850.000 - 4775.000	2700.000 - 4800.000	3950.000 - 6300.000	2700.000 - 6300.000	

Another COOL package, summarytools

Another really cool package that is useful for getting a quick summary of what is in your dataset along with some quick summary stats and tiny charts.

Learn more at:

- <https://cran.r-project.org/web/packages/summarytools/>
- <https://cran.r-project.org/web/packages/summarytools/vignettes/introduction.html>

Let's look at the `penguins` dataset again.

And like the `arsenal` package, when we use the `summarytools` package, you need to add `results = "asis"` to the r-chunk options.

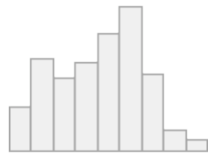
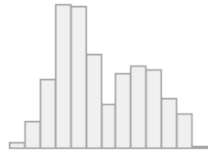
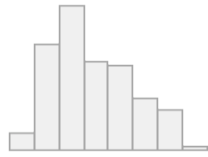


```
library(summarytools)
dfSummary(penguins,
  plain.ascii = FALSE,
  style       = "grid",
  graph.magnif = 0.75,
  valid.col   = FALSE,
  tmp.img.dir = "/tmp")
```

Data Frame Summary

penguins Dimensions: 344 x 8

Duplicates: 0

No	Variable	Stats / Values	Freqs (% of Valid)	Graph	Missing
1	species [factor]	1. Adelie 2. Chinstrap 3. Gentoo	152 (44.2%) 68 (19.8%) 124 (36.0%)		0 (0.0%)
2	island [factor]	1. Biscoe 2. Dream 3. Torgersen	168 (48.8%) 124 (36.0%) 52 (15.1%)		0 (0.0%)
3	bill_length_mm [numeric]	Mean (sd) : 43.9 (5.5) min < med < max: 32.1 < 44.5 < 59.6 IQR (CV) : 9.3 (0.1)	164 distinct values		2 (0.6%)

No	Variable	Stats / Values	Freqs (% of Valid)	Graph	Missing
4	bill_depth_mm [numeric]	Mean (sd) : 17.2 (2) min < med < max: 13.1 < 17.3 < 21.5 IQR (CV) : 3.1 (0.1)	80 distinct values		2 (0.6%)
5	flipper_length_mm [integer]	Mean (sd) : 200.9 (14.1) min < med < max: 172 < 197 < 231 IQR (CV) : 23 (0.1)	55 distinct values		2 (0.6%)
6	body_mass_g [integer]	Mean (sd) : 4201.8 (802) min < med < max: 2700 < 4050 < 6300 IQR (CV) : 1200 (0.2)	94 distinct values		2 (0.6%)
7	sex [factor]	1. female 2. male	165 (49.5%) 168 (50.5%)		11 (3.2%)
8	year [integer]	Mean (sd) : 2008 (0.8) min < med < max: 2007 < 2008 < 2009 IQR (CV) : 2 (0)	2007 : 110 (32.0%) 2008 : 114 (33.1%) 2009 : 120 (34.9%)		0 (0.0%)

summarytools::ctable()

Get a nice crosstable for 2 categorical variables using `ctable()` function. Let's look at species and sex in the penguins dataset.

NOTE: At the moment `ctable()` will only work for HTML output. This does not work for DOC or PDF formats.

```
library(magrittr)
penguins %$% # Acts like with(penguins, ...)
  ctable(x = species, y = sex,
         useNA = "no",
         chisq = TRUE,
         OR     = TRUE,
         RR     = TRUE,
         headings = FALSE) %>%
  print(method = "render")
```

More fun packages to try out

These can all be fun to play with but with “great power comes great responsibility” - the key is looking for examples to adapt and reading the documentation.

For all of these getting the formatting to work across multiple output formats is really challenging. Typically, the developers get HTML and/or PDF (through LaTeX) working first and MS WORD DOCX formats are the hardest to adapt. Although if all fails (sometimes) you can simply cut and paste HTML output over into a WORD document - see **kableExtra** short video http://haozhu233.github.io/kableExtra/kableExtra_and_word.html.

- **reactablefmtr** <https://kcuilla.github.io/reactablefmtr/index.html>
- **gtExtras**
 - <https://jthomasmock.github.io/gtExtras/index.html>
 - <https://themockup.blog/posts/2022-06-13-gtextras-cran/>
- **flextable** <https://ardata-fr.github.io/flextable-book/> and gallery examples at <https://ardata-fr.github.io/flextable-gallery/gallery/>
- **kableExtra** for added functionality for **knitr::kable()**, see <https://cran.r-project.org/web/packages/kableExtra/>

More links:

- <https://towardsdatascience.com/top-7-packages-for-making-beautiful-tables-in-r-7683d054e541>
- Tables in Clinical Trials with R - online book
- Tables for presentation - Ch29 in the The Epidemiologist R Handbook - online book