Data Frame Summaries in PDF's

Dominic Comtois

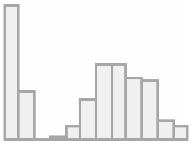
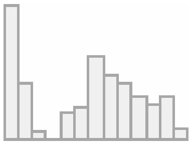
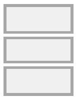
2021-07-30

Here are the instructions for setting up *R Markdown* documents in order to generate *pdf* documents with data frame summaries (`summarytools::dfSummary()`) that contain images.

1. The Graphics Alignment Problem

Although generating *html* or *Word* documents from *Rmd*'s containing `dfSummary()` outputs is a smooth and painless process, there is a major problem when it comes to generating *pdf*'s. The graphs, instead of being vertically centered, appear as though they were sitting on top of all the other cells' content:

```
dfSummary(iris[3:5], headings = FALSE)
```

No	Variable	Stats / Values	Freqs (% of Valid)	Graph	Missing
1	Petal.Length [numeric]	Mean (sd) : 3.8 (1.8) min < med < max: 1 < 4.3 < 6.9 IQR (CV) : 3.5 (0.5)	43 distinct values		0 (0.0%)
2	Petal.Width [numeric]	Mean (sd) : 1.2 (0.8) min < med < max: 0.1 < 1.3 < 2.5 IQR (CV) : 1.5 (0.6)	22 distinct values		0 (0.0%)
3	Species [factor]	1. setosa 2. versicolor 3. virginica	50 (33.3%) 50 (33.3%) 50 (33.3%)		0 (0.0%)

2. The Solution

To correct this issue, we need to redefine the `\includegraphics` command. This can be done in multiple ways, but the simplest is to include in your document's header a *tex* file which is designed to do just that. It can be achieved by configuring the YAML section as follows.

2.1 The YAML Section

```
---
title: "My Own Private PDF"
output:
  pdf_document:
    highlight: tango
    latex_engine: xelatex
    includes:
      in_header:
      - !expr system.file("includes/fig-valign.tex",
                          package = "summarytools")
papersize: letter
---
```

The solution presented here requires that some *tex* code be included in the YAML section of the Rmd document. You can use your own *tex* file, or use the one that is part of the package as of version 1.0.0 (July 2021). and include it in from the YAML section using `system.file()`.

The `latex_engine: xelatex` part is not mandatory for the solution to work. But there are several advantages to using it; I use it systematically and see only advantages to it, so I can only advise you do the same.

This solution is not perfect; if your *pdf* document relies on the use of `\includegraphics` in other sections, you might notice newly *mis*aligned images. Thankfully, there is a way to go around this (see section 2.3).

Using Your Own *tex* File

If you prefer including your own *tex* file, here is what it should (minimally) contain:

```
\usepackage{graphicx}
\usepackage[export]{adjustbox}
\usepackage{letltxmacro}
\LetLtxMacro{\OldIncludegraphics}{\includegraphics}
\renewcommand{\includegraphics}[2][]{\raisebox{0.5\height}%
  {\OldIncludegraphics[valign=t,#1]{#2}}}
```

The only impact on your YAML section will be the `in_header:` attribute which will need to point to this file, using an absolute or relative path. If the file is kept in the same directory as your *Rmd* document, you'll use `in_header: fig-align.tex` (supposing you use that file name).

2.2 Example

Here is a setup chunk, followed by a call to `dfSummary()`.

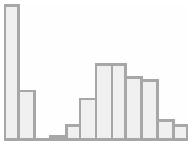
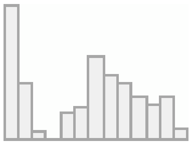
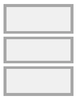
```
library(summarytools)
st_options(
  plain.ascii             = FALSE,
  subtitle.emphasis       = FALSE,
  style                   = "rmarkdown", # For other summarytools objects (freq, descr...)
  dfSummary.style         = "grid",
  dfSummary.graph.magnif  = .5,
  dfSummary.valid.col     = FALSE,
  tmp.img.dir             = "/tmp"  # Recommended on Linux/OS X; On
                                    # Windows, "img" is suggested
)
```

Now that the setup is done, we can generate the results.

```
define_keywords(title.dfSummary = "Data Frame Summary in PDF Document")
dfSummary(iris[3:5])
```

Data Frame Summary in PDF Document

iris
Dimensions: 150 x 3
Duplicates: 47

No	Variable	Stats / Values	Freqs (% of Valid)	Graph	Missing
1	Petal.Length [numeric]	Mean (sd) : 3.8 (1.8) min < med < max: 1 < 4.3 < 6.9 IQR (CV) : 3.5 (0.5)	43 distinct values		0 (0.0%)
2	Petal.Width [numeric]	Mean (sd) : 1.2 (0.8) min < med < max: 0.1 < 1.3 < 2.5 IQR (CV) : 1.5 (0.6)	22 distinct values		0 (0.0%)
3	Species [factor]	1. setosa 2. versicolor 3. virginica	50 (33.3%) 50 (33.3%) 50 (33.3%)		0 (0.0%)

3. A More Robust Solution

If redefining the `\includegraphics` command causes problems elsewhere in your document[1], following these instructions should take care of it (file names and location are suggestions only):

1. Split the contents of `fig-valign.tex` into two files in your *Rmd* document's directory:
   i. `load-pkgs.tex` – contains only the first three lines (the `\usepackage` commands only)
   ii. `renew-cmd.tex` – contains the remaining lines, which store the existing `\includegraphics` command as a macro and redefine it
2. Include the first file in the YAML section (`in_header: load-pkgs.tex`)
3. Before the `dfSummary()` chunk(s), paste this *tex* command on a new line:

   `\input{renew-cmd.tex}`
4. After the chunk(s), set the `\includegraphics` back to its original value using the following command (also on a new line):

   `\let\includegraphics\OldIncludegraphics`

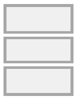
You might need to repeat steps 3 and 4 several times if your document alternates between `dfSummary()` tables and other content with images.

[1] There must be a *law of conservation of brokenness* sitting somewhere waiting to be discovered (although one could argue that it is merely a corollary to Murphy's law)

Proof That `includegraphics` Is Restored to Original

At this stage, the `\let\includegraphics\OldIncludegraphics` *tex* command has been executed.

```r
dfSummary(iris[5], headings = FALSE)
```

No	Variable	Stats / Values	Freqs (% of Valid)	Graph	Missing
1	Species [factor]	1. setosa 2. versicolor 3. virginica	50 (33.3%) 50 (33.3%) 50 (33.3%)		0 (0.0%)

If the operation of restoring the command worked, the results should be back to being misaligned, just as they were in the very first section.

Closing Remarks

If you are a LaTeX guru and can think of a simpler solution, please do let me know either by opening an issue or by sending me an email[2]. my address is available in the package's GitHub page as well as in the package's auto-generated pdf manual.

Useful links:

1. Introduction to summarytools (package vignette)
2. Summarytools in R Markdown Documents (package vignette)
3. Custom Statistics in dfSummary (supplemental documentation)
4. This StackOverflow question provides an additional example of how to revert a renewed command back to its original value.

[2] My email address is available in the package's GitHub page as well as in the package's auto-generated pdf manual.