

# Data Frame Summaries in PDF's


Dominic Comtois

2021-07-29

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

## 1. The Graphics Problem

```
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%)

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.

## 2. The Solution

To correct this issue, we need to redefine the `\includegraphics` command. If this breaks some other parts of your document<sup>1</sup>, see [section 2.3](#).

### 2.1 YAML Header

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

<sup>1</sup>There must be a *law of conservation of brokenness* sitting somewhere waiting to be formalized (although one could argue that this is merely a corollary to Murphy's law)

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 (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.

### 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}}}
```

### Modified YAML Section

Supposing you choose to keep the name `fig-valign.tex`, your YAML section should now look something like this:

```
---
title: "My Own Private PDF"
output:
  pdf_document:
    latex_engine: xelatex
    includes:
      in_header: fig-valign.tex
---
```

The *tex* file's name is entirely up to you; `fig-valign.tex` is the name used for the one in `summarytools'` `includes` directory, but it has no special meaning whatsoever.

## 2.2 Example

Here is a setup chunk which reproduces what has been used for this document, followed by a call to `dfSummary()`:

```
library(summarytools)
st_options(
  plain.ascii          = FALSE,
  subtitle.emphasis    = FALSE,
  style                = "rmarkdown", # For any other summarytools objects
  dfSummary.style      = "grid",
  dfSummary.graph.magnif = .5,
  dfSummary.valid.col  = FALSE,
  tmp.img.dir          = "/tmp"      # Recommended for Linux/OS X;
                                     # For Windows, using "img" is
                                     # a good habit
)

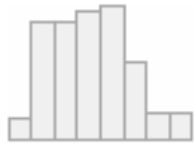
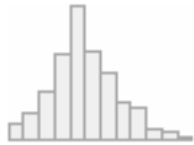
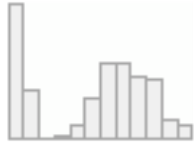
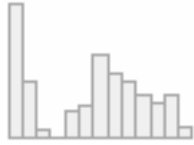

define_keywords(title.dfSummary = "Data Frame Summary in PDF Document")
dfSummary(iris)
```

## Data Frame Summary in PDF Document

iris

Dimensions: 150 x 5

Duplicates: 1

No	Variable	Stats / Values	Freqs (% of Valid)	Graph	Missing
1	Sepal.Length [numeric]	Mean (sd) : 5.8 (0.8) min < med < max: 4.3 < 5.8 < 7.9 IQR (CV) : 1.3 (0.1)	35 distinct values		0 (0.0%)
2	Sepal.Width [numeric]	Mean (sd) : 3.1 (0.4) min < med < max: 2 < 3 < 4.4 IQR (CV) : 0.5 (0.1)	23 distinct values		0 (0.0%)
3	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%)
4	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%)
5	Species [factor]	1. setosa 2. versicolor 3. virginica	50 (33.3%) 50 (33.3%) 50 (33.3%)		0 (0.0%)

## 2.3 A More Robust Solution

If redefining the `\includegraphics` command causes problems elsewhere in your document, following these instructions should take care of it<sup>2</sup>.

- Split the contents of `fig-valign.tex` into two files in your *Rmd* document's directory:
  - `load-pkgs.tex` – contains only the first three lines (the `\usepackage` commands only)
  - `renew-cmd.tex` – contains the remaining lines, which store the existing `\includegraphics` command as a macro and redefine it.
- Include the first file with YAML (`\\` indicates line feed):

```
output: \\ pdf-document: \\ includes: \\ in_header: load-pkgs.tex
```
- Before the `dfSummary()` chunk(s), paste this *tex* command, also on a new line:

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




```
\let\includegraphics\OldIncludegraphics
```

<sup>2</sup>File names and locations are suggestions only; adapt the instructions to your own needs.

## Proof That includegraphics Is Restored to Original

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

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

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

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  $\text{\LaTeX}$  guru and can think of a simpler solution, please do let me know either by opening an [issue](#) or by sending me an email; 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.