

# Some ideas on creating publication-quality tables in R

Stephen Weigand

Division of Biostatistics, Mayo Clinic  
email: weigand@mayo.edu

## Introduction and motivation

I put forth ideas on generating publication-quality tables in R using a formula interface and wrapping the results in an object that includes both the tabular summaries and components of a published table such as a caption and footnotes. The motivation for this work is that (1) biomedical publications make heavy use of tables; (2) these tables are often “one-of-a-kind” and used for a single manuscript (although re-created multiple times as the data or layout change); (3) creating such custom tables occupies a sizable portion of an analyst’s time. Currently, the table-making tools available in R are much less advanced than those for making plots. Since tables will remain a mainstay of biomedical publications for some time—the argument in favor of fewer tables and more effective graphical displays notwithstanding—additional tools could save analyst time while improving accuracy and reproducibility. There are four goals of this work:

1. define an R object that incorporates the structure of a complete table
2. propose a formula interface for constructing the column headers
3. propose a formula interface for constructing the tabular summaries to “fill” the cells
4. get feedback and suggestions

## Anatomy of a table

A prototypical table is below. Two key ideas are that (1) column headings can span rows and/or columns and that (2) rows can be “grouped” (e.g., the “Gender” group consists of three rows).

Table 1. A prototypical table

Stub column	Spanning header		<i>P</i>
	Col. head 1	Col. head 2	
Median (IQR) age, y	78 (51, 93)	79 (55, 92)	0.82
Gender, no. %			0.94
Men	34 (48.6)	32 (47.8)	
Women	36 (51.4)	35 (52.2)	

IQR, interquartile range

## A tableWrap object

The proposed table structure is called a `tableWrap` and is borrowed from the `<table-wrap>` element of the Journal Publishing Tag Set

Library (<http://dtd.nlm.nih.gov/publishing/tag-library/2.3/index.html>), an effort of the National Center for Biotechnology Information (NCBI) “to create a common format for the creation of journal content in XML.” The `<table-wrap>` construct is in some ways a wrapper for an XHTML `<table>` with a caption, headers, table body, and footnotes. The prototypical table would be constructed as follows.

```
tableWrap(id = "t1",
  caption = "A prototypical table",
  thead = thead(f), #create heads via formula
  tbody = rbind(b1, b2), #rbind body rows
  tableWrapFoot(xref(rid = "f1",
    "IQR, interquartile range")))
```

## Specifying the headings

The table header would be specified by passing a one-sided formula to `thead`. The formula has terms (back-ticked as necessary) representing column heads and interactions indicating grouped columns (columns “spanned” by a header). The columns for the prototypical table would be specified with

```
f <- ~ `Stub column` +
  `Spanning header`:(`Col head 1` +
    `Col head 2`) + P
```

`thead` would parse the formula to create a data frame representation of the column headings that specifies row and column positions and whether a header spans any rows or columns as shown below.

cell	row	col	rowspan	colspan
Stub column	1	1	2	1
Spanning header	1	2	1	2
P	1	4	2	1
...				

## Specifying the body with formulas

The body of the table would consist of one or more `tbody` objects each of which contains a row heading (and any subheadings) and the table cell entries. The body of the prototypical table could be created from two `tbody` objects:

```
b1 <-
  tbody(age ~ group, data = mydata,
    FUN = medIQR, P = "wilcox",
```

```
    head = "Median (IQR) age, y")
b2 <-
  tbody(gender ~ group, data = mydata,
    FUN = noPct, P = "chisq",
    head = "Gender, no. %",
    subhead = c("Men", "Women"))
```

As with the header information, a `tbody` object would be stored as a data frame indicating the cell contents, row, column, rowspan, and colspan information.

## Specifying the body with the matrix method

Constructing row groups using a formula interface has limitations. For example, summarizing effect sizes in a logistic model with a formula would be unnatural. For this one could pass a matrix to `tbody` as such:

```
tbody(rbind(c("1.3", "(1.1, 1.6)", "0.03"),
  c("2.4", "(1.2, 5.0)", "<0.01"),
  head = "Model results",
  subhead = c("Age, 10 y increase",
    "BMI, 1 unit increase")))
```

Of course, for this to be efficient, one needs “extractor” functions that generate the cell contents matrix.

## Footnotes

To footnote a header element, one would wrap it in an `xref` function and specify a reference ID (`rid`) and the note as follows:

```
~ Term + xref(P, rid = "tfn1", fn = "2-sided")
```

Elements in a `tbody` object could be footnoted using an `xref` argument specifying the cell (row and column), the `rid`, and the note itself. General notes could also be specified as shown in the example `tableWrap` object.

## HTML, L<sup>A</sup>T<sub>E</sub>X, and print methods

The structure of a `tableWrap` object would make marking it up in HTML easy, thus allowing the browser and style sheets to handle display and layout issues. Marking the table up in L<sup>A</sup>T<sub>E</sub>X would also be relatively straightforward although the limitations of the L<sup>A</sup>T<sub>E</sub>X table model and lack of a style-sheet concept typically would involve compromises in terms of keeping content and layout separate. A promising avenue to a print method would be to have a text-based Web browser render the table to the screen.