Creating Tables and Figures with \LaTeX

Ricky Patterson

bit.ly/LaTeXTables

5 Oct 2016
Outline

Introduction

Tables
  - tabular environment
  - multicolumn
  - @-expressions

Figures
  - picture environment
  - Importing Graphics

References
**Introduction**

- **\LaTeX** is a document preparation system for high quality typesetting.
- Figures and Tables are usually handled as ”Floats” which, unlike words, cannot be broken across a line or a page.
- **\LaTeX** allows you much control over the appearance and placement of tables and figures.
\LaTeX{} Tables

Tables are usually created within the \texttt{tabular} environment.

\begin{tabular}[pos]{table spec}

The table spec specifies the total number of columns, as well as the alignment for each column (l, c, or r). The option \texttt{pos} argument specifies the vertical location of the table with respect to the surrounding text (b, c, t). c is the default.
Keep in mind that \LaTeX{} is not a spreadsheet, and it is best to generate the values for a complex table externally, and then import the results into \LaTeX{}, adding the necessary formatting in an automated way if possible.
Keep in mind that \LaTeX\ is not a spreadsheet, and it is best to generate the values for a complex table externally, and then import the results into \LaTeX, adding the necessary formatting in an automated way if possible.

For instance: Each element in a \LaTeX\ table is separated from the adjacent column by an \&. Unless you enjoy typing \& symbols, you’ll want to format the write statement of the code that is writing out your data to add a \& symbol between columns to make it \LaTeX\ ready.
# \LaTeXe{} Tables

<table>
<thead>
<tr>
<th>header 1</th>
<th>header 2</th>
<th>header 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>header 4</td>
<td>cell 1</td>
<td>cell 2</td>
</tr>
<tr>
<td>header 5</td>
<td>cell 4</td>
<td>cell 5</td>
</tr>
</tbody>
</table>
**\LaTeX** Tables

<table>
<thead>
<tr>
<th>header 1</th>
<th>header 2</th>
<th>header 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>cell 1</td>
<td>cell 2</td>
<td>cell 3</td>
</tr>
<tr>
<td>cell 4</td>
<td>cell 5</td>
<td>cell 6</td>
</tr>
</tbody>
</table>

\begin{tabular}{c||c|c|c|c|c|c|c|c|c|c|c|}
\hline
|   | cell 1 | cell 2 | cell 3 | cell 4 | cell 5 | cell 6 |
\hline
header 5 & cell 4 & cell 5 & cell 6 & header 4 & cell 1 & cell 2 & cell 3 & header 4 & cell 4 & cell 5 & cell 6 & \textbf{header 1} & \textbf{header 2} & \textbf{header 4} & \end{tabular}
There are some additional symbols which can be used to describe table columns. Obviously | puts a vertical line between columns, and || puts two vertical lines.
There are some additional symbols which can be used to describe table columns. Obviously \( | \) puts a vertical line between columns, and \( || \) puts two vertical lines. In addition \( \texttt{p\{’width’\}} \) specifies a ”paragraph” column, of width ’width’.
There are some additional symbols which can be used to describe table columns. Obviously \verb+|+ puts a vertical line between columns, and \verb+||+ puts two vertical lines. In addition \verb+p{’width’} specifies a ”paragraph” column, of width ’width’. A non-p column does not wrap text within the cell, so if you have a lot of text in a cell, you may want to use \verb+p to control the appearance.
\begin{tabular}{| l | l | l | l |}
<table>
<thead>
<tr>
<th>Day</th>
<th>Min Temp</th>
<th>Max Temp</th>
<th>Summary</th>
</tr>
</thead>
<tbody>
<tr>
<td>Monday</td>
<td>11°C</td>
<td>22°C</td>
<td>A clear day with lots of sunshine.</td>
</tr>
</tbody>
</table>
\end{tabular}
\begin{tabular}{|l|l|l|l|}
\hline
Day & Min Temp & Max Temp & Summary \\
\hline
Monday & 11C & 22C & A clear day with lots of sunshine. \\
\hline
\end{tabular}

versus

\begin{tabular}{|l|l|l|p{5cm}|}
\hline
Day & Min Temp & Max Temp & Summary \\
\hline
Monday & 11C & 22C & A clear day with lots of sunshine. However, the strong breeze will bring down the temperatures. \\
\hline
\end{tabular}
Use \texttt{\textbackslash multicolumn} for cells which span more than one column:
\texttt{\textbackslash multicolumn\{number cols\}\{align\}\{text\}}% align: l,c,r
# \LaTeX Tables

<table>
<thead>
<tr>
<th>header 4</th>
<th>header 2</th>
<th>header 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>cell 1 and 2</td>
<td>cell 4</td>
<td>cell 5</td>
</tr>
<tr>
<td>cell 3</td>
<td>cell 6</td>
<td>cell 6</td>
</tr>
</tbody>
</table>
# \LaTeX Tables

<table>
<thead>
<tr>
<th>header 1</th>
<th>header 2</th>
<th>header 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>header 4</td>
<td>cell 1 and 2</td>
<td>cell 3</td>
</tr>
<tr>
<td>header 5</td>
<td>cell 4</td>
<td>cell 5</td>
</tr>
</tbody>
</table>

\begin{tabular}{c||c|c|c|c|}
\textbf{header 1} & \textbf{header 2} & \textbf{header 4} \\
\hline
\hline
header 4 & cell 1 and 2 & cell 3 \\
\hline
header 5 & cell 4 & cell 5 & cell 6 \\
\end{tabular}
\textbf{\LaTeX Tables}

\begin{tabular}{c||c|c|c|}
  header 4 & cell 1 and 2 & header 4 & cell 3 \\
  header 5 & cell 4 & cell 5 & cell 6 \\
\end{tabular}

\begin{verbatim}
\begin{tabular}{c||c|c|c|}
& header 1 & header 2 & header 4 \\
header 4 & cell 1 and 2 & & cell 3 \\
header 5 & cell 4 & cell 5 & cell 6 \\
\end{tabular}
\end{verbatim}
**\LaTeX** Tables

<table>
<thead>
<tr>
<th>header 1</th>
<th>header 2</th>
<th>header 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>header 4</td>
<td>cell 1 and 2</td>
<td>cell 3</td>
</tr>
<tr>
<td>header 5</td>
<td>cell 4</td>
<td>cell 5</td>
</tr>
</tbody>
</table>
### \LaTeX Tables

<table>
<thead>
<tr>
<th>header 1</th>
<th>header 2</th>
<th>header 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>header 4</td>
<td>cell 1 and 2</td>
<td>cell 3</td>
</tr>
<tr>
<td>header 5</td>
<td>cell 4</td>
<td>cell 5</td>
</tr>
</tbody>
</table>

\begin{tabular}{c||c|c|c|}
\hline
header 1 & header 2 & header 4 \\
\hline
\textbf{header 1} & \textbf{header 2} & \textbf{header 4} \\
\hline
\textbf{header 4} & cell 1 and 2 & cell 3 \\
\hline
\textbf{header 5} & cell 4 & cell 5 & cell 6 \\
\end{tabular}
Note: \texttt{\textbackslash multicolumn\{1\}\{align\}\{text element\}} can be used to simply alter the default column centering for that particular text element.
A Column Separator can be specified using the @-expression.
A Column Separator can be specified using the `@-expression`. By default, columns are separated with tabs (or vertical lines). However, using an `@-expression`, columns can be separated with decimal points or dashes (or any other character).
A Column Separator can be specified using the \texttt{@-expression}. By default, columns are separated with tabs (or vertical lines). However, using an \texttt{@}-expression, columns can be separated with decimal points or dashes (or any other character). This can be very useful if you wish to align a list of numbers on the decimal point:
A Column Separator can be specified using the \texttt{@-expression}. By default, columns are separated with tabs (or vertical lines). However, using an \texttt{@}-expression, columns can be separated with decimal points or dashes (or any other character). This can be very useful if you wish to align a list of numbers on the decimal point:

- left-aligned
  3.14159
  16.2
  123.456
A Column Separator can be specified using the \texttt{\@}-expression. By default, columns are separated with tabs (or vertical lines). However, using an \texttt{\@}-expression, columns can be separated with decimal points or dashes (or any other character). This can be very useful if you wish to align a list of numbers on the decimal point:

<table>
<thead>
<tr>
<th>left-aligned</th>
<th>centered</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.14159</td>
<td>3.14159</td>
</tr>
<tr>
<td>16.2</td>
<td>16.2</td>
</tr>
<tr>
<td>123.456</td>
<td>123.456</td>
</tr>
</tbody>
</table>
A Column Separator can be specified using the @-expression. By default, columns are separated with tabs (or vertical lines). However, using an @-expression, columns can be separated with decimal points or dashes (or any other character). This can be very useful if you wish to align a list of numbers on the decimal point:

<table>
<thead>
<tr>
<th>left-aligned</th>
<th>centered</th>
<th>right-aligned</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.14159</td>
<td>3.14159</td>
<td>3.14159</td>
</tr>
<tr>
<td>16.2</td>
<td>16.2</td>
<td>16.2</td>
</tr>
<tr>
<td>123.456</td>
<td>123.456</td>
<td>123.456</td>
</tr>
</tbody>
</table>
A Column Separator can be specified using the \texttt{\@}-expression. By default, columns are separated with tabs (or vertical lines). However, using an \texttt{\@}-expression, columns can be separated with decimal points or dashes (or any other character). This can be very useful if you wish to align a list of numbers on the decimal point:

<table>
<thead>
<tr>
<th>left-aligned</th>
<th>centered</th>
<th>right-aligned</th>
<th>@-aligned</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.14159</td>
<td>3.14159</td>
<td>3.14159</td>
<td>3.14159</td>
</tr>
<tr>
<td>16.2</td>
<td>16.2</td>
<td>16.2</td>
<td>16.2</td>
</tr>
<tr>
<td>123.456</td>
<td>123.456</td>
<td>123.456</td>
<td>123.456</td>
</tr>
</tbody>
</table>
A Column Separator can be specified using the `@-expression`. By default, columns are separated with tabs (or vertical lines). However, using an @-expression, columns can be separated with decimal points or dashes (or any other character). This can be very useful if you wish to align a list of numbers on the decimal point:

<table>
<thead>
<tr>
<th>left-aligned</th>
<th>centered</th>
<th>right-aligned</th>
<th>@-aligned</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.14159</td>
<td>3.14159</td>
<td>3.14159</td>
<td>3.14159</td>
</tr>
<tr>
<td>16.2</td>
<td>16.2</td>
<td>16.2</td>
<td>16.2</td>
</tr>
<tr>
<td>123.456</td>
<td>123.456</td>
<td>123.456</td>
<td>123.456</td>
</tr>
</tbody>
</table>

\begin{tabular}{r@{.}l}
3 & 14159 \\
16 & 2 \\
123 & 456 \\
\end{tabular}
It is possible to create figures within \LaTeX (usually within the picture environment), and there are packages available which simplify the creation of certain types of figures (like the tikz package for flowcharts, or pstricks which enhances the picture environment). This has a couple of advantages: the document is self-contained and small, and you have access to the fonts and math notation within \LaTeX.

\[ \beta = \frac{v}{c} = \tanh \chi \]
\[ F = \sqrt{s(s - a)(s - b)(s - c)} \]

\[ s := \frac{a + b + c}{2} \]
However, most complex figures you use will probably be produced in other programs (it’s just easier), and the resulting PDF or EPS file is imported into the LaTeX file. You will need to use the graphicx package in order to import graphics.
However, most complex figures you use will probably be produced in other programs (it’s just easier), and the resulting PDF or EPS file is imported into the \LaTeX{} file. You will need to use the graphicx package in order to import graphics.

Keep in mind: If you are compiling directly with latex command, you can only use EPS files. However, most compilation is now done with pdflatex, which can handle additional formats, including JPG (good for photos), PNG (lossless, good for diagrams and screenshots), PDF (can be used for vector graphics), and EPS (but you may have to load the epstopdf package).
Using the graphicx environment, you will use the includegraphics command to import the figure, while scaling, rotating and offsetting it to place it exactly where you want it.

```latex
\includegraphics[attr1=val1, ... attrn=valn]{image}
```

Possible attributes include: width, height, scale, angle, resolution, and keepaspectratio, as well as trim, clip, and page.

Let’s try importing a figure.

```latex
\includegraphics{1.eps}
```
Set the scale to 40%, so it fits better on the page:

```
\includegraphics[scale=0.4]{14.eps}
```
Now let’s rotate it, just for fun:

\includegraphics[angle=20]{14.eps}
And trim it a bit (note that trim is performed prior to rotation in this case):

\includegraphics[trim= 3mm 60mm 1mm 10cm, scale=0.4, angle=]
CTAN (Comprehensive \TeX\ Archive Network): ctan.org
TUG (\TeX\ Users Group): tug.org
The \LaTeX\ Companion (Mittelbach and Goossens, 2004)
A Guide to \LaTeX: Document Preparation for Beginners and Advanced Users (Kopka and Daly, 1999)
The Not So Short Intro to \LaTeX\ (Oetiker, updated July 2015 → lshort.pdf)
\LaTeX\ WikiBook https://en.wikibooks.org/wiki/LaTeX/
This presentation (code and PDF): http://bit.ly/LaTeXTables
Materials from previous \LaTeX\ workshops:
http://data.library.virginia.edu/statlab/past-workshops/
Overleaf.com Templates and Intro Guides.