Writing CHI Extended Abstracts With R Markdown

First Author
University of Author
Authortown, CA, USA
author1@anotherco.edu

Second Author
VP, Authoring
Authorship Holdings, Ltd.
Awdur, UK
author2@author.ac.uk

Third Author
Lēkhaka Labs
Bengaluru, India
author3@another.com

Fourth Author
YetAuthorCor, Inc.
Authortown, BC, Canada
author4@anotherco.com

Fifth Author
Université de Auteur-Sud
Auteur, France
author5@author.fr

Sixth Author
University of Umbhali
Pretoria, South Africa
author6@umbhaliu.ac.za

ABSTRACT

This is a guided illustration of how to write a CHI Extended Abstracts paper in R Markdown, using the latest official ACM LaTeX template Version 1.55, September 11, 2018. This R package is last updated on 23rd October 2018.

CCS CONCEPTS

• **Computer systems organization** → **Embedded systems**; *Redundancy*; Robotics; • **Networks** → Network reliability;

WOODSTOCK'97, July 1997, El Paso, Texas USA

How to Make a Side Bar

Use LaTeX Syntax: Sidebars need to be placed in the sidebar LaTeX environment. To achieve that, you can either: 1) write the text for the sidebar with markdown syntax, compile to PDF (if you are using RStudio, then just click the 'Knit' button), then open the generated **.tex** file for your manuscript wrap the desired parts inside \begin{sidebar} and \end{sidebar}, 2) simply all of the content for your sidebar using LaTeX syntax in your R Markdown manuscript.

The first option is better if you want to be able to compile your content to other formats than PDF, for example HTML or Microsoft Word.

In the source file for this example document, the sidebar is written in LaTeX syntax inside the R Markdown document.

Margin figures and tables: For these, however, there is R Markdown syntax you can use to automatically put dynamically generated tables or figures, as well as static figures, in the right LaTeX environment. This is explained later.

Sidebar 1: This is the optional caption

KEYWORDS

CHI Extended Abstracts; LaTeX; R Markdown; reproducible papers

ACM Reference Format:

First Author, Second Author, Third Author, Fourth Author, Fifth Author, and Sixth Author. 1997. Writing CHI Extended Abstracts With R Markdown. In *Proceedings of ACM Woodstock conference (WOODSTOCK'97).* ACM, New York, NY, USA, 9 pages. https://doi.org/10.475/123_4

INTRODUCTION

Using a tool like R Markdown to write scientific papers makes your work more transparent and reproducible. It also reduces the risk of errors, because you can dynamically insert tables, figures, and summary statistics directly from the data they are generated from insted of transferring results manually from statistical software to manuscript.

This example illustrates how to use the latest ACM LaTeX templates (Version 1.55, from September 11, 2018) with R Markdown to write papers for the CHI conference, in the CHI Extended Abtracts format. The content in this example is adapted and adjusted from content in **sample-sigchi.tex** as well as **sample-sigchi-a.tex** included with the ACM template, to illustrate how to create the same content through an R Markdown workflow as well as to showcase additional features made possible by R Markdown.

PAPER META DATA

Set meta data (copyright, authors, keywords, title, keywords, etc.) in the YAML header of the .Rmd file in which you write the manuscript. This is done in the form of key: value pairs, e.g. title: Writing CHI Proceedings Papers With R Markdown. When compiling to a PDF (in RStudio, just click the 'Knit' button), the information in the YAML header is plugged into the CHI Extended Abstracts LaTeX template (specified in the YAML header with template: sample-sigchi-a.tex). If you take a look at this template file, you will see e.g. \title[$short-title$]{$title$} - things between dollar signs are interpreted as a variable to be searched for in the YAML header and plugged into the template when generating a PDF.

Note the sole exception for adding paper meta data: The CCS Concepts are messy to insert from the YAML header, so you should manually insert this into either the template sample-sigchi-a.tex or the **.tex** file generated for your manuscript when you knit to PDF.

THE BODY OF THE PAPER

Typically, the body of a paper has a hierarchical structure, with numbered or unnumbered headings for sections, subsections, sub-subsections, and paragraphs. Whereas in LaTeX you use the command

[1]By the way, this is how to insert footnotes. In the Extended Abstracts format, footnotes are displayed as side notes in the margin.

\section for main sections, in R Markdown you simply use #, as in # The Body of The Paper. For subsections, or sub-subsections, use additional hashes, as in ## This Become a Subsection, and #### This Becomes a Paragraph Heading.[1]

If you want some section to be unnumbered in the output, add {-} after the section name, as in # Unnumbered Section{-}.

Indicate the start of a new paragraph with a blank line in your input file; that is why this sentence forms a separate paragraph. This line, however, does not form a separate paragraph.

Type Changes and *Special* Characters

Make words or phrases *italicized* by surrounding them with a single *; **embolden** them by surrounding them with **two**. Typewriter-style (for instance, for computer code) you create by surrounding text with `backticks`.

Citations

Citations to articles [1, 2, 4], conference proceedings [3] or maybe books [5, 6] listed in the Bibliography section of your article will occur throughout the text of your article. To insert a reference in the R Markdown syntax, type @ followed by the citation key. The key is a short reference uniquely identifying each entry in in the .bib file for your article, in which your references are listed in BibTex format.

For example, to cite the article "Deciding equivalances among conjunctive aggregate queries" from our .bib file, write [@Cohen07]. If you drop the []'s, you get author names, as well as the citation: Cohen et al. [4]. See this short guide for more.

DYNAMIC REPORTING

One of the most important benefits of writing in R Markdown (aside from being able to compile to other formats than PDF, such as HTML or even Microsoft Word), is the ability to insert results dynamically into your manuscript using code chunks or inline code. This means that you can do analyses **directly** in your manuscript or, probably better, read file(s) with data, summaries, or results directly into your manuscript and refer to them dynamically.

This is important for two (related) reasons: 1. You avoid initial manual transfer of results from statistical software to manuscript, which reduces the risk of error. 2. If at a later stage you update the analysis files, the results reported in your manuscript are automatically also updated - this again reduces the risk of mistakes, because you don't need to manually update figures and tables.

In R Markdown syntax, **code chunks** have the following form:

```{ coding_language chunk_label, chunk_options}
```

```
# your code goes here
```

Inline code has the form `coding_language #code here`.

Setup chunks

The first chunk in an R Markdown document is usually used to load packages and set default chunk options, for example like so:

```
```{ r setup, include=FALSE}
library(tidyverse)
knitr::opts_chunk$set(echo = FALSE, message = FALSE, warning = FALSE)
# by default, don't include code output, messages, or warnings in manuscript
```
```

You might read in a made-up data set of goals scored by basketball players like so (the chunk option include=FALSE means we don't want this chunk to have any output in the manuscript):

```
```{ r, include=FALSE}
data <- read_csv("data/fakeBasketData.csv")
```
```

Inline results

You can then use inline code to dynamically report properties of this data set. For example, "there are a total of 270 observations of goals scored. The mean number of goals made by any player in a given game is: 17.2555556".

Tables

You can also automatically create corresponding LaTeX tables from your data The easiest way is probably to use kable function For example, Table 1 shows the first 5 rows in our basket data set.

You can reference Table 1 with \@ref(tab:basket-data).

You can also do arbitrary transformations and analyses of the data before creating a table, as in Table 2.

Margin tables. To place a table in the margin, put it in a \margintable environment by adding the parameter table.env = 'margintable' when calling the kable function, like in Table 2.

Full width tables. To make a table take up the whole width of the page, put it in a \table* environment by adding the parameter table.env = 'table*' to kable, as in Table 3.

Table 2: Summary statistics of goals scored by top players in made-up basketball season.

Player	Total goals scored
Blake Griffin	406
Brook Lopez	776
Carmelo Anthony	166

Table 1: The first 5 rows of some made-up basket data.

Player	goals
Carmelo Anthony	4
Carmelo Anthony	2
Carmelo Anthony	10
Carmelo Anthony	3
Carmelo Anthony	1
Carmelo Anthony	3

Table 3: Bigger display of more summary statistics of goals scored by top players in made-up basketball season.

Player	Total goals scored	Goals per game	Length of name	Goals per letter in name	Goals per letter per game
Blake Griffin	406	13.533333	13	31.23077	3.470086
Brook Lopez	776	25.866667	11	70.54545	7.838384
Carmelo Anthony	166	5.533333	15	11.06667	1.229630
Damian Lillard	808	26.933333	14	57.71429	6.412698
David Lee	362	12.066667	9	40.22222	4.469136
David West	492	16.400000	10	49.20000	5.466667
Demar Derozan	972	32.400000	13	74.76923	8.307692
Deron Williams	365	12.166667	14	26.07143	2.896825
Dwyane Wade	312	10.400000	11	28.36364	3.151515

Figure 1: Here's a little pretty fly.

Figures

Static figures. Figures you similarly include via code chunks. You can include arbitrary static image files, as in Figure 1.

You can resize the figures with the chunk options `out.height` and `out.width`, as in Figure 2.

Figure 2: A sample black and white graphic that has been resized with the out.height **and** out.width **chunk options.**

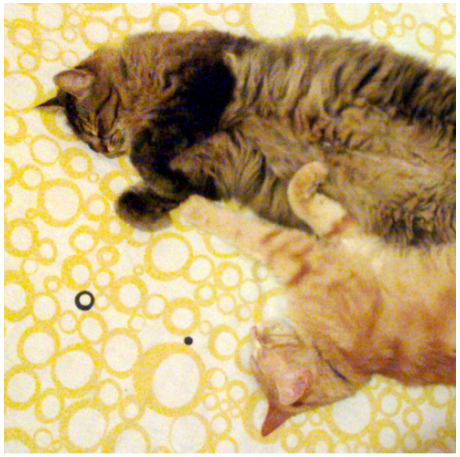

Figure 3: In this image, the cats are tessellated within a square frame. Images should also have captions and be within the boundaries of the sidebar on page 2. Photo: ⓒⓘ jofish on Flickr.

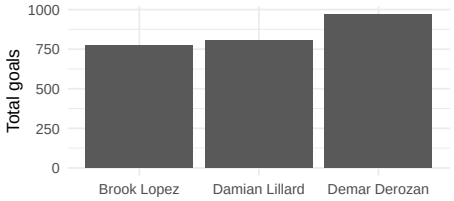

Figure 5: Total number of goals by the top 3 players in made-up basketball season

At the moment, if you need to style text in the caption, or include references in the caption, you need to use LaTeX rather than markdown syntax. As the figure caption is a string, you must escape the LaTeX syntax's \ with another \, as we did in Figure 2.

Margin figures. To place a figure in the margin, put it in a \marginfigure environment by setting the chunk option fig.env = 'marginfigure' as in Figure 3. Note that we also set the width to the width of the margin with out.width='\\marginparwidth'

Full width figures. To make figures take up to the full width of the page, set the environment to figure* with the chunk option fig.env = 'figure*'. You may want to set its width to the full text width with the chunk option out.width='\\fulltextwidth', as in Figure 4.

Dynamic figures. Again, the power of R Markdown is that you can include e.g. plots that are dynamically generated from the underlying data. For example, Figure 5 is a simple visualisation of the basket data.

Math Equations

You may want to display math equations in three distinct styles: inline, numbered or non-numbered display. Each of the three are discussed in the next sections. You can use usual LaTeX syntax directly, or R Markdown.

Inline (In-text) Equations. A formula that appears in the running text is called an inline or in-text formula. It is produced by the **math** environment, which can be invoked by surrounding text with dollar signs: $. You can use any of the symbols and structures, from α to ω, available in LaTeX. For example, here's a nice equation inline: $\lim_{n\to\infty} x = 0$. If you're writing in RStudio, you can even hover over it to see the rendered output displayed!

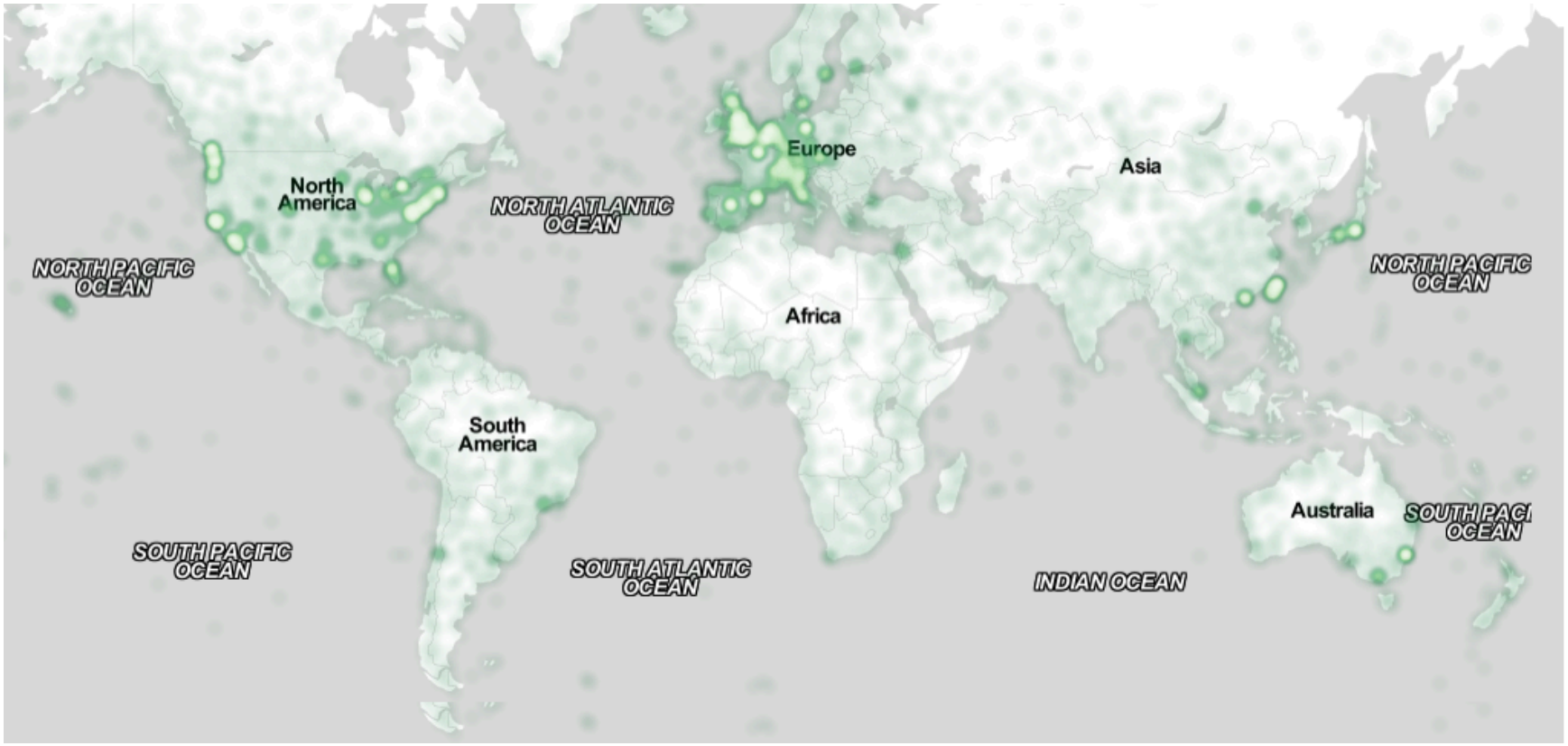

Figure 4: In this image, the map maximizes use of space. Note that LaTeX tends to render large figures on a dedicated page. Image: ayman on Flickr.

Display Equations. A numbered display equation—one set off by vertical space from the text and centered horizontally—is produced by using LaTeX syntax directly to put the content in an equation environment[2]. So here's that nice equation from above:

$$\lim_{n \to \infty} x = 0 \tag{1}$$

[2]In fact, you can use any arbitrary LaTeX syntax directly in your .Rmd document.

To make an unnumbered display equation, surround the expression with two dollar signs:

$$\lim_{n \to \infty} x = 0$$

Theorem-like Constructs

See bookdown.org for guidance if you want to do it the R Markdown way. Here's the usual way, using LaTeX syntax:

THEOREM 0.1. *Let f be continuous on $[a, b]$. If G is an antiderivative for f on $[a, b]$, then*

$$\int_a^b f(t)\,dt = G(b) - G(a).$$

Here is a definition:

Definition 0.2. If z is irrational, then by e^z we mean the unique number that has logarithm z:

$$\log e^z = z.$$

In the ACM LaTeX template, pre-defined theorem-like constructs are **theorem**, **conjecture**, **proposition**, **lemma** and **corollary**. The pre-defined definition-like constructs are **example** and **definition**. See the documentation for the 2017 ACM Master article template.

FINISHING TOUCHES

For the Extended Abstracts template in particular, it can be fiddly to get the final touches of the layout right, mostly in relation to the arrangement of the tables and figures in the margin. By default, `keep_tex:` is therefore set to `true` in the YAML header. This keeps the intermediary **.tex** file in your working directory after you have compiled to PDF. You can then make finishing touches in this file (e.g. using OverLeaf), e.g. you might want to play around with varying amounts of `\vspace` to position the margin content.

CONCLUSIONS

If you prefer using Markdown syntax over LaTeX, the CHI Proceedings format is a bit more straightforward than the Extended Abstracts format, because the latter may force you to resort to more direct use of LaTeX syntax to handle margin content appropriately. If you love LaTeX, of course, it's entirely possible to exclusively use LaTeX within R Markdown and only use R Markdown for its benefits of dynamic reporting of results.

REFERENCES

[1] Mic Bowman, Saumya K. Debray, and Larry L. Peterson. 1993. Reasoning About Naming Systems. *ACM Trans. Program. Lang. Syst.* 15, 5 (November 1993), 795–825. https://doi.org/10.1145/161468.161471

[2] Johannes Braams. 1991. Babel, a Multilingual Style-Option System for Use with LaTeX's Standard Document Styles. *TUGboat* 12, 2 (June 1991), 291–301.

[3] Malcolm Clark. 1991. Post Congress Tristesse. In *TeX90 Conference Proceedings*. TeX Users Group, 84–89.

[4] Sarah Cohen, Werner Nutt, and Yehoshua Sagic. 2007. Deciding equivalances among conjunctive aggregate queries. *J. ACM* 54, 2, Article 5 (April 2007), 50 pages. https://doi.org/10.1145/1219092.1219093

[5] Leslie Lamport. 1986. *LaTeX: A Document Preparation System.* Addison-Wesley, Reading, MA.

[6] S.L. Salas and Einar Hille. 1978. *Calculus: One and Several Variable.* John Wiley and Sons, New York.