

Template for contribution to Computo

Example dedicated to Python users

Jane Doe¹  John Doe² 

¹Statistics, Name of Affiliation one ²Computer Science, Name of Affiliation two

Date published: 2026-08-02 Last modified: 2026-08-02

Abstract

Template for preparing a submission to Computo, using the [quarto journal extension](#), the Jupyter kernel (Python user) and the environment files to set-up the dependencies. For a detailed, step-by-step guide on preparing your article, setting up the continuous integration and submitting it to Computo, please consult the [guidelines for authors](#).

Keywords: key1, key2, key3

Contents

1	Introduction	1
1.1	About this document	1
1.2	Quarto	2
1.3	Requirements	2
2	Formatting	2
2.1	Basic markdown formatting	2
2.2	Mathematics	3
2.2.1	Mathematical formulae	3
2.2.2	Theorems and other amsthm-like environments	3
2.3	Python Code	3
2.4	Figures	3
2.5	Tables	5
2.6	Handling references	5
2.6.1	Bibliographic references	5
2.6.2	Other cross-references	5
2.7	Advanced formatting	5
References		5

1 Introduction

1.1 About this document

This document, accompanied by the [customized GitHub repository](#), provides a template for writing contributions to **Computo** (Computo Team 2020).

*Corresponding author: janedoe@nowhere.moon

Note

This document provides only the key formatting principles. For a detailed, step-by-step guide on preparing your article and submitting it to Computo, please consult the [guidelines for authors](#).

1.2 Quarto

Quarto is a versatile formatting system for authoring documents integrating markdown, LaTeX and code blocks interpreted either via Jupyter or Knitr (thus supporting Python, R and Julia). It relies on the Pandoc document converter.

1.3 Requirements

You need [quarto](#) installed on your system and the [Computo extension](#) to prepare your document. For the latter, once quarto is installed, run the following to install the extension in the current directory (it creates a `_extension` directory which is ignored by git thanks to `.gitignore` by default):

```
quarto add computorg/computo-quarto-extension
```

[Python](#) and [Jupyter](#) must be installed on your computer.

2 Formatting

This section covers basic formatting guidelines for quarto documents.

To render a document, run `quarto render`. By default, both PDF and HTML documents are generated:

```
quarto render # renders both HTML and PDF
```

Note

To check the syntax of the formatting below, you can use the `</>` source button at the top right of this document.

2.1 Basic markdown formatting

Bold text or *italic*

- This is a list
- With more elements
- It isn't numbered.

But we can also do a numbered list

1. This is my first item
2. This is my second item
3. This is my third item

2.2 Mathematics

2.2.1 Mathematical formulae

LaTeX code is natively supported², which makes it possible to use mathematical formulae:

$$f(x_1, \dots, x_n; \mu, \sigma^2) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{1}{2\sigma^2} \sum_{i=1}^n (x_i - \mu)^2\right)$$

It is also possible to cross-reference an equation, see Equation 1:

$$\begin{aligned} D_{x_N} &= \frac{1}{2} \begin{bmatrix} x_L^\top & x_N^\top \end{bmatrix} \begin{bmatrix} L_L & B \\ B^\top & L_N \end{bmatrix} \begin{bmatrix} x_L \\ x_N \end{bmatrix} \\ &= \frac{1}{2} (x_L^\top L_L x_L + 2x_N^\top B^\top x_L + x_N^\top L_N x_N), \end{aligned} \tag{1}$$

2.2.2 Theorems and other amsthm-like environments

Quarto includes a nice support for theorems, with predefined prefix labels for theorems, lemmas, proposition, etc. see [this page](#). Here is a simple example:

Theorem 2.1 (Strong law of large numbers). *The sample average converges almost surely to the expected value:*

$$\bar{X}_n \xrightarrow{a.s.} \mu \quad \text{when } n \rightarrow \infty.$$

See Theorem 2.1.

2.3 Python Code

Quarto uses either Jupyter or knitr to render code chunks. This can be triggered in the yaml header. In this tutorial, we use Jupyter (Python and Jupyter must be installed on your computer).

```
---
title: "My Document"
author: "Jane Doe"
jupyter: python3
---
```

python code chunks may be embedded as follows:

```
array([ 0.76100389, -0.92590065, -0.0218411 , -0.30236353,  1.41199425,
        1.61661284,  1.62065843, -0.42576772, -0.59728373, -1.37612201])
```

2.4 Figures

Plots can be generated as follows:

It is also possible to create figures from static images:

²We use [lualatex](#) for this purpose.

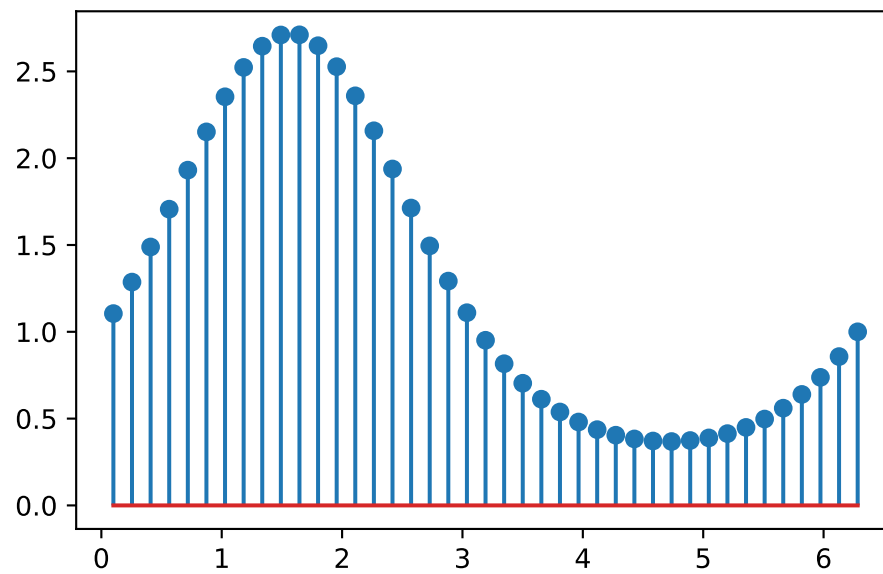


Figure 1: A basic Stem plot

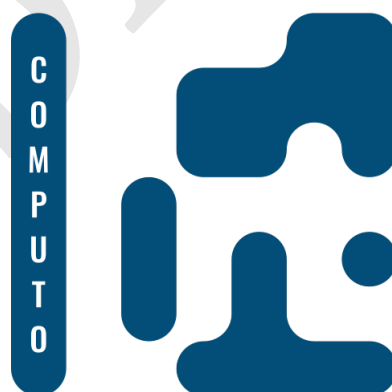


Figure 2: Computo logo (label)

2.5 Tables

Tables (with label: @tbl-mylabel renders Table 1) can be generated with markdown as follows

Table 1: my table caption

Tables	Are	Cool
col 1 is	left-aligned	\$1600
col 2 is	centered	\$12
col 3 is	right-aligned	\$1

2.6 Handling references

2.6.1 Bibliographic references

References are displayed as footnotes using [BibTeX](#), e.g. [computo] will be displayed as (Computo Team 2020), where computo is the bibtex key for this specific entry. The bibliographic information is automatically retrieved from the .bib file specified in the header of this document (here: references.bib).

2.6.2 Other cross-references

As already (partially) seen, Quarto includes a mechanism similar to the bibliographic references for sections, equations, theorems, figures, lists, etc. Have a look at [this page](#).

2.7 Advanced formatting

Advanced formatting features are possible and documented (including interactive plots, pseudo-code, (Tikz) diagrams, Lua filters, mixing R + Python in the same document), but are beyond the scope of this simple introduction. We point to several entries in this direction.

More information

- [The Quarto web site](#) for comprehensive documentation, including:
 - [Tutorial](#)
 - [User guide](#)
 - [Options reference](#)
- [The template distributed with the Computo Quarto extension](#), which uses such advanced features.
- [Our mock version of the t-SNE paper](#), a full and advanced example using Python and the Jupyter kernel.
- [The previously published papers in Computo](#) can be used as references.

References

Computo Team. 2020. *Computo: Reproducible Computational/Algorithmic Contributions in Statistics and Machine Learning*.