Generalized Quarto Template

Sam Schiano John Doe

2024-01-25

Table of contents

YAML

The area at the top is the YAML command area which dictates some portions of the document. Additional and in-depth details for this portion can also be found at https://quarto.org/docs/reference/formats/opml.html#format-options

The list of commands include

- title
- author (name and information)
  - name
  - orcid

- date
  - today, now, last-modified
  - or dates in these orders
    * MM/dd/yyyy
    * MM-dd-yyyy
    * MM/dd/yy
    * MM-dd-yy
    * yyyy-MM-dd
    * dd MM yyyy
    * MM dd, yyyy
    * YYYY-MM-DDTHH:mm:ssZ
- format (or multiple formats)

  format:

  html: default

  pdf: default

  docx
- toc: table of contents
- code-fold: (true/false) - in an HTML format, the code is hidden in a drop down tab for the use to view or not view
- toc-depth: specify the number of section levels to include in the table of contents
- number-sections

You can set all R code chunk options here:

execute:

Table 1: These options are listed after the execute: command with a tab inset followed by the option with a colon, space and the action.

Execute options	Actions
echo	true, false
warning	true, false
eval	true, false, [...]
output	true, false, asis
warning	true, false
error	true, false
include	true, false
cache	true, false, refresh

Execute options	Actions
freeze	true, false, auto

Template

Quarto enables you to weave together content and executable code into a finished document. To learn more about Quarto see https://quarto.org.

This template consists of generalized code that could be used as a reference when creating a new quarto document. Tips and tricks that were learned along the way have been included in the respective section. A lot of Rmarkdown notation can be utilized into quarto with previous Rmarkdown documents still able to be rendered when converted to a .qmd (to my knowledge).

Formatting

Headings and Page Breaks

Headings can be noted by using a specified number of #. A single # indicates the *Header 1*, two # indicates *Header 2*, three # indicates *Header 3*, and so on.

Page breaks can be added manually into the text when editing the document in the source editor (not visual). The notation is `<br` which will create a manual page break between r objects.

Font Changes

- *Italic* is indicated through a word or set of words encased in *

  `*word*`

- **Bold** is indicated through a word or set of words encased by a set of double *

  `**word**`

Equations

Sample equation (inline): $R_y = \frac{\alpha * SSB_y}{1+\beta * SSB_y}$ Code: `$R_{y}=\frac{\alpha*SSB_{y}}{1+\beta*SSB_{y}}$`

Sample equation (own line):

$$R_y = \frac{\alpha * SSB_y}{1 + \beta * SSB_y}$$

Greek letters notations:

Generalized notation - `$\alpha$`

Greek Letter Spelling	Lowercase	Capital
alpha	α	A
beta	β	B
gamma	γ	Γ
delta	δ	Δ
epsilon	ϵ	E
zeta	ζ	Z
eta	η	H
theta	θ	Θ
kappa	κ	K
lambda	λ	Λ
mu	μ	M
nu	ν	N
omicron	o	O
rho	ρ	P
sigma	σ	Σ
tau	τ	T
upsilon	υ	Υ
chi	χ	X
psi	ψ	Ψ
iota	ι	I
xi	ξ	Ξ
pi	π	Π
phi	ϕ	Φ
omega	ω	Ω

Other Helpful Functions for Equations

Component	Notation	Output
Subscript	`$*symbol*\_{sub}$`	$*symbol*_{sub}$
Superscript	`$*symbol*\^{sup}$`	$*symbol*^{sup}$
Fractions	`$\frac{num}{denom}$`	$\frac{num}{denom}$
Roots	`$\sqrt{num}$`	$\sqrt{num}$

For in-line text, superscript and subscript notation is different.

- Superscript^2 - `superscript^2^`
- Subscript$_2$ - `subscript~2~`

Running Code

When you click the **Render** button a document will be generated that includes both content and the output of embedded code. You can embed code like this:

```
x = 1 + 1
print(x)
```

```
[1] 2
```

You can add options to executable code to remove the code chunk in the rendered document like this

```
[1] 4
```

The `echo: false` option disables the printing of code (only output is displayed).

Other options include:

Option	Description
eval	Evaluate the code chunk
output	Include the source code in output
warning	Include warnings in the output
error	Include errors in the output
include	Catch all for preventing any output code (code or results)

Figures

Inline references to figures can be generate by using the @. If one referenced a figure in the test, you would need to label the figure in the R code chunk (Figure 1).

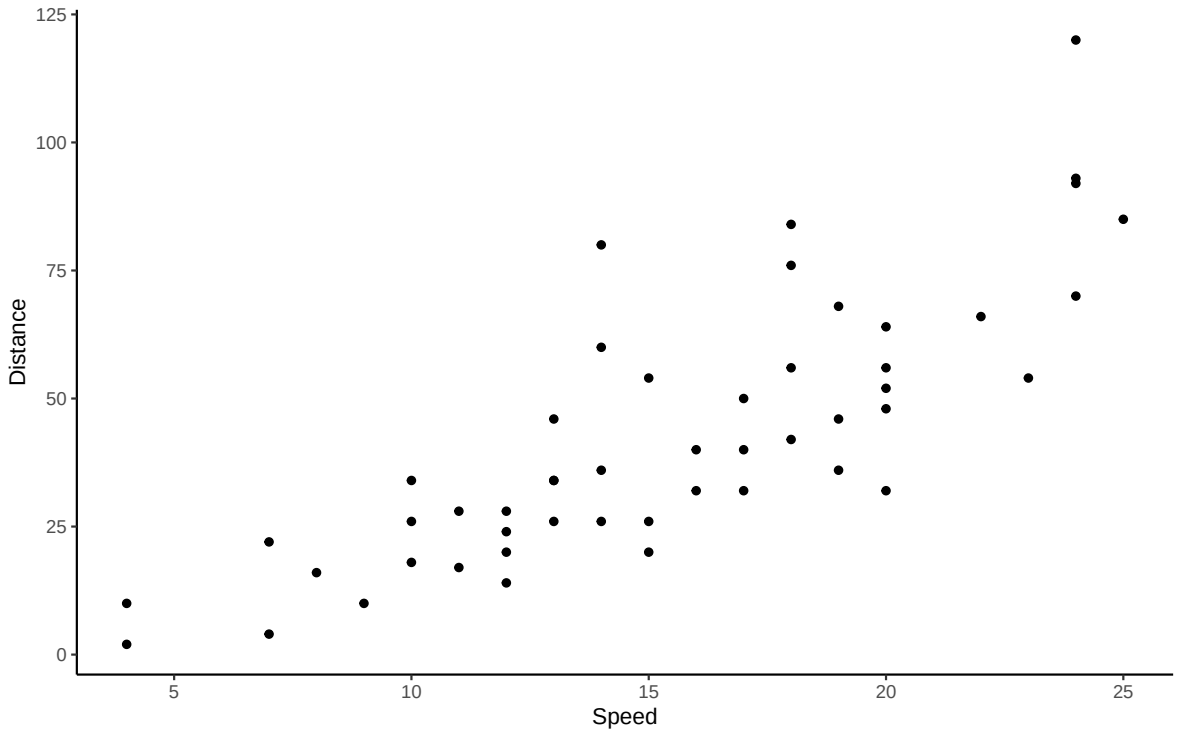


Figure 1: Test figure caption and example to add in.

Tables

Adding flextable objects into the document.

Table 5: Test1 table caption and example to add in.

xy
12021
22022
32023
42024

Table 6: Test2 table caption to add in break.

xy
52021
62022
72023
82024

Modular Workflow

Quarto and Rmarkdown also allow the user to develop a template from a set of .qmd or .rmd documents from calling them in R chunks.

Child Documents

One can combine a series of qmd files when the report is very long and would be easier to work with in sections rather than a single full report. In order to add one to the template document, the user would need to add it as an R chunk that includes an option of child = "child_file.qmd". Child files will be added as a new section into the template.

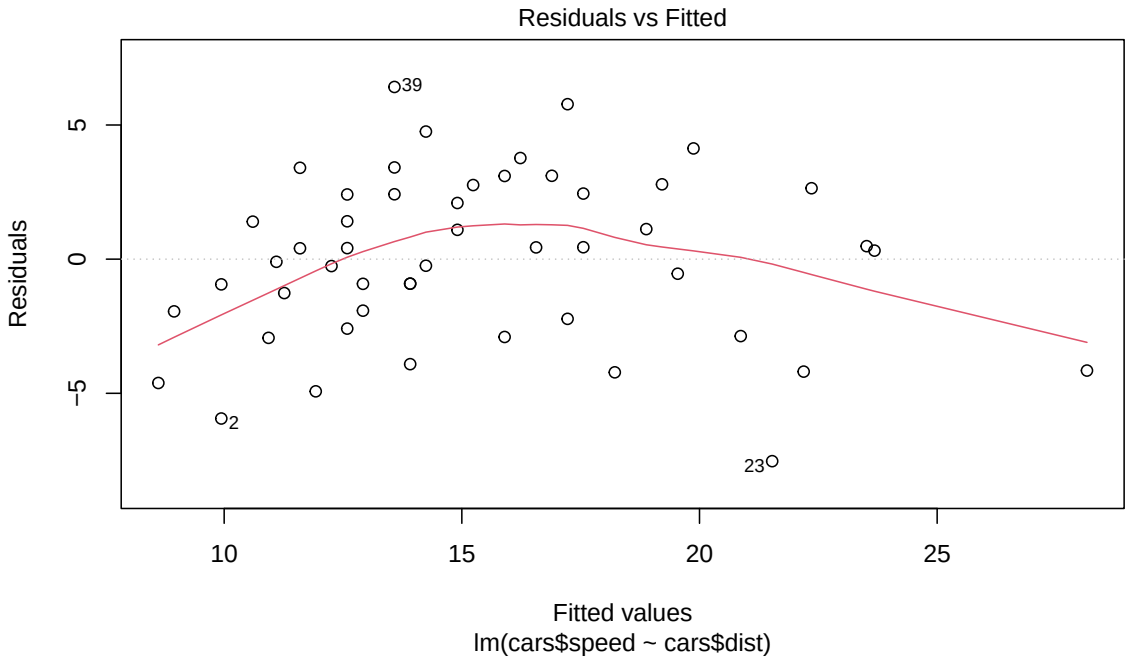
Child Template Example

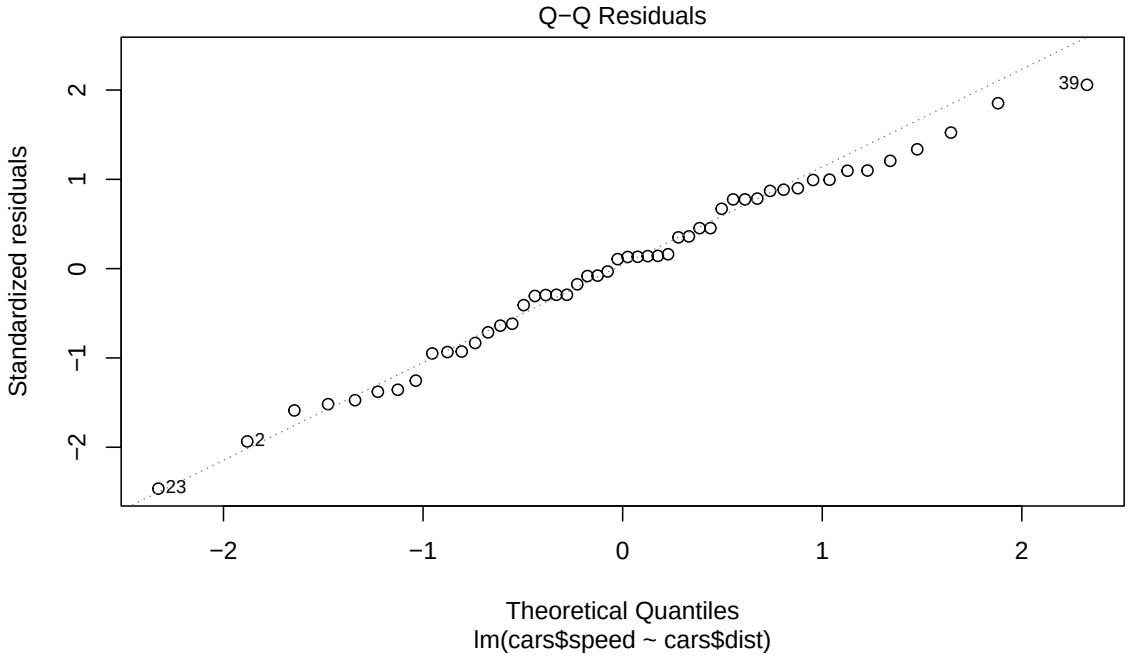
This document is designed to be built with the quarto template as an example child. It is implemented modularly to make the format easier to work with. Example R code is provided to test the functionality when adding it into the template document.

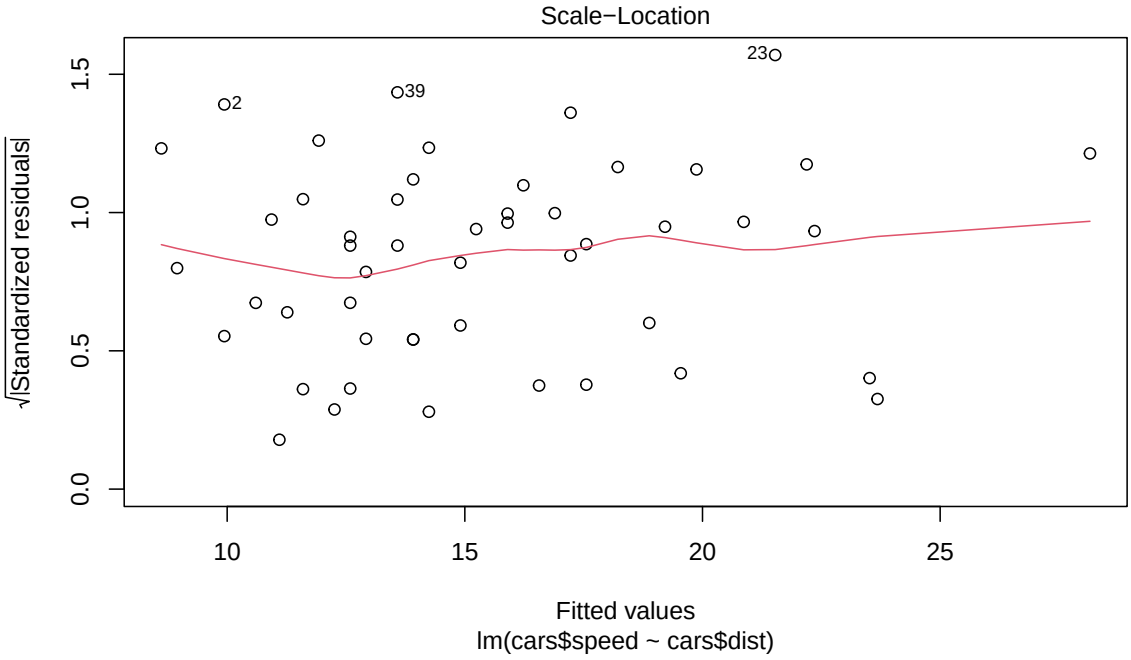
Function test:

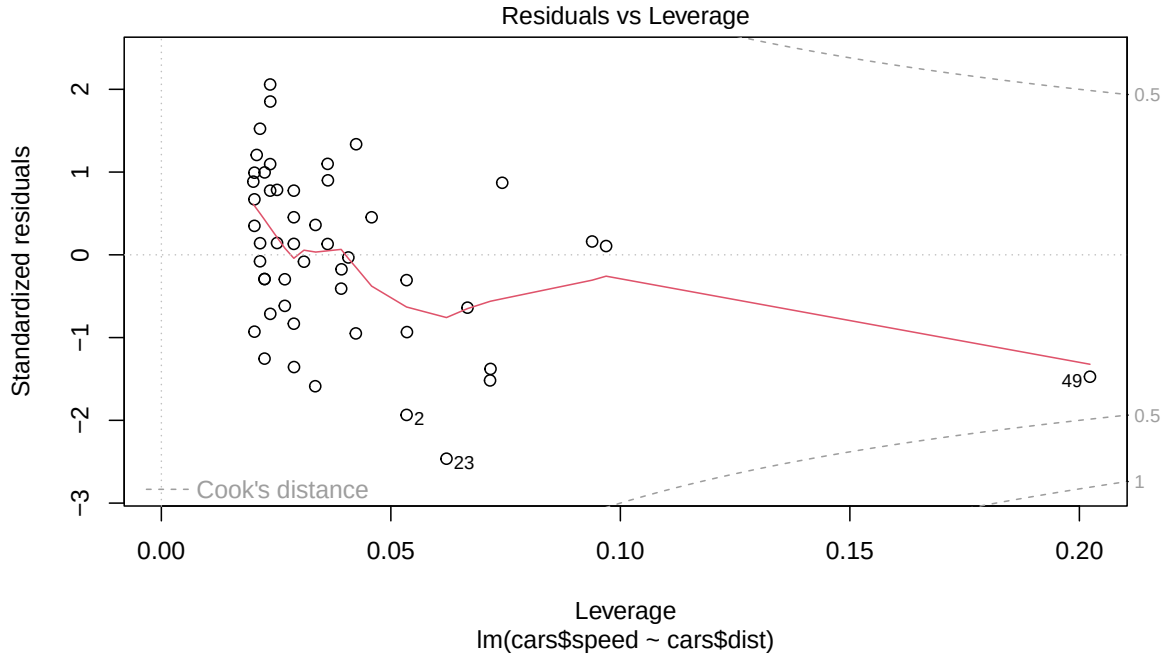
$$Z_{a,j} = F_{a,j} + M_{a,j}$$

Linear model and results for R code test:


Residuals vs Fitted
Residuals
Fitted values
lm(cars$speed ~ cars$dist)
39
2
23
10
15
20
25
5
0
−5


Q−Q Residuals
Standardized residuals
Theoretical Quantiles
lm(cars$speed ~ cars$dist)
39
2
23
−2
−1
0
1
2


Scale−Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
10
15
20
25
Fitted values
lm(cars$speed ~ cars$dist)
23
39
2


Residuals vs Leverage
Standardized residuals
−3
−2
−1
0
1
2
0.00
0.05
0.10
0.15
0.20
Leverage
lm(cars$speed ~ cars$dist)
Cook's distance
0.5
0.5
1
49
2
23

If the user just wants the output of the R code from the child, the user would set an additional option of eval = T.

Testing the visual editor when adding citations [@base].

References