New Quarto Website

A Book Made with Quarto

Liz Dobbins

Table of contents

1 New Quarto Website

A Book Made with Quarto

1.1 Welcome

Here is the place to put your welcome message

1.2 Section 1

Put the Section 1 text here

1.3 Section 2

This is a stoat

Figure 1.1: This is a stoat.

1.4 `renv` Package Management

Packages in this repo are managed by [renv](). The workflow takes a second to get used to, but doesn't have to be a big deal. Similar to `npm`, `renv` is globally installed on your system independent of the repo environment.

Once the repository has been cloned to your computer and `renv` is installed on your system, install the necessary R dependencies:
1. In a terminal window, navigate to the root directory of the repo
2. Start R by typing `R` and pressing Enter
3. In the resulting R console, run the following command

```
renv::restore() #installs packages in the `renv.lock` file
```

As you write code that needs additional packages, include them at the top of your scripts as `library("new-package")`. Then, before committing, run steps 1 through 3 above, but use `renv::snapshot()` in step 3. This will comb through the repo to look for packages not in the renv.lock file. When you are prompted for a selection, select option 2: `"Install the packages, then snapshot"` to bring the new packages into the fold.

*Note: To get you started, the renv.lock file of this repo includes all **tidyverse** packages*

1.5 Rendering a PDF

To render a PDF of this code, open a terminal, navigate to the root of the repo and run the following command. Make sure to change the output directory.

```
quarto render --to pdf --output-dir /path/to/output/dir
```

Please note that rendering to PDF requires a LaTeX distribution to be installed on your system. If you don't have one, you can install TinyTeX, a lightweight, portable, and easy-to-maintain LaTeX distribution. You can install it in R with `tinytex::install_tinytex()`.

1.6 Conditional Rendering to HTML and PDF

Quarto bills itself as a "scientific and technical publishing system" and the system delivers. Code can be rendered into many formats, such as articles, presentations, dashboards, websites, blogs, and books.
The data team at ACEP has explored rendering to HTML and to PDF formats in order to build interactive websites and print publications.

Despite Quarto's powers, QMD files that generate Javascript and HTML won't render to PDF. In some cases, the render will throw an error. In order to maintain a code base that builds both an interactive website and a PDF document, we utilize seperate code blocks and conditional rendering. Effectively, code blocks are turned on or off depending on the target output. This is done in the chunk header.
If you'd like the chunk to run when building a website, use: `{r, eval=knitr::is_html_output()}`
If you'd like the chunk to run when building a PDF, use: `{r, eval=knitr::is_latex_output()}`
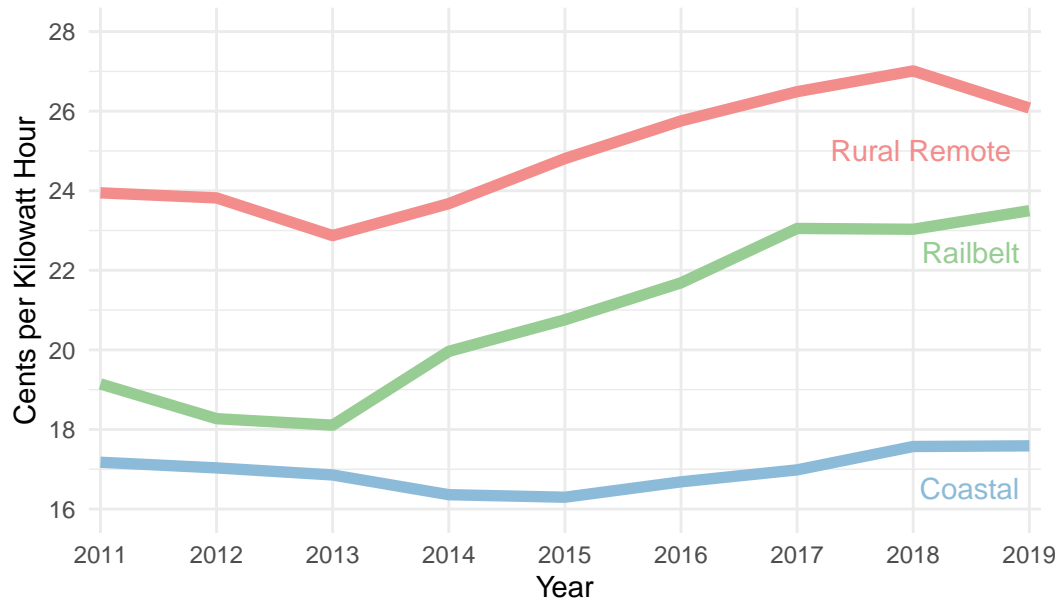
1.7 A tale of two chunks

Below is code to render two graphs using R. One code block is straight `ggplot`, the other contains an interactive tooltip from the `ggiraph` package. Since the interative tooltip builds out as HTML and Javascript, it will break if Quarto is rendering a PDF.
And so we have two chunks, one with interactivity that's destined for the web, and the other is static for the PDF report. By including them side-by-side, the flow of the document is preserved. By conditionally evaluating using the header code from above, we can build a website and static report from the same code base.
You will only see one graph no matter what the render format because the other is conditionally executed for the other format. Look below the graph to see either "HTML code block active" or "PDF code block active" depending on the format you are viewing.

Figure 2: Residential Price Over Time

[1] "PDF code block active"

A About

About this site.