tidyTouch: An interactive visualization tool for data science education

Jonah DeVaney[1] & Matthew McBee[1]

[1] East Tennesse State University

Abstract

One or two sentences providing a **basic introduction** to the field, comprehensible to a scientist in any discipline.

Two to three sentences of **more detailed background**, comprehensible to scientists in related disciplines.

One sentence clearly stating the **general problem** being addressed by this particular study.

One sentence summarizing the main result (with the words "**here we show**" or their equivalent).

Two or three sentences explaining what the **main result** reveals in direct comparison to what was thought to be the case previously, or how the main result adds to previous knowledge.

One or two sentences to put the results into a more **general context**.

Two or three sentences to provide a **broader perspective**, readily comprehensible to a scientist in any discipline.

Keywords:

Word count:

tidyTouch: An interactive visualization tool for data science education

Introduction

Technology is an absolute necessity in professional and academic spaces, where engineers, researchers, programmers, and more utilize an ever-growing collection of digital tools for organizing and analyzing the information vital to their work. Free and open-source software (FOSS) provides opportunities for unconditional access to useful programs and their source code for the sake of modification, improvement, and further sharing (Open Source Initiative, 2020). In cases where software is used for statistical analysis, many find R, a programming language for statistical analyses, to be a universally applicable tool to which many dedicated maintainers and community members contribute (R Core Team, 2020). While accessibility and extensive documentation make R available to individuals with limited knowledge or experience with programming, it is a more technically advanced tool, where a user writes code to read data, perform analyses, and create reports. This barrier gives reason to consider software options that may have limited capability but provide a more intuitive interface.

Combinations of spreadsheet editing programs like Microsoft Excel (Microsoft, 2019) and statistical analysis software like Minitab (Minitab, 2020) and IBM SPSS (IBM, 2017) allow less experienced analysts, like students, to visualize the possible structures and operations available for use with their data. These are typically marketed with intentions of the majority of users taking advantage of the graphical user interfaces (GUI), which are designed to give a point-and-click interaction method that engages the underlying code. These have the disadvantages of limited automation and accessibility, where users must manually perform steps of their analyses, often multiple times, on systems granted permission through paid subscriptions for software usage **citation needed?**.

The R community and immensely popular integrated development environment (IDE), RStudio, encourage the same transparency and information-sharing reflected in the mentality

of FOSS distribution (RStudio Team, 2020). Analyses in R can be performed in the console, where commands are given in the R language to be interpreted by the system. These analyses can just as easily be written in the form of a script that can be run as a combination of all operations intentionally recorded. Providing a powerful set of methods with infinite complexity, R programming is useful for anyone that works with data. As the practice of using large amounts of data to inform processes in various fields becomes more common through the expansion of data science as a field, education has and will continue to experience significant impacts (Piccianio, 2012). This can be observed on multiple fronts, where data science practices can be utilized by educational institutions in operation, as well as be implemented more as instructional content (Williamson, 2017).

Data science education has the potential to bring those that would otherwise use proprietary, GUI-based programs, like those previously mentioned, the small amount of technical training required to begin developing proficiency with tools like R. The language and its additional packages, supplemental files that allow expanded capabilities defined by their authors **citation needed?**, have extensive documentation that can be easily viewed within the RStudio IDE (RStudio Team, 2020); however, these are written as technical manuals that read in the format of R. The need to assist new students of R in gaining a literacy level in which they can solve problems on their own has motivated projects like the development of RStudio's Primers, online tutorials that teach basic example scenarios to utilize the RStudio suite of packages known as the `tidyverse` (RStudio Team, 2020; Wickham, 2017). These kinds of resources for data science education are crucial for training new academics and professionals as they work to embrace the ever-growing importance of data.

Data Visualization

Working with complex data is not a part of every position in an organization, but the information drawn from it can be meaningful to anyone. The importance of communication is highlighted by data visualization, arguably one of the most familiar aspects of data science to anyone outside of the field. Data visualization is a broad term describing any case in which data is visually represented in a form that allows interpretation of relationships and attribution of meaning to the information (Murray, 2017). Popular proprietary software makes generating plots a quick and simple task, providing ranges of options for visualization types in menus for easy selection; this is a commonly used feature of Microsoft Excel, where inserting "charts" requires few clicks to choose a visualization and designate portions of the available data to be used (Microsoft, 2019). This process is similar for even more specialized software for statistical analysis, like SPSS.

The approach to generating the same type of graphical product in R requires a knowledge of the programming language and the preferred `tidyverse` package for visualization, `ggplot2`. Rather than choosing from menus, a user must create the code for a plot. This is the challenge in teaching R and `ggplot2`, that other options are easier to use. As mentioned, a wide range of careers can be supported by an individual becoming a student of R. A tool that uses basic R and `ggplot2` language in an intuitive point-and-click interface would allow students and others in the early stages of their data science education to build their knowledge base without sacrificing convenience of ease. Providing this solution is the intention of this project, `tidyTouch`, a web application written entirely in R that provides a GUI for simple data visualization using `ggplot2`.

ggplot2 and The Tidyverse

The popular package `ggplot2` was developed based on a concept known as the grammar of graphics (Wickham, 2016). The grammar of graphics breaks down components

of a visual representation of data, what many would call a chart, and give specificity to its components in such a way that combinations of these characteristics can be used to generate more unique and meaningful "graphics" (Wilkinson, 2005). This design creates a systematic structure for the `ggplot2` package's own language of functions, the words used to specify modifications to what would otherwise be basic plots.

Every graphic is composed of the same basic components that are each addressed by a dedicated section of the code that creates a `ggplot2` graph. The *data* to be used is the first component specified, with a typical second addition being the plot's *layers*. A *layer*, in its simplest form, is the geometric object to be used in representing chosen data, what could be considered the type of graph (Wickham, 2016). The *data* and a *layer* are the only components that need to be specified by a user to create a graphic with `ggplot2`, as the other components are set to defaults. These are *scales*, specifications of size dimensions for the plot axes, the related *coordinate system*, *themes* that alter the plot's overall appearance, and *facets* that can divide information among multiple plots in a combined graphic (Wickham, 2016). Two basic plots showing differences between *themes* and *faceting* can be seen in Figure 1.

```
mtcars %>% ggplot(aes(x = hp, y = mpg)) + geom_point()

mtcars %>% ggplot(aes(x = hp, y = mpg)) + geom_point() +
  theme_bw() + facet_wrap(~cyl)
```

The code displayed for these graphs subtly demonstrates a concept at the core of `ggplot2` and other `Tidyverse` packages. These tools are designed to be utilized in the infinitely complex workflow of a data scientist, where a person is performing multiple operations on an object in a series of steps (Wickham, 2017). This object is often a dataset that is being read, summarized, or transformed to a "tidy" format, hence the name of the suite. Packages like `dplyr` and `ggplot2`, that use datasets saved in R, require this tidy format, which can be simply described as every variable having a column to represent it and

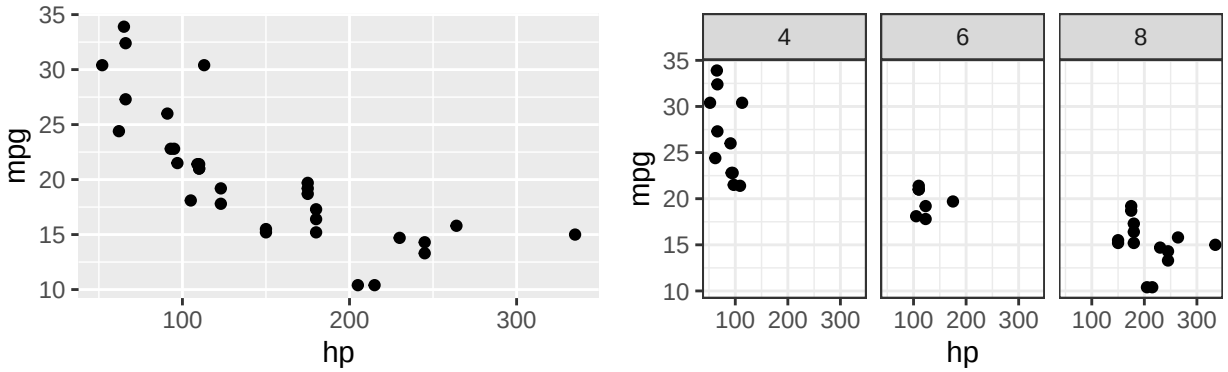

Figure 1. The two graphics are both comparing horsepower with gas mileage of cars in a sample R dataset. The (right) shows faceting by the number of cylinders a car has.

each data point or observation for that variable makes a row (Wickham, 2016). Understanding how to establish an efficient workflow is an essential skill to be covered in data science education, and `tidyTouch` utilizes a few of the most common packages for working with data.

Using R often involves much more than importing a dataset and performing the desired analysis. Once a file is imported, manipulating the data is often necessary. This is referred to as transformation, which involves rearranging, filtering, condensing, or calculating new components of datasets. All of these operations are done with simple functions, verbs that represent complex operations in R and take on the form of `f(x, y, ...)` in code (Wickham et al., 2020). The `x` represents the object being acted on with `y, ...` representing any options to specify for the operation. As an analyst transforms data, the series of steps used can be simplified by combining the verbs into a single string of multiple operations using a pipe (`%>%`). In the code below, a set of examples partially represented in `dplyr` documentation (Wickham et al., 2020) is given showing the same process with and without pipes.

```
#without pipes
a <- mutate(mtcars, gpm = 1/mpg)
b <- arrange(a, gpm)
```

```
#with pipes
c <- mtcars %>%
  mutate(gpm = 1/mpg) %>%
  arrange(gpm)
```

In both cases, the original dataset was used to create a new variable, and the rows of the expanded dataset were arranged by increasing order for the new variable. The first example requires saving multiple individual functions and changing the object being referenced. In a short series, this makes little difference compared to the second example; however the format with pipes will remain consistent with additional steps. This process could be expanded to include significantly more verbs, and with pipes, the end state of the process would be easily carried forward. The design of `tidyTouch` incorporates many `Tidyverse` package functions, as well as the use of pipes, to familiarize students of R with both the operations and format that they will find consistent in work across the data science field.

Shiny: R and Web Applications

Web applications (apps) are designed to make useful programs accessible online, able to be used freely without the hassle of installation processes. R can be used to create web apps with the help of the `Shiny` package (Chang, Cheng, Allaire, Xie, & McPherson, 2019), which converts R instructions to the type of code needed to design and format web pages. By using `Shiny` functions, an R programmer can create a web app that responds to user interaction by providing that user's input to a pre-built R program. This is useful for providing data and intended manipulations to non-programmers through educational tools (Chang et al., 2019; RStudio Team, 2020).

————————————-Break

Methods

We report how we determined our sample size, all data exclusions (if any), all manipulations, and all measures in the study.

Participants

Material

Procedure

Data analysis

Results

Discussion

R Packages and Session Info

To recognize those that contribute to R, tools used by members of the R community, and the continually developing field of data science, the software used in creating the tidyTouch app is listed: R (Version 3.6.3; R Core Team, 2020) and the R-packages *cowplot* (Version 1.0.0; Wilke, 2019), *dplyr* (Version 0.8.5; Wickham et al., 2020), *ggplot2* (Version 3.2.1; Wickham, 2016), *haven* (Version 2.1.1; Wickham & Miller, 2019), *papaja* (Version 0.1.0.9942; Aust & Barth, 2020), *reactable* (Version 0.1.0; Lin, 2019), *readr* (Version 1.3.1; Wickham, Hester, & Francois, 2018), *readxl* (Version 1.3.1; Wickham & Bryan, 2019), *rmarkdown* (Version 2.1; Xie, Allaire, & Grolemund, 2018), *shiny* (Version 1.4.0.9000; Chang

et al., 2019; Chang, 2018; Sievert, 2019), *shinymeta* (Version 0.2.0; Sievert, 2019), *shinythemes* (Version 1.1.2; Chang, 2018), and *tidyr* (Version 1.0.2; Wickham & Henry, 2020). This document was created using `papaja` and `rmarkdown` (Aust & Barth, 2020; Xie et al., 2018).

The session info for this project in its current state - containing the R version and additional loaded packages used during the development of this app, as well as the generation of this document - is printed below.

```
## R version 3.6.3 (2020-02-29)
## Platform: x86_64-pc-linux-gnu (64-bit)
## Running under: Ubuntu 18.04.4 LTS
##
## Matrix products: default
## BLAS:   /usr/lib/x86_64-linux-gnu/blas/libblas.so.3.7.1
## LAPACK: /usr/lib/x86_64-linux-gnu/lapack/liblapack.so.3.7.1
##
## locale:
##  [1] LC_CTYPE=en_US.UTF-8       LC_NUMERIC=C
##  [3] LC_TIME=en_US.UTF-8        LC_COLLATE=en_US.UTF-8
##  [5] LC_MONETARY=en_US.UTF-8    LC_MESSAGES=en_US.UTF-8
##  [7] LC_PAPER=en_US.UTF-8       LC_NAME=C
##  [9] LC_ADDRESS=C               LC_TELEPHONE=C
## [11] LC_MEASUREMENT=en_US.UTF-8 LC_IDENTIFICATION=C
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
```

```
## other attached packages:
##  [1] cowplot_1.0.0     rmarkdown_2.1     reactable_0.1.0   haven_2.1.1
##  [5] tidyr_1.0.2       readxl_1.3.1      readr_1.3.1       shinythemes_1.1.2
##  [9] shinymeta_0.2.0   shiny_1.4.0.9000  dplyr_0.8.5       ggplot2_3.2.1
## [13] papaja_0.1.0.9942
##
## loaded via a namespace (and not attached):
##  [1] styler_1.2.0      tidyselect_1.0.0  xfun_0.13         purrr_0.3.4
##  [5] colorspace_1.4-1  vctrs_0.2.4       sourcetools_0.1.7 htmltools_0.4.0
##  [9] yaml_2.2.1        rlang_0.4.5       pillar_1.4.3      later_1.0.0
## [13] glue_1.4.0        withr_2.1.2       lifecycle_0.2.0   stringr_1.4.0
## [17] munsell_0.5.0     gtable_0.3.0      cellranger_1.1.0  htmlwidgets_1.5.1
## [21] evaluate_0.14     labeling_0.3      forcats_0.4.0     knitr_1.28
## [25] fastmap_1.0.1     httpuv_1.5.2      fansi_0.4.1       highr_0.8
## [29] Rcpp_1.0.4.6      xtable_1.8-4      scales_1.0.0      promises_1.1.0
## [33] backports_1.1.6   mime_0.9          hms_0.5.1         digest_0.6.25
## [37] stringi_1.4.6     bookdown_0.18     grid_3.6.3        cli_2.0.2
## [41] tools_3.6.3       magrittr_1.5      lazyeval_0.2.2    tibble_3.0.0
## [45] crayon_1.3.4      pkgconfig_2.0.3   ellipsis_0.3.0    assertthat_0.2.1
## [49] R6_2.4.1          compiler_3.6.3
```

References

Aust, F., & Barth, M. (2020). *papaja: Create APA manuscripts with R Markdown*. Retrieved from https://github.com/crsh/papaja

Chang, W. (2018). *Shinythemes: Themes for shiny*. Retrieved from https://CRAN.R-project.org/package=shinythemes

Chang, W., Cheng, J., Allaire, J., Xie, Y., & McPherson, J. (2019). *Shiny: Web application framework for r*. Retrieved from http://shiny.rstudio.com

IBM. (2017). SPSS, version 25.0. Retrieved from https://www.ibm.com/analytics/spss-statistics-software

Lin, G. (2019). *Reactable: Interactive data tables based on 'react table'*. Retrieved from https://CRAN.R-project.org/package=reactable

Microsoft. (2019). Microsoft excel, version 16.0.12819.37950. Retrieved from https://products.office.com/en-us/excel

Minitab. (2020). Minitab statistical software, version 19.2020.1. Retrieved from http://www.minitab.com/en-us/products/minitab/

Murray, S. (2017). *Interactive data visualization for the web*. (M. Foley, Ed.) (2nd ed.). O'Reailly Media, Inc.

Open Source Initiative. (2020). OSI: Open source definition. Retrieved from https://opensource.org/docs/osd

Piccianio, A. G. (2012). The evolution of big data and learning analytics in american higher education. *Journal of Asynchronous Learning Networks*, *16*(3), 9–20.

R Core Team. (2020). *R: A language and environment for statistical computing*. Vienna, Austria: R Foundation for Statistical Computing. Retrieved from https://www.R-project.org/

RStudio Team. (2020). *RStudio: Integrated development environment for r*. Boston, MA: RStudio, Inc. Retrieved from http://www.rstudio.com/

Sievert, C. (2019). *Shinymeta: Record and expose shiny app logic using metaprogramming*.

Wickham, H. (2016). *Ggplot2: Elegant graphics for data analysis*. Springer-Verlag New York. Retrieved from https://ggplot2.tidyverse.org

Wickham, H. (2017). *Tidyverse: Easily install and load the 'tidyverse'*. Retrieved from https://CRAN.R-project.org/package=tidyverse

Wickham, H., & Bryan, J. (2019). *Readxl: Read excel files*. Retrieved from https://CRAN.R-project.org/package=readxl

Wickham, H., François, R., Henry, L., & Müller, K. (2020). *Dplyr: A grammar of data manipulation*. Retrieved from https://CRAN.R-project.org/package=dplyr

Wickham, H., & Henry, L. (2020). *Tidyr: Tidy messy data*. Retrieved from https://CRAN.R-project.org/package=tidyr

Wickham, H., Hester, J., & Francois, R. (2018). *Readr: Read rectangular text data*. Retrieved from https://CRAN.R-project.org/package=readr

Wickham, H., & Miller, E. (2019). *Haven: Import and export 'spss', 'stata' and 'sas' files*. Retrieved from https://CRAN.R-project.org/package=haven

Wilke, C. O. (2019). *Cowplot: Streamlined plot theme and plot annotations for 'ggplot2'*. Retrieved from https://CRAN.R-project.org/package=cowplot

Wilkinson, L. (2005). *The grammar of graphics* (2nd ed.). Springer Scinece & Business Media.

Williamson, B. (2017). *Big data in education.* (J. Clark, Ed.). SAGE Publications Inc.

Xie, Y., Allaire, J., & Grolemund, G. (2018). *R markdown: The definitive guide.* Boca Raton, Florida: Chapman; Hall/CRC. Retrieved from https://bookdown.org/yihui/rmarkdown