DATA SCIENCE, DASHBOARDS, AND THE WAY IT WORKS WITH
STATISTICS

by

Denise Renee Bradford

A DISSERTATION

Presented to the Faculty of

The Graduate College at the University of Nebraska

In Partial Fulfillment of Requirements

For the Degree of Doctor of Philosophy

Major: Statistics

Under the Supervision of Susan R. VanderPlas, Ph.D

Lincoln, Nebraska

Month, Year

DATA SCIENCE, DASHBOARDS, AND THE WAY IT WORKS WITH STATISTICS

Denise Renee Bradford, Ph.D.

University of Nebraska, Year

Adviser: Susan R. VanderPlas, Ph.D

Here is my abstract. *(350 word limit)*

COPYRIGHT

DEDICATION

Dedicated to...

ACKNOWLEDGMENTS

Thank you to all my people!

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

Chapter 2

Literature Review

2.1   History of Visual Inference

Visual inference uses our ability to detect graphical anomalies. The idea of formal testing remains the same in visual inference – with one exception: The test statistic is now a graphical display which is compared to a "reference distribution" of plots showing the null.[REFERENCE WEBSITE]

Plots of data are important for exploratory data analysis (EDA) along with model diagnostics. Plots for EDA are a useful tool for guiding an analyst in discovering the relationships between variables in their data. In the case when using plots in model diagnostics, plots help analysts determine whether or not the model is appropriate way to model. During the initial EDA stage, an analyst may find that a variable or a covariate is directly related to the dependent variable when looking at a correkation heatmap or a scatterplot. This will be important to know before starting a linear model analysis.

2.1.1   Formal Definition

2.1.2   Applications

2.2   History of Dashboard Design

2.2.1   Formal Definitions

2.2.2   Stephen Few Influences

2.2.3   Applications

2.3   Parallel Coordinate Plot Visualizations

2.3.1 R Packages

2.3.2 ggpcp package importance

2.4   Summary

Chapter 3

Chapter Paper on Rural Shrink Smart Manuscript submitted to Journal of Data Science Special Issue

3.1   Abstract

Many small and rural places are shrinking.   Interactive dashboards are the most common use cases for data visualization and context for exploratory data tools. In our paper, we will explore the specific scope of how dashboards are used in small and rural area to empower novice analysts to make data-driven decisions. Our framework will suggest a number of research directions to better support small and rural places from shrinking using an interactive dashboard design, implementation and use for the every day analyst.

3.2   Introduction

As the amount of data has increased in nearly every facet of life, the need to make sense of that data in an approachable, accessible form has become ever more important. As a result, many companies and organizations use interactive dashboards to present these data in a more useful and visually appealing form (Sarikaya, Correll, Bartram, Tory, & Fisher, 2019).

In many cases, dashboards support viewers' information processing, helping to make sense of complex data, navigate through a dataset, and supporting decision making based on the data.

Dashboards are often used, as with the car display of the same name, to provide summary information about many separate attributes of a common entity. One glance at a car's dashboard will tell you the speed, RPM, engine temperature, amount of gas in the tank; more importantly, however, the goal is not for the user to remember all of these characteristics, but to assess whether any of these quantities is outside of the expected range. Similarly, interactive dashboards for data are often used to display many different attributes and performance metrics which are of importance for stakeholders.

In this paper, we discuss the process of designing a dashboard to present publicly available government data to stakeholders in small Iowa towns to facilitate decision making and objective comparison with other similarly-situated towns.

Some communities continue to thrive as they lose population because they adapt, maintaining quality of life and community services for residents while investing in the future. This process, *smart shrinkage*, is important for rural areas who have experienced shrinking populations for decades. As small rural towns do not have access to data scientists or even the ability to easily leverage data collected locally to support decisions, our research team will provide communities with data about services in small town Iowa in order to assist with developing strategies to improve quality of life for their residents amid shrinking populations (Rural Shrink Smart Team, 2022). We hope to allow towns to explore their own data and compare to other similar towns, centering

decision-making on data in the context of small-town Iowa life.

3.3   Data Description

The Smart and Connected Community (SCC) dashboard data are primarily assembled from `data.iowa.gov` (State of Iowa, 2020), with some additional datasets assembled from federal and private sources. Most of these data sets are collected at a town/city or county spatial resolution, requiring us to carefully join data to ensure that these differences are respected while collating relevant information at the city level. In addition to the more commonly available statistics derived from e.g. the census and American Community Survey, `data.iowa.gov` contains several unique data sets, including local liquor sales, school building locations, town budgets and expenditures, hospital beds, Medicaid reimbursements, and other details that may provide information about local quality of life.

Data available on Iowa's data portal were augmented in some cases with higher-quality data sets in cases where the Iowa data were out of date or insufficiently accurate. Data collected from ELSI (National Center for Education Statistics, 2020) from `https://nces.ed.gov` were used to show the distance to any private or public school. The National Center for Education Statistics (NCES) is the primary federal entity for collecting and analyzing data related to education (Zarecor, Peters, & Hamideh, 2021).

Data collected from the Index of Relative Rurality (IRR) (USDA - ERS, 2020a) were used in the SCC dashboard to help classify the towns. The Index of Relative Rurality (IRR) is a continuous, threshold-free, and unit-free measure of rurality. It is an alternative to the traditional discrete threshold-based

classifications.The IRR ranges between 0 (low level of rurality, i.e., urban) and 1 (most rural). Four steps are involved in its design:

1. Identifying the dimensions of rurality: population size, density, remoteness, and built-up area.

2. Selecting measurable variables to adequately represent each dimension:

   - Size: logarithm of population size

   - Density: logarithm of population density.

   - Remoteness: network distance.

   - Built-up area: urban area (as defined by the US Census Bureau) as a percentage of total land area.

3. Re-scaling the variables onto bounded scales that range from 0 to 1.

4. Selecting a link function: unweighted average of the four re-scaled variable.

Data collected from Rural Urban Commuting Area Codes (USDA - ERS, 2020b) were used to help identify towns with commuting behaviors in our rural areas. The rural-urban commuting area (RUCA) codes classify U.S. census tracts using measures of population density, urbanization, and daily commuting. This data is on a zip code-level that will help identify those communities that commute to more urban areas. The most recent RUCA codes are based on data from the 2010 decennial census and the 2006-10 American Community Survey. The classification contains two levels. Whole numbers (1-10) delineate metropolitan, micropolitan, small town, and rural commuting areas based on the size and direction of the primary (largest) commuting flows.
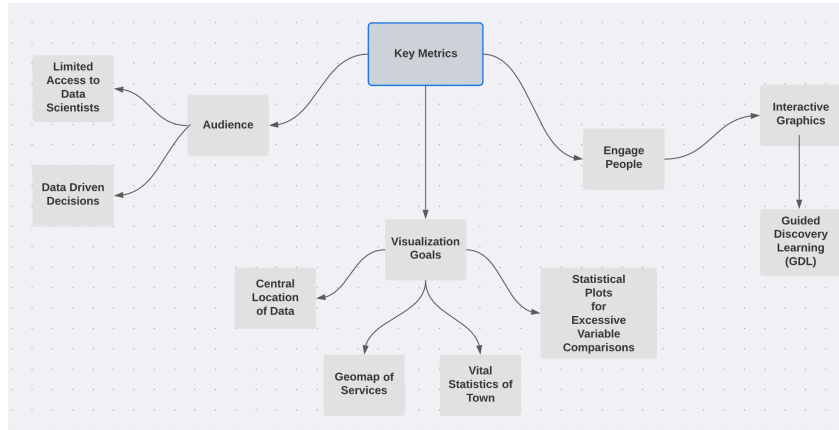
Figure 3.1: Diagram of considerations for our dashboard design process.

One of the interesting features of this assembled data set is that missing data can be missing for multiple reasons: not all state data is complete, but data about certain services may also be missing because towns do not offer that service. Thus, in addition to the usual challenges of working with real-world data that is "messy" in a variety of ways, we also have to contend with missing data that is missing due to the size of the community or the lack of services. This makes both visualization and statistical analysis more complicated (and more interesting).

3.4   Dashboard Design Considerations

One problem we identified early in the process of assessing smart-shrinkage strategies in small towns is that these towns do not have the resources to make data-driven decisions. Typically, small towns in Iowa are managed by at most a few part-time employees or volunteers. In some cases, essential management functions of the town are paid, but the municipalities we are interested in do not have sufficient funding to hire professionals to gather and analyze data.

As part of a wider project investigating the strategies towns use to main-

tain quality of life amid shrinking population, our research team provides communities with data about their own town, but also comparable towns across the state which may have a different approach to city services. In combination with other engagement strategies that are more qualitative, we hope to use this interactive dashboard approach to assist small Iowa cities with generalizing and developing strategies to improve or maintain quality of life amid shrinking populations.

One factor at the forefront of our visualization design is the importance of reducing the cognitive demands on viewers: we have assembled an incredible amount of data, and it is easy for even statisticians who deal with much larger datasets to get lost in the details of this data. At the same time, we want to invite viewers to engage with the data - to imagine, to draw comparisons, to generalize across towns, and to integrate outside information into the conclusions drawn based on the data we present. This invitation to engage with the data is similar to the approach advocated in Guided Discovery Learning, a framework leverages hints, feedback, and other helpful information to guide users in interactive exploration (DeDonno, 2016).

We expect that users will be interested in "sets" of variables from the wider dataset, which we assembled based on quality of life factors in the Iowa Small Town Poll (Peters, 2019). For instance, users might be interested in medical and social services available to residents, such as a local primary care clinic, nursing homes which are within driving distance, and the distance to the nearest emergency room; these factors might be explored separately from variables describing the services provided directly by city government, such as parks and recreation expenditures, snow removal services, and the distance to

the closest fire station.

As a consequence of this massively multivariate structure, we very quickly focused on the use of parallel coordinate plots; other alternatives, such as tours (Wickham, Cook, Hofmann, & Buja, 2011), require much more sustained attention to interactive plots as well as a deeper understanding of projections in multidimensional space which we cannot assume our users will have. Introduced in the 1880s (d'Ocagne, 1885), parallel coordinate or parallel set plots feature a series of vertical axes representing different variables arranged horizontally, with lines connecting each observation. When representing categorical data, parallel set plots may show "blocks" of data instead of individual lines, and are useful for representing conditional relationships between adjacent variables (Bendix, Kosara, & Hauser, 2005); modifications of this design, such as common-angle plots (Hofmann & Vendettuoli, 2013), address the issues which arise due to line-width illusions VanderPlas & Hofmann (2015). Parallel coordinate plots have been generalized to allow for continuous data and additional summaries beyond individual data points, such as densities (Heinrich & Weiskopf, 2009). In this paper, we use the `ggpcp` package, which leverages the grammar-of-graphics framework introduced in Wickham (2016), allowing us to use not only parallel coordinate plots, but also to overlay other statistical summaries, such as boxplots or violin plots, to provide additional context about the marginal distributions of each variable in addition to allowing for exploration of the multivariate space.

We also anticipate that users will be interested in comparing their town to other, similar towns. We will discuss the different ways that this comparison strategy was implemented in each dashboard in the next section, which de-

scribes the evolution of the dashboard over time and accounting for feedback from users and other researchers on the wider project.

One final component of this project is that our dashboard is part of a wider effort to work with towns to understand the different strategies used to maintain resident quality of life amid shrinking populations. Thus, while the town leaders are our primary audience, we also are creating this applet for use in parallel with a team of other researchers: sociologists, economists, city planning specialists, and artists. These researchers opinions and feedback about the dashboard are also useful and important, as they regularly work with town leaders in different capacities and have an understanding of what factors are most important to them and what types of questions these leaders may have when faced with data and unfamiliar statistical visualizations.

Throughout the design process, we will assess our visualizations to determine which strategies for user interface and interactive graphics design are most useful to empower town leaders to make discoveries in publicly available data assembled with a focus on items that impact rural quality of life.

3.5   Guiding Design Principles

Research on dashboard creation and interactive visualization tends to be very task-specific and hard to apply to more generalized settings. That is, it is relatively easy to create a dashboard that works for a particular task, but it is hard to generalize from that process what will work for the next dashboard. With this in mind, we set out to clearly document our intentions at each stage of the design and evaluation process, with the goal of gathering some useful information about general dashboard design from the process of creating this

specific dashboard.

Thus, our initial set of dashboard design principles is as follows:

- The town leaders are the focus audience; thus, the town itself should be the central focus of the app.

- We should facilitate comparisons with other towns in order to allow the user to explore other potential solutions to offering services that enhance resident quality of life.

- We will present the user with peer comparisons in order to widen the scope of exploration beyond the initial set of obvious peers in the local region.

- We will implement feedback mechanisms that allow us to provide more detailed data and respond to feature requests to improve the dashboard design over time.

As with many dashboards, this project is under continuous development; while it makes for an unsatisfactory conclusion, we do not have a "final" dashboard design because the application will continue to evolve. However, we have some useful insights into the process of creating an application designed to invite users to explore a large and complex dataset that we believe to be a useful contribution to work in this area.

3.6   Dashboard Design Process

3.6.1   Dashboard Components

In this section, we discuss the philosophy behind the basic "building blocks" of the dashboard. This philosophy is present in all of the iterations of the dashboard that we present in this discussion, and we will evaluate the overall philosophy's effectiveness in the conclusion.

The large set of publicly available data (primarily from `data.iowa.gov`) we have assembled is useful, but we must be careful with how we present this data because it would be easy to overwhelm the user with small details that mask the bigger picture. We select a small subset of towns (out of the 999 towns in Iowa) and a small subset of variables of interest to start with, and then allow the user to increase the complexity of the display in accordance with their interest. This avoids some of the pitfalls of dashboard design that can easily lead to user overload (Few, 2006).

Our primary objective is to provide users with a town-centric approach: their town is at the center of our application, and comparisons to other, similar towns are secondary. As a result, the next component of the dashboard is intended to provide a brief overview of the information we have about a specific town of interest. This design is based on research into visualization sensemaking (Lee et al., 2016), in that we allow users to explore outward from the familiar to the unknown. The map visuals were built using Open Source Routing Machine (OSRM) route functions (Luxen & Vetter, 2011) in R (R Core Team, 2022) to amplify the accuracy of the distances from necessary services in town-centric point. OSRM allows for finding the "As the Crow

Flies" distance and time on the road for our vital services map, since OSRM technology is similar to Google maps.

When faced with the next component, a parallel coordinate plot (PCP), a novice user will be able to determine two basic components: Visual Object (textual objects and non-textual objects) and Frame (frame of content and frame of visual encoding).

Taken together, the app is a single page; the initial "solid ground" which the user explores from consists of maps showing the route from the center of town to necessary services, including the fire department, schools, post offices, and hospitals. In version 2, as shown in Figure 3.3, the map portion is condensed, and more space is given to value boxes that show vital statistics about the town's QoL and financial metrics. This relatively straightforward display is followed by a parallel coordinate plot that allows the user to see similar towns along dimensions such as economic indicators or population size.

3.6.2   Initial Draft

The initial design sketch and implementation are shown in Figure 3.2.

Users' towns are at the center of our application, and comparisons to other, similar towns are secondary. As it can be extremely difficult to predict which towns are optimal for comparison purposes (similar may involve population, region, economic indicators, sports rivalries, and any number of other variables), we allow users to modify a set of suggested comparison towns to indicate other towns of interest.

We implemented some suggested town comparisons using unsupervised clustering methods to help our towns make decisions that are informed in
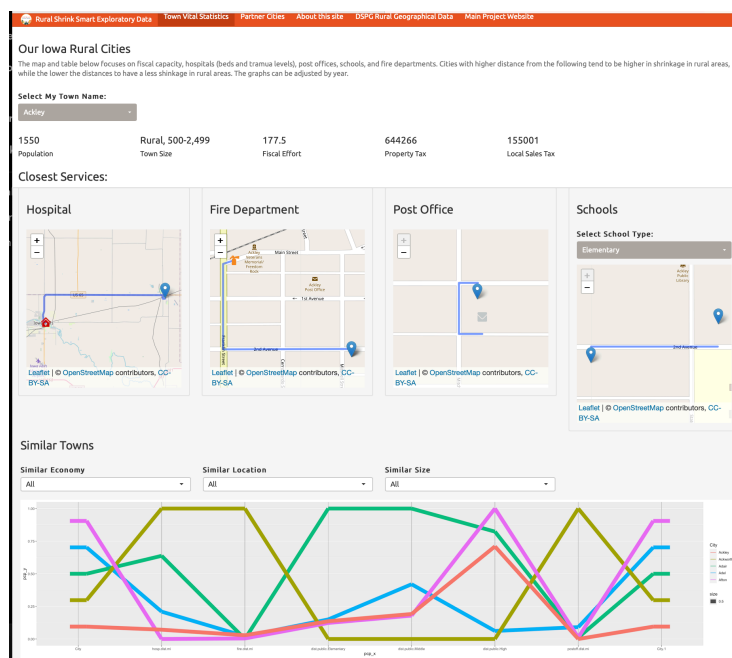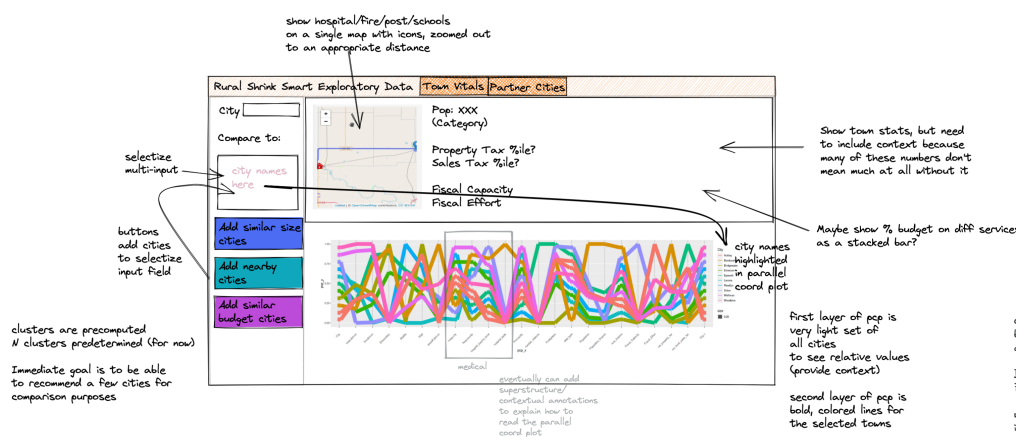
Figure 3.2: Initial dashboard design sketch (top) and implementation (bottom).

comparison to similar towns, for budget size, population size and location. We initially focused on determining the next five to ten similar towns, based on distances to services. This feature became an important diagnostic for our data quality, as it became clear that towns which were grouped with big cities but which did not have a large population were so grouped because of missing data. Unfortunately, this clustering feature was not as useful to the application users, as they came to the dashboard with a pre-existing set of towns to compare to; our suggested comparisons were in the way.

The initial dashboard design featured several responsive maps showing the distance to the nearest hospital, fire department, post office, and school. These maps were ineffective for several reasons:

- Town residents already know this information (though it was useful for us as the dashboard designers, because we aren't nearly as familiar with the 900+ small towns in Iowa)
- We computed distance from services relative to the center of town - coordinates provided in the data from `data.iowa.gov`. Generally speaking, the post office is at the center of town and the fire department is usually very close to the center of town; these two maps were useless. The school and hospital maps were less useless, but still did not provide particularly useful information to people already familiar with the town.
- It became clear that it might be more useful to show the comparison towns on a map (relative to the town of interest) so that users could compare geographical ratings for unfamiliar data to familiar data.

In addition, we received feedback on the parallel coordinate plot at the

bottom of the app which was surprising: the viewers (in this case, other researchers on the team) were not as intimidated by the parallel coordinate plot as we had expected. They did need some explanation of how to read the plot, and these hints need to be included in the dashboard, but they grasped the fundamental idea of the plot very quickly.

Our conclusion, based on this initial dashboard draft, was that we needed to restructure the application. Our attempt to show familiar information first to "build up" to the more unfamiliar structure of a parallel coordinate plot was not effective; there was too much clutter and not enough new information to draw users in.

3.6.3   Redesign

In the initial design, we included a map for each vital service, this initially created a lag for the users' experience. As a result, we cached map directions from OSRM for each service in our database, which drastically reduced the response time for the user. Our initial design did not naturally focus the user's eye on the most important parts of the dashboard; the redesign allowed for a cleaner flow from the top to the bottom.

In addition to the timing due to the map loading slowly, we added the vital statistics at the county level to allow for a more robust understanding of the town and it's surroundings. The rurality index provided a better classification and the USDA sources allowed for the town to understand the impact of the closest major city due to commuting for work and shopping at larger stores not available within the town.

We also modified the parallel coordinate plots in several ways:

Figure 3.3: A second iteration of the sketched design (top) and the implementation (bottom).

- Our x-axis had a large number of variables that we as researchers believed to be the most strongly associated with quality of life. However, there were still too many variables for users to successfully parse. We reduced the number of variables, focusing on variables that had the highest data quality, and we grouped these variables by quality of life factor (Peters, 2019).

- Originally, parallel coordinate bands were scaled based on the selected comparison towns. This had the effect of truncating the range of variables and over-emphasizing differences between selected towns relative to the overall range of each variable over all towns in our data set. We chose to show all towns in the data set in a very light α grey color to provide some information about the overall range of each variable. Unfortunately, even with the low-α value, this increased the visual complexity of the plot and confused users. Future iterations will likely make use of another aesthetic, such as boxplots or violin plots, to show the range of values for all towns, and then use lines only for towns that are selected by the user. This should strike a balance between visual complexity and representing the data accurately.

- We noticed that users did not make use of our suggested comparison towns, and so we removed that option in favor of allowing users to enter their own comparison towns directly. Users already had pre-determined towns they wanted to compare to, and our suggestions were just in the way.

While not all of these modifications were well received in our second round of user testing, the changes did incrementally move the dashboard display to-

wards our goal of allowing users to explore the data and engage with it. We continued to be surprised with how well users reacted to the parallel coordinate plots, which we initially thought might be too abstract for users unfamiliar with multivariate data displays, but the ability to compare towns across multiple dimensions and examine the similarities and differences between their approaches to different services seemed to be intuitive for users once they understood that each vertical axis was a different variable.

3.7   Discussion

Our dashboard design philosophy worked primarily to promote a town-centric approach application with comparisons to other similar towns being secondary. This approach created a way for the user to see their town information at the top of the page and to explore the PCP after reviewing their own town's essential statistics. The PCP in the lower part of the dashboard allowed for the user to see the plot and adjust to the fact that they could add more towns to the plot, providing an opportunity to explore the wider dataset from a base of familiar knowledge.

While we initially framed the design around guided discovery learning, the approach did not seem to suffice for our user base; instead, we found that users were more drawn to the unfamiliar from the start. We will likely leverage this in future iterations by using visual forms such as flower plots to draw the users in; even though these plots are not ideal for numerical display of data, the visual novelty and aesthetic appeal will provide some motivation to continue exploring and thinking about the data.

One factor that we have briefly considered and have seen hints of in our user

feedback is that towns may not want to be compared negatively with other towns. While users have very definite ideas about which towns they would like to compare to, we can always mask the town names and move back to comparisons based on town size and other factors (for instance, whether or not a town is the county seat is a factor that is important outside of population). Using this approach, we would label each town as "Town 1," "Town 2," and so on, which would eliminate some of the fears about negative comparisons, but would also remove some of the novelty of the data dashboard for our users and would prevent users from drawing on their own outside knowledge about each of the comparison towns.

We also recognize that we need to leverage the expertise of others in our research team: we are working with artists, researchers in architecture, economists, and sociologists; these researchers provide outside knowledge that we do not have and may be able to help us create insightful use-cases to showcase the app and teach towns how to use it. We can also leverage the app to connect users with our research team, providing additional value to those who use the applet and facilitating development of strategies for maintaining quality of life amid shrinking populations.

3.8   Future Work

One avenue we will explore in future iterations of the dashboard is to incorporate other dashboards generated by different groups within this project. This will create a wider field of information to explore: for instance, some of the additional work will focus on the 99 towns featured in the Iowa Small Town Poll; this will allow us to showcase survey-based measures of quality

of life alongside the more objective measurements assembled in the dataset discussed in this paper. While at least one tab of this omni-dashboard will still focus on wider EDA and discovery, we hope to incorporate other information as well to provide a more well-rounded data display encompassing most of the facets of this complex project.

We are also mindful of a distinction between "eye candy" and purpose-driven data visualization. While we have typically focused on the latter, there is certainly a place in our dashboard for the former as well. "Eye candy" visualization is intended to draw the viewer in and motivate them to explore; while these visualizations may not be particularly effective at communicating quantitative information, if they motivate the user to engage with the rest of the dashboard, they still serve a purpose. It is with this mindset that we intend to explore the use of flower plots - the artistic opportunities combined with the display of quantitative information (even in a form that isn't optimal for quantitative comparisons) may be useful to engage viewers before transitioning to more useful data visualizations intended to provide accurate quantitative comparisons.

EDA can be a difficult for a variety of groups of people, novice users and experienced researchers. One of the more difficult components of this project has been clearly articulating the purposes of EDA to a diverse group of researchers unfamiliar with the concept. One of the most useful parts of this dashboard iteration process has been as an aid to data discovery: that is, the dashboard motivated us to find additional data sources and incorporate them into the project. Having conversations with other researchers about the EDA process helped to facilitate these conversations, as each discussion seemed to

uncover additional data sources that someone remembered after looking at the dashboard. While this facet of the dashboard process may be difficult to study formally, it would be an interesting avenue for investigation.

3.9 Conclusions

In this paper, we have documented the process of designing a dashboard for exploration and visualization of a large and complex data set assembled from many different sources. Our primary audience was leaders of small towns in Iowa, with a secondary audience of researchers in fields other than statistics collaborating on this project with us. Through the process of revising our dashboard, we found that the idea of guided discovery learning as implemented in our first version did not work as well as we had anticipated. It was more important to focus on allowing users to explore their questions about the dataset by facilitating user-driven comparisons and exploration, rather than attempting to anticipate user desires by providing comparison towns. In addition, we found that it would be more effective to draw users in with novel visual displays, as these seemed to attract more interest than providing known facts and an opportunity to explore outwards from an initial area of familiarity.

While it is hard to apply the findings from one fairly specific visualization project more widely, there is a lack of resources in this area that provide both design philosophies and actual analysis of user feedback in a qualitative sense. We have attempted to address this dearth of information by providing the design strategies, user feedback, and our planned and executed modifications, in the hopes that others facing the daunting challenge of designing a dashboard for EDA may learn something from our experiences.

Chapter 4

Tables, Graphics, References, and Labels

4.1 Tables

By far the easiest way to present tables in your thesis is to store the contents of the table in a CSV or Excel file, then read that file in to your R Markdown document as a data frame. Then you can style the table with the `kable` function, or functions in the kableExtra pacakge.

In addition to the tables that can be automatically generated from a data frame in **R** that you saw in [R Markdown Basics] using the `kable` function, you can also create tables using *pandoc*. (More information is available at http://pandoc.org/README.html#tables.) This might be useful if you don't have values specifically stored in **R**, but you'd like to display them in table form. Below is an example. Pay careful attention to the alignment in the table and hyphens to create the rows and columns. Generally I don't recommend this approach of typing the table directly into your R Markdown document.

Table 4.1: Correlation of Inheritance Factors for Parents and Child

Factors	Correlation between Parents & Child	Inherited
Education	-0.49	Yes
Socio-Economic Status	0.28	Slight
Income	0.08	No
Family Size	0.18	Slight
Occupational Prestige	0.21	Slight

We can also create a link to the table by doing the following: Table 4.1. If you go back to [Loading and exploring data] and look at the `kable` table, we can create a reference to this max delays table too: Table **??**. The addition of the (\#tab:inher) option to the end of the table caption allows us to then make a reference to Table `\@ref(tab:label)`. Note that this reference could appear anywhere throughout the document after the table has appeared.

4.2   Figures

If your thesis has a lot of figures, *R Markdown* might behave better for you than that other word processor. One perk is that it will automatically number the figures accordingly in each chapter. You'll also be able to create a label for each figure, add a caption, and then reference the figure in a way similar to what we saw with tables earlier. If you label your figures, you can move the figures around and *R Markdown* will automatically adjust the numbering for you. No need for you to remember! So that you don't have to get too far into LaTeX to do this, a couple **R** functions have been created for you to assist. You'll see their use below.

In the **R** chunk below, we will load in a picture stored as `uw.png` in our main directory. We then give it the caption of "UW logo," the label of "uwlogo," and specify that this is a figure. Make note of the different **R** chunk options that are given in the R Markdown file (not shown in the knitted document).

```
include_graphics(path = "figure/unl.png")
```

Here is a reference to the UW logo: Figure 4.1. Note the use of the `fig:` code here. By naming the **R** chunk that contains the figure, we can then reference that figure later as done in the first sentence here. We can also specify the caption for the figure via the R chunk option `fig.cap`.
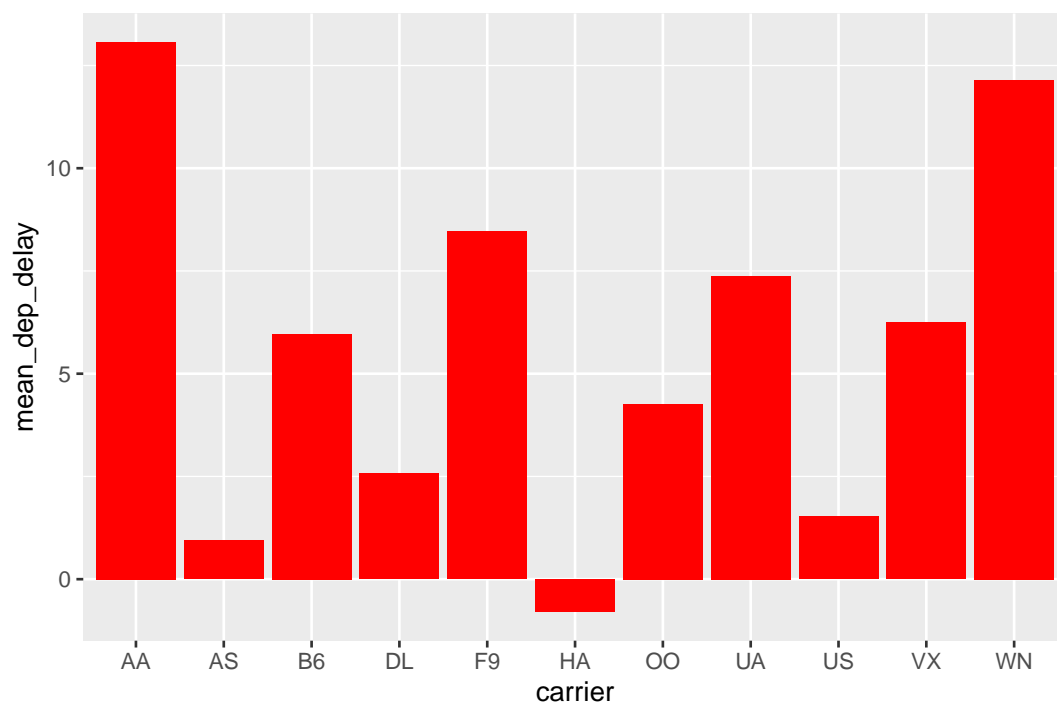
Figure 4.1: logo

Figure 4.2: Mean Delays by Airline

Below we will investigate how to save the output of an **R** plot and label it in a way similar to that done above. Recall the `flights` dataset from Chapter 2. (Note that we've shown a different way to reference a section or chapter here.) We will next explore a bar graph with the mean flight departure delays by airline from Portland for 2014. Note also the use of the `scale` parameter which is discussed on the next page.

```
flights %>%
    group_by(carrier) %>%
    summarize(mean_dep_delay = mean(dep_delay)) %>%
    ggplot(aes(x = carrier, y = mean_dep_delay)) +
    geom_bar(position = "identity", stat = "identity",
        fill = "red")
```

Here is a reference to this image: Figure 4.2.

A table linking these carrier codes to airline names is available at `https://github.com/ismayc/pnwflights14/blob/master/data/airlines.csv`.
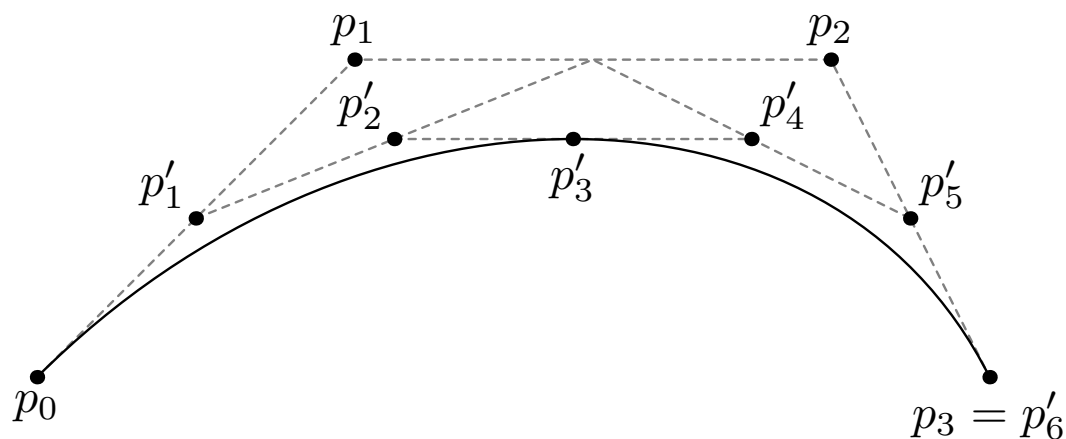
Figure 4.3: Subdiv. graph

Next, we will explore the use of the `out.extra` chunk option, which can be used to shrink or expand an image loaded from a file by specifying `"scale= "`. Here we use the mathematical graph stored in the "subdivision.pdf" file.

Here is a reference to this image: Figure 4.3. Note that `echo=FALSE` is specified so that the **R** code is hidden in the document.

More Figure Stuff

Lastly, we will explore how to rotate and enlarge figures using the `out.extra` chunk option. (Currently this only works in the PDF version of the book.)

As another example, here is a reference: Figure 4.4.

4.3 Footnotes and Endnotes

You might want to footnote something.[1] The footnote will be in a smaller font and placed appropriately. Endnotes work in much the same way.
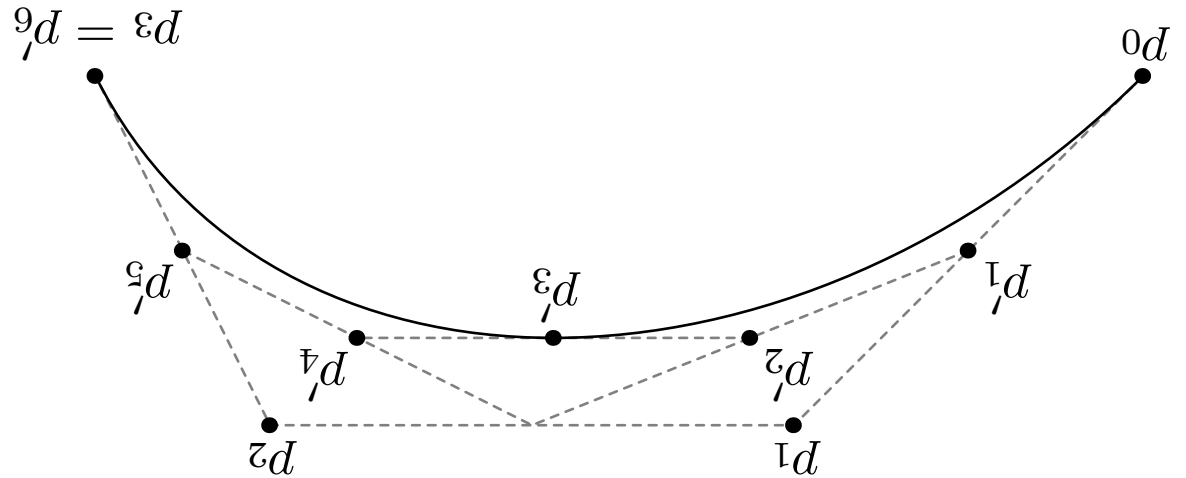
---

[1]footnote text

Figure 4.4: A Larger Figure, Flipped Upside Down

4.4 Cross-referencing chapters and sections

The bookdown documentation is an excellent source for learning how to cross-reference in a bookdown project such as a huskydown document. Here we only cover the most common uses for a typical thesis. If you want something more complex or fancy, please refer to the bookdown documentation and seek help from the developers of that package.

By default, all of your chapter and section headers will get an auto-generated ID label For example, e.g., `# Chapter 1` will have an auto-generated ID `chapter-1`. Note that the ID label is all lower case, and has no spaces. If you have any kind of punctuation in your header, such as a colon (:), it will not appear in the ID label. Then in your text you can reference chapter one in your Rmd file like this: 'as discussed in Chapter `\@ref(chapter-1)`,' which will print as 'as discussed in Chapter 1'

We strongly recommend that you to manually assign ID labels to your chapter header to make it easy to cross-reference. For example, at the top of the Rmd file for this chapter, you can see:

Tables, Graphics, References, and Labels {#ref-labels}

The `{#ref-labels}` part of this header is the ID label. It doesn't show in the output, but is there for us to use for easy cross-referencing, because it can be short, and we don't need to change it elsewhere our document when we update the chapter header. We can use this custom ID label in our Rmd document like this: 'as discussed in Chapter `\@ref(ref-labels)`,' which will print as 'as discussed in Chapter 4.' If you need to show custom text instead of the chapter number, you use this syntax in your Rmd document: `see [my chapter about labels](#ref-labels) for more details` which will appear as 'see my chapter about labels for more details'

To cross-reference a specific section in the same chapter, we recommend adding a custom ID label to the section header, and using that to cross-reference. For example, earlier in this chapter we have a section on tables and in the Rmd file we see `## Tables {#tables}`. We can cross-reference that in the text like this 'as discussed in the section on `[tables](#tables)`' which will appear as 'as discussed in the above section on tables'

To cross-reference a section in a different chapter we can use the ID label from that section directly. For example, we can write in our Rmd document `as discussed in the section on [R code chunks](#r-chunks) in Chapter \@ref(rmd-basics)` which will appear as 'as discussed in the section on R code chunks in Chapter 2.'

If you prefer to cross-reference by the section number, we can use custom ID labels in our Rmd document. For example, to refer to a section in our first chapter, we can write in the Rmd document: `as discussed in section \@ref(r-chunks) in Chapter \@ref(rmd-basics)`. This will appear with

section and chapter numbers like so: as 'as discussed in section **??** in Chapter 2.'

4.5 Bibliographies

Of course you will need to cite things, and you will probably accumulate an armful of sources. There are a variety of tools available for creating a bibliography database (stored with the .bib extension). In addition to BibTeX suggested below, you may want to consider using the free and easy-to-use tool called Zotero. Some Zotero documentation is at `http://libguides.reed.edu/citation/zotero`. In addition, a tutorial is available from Middlebury College at `http://sites.middlebury.edu/zoteromiddlebury/`.

R Markdown uses *pandoc* (`http://pandoc.org/`) to build its bibliographies. One nice caveat of this is that you won't have to do a second compile to load in references as standard LaTeX requires. To cite references in your thesis (after creating your bibliography database), place the reference name inside square brackets and precede it by the "at" symbol. For example, here's a reference to a book about worrying: (Molina & Borkovec, 1994). This `Molina1994` entry appears in a file called `thesis.bib` in the `bib` folder. This bibliography database file was created by a program called BibTeX. You can call this file something else if you like (look at the YAML header in the main .Rmd file) and, by default, is to placed in the `bib` folder.

For more information about BibTeX and bibliographies, see (`http://web.reed.edu/cis/help/latex/index.html`)[2]. There are three pages on this topic: *bibtex* (which talks about using BibTeX, at `http://web.reed.edu/`

---

[2]Reed College (2007)

`cis/help/latex/bibtex.html`), *bibtexstyles* (about how to find and use the bibliography style that best suits your needs, at `http://web.reed.edu/cis/help/latex/bibtexstyles.html`) and *bibman* (which covers how to make and maintain a bibliography by hand, without BibTeX, at `http://web.reed.edu/cis/help/latex/bibman.html`). The last page will not be useful unless you have only a few sources.

If you look at the YAML header at the top of the main .Rmd file you can see that we can specify the style of the bibliography by referencing the appropriate csl file. You can download a variety of different style files at `https://www.zotero.org/styles`. Make sure to download the file into the csl folder.

Tips for Bibliographies

- Like with thesis formatting, the sooner you start compiling your bibliography for something as large as thesis, the better.
- The cite key (a citation's label) needs to be unique from the other entries.
- When you have more than one author or editor, you need to separate each author's name by the word "and" e.g. `Author = {Noble, Sam and Youngberg, Jessica},`.
- Bibliographies made using BibTeX (whether manually or using a manager) accept LaTeX markup, so you can italicize and add symbols as necessary.
- To force capitalization in an article title or where all lowercase is generally used, bracket the capital letter in curly braces.

4.6   Anything else?

If you'd like to see examples of other things in this template, please contact us (email `bmarwick@uw.edu`) with your suggestions. We love to see people using *R Markdown* for their theses, and are happy to help.

Conclusion

If we don't want Conclusion to have a chapter number next to it, we can add the `{-}` attribute.

More info

And here's some other random info: the first paragraph after a chapter title or section head *shouldn't be* indented, because indents are to tell the reader that you're starting a new paragraph. Since that's obvious after a chapter or section title, proper typesetting doesn't add an indent there.

Appendix A

The First Appendix

This first appendix includes all of the R chunks of code that were hidden throughout the document (using the `include = FALSE` chunk tag) to help with readibility and/or setup.

In the main Rmd file

```r
# This chunk ensures that the huskydown
# package is installed and loaded. This
# huskydown package includes the template
# files for the thesis.
if (!require(devtools)) install.packages("devtools",
    repos = "http://cran.rstudio.com")
if (!require(huskydown)) devtools::install_github("benmarwick/huskydown")
library(huskydown)
```

In Chapter 4:

```r
# This chunk ensures that the huskydown
# package is installed and loaded. This
# huskydown package includes the template
# files for the thesis and also two
# functions used for labeling and
```

```r
# referencing
if (!require(devtools)) install.packages("devtools",
    repos = "http://cran.rstudio.com")
if (!require(dplyr)) install.packages("dplyr",
    repos = "http://cran.rstudio.com")
if (!require(ggplot2)) install.packages("ggplot2",
    repos = "http://cran.rstudio.com")
if (!require(ggplot2)) install.packages("bookdown",
    repos = "http://cran.rstudio.com")
if (!require(huskydown)) {
    library(devtools)
    devtools::install_github("benmarwick/huskydown")
}
library(huskydown)
flights <- read.csv("data/flights.csv")
```

Appendix B

The Second Appendix, for Fun

Colophon

This document is set in EB Garamond, Source Code Pro and Lato. The body text is set at 11pt with *lmr*.

It was written in R Markdown and $\LaTeX$, and rendered into PDF using huskydown and bookdown.

This document was typeset using the XeTeX typesetting system, and the University of Washington Thesis class class created by Jim Fox. Under the hood, the University of Washington Thesis LaTeX template is used to ensure that documents conform precisely to submission standards. Other elements of the document formatting source code have been taken from the Latex, Knitr, and RMarkdown templates for UC Berkeley's graduate thesis, and Dissertate: a LaTeX dissertation template to support the production and typesetting of a PhD dissertation at Harvard, Princeton, and NYU

The source files for this thesis, along with all the data files, have been organised into an R package, xxx, which is available at `https://github.com/xxx/xxx`. A hard copy of the thesis can be found in the University of Washington library.

This version of the thesis was generated on 2022-08-11 17:51:13. The repository is currently at this commit:

The computational environment that was used to generate this version is as follows:

```
## - Session info -----------------------------------------
##   setting  value
##   version  R version 4.1.0 (2021-05-18)
##   os       macOS 12.3.1
##   system   x86_64, darwin17.0
##   ui       RStudio
##   language (EN)
##   collate  en_US.UTF-8
##   ctype    en_US.UTF-8
##   tz       America/New_York
##   date     2022-08-11
##   rstudio  1.4.1717 Juliet Rose (desktop)
##   pandoc   2.11.4 @ /Applications/RStudio.app/Contents/MacOS/pandoc/ (via rmarkdown)
##
## - Packages ---------------------------------------------
##   package     * version date (UTC) lib source
##   assertthat    0.2.1   2019-03-21 [2] CRAN (R 4.1.0)
##   bookdown      0.26    2022-04-15 [1] CRAN (R 4.1.2)
##   brio          1.1.3   2021-11-30 [1] CRAN (R 4.1.0)
##   cachem        1.0.6   2021-08-19 [1] CRAN (R 4.1.0)
##   callr         3.7.0   2021-04-20 [2] CRAN (R 4.1.0)
##   cli           3.3.0   2022-04-25 [1] CRAN (R 4.1.2)
##   codetools     0.2-18  2020-11-04 [2] CRAN (R 4.1.0)
##   colorspace    2.0-3   2022-02-21 [1] CRAN (R 4.1.2)
##   crayon        1.5.1   2022-03-26 [1] CRAN (R 4.1.0)
##   DBI           1.1.2   2021-12-20 [1] CRAN (R 4.1.0)
##   desc          1.4.1   2022-03-06 [1] CRAN (R 4.1.2)
##   devtools    * 2.4.3   2021-11-30 [1] CRAN (R 4.1.0)
```

```
## digest       0.6.29  2021-12-01 [1] CRAN (R 4.1.0)
## dplyr       * 1.0.9   2022-04-28 [1] CRAN (R 4.1.2)
## ellipsis      0.3.2   2021-04-29 [2] CRAN (R 4.1.0)
## evaluate      0.15    2022-02-18 [1] CRAN (R 4.1.2)
## fansi         1.0.3   2022-03-24 [1] CRAN (R 4.1.2)
## farver        2.1.1   2022-07-06 [1] CRAN (R 4.1.2)
## fastmap       1.1.0   2021-01-25 [2] CRAN (R 4.1.0)
## formatR       1.12    2022-03-31 [1] CRAN (R 4.1.2)
## fs            1.5.2   2021-12-08 [1] CRAN (R 4.1.0)
## generics      0.1.3   2022-07-05 [1] CRAN (R 4.1.2)
## ggplot2     * 3.3.6   2022-05-03 [2] CRAN (R 4.1.2)
## git2r         0.30.1  2022-03-16 [1] CRAN (R 4.1.2)
## glue          1.6.2   2022-02-24 [1] CRAN (R 4.1.2)
## gtable        0.3.0   2019-03-25 [2] CRAN (R 4.1.0)
## htmltools     0.5.2   2021-08-25 [1] CRAN (R 4.1.0)
## huskydown   * 0.0.5   2022-01-16 [1] Github (benmarwick/huskydown@addb48e)
## knitr       * 1.39    2022-04-26 [1] CRAN (R 4.1.2)
## labeling      0.4.2   2020-10-20 [2] CRAN (R 4.1.0)
## lifecycle     1.0.1   2021-09-24 [1] CRAN (R 4.1.0)
## magrittr      2.0.3   2022-03-30 [1] CRAN (R 4.1.2)
## memoise       2.0.1   2021-11-26 [1] CRAN (R 4.1.0)
## munsell       0.5.0   2018-06-12 [2] CRAN (R 4.1.0)
## pillar        1.7.0   2022-02-01 [1] CRAN (R 4.1.2)
## pkgbuild      1.3.1   2021-12-20 [1] CRAN (R 4.1.0)
## pkgconfig     2.0.3   2019-09-22 [2] CRAN (R 4.1.0)
## pkgload       1.2.4   2021-11-30 [1] CRAN (R 4.1.0)
## prettyunits   1.1.1   2020-01-24 [2] CRAN (R 4.1.0)
## processx      3.5.3   2022-03-25 [2] CRAN (R 4.1.2)
## ps            1.7.0   2022-04-23 [2] CRAN (R 4.1.2)
## purrr         0.3.4   2020-04-17 [2] CRAN (R 4.1.0)
## R6            2.5.1   2021-08-19 [1] CRAN (R 4.1.0)
```

```
##   remotes      2.4.2    2021-11-30 [1] CRAN (R 4.1.0)
##   rlang        1.0.4    2022-07-12 [1] CRAN (R 4.1.0)
##   rmarkdown    2.14     2022-04-25 [1] CRAN (R 4.1.2)
##   rprojroot    2.0.3    2022-04-02 [2] CRAN (R 4.1.2)
##   rstudioapi   0.13     2020-11-12 [2] CRAN (R 4.1.0)
##   scales       1.2.0    2022-04-13 [2] CRAN (R 4.1.2)
##   sessioninfo  1.2.2    2021-12-06 [1] CRAN (R 4.1.0)
##   stringi      1.7.8    2022-07-11 [1] CRAN (R 4.1.0)
##   stringr      1.4.0    2019-02-10 [2] CRAN (R 4.1.0)
##   testthat     3.1.4    2022-04-26 [1] CRAN (R 4.1.2)
##   tibble       3.1.7    2022-05-03 [1] CRAN (R 4.1.2)
##   tidyselect   1.1.2    2022-02-21 [1] CRAN (R 4.1.0)
##   tinytex      0.39     2022-05-16 [1] CRAN (R 4.1.2)
##   usethis    * 2.1.6    2022-05-25 [1] CRAN (R 4.1.2)
##   utf8         1.2.2    2021-07-24 [2] CRAN (R 4.1.0)
##   vctrs        0.4.1    2022-04-13 [1] CRAN (R 4.1.2)
##   withr        2.5.0    2022-03-03 [1] CRAN (R 4.1.2)
##   xfun         0.31     2022-05-10 [1] CRAN (R 4.1.2)
##   yaml         2.3.5    2022-02-21 [1] CRAN (R 4.1.2)
##
##  [1] /Users/dbradford4/Library/R/x86_64/4.1/library
##  [2] /Library/Frameworks/R.framework/Versions/4.1/Resources/library
##
## ---------------------------------------------------------
```

References

Angel, E. (2000). *Interactive computer graphics : A top-down approach with OpenGL.* Boston, MA: Addison Wesley Longman.

Angel, E. (2001a). *Batch-file computer graphics : A bottom-up approach with QuickTime.* Boston, MA: Wesley Addison Longman.

Angel, E. (2001b). *Test second book by angel.* Boston, MA: Wesley Addison Longman.

Bendix, F., Kosara, R., & Hauser, H. (2005). Parallel Sets: Visual Analysis of Categorical Data. In *2005 IEEE symposium on information visualization (INFOVIS'05)* (pp. 133–140). IEEE. http://doi.org/10.1109/INFOVIS.2005.27

d'Ocagne, M. (1885). Coordonnées parallèles et axiales : Méthode de transformation géométrique et procédé nouveau de calcul graphique déduits de la considération des coordonnées parallèles. *Gauthier-Villars*, 112. Retrieved from https://archive.org/details/coordonnesparal00ocaggoog/page/n10

Day, R. H., & Stecher, E. J. (1991). Sine of an illusion. *Perception*, *20*, 49–55.

DeDonno, M. A. (2016). The influence of IQ on pure discovery and guided discovery learning of a complex real-world task. *Learning and Individual Differences*, *49*, 11–16.

Few, S. (2006). *Information dashboard design: The effective visual communication of data.* Newton, MA: O'Reilly Media, Inc.

Heinrich, J., & Weiskopf, D. (2009). Continuous Parallel Coordinates. *IEEE Transactions on Visualization and Computer Graphics*, *15*(6), 1531–1538. http://doi.org/10.1109/TVCG.2009.131

Hofmann, H., & Vendettuoli, M. (2013). Common Angle Plots as Perception-True Visualizations of Categorical Associations. *IEEE Transactions on Visualization and Computer Graphics*, *19*(12), 2297–2305. http://doi.org/10.1109/TVCG.2013.140

Lee, S., Kim, S.-H., Hung, Y.-H., Lam, H., Kang, Y., & Yi, J. S. (2016). How do people make sense of unfamiliar visualizations?: A grounded model of novice's information visualization sensemaking. *IEEE*, *22*, 499–508.

Luxen, D., & Vetter, C. (2011). Real-time routing with OpenStreetMap data. In *Proceedings of the 19th ACM SIGSPATIAL international conference on advances in geographic information systems* (pp. 513–516). New York, NY, USA: ACM. http://doi.org/10.1145/2093973.2094062

Molina, S. T., & Borkovec, T. D. (1994). The Penn State worry questionnaire: Psychometric properties and associated characteristics. In G. C. L. Davey & F. Tallis (Eds.), *Worrying: Perspectives on theory, assessment and treatment* (pp. 265–283). New York: Wiley.

National Center for Education Statistics. (2020). National center for education statistics. https://nces.ed.gov/.

Peters, D. J. (2019). Community Resiliency in Declining Small Towns: Impact of Population Loss on Quality of Life over 20 Years. *Rural Sociology*, *84*(4), 635–668. http://doi.org/10.1111/ruso.12261

R Core Team. (2022). *R: A language and environment for statistical computing.* Vienna, Austria: R Foundation for Statistical Computing. Retrieved from https://www.R-project.org/

Reed College. (2007). LaTeX your document. Retrieved from http://web.reed.edu/cis/help/LaTeX/index.html

Rural Shrink Smart Team. (2022). Rural shrink smart. https://ruralshrinksmart.org/.

Sarikaya, A., Correll, M., Bartram, L., Tory, M., & Fisher, D. (2019). What do we talk about when we talk about dashboards? *IEEE Transactions on Visualization and Computer Graphics*, *25*(1), 682–692. http://doi.org/10.1109/TVCG.2018.2864903

State of Iowa. (2020). Iowa data portal. https://data.iowa.gov.

USDA - ERS. (2020a). Rural classifications. https://www.ers.usda.gov/topics/rural-economy-population/rural-classifications/.

USDA - ERS. (2020b). Rural-urban commuting area codes. https://www.ers.usda.gov/data-products/rural-urban-commuting-area-codes/.

VanderPlas, S., & Hofmann, H. (2015). Signs of the sine illusion–why we need to care. *Journal of Computational and Graphical Statistics*, *24*(4),

1170–1190. http://doi.org/10.1080/10618600.2014.951547

Wickham, H. (2016). *ggplot2: Elegant Graphics for Data Analysis* (2nd ed.). Springer-Verlag New York. Retrieved from `https://ggplot2.tidyverse.org`

Wickham, H., Cook, D., Hofmann, H., & Buja, A. (2011). tourr: An R Package for Exploring Multivariate Data with Projections. *Journal of Statistical Software, Articles, 40*(2), 1–18. http://doi.org/10.18637/jss.v040.i02

Zarecor, K. E., Peters, D. J., & Hamideh, S. (2021). Rural smart shrinkage and perceptions of quality of life in the american midwest. In *Handbook of quality of life and sustainability* (pp. 395–415). Springer.