Doing Democratic Data Analysis

Corban Nemeth

2020-06-06

Contents

Preface

I believe that data, *in the hands of public administrators and policy analysts*, has the power to transform the way government works.

Big questions will, and should, be asked of big data— the role of government in regulating algorithmic bias, facial recognition, and consumer data privacy is a vital conversation. However, these topics should not detract or deter public administrators and policy analysts from leaning into **small data** for decision-making purposes.

Public administrators and analysts who are data literate will be able to make and inform better decisions while avoiding the pitfalls posed by the latest technological trends. This book represents an opportunity for public administrators and policy analysts to join their subject matter expertise with foundation principles and practices of democratic data analysis— data analysis that is **transparent, relevant, and grounded in the context of ethical and effective governance**.

In this guide, I present an **opinionated framework for data analysis in public sector organizations**. By opinionated, I mean that I will teach you what I think is the right way to do things given my own experience as a public sector policy and data analyst. Your experience might differ– and that's great! I hope that where you can use your experience in place of mine, you do to the fullest extent. With that in mind, it is often said that you have to know the rules to break them, so I will teach you the "rules" of data analysis as I understand them, applied to common situations in public sector organizations.

Chapter 1

Introduction

> Data analysis is a process of inspecting, cleansing, transforming, and modeling data with the goal of discovering useful information, informing conclusions and supporting decision-making. - *Wikipedia*

1.1 Principles and Practices

Nobody *teaches* Excel anymore. At least, that's been my experience in public sector organizations. For many folks, data analysis and Excel are synonymous. And most often– again, in my experience– that consists of finding the "Excel Person" (usually one of two types of people– the young'un who is also the *"Sharepoint/Teams/How-Do-I-Use-This-New-Technology?"* guru, or long-timer who started using Lotus 1-2-3 when Ronald Reagan was president). Now, there is nothing wrong with being the "Excel person" (as you may have guessed, this is a role that I fill frequently). However, I strongly believe there is value in intentionally learning how to analyze data outside of just compiling survey results from your manager or making a chart from a canned report. And there is so much more out there than traditional Excel-based tools.

The public sector increasingly runs on data. *Data-driven decision making. Objective. Based on the facts.* Too often, these terms are meaningless platitudes thrown around to discard dissenting opinions. The truth is, **the data can never answer your question**. Only you can answer your question. Data can certainly help. But **data can never speak for themselves**. Data, and data analysis, is always interpreted through humans, and humans are inherently messy decisionmakers who weigh experience, intuition, and heuristics when making the call.

This may seem like a strange way to introduce a book on data analysis. But I would argue that the messiness of decisionmaking is what makes democratic

data analysis all the more valuable. In this handbook, I will try to convince you that data analysis is worth doing purposely, especially if you are someone who does not consider yourself a data analyst.

1.2 What is data analysis?

Data analysis, in the way I am using the term, is the process of examining, transforming, and modeling collected facts and figures on a screen to insight into the real world. It's looking data and gaining insight from it. Data analysis can be as simple as adding totals into a column to see cumulative effects, or as complicated as time-series forecasting. But fundamentally, all data analysis is taking inputs and applying those inputs to the real world to gain insight into the real world. It also may be helpful to think about what data analysis *isn't*:

1. Data analysis isn't math.

- Calculations are great, but `a7 + b8` in Excel is deterministic. It gives you one answer. This book is not interested in data analysis that gives you the right answer, because there is no such thing. There are many answers to many questions, depending on how those questions are asked and how the data is analyzed.

2. Data analysis isn't statistics.

- This book is about reading and telling the story of your data in a way that can complement expertise and experience to make better decisions. Statistics are often used as a cheap stand-in for domain expertise and are often abused in favor of trusting the analyst or administrator to back up their assumptions with both quantitative and qualitative data.

3. Data analysis isn't research methods.

- No set of tools and practices can stand in for asking the right questions, and transforming data into information to answer that question. This book will give you the tools to work with your quantitative data to answer relevant questions, but all good analysis begins with a good question.

1.3 What is democratic data analysis?

I propose the following four principles of democratic data analysis. Democratic data analysis is:

- Tidy
- Pivot-able
- Reproducible,
- Uncertainty-oriented, and
- Audience-focused

Why focus on principles rather then specific tutorials? Data analysis is enabled by the technologies that we have access to. Whether it be the venerable pivot table, or a new-school dashboard platform, or a data-oriented programming language, the principles that I lay out in this handbook supersede specific technologies. Think of it like grammar. You may write by hand, on a computer, using text-to-speech. You may be writing a poem, a novel, an argument, or an instruction manual. But the basic rules of grammar are relevant in whatever medium you choose.

Similarly, this guide will teach you the basic "grammar" of democratic data analysis. This will allow you to apply this knowledge in whatever platform or technology you are interested in or have access to. But similar to learning language, it helps to practice. It isn't much to use to study grammar without ever writing a sentence.

1.4 Tools and Techniques

The principles section of this guide will include examples in both Excel and `R`. Government runs on Excel, so all of the examples and exercises will be Excel compatible. If you are comfortable with Excel[1] and want to challenge yourself, boost your resume, and become a data superstar, I would highly recommend learning `R`. But run what you brung, as they say.

And a note on Excel– if you are comfortable in Excel, use modern Excel tools! Institutional knowledge and inertia are strong in large organizations, and there are extremely capable tools sitting under your nose. The Microsoft "Power" (PowerPivot, PowerQuery, PowerBI) Excel stack integrate neatly into the principles that I outline here. Adopting these tools was how I started on the path of incorporating reproducible and democratic data analysis in public sector organizations. By adopting these tools, you will quickly learn why the principles outlined here are important, as you are forced to think about tabular data and calculated summaries. But more on that later. To start, here is a quick start guide to Power Query/Get and Transform and Power Pivot/DAX.

[1] aka you use `vlookup`, `index(match)`, pivot tables, or *Get & Transform*/PowerPivot on a somewhat regular basis

Chapter 2

Tidy Data

Tidy data refers to having your data organized in a specific manner suitable for analysis. This chapter will walk through common problems and approaches to cleaning and tidying your data. Keeping your data in a tidy format for analysis will help because it is a fundamentally flexible way of working with data. Keeping scattered, lose data in spreadsheets is a sure way to cause confusion for yourself and others.

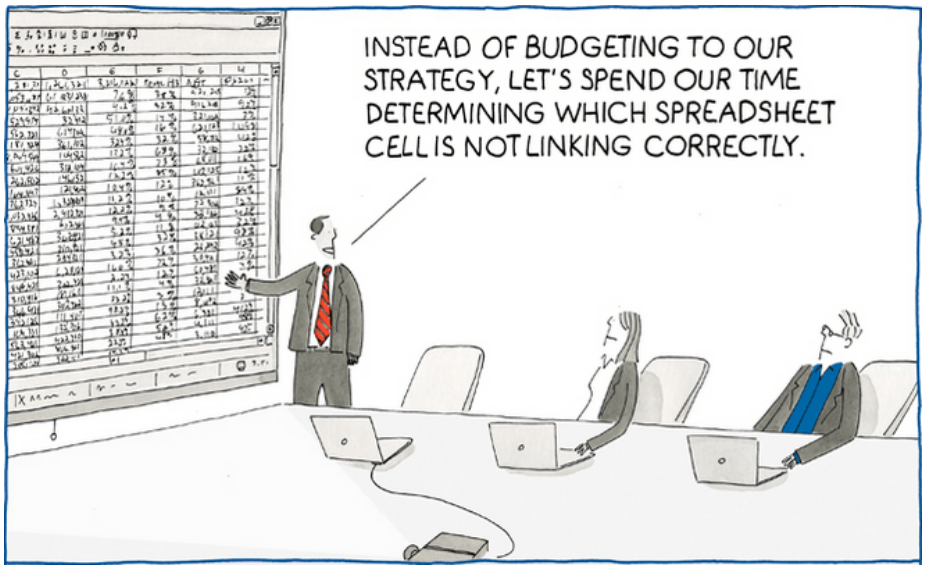


Figure 2.1: spreadsheet hell

*Image courtesy of hive9.com

2.1 The Grammar of Tidy Data[1]

Democratic data analysis has an underlying structure, like a sentence. Rules exist so sentences (hopefully) make sense to you, the reader. Similarly, by following common conventions of democratic data analysis, others will be able to "read" your analysis like you are reading this sentence. And also, like grammar, you can and should break the rules– but it helps to know them first.

Here are a couple definitions that will help as you move through this text. Don't worry about memorizing them, as I will refer back to these definitions frequently.

2.1.1 Fields

A field is a fancy name for a column.

From here on out, **every calculation, manipulation, formula, will be operated on a field**.

I want you to forget that you could ever modify a lonely cell in Excel. No more formulas in cells. No more typing in values to a cell. Certainly no more writing over data in a cell. And for the love of everything that is good, no more color coding cells. Democratic data analysis depends on formulas that work on entire fields. Everything you would need to do to a single cell in Excel can– and should!– be done to an entire column. This may sound extreme and restrictive, but this approach will pay big dividends in your democratic data analysis journey and ultimately allow you to be more creative in your analysis.

2.1.2 Variables

A variable is something in your data that can change. That's it! Variables become very important when looking at how to structure your data. Each variable should have its own field (column).

2.1.3 Observations

Observations make up the rows of your data set. Each observation should correspond to a specific "thing." This will make more sense later, I promise.

2.1.4 Values

Values are the actual data in your table.

[1] Adapted from Hadley Wickham's paper on Tidy Data

Each value belongs to 1 (one) (uno) observation and 1 (one) variable. That's it.

This means, among other things, no more merged cells, merged rows, or groups. A one-to-one relationship between fields and observations is critical for getting the most out of your data analysis.

2.1.5 Tables

A table is the grouping of all observations of a similar type.

The table sets the foundation for analyzing your data. **If I could only convince you of one thing, it would be the value in keeping your data in a table-based excel format**. If you are trying to do math or statistics (again, not data analysis. These things are different!), then the table based framework can sometimes fall short. But if you are concerned with working with data to draw conclusions, then tables are the way to go.

You may already be able to see how these definitions foreshadow some of what is coming in later sections. For example, there are no references to cells. This is intentional. The most important distinction between democratic data analysis and simply working in excel is that in democratic data analysis, (virtually) *everything* is done on the `field` level. Changes are made to entire columns, calculations are made on entire columns. Thinking in `fields` is the first step on the path to democratic data enlightenment. Having data formatted in the structure outlined above forces good data hygiene that will be very helpful later on when discussing reproduciblity and pivot-ability.

2.2 Cleaning and Tidying

I hate cleaning, but love tidying. Unfortunately, as in with life, one must clean before one tidies. But let's start with some conceptual definitions.

Cleaning refers to the process of scrubbing the data into a way that makes sense to you, the analyst. Oftentimes, and especially in public sector organizations, the data is not clean. Whether you are looking at the output of a Survey Monkey survey or a canned report that is run from the IT department, your data will come in all shapes and sizes. Cleaning data is the process of transforming data values into ones that make sense for the purposes of your analysis.

Here is the first major departure from what you may have been taught about data analysis in Excel. When you get messy data *do not* change individual cell values. Recall from the introductory chapter the difference between cells and fields. Fields, as a reminder, are columns that represent one variable. Use data analysis tools to make changes to the entire field, rather than specific cells.

Most data analysis software, outside of Excel, make it difficult or impossible to change individual cell values. At first glance, this can seem unnecessary,

limiting, and supremely annoying. But it's actually very helpful. Thinking in terms of fields, and making changes to entire fields, will save you *a lot* of work and headache in the long run.

Let's look at a sample dataset that may be similar to one you would encounter in real life. Here is a survey collected by a field manager of a local parks and recreation department on employment.

```
library(tidyverse)
library(DT)

sites <- tribble(
  ~"Employee", ~"Location", ~"Telecommute?", ~"Hire Date",
  "ron swanson", "Pawnee City Hall", "never", "Unknown",
  "Knope, Leslie", "Field Duty", "1 day/week", "2011-6-1",
  "Andy Dwyer", "sullivan street pit", "40 hours", "March 1, 2013",
  "Jerry Gergich", "City Hall", "never", "6/1/1985",
  "Garry Gergich", "City Hall", "never", "6/1/1985",
  "ben wyatt", "Partridge, Minnesota","" , "Jan. 1, 2010"
)

sites %>% datatable(
    extensions = 'Buttons',
    options = list(dom = 'Bfrtip',
                   buttons = 'excel',
                   searching = FALSE))
```

	Employee	Location	Telecommute?	Hire Date
1	ron swanson	Pawnee City Hall	never	Unknown
2	Knope, Leslie	Field Duty	1 day/week	2011-6-1
3	Andy Dwyer	sullivan street pit	40 hours	March 1, 2013
4	Jerry Gergich	City Hall	never	6/1/1985
5	Garry Gergich	City Hall	never	6/1/1985
6	ben wyatt	Partridge, Minnesota		Jan. 1, 2010

Showing 1 to 6 of 6 entries Previous 1 Next

In this example, it would be trivial to go in to the Excel file and clean up the dates, names, and locations by hand. However, you could imagine this survey

replicated for a department of forty employees. It quickly becomes unfeasible to make those edits by hand. When this is the case, there are functions in Excel and R that will make your life much easier.

Here is annotated code for how I would go about cleaning this table in R. The friendly syntax of the `tidyverse` packages makes it easy to follow along, even if you aren't comfortable writing it yourself. You can accomplish all these similar transformations using Get & Transform in Excel.

```
#creating a new table called "sites_cleaned", starting with the old table "sites"
sites_cleaned <- sites %>%
  #switching the order of names that are backwards
  mutate(Employee = if_else(Employee == "Knope, Leslie", "Leslie Knope", Employee)) %>%
  #separate employee names into two columns
  separate(Employee, into = c("first_name", "last_name")) %>%
  #renaming column names to standard format
  rename(location = Location,
         telecommute_hours =`Telecommute?`,
         hire_date = `Hire Date`) %>%
  #changing first and last names and locations to Title Case
  mutate(first_name = str_to_title(first_name),
         last_name = str_to_title(last_name),
         location = str_to_title(location)) %>%
  #coding location data to three categories, In Office, In Field, or Other
  mutate(location = case_when(
           str_detect(location, "City Hall") ~ "In Office",
           str_detect(location, "Field") ~ "In Field",
           str_detect(location, "Street") ~ "In Field",
           TRUE ~ "Other"),
  #coding hours to numeric
         telecommute_hours = case_when(
           telecommute_hours == "never" ~ 0,
           telecommute_hours == "1 day/week" ~ 8,
           telecommute_hours == "40 hours" ~ 40
         )
         )

#print to datatable
sites_cleaned%>% datatable(
    extensions = 'Buttons',
    options = list(dom = 'Bfrtip',
                   buttons = 'excel',
                   searching = FALSE))
```

	first_name	last_name	location	telecommute_hours	hire_date
1	Ron	Swanson	In Office	0	Unknown
2	Leslie	Knope	In Field	8	2011-6-1
3	Andy	Dwyer	In Field	40	March 1, 2013
4	Jerry	Gergich	In Office	0	6/1/1985
5	Garry	Gergich	In Office	0	6/1/1985
6	Ben	Wyatt	Other		Jan. 1, 2010

Showing 1 to 6 of 6 entries Previous 1 Next

This may seem like a lot of work for a small table. But as your data grows, it is much easier to operate on entire fields at a time. This is especially true in Get & Transform, which makes it very easy to transform and clean data using all the same steps as the R code above. As you can see, our table is now "cleaned" and organized consistently.

2.3 Tidying

Tidy data is the way your data should be organized before you begin your analysis. In tidy data, each column is a *variable*, each row is an *observation*, and each table is an *associated set of observations*. What does that mean in practice? Consider the following example.

Below is a table[2] that shows types of retirement visits for a month at a state's Department of Retirement Services by the employee who took the visit and the visit type.

```
#build sample data table
visits <- tribble(
  ~"Employee", ~"Phone Visits", ~"Office Visits", ~"Online Visits",
  "Danielle", 6, 11, 23,
  "Ramona", 11, 5, 18,
  "Ross", 10, 10, 10
)

#print to datatable
```

[2]Data was created for demonstration purposes

```
visits%>% datatable(
    extensions = 'Buttons',
    options = list(dom = 'Bfrtip',
                   buttons = 'excel',
                   searching = FALSE),
                   caption = "Visits to the Dept. of Retirement Services in a given month by empl
                   )
```

Visits to the Dept. of Retirement Services in a given month by employee

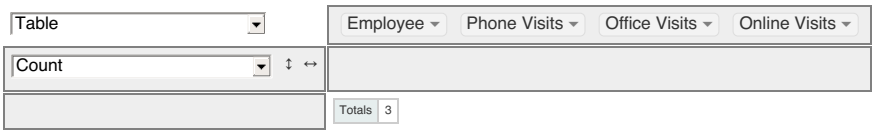
	Employee	Phone Visits	Office Visits	Online Visits
1	Danielle	6	11	23
2	Ramona	11	5	18
3	Ross	10	10	10

Showing 1 to 3 of 3 entries Previous 1 Next

Data are frequently displayed in this "wide" format. It works great for presentation, but not great for data analysis.

The shortcomings of data in this format may become apparent when you attempt to work with the data in a pivot table. This is because our columns aren't truly variables. Remember, variables are elements of an observation that can change. You can drag the fields from the top row to the gray box below, for columns, and the left, for rows. This becomes unmanageable quickly.

```
rpivotTable::rpivotTable(visits, width = "60%", height = "60%")
```



2.4 Tidy Data- From Wide to Long

Let's apply our criteria of tidy data to this set:

- Variables
  - At first glance, it doesn't look like this is a problem. But think again. Is `phone visits` really a variable? Or is the real variable of interest number of visits? And are our column names are actually variables too (type of visit)?

Let's take another swing at setting up our table for data analysis purposes. This can be accomplished easily in R using the code below, or in Excel by loading the data with `Get and Transform` -> selecting the three "visits" columns -> right clicking -> and selecting "unpivot columns."

```
#We have already loaded the "tidyverse" library so we do not have to do it again
#we are editing the "visits" table already created by storing it in a new table pivot_
pivot_visits <- visits %>%
  #using pivot_longer on every column except "employee" and setting the name of the ne
  pivot_longer(-Employee, names_to = "Visit Type", values_to = "Number of Visits")


knitr::kable(pivot_visits, caption = "Visits to the Dept. of Retirement Services in a
```

Now this is a table that is much easier to analyze in an Excel pivot table or with a variety of R functions. However, it does look worse and is less intuitive for human readers. Thankfully, using data in this format, it is easy to recreate the original table for presentation, while also giving a variety of options for formatting and plotting. Use the pivot table below to recreate the original table using the tidy data. *Hint- Instead of Count, select Sum -> Number of Visits as the value field. It is far easier to work with fields when they are in a tidy

Table 2.1: Visits to the Dept. of Retirement Services in a given month

Employee	Visit Type	Number of Visits
Danielle	Phone Visits	6
Danielle	Office Visits	11
Danielle	Online Visits	23
Ramona	Phone Visits	11
Ramona	Office Visits	5
Ramona	Online Visits	18
Ross	Phone Visits	10
Ross	Office Visits	10
Ross	Online Visits	10

format. 99% of the time, tidying your data is taking tables in "wide" format and making them "long". Whenever you have a wide table, examine your columns. Do columns have something in common? If they do, it is likely that you have variables in column titles, and they should be transformed to make your table longer, and thus tidy-er.

Let's introduce a slightly more complicated tidy data problem, using the same base data as before.

```
visits_retirements <- tribble(
  ~"Employee", ~"Phone Visits", ~"Phone Retirements", ~"Office Visits", ~"Office Retirements", ~'
  "Danielle", 6, 4, 11, 8, 23, 15,
  "Ramona", 11, 7, 5, 3, 18, 15,
  "Ross", 10, 8, 10, 7, 10, 9
)
```

Hopefully you will see a similar pattern here. Now, there are three variables: Visit type, number of visits, and number of retirements. Again, this data works fine for presentation but could use tidying to ease in analysis.

```
visits_retirements %>%
  DT::datatable(
    extensions = 'Buttons',
    options = list(dom = 'Bfrtip',
                   buttons = 'excel',
                   searching = FALSE))
```

	Employee	Phone Visits	Phone Retirements	Office Visits	Office Retirements	Online Visits	Online Retirements
1	Danielle	6	4	11	8	23	15
2	Ramona	11	7	5	3	18	15
3	Ross	10	8	10	7	10	9

Showing 1 to 3 of 3 entries Previous 1 Next

Try to tidy this in R or Excel Get and Transform. See this footnote[3] or look at the code if you need a hint.

```
visits_retirements_tidy <- visits_retirements %>%
  pivot_longer(cols = -Employee,
               names_to = c("Visit Location", "Type"),
               names_sep = " ")

print(visits_retirements_tidy)
```

```
## # A tibble: 18 x 4
##    Employee `Visit Location` Type        value
##    <chr>    <chr>            <chr>       <dbl>
##  1 Danielle Phone            Visits          6
##  2 Danielle Phone            Retirements     4
##  3 Danielle Office           Visits         11
##  4 Danielle Office           Retirements     8
##  5 Danielle Online           Visits         23
##  6 Danielle Online           Retirements    15
##  7 Ramona   Phone            Visits         11
##  8 Ramona   Phone            Retirements     7
##  9 Ramona   Office           Visits          5
## 10 Ramona   Office           Retirements     3
## 11 Ramona   Online           Visits         18
## 12 Ramona   Online           Retirements    15
## 13 Ross     Phone            Visits         10
```

[3] powerquery hints

```
## 14 Ross     Phone            Retirements     8
## 15 Ross     Office           Visits         10
## 16 Ross     Office           Retirements     7
## 17 Ross     Online           Visits         10
## 18 Ross     Online           Retirements     9
```

In this case, we actually pivoted too far. It will probably be more useful to have the counts of visits and retirements in their own category. Keep in mind the scope of the observation– It is perfectly valid for each to have their own column, as it is visits and retirements per month.

```
visits_retirements_tidy2 <- visits_retirements_tidy %>%
  pivot_wider(id_cols = c(Employee, `Visit Location`, Type), names_from = Type, values_from = val

print(visits_retirements_tidy2)
```

```
## # A tibble: 9 x 4
##   Employee `Visit Location` Visits Retirements
##   <chr>    <chr>             <dbl>       <dbl>
## 1 Danielle Phone                 6           4
## 2 Danielle Office               11           8
## 3 Danielle Online               23          15
## 4 Ramona   Phone                11           7
## 5 Ramona   Office                5           3
## 6 Ramona   Online               18          15
## 7 Ross     Phone                10           8
## 8 Ross     Office               10           7
## 9 Ross     Online               10           9
```

From here, it is easy to do calculations based on fields, rather than cells. For example, in R or Get and Transform, you could add the following:

```
visits_pct <- visits_retirements_tidy2 %>%
  mutate(pct_retirements = Retirements / Visits)

print(visits_pct)
```

```
## # A tibble: 9 x 5
##   Employee `Visit Location` Visits Retirements pct_retirements
##   <chr>    <chr>             <dbl>       <dbl>           <dbl>
## 1 Danielle Phone                 6           4           0.667
## 2 Danielle Office               11           8           0.727
## 3 Danielle Online               23          15           0.652
## 4 Ramona   Phone                11           7           0.636
```

```
## 5 Ramona   Office              5           3          0.6
## 6 Ramona   Online             18          15          0.833
## 7 Ross     Phone              10           8          0.8
## 8 Ross     Office             10           7          0.7
## 9 Ross     Online             10           9          0.9
```

And then, one of the most useful things you can do is develop formulas by grouping of rows. For example, you may want to know the total number of visits and retirements by retiree, regardless of visit location. That can be accomplished in a pivot table, which we will cover in the next section.

Chapter 3

Thinking in Pivots

Why did we go through all the trouble of tidying data in the last section? So we can pivot. Thinking in terms of pivots, by which I mean fields and summaries, is an important component of doing democratic data analysis.

3.1 Pivot Tables and the Meaning of Everything

Pivot tables offer a common, helpful, and underrated framework for data analysis . If you understand the mechanics of the pivot table, you have everything you need to be a data analysis expert in the technology of your choice. PowerBI/Tableau interactive charts and graphs are simply pivot tables in disguise. Understanding what is needed to make a pivot table effective is the key to the wide world of data analysis.

What is so important about pivot tables? First and foremost, pivot tables force you to think in terms of fields, not in terms of cells. In order for a pivot table to be effective, the data has to be organized in a table. And there is a right and a wrong way to putting data in a table. If your pivot table is not working properly, it is likely because your data isn't tidy.

A pivot table groups data by field and allows the user to drag fields to the rows or columns of the pivot table. This is effective when each field is a variable (something that can change), and each row is a separate observation of some phenomena of interest.

In short, pivot tables depend on **tidy data**.

Recall our sample data from the last chapter:

Table 3.1: Visits to the Dept. of Retirement Services in a given month

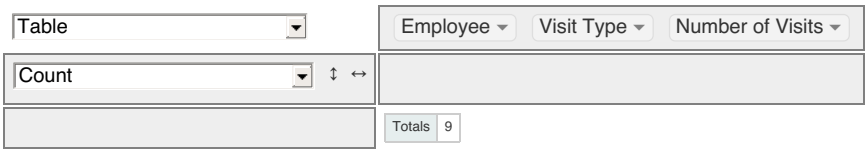
Employee	Visit Type	Number of Visits
Danielle	Phone Visits	6
Danielle	Office Visits	11
Danielle	Online Visits	23
Ramona	Phone Visits	11
Ramona	Office Visits	5
Ramona	Online Visits	18
Ross	Phone Visits	10
Ross	Office Visits	10
Ross	Online Visits	10

```
#We have already loaded the "tidyverse" library so we do not have to do it again
#we are editing the "visits" table already created by storing it in a new table pivot_
pivot_visits <- visits %>%
  #using pivot_longer on every column except "employee" and setting the name of the ne
  pivot_longer(-Employee, names_to = "Visit Type", values_to = "Number of Visits")


knitr::kable(pivot_visits, caption = "Visits to the Dept. of Retirement Services in a
```

Once the data has been tidied, it is easy to manipulate fields in pivot tables.

```
rpivotTable::rpivotTable(pivot_visits, width = "40%", height = "400px")
```



3.2 Groups and Summaries

The key with pivot tables, such as the one above, is you are moving *fields* into the rows and columns of a new table. Hence the term "pivot". You are then selecting a calculation to ***summarize*** the intersection of the two *fields* that you brought into the rows and columns of your pivot table. Above, you can click on the "Count" button to see the other ways this lightweight pivot table allows you to *summarize* your data. More powerful data analysis tools, such as PowerPivot in Excel, PowerBI, and several R packages allow you infinitely flexible formulas to define your ***summaries*** - what happens at the intersection of the fields that you brought into a pivot table. Think in terms of fields and summaries and you are well on your way to effective data analysis.

In a pivot table, the field that you drag to the row becomes your **group**. Grouping allows you to perform summaries on the distinct values of the field that you brought to the row of the pivot table. In the pivot table above, drag the `employee` field to the row. You are now grouping by the distinct **levels** of employee– namely, Danielle, Ross, and Ramona. You will see that the column of the pivot table defaults to `count`. This is true in Excel as well. The pivot table will default to summarizing your data by showing how many times each **level** of your **group** appeared in the field. Most commonly, you will group by the **categorical** elements of your table– the fields with names, rather than numbers.

You can also drag another categorical field to the top of the pivot table to become

Table 3.2: Data summaries of number of visits by visit type

Visit Type	Avg Visits	Total Visits	Std Dev of Visits
Office Visits	8.666667	26	3.214550
Online Visits	17.000000	51	6.557438
Phone Visits	9.000000	27	2.645751

a column. This will now show your data grouped by the two elements you have selected. For example, drag `visit type` to the columns in the above table. You will now see the count of the intersection of those elements. However, I typically do not recommend using the columns of your pivot table for additional categorical groupings. Instead, I recommend dragging both fields to the row section of the pivot table. In the table above, drag `visit type` underneath `employee`. You will now see the data grouped the same way as before, but with only a `totals` column. I find reserving the columns of the pivot table for summaries to be helpful, as it makes things simpler and allows for easy totaling.

Reserving columns for **summaries** makes it easy to change what those summaries are. Excel gives you a variety of options with the `show values as` button. The table above gives you additional options. Remove `visit type` from the rows, and play around with the different summaries that you can apply to the number of visits per employee.

Summarizing your fields also doesn't have to happen within the context of a pivot table. However, the same concepts of pivot tables apply. First, create your groupings (rows). Then, select how you want to summarize your data (mean, median, minimum, maximum, etc.). Then create columns out of those summaries. This same principle applies when creating charts, dashboards, and more complicated forms of modeling.

For example, creating descriptive statistics across groupings is very straightforward. You can replicate the `R` code below with the pivot table above. But the `R` syntax makes it very clear what the data analysis is accomplishing. See if you can replicate the values with the pivot table above.

```
desc_visits <- pivot_visits %>%
  group_by(`Visit Type`) %>%
  summarise(`Avg Visits` = mean(`Number of Visits`),
            `Total Visits` = sum(`Number of Visits`),
            `Std Dev of Visits` = sd(`Number of Visits`))

knitr::kable(desc_visits, caption = "Data summaries of number of visits by visit type")
```

And it is then trivial to change the grouping to generate different summary statistics.

Table 3.3: Data summaries of number of visits by employee

Employee	Avg Visits	Total Visits	Std Dev of Visits
Danielle	13.33333	40	8.736895
Ramona	11.33333	34	6.506407
Ross	10.00000	30	0.000000

```
emp_visits <- pivot_visits %>%
  group_by(Employee) %>%
  summarise(`Avg Visits` = mean(`Number of Visits`),
            `Total Visits` = sum(`Number of Visits`),
            `Std Dev of Visits` = sd(`Number of Visits`))

knitr::kable(emp_visits, caption = "Data summaries of number of visits by employee")
```

This is where pivot tables, and the way they force you to think about data, become *enormously* powerful. Enter **PowerPivot**, stage left. PowerPivot allows you to write very detailed and useful functions for the values that appear in the **summaries** of your pivot table. These summaries, or **calculated fields** or **measures** as they are called in PowerPivot and PowerBI, allow you to define precisely what you want to calculate by group for your data summaries.

Above, I used R to calculate the average number of visits, total visits, and standard deviation of visits by visit type. But what if I quickly wanted to view this by Employee instead? Or what if I wanted to summarize the number of visits as the percent of a whole, or as a proportion to another value? R and PowerPivot make this very easy. I'm not going to go into detail on the mechanics and tutorials of PowerPivot, as it is extensively covered elsewhere– see the Practices section of this guide.

Here's the key– Every fancy data dashboard, analysis technique, interactive visualization– is a pivot table with a defined data **summary** that operates on the group you select. That's it. If you can think in terms of fields and calculated summaries based on those fields, PowerBI, PowerPivot, Tableau, Qlik, ggplot2, etc. will be second nature. The only thing that changes when crating a viz instead of a pivot table is that your fields that you drag into a pivot table become the axis of your chart, and your calculated field becomes a "mark" on your plot. Scroll back up to our sample pivot table and select a different way to view the data than just a table in the upper left hand corner. Our humble pivot table can become a chart, heatmap, or dashboard with little additional effort.

3.3 Conclusion

Thinking in terms of pivot tables is the core of democratic data analysis. This may not feel intuitive at first glance. But thinking this way forces the analyst to avoid the classic pitfalls of *un*democratic data analysis. Those pitfalls are the cell-based, hard-coded, pages and pages of tabs excel workbook nightmare that only the analyst themselves understands (and that's if they are lucky). Thinking in terms of the language of data analysis, emphasizing the use of fields and summaries, forces your analysis to be flexible, understandable, and reproducible. It is easy for an analyst who is also fluent in data analysis to pick up your work and immediately see how it operates because it is constructed using best practices and principles. It is also easy for someone else, or your future self, to write new calculations for data summaries, compare results, and use different fields.

Chapter 4

Reproducible Analysis

"And that's why you always leave a note." — J. Walter Weatherman

4.1 Do It For Your Future Self

Many things take more time to do up front, but save you from massive headaches down the road. Brushing your teeth. Oil changes. Preventative maintenance is the name of the game. The same thing applies in democratic data analysis. Learning how to brush the teeth of your analysis will pay massive dividends down the road, as someone else (or you, more likely), need to go back through and understand, replicate, or validate your findings.

The second principle of democratic data analysis is reproducability. By this, I mean anything that makes it easy to look at your analysis and understand what is going on. This is where classic data analysis in Excel falls short. I believe it is almost a universal experience in the public sector to receive a workbook full of broken links, formulas pointing in every direction, and no sense of where the original data is or what has happened to it since.

In thinking about creating reproducible data analysis, it is important to keep in mind that data analysis should be structured from beginning to end, like a story. In the beginning, there is raw data that you pulled from a report, compiled yourself, or otherwise received. In Act 1, you use the practices we learned in the previous section to make the raw data tidy– without destroying the original data. You should use tools that allow to to non-destructively manipulate and iterate on your data. Both Get & Transform and R allow you to do this by default. In Act 2, which will be the next chapter, you use your data to create a picture of the world before you share it with others in the final Act 3.

The practices of reproducability that you will use here apply throughout the other chapters. It may seem like a waste of time, but if you have ever come

back to a complicated excel workbook after spending even days away, this will make your life much easier.

4.1.1 Do Not Destroy

As I mentioned before, the existential dread that occurs when opening a colleague's workbook and immediately receiving broken links, color-coding[1], and a spiderweb of formulas may be a universal experience in the public sector. But there is a better way to do things. Reproducible analysis is linear. It progresses in a certain direction– from data load to final analysis. Things happen discretely. The blessing and curse of spreadsheets is that they are unbound by time. There is no natural direction, just a sea of little boxes spreading out as far as the eye can see[2]. However, there are ways to impose a linear structure to your analysis.

The first thing I want to emphasize is PLEASE DO NOT DESTROY, ALTER, OR MANIPULATE YOUR UNDERLYING DATA. Your underlying data is like the foundation of your house. Democratic data analysis starts with a foundation of data, and builds on top of it. Often, it seems easier to simply click and drag cells around in an excel workbook, changing values here and there as you see fit. This may work in small use cases, but what if you have another idea? Or come up with a different question, where your data needs to be coded differently? Reproducible analysis makes it substantially easier to revise and rewrite after the fact. If you were writing a well-sourced research article, you wouldn't delete your notes and references for the material that didn't make it into your final product. In the same way, keeping record of the changes that you make to your analysis will pay dividends when your approach changes.

4.1.2 Healthy Habits for Reproducability

I'll start with a list of things you want to avoid in the pursuit of reproducible democratic data analysis

4.1.2.1 Avoid Copy and Paste

If you find yourself copying and pasting values in an excel workbook, you are not engaging in reproducible analysis– full stop. Copy and paste (or worse, cut and paste) doesn't leave breadcrumbs for you or anyone who may come after you. It is incredibly difficult to follow the trail of an analysis built on top of copy-paste.

[1] for the love of democracy, PLEASE do not color code your data

[2] This is where programming languages such as R have an inherent advantage. Code runs in order, from first to last

4.1.2.2 Avoid Repeating Yourself

There is an old adage in programming - *D*on't *R*epeat *Y*ourself. DRY. Keeping your data analysis DRY is a good habit to get in to. If you find yourself repeating the same task more than three times, chances are there is a better, more programmatic way to go about what you are doing.

What do I mean by repeating yourself? This would be going through every row of an 100-row table to add (or remove) a space between words, capitalizing letters, doing specific calculations. All of these tasks can be **easily** automated using virtually all data analysis tools. This not only saves you time, but makes it easier for your reader to see how the data has changed in the course of your analysis.

Get and Transform data tools in Excel allow you to make flexible value transformations on entire fields at a time. This reduces the need for repetitive data cleaning. And if you are already taking my advice and no longer editing individual data cells, you will have a much easier time avoiding repeating yourself.

4.1.2.3 Comment Everything

Comments are wonderful. They are notes to yourself that you should leave at almost every step of your analysis. I frequently do not leave comments. Never have I come back to an uncommented data transformation and been happy with my past self. At worst, leaving comments takes a couple seconds of your time you will never get back. At best, it saves you or your organization from a massive headache when you are able to catch your own errors or update your analysis easier in the future.

4.1.2.3.1 Give Yourself Credit Don't hard code (e.g. manually type) Excel values in cells. Build calculated summaries as discussed in the pivot-powered tab. Link to data sources. Highlight your expertise. This is easier to do when you use the tools shown in the practices section.

4.2 Reproducible Resources

4.2.1 Git/GitHub

If you are inclined to use R or another programming language for data analysis (and you really should be) then git is an essential tool in your toolkit. Full disclosure, even I am not an expert on it. But there are several tools available to make version control using git seamless and effortless.

Git is an automated version control system that backs up your changes to the cloud, typically to github.com or another similar provider. It is very cool. For example, you can examine all the source code for this very book on GitHub.

4.2.2 Code based solutions (even when the code is hidden)

The value of Get & Transform Data is that it forces you to build on top of your data foundation. It also conveniently records each step along the way. You even have the ability to save comments right there in the query editor. This is a remarkably easy and intuitive way to build a data transformation pipeline that will be valuable know and into the future.

The Berkeley Open Policy Analysis Initiative

Open Policy Analysis Guidelines

Chapter 5

Uncertainty Oriented Analysis

All models are wrong, but some are useful– George Box

5.1 Embracing Uncertainty for Better Outcomes

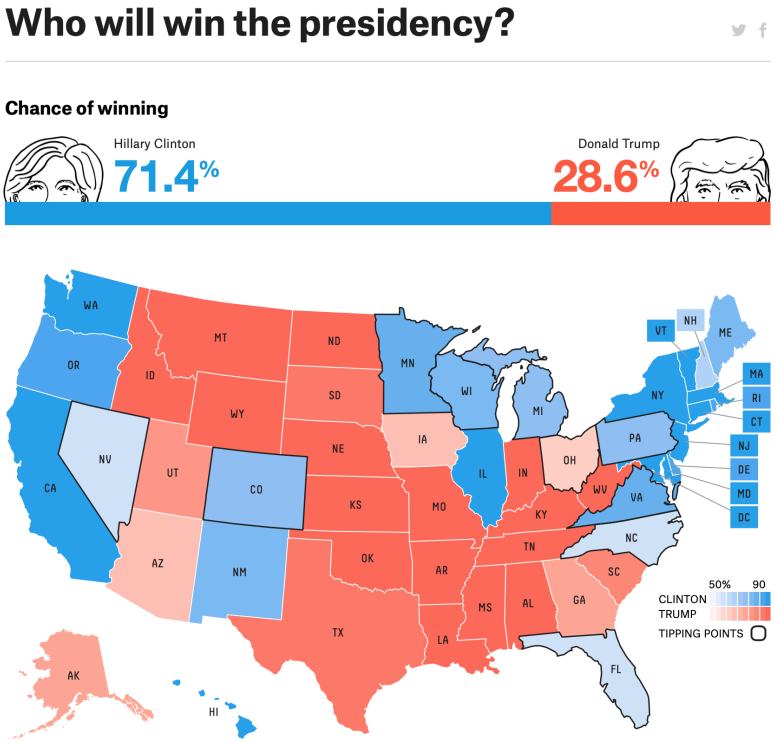
What is the point of data analysis? Often, it is to use data to summarize the world around you. In a sense, all data analysis is model building, and by definition, a model is a simplified version of the world. Any time an analyst is using data analysis to inform decision making, she is in a sense making a model. Model building, and data analysis more generally, never gives you "the" answer. Data analysis only gives you an answer, if it gives you an answer at all.

If you have been following along to this point, you have learned the value of data analysis that is structured and built up, not out. Data analysis is valuable because it tells you what isn't the answer. As in the quote above, knowing the difference between the watches tells you something important. Knowing how to build uncertainty into your analysis is incredibly valuable as models are used to implement important policies that impact the public good. Incorporating uncertainty into your analysis will make you both more credible and force you to use your subject matter expertise in addition to your data skills.

5.2 Incorporating Uncertainty Into Models

As I mentioned before, something as simple as the mean value of a field can be thought of as a model. You may recall from Chapter 3 on pivoting that we walked though many such examples of creating calculated summaries of our data. Am I telling you that a pivot table is a model? Yes. Anything that is a representation of the real world is a model.

Consider the following model of the 2016 presidential election from the popular data journalism website 538. How does, or doesn't, this representation of a model communicate uncertainty? Through the use of the chance of winning bar at the top. How uncertain does this projection *feel* to you? If you are like me, not very uncertain. There are decimal places to the tenth of a percent, state maps, and that big blue bar that communicates authoritatively that Clinton will win the election.



Now consider the following histogram of possible election outcomes. This chart takes a bit more effort to decipher, but it communicates uncertainly much more effectively than the bar and map above. If you look at it, the actual outcome of the 2016 election falls in the thick part of the probability distribution– not an outlier at all.

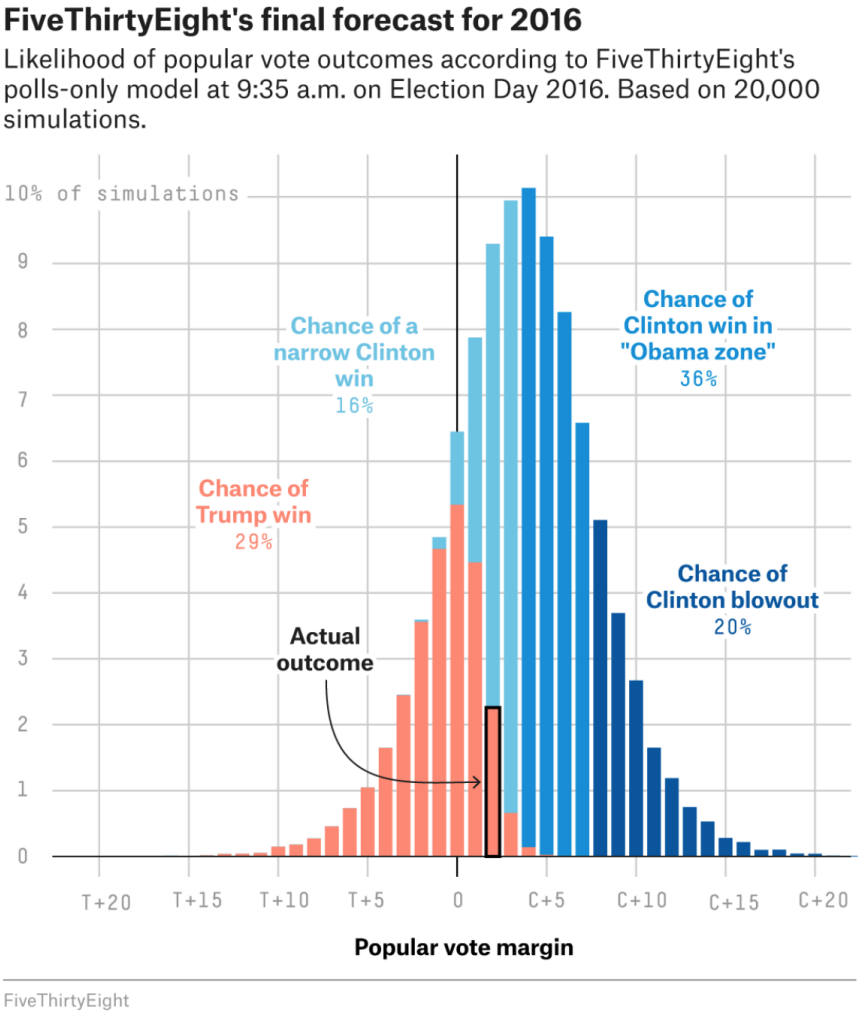


Figure 5.1: distribution of possible results

Thinking critically about how to communicate the uncertainty of your models will make your analysis more democratic by revealing what you do and do not know. I would argue this promotes better decision-making and outcomes than false confidence brought about by hiding, rather than promoting, uncertainty in analysis.

You may be thinking that the election forecasting model above is far and above the data analysis work that you do in your public sector organization. But think back to earlier in this section when we talked about the simplest of models, the average. Does presenting the average of a field show uncertainty? No. If you are making decisions based on averages, then there is a high likelihood that your decisionmaking is highly influenced by outliers, because averages are highly influenced by outliers. How could you communicate an average for decisionmaking purposes, but also show potential outliers? This is where data visualization plays a key role. making sure to always look at your data before presenting summary statistics is vitally important to embedding uncertainty into your analysis. For example, consider Anscombe's Quartet. These remarkable datasets have identical descriptive statistics, yet tell vastly different stories about the world. Consider how you can visualize your data for a decisionmaking audience to avoid the simplifications posed by models that don't tell the whole story.

5.3 How does this impact democratic data analysis?

A model is anything that is used to represent the world. By definition, anything that represents the world simplifies it, because the world is vastly complex. When you are summarizing your data, think critically about what is being simplified. Are there outliers waiting to tell a story, but obscured by a median value? What assumptions are being brought to the table, and what assumptions are taken for granted? Emphasizing and clarifying assumptions, uncertainties, and simplifications makes your analysis stronger, not weaker. This approach allows you to incorporate your expertise as you use data analysis to simplify and storytell for the public good.

Chapter 6

Audience-Focused

Just like writing, your data analysis always has an audience. Whether the audience is you, your coworker, your boss, or a policymaker, knowing your audience helps tailor how you present your findings.

6.1 Considerations for Public Sector Analysis

When sharing data analysis with a policymaker or decision maker, it is important to keep in mind that the audience will often be much wider than initially assumed. Once your analysis is in the hands of a policymaker, it may go to the press, lobbyist organizations, other interest groups, or others inside or outside your organization. Additionally, policymakers often prefer paper or other "hard copy" type analysis, limiting your flexibility to simply share a file. This poses two unique constraints:

1) The mechanics of your analysis are often hidden in a print or a PDF
2) Others will see your work and either take it for granted or want to dissect it.

This is where having tidy, reproducible, and flexible backup becomes incredibly important. By having this backup ready to go as soon as you present or deliver your analysis to the policymaker, you are putting yourself in a strong position to share backup when questions inevitably arise.

If you are sharing your work electronically, then there are several emerging technologies and techniques that make this process easier. It is possible to embed Excel workbooks within Word documents, which ensures that the recipient has access to both your analysis and your methods. However, this can be tricky

for ensuring adequate version control and reproducible, as these workbooks will often link to data that the end user doesn't have access to.

This problem is largely solved by using dashboards and other online solutions. Dashboards are an effective way to share analysis with decision makers because you can present your conclusions and also offer the tailored flexibility for the end user to interact as well. Tools such as PowerBI and Tableau run on the same tidy data and pivot-oriented platform discussed here. It is also straightforward to share the mechanics of the model with anyone who is interested– especially if you follow the best practices for reproducibility outlined earlier. The downside of such dashboard solutions is they can be expensive, proprietary, and it can be difficult to configure them for specific audiences.

6.2 An Introduction to Literate Data Analysis

That brings me to what form democratic data analysis would take in the World-According-To-Corban. Another tool for presenting data analysis to decision-makers are interactive notebooks that merge explanatory text, data, code, and graphics in one file that can be tailored for a variety of audiences. I call this approach ***literate data analysis***[1]. Code-and-text driven notebooks are becoming exceedingly common in academic domains. In fact, this handbook is an example of an interactive HTML document written in `r`[2], but they certainly do not have to be this involved. Interactive notebooks combine text, code, and output in one place. They can be structured so that the file shows either plain-text analysis and charts in a web browser, but all of the code that generates it if you open the file in an editor such as RStudio. For example, here is an R notebook that I authored for a research project on legislative voting behavior, and I'll link to other interactive notebooks in the Practices section.

Here are three platforms of interactive, code-driven data analysis notebooks:

- Rmarkdown notebooks
- Jupyter notebooks
- Observable notebooks

I firmly believe that the notebook approach is the future of policy and data analysis in the public sector. These tools are free, robust, and accessible to those with varying levels of technical expertise. They also inherently solve the challenges of reproducible data analysis that I have been examining here. Version-controlled interactive notebooks are flexible, shareable, editable, and allow both decision-makers and analysts to use the same file to either draw conclusions or validate results.

[1] derived from Literate Programming, but specific to subject matter expert data analysis
[2] using Rmarkdown files and the bookdown package

I dream of a future when budgetary fiscal notes, quantitative bill analysis, and data models in the public sector are written as interactive notebooks, with explanatory text and analysis bundled together, version controlled, and presented with assumptions clearly documented.

This isn't to say that there aren't downsides to these approaches. The largest, as it currently stands, is the learning curve required to use these tools. I have yet to come across a notebook format that is as easily intuitive as an Excel workbook[3]. For now, the biggest downside of these tools is that they require at least a baseline understanding of `r`, `python`, `javascript`, or another similar scripting language. As we have already covered, Excel is a programming language, and probably the world's most common one at that. But Excel is familiar and relatively beginner friendly because it hides the fact that the user is actually programming.

Once that baseline level of understanding is acquired, the notebook format becomes an intuitive output for reports. The same document can result in a PDF file to be shared with decision makers, an interactive document to be shared online, and a complete record of your analysis for a peer. Long story short, I'm excited to work towards integrating the notebook philosophy to the public sector as the next frontier of democratic data analysis.

Yihui Xie, the creator of Rmarkdown, summarizes these principles nicely: > I think notebooks are popular for the same reason that explains the popularity of spreadsheets such as Excel. I haven't met a single software engineer who loves Excel. Everyone hates it and makes fun of it, but why do so many users still use it? Again, Excel makes things tangible. You can touch the data (although it is usually a very bad idea), and draw graphics in a sheet that contains the source data (bad idea again). It makes you feel everything is well under your control: oh here is my data, and here is a graph next to it; oh I should use that column to draw the graph instead, so let me change it and I can see the updated graph immediately.2 You can do everything in a single place, and the short distance between the source (data) and the output is ace.

> Excel makes things tangible at the price of making things messy (e.g., it may contain manually edited data that is hard to keep track of, or merged cells or graphs that make it hard for other software to read the data). By comparison, although notebooks can mess up the state, but that is only an intermediate problem. At its core, it is still relatively clean and encourages the reproducibility principle, i.e., you shall use code to generate results automatically instead of manually copying and pasting results in your report. If you are concerned about the state, you can restart the session and recompile the whole notebook from scratch. Spreadsheets are often hopeless here—you cannot easily restart your brain and redo exactly the same things.

[3]Grid looks promising, but I haven't had a chance to full look into it yet

-Yihui Xie, Creator of Rmarkdown on notebooks and excel

Chapter 7

Applications and Resources

> "I hear the jury's still out on science." - Gob Bluth
>
> "There are dozens of us. Dozens!" - Tobias Fünke

What does democratic data analysis look like in the real world? And how can you, as a public servant, incorporate this into your everyday work? The examples collected below show what democratic data analysis can look like, and how it can be used, to promote transparency and good governance.

7.1 Applications of Democratic Data Analysis

- Dynamic Documentation for Senator Warren's Wealth Tax Policy Analysis
- The Berkeley Open Policy Analysis Initiative
- Open Policy Analysis Guidelines
- Using version control to change laws
- UK Government Data Science repository
- UK publishing official government statistics with Rmarkdown
- The Consumer Financial Protection Bureau on Github
- Federal Spending Transparency
- Make your own tax plan
    - Documentation
- And all of the spreadsheets created by public servants using tidy, pivot-powered, reproducible, uncertainty-oriented, and audience focused principles! Share examples as you are able at dataisfordemocracy@gmail.com.

7.2 Additional Resources

7.2.1 R

- R for Data Science
- R for Journalism
- BBC Journalism Guide to R Graphics
- Graphing in R- ggplot2 cheat sheet, Fundamentals of Data Visualization
- Interactive Documents in R- Guide to Rmarkdown; *learn how to make books and reports like this one!*
- Data Wrangling in R- dplyr for manipulation and tidyr for tidying

7.2.2 Excel

- Getting Started with PowerPivot
- Principles for Data Organization in Spreadsheets
- I am not the only one saying these things, promise!
- Power Pivot Quick Reference Card

7.2.3 Reproduciblity

- ROpenSci