post 01: Data Analysis Cycle

Joanne Chen

10/30/2017

`The Data Analysis Cycle`

Introduction

As our modern day world produces more information, data analysis has become essential in decision making. This process of applying technical skills and statistical theory to describe, visualize, and draw conclusions from data has arisen as a novel and valuable set of skills for people to have. Industries and businesses are relying more heavily on data in order to be successful, and consequently, on the people who understand the steps and proper practices of data analysis. What, you may ask, does “data analysis” even entail? What is this grandiose but vaguely defined term that everyone is excited about? The goal of this blog post is to give a rundown on the data analysis cycle and elucidate the details of this process with code and examples.

Preparation

Before any data set can be properly analyzed, it must undergo massive amounts of preparation. This includes evaluating the quality of the data at hand, and subsequently, tidying and reshaping the data to give us a clean data set that we can use reproducibly. For the purposes of this blog post, I will assume that our data has already been collected with proper sampling procedures and that the information lives inside our directories, ready to be worked with. Data quality assurance is one of the crucial beginning steps to be undertaken before any data set is used for a project or study. This includes data profiling—the systematic analysis of its content—to detect inconsistencies and anomalies, illegal or missing values, misspellings, and other “dirt” in our data. This also entails finding out whether the data is at an appropriate level of detail and granularity for the purposes of the project (often through exploratory data analysis, EDA, which will be demonstrated in later examples). For example, if we wanted to study hourly patterns of number of cars on a highway, a data set with a granularity of daily or weekly records would be useless to us—and data set with a granularity of seconds would be unideal as well. In either case, we would have to either toss the data or reshape it in some way to give us what we need—this is where cleaning, tidying, and reshaping come in. Ensuring data quality eliminates unnecessary dilemmas that we could run into in the future, as well as improves our understanding of our data—never a bad thing. I will now dive into a specific example in which I explore the contents of a real life data set and play around with it.

```
calls = read.csv("data/Berkeley_PD_-_Calls_for_Service.csv")
head(calls)
```

```
##     CASENO              OFFENSE                EVENTDT EVENTTM
## 1 17091420        BURGLARY AUTO 07/23/2017 12:00:00 AM   06:00
## 2 17020462    THEFT FROM PERSON 04/13/2017 12:00:00 AM   08:45
## 3 17050275        BURGLARY AUTO 08/24/2017 12:00:00 AM   18:30
## 4 17019145           GUN/WEAPON 04/06/2017 12:00:00 AM   17:30
## 5 17044993       VEHICLE STOLEN 08/01/2017 12:00:00 AM   18:00
## 6 17037319 BURGLARY RESIDENTIAL 06/28/2017 12:00:00 AM   12:00
##                 CVLEGEND CVDOW               InDbDate
## 1     BURGLARY - VEHICLE     0 08/29/2017 08:28:05 AM
## 2                LARCENY     4 08/29/2017 08:28:00 AM
## 3     BURGLARY - VEHICLE     4 08/29/2017 08:28:06 AM
## 4        WEAPONS OFFENSE     4 08/29/2017 08:27:59 AM
## 5    MOTOR VEHICLE THEFT     2 08/29/2017 08:28:05 AM
## 6 BURGLARY - RESIDENTIAL     3 08/29/2017 08:28:03 AM
##                                                Block_Location
## 1  2500 LE CONTE AVE\nBerkeley, CA\n(37.876965, -122.260544)
## 2  2200 SHATTUCK AVE\nBerkeley, CA\n(37.869363, -122.268028)
## 3 200 UNIVERSITY AVE\nBerkeley, CA\n(37.865491, -122.310065)
## 4    1900 SEVENTH ST\nBerkeley, CA\n(37.869318, -122.296984)
## 5     100 PARKSIDE DR\nBerkeley, CA\n(37.854247, -122.24375)
## 6     1500 PRINCE ST\nBerkeley, CA\n(37.851503, -122.278518)
##              BLKADDR     City State
## 1  2500 LE CONTE AVE Berkeley    CA
## 2  2200 SHATTUCK AVE Berkeley    CA
## 3 200 UNIVERSITY AVE Berkeley    CA
## 4    1900 SEVENTH ST Berkeley    CA
## 5    100 PARKSIDE DR Berkeley    CA
## 6     1500 PRINCE ST Berkeley    CA
```

Part of the process of getting to know our data and gauging its applicability is done by simply looking at the structure of our data and displaying its contents. Above I read in a dataset of Berkeley UCPD calls for service into a Python table. Each record is an individual call; the columns include information such as the time of the call, the offense reporting, location, etc. At first glance, it seems that the OFFENSE and CVLEGEND columns have similar content. However, we can use row grouping methods to discover that CVLEGEND is a broader category, and that OFFENSE contains more detailed information.

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
calls %>%
  filter(calls['CVLEGEND'] == "LARCENY") %>% # Limit rows to one entry within CVLEGEND
  group_by(OFFENSE) # Group resulting rows by OFFENSE
```

```
## # A tibble: 1,102 x 11
## # Groups:   OFFENSE [3]
##      CASENO                  OFFENSE                EVENTDT EVENTTM
##       <int>                   <fctr>                 <fctr>  <fctr>
##  1 17020462        THEFT FROM PERSON 04/13/2017 12:00:00 AM   08:45
##  2 17090841 THEFT MISD. (UNDER $950) 04/26/2017 12:00:00 AM   15:00
##  3 17090794 THEFT MISD. (UNDER $950) 04/20/2017 12:00:00 AM   12:00
##  4 17014095 THEFT FELONY (OVER $950) 03/13/2017 12:00:00 AM   16:45
##  5 17049346 THEFT MISD. (UNDER $950) 08/20/2017 12:00:00 AM   23:20
##  6 17091319 THEFT MISD. (UNDER $950) 07/09/2017 12:00:00 AM   04:15
##  7 17043329 THEFT FELONY (OVER $950) 07/25/2017 12:00:00 AM   16:00
##  8 17029195 THEFT MISD. (UNDER $950) 05/23/2017 12:00:00 AM   12:30
##  9 17046188 THEFT MISD. (UNDER $950) 08/06/2017 12:00:00 AM   23:30
## 10 17041623 THEFT MISD. (UNDER $950) 07/16/2017 12:00:00 AM   13:00
## # ... with 1,092 more rows, and 7 more variables: CVLEGEND <fctr>,
## #   CVDOW <int>, InDbDate <fctr>, Block_Location <fctr>, BLKADDR <fctr>,
## #   City <fctr>, State <fctr>
```

One can see from the table produced above that under a single entry in CVLEGEND, “LARCENY”, there are three subcategories in OFFENSE. If our study was centered around, say, the relationship between types of offenses and their locations, it may now be appropriate to group our table by the OFFENSE column rather than by individual call ID. This is an example of how exploratory data analysis is used to better understand our data before deciding whether it is appropriate for our use.

As mentioned earlier, another good practice in data quality checking is ensuring that there are no missing values. The particular data frame that we are working with does not seem to have NA values, because the output of the following code chunk does not produce anything:

```
calls[is.na(calls)]
```

```
## character(0)
```

Another step we can take to elucidate our data is to assign meaningful information to the entries in the CVDOW column (since they aren’t very informative at the moment). According to the data source, the numbers (0-6) are actually the days of the week. Before we jump into analyzing the dataset it may be helpful to replace CVDOW with strings of the week days.

```
new_cvdow = calls["CVDOW"] %>% # Replace every numeric entry with a day of the week.
replace(calls['CVDOW']==0, "Sunday") %>%
  replace(calls['CVDOW']==1, "Monday") %>%
  replace(calls['CVDOW']==2, "Tuesday") %>%
  replace(calls['CVDOW']==3, "Wednesday") %>%
  replace(calls['CVDOW']==4, "Thursday") %>%
  replace(calls['CVDOW']==5, "Friday") %>%
  replace(calls['CVDOW']==6, "Saturday")
calls["CVDOW"] = new_cvdow # Replace the CVDOW row in calls data frame with the new CVDOW.
head(calls)
```

```
##     CASENO              OFFENSE                EVENTDT EVENTTM
## 1 17091420        BURGLARY AUTO 07/23/2017 12:00:00 AM   06:00
## 2 17020462    THEFT FROM PERSON 04/13/2017 12:00:00 AM   08:45
## 3 17050275        BURGLARY AUTO 08/24/2017 12:00:00 AM   18:30
## 4 17019145           GUN/WEAPON 04/06/2017 12:00:00 AM   17:30
## 5 17044993       VEHICLE STOLEN 08/01/2017 12:00:00 AM   18:00
## 6 17037319 BURGLARY RESIDENTIAL 06/28/2017 12:00:00 AM   12:00
##                  CVLEGEND     CVDOW               InDbDate
## 1     BURGLARY - VEHICLE    Sunday 08/29/2017 08:28:05 AM
## 2                LARCENY  Thursday 08/29/2017 08:28:00 AM
## 3     BURGLARY - VEHICLE  Thursday 08/29/2017 08:28:06 AM
## 4        WEAPONS OFFENSE  Thursday 08/29/2017 08:27:59 AM
## 5    MOTOR VEHICLE THEFT   Tuesday 08/29/2017 08:28:05 AM
## 6 BURGLARY - RESIDENTIAL Wednesday 08/29/2017 08:28:03 AM
##                                                Block_Location
## 1  2500 LE CONTE AVE\nBerkeley, CA\n(37.876965, -122.260544)
## 2  2200 SHATTUCK AVE\nBerkeley, CA\n(37.869363, -122.268028)
## 3 200 UNIVERSITY AVE\nBerkeley, CA\n(37.865491, -122.310065)
## 4    1900 SEVENTH ST\nBerkeley, CA\n(37.869318, -122.296984)
## 5     100 PARKSIDE DR\nBerkeley, CA\n(37.854247, -122.24375)
## 6     1500 PRINCE ST\nBerkeley, CA\n(37.851503, -122.278518)
##              BLKADDR     City State
## 1  2500 LE CONTE AVE Berkeley    CA
## 2  2200 SHATTUCK AVE Berkeley    CA
## 3 200 UNIVERSITY AVE Berkeley    CA
## 4    1900 SEVENTH ST Berkeley    CA
## 5    100 PARKSIDE DR Berkeley    CA
## 6     1500 PRINCE ST Berkeley    CA
```

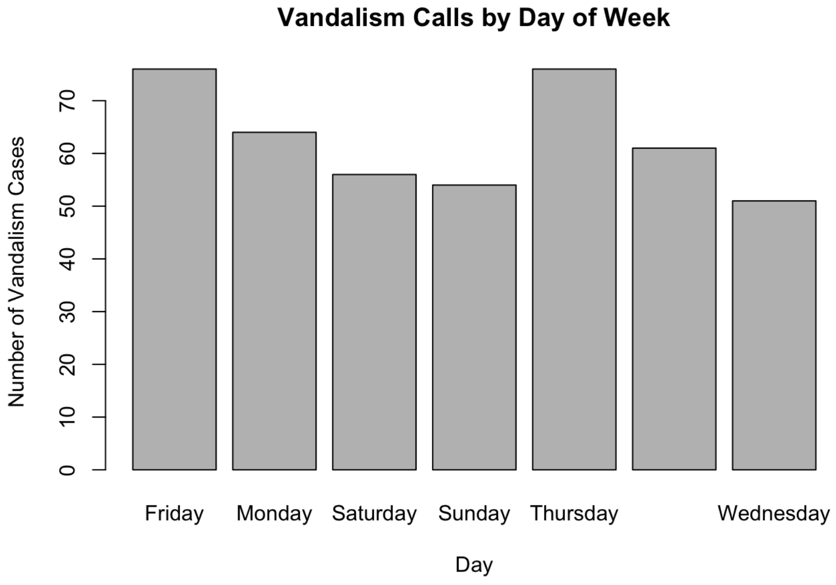
Analysis

Once your data is prepared for use, patterns within the data can be discovered through transformations, model fitting and visualization.

The benefits of data visualization are many fold. Not only are they useful for comprehending information quickly and identifying relationships; they are also easy ways to communicate your thought process to your audience—something we will discuss in the next section. Packages like ggplot2 or R base can be used to draw frequency bar plots, histograms, line plots, and density plots that help us detect patterns that we could pay closer to attention to and investigate.

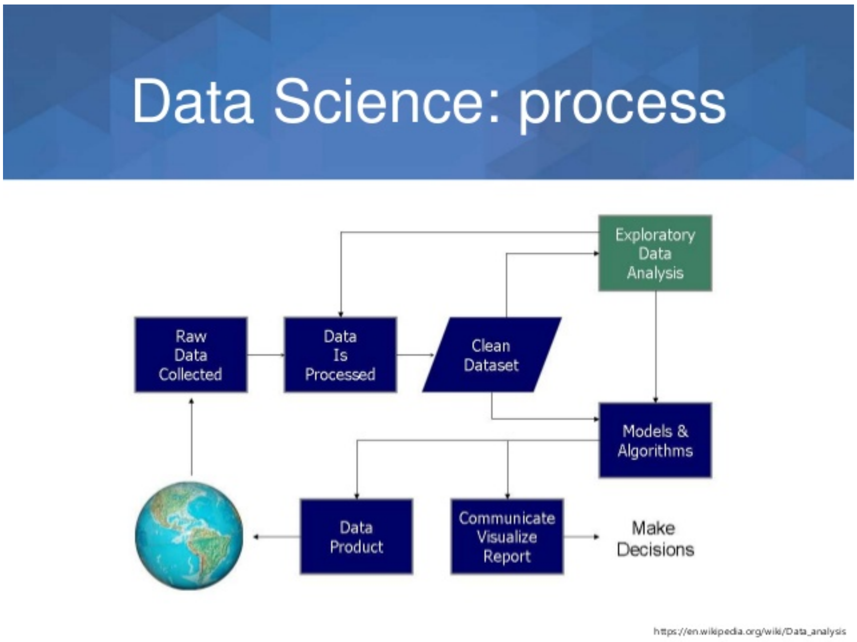
We can use graphics like hist and barplot (see below) to visualize patterns across different times and days of the week. We find that vandalism cases peak toward the end of the week, on Thursdays and Fridays. Once we have some intuition about patterns like this, we can investigate further into a pattern using other data sets, conduct hypothesis tests (a complex topic which I will not delve into for this blog), or form new questions.

```
crime = filter(calls, calls["OFFENSE"] == "VANDALISM") # Filter rows to only Vandalism cases.
counts = count(group_by(crime, CVDOW)) # Group by days of week, aggregated by counts
barplot(counts[["n"]], names.arg = counts[["CVDOW"]], main = "Vandalism Calls by Day of Week", xlab = "Day", ylab
= "Number of Vandalism Cases")
```



Reporting

In every data analysis project, condensing all your efforts into a coherent report is the final yet often most time consuming stepping stone. In writing one of these reports, you should be able to cite the sources of your raw data, as well as describe what transformations and adjustments made to your data. Include images of the charts and plots that aided you along the way in your analysis. It is crucial that your results and code leading up to your results are reproducible (able to be repeated by others); therefore, you should preserve your data in all of its past and present forms and make sure your code is written in a way that is clear and able to be run even with minor modifications. Remember that your overall message will not be about how much trouble you had with cleaning the data, or what specific techniques you used. Instead, the audience will want to hear your main conclusions—this is the most important thing to include in your write up.



Data Science: process

https://en.wikipedia.org/wiki/Data_analysis

Data Analysis Process

The above image is a general summary of the process described in this post. As one can see, the data science "cycle" is not always a unidirectional circle. There can be many mini-cycles that occur between steps (e.g. exploratory data analysis and cleaning) before moving on. The take-home message: the data analysis cycle is a complex one - one that involves creativity, close attention to detail, and critical thinking.

References

- "Responsible Conduct in Data Management." Data Analysis, ori.hhs.gov/education/products/n_illinois_u/datamanagement/datopic.html.
- Hover, Jason. "THE DATA DOWNLOAD." Data Profiling: What, Why and How?, ds.datasourceconsulting.com/blog/data-profiling/.
- Story, Peter. Experimental Design for Behavioral and Social Sciences.
- "Data Visualization Overview." DaSy Center, SRI Education, dasycenter.org/data-visualization-overview/.
- Publishing, ReliaSoft. "Data Analysis and Reporting." Weibull.com – Free Data Analysis and Modeling Resources for Reliability Engineering.
- Spark Summit Follow. "Data Science lifecycle with Apache Zeppelin and Spark by Moonsoo Lee." LinkedIn SlideShare, 3 Nov. 2015.
- Data Science 100 Course Material (CS100)