Universitat Autònoma de Barcelona

Crime data in the US
Creation of a unified Data Hub using Open Data

Cristian Estany   Alfredo Hernández   Alejandro Jiménez

17th November

Abstract

Crime Data Analysis is an already established yet flourishing topic in the United States. Despite that, there is no public user-friendly global data hub to visualise crime statistics across the US territory. The goal of this work is to create dynamic time series visualisations of the evolution of crime in different cities of the US. The mission of this work is to develop visualisation tools in order to further ease the access to already open data. Using the R language we gather open databases of different cities and unify the data into a single standard format, leading to straightforward comparative methods. We would like to see whether or not crime statistics change significantly throughout the different regions of the US and if there are clear trends that help us identify geographical differences in the American society. Last but not least, we would like to look for temporal patterns that could be identified with major global or local events.

Contents

1 Introduction

1.1 Summary and goals

The arrival of Data Science has caused a great impact in many different topics and professions. From every workplace we are seeing a huge effort to adapt the usual workflow into a new, more data-driven decision making environment, in which a proper data analysis can determine the fate of any kind of professional or academic endeavour.

One of the ecosystems in which Data Science is landing and leading a new path is in criminology. A whole new topic of study is emerging from the combination of criminology and applied statistics, often called *Crime Analysis*; and this topic is starting to be one of the most important tools in the decision making processes in many Public Safety Bureaus. Specifically in the United States is possible to notice an important effort from several different cities to build open data portals in which they publish police records, arrest records, distress calls, etc.

Despite this gigantic effort in organising and publishing large amounts of data, we noticed how heterogeneous these data bases were from one city to another, and how messy and unclear could be for a crime analyst to get insights from this data. For every city you can find many different data warehouses, and everyone of them stores data from various sources in some specific and usually different static structures and categories. This situation might have been sufficient in its early stages, when data were small enough, but it can get utterly messy when trying to work with data from different cities and sky-rocket the budget of a crime analysis department. This problem is not only about crime analysis departments, it is a generalised concern on *Big Data*.

The *Data Hub* is emerging as one of the major solutions for these Big Data issues. A stable Data Hub alleviates the costs and complications of exploratory analysis, empowering non-technical users to exploit the possibilities of data intelligence.

Our proposal in order to make this situation simpler and clearer in *Crime Data Analysis* is to build a national *Crime Data Hub*, in which any Police Department or any agency concerned about public safety could enter and explore uncomplicated data.

For this *proof of concept* we aim to present an initial stage of what an stable Crime Data Hub could be. A Data Hub that gathers, processes, normalises, and standardises crime data from several different cities around the United States for any analysis purpose. And more importantly, this has to be done working around the computational difficulties and limitations of dealing with huge data sets.

Our implementation is hosted on `https://aldomann.shinyapps.io/crimes-hub/`, and the code and scripts developed for this project can be found on `https://github.com/aldomann/open-data-project`, all released under the GNU General Public License v3.0.

2 Tools and data

2.1 Data description

Before starting mining the data and programming the scripts needed for this project, we wanted to familiarise ourselves with Crime Analysis both from an academic and a real-life point of view.

From the academic perspective, we found [1] and [2] incredibly insightful to see why data sets are structured in the way they are: Police Departments across the United States use the Uniform Crime Reporting (UCR) Program, a national standard of classifying crime uniformly that disregards state laws, providing a way to consistently count crime across the US.

On the real-life side of things, we really liked how the Chicago Tribune has a dedicated data hub on crime statistics [3] open for citizens to explore. The main feature of this data hub is the time series visualisation of crimes in the city by month (depicted in figure 1).

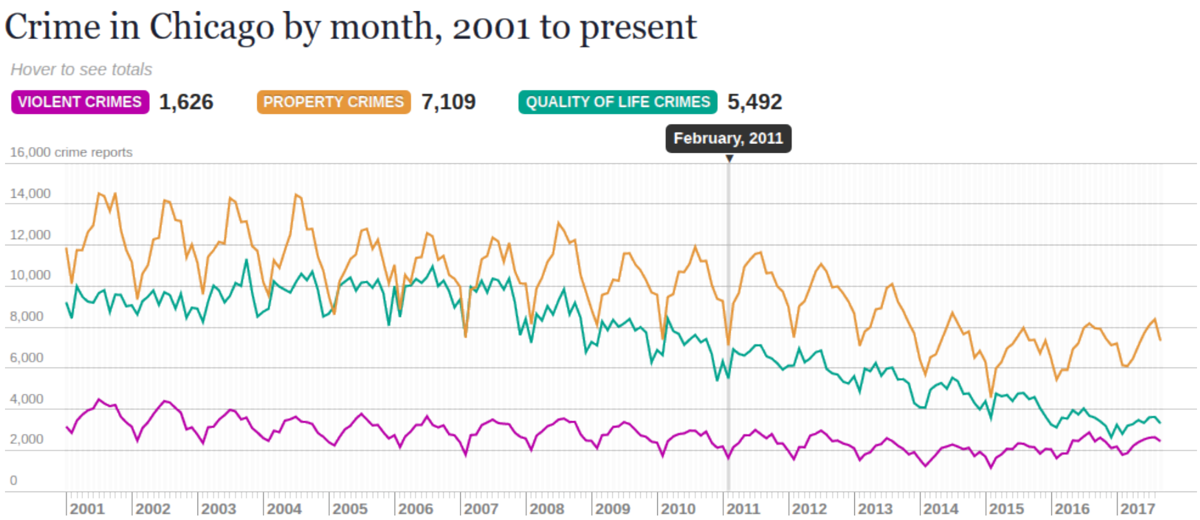

Figure 1: *Crime in Chicago by month*, from [3]

A really interesting aspect about this visualisation is that they divide crimes into three categories: (i) violent crimes, (ii) property crimes, and (iii) quality of life crimes.

We took this time series as a template of how we wanted to visualise data in our Data Hub. The main difference will be how we categorise crimes; there's some literature on how property and violent crimes are defined [2], but quality of life crimes are very difficult to define. For this reason we decided to use *other crimes* as a third category (we'll expand on this categorisation process later). Apart from all this, we think it is important to include the amount of total crimes as well, as it could be really useful for an exploratory analysis.

Having said this, let's discuss on the data sets themselves. As we have mentioned before, our main goal is to analyse crime statistics from different cities across the US. To ensure some kind of evenly distributed sample, we chose to use data sets from 3 to 4 cities for each of the main Cultural Regions of the US (depicted in figure 2).

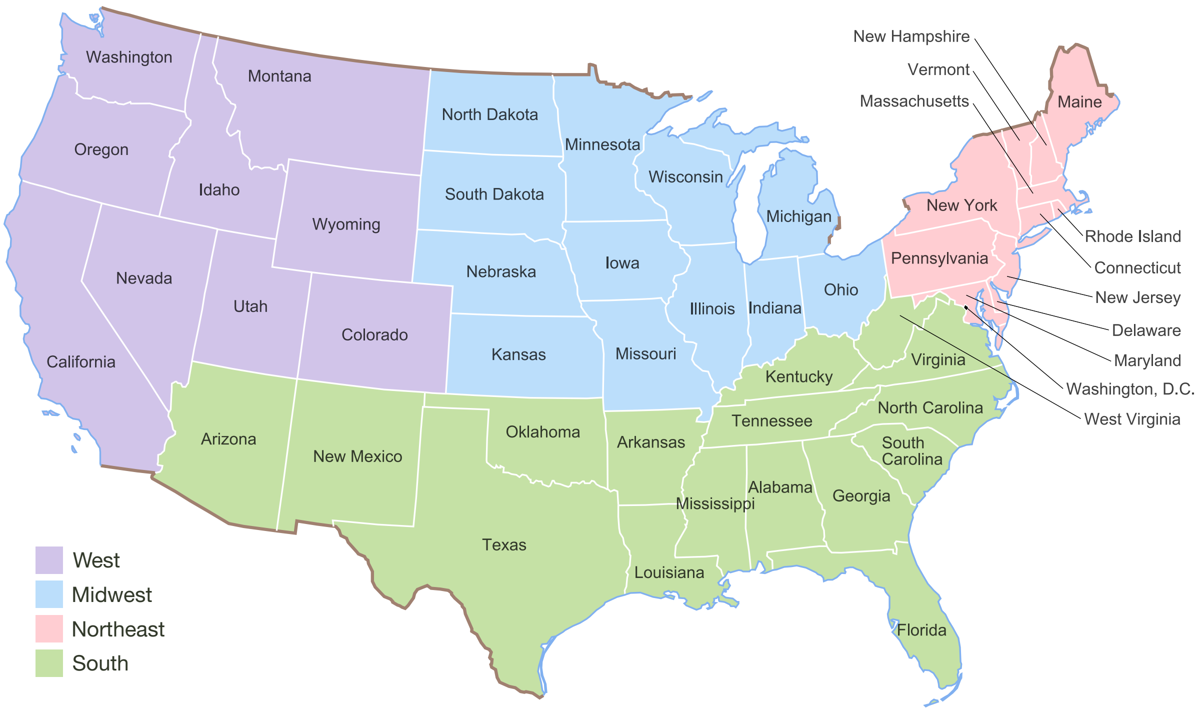

Figure 2: Map of the United States divided by the four main Cultural Regions

In table 2.1 we can see a full description of the data sets, including geographical location, life cycle, size of the data set, and the license under which the data is released.

At a very first glance we can see how some of the data sets do not specify a license of any kind. The thing is that these databases are initially thought as a tool for Police Departments as a way to ease the report of the statistics to the FBI; the involvement of specialised professionals (that is, data scientists or crime analysts) is relatively new and they may not care about specifying a license at all[1]. We can be thankful to the Police Departments, though, for making available the data in their respective city's Open Data portal. In these cases we assume they don't impose any restriction on the data and that they are Public Domain.

[1]From my personal experience in the Free Software and Science communities (specially the latter), it's usual to see people that don't realise the importance of licensing and compliance.

City	State	Region	Life Cycle	Data Set Size	License
Atlanta	Georgia	South	2009–2016	267 K	Not Specified
Austin	Texas	South	2014–2015	159 K	CC0 1.0
Baltimore	Maryland	Northeast	2012–2016	277 K	CC0 1.0
Chicago	Illinois	Midwest	2001–2016	6.4 M	Not Specified
Dallas	Texas	South	2014–2016	330 K	ODC-By 1.0
Detroit	Michigan	Midwest	2009–2016	1.2 M	Public Domain
Los Angeles	California	West	2010–2016	1.6 M	CC0 1.0
Metro Area	Washington, DC	Northeast	2008–2016	343 K	Not Specified
Minneapolis	Minnesota	Midwest	2010–2016	136 K	CC BY-SA 3.0
New York City	New York	Northeast	2014–2016	5.6 M	CC0 1.0
Philadelphia	Pennsylvania	Northeast	2006–2016	343 K	Not Specified
Portland	Oregon	West	2015–2016	140 K	Not Specified
San Francisco	California	West	2003–2016	2.1 M	PDDL 1.0
Seattle	Washington	West	2009–2016	1.5 M	CC0 1.0

Table 1: Summary of the data we gathered during the process of building the Data Hub

As a side note, we would like to clarify a bit why we focused on data sets only from the United States. The original scope of our project was to create a truly global Data Hub, gathering data from the Americas, Europe, Asia, etc. We found, alas, that although a powerful Open Data infrastructure does exist for some of the countries we checked (Spain, the Netherlands, Germany, France, the UK, Japan, Hong Kong, South Korea, etc.) the data shines for its absence. Hopefully this will change in the coming years.

The difference between the adoption ratio of Open Data in the United States vs *the rest of the World* is absolutely abysmal. We believe the main reason behind this is the federal organisation of the US, which facilitates the conception of controlled and efficient policies. Another point to keep in mind is that the social and administrative concerns across the US are much more uniform than across Europe, for instance.

2.2 Data mining process

One of the key points of a functional and stable Data Hub is the standardisation of its data. From one data warehouse to another we can find several different structures, categories and types of variables. So this section is about automatising the *data mining* process, in which we shall download and clean all the data, select the more relevant[2] variables and process them in a way so that the final output would be the same for *every* data set.

[2]Relevant for the purpose of this project, not in an statistical sense

Downloading the data

As obvious as it may seem, the first thing to do is to download the raw data. Every database has its own URL and its own origin; so this has to be treated individually for each city.

Here we show, as an example, a snippet of the code used for that purpose for San Francisco:

```
file = "data/sanfrancisco_raw.csv"
url = "https://data.sfgov.org/api/views/tmnf-yvry/rows.csv"
download.file(url), destfile = file)
```

One particularly difficult example of this process was the city of Atlanta, for which we had the files separated by years (one file per year) and every file was formatted slightly differently. The challenge here was to merge the files together into a unified data set. For this city we developed an extra specific function for this purpose: `merging_atlanta.R`

Reading & cleaning the data

Once we have obtained the raw data, the next step is to filter and clean it. This is a subtle point, for the process may be different from one data set to another, and the code has to be specifically tuned for every one of them.

It is usual that most of the variables (columns) in a data set are redundant or simply not necessary for our purpose, so a simple step is to select just those who are needed. Once we have just the columns we want, the challenge is to format them in a standard way:

```
sanfrancisco.df <- fread(file="data/sanfrancisco_raw.csv", sep = ",", header = TRUE, select = c(2,5,10,11))

sanfrancisco.df <- sanfrancisco.df %>%
  dplyr::rename(Primary.Type = `Category`) %>%
  dplyr::rename(Longitude = `X`) %>%
  dplyr::rename(Latitude= `Y`) %>%
  mutate(Date = format(as.Date(Date, format="%m/%d/%Y"), "%Y-%m-%d")) %>%
  filter(is.na(Longitude) != TRUE)
```

One particularly difficult situation in this process was found when one column was containing `latitude` and `longitude` in a complicated way, containing names of streets and such stuff. We used regular expressions to process the data and clean the messy structure of the variables:

```
# Functions gathering geolocalization data
longitude <- function(x){
  a <- strsplit(x, "\\(", fixed = FALSE, perl = FALSE, useBytes = FALSE)[[1]][2]
  a <- gsub(")","", a)
  longitude <- as.numeric(strsplit(a, ",")[[1]][1])
```

```
}

latitude <- function(x){
  a <- strsplit(x, "\\(", fixed = FALSE, perl = FALSE, useBytes = FALSE)[[1]][2]
  a <- gsub(")","", a)
  latitude <- as.numeric(strsplit(a, ",")[[1]][2])
}

# Extracting longitude and latitude
detroit.df$longitude <- sapply(detroit.df$LOCATION, function(x) longitude(x) )
detroit.df$latitude <- sapply(detroit.df$LOCATION, function(x) latitude(x) )
```

Categorising Primary Types

And then we arrive to the *secret sauce* of this project. One crucial step for the purpose of this project was the classification of crimes. Doing this is quite complicated since there are a huge amount of different names used to classify the typology of crimes, often called *Primary Type*. To fulfil this necessity we have created a function that contains an ever-growing dictionary of classification of crime names. This function classified all kind of crimes into three self-explanatory categories: *Violent Crimes*, *Property Crimes*, and *Other Crimes*.

Considering the immense size of the function, the snippet below shows a small sample of the functions we use for the categorisation process:

Snippet 2.1: Simplified structure of `prim_types_functions.R`

```
property <- c( "ARSON", "AUTO THEFT", "BIKE THEFT", "BURGLARY", ... )

violent <- c( "AGG ASSAULT", "BATTERY", "HOMICIDE", "MURDER", ... )

other <- c( "DRUNKENNESS", "FORTUNE TELLING", "TRAFFIC", ... )

# Classify primary type into Categories
find_prim_type <- function(a){
  return (ifelse(toupper(a) %in% toupper(property), "PROPERTY",
                 ifelse(toupper(a) %in% toupper(violent), "VIOLENT",
                        "OTHER")))
}

# List non-classified primary types
find_new_prim_type <- function(df){
  list <- c(unique(toupper(df$Primary.Type)))
  new <- c()
  for (a in list) {
    if ( a %in% toupper(property) | a %in% toupper(violent) | a %in% toupper(other) ) {}
    else { new <- c(new, a) }
```

```
  }
  return (new)
}
```

The maintenance of this function is quite complicated, for new crime categories appear with every new data set, and then our *dictionary* has to be expanded.

Summarising the data

Now we have almost all needed variables, but the data set is still too large for being uploaded into a Data Hub to be processed in real time by the server. Now what we do is to summarise the data grouped into months. That is, we put together all crimes occurred at the same month.

```
sanfrancisco.df <- sanfrancisco.df %>%
  group_by(Category, year = year(Date), month = month(Date)) %>%
  summarise(N=n())
```

With this, we have obtained a time series of crimes classified by categories.

A usual variable considered in typical crime analysis, is the number of crimes compared to the population of the city; obtaining the crimes per inhabitant. For this, we merged these data sets with the public access population data of every city [4].

```
# San Francisco Population -----------------------------------------------------

pop.file = "pop-data/population_sanfrancisco.csv"
sanfrancisco.pop <- fread(pop.file, sep = ";", header= TRUE, select = c(1,2))
sanfrancisco.pop$population <- as.numeric(gsub(",", "", sanfrancisco.pop$population))

sanfrancisco.df <- merge(sanfrancisco.df, sanfrancisco.pop)
sanfrancisco.df$population <- sanfrancisco.df$population
```

Deleting invalid data

At this point we should have the data ready to be easily visualised and analysed. Before uploading the data, a careful exploration is needed to avoid unexpected surprises. By doing this, we spotted some errors or inconsistencies in the data. Usually we found a lot of gaps in the first years of the data sets, comprehensively still getting started; so we just deleted those years in order to maintain the published data consistent and reliable.

Unexpectedly, we have found a strange behaviour in the Seattle data set. There were massive holes within two years of data records. This data set is shown in our public Data Hub for the *proof of concept* version, as an example of badly formatted data (this is expanded in § 3).

Saving the processed data set into a file

Simple as it sounds, we just create a `.csv` file containing the data set.

The success of the whole process is reflected by the fact that the initial raw data was of $\sim 1\,\text{GB}$, whilst the final output is of $\sim 10\,\text{KB}$

2.3 Programming the application

Since all the data mining process was done using R, we thought that programming the web app using the `shiny` package [5] and `shinyapps.io` [6] as the platform for hosting our web app would be the most straightforward process to follow.

In short, the app consists on two parts: a `ui.R` file consisting on the frontend (the visual and interactive structure of the web app), and a `server.R` file consisting on the backend (how the data is computed according to the user's interaction). These two files are sourced in the `app.R` base file, allowing us to have a more structured and maintainable code.

In snippet 2.2 we can see the whole `app.R` script. As we can see, the structure is really basic and self-explanatory. The `shiny.sanitize.errors = FALSE` option is used to be able to acess the log files from the server, which is essential when debugging the app once it's deployed.

Snippet 2.2: Structure of `app.R`

```
## app.R ##
library(shiny)
library(markdown)

# Source UI and Server -------------------------------------

options(shiny.sanitize.errors = FALSE)

source("ui.R", local=F)
source("ui.R", local=F)

shinyApp(ui, server)
```

The UI part, snippet 2.3, is very reminiscent of what a standard program may look like. We have objects and functions that are called or used in a *main function* (in this case, the `ui` object):

Snippet 2.3: Simplified structure of `ui.R`

```
# Libraries ------------------------------------------------

# UI Libs
library(shinydashboard)
library(dygraphs)

```

```
# UI ---------------------------------------------------

header <- dashboardHeader(
  title = "Crimes␣Data␣Hub"
)

cities.list <- c( "Atlanta", "Austin", ... )

sidebar <- dashboardSidebar( ... )
body <- dashboardBody( ... )

ui <- dashboardPage(
  skin = "purple",
  header,
  sidebar,
  body
)
```

The server part, snippet 2.4, is probably the most difficult to work with, as you have to be really careful about how the input conditions are used to dynamically modify the data frames R is internally working with.

Snippet 2.4: Simplified structure of `server.R`

```
# Libraries --------------------------------------------

# Server Libs
library(dplyr)
library(xts)

# Server -----------------------------------------------

server <- function(input, output) {
  output$single.plot <- renderDygraph( { ... } )

  output$compare.plot <- renderDygraph( { ... } )
}
```

The star of the show in the app is the `dygraphs` library, an R interface to the dygraphs JavaScript charting library. This allows us to:

- Automatically plot `xts` time series objects.
- Have rich interactive features including zoom/pan and series/point highlighting.
- Have various graph overlays in a same plot.

After finishing the app, we have to deal with the deployment process into the server. Using `http://www.shinyapps.io/` instead of a server of our own, the process is really simple. The steps to follow are, in essence:

```
# Authenticating (only once)
rsconnect::setAccountInfo(name, token, secret)
```

```

# Deploying the app
rsconnect::deployApp("crimes-hub")
```

More details about the deployment process can be found on the documentation [7].

3 Results

Some of the motivations behind developing a Data Hub were:

- Allowing users to compare crime statistics from different cities.
- Providing an easy method to do exploratory analysis.

In figure 3 we can see that the total number of crimes in Chicago seems to be decreasing at a fast rate; this could be interpreted in lots of ways. First of all, we could simply think the Chicago Police Department is doing their job, and crime is simply decreasing. But there's something important we're forgetting: property and violent crimes numbers seem to be quite stable; could this be an example of the CPD *juking the stats* by registering less non-property and non-violent crimes? Could this be instead a result of the CPD having a lower records management system (RMS) budget? This is something really difficult to state for sure without having detailed budgetary and sociological information about the city; but this is definitely a starting point for a more comprehensive crime analysis.

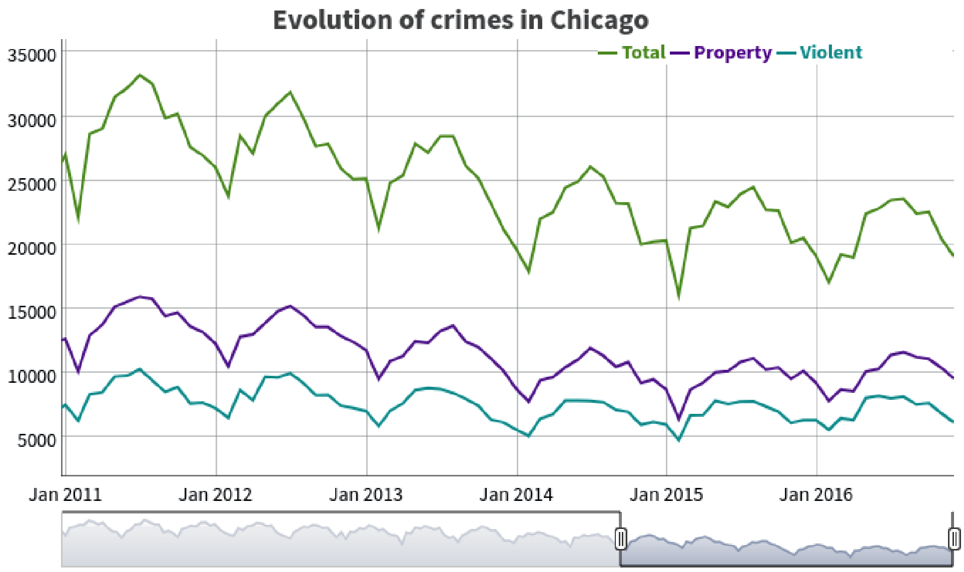

Figure 3: Crime statistics in Chicago between January 2011 and December 2016

But this is just the analysis of one very specific city. Are more cities experiencing the same decrease in registered crimes? In figure 4 we can see a comparison of four cities: Atlanta, Chicago, Philadelphia, and San Francisco (from the South, Midwest, Northeast, and West regions, respectively).

This time series opens a new picture on our understanding of crime statistics. It's well known that Chicago is particularly exemplary [2] on the early adoption of RMS systems, leading to an statistically robust data set. But how can we guess if this is the case with other cities? It is quite glaring that total crimes number by themselves are probably not the best number to analyse more than one city at the same time. For this reason we decided to give the user the ability to normalise data using two criteria: (a) normalise by population[3], depicted on figure 5 ; and (b) normalise by crime mean[4].

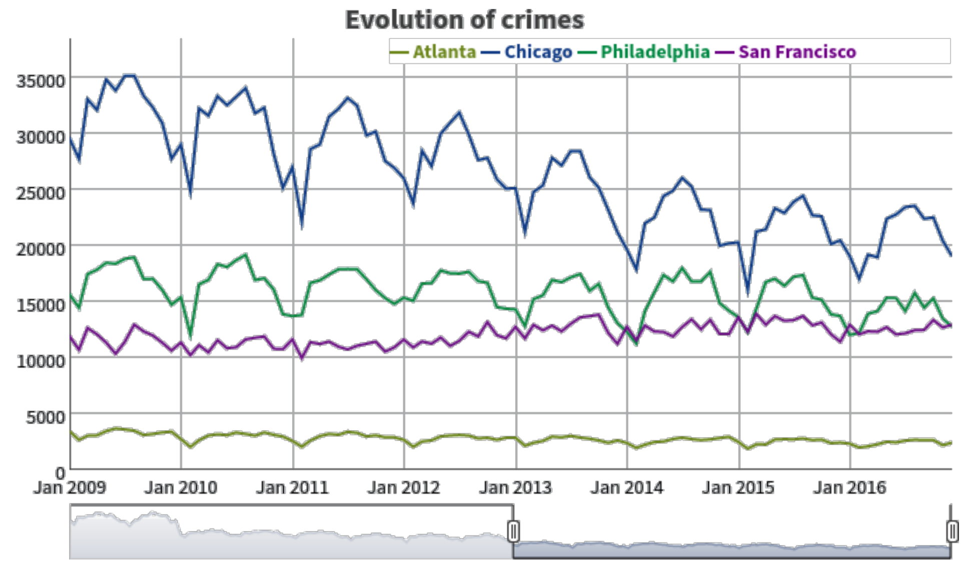

Figure 4: Comparison of crime statistics between Atlanta, Chicago, Philadelphia, and San Francisco

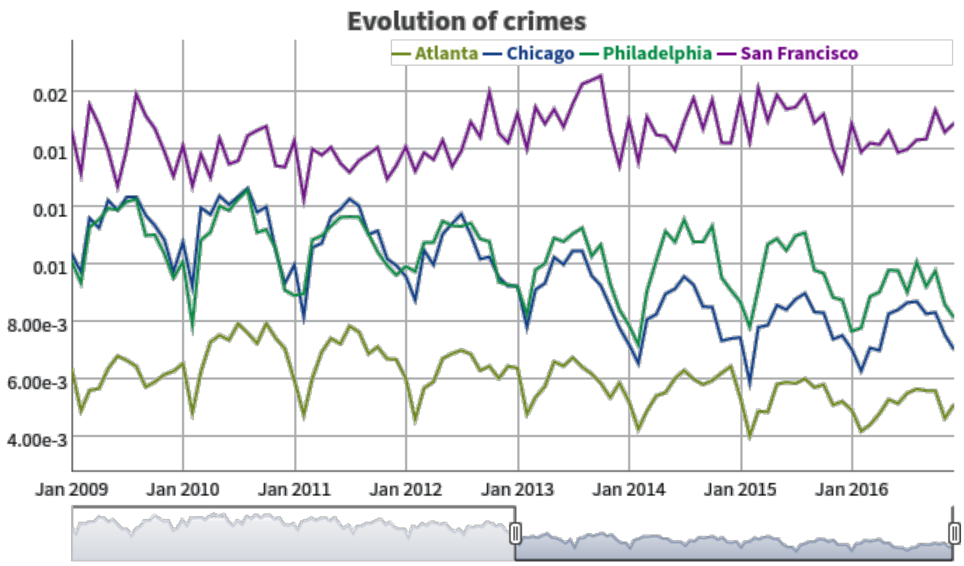

Figure 5: Comparison of crime statistics between Atlanta, Chicago, Philadelphia, and San Francisco; normalised by population

[3]This ratio shows the crimes per inhabitant. Widely used while comparing crime reports of different cities. [1]

[4]Crime mean is the all-time average of the city crimes. It is usually used to study the relative time-evolution of crimes. [1]

The differences in the three time series are remarkable. Using normalisation methods give us a much richer approach to do a qualitative comparative analysis. It's remarkable how different the West (San Francisco, Los Angeles, Portland, ...) is from the rest of the US: they are all seem to follow completely different cyclic patterns; crimes are on the rise, whilst the rest of the country seems to be improving their cities in regards to crime statistics.

Again, we could interpret these statistics in many ways. The obvious one seems to be that young and upper-class people, specially because of Hollywood and the Silicon Valley, are moving to the West coast, leading to more opportunities to commit crimes (usually property crimes). But without any proper analysis on the social predictors, this is just a wild conjecture. Quoting Rachel Boba [1]

> *All of these types of data serve a purpose in providing a picture of crime. However, depending on which data are used the picture may be very different.*

Last but not least, we would like to talk about Seattle (figure 6). We included this city as an example of why data scientists and data analysts are required in today's world. When we first saw the data, we were quite shocked. How come there's such a huge gap between February 2013 and August 2014? What we found is that for some reason, they don't have the *Event Clearance Date* for most of the crimes reported in this period, but they do have the *General Offense Number* (in the format `YYYYN`, where `YYYY` is the year `N` seems to be an internal tracking code); this surely could be used by a data scientist to repopulate the missing values using data governance techniques.

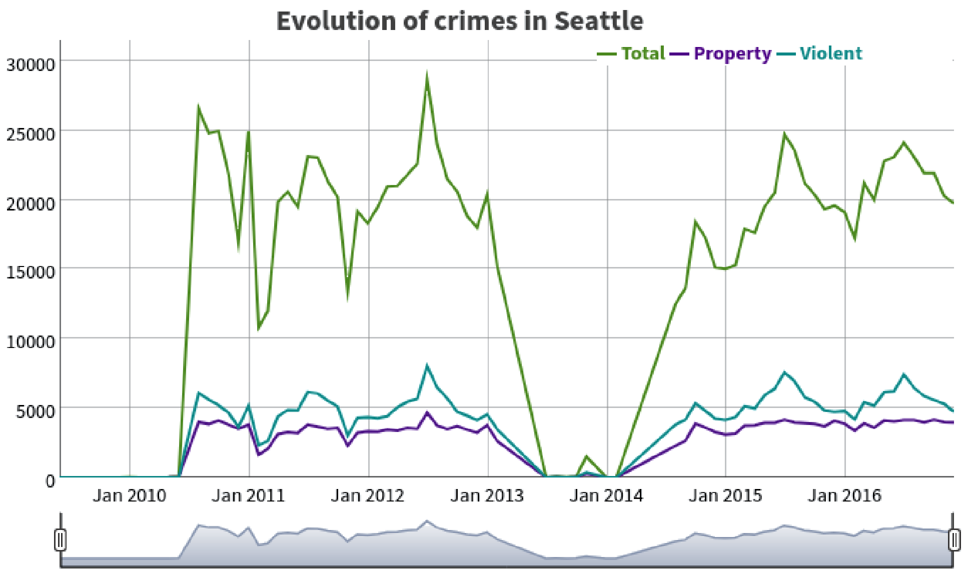

Figure 6: Crime statistics in Seattle between January 2011 and December 2016

4 Conclusions

Open Data is a quite new movement so it is usually hard to find good Open Data Sources that are not unclean, wrongly formatted or incomplete; making the cleanse of data a necessity. Cleaning data is one of the most tedious steps of data mining, and even more when you are extracting data from many sources and comparing it. For this is a process subject to a lot of unexpected problems, there is no easy nor systematic way to solve them. Specially in this field *Crime Analytics*, where almost every data set is formatted differently, despite the existence of standard guidelines.

Whilst the development of this project, we found out how straightforward and useful was to explore the data seeking for patterns or insights about the criminal situation of those cities. We can be satisfied about the result of this proof of concept, yet further development is needed to achieve a final stable product.

First of all, following the philosophy of *Open Data*, a proper *Data Hub* should facilitate a way to download the cleaned data whilst visualising it. Although this shouldn't be difficult we were more focused and concerned about the data extraction and cleaning; and we did not want to jeopardy the stability of the host loading too much traffic data. Since this is just an early stage of the whole project, we will keep this point as one of our first main objectives in the future in order to improve the *Data Hub*.

Additionally, we would like to expand the cities in the catalogue so the portal can be even more useful overall. Our main limitation here was time, we just tried to add as many cities as we could. At this point, the more time-consuming step was the classification of the crime's *Primary Type*, since it is done manually; so we are considering different machine learning techniques to automatise this process and save time and resources. This process could be a project by itself so it lands out of the scope of this one. However, we will keep it as an exciting possibility to enrich and improve the *Data Hub*

Another useful improvement could be the section of single city. We have in mind the addition of geographical data (in the data mining process, we cleaned and prepared `latitude` and `longitude` variables ready to be loaded in the Data Hub, probably in the form of *heatmaps* of crime reports in the city map. The visualisation of this kind of data is much difficult and requires greater resources from the server if we aim to keep the site interactive; so we discarded it for this first stage; but it would be a valuable addition and a really useful tool for crime analysis and we are currently working on this.

As a final desire, the ultimate goal of future improvements would be to automatise all the data mining process so that the site could grow faster and easier.

References

Basic

[1] R. Boba. Interpreting Crime Data and Statistics. *Exploring Crime*:158–162, 2004.

[2] B. Beardsley and S. A. Matthews. A Spatial Analysis of Predictors of Different Types of Crime in Chicago Community Areas Research Questions:1–14, 2014.

[3] Crime in Chicagoland - Chicago Tribune. URL: `http://crime.chicagotribune.com/`.

[4] Biggest US Cities. URL: `https://www.biggestuscities.com/`.

[5] Shiny. URL: `https://shiny.rstudio.com/`.

[6] shinyapps.io by RStudio. URL: `https://www.shinyapps.io/`.

[7] shinyapps.io user guide. URL: `http://docs.rstudio.com/shinyapps.io/`.

Data Sets

[8] Minneapolis RAW Data Set. URL: `http://opendata.minneapolismn.gov/datasets?t=public%20safety`.

[9] Baltimore RAW Data Set. URL: `https://data.baltimorecity.gov/browse?category=Public%20Safety`.

[10] New York City RAW Data Set. URL: `https://data.cityofnewyork.us/Public-Safety/NYPD-Complaint-Data-Historic/qgea-i56i`.

[11] Austin RAW Data Set. URL: `https://data.austintexas.gov/Public-Safety/APD-Incident-Extract-YTD/b4y9-5x39`.

[12] Los Angeles RAW Data Set. URL: `https://data.lacity.org/A-Safe-City/Crime-Data-from-2010-to-Present/y8tr-7khq`.

[13] Seattle RAW Data Set. URL: `https://data.seattle.gov/Public-Safety/Seattle-Police-Department-911-Incident-Response/3k2p-39jp`.

[14] Dallas RAW Data Set. URL: `https://www.dallasopendata.com/Public-Safety/All-Crime/p9zb-d4n6`.

[15] San Francisco RAW Data Set. URL: `https://data.sfgov.org/Public-Safety/Police-Department-Incidents/tmnf-yvry`.

[16] Chicago RAW Data Set. URL: `https://data.cityofchicago.org/Public-Safety/Crimes-2001-to-present/ijzp-q8t2`.

[17] Detroit RAW Data Set. URL: `https://data.detroitmi.gov/Public-Safety/DPD-All-Crime-Incidents-January-1-2009-December-6-/invm-th67`.

[18] Metro Area RAW Data Set. URL: `http://crimemap.dc.gov/CrimeMapSearch.aspx#tabs-GeoOther`.

[19] Philadelphia RAW Data Set. URL: https://www.opendataphilly.org/dataset/crime-incidents.

[20] Atlanta RAW Data Set. URL: http://www.atlantapd.org/i-want-to/crime-data-downloads.

[21] Portland RAW Data Set. URL: https://www.portlandoregon.gov/police/71978.

Licenses

[22] Creative Commons 0 1.0 Universal (CC0 1.0). URL: https://creativecommons.org/publicdomain/zero/1.0/.

[23] Creative Commons Attribution-ShareAlike 3.0 (CC BY-SA 3.0). URL: https://creativecommons.org/licenses/by-sa/3.0/.

[24] ODC Public Domain Dedication and Licence (PDDL). URL: https://opendatacommons.org/licenses/pddl/1.0/.

[25] ODC Attribution License 1.0 (ODC-By). URL: https://opendatacommons.org/licenses/by/1.0/.