Detecting Anomalies in Censorship Circumvention Data

Jack O'Callaghan
Oxford Internet Institute

18th August 2023

Introduction and context

This project examines censorship circumvention data from Tor, which is a browser software that allows people to bypass content restriction / content filtering (relays) and prevents someone monitoring your connection from knowing which websites you visit (multiple-layered encryption / onion routing).

For example, with Tor, a person might aim to circumvent content restriction controls to access websites such as Twitter or BBC News in a country where access to these websites is normally filtered or blocked.

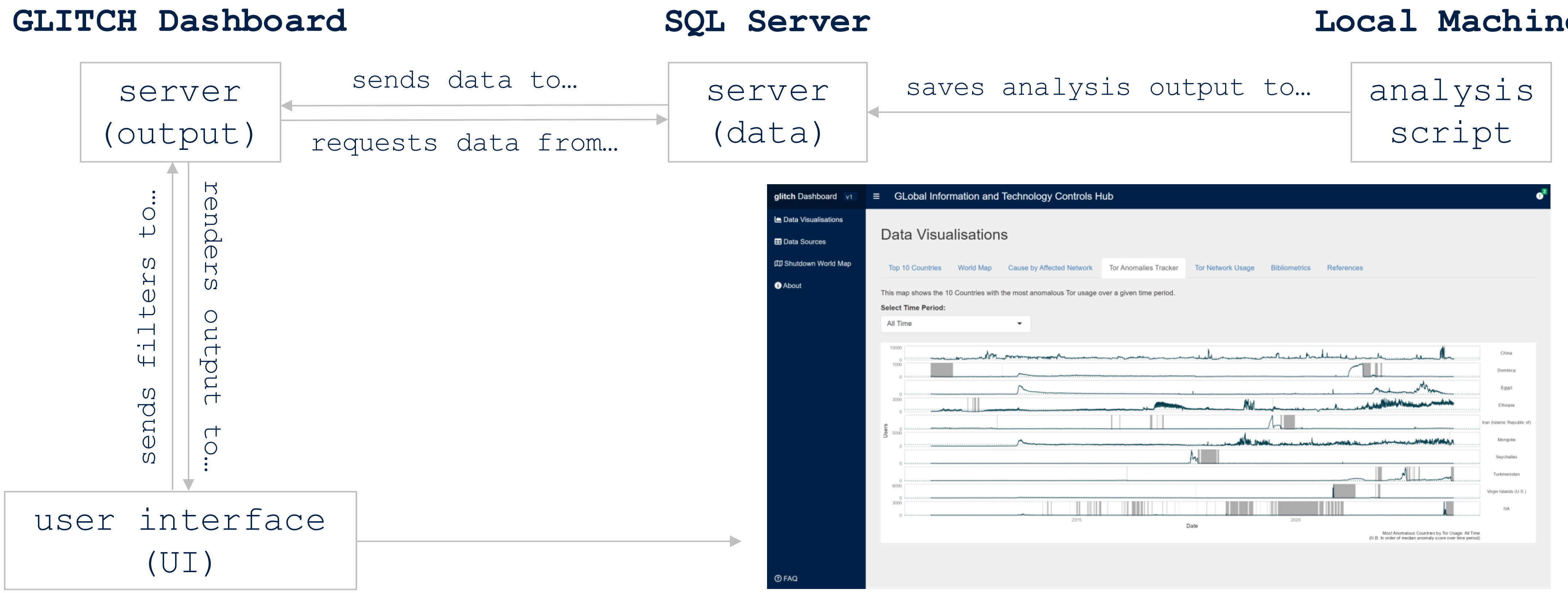
Image source: https://www.torproject.org/about/trademark

What were the initial aims of your project and how did these develop during the internship?

1. To improve the existing anomaly detection algorithm by:
    a. Improving the structure of the existing anomaly detection algorithm; changing existing base R approaches for more intuitive dplyr options where applicable.
    b. Decouple data preparation and analysis scripts for separate execution; write daily analysis results to a PostgreSQL server for external reconstruction.

2. To integrate the tool into the Oxford Internet Institute's GLITCH dashboard:
    a. Once the data processing is disaggregated, integrate the output processing script and UI output into the GLITCH dashboard.

Figure 1: Technical Structure of Analysis Workstreams



What methods, sources or approaches did you use in your project?

1. Statistical Methods used:

    a. Principal Component Analysis (PCA): Reduces the number of predictor variables in a dataset and makes it simpler to interpret.

    b. Median Absolute Deviation (MAD): Allows us to examine by how much a data point varies from the median value, implying a likelihood that an event is a statistical anomaly.

2. Data Visualisation Packages used:

    a. *shiny*: Can be used to build interactive web applications (such as dashboards), which can then be deployed online, using services such as shinyapps.io.

    b. *ggplotly*: Converts static *ggplot2* objects into *plotly.js* objects, which helps to integrate interactive and downloadable charts into the GLITCH dashboard.

1a. Improving the structure of the existing anomaly detection algorithm; changing existing base R approaches for more intuitive dplyr options where applicable.

Before:

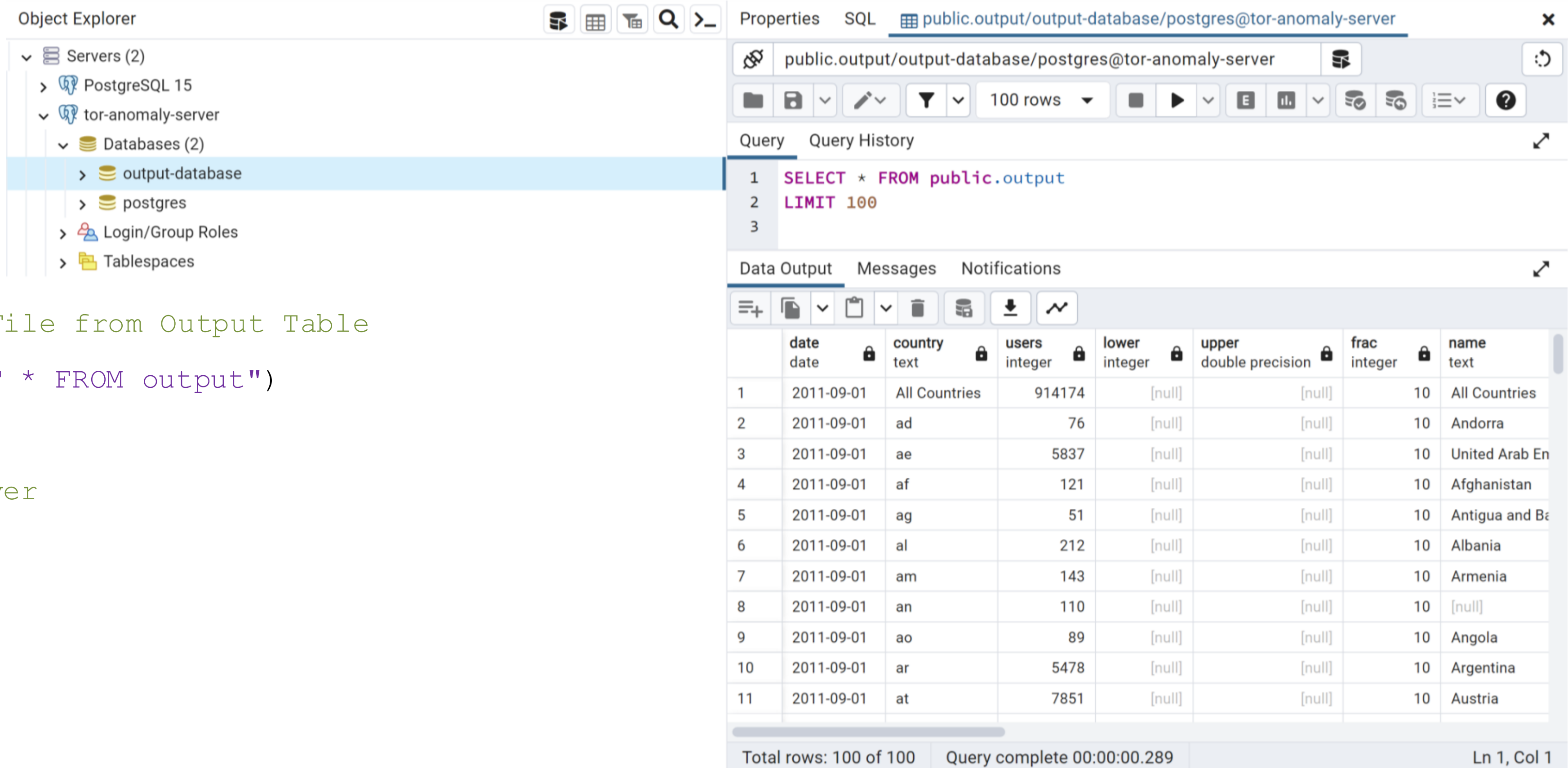
```
download.file(url="https://metrics.torproject.org/userstats-relay-country.csv", destfile="clients-new.csv", method="curl")

data <- read.csv( "clients-new.csv", comment.char="#" )

data.long <- data[,c("date", "country", "users")] # Select the three relevant variables

colnames( data.long ) <- c( "date", "country", "clients" ) # Rename "users" column

data.wide <- dcast( data.long, value.var="clients", date ~ country, sum ) # Reshape data.long to a wide format

data.wide <- data.wide[-1,] # Manually remove the outlier "2011-03-06", which is the first row

data.wide$country.name <- countrycode( toupper(fix.in(data.wide$country)), "iso2c", "country.name" )

# Add country name column

data.wide <- data.wide[,- which(names(data.wide.stripped) %in% c("ap", "eu", "a1", "a2", "o1", "??"))] # RM Non-Countries
```

After:

```
download.file(url="https://metrics.torproject.org/userstats-relay-country.csv", destfile="clients-new.csv", method="curl")

data <- fread("clients-new.csv")%>%

    filter(date != "2011-03-06") %>% # Remove Outlier Date

    left_join(x = data, y = names, by = "country") %>% # Add Country Name Column

    filter(! country %in% c("ap", "eu", "a1", "a2", "o1", "??")) # Remove Non-Countries
```

1b. Decouple data preparation and analysis scripts for separate execution; write daily analysis results to a PostgreSQL server for external reconstruction.

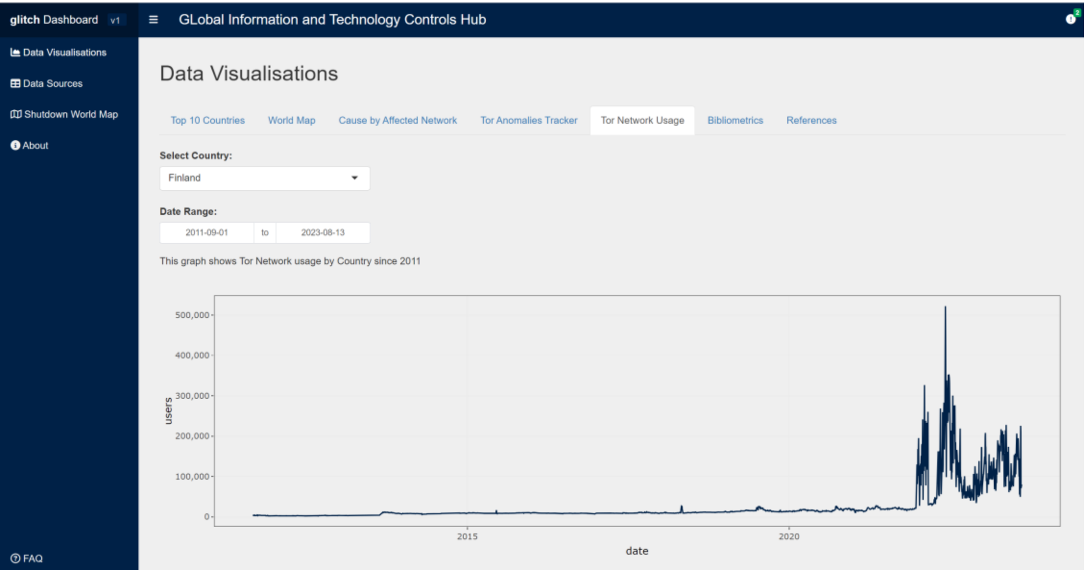
```r
## Connect to PostgreSQL Server
conn <- dbConnect(odbc::odbc(), Driver = "{PostgreSQL ODBC Driver(ANSI)}",
     Database = "output-database",
     UserName = "postgres",
     Password = pass_conn,
     Servername = "localhost",
     Port = 5432)

## Get the Latest Daily Analysis File from Output Table
output <- dbGetQuery(conn, "SELECT * FROM output")

## Disconnect from PostgreSQL Server
dbDisconnect(conn)
```

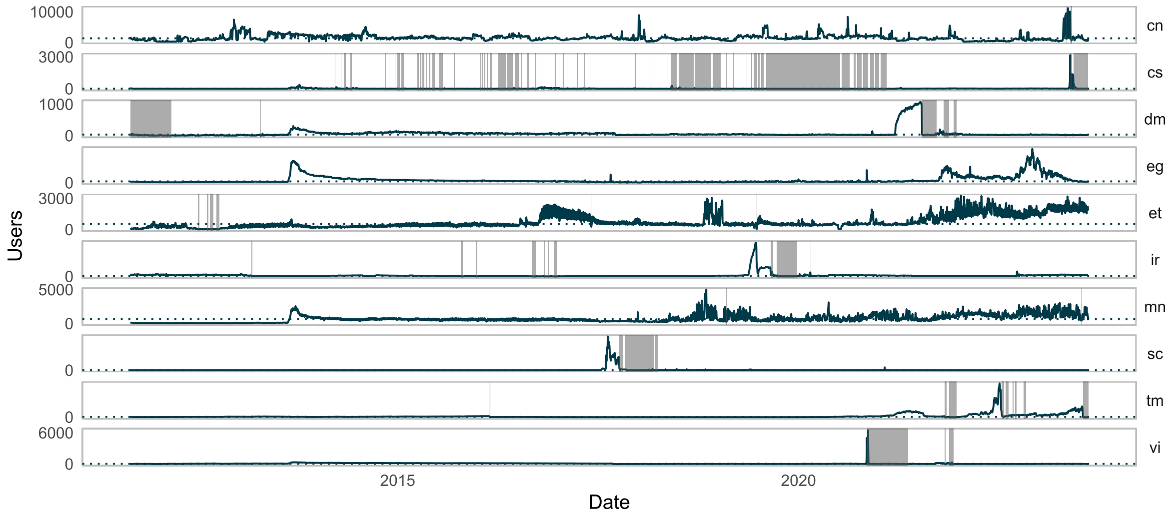


	date date	country text	users integer	lower integer	upper double precision	frac integer	name text
1	2011-09-01	All Countries	914174	[null]	[null]	10	All Countries
2	2011-09-01	ad	76	[null]	[null]	10	Andorra
3	2011-09-01	ae	5837	[null]	[null]	10	United Arab En
4	2011-09-01	af	121	[null]	[null]	10	Afghanistan
5	2011-09-01	ag	51	[null]	[null]	10	Antigua and Ba
6	2011-09-01	al	212	[null]	[null]	10	Albania
7	2011-09-01	am	143	[null]	[null]	10	Armenia
8	2011-09-01	an	110	[null]	[null]	10	[null]
9	2011-09-01	ao	89	[null]	[null]	10	Angola
10	2011-09-01	ar	5478	[null]	[null]	10	Argentina
11	2011-09-01	at	7851	[null]	[null]	10	Austria

2a. Once the data processing is disaggregated, integrate the output processing script and UI output into the GLITCH dashboard.

```
## Plot Top 10 Countries With Highest MAD - All Time Data Sample: 1st September 2011 Onwards
ggplot(data.all.time.plot) +
    geom_line(aes(x = date, y = users, group = 1)) + # Users by Date Line
    geom_hline(aes(yintercept = median)) + # Median Users Line for Comparison
    geom_rect(data=anom.rect.df.all.time, aes(xmin=xmin,xmax=xmax,ymin=-Inf,ymax=Inf)) +
    # Add Shaded Rectangles for Periods Identified as Anomalous
    facet_grid(name~. , scales = "free_y") + # Facet by Country Name
    labs(caption = "Most Anomalous Countries by Tor Usage: All Time”) +
    labs(x = "Date", y = "Users")
```





Most Anomalous Countries by Tor Usage: All Time
(N.B. In order of median anomaly score over time period)

Is there anything you would do differently if you started this project again?

If I were to start this project anew, I would allocate a greater portion of time at the start of the project to researching the specific statistical techniques involved, to improve how quickly I would have been able to start understanding and building on the existing software.

If I were able to be able to spend more time on this project, I would have liked to spend more time testing the alternative normalized usership approach to measuring anomalies that I constructed in the final few weeks of the project. This approach would potentially be an improvement on the existing approach, given that it is objectively simpler, does not rely on PCA, and is able to track the directional trends of anomalous usership periods.