

chronos__data__intro

Clara Suong

15 May, 2019

Contents

1	Load libraries and set the working directory	3
2	Databases and external datasets	5
2.1	MySQL databases	5
2.2	Key fields/variables in the database ‘declassification_frus’	5
2.3	Key fields/variables in the database ‘declassification_cables’	5
2.4	External dataset sources:	6
3	Data Overview	6
3.1	List the collections	6
3.2	Download the table “docs” for all databases	7
3.3	Number of documents and date ranges for each collection	7
3.4	Frequency tables for full text vs. non-full text	8
4	CFPF Collection (declassification_cables)	10
4.1	FIGURE: Bar Graph of Number of Cables by Month and Classification	10
4.2	Example Cable	11
4.3	Frequency Tables	13
4.3.1	TABLE: Number of Documents with Non-Missing Values by Variable	13
4.3.2	TABLE: Number of Cables by Year	15
4.3.3	TABLE: Number of Cables by Classification	16
5	FRUS Collection	17
5.1	FIGURE: Number of Documents by Year and Classification	17
5.2	Example Document	19
5.3	Frequency Tables	21
5.3.1	TABLE: Number of Documents with Non-Missing Values by Variable	21
5.3.2	TABLE: Number of Documents by Year	22
5.3.3	TABLE: Number of Documents by Classification	24
6	Country TAG Traffic	25
6.1	Examine the different country codes across datasets	25
6.2	Create a dataframe linking country codes and tag_ids	27
6.3	Tag traffic by country-year and by country	28
6.3.1	Download, save, or load the tables for tags and docs (doc_id and date) in the working directory and count the number of cables tagged for each country	28
6.3.2	TABLE: Summary Statistics of Country TAG Traffic by Country-Year (Only Contemporary Non-US Countries)	30
6.3.3	TABLE: Summary Statistics of Country TAG Traffic by Country (Only Contemporary Non-US Countries)	31
6.3.4	FIGURE: Country TAG Traffic at Country-Year and Country Levels	32
6.3.5	Percentile for Specific Values	33
6.3.6	TABLE: Summary Statistics of Country TAG Traffic by Country-Year (Including Former Countries and the US)	34

6.3.7	TABLE: Summary Statistics of Country TAG Traffic by Country (Incl. Former Countries and the US)	34
6.3.8	FIGURE: Country TAG Traffic at Country-Year and Country Levels (All Countries) .	35
6.3.9	TABLE: Country TAG Traffic vs. Cable Traffic	36
6.3.10	Country TAG Traffic for Certain Countries	38
6.3.11	TABLE: Non-US Country-Years with Most Cables	38
6.3.12	TABLE: Non-US Country-Years Tagged in Fewest Cables	39
6.3.13	TABLE: Countries Most Frequently Tagged in Cables	40
6.3.14	TABLE: Non-U.S. Countries Least Frequently Tagged in Cables	42
6.3.15	FIGURE: Country TAG Traffic of Key Non-US Countries by Month	43
6.3.16	TABLE: Country TAG Traffic vs. Total Population	44

1 Load libraries and set the working directory

```
rm(list = ls()) # clear objects in memory
library(plyr)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
## The following objects are masked from 'package:plyr':
##
##   arrange, count, desc, failwith, id, mutate, rename, summarise,
##   summarize
## The following objects are masked from 'package:stats':
##
##   filter, lag
## The following objects are masked from 'package:base':
##
##   intersect, setdiff, setequal, union
```

```
library(dbplyr)
```

```
##
## Attaching package: 'dbplyr'
## The following objects are masked from 'package:dplyr':
##
##   ident, sql
```

```
library(tidyverse)
```

```
## -- Attaching packages -----
## v ggplot2 3.1.0    v readr    1.3.1
## v tibble  2.0.1    v purrr   0.3.0
## v tidyr   0.8.2    v stringr 1.4.0
## v ggplot2 3.1.0    v forcats 0.4.0
```

```
## -- Conflicts ----- tidy
## x dplyr::arrange() masks plyr::arrange()
## x purrr::compact() masks plyr::compact()
## x dplyr::count()   masks plyr::count()
## x dplyr::failwith() masks plyr::failwith()
## x dplyr::filter()  masks stats::filter()
## x dplyr::id()       masks plyr::id()
## x dbplyr::ident()   masks dplyr::ident()
## x dplyr::lag()       masks stats::lag()
## x dplyr::mutate()   masks plyr::mutate()
## x dplyr::rename()   masks plyr::rename()
## x dbplyr::sql()      masks dplyr::sql()
## x dplyr::summarise() masks plyr::summarise()
## x dplyr::summarize() masks plyr::summarize()
```

```
library(RMySQL) #For connecting to the database
```

```
## Loading required package: DBI
```

```

library(htmlTable) #For creating Word-compatible tables
library(lubridate) #For temporal variables

##
## Attaching package: 'lubridate'
## The following object is masked from 'package:plyr':
##
##     here
## The following object is masked from 'package:base':
##
##     date
library(zoo) #For temporal variables

##
## Attaching package: 'zoo'
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
library(foreign)
library(ggplot2)
library(reshape2)

##
## Attaching package: 'reshape2'
## The following object is masked from 'package:tidyr':
##
##     smiths
library(countrycode) #For reconciling different country codes across dataset
library(ISOcodes) #A package for ISO country codes
library(stargazer)

##
## Please cite as:
## Hlavac, Marek (2018). stargazer: Well-Formatted Regression and Summary Statistics Tables.
## R package version 5.2.2. https://CRAN.R-project.org/package=stargazer
library(corrplot)

## corrplot 0.84 loaded
library(rowr) #For cbind with fill

##
## Attaching package: 'rowr'
## The following objects are masked from 'package:dplyr':
##
##     coalesce, count
## The following object is masked from 'package:plyr':
##
##     count

```

```
library(gridExtra)

##
## Attaching package: 'gridExtra'
## The following object is masked from 'package:dplyr':
##
##      combine

library(janitor)
```

2 Databases and external datasets

2.1 MySQL databases

- declassification_cables
- declassification_ddrs
- declassification_frus
- declassification_kissinger
- declassification_pdb
- declassification_clinton
- declassification_cabinet
- declassification_cpdoc

2.2 Key fields/variables in the database ‘declassification_frus’

- body
- subject
- date (year)
- classification
- urgency
- length
- (handling)
- (page_count)
- (line_count)
- office
- from_field
- to_field
- tag

2.3 Key fields/variables in the database ‘declassification_cables’

- body
- subject
- date (year)
- classification
- urgency
- length
- (handling)
- (page_count)
- (line_count)

- office
- from_field
- to_field
- tag

2.4 External dataset sources:

- Download the following datasets in the folder “external_data”
- COW country codes (cow): http://www.correlatesofwar.org/data-sets/cow-country-codes/cow-country-codes/at_download/file

3 Data Overview

3.1 List the collections

```
setwd("/Users/clarahsuong/chronos_data_intro")

#Re-connect to the database
driver = dbDriver("MySQL")
connection = dbConnect(driver, host='history-lab.org', password='XreadF403', user='de_reader')
dbGetQuery(connection, 'show databases;')
```

```
## Database
## 1 information_schema
## 2 authentication
## 3 bookwormDB
## 4 clinton_test
## 5 clinton_test_2
## 6 ddrs_equity
## 7 declassification
## 8 declassification_api
## 9 declassification_api_test
## 10 declassification_api_update
## 11 declassification_cabinet
## 12 declassification_cables
## 13 declassification_clinton
## 14 declassification_clinton_staging
## 15 declassification_cpdoc
## 16 declassification_ddrs
## 17 declassification_foia_dod
## 18 declassification_frus
## 19 declassification_frus_update
## 20 declassification_kissinger
## 21 declassification_pdb
## 22 declassification_pdb_test
## 23 etc
## 24 historylab_user_information
## 25 mysql
## 26 performance_schema
## 27 predict_history
## 28 predict_history_new
```

```
## 29          predictify_source
## 30          predictify_target
## 31          sys
## 32          user_information
## 33          visualizations
```

3.2 Download the table “docs” for all databases

```
db_docs <- function(mydb) {
  mydb2 = dbConnect(driver='history-lab.org', password='XreadF403', user='de_reader', dbname=mydb)
  docs<-dplyr::tbl(mydb2, 'docs') %>%
    collect(n = Inf) %>%
    distinct()
  return(docs)
}

#db_docs('declassification_cables')
load("/Users/clarahsuong/Dropbox/nyu_postdoc/chronos_data_intro/raw_data/cfpf_docs.RData")
cables_docs<-docs
load("/Users/clarahsuong/Dropbox/nyu_postdoc/chronos_data_intro/raw_data/frus_docs.RData")
#frus_docs<-db_docs('declassification_frus')
clinton_docs<-db_docs('declassification_clinton')
pdb_docs<-db_docs('declassification_pdb')
kissinger_docs<-db_docs('declassification_kissinger')
ddrs_docs<-db_docs('declassification_ddrs')
cabinet_docs<-db_docs('declassification_cabinet')
cpdoc_docs<-db_docs('declassification_cpdoc')

## Warning in .local(conn, statement, ...): Decimal MySQL column 3 imported as
## numeric

## Warning in .local(conn, statement, ...): Decimal MySQL column 3 imported as
## numeric
```

3.3 Number of documents and date ranges for each collection

```
db_doc_no_date <- function(mydb) {
  mydb2<-eval(parse(text=paste(mydb, sep = "")), env=.GlobalEnv)
  mydb2<-mydb2 %>%
    select(id, date) %>%
    collect() %>%
    distinct()

  return(c(nrow(mydb2), range(mydb2$date, na.rm = TRUE)))
}

db_doc_no_date('cables_docs')

## [1] "3214293"      "1973-01-01" "1979-12-31"

db_doc_no_date('frus_docs')

## [1] "209046"      "1861-05-02 00:00:00" "1985-04-05 19:00:00"
```

```

db_doc_no_date('pdb_docs')

## [1] "5011" "1961-06-17 00:00:00" "1977-01-20 00:00:00"
db_doc_no_date('kissinger_docs')

## [1] "4552" "1973-01-02 00:00:00" "1976-12-24 13:15:00"
db_doc_no_date('clinton_docs')

## [1] "54149" "2009-03-09 13:48:00" "2013-07-07 08:39:00"
db_doc_no_date('ddrs_docs')

## [1] "117509" "1900-06-15 00:00:00" "2008-05-12 00:00:00"
db_doc_no_date('cabinet_docs')

## [1] "42539" "1907-10-19 00:00:00" "1990-12-13 00:00:00"
db_doc_no_date('cpdoc_docs')

## [1] "10279" "1973-11-15 00:00:00" "1979-11-24 00:00:00"

```

3.4 Frequency tables for full text vs. non-full text

```

sum(!is.na(cables_docs$body))

## [1] 2654414
sum(!is.na(frus_docs$body))

## [1] 209046
sum(!is.na(pdb_docs$body))

## [1] 5011
sum(!is.na(kissinger_docs$body))

## [1] 4552
sum(!is.na(clinton_docs$body))

## [1] 54149
sum(is.na(ddrs_docs$body))

## [1] 0
sum(!is.na(cabinet_docs$body))

## [1] 42539
sum(!is.na(cpdoc_docs$body))

## [1] 10279
sum(sum(!is.na(cables_docs$body)),
sum(!is.na(frus_docs$body)),
sum(!is.na(pdb_docs$body)),
sum(!is.na(kissinger_docs$body)),

```



```
sum(!is.na(clinton_docs$body)),
sum(is.na(ddrs_docs$body))#,
#sum(!is.na(cabinet_docs$body)),
#sum(!is.na(cpdoc_docs$body))
)
```

```
## [1] 2927172
```

```
sum(sum(!is.na(cables_docs$body)),
sum(!is.na(frus_docs$body)),
sum(!is.na(pdb_docs$body)),
sum(!is.na(kissinger_docs$body)),
sum(!is.na(clinton_docs$body)),
sum(is.na(ddrs_docs$body)),
sum(!is.na(cabinet_docs$body)),
sum(!is.na(cpdoc_docs$body))
)
```

```
## [1] 2979990
```

```
sum(is.na(cables_docs$body))
```

```
## [1] 559879
```

```
sum(is.na(frus_docs$body))
```

```
## [1] 0
```

```
sum(is.na(pdb_docs$body))
```

```
## [1] 0
```

```
sum(is.na(kissinger_docs$body))
```

```
## [1] 0
```

```
sum(is.na(clinton_docs$body))
```

```
## [1] 0
```

```
sum(!is.na(ddrs_docs$body))
```

```
## [1] 117509
```

```
sum(is.na(cabinet_docs$body))
```

```
## [1] 0
```

```
sum(is.na(cpdoc_docs$body))
```

```
## [1] 0
```

```
sum(sum(is.na(cables_docs$body)),
sum(is.na(frus_docs$body)),
sum(is.na(pdb_docs$body)),
sum(is.na(kissinger_docs$body)),
sum(is.na(clinton_docs$body)),
sum(!is.na(ddrs_docs$body))#,
#sum(is.na(cabinet_docs$body)),
#sum(is.na(cpdoc_docs$body))
)
```

```
## [1] 677388
```

```
sum(sum(is.na(cables_docs$body)),
sum(is.na(frus_docs$body)),
sum(is.na(pdb_docs$body)),
sum(is.na(kissinger_docs$body)),
sum(is.na(clinton_docs$body)),
sum(!is.na(ddrs_docs$body)),
sum(is.na(cabinet_docs$body)),
sum(is.na(cpdoc_docs$body)))
```

```
## [1] 677388
```

4 CFPF Collection (declassification_cables)

4.1 FIGURE: Bar Graph of Number of Cables by Month and Classification

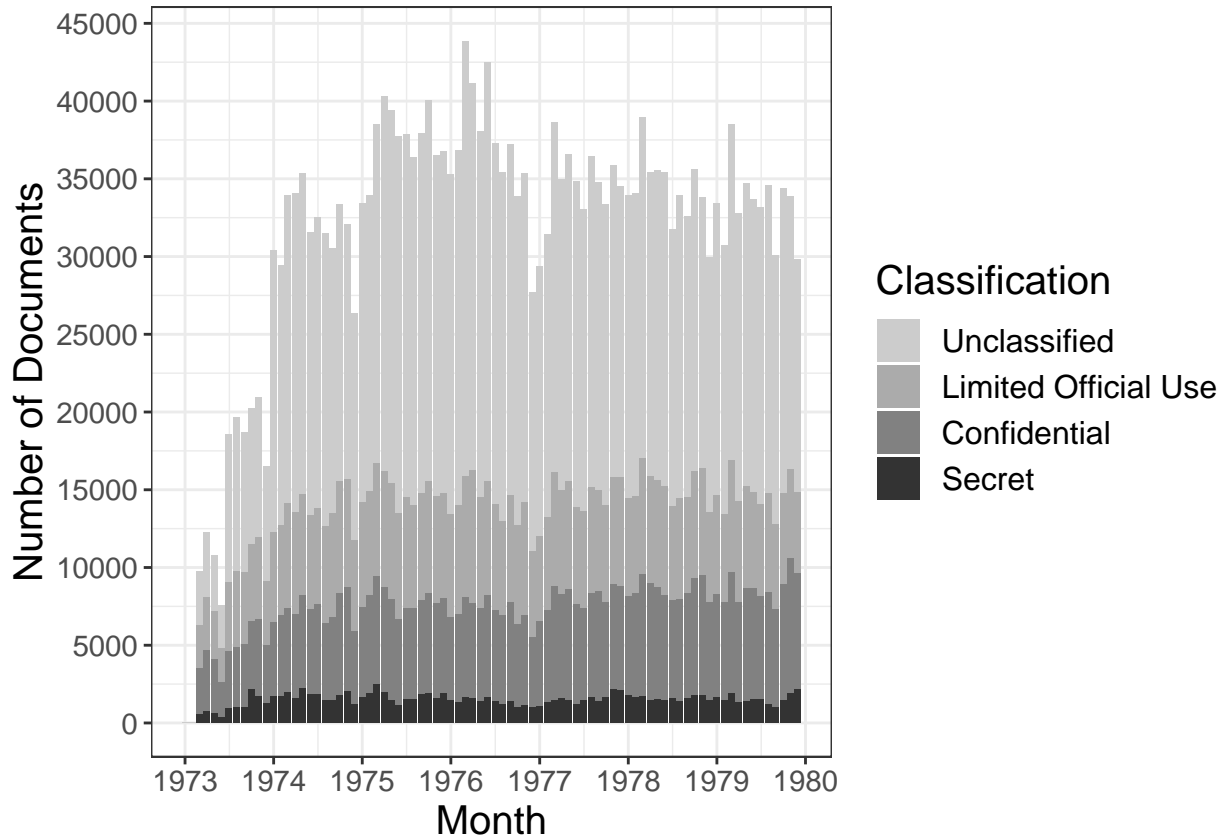
```
setwd("/Users/clarahsuong/chronos_data_intro")

cables_db = dbConnect(driver='history-lab.org', password='XreadF403', user='de_reader', dbname='de_reader')

classification_doc3 <- tbl(cables_db, 'classification_doc') %>%
  collect() %>%
  distinct() %>%
  mutate(
    date=as_date(date),
    month = as_date(cut(date, breaks = "month")),
    classification=ifelse(classification_id==1,"Secret",
                          ifelse(classification_id==2,"Confidential",
                                ifelse(classification_id==5,"Unclassified",
                                      ifelse(classification_id==7,"Limited Official Use", NA)
                                )
                          )
  ),
  classification =factor(classification, levels = c("Unclassified", "Limited Official Use", "Confidential", "Secret")) %>%
  select(classification, month)

#png("./data_analysis_output/cables_n_month_class.png", width = 600, height = 450)
ggplot(classification_doc3, aes(month)) +
  geom_bar(aes(fill=classification)) +
  scale_x_date(breaks=scales::pretty_breaks(10)) +
  scale_y_continuous(breaks=scales::pretty_breaks(10)) +
  labs(#title = "",
       #subtitle = "Data Plotted by Year",
       y = "Number of Documents",
       x = "Month") +
  scale_fill_grey(start=0.8, end=0.2) +
  theme_bw() +
  theme(text = element_text(size=15),
        axis.text.x = element_text(size=11),
```

```
axis.text.y = element_text(size=11)#,
#legend.title=element_blank()#,
#legend.position = c(0.1, 0.9),
#legend.justification = c(0.1, 0.9)
) + labs(fill = "Classification")
```



```
#scale_fill_manual(
#   values = cols,
#   aesthetics = c("colour", "fill"),
#   breaks=c("Secret", "Confidential", "Limited Official Use", "Unclassified")
# )

#dev.off()
```

4.2 Example Cable

```
cables_db = dbConnect(driver='history-lab.org', password='XreadF403', user='de_reader', dbname='de')

dbListTables(cables_db)

## [1] "classification_countries" "classification_doc"
## [3] "classifications"         "concept_doc"
## [5] "concepts"                "countries"
## [7] "country_doc"             "doc_counts"
## [9] "docs"                    "from_to_sum"
## [11] "network_docs"            "network_nodes"
```

```
## [13] "office_doc"          "offices"
## [15] "person_doc"          "persons"
## [17] "reference_doc"       "tag_doc"
## [19] "tag_doc_staging"    "tagname_doc"
## [21] "tagnames"           "tags"
## [23] "tags_staging"       "tokens"
## [25] "top_classifications" "top_countries"
## [27] "top_network"        "top_persons"
## [29] "top_topics"         "topic_doc"
## [31] "topic_token"        "topics"
## [33] "urgency"            "urgency_doc"
```

```
tbl(cables_db, 'tag_doc') %>%
  filter(doc_id=="1976BAGHDA01815")
```

```
## # Source:   lazy query [?? x 2]
## # Database: mysql 5.7.26-0ubuntu0.16.04.1
## #   [de_reader@history-lab.org:/declassification_cables]
##   tag_id doc_id
##   <int> <chr>
## 1    117 1976BAGHDA01815
## 2    118 1976BAGHDA01815
## 3    441 1976BAGHDA01815
```

```
tbl(cables_db, 'tags') %>%
  filter(id==117 | id==118 | id==441)
```

```
## # Source:   lazy query [?? x 7]
## # Database: mysql 5.7.26-0ubuntu0.16.04.1
## #   [de_reader@history-lab.org:/declassification_cables]
##   id tag title description class category action
##   <int> <chr> <chr> <chr> <chr> <chr> <chr>
## 1  117 PINS Internal ~ Use for papers dealing wi~ politi~ subject <NA>
## 2  118 PINT Internal ~ Use for all aspects of a ~ politi~ subject <NA>
## 3  441 IZ Iraq <NA> country geograp~ <NA>
```

```
tbl(cables_db, 'country_doc') %>%
  filter(doc_id=="1976BAGHDA01815")
```

```
## # Source:   lazy query [?? x 4]
## # Database: mysql 5.7.26-0ubuntu0.16.04.1
## #   [de_reader@history-lab.org:/declassification_cables]
##   country_id doc_id country_count date
##   <chr> <chr> <int> <chr>
## 1 368 1976BAGHDA01815 NA 1976-12-11 00:00:00
```

```
tbl(cables_db, 'countries') %>%
  filter(id==368)
```

```
## # Source:   lazy query [?? x 5]
## # Database: mysql 5.7.26-0ubuntu0.16.04.1
## #   [de_reader@history-lab.org:/declassification_cables]
##   id name deleted official tag_id
##   <chr> <chr> <int> <int> <int>
## 1 368 Iraq 0 1 441
```

```
tbl(cables_db, 'topic_doc') %>%
  filter(doc_id=="1976BAGHDA01815")
```

```
## # Source:   lazy query [?? x 3]
## # Database: mysql 5.7.26-0ubuntu0.16.04.1
## #   [de_reader@history-lab.org:/declassification_cables]
##   doc_id      topic_id topic_score
##   <chr>      <int>    <dbl>
## 1 1976BAGHDA01815      31      0.0316
## 2 1976BAGHDA01815      15      0.0380
## 3 1976BAGHDA01815       0      0.0506
```

```
tbl(cables_db, 'topics') %>%
  filter(id==31 | id==15 | id==0)
```

```
## # Source:   lazy query [?? x 3]
## # Database: mysql 5.7.26-0ubuntu0.16.04.1
## #   [de_reader@history-lab.org:/declassification_cables]
##   id title      name
##   <int> <chr>      <chr>
## 1     0 {sadat, saudi, plo} <NA>
## 2    15 {la, film, nous} <NA>
## 3    31 {panama, peace, christian} <NA>
```

4.3 Frequency Tables

4.3.1 TABLE: Number of Documents with Non-Missing Values by Variable

```
#driver = dbDriver("MySQL")
#connection = dbConnect(driver, host='history-lab.org', password='XreadF403', user='de_reader')
#mydb = dbConnect(driver, host='history-lab.org', password='XreadF403', user='de_reader', dbname='declas')

#cfpf_docs <- tbl(mydb, 'docs') %>%
# collect()
#save("/Users/clarahsuong/Dropbox/nyu_postdoc/ner/dataset_intro/cfpf_docs.RData")
#load("/Users/clarahsuong/Dropbox/nyu_postdoc/ner/dataset_intro/cfpf_docs.RData")
#cables_docs<-docs

setwd("/Users/clarahsuong/chronos_data_intro")

docs<-
  cables_docs %>%
  dplyr::select("collection",
    "id",
    "body",
    "date",
    "classification",
    "subject",
    "from_field",
    "to_field",
    #"tags",
    "concepts",
    "office",
```

```

    "handling",
    "type")

C1<-c("collection",
    "id",
    "body",
    "date",
    "classification",
    "subject",
    "from_field",
    "to_field",
    #"tags",
    "concepts",
    "office",
    "handling",
    "type")

C2<-c(
sum(!is.na(docs$collection)),
sum(!is.na(docs$id)),
sum(!is.na(docs$body)),
sum(!is.na(docs$date)),
sum(!is.na(docs$classification)),
sum(!is.na(docs$subject)),
sum(!is.na(docs$from_field)),
sum(!is.na(docs$to_field)),
sum(!is.na(docs$concepts)),
sum(!is.na(docs$office)),
sum(!is.na(docs$type))
)

table_cables_n_na<-cbind(C1, C2)

## Warning in cbind(C1, C2): number of rows of result is not a multiple of
## vector length (arg 2)

colnames(table_cables_n_na) <- c("Variable","Number of Documents with Non-Missing Values")

stargazer(table_cables_n_na,
    summary = FALSE,
    rownames = FALSE,
    type = "text",
    title="Number of Documents with Non-Missing Values by Variable",
    digits=1,
    out="./data_analysis_output/table_cables_n_na.txt"
)

##
## Number of Documents with Non-Missing Values by Variable
## =====
## Variable      Number of Documents with Non-Missing Values
## -----
## collection           3214293
## id                   3214293
## body                 2654414

```

```
## date 3214293
## classification 2654414
## subject 2876678
## from_field 3214094
## to_field 3213050
## concepts 3063262
## office 2654414
## handling 2654414
## type 3214293
## -----

#stargazer(table_cables_n_na,
#          summary = FALSE,
#          rownames = FALSE,
#          type = "html",
#          title="Number of Documents with Non-Missing Values by Variable",
#          digits=1,
#          out="./data_analysis_output/table_cables_n_na.html"
#          )
```

4.3.2 TABLE: Number of Cables by Year

```
setwd("/Users/clarahsuong/chronos_data_intro")

table_cables_n_year<-
  cables_docs %>%
  mutate(year=lubridate::year(date)) %>%
  group_by(year) %>%
  tally() %>%
  mutate(total_n = sum(n),
         rel.freq = paste0(round(100 * n/total_n, 2), "%")) %>%
  select(year, n, rel.freq) %>%
  adorn_totals("row")

stargazer(table_cables_n_year[c("year","n", "rel.freq")],
          summary = FALSE,
          rownames = FALSE,
          type = "text",
          title="Number of Cables By Year",
          digits=1,
          out="./data_analysis_output/table_cables_n_year.txt",
          covariate.labels=c("Year","Number of Cables", "Relative Frequency")
          )
```

```
##
## Number of Cables By Year
## =====
## Year   Number of Cables Relative Frequency
## -----
## 1973      179253           5.58%
## 1974      442301          13.76%
## 1975      531102          16.52%
## 1976      554864          17.26%
```

```
## 1977      474671      14.77%
## 1978      500577      15.57%
## 1979      531525      16.54%
## Total    3214293      -
## -----
```

```
#stargazer(table_cables_n_year[c("year", "n", "rel.freq")],
#          summary = FALSE,
#          rownames = FALSE,
#          type = "html",
#          title="Number of Cables By Year",
#          digits=1,
#          out="./data_analysis_output/table_cables_n_year.html",
#          covariate.labels=c("Year", "Number of Cables", "Relative Frequency")
#          )
```

4.3.3 TABLE: Number of Cables by Classification

```
setwd("/Users/clarahsuong/chronos_data_intro")

#driver = dbDriver("MySQL")
#connection = dbConnect(driver, host='history-lab.org', password='XreadF403', user='de_reader')
cables_db = dbConnect(driver, host='history-lab.org', password='XreadF403', user='de_reader', dbname='de_reader')

classification_doc2 <- tbl(cables_db, 'classification_doc') %>%
  collect() %>%
  distinct() %>%
  group_by(classification_id) %>%
  tally() %>%
  ungroup() %>%
  mutate(total_n = sum(n),
         rel.freq = paste0(round(100 * n/total_n, 2), "%"),
         classification=ifelse(classification_id==1,"Secret",
                               ifelse(classification_id==2,"Confidential",
                                       ifelse(classification_id==5,"Unclassified",
                                             ifelse(classification_id==7,"Limited Official Use", NA)
                                             )
                               )
         )
  ) %>%
  select(classification, n, rel.freq) %>%
  adorn_totals("row")

#classification_doc=apply_labels(classification_doc,
#                                classification_id="Classification",
#                                classification_id=num_lab("1 Secret
#                                2 Confidential
#                                7 Limited Official Use
#                                5 Unclassified")
#                                )

#table_classification = fre(classification_doc$classification_id) %>%
# set_caption("Table: Documents by Classification") %>%
```



```
# htmlTable()

stargazer(classification_doc2[c("classification","n", "rel.freq")],
  summary = FALSE,
  rownames = FALSE,
  type = "text",
  title="Number of Documents By Classification Level",
  digits=1,
  out="./data_analysis_output/table_cables_n_class.txt",
  covariate.labels=c("Classification","Number of Documents", "Relative Frequency"))

##
## Number of Documents By Classification Level
## =====
## Classification      Number of Documents Relative Frequency
## -----
## Secret              127332              4.8%
## Confidential        494823              18.64%
## Unclassified        1518305             57.2%
## Limited Official Use  513769             19.36%
## Total               2654229              -
## -----

#stargazer(classification_doc2[c("classification","n", "rel.freq")],
#  summary = FALSE,
#  rownames = FALSE,
#  type = "html",
#  title="Number of Documents By Classification Level",
#  digits=1,
#  out="./data_analysis_output/table_cables_n_class.html",
#  covariate.labels=c("Classification","Number of Documents", "Relative Frequency"))
```

5 FRUS Collection

5.1 FIGURE: Number of Documents by Year and Classification

```
setwd("/Users/clarahsuong/chronos_data_intro")

frus_n_date<-
  frus_docs %>%
  dplyr::select(id, date, classification) %>%
  mutate(date=as_date(date),
    Classification = replace_na(classification, "Missing"),
    year = as_date(cut(date, breaks = "year")),
    Classification =factor(Classification, levels = c("Missing", "Confidential","Secret","Top Secret"))

#cols <- c(
#  #"Confidential" =
#  #"999999",
#  #"Missing" =
#  #"CCCCC",
#  #"Secret" =
```

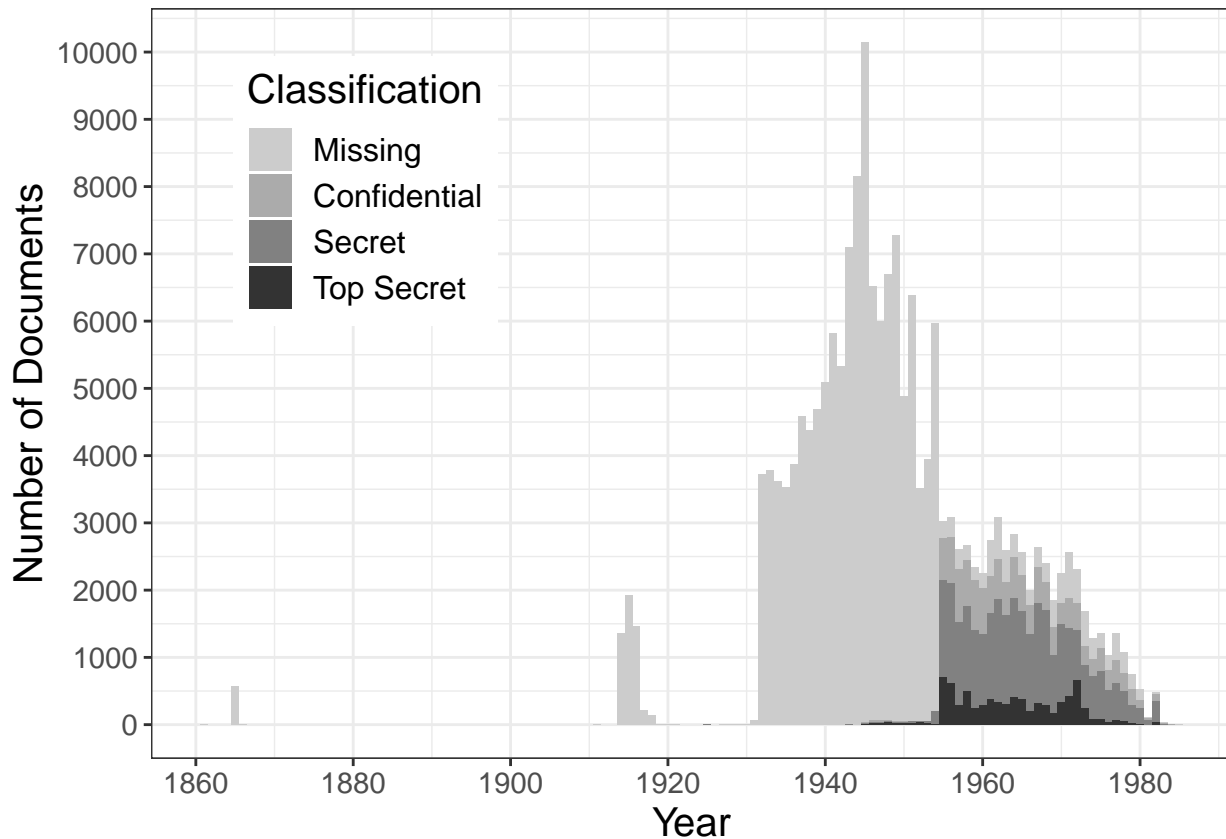
```

#           "#666666",
#           #"Top Secret" =
#           "#333333")

#png("./data_analysis_output/frus_n_year_class.png", width = 600, height = 450)
#layout(matrix(c(1:3), 3, 1,
# byrow = TRUE))
ggplot(frus_n_date, aes(year)) +
  #geom_bar()
  geom_bar(aes(fill=Classification)) +
  scale_x_date(breaks=scales::pretty_breaks(10)) +
  scale_y_continuous(breaks=scales::pretty_breaks(10)) +
  labs(#title = "",
        #subtitle = "Data Plotted by Year",
        y = "Number of Documents",
        x = "Year") +
#  scale_fill_manual(
#    values = cols,
#    aesthetics = c("colour", "fill"),
#    breaks=c("Top Secret", "Secret", "Confidential", "Missing")
#  ) +
  theme_bw() +
  theme(text = element_text(size=15),
        axis.text.x = element_text(size=11),
        axis.text.y = element_text(size=11),
        #legend.title=element_blank(),
        legend.position = c(0.1, 0.9),
        legend.justification = c(0.1, 0.9)) +
  scale_fill_grey(start=0.8, end=0.2) #+

```

```
## Warning: Removed 22767 rows containing non-finite values (stat_count).
```



```
# scale_fill_discrete(breaks=c("Missing", "Confidential", "Secret", "Top Secret"))
#dev.off()
```

5.2 Example Document

```
#driver = dbDriver("MySQL")
#connection = dbConnect(driver, host='history-lab.org', password='XreadF403', user='de_reader')
frus_db = dbConnect(driver, host='history-lab.org', password='XreadF403', user='de_reader', dbname='decl')

dbListTables(frus_db)

## [1] "authorship"           "classification_countries"
## [3] "classification_doc"    "classification_persons"
## [5] "classification_topics" "classifications"
## [7] "countries"            "country_doc"
## [9] "country_doc_bak"      "country_doc_staging"
## [11] "curated_topics"       "doc_counts"
## [13] "docs"                 "docs_bak"
## [15] "old_classification_topics" "old_top_topics"
## [17] "old_topic_doc"        "old_topics"
## [19] "person_doc"           "persons"
## [21] "persons_master"       "refs"
## [23] "term_doc"             "terms"
## [25] "tokens"               "top_classifications"
## [27] "top_countries"        "top_persons"
## [29] "top_topics"           "topic_doc"
```

```
## [31] "topic_token"          "topics"
## [33] "volumes"

tbl(frus_db, 'country_doc') %>%
  filter(doc_id=="frus1945v02d128")

## # Source:   lazy query [?? x 4]
## # Database: mysql 5.7.26-0ubuntu0.16.04.1
## #   [de_reader@history-lab.org:/declassification_frus]
##   country_id doc_id          country_count date
##   <chr>      <chr>              <int> <chr>
## 1 156        frus1945v02d128          3 1945-09-22 00:00:00
## 2 250        frus1945v02d128          3 1945-09-22 00:00:00

tbl(frus_db, 'countries') %>%
  filter(id==156 | id==250)

## # Source:   lazy query [?? x 4]
## # Database: mysql 5.7.26-0ubuntu0.16.04.1
## #   [de_reader@history-lab.org:/declassification_frus]
##   id  name  deleted official
##   <chr> <chr>   <int>   <int>
## 1 156  China    0        1
## 2 250  France   0        1

tbl(frus_db, 'topic_doc') %>%
  filter(doc_id=="frus1945v02d128")

## # Source:   lazy query [?? x 4]
## # Database: mysql 5.7.26-0ubuntu0.16.04.1
## #   [de_reader@history-lab.org:/declassification_frus]
##   doc_id      topic_id topic_score date
##   <chr>      <int>      <dbl> <chr>
## 1 frus1945v02d128    1059      0.0426 1945-09-22 00:00:00
## 2 frus1945v02d128    1062      0.0567 1945-09-22 00:00:00
## 3 frus1945v02d128    1069      0.0426 1945-09-22 00:00:00

tbl(frus_db, 'topics') %>% #Replace with 'curated_topics' later.
  filter(id==1059 | id==1062 | id==1069)

## # Source:   lazy query [?? x 3]
## # Database: mysql 5.7.26-0ubuntu0.16.04.1
## #   [de_reader@history-lab.org:/declassification_frus]
##   id title                      name
##   <int> <chr>                          <chr>
## 1 1059 {each, missile, threat} Conventions Conferences and Negotiations
## 2 1062 {system, message, radio} <NA>
## 3 1069 {bank, price, credit} Eximbank and Foreign Credit

a<-frus_docs %>%
  filter(id=="frus1945v02d128")
```

5.3 Frequency Tables

5.3.1 TABLE: Number of Documents with Non-Missing Values by Variable

```
setwd("/Users/clarahsuong/chronos_data_intro")

C1<-c("collection",
      "id",
      "body",
      "date",
      "classification",
      "volume_id",
      "chapt_title",
      "title",
      #"subject",
      #"location",
      "p_from",
      "p_to",
      "source"
)

C2<-c(sum(!is.na(frus_docs$collection)),
      sum(!is.na(frus_docs$id)),
      sum(!is.na(frus_docs$body)),
      sum(!is.na(frus_docs$date)),
      sum(!is.na(frus_docs$classification)),
      sum(!is.na(frus_docs$volume_id)),
      sum(!is.na(frus_docs$chapt_title)),
      sum(!is.na(frus_docs$title)),
      sum(!is.na(frus_docs$p_from)),
      sum(!is.na(frus_docs$p_to)),
      sum(!is.na(frus_docs$source))
)

table_frus_n_na<-cbind(C1, C2)
colnames(table_frus_n_na) <- c("Variable","Number of Documents with Non-Missing Values")

#htmlTable(ns,
#          ctable=c("solid", "double"),
#          caption="Number of Documents with Non-Missing Values")

stargazer(table_frus_n_na,
           summary = FALSE,
           rownames = FALSE,
           type = "text",
           title="Number of Documents with Non-Missing Values by Variable",
           digits=1,
           out="./data_analysis_output/table_frus_n_na.txt"
)

##
## Number of Documents with Non-Missing Values by Variable
## =====
## Variable      Number of Documents with Non-Missing Values
```

```
## -----
## collection                209046
## id                        209046
## body                      209046
## date                      186279
## classification            52580
## volume_id                 209046
## chapt_title               178050
## title                     209034
## p_from                    97657
## p_to                      51797
## source                    59028
## -----

#stargazer(table_frus_n_na,
#           summary = FALSE,
#           rownames = FALSE,
#           type = "html",
#           title="Number of Documents with Non-Missing Values by Variable",
#           digits=1,
#           out="./data_analysis_output/table_frus_n_na.html"
#           )
```

5.3.2 TABLE: Number of Documents by Year

```
setwd("/Users/clarahsuong/chronos_data_intro")

table_frus_n_year<-
  frus_docs %>%
  mutate(year=lubridate::year(date)) %>%
  group_by(year) %>%
  tally() %>%
  mutate(total_n = sum(n),
         rel.freq = paste0(round(100 * n/total_n, 2), "%")) %>%
  ungroup() %>%
  adorn_totals("row")

stargazer(table_frus_n_year[c("year","n", "rel.freq")],
          summary = FALSE,
          rownames = FALSE,
          type = "text",
          title="Number of Documents By Year",
          digits=1,
          out="./data_analysis_output/table_frus_n_year.txt",
          covariate.labels=c("Year", "Number of Documents", "Relative Frequency")
          )

##
## Number of Documents By Year
## =====
## Year   Number of Documents Relative Frequency
## -----
## 1861           1                0%
```

## 1865	565	0.27%
## 1866	3	0%
## 1911	2	0%
## 1914	1360	0.65%
## 1915	1921	0.92%
## 1916	1464	0.7%
## 1917	209	0.1%
## 1918	147	0.07%
## 1919	6	0%
## 1920	1	0%
## 1921	1	0%
## 1925	1	0%
## 1927	2	0%
## 1928	1	0%
## 1929	10	0%
## 1930	11	0.01%
## 1931	71	0.03%
## 1932	3726	1.78%
## 1933	3777	1.81%
## 1934	3616	1.73%
## 1935	3533	1.69%
## 1936	3877	1.85%
## 1937	4584	2.19%
## 1938	4380	2.1%
## 1939	4692	2.24%
## 1940	5099	2.44%
## 1941	5817	2.78%
## 1942	5327	2.55%
## 1943	7094	3.39%
## 1944	8162	3.9%
## 1945	10144	4.85%
## 1946	6519	3.12%
## 1947	6005	2.87%
## 1948	6689	3.2%
## 1949	7275	3.48%
## 1950	4887	2.34%
## 1951	6390	3.06%
## 1952	3514	1.68%
## 1953	3953	1.89%
## 1954	5975	2.86%
## 1955	3026	1.45%
## 1956	3083	1.47%
## 1957	2613	1.25%
## 1958	2677	1.28%
## 1959	2341	1.12%
## 1960	2248	1.08%
## 1961	2741	1.31%
## 1962	3078	1.47%
## 1963	2589	1.24%
## 1964	2827	1.35%
## 1965	2558	1.22%
## 1966	1998	0.96%
## 1967	2638	1.26%
## 1968	2393	1.14%

```
## 1969      1852      0.89%
## 1970      2244      1.07%
## 1971      2565      1.23%
## 1972      2303      1.1%
## 1973      1686      0.81%
## 1974      1284      0.61%
## 1975      1362      0.65%
## 1976      1023      0.49%
## 1977      1360      0.65%
## 1978      1073      0.51%
## 1979       753      0.36%
## 1980       527      0.25%
## 1981       117      0.06%
## 1982       474      0.23%
## 1983        25      0.01%
## 1984         8       0%
## 1985         2       0%
##          22767     10.89%
## Total    209046      -
## -----
```

```
#stargazer(table_frus_n_year[c("year", "n", "rel.freq")],
#          summary = FALSE,
#          rownames = FALSE,
#          type = "html",
#          title="Number of Documents By Year",
#          digits=1,
#          out="./data_analysis_output/table_frus_n_year.html",
#          covariate.labels=c("Year", "Number of Documents", "Relative Frequency")
#          )
```

5.3.3 TABLE: Number of Documents by Classification

```
setwd("/Users/clarahsuong/chronos_data_intro")

table_frus_n_class<-
  frus_docs %>%
  mutate(year=lubridate::year(date)) %>%
  group_by(classification) %>%
  tally() %>%
  mutate(total_n = sum(n),
         rel.freq = paste0(round(100 * n/total_n, 2), "%")) %>%
  ungroup() %>%
  adorn_totals("row")

stargazer(table_frus_n_class[c("classification", "n", "rel.freq")],
          summary = FALSE,
          rownames = FALSE,
          type = "text",
          title="Number of Documents By Classification Level",
          digits=1,
          out="./data_analysis_output/table_frus_n_class.txt",
          covariate.labels=c("Classification", "Number of Documents", "Relative Frequency"))
```



```
##
## Number of Documents By Classification Level
## =====
## Classification Number of Documents Relative Frequency
## -----
##           156466           74.85%
## Confidential    13512           6.46%
## Secret          29937           14.32%
## Top Secret      9131            4.37%
## Total           209046           -
## -----

#stargazer(table_frus_n_class[c("classification","n", "rel.freq")],
#          summary = FALSE,
#          rownames = FALSE,
#          type = "html",
#          title="Number of Documents By Classification Level",
#          digits=1,
#          out="./data_analysis_output/table_frus_n_class.html",
#          covariate.labels=c("Classification", "Number of Documents", "Relative Frequency"))
```

6 Country TAG Traffic

6.1 Examine the different country codes across datasets

```
setwd("/Users/clarahsuong/chronos_data_intro")

#Re-connect to the database
#driver = dbDriver("MySQL")
#connection = dbConnect(driver,host='history-lab.org', password='XreadF403', user='de_reader')
mydb = dbConnect(driver,host='history-lab.org', password='XreadF403', user='de_reader', dbname='declass')

#A list of countries according to our database
countries<-
  tbl(mydb, 'countries') %>%
  collect()

#This table is incomplete. Note that there is no tag for "South Vietnam" but tag "VM" (id: 557) for "Vi
#US is included.
#Note there is no tag_id for the Soviet Union but one for Russia.

#Merge ISO_3166_1 and ISO_3166_3 (ISO country codes for withdrawn countries). Note that this list often
iso_3166<-
  tibble::as_tibble(full_join(ISO_3166_1, ISO_3166_3, by = c("Alpha_3","Numeric","Name")))%>%
  mutate(Numeric=as.integer(Numeric)) %>%
  dplyr::select("Alpha_3",
               "Numeric",
               "Name",
               "Official_name",
               "Common_name")

#Generate a dataframe for all (former and existing) countries according to COW. Note that this includes
all_states<-
```

```

read_csv("./external_data/cow/states2016.csv") %>%
dplyr::select("stateabb", "ccode", "statenme") %>%
#filter(!ccode==2) %>% #Leave out the US
rename(cow_ccode=ccode,
        cow_stateabb=stateabb,
        cow_statename=statenme) %>%
mutate(cow_stateabb=as.character(cow_stateabb),
        cow_statename=as.character(cow_statename)) %>%
distinct() #There are duplicates. e.g. countries that existed, disappeared, and then re-appeared.

## Parsed with column specification:
## cols(
##   stateabb = col_character(),
##   ccode = col_double(),
##   statenme = col_character(),
##   styear = col_double(),
##   stmonth = col_double(),
##   stday = col_double(),
##   endyear = col_double(),
##   endmonth = col_double(),
##   endday = col_double(),
##   version = col_double()
## )

#Generate a dataframe for all (former and existing) countries for years 1973-79. Note that this includes
all_states_year<-
  all_states %>%
  rowr::cbind.fill(c(1973:1979), fill = NA) %>%
  rename(year=object) %>%
  expand(year = 1973:1979, nesting(cow_stateabb,
                                   cow_ccode,
                                   cow_statename))

#Generate a dataframe for countries existing during the period of 1973-79. Note that the universe of countries
states_70s_year<-
  read_csv("./external_data/cow/system2016.csv") %>%
  dplyr::select("stateabb", "ccode", "year") %>%
  filter(year>1972 & year<1980) %>%
  rename(cow_ccode=ccode,
        cow_stateabb=stateabb) %>%
  left_join(all_states, by=c("cow_ccode", "cow_stateabb")) #Include COW state names.

## Parsed with column specification:
## cols(
##   stateabb = col_character(),
##   ccode = col_double(),
##   year = col_double(),
##   version = col_double()
## )

states_70s<-
  states_70s_year %>%
  dplyr::select(-year) %>%
  distinct()

```

6.2 Create a dataframe linking country codes and tag_ids

```
#Re-connect to the database
#driver = dbDriver("MySQL")
#connection = dbConnect(driver,host='history-lab.org', password='XreadF403', user='de_reader')
#mydb = dbConnect(driver,host='history-lab.org', password='XreadF403', user='de_reader', dbname='declass')

#Note that this includes country codes and tag_id for the US.
country_code_tag<-
  tbl(mydb, 'countries') %>%
  collect() %>%
  mutate(country_id=as.integer(id)) %>%
  dplyr::select(-id) %>%
  mutate(cow_ccode=countrycode(name, 'country.name', 'cown')) %>% #Derive COW country codes from the va
  mutate(iso3n=countrycode(name, 'country.name', 'iso3n')) #Derive iso numeric country codes from the v

## Warning in countrycode(name, "country.name", "cown"): Some values were not matched unambiguously: Al
## Warning in countrycode(name, "country.name", "iso3n"): Some values were not matched unambiguously: A
## Warning in countrycode(name, "country.name", "iso3n"): Some strings were matched more than once, and

#Check whether the variable "country_id" in the table "countries" is from ISO 3166.
#cow_ccode for Vietnam should be 816, not 817 (error in the package countrycode) and cow_ccode for West
#iso3n for South Vietnam should not be 704 (country code of Vietnam) but 714 (error in the R package co
#country_code_tag$cowid2<-countrycode(country_code_tag$country_id, 'iso3n', 'cown')

all(country_code_tag$country_id %in% iso_3166$Numeric)

## [1] FALSE

all(iso_3166$Numeric %in% country_code_tag$country_id)

## [1] FALSE

setdiff(country_code_tag$country_id, iso_3166$Numeric)

## [1] 80 230 274 280 282 284 532 590 594 650 658 698 714 736 830 886 890
## [18] 900 901 902 903 904 905 906 907 908 909 910 911 912 913 914 915 916
## [35] 917 918 919 920 921 922 923 924 925 926 927 928 929 930 931 932

country_code_tag[country_code_tag$country_id %in% setdiff(country_code_tag$country_id, iso_3166$Numeric)]

## # A tibble: 50 x 7
##   name                deleted official tag_id country_id cow_ccode iso3n
##   <chr>              <int>    <int> <int>      <int>    <int> <int>
## 1 British Antarctic Te~    1        0    NA         80        NA    NA
## 2 Ethiopia              1        0   405        230       530   231
## 3 Gaza Strip            0        0   417        274        NA   275
## 4 West Germany          1        0   418        280       255   276
## 5 East Berlin           0        0    NA        282        NA    NA
## 6 West Berlin           0        0   560        284        NA    NA
## 7 Netherlands Antilles~  1        0   484        532        NA   533
## 8 Panama                1        0    NA        590        95   591
## 9 Panama Canal Zone     1        0   378        594        95   591
## 10 Ryukyu Islands       0        0   511        650        NA    NA
## # ... with 40 more rows
```

```

#Most of the items with a discrepancy between the database's country_id and iso-3166 numeric seem to be

#Replace the wrong COW country codes and tag_id
country_code_tag<-
  country_code_tag %>%
  mutate(cow_ccode= replace(cow_ccode, name=="Vietnam", 816)) %>% #Fix cow_ccode for Vietnam
  mutate(cow_ccode= replace(cow_ccode, name=="West Germany", 260)) %>% #Fix cow_ccode for West Germany
  mutate(tag_id=replace(tag_id, name=="South Vietnam", 1973)) %>% #Insert tag_id for South Vietnam
  rbind(c("Vietnam",
    0,
    1,
    1976,
    704,
    816,
    704
  )
  ) %>% #Insert the second tag_id value (1976) for Vietnam
  mutate(cow_ccode=as.integer(cow_ccode),
    tag_id=as.integer(tag_id)) %>%
  inner_join(all_states, by="cow_ccode") %>% #Include state names from the COW list of all states that
  rename(country_name=name) %>%
  filter(!is.na(cow_ccode) & !is.na(tag_id)) %>% #Drop the observations with missing COW country code a
  dplyr::select(-iso3n) #Note the 2 tags for Vietnam.

```

6.3 Tag traffic by country-year and by country

6.3.1 Download, save, or load the tables for tags and docs (doc_id and date) in the working directory and count the number of cables tagged for each country

```

setwd("/Users/clarahsuong/chronos_data_intro")

#Re-connect to the database
#driver = dbDriver("MySQL")
#connection = dbConnect(driver,host='history-lab.org', password='XreadF403', user='de_reader')
#mydb = dbConnect(driver,host='history-lab.org', password='XreadF403', user='de_reader', dbname='declass

tags<-
  tbl(mydb, 'tags') %>%
  dplyr::select(id, tag, category) %>%
  collect()

tag_doc<-
  tbl(mydb, 'tag_doc') %>%
  collect() #This table includes cables tagged with South Vietnam (tag_id 1973) and Vietnam (tag_id: 19
#tag_doc %>% filter(tag_id==1973)
#tag_doc %>% filter(tag_id==1976)
#tag_doc %>% filter(tag_id==557)

doc_date2<-
  tbl(mydb, 'docs') %>%
  dplyr::select(id, date) %>%
  rename(doc_id=id) %>%

```

```

collect()

country_tag_doc2<-
  tag_doc %>%
  inner_join(doc_date2, by = "doc_id") %>%
  inner_join(tags, by = c("tag_id"="id")) %>%
  inner_join(country_code_tag, by="tag_id") %>%
  mutate(year=lubridate::year(date),
         month=lubridate::month(date),
         date=lubridate::ymd(date),
         ym=as.yearmon(paste(year, month),"%Y %m")
  )

#save(country_tag_doc2, file = "./data/country_tag_doc2.RData")
#load("./data/country_tag_doc2.RData")

cable_n_country_day<-
  country_tag_doc2 %>%
  group_by(.dots=c("cow_ccode",
                  "cow_statename",
                  "country_id",
                  "country_name",
                  "date")) %>%

  tally() %>%
  ungroup()

#save(cable_n_country_day, file = "./data/cable_n_country_day.RData")
#load("./data/cable_n_country_day.RData") #Note that this includes neither all dates nor all countries

#The table country_doc attempts to add West Germany and South Vietnam based on regex matching in body.
#country_doc<-
# tbl(mydb, 'country_doc') %>% collect()
#However, this table is also missing South Vietnam (country_id 714 or tag_id 1973). It seems to group (
#country_doc %>% filter(country_id==714)
#country_doc %>% filter(country_id==704)
#It is meaningful to distinguish cables related to South Vietnam from those about (North) Vietnam. Thus

#Yearly tag traffic by state-year, including 0 cables by some countries that did not exist in the 1970s
cable_n_all_states_year<-
  country_tag_doc2 %>%
  group_by(year, cow_ccode, cow_statename, cow_stateabb) %>%
  tally() %>%
  right_join(all_states_year, by=c("year", "cow_ccode", "cow_stateabb","cow_statename")) %>%
  rename(n_c_y=n) %>%
  mutate(n_c_y= replace(n_c_y, is.na(n_c_y), 0)) %>%
  ungroup() %>%
  mutate(total_n = sum(n_c_y)) %>%
  arrange(year, cow_ccode)

## Warning: Column `cow_stateabb` joining character vector and factor,
## coercing into character vector

## Warning: Column `cow_statename` joining character vector and factor,
## coercing into character vector

```

```

#save(cable_n_all_states_year, file = "./data/cable_n_all_states_year.RData")
#load("./data/cable_n_all_states_year.RData")

#Tag traffic by state, including 0 cables by some countries that did not exist in the 1970s. Note that
cable_n_all_states<-
  country_tag_doc2 %>%
  group_by(cow_ccode, cow_statename, cow_stateabb) %>%
  tally() %>%
  right_join(all_states, by=c("cow_ccode", "cow_stateabb", "cow_statename")) %>%
  rename(n_c=n) %>%
  mutate(n_c= replace(n_c, is.na(n_c), 0)) %>%
  ungroup() %>%
  mutate(total_n = sum(n_c)) %>%
  arrange(desc(n_c))
#save(cable_n_all_states, file = "./data/cable_n_all_states.RData")
#load("./data/cable_n_all_states.RData")

#Yearly tag traffic by state-year, excluding 0 cables by some countries that did not exist in the 1970s
cable_n_states_70s_year<-
  country_tag_doc2 %>%
  group_by(year, cow_ccode, cow_statename, cow_stateabb) %>%
  tally() %>%
  right_join(states_70s_year, by=c("year", "cow_ccode", "cow_stateabb", "cow_statename")) %>%
  rename(n_c_y=n) %>%
  mutate(n_c_y= replace(n_c_y, is.na(n_c_y), 0)) %>%
  ungroup() %>%
  mutate(total_n = sum(n_c_y)) %>%
  arrange(year, cow_ccode)
#save(cable_n_states_70s_year, file = "./data/cable_n_states_70s_year.RData")
#load("./data/cable_n_states_70s_year.RData")

#Tag traffic by state, excluding 0 cables by some countries that did not exist in the 1970s. Note that
cable_n_states_70s<-
  country_tag_doc2 %>%
  group_by(cow_ccode, cow_statename, cow_stateabb) %>%
  tally() %>%
  right_join(states_70s, by=c("cow_ccode", "cow_stateabb", "cow_statename")) %>%
  rename(n_c=n) %>%
  mutate(n_c= replace(n_c, is.na(n_c), 0)) %>%
  ungroup() %>%
  mutate(total_n = sum(n_c)) %>%
  arrange(desc(n_c))
#save(states_70s, file = "./data/cable_n_states_70s.RData")
#load("./data/cable_n_states_70s.RData")

#Note that the total ns for each dataset for differs a bit.

```

6.3.2 TABLE: Summary Statistics of Country TAG Traffic by Country-Year (Only Contemporary Non-US Countries)

```
setwd("/Users/clarahsuong/chronos_data_intro")
```

```
stargazer(as.data.frame(cable_n_states_70s_year[cable_n_states_70s_year$cow_ccode!=2,])[c("year", "cow_
type = "text",
title="Summary Statistics of Tag Traffic by Country-Year (Only Contemporary Non-US Countries)",
digits=1,
out="./data_analysis_output/desc_cable_n_nonus_states_70s_year.txt",
covariate.labels=c("Year", "COW Codes of Countries", "Country TAG Traffic"))

##
## Summary Statistics of Tag Traffic by Country-Year (Only Contemporary Non-US Countries)
## =====
## Statistic          N      Mean   St. Dev.  Min  Pctl(25) Pctl(75)  Max
## -----
## Year                1,040 1,976.1    2.0    1,973  1,974    1,978  1,979
## COW Codes of Countries 1,040 460.3    247.3    20   253.8    663    990
## Country TAG Traffic   1,040 2,545.4 3,019.4    21   678.2   3,394.5 24,856
## -----
#stargazer(as.data.frame(cable_n_states_70s_year[cable_n_states_70s_year$cow_ccode!=2,])[c("year", "cow_
# type = "html",
# title="Summary Statistics of Tag Traffic by Country-Year (Only Contemporary Non-US Countries)",
# digits=1,
# out="./data_analysis_output/desc_cable_n_nonus_states_70s_year.html",
# covariate.labels=c("Year", "COW Codes of Countries", "Country TAG Traffic"))
```

6.3.3 TABLE: Summary Statistics of Country TAG Traffic by Country (Only Contemporary Non-US Countries)

```
setwd("/Users/clarahsuong/chronos_data_intro")

stargazer(as.data.frame(cable_n_states_70s[cable_n_states_70s$cow_ccode!=2,])[c("cow_ccode", "n_c")],
type = "text",
title="Summary Statistics of Tag Traffic by Country (Only Contemporary Non-US Countries)",
digits=1,
out="./data_analysis_output/desc_cable_n_nonus_states_70s.txt",
covariate.labels=c("COW Codes of Countries", "Country TAG Traffic"))

##
## Summary Statistics of Tag Traffic by Country (Only Contemporary Non-US Countries)
## =====
## Statistic          N      Mean   St. Dev.  Min  Pctl(25) Pctl(75)  Max
## -----
## COW Codes of Countries 156 459.6    253.6    20   233.8    663.8    990
## Country TAG Traffic   156 17,036.6 19,338.0 277  4,643   22,983.2 144,726
## -----
#stargazer(as.data.frame(cable_n_states_70s[cable_n_states_70s$cow_ccode!=2,])[c("cow_ccode", "n_c")],
# type = "html",
# title="Summary Statistics of Tag Traffic by Country (Only Contemporary Non-US Countries)",
# digits=1,
# out="./data_analysis_output/desc_cable_n_nonus_states_70s.html",
# covariate.labels=c("COW Codes of Countries", "Country TAG Traffic"))
```

6.3.4 FIGURE: Country TAG Traffic at Country-Year and Country Levels

```
setwd("/Users/clarahsuong/chronos_data_intro")

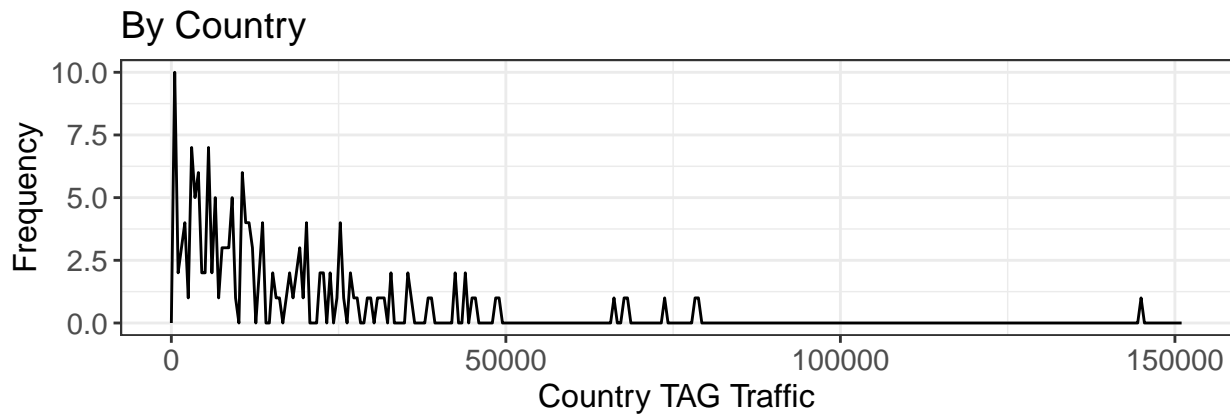
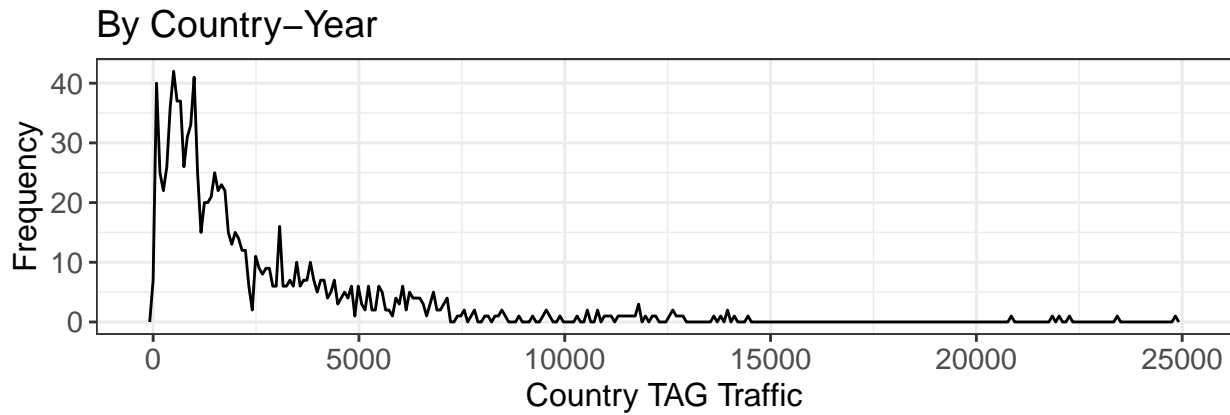
options(scipen=1000000)

p1<-ggplot(cable_n_states_70s_year[cable_n_states_70s_year$cow_ccode!=2,], aes(n_c_y)) +
  geom_freqpoly(bins = 300) +
  theme_bw() +
  labs(title = "By Country-Year",
       #subtitle = "Data Plotted by Year",
       y = "Frequency",
       x = "Country TAG Traffic"
  ) +
  theme(text = element_text(size=12),
        axis.text.x = element_text(size=11),
        axis.text.y = element_text(size=11)#,
        #legend.title=element_blank()#,
        #legend.position = c(0.1, 0.9),
        #legend.justification = c(0.1, 0.9)
  )

p2<-ggplot(cable_n_states_70s[cable_n_states_70s$cow_ccode!=2,], aes(n_c)) +
  geom_freqpoly(bins = 300) +
  theme_bw() +
  labs(title = "By Country",
       #subtitle = "Data Plotted by Year",
       y = "Frequency",
       x = "Country TAG Traffic") +
  theme(text = element_text(size=12),
        axis.text.x = element_text(size=11),
        axis.text.y = element_text(size=11)#,
        #legend.title=element_blank()#,
        #legend.position = c(0.1, 0.9),
        #legend.justification = c(0.1, 0.9)
  ) +
  xlim(0, 151000)

#png("./data_analysis_output/cable_n_nonus_states_70s_year_freq.png")
grid.arrange(p1, p2)

## Warning: Removed 2 rows containing missing values (geom_path).
```

```
#dev.off()
```

6.3.5 Percentile for Specific Values

```
ecdf_fun <- function(x,perc) ecdf(x)(perc)
ecdf_fun(cable_n_states_70s_year[cable_n_states_70s_year$cow_ccode!=2,]$n_c_y,5000)
```

```
## [1] 0.8586538
```

```
ecdf_fun(cable_n_states_70s_year[cable_n_states_70s_year$cow_ccode!=2,]$n_c_y,10000)-ecdf_fun(cable_n_s
```

```
## [1] 0.1048077
```

```
1-ecdf_fun(cable_n_states_70s_year[cable_n_states_70s_year$cow_ccode!=2,]$n_c_y,10000)
```

```
## [1] 0.03653846
```

```
ecdf_fun(cable_n_states_70s[cable_n_states_70s$cow_ccode!=2,]$n_c,25000)
```

```
## [1] 0.7692308
```

```
ecdf_fun(cable_n_states_70s[cable_n_states_70s$cow_ccode!=2,]$n_c,75000)-ecdf_fun(cable_n_states_70s[ca
```

```
## [1] 0.2115385
```

```
1-ecdf_fun(cable_n_states_70s[cable_n_states_70s$cow_ccode!=2,]$n_c,75000)
```

```
## [1] 0.01923077
```

6.3.6 TABLE: Summary Statistics of Country TAG Traffic by Country-Year (Including Former Countries and the US)

```
setwd("/Users/clarahsuong/chronos_data_intro")

stargazer(as.data.frame(cable_n_all_states_year)[c("year", "cow_ccode", "n_c_y")],
  type = "text",
  title="Summary Statistics of Country TAG Traffic by Country-Year (Incl. Former Countries and the US)",
  digits=1,
  out="./data_analysis_output/desc_cable_n_all_states_year.txt",
  covariate.labels=c("Year", "COW Codes of Countries", "Country TAG Traffic"))

##
## Summary Statistics of Country TAG Traffic by Country-Year (Incl. Former Countries and the US)
## =====
## Statistic          N      Mean   St. Dev.  Min  Pctl(25) Pctl(75)   Max
## -----
## Year                1,519 1,976.0    2.0    1,973  1,974    1,978    1,979
## COW Codes of Countries 1,519 460.0   256.6     2    271    670     990
## Country TAG Traffic   1,519 2,220.0 7,652.6    0    33.5  2,224.5 138,438
## -----

#stargazer(as.data.frame(cable_n_all_states_year)[c("year", "cow_ccode", "n_c_y")],
#  type = "html",
#  title="Summary Statistics of Country TAG Traffic by Country-Year (Incl. Former Countries and the US)",
#  digits=1,
#  out="./data_analysis_output/desc_cable_n_all_states_year.html",
#  covariate.labels=c("Year", "COW Codes of Countries", "Country TAG Traffic"))
```

6.3.7 TABLE: Summary Statistics of Country TAG Traffic by Country (Incl. Former Countries and the US)

```
setwd("/Users/clarahsuong/chronos_data_intro")

stargazer(as.data.frame(cable_n_all_states)[c("cow_ccode", "n_c")],
  type = "text",
  title="Summary Statistics of Country TAG Traffic by Country (Incl. Former Countries and the US)",
  digits=1,
  out="./data_analysis_output/desc_cable_n_all_states.txt",
  covariate.labels=c("COW Codes of Countries", "Country TAG Traffic"))

##
## Summary Statistics of Country TAG Traffic by Country (Incl. Former Countries and the US)
## =====
## Statistic          N      Mean   St. Dev.  Min  Pctl(25) Pctl(75)   Max
## -----
## COW Codes of Countries 217 460.0   257.1     2    271    670     990
## Country TAG Traffic    217 15,540.2 50,372.7    0    277   18,121 705,142
## -----

#stargazer(as.data.frame(cable_n_all_states)[c("cow_ccode", "n_c")],
#  type = "html",
#  title="Summary Statistics of Country TAG Traffic by Country (Incl. Former Countries and the US)",
#  digits=1,
#  out="./data_analysis_output/desc_cable_n_all_states.html",
#  covariate.labels=c("COW Codes of Countries", "Country TAG Traffic"))
```

```
#         digits=1,
#         out="./data_analysis_output/desc_cable_n_all_states.html",
#         covariate.labels=c("COW Codes of Countries", "Country TAG Traffic"))
```

6.3.8 FIGURE: Country TAG Traffic at Country-Year and Country Levels (All Countries)

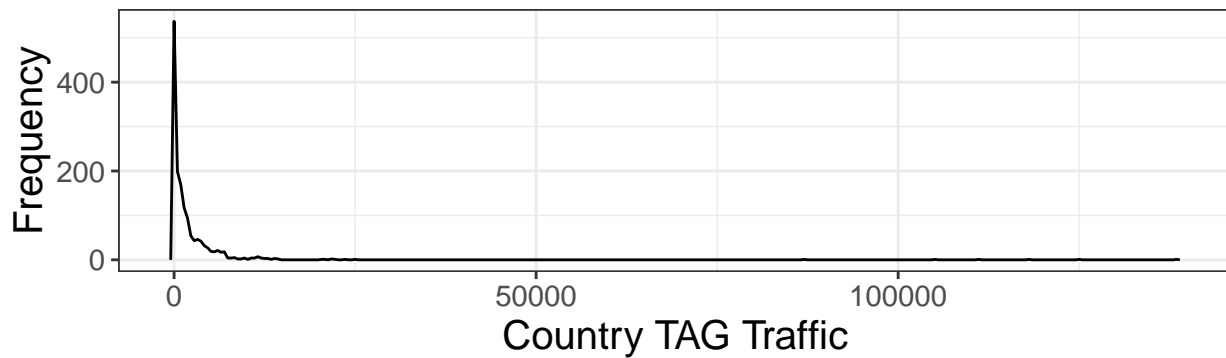
```
options(scipen=10000000)

p3<-
  ggplot(cable_n_all_states_year, aes(n_c_y)) +
# geom_histogram(bins = 300) +
  geom_freqpoly(bins = 300) +
  theme_bw() +
  labs(title = "By Country-Year",
        #subtitle = "Data Plotted by Year",
        y = "Frequency",
        x = "Country TAG Traffic") +
  theme(text = element_text(size=15),
        axis.text.x = element_text(size=11),
        axis.text.y = element_text(size=11)#,
        #legend.title=element_blank()#,
        #legend.position = c(0.1, 0.9),
        #legend.justification = c(0.1, 0.9)
        )

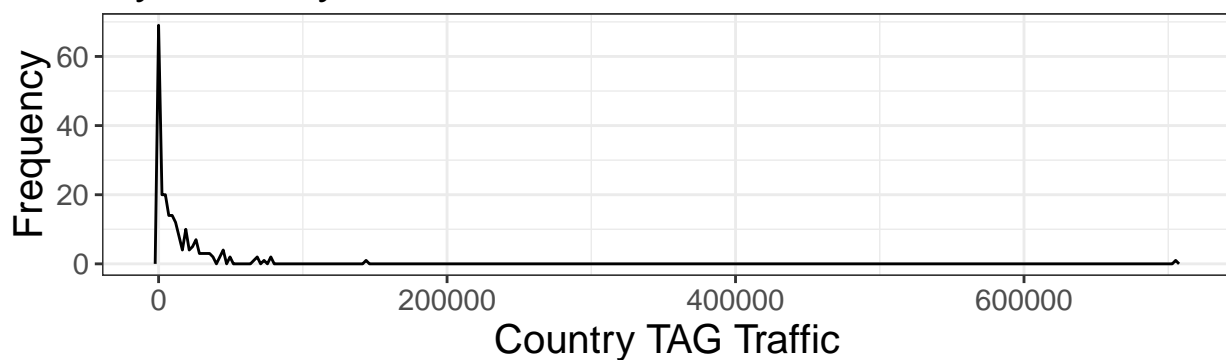
p4<-
  ggplot(cable_n_all_states, aes(n_c)) +
# geom_histogram(bins = 300) +
  geom_freqpoly(bins = 300) +
  theme_bw() +
  labs(title = "By Country",
        #subtitle = "Data Plotted by Year",
        y = "Frequency",
        x = "Country TAG Traffic") +
  theme(text = element_text(size=15),
        axis.text.x = element_text(size=11),
        axis.text.y = element_text(size=11)#,
        #legend.title=element_blank()#,
        #legend.position = c(0.1, 0.9),
        #legend.justification = c(0.1, 0.9)
        )

#png("./data_analysis_output/cable_n_all_states_year_freq.png")
grid.arrange(p3, p4)
```

By Country–Year



By Country



```
#dev.off()
```

6.3.9 TABLE: Country TAG Traffic vs. Cable Traffic

```
setwd("/Users/clarahsuong/chronos_data_intro")

russia_cable_traffic_1<-
  cables_docs %>%
  filter(str_detect(to_field, "MOSCOW") |
         str_detect(to_field, "LENINGRAD") |
         str_detect(from_field, "MOSCOW") |
         str_detect(from_field, "LENINGRAD")) %>%
  mutate(year=lubridate::year(date)) %>%
  group_by(year) %>%
  tally()

russia_cable_traffic_2<-
  cables_docs %>%
  filter(str_detect(to_field, "MOSCOW") |
         #str_detect(to_field, "LENINGRAD") |
         str_detect(from_field, "MOSCOW") #/
         #str_detect(from_field, "LENINGRAD")
         ) %>%
  mutate(year=lubridate::year(date)) %>%
  group_by(year) %>%
```

```

tally()

russia_cable_traffic_3<-
  cables_docs %>%
  filter(#str_detect(to_field, "MOSCOW") |
         str_detect(to_field, "LENINGRAD") |
         #str_detect(from_field, "MOSCOW") |
         str_detect(from_field, "LENINGRAD")
         )%>%
  mutate(year=lubridate::year(date)) %>%
  group_by(year) %>%
  tally()

russia_tag<-
  cable_n_states_70s_year %>%
  filter(cow_statename=="Russia")

russia_tag_cable_traffic<-cbind(russia_tag[c("year", "n_c_y")],
                                #russia_cable_traffic_1["n"],
                                russia_cable_traffic_2["n"],
                                russia_cable_traffic_3["n"]
                                )

#colnames(russia_tag_cable_traffic) <- c("Year", "Country TAG Traffic", "Cable Traffic")

stargazer(russia_tag_cable_traffic,
  type = "text",
  #flip = TRUE,
  summary = FALSE,
  rownames = FALSE,
  title="Comparison of Country TAG Traffic and Cable Traffic",
  digits=1,
  out="./data_analysis_output/russia_tag_cable_traffic.txt",
  covariate.labels=c("Year",
                      "Number of Cables Tagged<br>with the USSR",
                      #"Number of Cables Sent by/to<br>the US Embassy in Moscow<br>and the Consu
                      "Number of Cables Sent by/to<br>the US Embassy in Moscow",
                      "Number of Cables Sent by/to<br>the US Consulate General in Leningrad")
  )

##
## Comparison of Country TAG Traffic and Cable Traffic
## =====
## Year  Number of Cables Tagged<br>with the USSR Number of Cables Sent by/to<br>the US Embassy in Mosco
## -----
## 1,973          9,532                10,149
## 1,974         20,876                17,246
## 1,975         23,404                20,217
## 1,976         24,856                21,598
## 1,977         21,836                11,867
## 1,978         22,244                13,616
## 1,979         21,978                13,196
## -----

```

```
#stargazer(russia_tag_cable_traffic,
#           type = "html",
#           summary = FALSE,
#           rownames = FALSE,
#           title="Comparison of Country TAG Traffic and Cable Traffic",
#           digits=1,
#           out="./data_analysis_output/russia_tag_cable_traffic.html",
#           covariate.labels=c("Year",
#                               "Number of Cables Tagged with the USSR",
#                               "#Number of Cables Sent by/to<br>the US Embassy in Moscow<br>and the Cons",
#                               "Number of Cables Sent by/to the US Embassy in Moscow",
#                               "Number of Cables Sent by/to< the US Consulate General in Leningrad")
#           )
```

6.3.10 Country TAG Traffic for Certain Countries

```
## # A tibble: 2 x 2
##   year      n
##   <dbl> <int>
## 1 1973  3521
## 2 1974 10551

## # A tibble: 6 x 2
##   year      n
##   <dbl> <int>
## 1 1974  2028
## 2 1975  3054
## 3 1976  1907
## 4 1977  4148
## 5 1978  4830
## 6 1979  8384

## # A tibble: 1 x 2
##   year      n
##   <dbl> <int>
## 1 1978  3903
```

6.3.11 TABLE: Non-US Country-Years with Most Cables

```
setwd("/Users/clarahsuong/chronos_data_intro")

table_tag_state_year_top20<-
  cable_n_states_70s_year %>%
  filter(cow_ccode!=2) %>%
  mutate(rel.freq = paste0(round(100 * n_c_y/total_n, 2), "%")) %>%
  arrange(desc(n_c_y)) %>%
  top_n(n = 20, wt = n_c_y) %>%
  mutate(cow_statename= replace(cow_statename, cow_statename=="Russia", "Soviet Union")) #Replace "Russ

stargazer(table_tag_state_year_top20[c("year", "cow_statename", "n_c_y", "rel.freq")],
           summary = FALSE,
           rownames = FALSE,
```

```

type = "text",
title="Non-US Country-Years with Highest Tag Traffic",
digits=1,
out="./data_analysis_output/table_tag_state_year_top20.txt",
covariate.labels=c("Year", "Tagged Country", "Number of Cables", "Relative Frequency"))

```

```

##
## Non-US Country-Years with Highest Tag Traffic
## =====
## Year          Tagged Country      Number of Cables Relative Frequency
## -----
## 1976          Soviet Union         24856             0.74%
## 1975          Soviet Union         23404             0.7%
## 1978          Soviet Union         22244             0.66%
## 1979          Soviet Union         21978             0.66%
## 1977          Soviet Union         21836             0.65%
## 1974          Soviet Union         20876             0.62%
## 1979           Iran                 14433             0.43%
## 1977      United Kingdom           14145             0.42%
## 1979           Israel              13974             0.42%
## 1978           Israel              13918             0.42%
## 1976 German Democratic Republic    13775             0.41%
## 1977 German Democratic Republic    13606             0.41%
## 1976      United Kingdom           12885             0.38%
## 1979           Egypt              12764             0.38%
## 1978 German Democratic Republic    12733             0.38%
## 1978      United Kingdom           12630             0.38%
## 1979      United Kingdom           12605             0.38%
## 1975  Republic of Vietnam          12551             0.37%
## 1975 German Democratic Republic    12228             0.36%
## 1975           Japan              12087             0.36%
## -----

```

```

#stargazer(table_tag_state_year_top20[c("year", "cow_statename", "n_c_y", "rel.freq")],
#          summary = FALSE,
#          rownames = FALSE,
#          type = "html",
#          title="Non-US Country-Years with Highest Tag Traffic",
#          digits=1,
#          out="./data_analysis_output/table_tag_state_year_top20.html",
#          covariate.labels=c("Year", "Tagged Country", "Number of Cables", "Relative Frequency"))

```

6.3.12 TABLE: Non-US Country-Years Tagged in Fewest Cables

```

setwd("/Users/clarahsuong/chronos_data_intro")

table_tag_state_year_bottom20<-
  cable_n_states_70s_year %>%
  filter(cow_ccode!=2) %>%
  mutate(rel.freq = paste0(round(100 * n_c_y/total_n, 2), "%")) %>%
  arrange(desc(n_c_y)) %>%
  top_n(n = -20, wt = n_c_y) %>%
  mutate(cow_statename= replace(cow_statename, cow_statename=="Russia", "Soviet Union")) #Replace "Russ

```

```
stargazer(table_tag_state_year_bottom20[c("year", "cow_statename", "n_c_y", "rel.freq")],
  summary = FALSE,
  rownames = FALSE,
  type = "text",
  title="Non-US Country-Years with Lowest Tag Traffic",
  out="./data_analysis_output/table_tag_state_year_bottom20.txt",
  covariate.labels=c("Year", "Tagged Country", "Number of Cables", "Relative Frequency"))
```

```
##
## Non-US Country-Years with Lowest Tag Traffic
## =====
## Year      Tagged Country      Number of Cables Relative Frequency
## -----
## 1977      Mongolia            75                0%
## 1979      Maldives            75                0%
## 1978      Equatorial Guinea    72                0%
## 1979      Bhutan              68                0%
## 1977      Sao Tome and Principe 67                0%
## 1975      Mongolia            66                0%
## 1974      Bhutan              63                0%
## 1977      Equatorial Guinea    57                0%
## 1973      Albania             55                0%
## 1975      Maldives            55                0%
## 1978      Mongolia            50                0%
## 1979      Mongolia            48                0%
## 1973      Equatorial Guinea    45                0%
## 1973      Bhutan              35                0%
## 1975      Bhutan              31                0%
## 1977      Bhutan              31                0%
## 1976      Bhutan              28                0%
## 1973      Maldives            27                0%
## 1973      Congo              23                0%
## 1978      Bhutan              21                0%
## -----
```

```
#stargazer(table_tag_state_year_bottom20[c("year", "cow_statename", "n_c_y", "rel.freq")],
#  summary = FALSE,
#  rownames = FALSE,
#  type = "html",
#  title="Non-US Country-Years with Lowest Tag Traffic",
#  out="./data_analysis_output/table_tag_state_year_bottom20.html",
#  covariate.labels=c("Year", "Tagged Country", "Number of Cables", "Relative Frequency"))
```

6.3.13 TABLE: Countries Most Frequently Tagged in Cables

```
setwd("/Users/clarahsuong/chronos_data_intro")

table_tag_state_top20<-
  cable_n_states_70s %>%
  filter(cow_ccode!=2) %>%
  #group_by(cow_ccode, cow_stateabb, cow_statename) %>%
```



```

#summarise(n_c = sum(n_c)) %>%
#ungroup %>%
mutate(rel.freq = paste0(round(100 * n_c/total_n, 2), "%")) %>%
arrange(desc(n_c)) %>%
top_n(n = 20, wt = n_c) %>%
mutate(cow_statename= replace(cow_statename, cow_statename=="Russia", "Soviet Union")) #Replace "Russ

stargazer(table_tag_state_top20[c("cow_statename", "n_c","rel.freq")],
  summary = FALSE,
  rownames = FALSE,
  type = "text",
  title="Non-US Countries Most Frequently Tagged in Cables",
  out="./data_analysis_output/table_tag_state_top20.txt",
  covariate.labels=c("Country", "Number of Cables", "Relative Frequency"))

##
## Non-US Countries Most Frequently Tagged in Cables
## =====
## Country                Number of Cables Relative Frequency
## -----
## Soviet Union            144726                4.3%
## United Kingdom          78832                2.34%
## German Democratic Republic 78192                2.33%
## Japan                   73518                2.19%
## Israel                  68113                2.03%
## Egypt                   67582                2.01%
## France                  65907                1.96%
## Mexico                  48875                1.45%
## Canada                  48519                1.44%
## Iran                    45385                1.35%
## Italy                    44763                1.33%
## China                   43965                1.31%
## India                   43688                1.3%
## Thailand                42668                1.27%
## German Federal Republic 42379                1.26%
## South Korea             38899                1.16%
## Turkey                  38411                1.14%
## South Africa            35767                1.06%
## Philippines             35227                1.05%
## Poland                  35157                1.05%
## -----

#stargazer(table_tag_state_top20[c("cow_statename", "n_c","rel.freq")],
#  summary = FALSE,
#  rownames = FALSE,
#  type = "html",
#  title="Non-US Countries Most Frequently Tagged in Cables",
#  out="./data_analysis_output/table_tag_state_top20.html",
#  covariate.labels=c("Country", "Number of Cables", "Relative Frequency"))

```

6.3.14 TABLE: Non-U.S. Countries Least Frequently Tagged in Cables

```
setwd("/Users/clarahsuong/chronos_data_intro")

table_tag_state_bottom20<-
  cable_n_states_70s %>%
  filter(cow_ccode!=2) %>%
  mutate(rel.freq = paste0(round(100 * n_c/total_n, 0), "%")) %>%
  arrange(desc(n_c)) %>%
  top_n(n = -20, wt = n_c) %>%
  mutate(cow_statename= replace(cow_statename, cow_statename=="Russia", "Soviet Union")) #Replace "Russ

stargazer(table_tag_state_bottom20[c("cow_statename", "n_c","rel.freq")],
  summary = FALSE,
  rownames = FALSE,
  type = "text",
  title="Non-US Countries Least Frequently Tagged in Cables",
  digits=1,
  out="./data_analysis_output/table_tag_state_bottom20.txt",
  covariate.labels=c("Country", "Number of Cables", "Relative Frequency"))
```

```
##
## Non-US Countries Least Frequently Tagged in Cables
## =====
## Country                Number of Cables Relative Frequency
## -----
## Gambia                  2401                0%
## Congo                   2082                0%
## Seychelles              1897                0%
## Guinea-Bissau           1786                0%
## Yemen People's Republic 1772                0%
## Grenada                 1745                0%
## Albania                 1571                0%
## Cape Verde              1332                0%
## Djibouti                1188                0%
## Equatorial Guinea       950                0%
## Samoa                   665                0%
## Dominica                621                0%
## Maldives                577                0%
## Comoros                 577                0%
## Mongolia                553                0%
## Sao Tome and Principe   541                0%
## Solomon Islands         521                0%
## St. Lucia               496                0%
## St. Vincent and the Grenadines 354                0%
## Bhutan                  277                0%
## -----
```

```
#stargazer(table_tag_state_bottom20[c("cow_statename", "n_c","rel.freq")],
#          summary = FALSE,
#          rownames = FALSE,
#          type = "html",
#          title="Non-US Countries Least Frequently Tagged in Cables",
#          digits=1,
```

```
# out="./data_analysis_output/table_tag_state_bottom20.html",
# covariate.labels=c("Country", "Number of Cables", "Relative Frequency"))
```

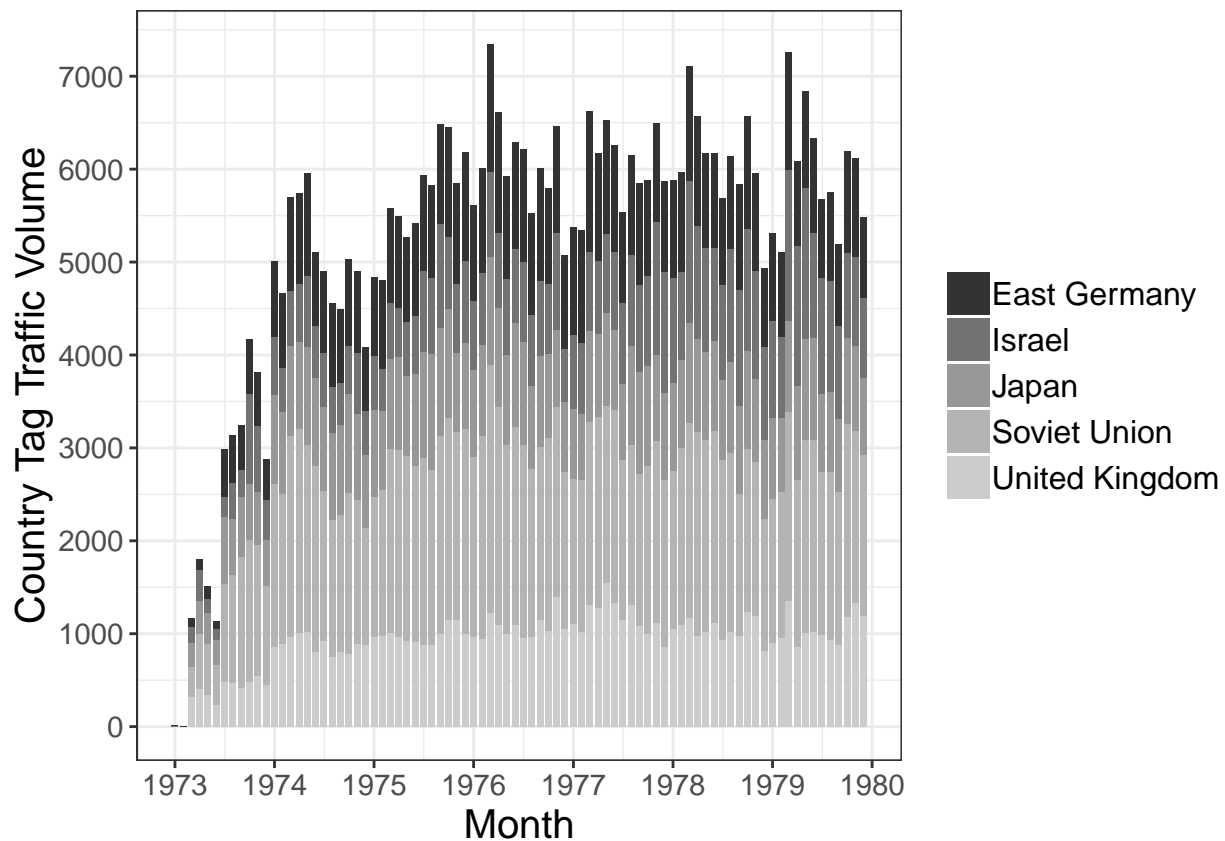
6.3.15 FIGURE: Country TAG Traffic of Key Non-US Countries by Month

```
setwd("/Users/clarahsuong/chronos_data_intro")

coi_5_list<-table_tag_state_top20$cow_ccode[1:5]

coi_5<-
  country_tag_doc2 %>%
  filter(cow_ccode %in% coi_5_list) %>%
  mutate(cow_statename= replace(cow_statename, cow_statename=="Russia", "Soviet Union"),
         cow_statename= replace(cow_statename, cow_statename=="German Democratic Republic", "East German"),
         #cow_statename2=ifelse(cow_ccode %in% coi_5_list, cow_statename, "Other"),
         date=as_date(date),
         month = as_date(cut(date, breaks = "month")))

#png("./data_analysis_output/tag_key5.png", width = 600, height = 450)
ggplot(coi_5, aes(month)) +
  geom_bar(aes(fill=cow_statename)) +
  scale_x_date(breaks=scales::pretty_breaks(10)) +
  scale_y_continuous(breaks=scales::pretty_breaks(10)) +
  labs(y = "Country Tag Traffic Volume",
       x = "Month") +
  scale_fill_grey() +
  theme_bw() +
  theme(text = element_text(size=15),
        axis.text.x = element_text(size=11),
        axis.text.y = element_text(size=11),
        legend.title=element_blank()#,
        #legend.position = c(0.95, 0.05),
        #legend.justification = c(0.95, 0.05)
        )
```



```
#dev.off()
```

6.3.16 TABLE: Country TAG Traffic vs. Total Population

```
setwd("/Users/clarahsuong/chronos_data_intro")

nmc_c_y<-
  read_csv("./external_data/NMC_5_0/NMC_5_0.csv") %>%
  dplyr::select("stateabb", "ccode", "year", "tpop")
```

```
## Parsed with column specification:
## cols(
##   stateabb = col_character(),
##   ccode = col_double(),
##   year = col_double(),
##   milex = col_double(),
##   milper = col_double(),
##   irst = col_double(),
##   pec = col_double(),
##   tpop = col_double(),
##   upop = col_double(),
##   cinc = col_double(),
##   version = col_double()
## )
```

```

pop_c <-
  read_csv("./external_data/NMC_5_0/NMC_5_0.csv") %>%
  dplyr::select("year", "ccode", "tpop") %>%
  filter(1972 < year & year < 1980 & ccode != 2) %>%
  left_join(states_70s_year, by = c("year" = "year", "ccode" = "cow_ccode")) %>%
  mutate(tpop = 1000 * tpop) %>%
  group_by(ccode, cow_statename) %>%
  summarise(mean_tpop = mean(tpop, na.rm = TRUE)) %>%
  ungroup() %>%
  arrange(desc(mean_tpop)) %>%
  mutate(mean_tpop_rank = row_number(),
         cow_statename = replace(cow_statename, cow_statename == "Russia", "Soviet Union")) #Replace "Russ

## Parsed with column specification:
## cols(
##   stateabb = col_character(),
##   ccode = col_double(),
##   year = col_double(),
##   milex = col_double(),
##   milper = col_double(),
##   irst = col_double(),
##   pec = col_double(),
##   tpop = col_double(),
##   upop = col_double(),
##   cinc = col_double(),
##   version = col_double()
## )

table_tag_state_top20 <-
  table_tag_state_top20 %>%
  mutate(tag_rank = row_number())

table_tag_pop_state_top20_comp <-
  table_tag_state_top20 %>%
  left_join(pop_c, by = "cow_statename") %>%
  dplyr::select("cow_statename", "tag_rank", "mean_tpop_rank")

stargazer(table_tag_pop_state_top20_comp,
           summary = FALSE,
           rownames = FALSE,
           type = "text",
           title = "Country TAG Traffic vs. Population",
           out = "./data_analysis_output/table_tag_pop_state_top20_comp.txt",
           covariate.labels = c("Top 20 Countries in Country TAG Traffic", "Rank in Country TAG Traffic",

##
## Country TAG Traffic vs. Population
## =====
## Top 20 Countries in Country TAG Traffic Rank in Country TAG Traffic Rank in Mean Population
## -----
## Soviet Union                                1                                3
## United Kingdom                             2                                12
## German Democratic Republic                 3                                35
## Japan                                      4                                 5
## Israel                                    5                                95

```

## Egypt	6	19
## France	7	14
## Mexico	8	11
## Canada	9	30
## Iran	10	23
## Italy	11	13
## China	12	1
## India	13	2
## Thailand	14	16
## German Federal Republic	15	10
## South Korea	16	21
## Turkey	17	17
## South Africa	18	27
## Philippines	19	15
## Poland	20	22
## -----		

```

#stargazer(table_tag_pop_state_top20_comp,
#          summary = FALSE,
#          rownames = FALSE,
#          type = "html",
#          title="Country TAG Traffic vs. Population",
#          out="./data_analysis_output/table_tag_pop_state_top20_comp.html",
#          covariate.labels=c("Top 20 Countries<br>in Country TAG Traffic", "Rank<br>in Country TAG Tra.

```