Coordenadas

Gabriel Petrini

June 7, 2021

Contents

Carregando pacotes e dados

Pacotes

Capitais

```
capitals <-maps::world.cities %>%
  filter(capital == "1") %>%
  select(country.etc, name) %>% # long and lat will be used in osm method
  rename(pais = country.etc, capital = name) %>%
  geocode(city = capital, country = pais, method = 'osm', lat = latitude, long = longitude) %>%
  mutate(ISO3C = countrycode::countrycode(pais, origin ='country.name',destination ='iso3c', warn = FALSE))

data.table::fwrite(capitals, file = "./clean/coord_capitais.csv")
```

Conflitos violentos

Tabela no texto (p. 366-7)

Importando dados

```
capitais <- data.table::fread(file = "./clean/coord_capitais.csv")

conflicts <- conflicts %>%
  rename(Year = V1, Country = V2, Proxy = V3, Continente = V4) %>%
  mutate(ISO3C = countrycode::countrycode(Country, origin ='country.name',destination ='iso3c', warn = FALSE)) %>%
  left_join(capitais) %>%
  mutate(pais = # Keep countries not found in dataset
           pais %>%
             is.na %>%
             ifelse(Country, pais) ) %>%
  mutate(ISO3C = replace_na(ISO3C, "404")) %>%
  mutate(Continente = factor(Continente))


## proxies <-conflicts %>% filter(ISO3C == "404") %>%
##   mutate(capital = case_when())
##   mutate(capital = Proxy) %>% # Keep countries not found in dataset
##   ## select(-c(latitude, longitude)) %>%
##   geocode(city = capital, method = 'osm', lat = latitude, long = longitude)

conflicts <- conflicts %>%
  mutate(capital = case_when(
         ISO3C == "404" ~ Proxy,
```

Table 1: Communist-led events

1945	Poland		Europa
1945	Austria		Europa
1946	North Korea		Asia
1946	China		Asia
1946	Greece		Europa
1947	Bulgaria		Europa
1948	Czechoslovakia	Prague	Europa
1948	East Germany	Bonn	Europa
1948	East Germany	Bonn	Europa
1948	China		Asia
1949	China		Asia
1950	South Korea		Asia
1950	China		Asia
1950	North and South Korea	South Korea	Asia
1953	East Germany	Bonn	Europa
1954	Vietinam		Asia
1954	Vietinam		Asia
1955	North and South Korea	South Korea	Asia
1956	Poland		Europa
1956	Hungary		Europa
1958	Taiwan		Asia
1959	Cuba		Americas
1960	East Germany	Bonn	Europa
1961	East Germany	Bonn	Europa
1962	Laos		Asia
1962	Cuba		Americas
1968	Vietinam		Asia
1968	Czechoslovakia	Prague	Europa
1969	North Korea		Asia
1969	Libya		Africa
1970	Cambodia		Asia
1975	Cambodia		Asia
1975	Vietinam		Asia
1975	Laos		Asia
1978	Afghanistan		Asia
1979	Nicaragua		Americas
1979	El Salvador		Americas
1979	Afghanistan		Asia
1981	Poland		Europa
1989	China		Asia

```
         TRUE ~ capital)) %>%
  mutate(ISO3C = case_when(
           pais == "East Germany" ~ "GERe",
           pais == "Czechoslovakia" ~ "oCZE",
           pais == "North and South Korea" ~ "PRK",
           TRUE ~ ISO3C # Não sei se faz isso, mas a ideia dessa linha é manter todo o restante como antes
         )) %>%
  mutate(pais = case_when( # Atualizando nome dos países para preencher coordenadas faltantes
           pais == "East Germany" ~ "Germany",
           pais == "Czechoslovakia" ~ "Czech Republic",
           pais == "North and South Korea" ~ "North Korea",
           TRUE ~ pais
         )) %>%
  select(-c(latitude, longitude)) %>%
  geocode(city = capital, country = pais, method = 'osm', lat = latitude, long = longitude)


conflicts <- conflicts %>%
  mutate(latitude  = case_when(
           ISO3C == "PRK" ~ 39.032, # https://pt.db-city.com/Coreia-do-Norte--Pyongyang
           TRUE ~ as.numeric(latitude)))
conflicts <- conflicts %>%
  mutate(longitude  = case_when(
           ISO3C == "PRK" ~ 125.75, # https://pt.db-city.com/Coreia-do-Norte--Pyongyang
           TRUE ~ as.numeric(longitude)))


conflicts <- conflicts %>%
  arrange(Year) %>%
  mutate(
    name = paste0(capital, "_", Year %>% as.character() %>% str_sub(start=-2), "_", Continente)
  ) %>%
  mutate(name = str_replace_all(name, " ", ""))


data.table::fwrite(conflicts, file = "./clean/coord_conflitos_violentos.csv")
```

Joining, by = “ISO3C”

Criando dataframe vazio

```
capitais <- data.table::fread(file = "./clean/coord_capitais.csv") %>% arrange(ISO3C)
conflitos <- data.table::fread(file = "./clean/coord_conflitos_violentos.csv") %>% arrange(ISO3C)

repeticoes <- conflitos %>% group_by(name) %>% group_size() # Equivalente ao Cr do artigo (vezes que ocorreu)

df <- matrix(
  nrow = capitais$longitude %>% length(),
  ncol = conflitos$name %>% unique() %>% length()
) %>%
  as.data.frame()


names(df) <- conflitos$name %>% unique()
df$ISO <- capitais$ISO3
df <- df[,c("ISO", conflitos$name)] %>%
  arrange(ISO)
df <- df %>% pivot_longer(!ISO,names_to = "Origem")
```

Função para calcular distância

```
distancia <- function(method = "geodesic", Destino, Origem) {
  geodist::geodist(
    x = capitais %>% filter(ISO3C == Destino) %>% select(latitude, longitude),
    y = conflitos %>% filter(name == Origem) %>% select(latitude, longitude),
    measure = method
    ## Nfrom = col_df %>% filter(name == matrix_col) %>% select(latitude) %>% as.numeric(), # latitude of origin
    ## Efrom = col_df %>% filter(name == matrix_col) %>% select(longitude) %>% as.numeric(), # latitude of origin
    ## Nto = row_df %>% filter(ISO3C == matrix_row) %>% select(latitude) %>% as.numeric(), # latitude of origin
    ## Eto = row_df %>% filter(ISO3C == matrix_row) %>% select(longitude) %>% as.numeric(), # latitude of origin
    ## units = units
  )[1]/1000 %>% as.numeric()
}
```

Aplicando função

```
## start <- sum(df$ISO == "") + 1
## for(i in start:nrow(df)){
##   for(j in 2:ncol(df)){
##     df[i,j] = distancia(
##       matrix_col = names(df)[j],
##       matrix_row = df$ISO[i]
##         )
##   }
## }

df <- data.table(df)
df[, Distancia := distancia(Origem = Origem, Destino = ISO), by = 1:nrow(df)]
df <- df %>% as.data.frame()
```

Exportando

```
df <- df %>%
  distinct() %>%
  pivot_wider(names_from = Origem, values_from = Distancia)

data.table::fwrite(df, file = "./raw/distancia_geodesica.csv")
```

Agrupando por regiões

```
df <- data.table::fread("./raw/distancia_geodesica.csv") %>% select(!c(value))
df <- df %>% # Remove colunas que contêm apenas NA
  select(
    where(
      ~!all(is.na(.x))
    )
  )

continentes <- c(
  "Africa",
  "Americas",
  "Asia",
  "Europa"
    )

for(continente in continentes){
  subset <- str_subset(names(df), continente)
  df <- df %>%
    drop_na(any_of(subset)) %>%
    mutate("Violentos_{continente}" := rowMeans(across(all_of(subset))))

}


subset <- str_subset(names(df), "Violentos")
```

```
df <- df %>% select(ISO, all_of(subset)) %>%
    mutate("Violentos" := rowMeans(across(all_of(subset))))
```

Exportando

```
data.table::fwrite(df, file = "./clean/conflitos_violentos_continentes.csv")
```

Conflitos não-violentos

Tabela no texto (p. 2 (apêndice))

Table 2: Communist-led events (non-violents)

1945	Czechoslovakia	Prague	Europa
1945	Bulgaria		Europa
1945	Yugoslavia	Belgrade	Europa
1945	Romania		Europa
1947	Poland		Europa
1947	Hungary		Europa
1947	Bulgaria		Europa
1947	Romania		Europa
1948	North Korea		Asia
1948	Hungary		Europa
1948	East Germany	Bonn	Europa
1949	USSR	Moscow	Europa
1949	North Korea		Asia
1949	East Germany	Bonn	Europa
1949	China		Asia
1952	East Germany	Bonn	Europa
1954	East Germany	Bonn	Europa
1960	Cuba		Americas
1960	Cuba		Americas
1970	Chile		Americas
1976	Vietinam		Asia
1977	Ethiopia		Africa

Importando dados

```
capitais <- data.table::fread(file = "./clean/coord_capitais.csv")

conflicts <- conflicts %>%
  rename(Year = V1, Country = V2, Proxy = V3, Continente = V4) %>%
  mutate(ISO3C = countrycode::countrycode(Country, origin ='country.name',destination ='iso3c', warn = FALSE)) %>%
  left_join(capitais) %>%
  mutate(pais = # Keep countries not found in dataset
           pais %>%
             is.na %>%
             ifelse(Country, pais) ) %>%
  mutate(ISO3C = replace_na(ISO3C, "404")) %>%
  mutate(Continente = factor(Continente))



conflicts <- conflicts %>%
  mutate(capital = case_when(
         ISO3C == "404" ~ Proxy,
         TRUE ~ capital)) %>%
  mutate(ISO3C = case_when(
           pais == "East Germany" ~ "GERe",
           pais == "Czechoslovakia" ~ "oCZE",
           pais == "North and South Korea" ~ "PRK",
           pais == "Yugoslavia" ~ "YUG",
           pais == "USSR" ~ "USSR",
           TRUE ~ ISO3C # Não sei se faz isso, mas a ideia dessa linha é manter todo o restante como antes
         )) %>%
  mutate(pais = case_when( # Atualizando nome dos países para preencher coordenadas faltantes
           pais == "East Germany" ~ "Germany",
           pais == "Czechoslovakia" ~ "Czech Republic",
           pais == "North and South Korea" ~ "North Korea",
           pais == "Yugoslavia" ~ "Serbia",
           pais == "USSR" ~ "Russia",
           TRUE ~ pais
         )) %>%
  select(-c(latitude, longitude)) %>%
  geocode(city = capital, country = pais, method = 'osm', lat = latitude, long = longitude)


conflicts <- conflicts %>%
  mutate(latitude  = case_when(
           ISO3C == "PRK" ~ 39.032, # https://pt.db-city.com/Coreia-do-Norte--Pyongyang
           TRUE ~ as.numeric(latitude)))
conflicts <- conflicts %>%
  mutate(longitude  = case_when(
           ISO3C == "PRK" ~ 125.75, # https://pt.db-city.com/Coreia-do-Norte--Pyongyang
           TRUE ~ as.numeric(longitude)))


conflicts <- conflicts %>%
  arrange(Year) %>%
  mutate(
    name = paste0(capital, "_", Year %>% as.character() %>% str_sub(start=-2), "_", Continente)
  ) %>%
  mutate(name = str_replace_all(name, " ", ""))


data.table::fwrite(conflicts, file = "./clean/coord_conflitos_nao_violentos.csv")
```

Joining, by = "ISO3C"

Criando dataframe vazio

```
capitais <- data.table::fread(file = "./clean/coord_capitais.csv") %>% arrange(ISO3C)
conflitos <- data.table::fread(file = "./clean/coord_conflitos_nao_violentos.csv") %>% arrange(ISO3C)

repeticoes <- conflitos %>% group_by(name) %>% group_size() # Equivalente ao Cr do artigo (vezes que ocorreu)

df <- matrix(
  nrow = capitais$longitude %>% length(),
  ncol = conflitos$name %>% unique() %>% length()
) %>%
  as.data.frame()


names(df) <- conflitos$name %>% unique()
```

```
df$ISO <- capitais$ISO3
df <- df[,c("ISO", conflitos$name)] %>%
  arrange(ISO)
df <- df %>% pivot_longer(!ISO,names_to = "Origem")
```

Aplicando função

```
df <- data.table(df)
df[, Distancia := distancia(Origem = Origem, Destino = ISO), by = 1:nrow(df)]
df <- df %>% as.data.frame()
```

Exportando

```
df <- df %>%
  distinct() %>%
  pivot_wider(names_from = Origem, values_from = Distancia)

data.table::fwrite(df, file = "./raw/distancia_geodesica_naoviolentos.csv")
```

Agrupando por regiões

```
df <- data.table::fread("./raw/distancia_geodesica_naoviolentos.csv") %>% select(!c(value))
df <- df %>% # Remove colunas que contêm apenas NA
  select(
    where(
      ~!all(is.na(.x))
    )
  )

continentes <- c(
  "Africa",
  "Americas",
  "Asia",
  "Europa"
    )

for(continente in continentes){
  subset <- str_subset(names(df), continente)
  df <- df %>%
    drop_na(any_of(subset)) %>%
    mutate("NaoViolentos_{continente}" := rowMeans(across(all_of(subset))))

}


subset <- str_subset(names(df), "Violentos")
df <- df %>% select(ISO, all_of(subset)) %>%
    mutate("NaoViolentos" := rowMeans(across(all_of(subset))))
```

Exportando

```
data.table::fwrite(df, file = "./clean/conflitos_naoviolentos_continentes.csv")
```

Selecionando países

Importando e combinando dataframes

```r
data.table::fread("./clean/conflitos_violentos_continentes.csv") -> violentos
data.table::fread("./clean/conflitos_naoviolentos_continentes.csv") -> naoviolentos

df <- violentos %>%
  arrange(ISO) %>%
  left_join(naoviolentos) %>%
  arrange(ISO)
```

Joining, by = "ISO"

Filtrando países (Tabela A.1)

```r
paises <- c(
  "Australia",
  "Canada",
  "Denmark",
  "Finland",
  "France",
  "Germany",
  "Ireland",
  "Italy",
  "Japan",
  "Netherlands",
  "New Zealand",
  "Norway",
  "Portugal",
  "Spain",
  "Sweden",
  "Switzerland",
  "UK",
  "USA" # Ausente no original
)


ISOs <- data.table::fread(file = "./clean/coord_capitais.csv") %>%
  filter(pais %in% paises) %>%
  select(ISO3C) %>%
  rename(ISO = ISO3C)

df <- df %>%
  filter(ISO %in% ISOs$ISO %>% c())
```

Exportando

```r
data.table::fwrite(df, file = "./clean/conflitos_filtrados_n_violentos_continentes.csv")
```

Calculando inverso da distância

$$ColdwarEvent = 1000 \cdot \sum W_j \cdot CR_j \tag{1}$$

$$W_j = dist^{-2} \tag{2}$$

em que dist é medido em quilometros.

```r
data.table::fread("./clean/conflitos_filtrados_n_violentos_continentes.csv") -> df

inv_dist2 <- function(x, pow=-1){
  x^pow*10^3
}

df <- df %>%
  mutate(across(where(is.numeric), ~inv_dist2(.x)))

data.table::fwrite(df, file = "./clean/coldwar_inv2_n_violentos_selecionados.csv")
```

Incluindo dados de distribuição

Script Lorena

```r
one_percent_income <- read.csv2("./raw/WID_Data_11042021-185939.csv")

# subseting
year <- one_percent_income[, 1, drop = FALSE]
income <- one_percent_income[, 2:19]

# cleaning data
# gsub() or nchar() as additional options
colnames(income) <- substring(colnames(income), 52)

# transforming in ISO3
colnames(income) <- countrycode(colnames(income), origin = "country.name", destination = "iso3c")

# joining subsets again
income_concentration <- cbind(year, income)

# exporting to csv
write.csv(income_concentration, "./clean/income_concentration.csv")
```

Renomeando países para ISO

```r
inicio <- "1945"
fim <- "1989"

df <- data.table::fread(
                    './clean/income_concentration.csv',
                    header=TRUE
                    ) %>%
  select(-c(V1)) %>%
  mutate(Year = lubridate::ymd(Year, truncated = 2L)) %>%
  rename(UK = United.Kingdom, `New Zealand` = New.Zealand) %>%
  pivot_longer(!Year, names_to = "pais", values_to = "concentracao") %>%
  mutate(ISO = countrycode::countrycode(pais, origin ='country.name',destination ='iso3c', warn = FALSE)) %>%
  select(-c(pais)) %>%
  pivot_wider(names_from = ISO, values_from = concentracao) %>% # Preenchendo NAs para selecionar os primeiros
  fill(everything(), .direction = "updown") %>% # Transpondo para criar primeiro e último valor
  pivot_longer(-Year) %>%
  pivot_wider(names_from = Year, values_from = value) %>%
  select(name, starts_with(fim), starts_with(inicio)) %>%
  setNames(c("ISO", "Fim", "Inicio")) %>%
  mutate(Diff = Fim - Inicio)

data.table::fwrite(df, paste0("./clean/diff_concentracao_", fim, "_", inicio, ".csv"))
```

Unindo com distâncias

```
data.table::fread(paste0("./clean/diff_concentracao_", fim, "_", inicio, ".csv")) -> concentracao
data.table::fread("./clean/coldwar_inv2_n_violentos_selecionados.csv") -> coldwar

df <- coldwar %>%
  left_join(concentracao)

data.table::fwrite(df, paste0("./clean/coldwar_concentracao_", fim, "_", inicio, ".csv"))
```

Joining, by = "ISO"

Unindo com dados do artigo de referência

Dicionário de variáveis

Variavel	Descricao	Uso
year	Year	Input
code	ISO code	Explicativa
cold_war_events	Cold War Events	Consulta
pc	Communist Party Share of seats	Explicativa
wage_bargaining_centralization	Wage bargaining centralization	Alternativa
decentralized	Decentralized Bargain	Explicativa
union	Union Density	Explicativa
Left_Executive	Left Executive	Alternativa
polity2	Polity IV index (Polity2 index?)	Explicativa
war_risk	War risk	Alternativa
cultural_distance_cw	Cultural Distance	Alternativa
strikes	Labor Strikes (?)	Alternativa
top_1	Top 1% of income	Consulta
top_5	Top 5% of income	Resultado
top_10	Top 10% of income	Resultado
gini	Gini	Resultado
sstran	Social security transfers (% GDP)	Resultado
yr_sch	Years of Schooling	Resultado

Importando dados

```
names(dicionario) <- as.matrix(dicionario[1, ])
dicionario <- dicionario[-1, ]
dicionario[] <- lapply(dicionario, function(x) type.convert(as.character(x)))

haven::read_dta('./raw/base_JCP_vR1_dez18.dta') -> df

df %>%
    data.table::fwrite(paste0('./clean/santanna_weller_full.csv'))

df <- df %>%
  filter(year >= 1945) %>%
  filter(year <= 1990) %>%
  group_by(code) %>%
  summarise(across(where(is.numeric), ~mean(.x, rm.na = TRUE))) %>%
```

```r
  ungroup() %>%
  select(all_of(dicionario$Variavel)) %>%
  select(-c(year))

df %>%
    data.table::fwrite(paste0('./clean/santanna_weller_subset_media.csv'))

flags <- dicionario %>% select(Uso) %>% filter(Uso != "Input") %>% unique() %>% pull() %>% as.character()

exporter <- function(df, flag){
  dicionario %>% filter(Uso == flag) %>% select(Variavel) %>% pull() %>% as.character() -> subset
  subset <- append("code", subset)

  df %>%
    select(all_of(subset)) %>%
    data.table::fwrite(paste0('./clean/santanna_weller_', flag, '_media.csv'))
}

for(flag in flags){exporter(df = df, flag = flag)}
```

Unindo as bases

Importante: A base de dados de Santanna e Weller que será mesclada é a que esta no arquivo `santanna_weller_Explicativa_medias.csv` em que estão os dados médios.

```r
data.table::fread(paste0("./clean/coldwar_concentracao_", fim, "_", inicio, ".csv")) -> df
data.table::fread("./clean/santanna_weller_Explicativa_media.csv") -> santanna_weller
santanna_weller <- santanna_weller %>%
  rename(ISO = code)

df <- df %>%
  left_join(santanna_weller)

data.table::fwrite(df, './clean/coldwar_concentracao_santanna_weller.csv')
```

Calibragem

Calibrando conflitos

Os conflitos serão calibrados de acordo com o método indireto. Para evitar a sensibilidade da amostra, transpõe-se o dataframe. Desse modo, as âncoras qualitativas serão associadas aos países e não as continentes que os conflitos ocorreram.

```r
data.table::fread('./clean/coldwar_concentracao_santanna_weller.csv') -> df

tmp <- df %>%
  select(
    ISO,
    Violentos_Europa, NaoViolentos_Europa,
    Violentos_Asia, NaoViolentos_Asia,
    Violentos_Africa, NaoViolentos_Africa,
    Violentos_Americas, NaoViolentos_Americas,
    Violentos, NaoViolentos
  ) %>%
  rename_with(.cols = -ISO, .fn = ~ paste0(.x, "_cal")) %>%
   pivot_longer(-ISO) %>%
   pivot_wider(names_from=ISO, values_from=value)

for(country in names(tmp)[-1]){
  new_name <- paste0(country, "_cal")
  x <- tmp[country] %>% as.vector() %>% t()
  # Find quantiles
  quant <- quantile(x, probs = seq(0, 1, 0.2))

  # Theoretical calibration
  x_cal <- NA
  x_cal[x <= quant[1]] <- 0
  x_cal[x > quant[1] & x <= quant[2]] <- .2
  x_cal[x > quant[2] & x <= quant[3]] <- .4
  x_cal[x > quant[3] & x <= quant[4]] <- .6
  x_cal[x > quant[4] & x <= quant[5]] <- .8
  x_cal[x > quant[5]] <- 1

  tmp[country] <- indirectCalibration(x %>% as.vector(), x_cal, binom = TRUE)
}

tmp <- tmp %>%
  ## select(name, ends_with("_cal")) %>%
  rename(Conflito = name) %>%
  pivot_longer(-Conflito) %>%
  pivot_wider(names_from=Conflito, values_from=value) %>%
  rename(ISO = name)

df <- df %>%
  left_join(tmp)

data.table::fwrite(df, './clean/full_distancias_calibradas.csv')
```

Calibrando outras variáveis

Tal como para os conflitos, será utilizado o método indireto. Diferentemente dos conflitos, o cálculo será feito entre países.

```r
data.table::fread('./clean/full_distancias_calibradas.csv') -> df

tmp <- df %>%
  select(
    ISO,
    Diff,
    pc,
    decentralized
  ) %>%
  as.data.frame()

for(variable in names(tmp)[-1]){
  x <- tmp[variable] %>% as.vector() %>% t()
  # Find quantiles
  quant <- quantile(x, probs = seq(0, 1, 0.2))

  # Theoretical calibration
  x_cal <- NA
  x_cal[x <= quant[1]] <- 0
  x_cal[x > quant[1] & x <= quant[2]] <- .2
  x_cal[x > quant[2] & x <= quant[3]] <- .4
  x_cal[x > quant[3] & x <= quant[4]] <- .6
  x_cal[x > quant[4] & x <= quant[5]] <- .8
  x_cal[x > quant[5]] <- 1

  tmp[variable] <- indirectCalibration(x %>% as.vector(), x_cal, binom = TRUE)
}

tmp <- tmp %>%
  rename_with(.cols = -ISO, .fn = ~ paste0(.x, "_cal"))

df <- df %>%
  left_join(tmp)

data.table::fwrite(df, './clean/raw_calibrados.csv')
data.table::fwrite(
              df %>% select(ISO,  ends_with("_cal")),
              './clean/calibrados.csv')
```

Tabela verdade

Presença do resultado (menor concentração, Diff decrescente)

```
df <- data.table::fread('./clean/calibrados.csv') %>% as.data.frame()

df <- df %>%
  set_names(
    ~ str_to_upper(.) %>%
                str_replace_all("_CAL", "") %>%
                str_replace_all("VIOLENTOS", "V") %>%
                str_replace_all("AFRICA", "AFR") %>%
                str_replace_all("AMERICAS", "AME") %>%
                str_replace_all("EUROPA", "EUR") %>%
                str_replace_all("NAO", "N")
  ) %>%
  rename(DEC = DECENTRALIZED)

conds <- df %>%  select(-c(DIFF, V, NV, ISO)) %>%
  names() %>%
  as.vector()

TT <- truthTable(
  data = df,
  outcome = "~DIFF",
  conditions = conds,
  complete = TRUE,
  show.cases = TRUE,
  incl.cut = 0.7,
  sort.by = "out, incl"
)

stargazerTT(TT, show.cases = TRUE, type = "text")
```

```
    ========================================================================================================================
V_EUR NV_EUR V_ASIA NV_ASIA V_AFR NV_AFR V_AME NV_AME PC DEC OUT n
incl PRI cases
```

```
221 0 0 1 1 0 1 1 1 0 0 1 1 0.916 0 15 214 0 0 1 1 0 1 0 1 0 1 1 1 0.728 0 1 803 1 1 0 0 1 0 0 0 1 0 1
2 0.718 0.471 11,13 819 1 1 0 0 1 1 0 0 1 0 1 3 0.706 0.459 5,16,17 817 1 1 0 0 1 1 0 0 0 0 0 2 0.682
0.446 6,7 801 1 1 0 0 1 0 0 0 0 0 0 2 0.681 0.436 3,14 818 1 1 0 0 1 1 0 0 0 1 0 1 0.524 0.174 4 802 1
1 0 0 1 0 0 0 0 1 0 2 0.520 0.197 9,10 804 1 1 0 0 1 0 0 0 1 1 0 1 0.499 0 8 814 1 1 0 0 1 0 1 1 0 1 0
2 0.421 0 2,18 964 1 1 1 1 0 0 0 0 1 1 0 1 0.382 0 12
```

Ausência do resultado (maior concentração, Diff crescente)

Minimização

TODOs

TODO Conferir NAs da base

- □ Como preencher se forem faltantes

TODO Decidir se as medidas de centralidade são sempre as médias