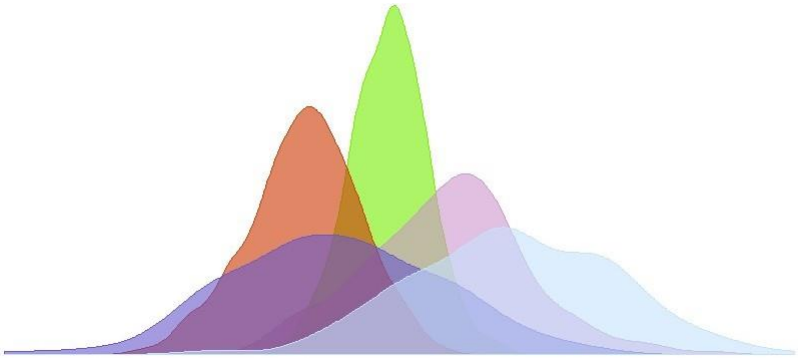
uc3m | Universidad Carlos III de Madrid

Trabajo individual sobre el Onu y los Países

Métodos estadísticos en minería de datos



Prof: **ALBERTO MUÑOZ GARCIA**

Enero 2020

FRANCESCO CARBONERA

100390180

Resumen de las páginas

Para este trabajo utilizo el conjunto de datos **un_vote** contenido en el package **unvotes**. Este conjunto de datos contiene información sobre la historia de los votos de los países de la ONU para diferentes temas. Las variables son:

rcid sirve como una clave para hacer "join" con otros conjuntos de datos en el package, como **un_roll_calls** y **un_roll_call_issues**;

country contiene el nombre en inglés del país;

country_code contiene el código ISO de 2 caracteres que representa el país;

vote es una variable factorial con el resultado del voto: yes / abstein / no.

Para este trabajo usaré muchas de las funciones contenidas en la biblioteca **dplyr** y para esto explicaré los diversos usos y utilidades en el análisis de un conjunto de datos.

```
library(unvotes)
library(dplyr)
```

En la primera parte explicaré y mostraré los resultados de los comandos solicitados en la entrega, en la segunda, con más libertad, llevaré a cabo análisis exploratorios y de cluster entre los diferentes países y los diferentes temas de votación.

Parte I

El primer comando que vemos es el "pipe operator", que es **%>%**, y se usa para simplificar el código cuando hay muchos paréntesis y esto hace que el código sea difícil de leer y entender. Por ejemplo, tenemos dos funciones $f: B \to C$ y $g: A \to B$, si quieremos hacer $f(g(x))$, g es el imput de la funcion f y x es el imput de la funcion g. De echo, puedo escribir esto como $f \circ g \to g\%>\%f$. Tomemos este ejemplo:

```
x=c(1,2,3,4)
g=function(d) d*2
f=function(d) (d^2)/2
```

Queremos encontrar el resultado de $f \circ g$ por los valores de x. Hay dos maneras: la segunda es la mas simple.

```
f(g(x))
```

```
## [1]  2  8 18 32
```

```
x %>% g %>% f
```

```
## [1]  2  8 18 32
```

Vuelvo al conjunto de datos inicial. De esta manera, con la funcion **count()**, se puede contar el número total de votos realizados por todas las naciones.Esto también es visible utilizando el comando **dim(un_votes)**.

```
vt<-un_votes %>% count
vt
```

```
## # A tibble: 1 x 1
##        n
##    <int>
## 1 738764
```

Ahora uso la función **filter()** que le permite seleccionar las observaciones de acuerdo con algunos criterios. Esto evita usar la función **subset()** y crear un nuevo subconjunto. En este caso, con esta función puede ver qué estados han votado sí (para los diferentes temas de votación), puede contar el número de veces que ha votado positivamente y calcular la frecuencia respecto al número total de votos.

```
un_votes %>% filter(vote=="yes")
```

```
## # A tibble: 588,800 x 4
##     rcid country                  country_code vote
##    <int> <chr>                    <chr>        <fct>
##  1     3 United States of America US           yes
##  2     3 Cuba                     CU           yes
##  3     3 Haiti                    HT           yes
##  4     3 Dominican Republic       DO           yes
##  5     3 Mexico                   MX           yes
##  6     3 Guatemala                GT           yes
##  7     3 Honduras                 HN           yes
##  8     3 El Salvador              SV           yes
##  9     3 Nicaragua                NI           yes
## 10     3 Costa Rica               CR           yes
## # … with 588,790 more rows
```

```
vy<-un_votes %>% filter(vote=="yes") %>% count
vy
```

```
## # A tibble: 1 x 1
##        n
##    <int>
## 1 588800
```

Se pueden usar dos comandos diferentes para calcular la frecuencia de los votos yes en todos los votos totales.

```
vy/vt
```

```
##           n
## 1 0.7970069
```

```
un_votes %>% summarize(total = n(),percent_yes = mean(vote =="yes"))
```

```
## # A tibble: 1 x 2
##    total percent_yes
##    <int>       <dbl>
## 1 738764       0.797
```

Con la función **group_by()**, calculo el porcentaje de votos positivos para cada país. En este caso la lista de países es ordenada alfabéticamente.

```
by_country <- un_votes %>%
  group_by(country) %>%
  summarize(votes = n(),
            percent_yes = mean(vote == "yes"))
by_country
```

```
## # A tibble: 200 x 3
##    country             votes percent_yes
##    <chr>               <int>       <dbl>
##  1 Afghanistan          4972       0.842
##  2 Albania              3514       0.716
##  3 Algeria              4527       0.898
##  4 Andorra              1564       0.645
##  5 Angola               3075       0.922
##  6 Antigua and Barbuda  2658       0.919
##  7 Argentina            5361       0.779
##  8 Armenia              1629       0.759
##  9 Australia            5399       0.552
## 10 Austria              4939       0.633
## # … with 190 more rows
```

Con la función **arrange()** podemos ordenar en orden ascendente o descendente (con **desc()**) en función del porcentaje de voto positivo.

```
arrange(by_country, percent_yes)
```

```
## # A tibble: 200 x 3
##    country                                                votes percent_yes
##    <chr>                                                  <int>       <dbl>
##  1 Zanzibar                                                   2       0
##  2 United States of America                                5390       0.284
##  3 Palau                                                    896       0.323
##  4 Israel                                                  4944       0.346
##  5 Federal Republic of Germany                             2067       0.396
##  6 Micronesia (Federated States of)                        1462       0.414
##  7 United Kingdom of Great Britain and Northern Ireland    5372       0.429
##  8 France                                                  5325       0.434
##  9 Marshall Islands                                        1600       0.489
## 10 Belgium                                                 5391       0.495
## # … with 190 more rows
```

```
arrange(by_country, desc(percent_yes))
```

```
## # A tibble: 200 x 3
##    country               votes percent_yes
##    <chr>                 <int>       <dbl>
##  1 Seychelles             1790       0.978
##  2 Timor-Leste             837       0.970
##  3 Sao Tome and Principe  2389       0.967
```

```
##  4 Cabo Verde             3292       0.960
##  5 Djibouti               3345       0.956
##  6 Guinea Bissau          3070       0.956
##  7 Comoros                2530       0.945
##  8 Mozambique             3456       0.943
##  9 United Arab Emirates   4031       0.941
## 10 Suriname               3410       0.941
## # … with 190 more rows
```

Con el comando **inner_join()**, combinamos dos conjuntos de datos basados en las palabras clave contenidas en **rcid**, un atributo presente en los dataset **un_votes**, **un_roll_calls** y **un_roll_calls_issues**. De esta forma, para cada voto (contenido en **un_votes**) tendremos en la tabla **votes_per** la información contenida en el conjunto de datos **un_roll_calls** (es decir, información sobre la fecha de la votación, el código de resolución y una descripción del tema de votación). La segunda tabla contiene la información de **un_votes** combinada con el dataset **un_roll_call_issues**, que contiene el código del tema de la votación y la descripción principal.

```
votes_per <- un_votes %>%
  inner_join(un_roll_calls, by = "rcid")
votes_per
```

```
## # A tibble: 738,764 x 12
##     rcid country country_code vote  session importantvote date       unres
amend
##    <int> <chr>   <chr>        <fct>   <dbl>         <dbl> <date>     <chr>
<dbl>
##  1     3 United… US           yes         1             0 1946-01-01 R/1/…
1
##  2     3 Canada  CA           no          1             0 1946-01-01 R/1/…
1
##  3     3 Cuba    CU           yes         1             0 1946-01-01 R/1/…
1
##  4     3 Haiti   HT           yes         1             0 1946-01-01 R/1/…
1
##  5     3 Domini… DO           yes         1             0 1946-01-01 R/1/…
1
##  6     3 Mexico  MX           yes         1             0 1946-01-01 R/1/…
1
##  7     3 Guatem… GT           yes         1             0 1946-01-01 R/1/…
1
##  8     3 Hondur… HN           yes         1             0 1946-01-01 R/1/…
1
##  9     3 El Sal… SV           yes         1             0 1946-01-01 R/1/…
1
## 10     3 Nicara… NI           yes         1             0 1946-01-01 R/1/…
1
## # … with 738,754 more rows, and 3 more variables: para <dbl>, short <chr>,
## #   descr <chr>
```

```
votes_issues <- un_votes %>%
  inner_join(un_roll_call_issues, by = "rcid")
votes_issues
```

```
## # A tibble: 768,674 x 6
##     rcid country                  country_code vote    short_name issue
##    <int> <chr>                    <chr>        <fct>   <chr>      <chr>
##  1     6 United States of America US           no      hr         Human rights
##  2     6 Canada                   CA           no      hr         Human rights
##  3     6 Cuba                     CU           yes     hr         Human rights
##  4     6 Dominican Republic       DO           abstain hr         Human rights
##  5     6 Mexico                   MX           yes     hr         Human rights
##  6     6 Guatemala                GT           no      hr         Human rights
##  7     6 Honduras                 HN           yes     hr         Human rights
##  8     6 El Salvador              SV           abstain hr         Human rights
##  9     6 Nicaragua                NI           yes     hr         Human rights
## 10     6 Panama                   PA           abstain hr         Human rights
## # … with 768,664 more rows
```

Creo una tabla ordenada en orden creciente de la frecuencia de los votos positivos de los países para las votaciónes con el tema colonialism.

```
cosa <- votes_issues %>%
  filter(issue == "Colonialism") %>%
  group_by(country) %>%
  summarize(percent_yes = mean(vote=="yes" )) %>%
  arrange(percent_yes)
cosa
```

```
## # A tibble: 199 x 2
##    country                                                percent_yes
##    <chr>                                                        <dbl>
##  1 United States of America                                     0.166
##  2 Micronesia (Federated States of)                             0.194
##  3 Israel                                                       0.233
##  4 Palau                                                        0.26
##  5 Federal Republic of Germany                                  0.268
##  6 France                                                       0.281
##  7 United Kingdom of Great Britain and Northern Ireland         0.283
##  8 Nauru                                                        0.374
##  9 Belgium                                                      0.388
## 10 Marshall Islands                                             0.396
## # … with 189 more rows
```

Cuento el total de votos para temas individuales y luego, para cada tema, el número de votos realizados por los estados individuales.

```
by_issues <- group_by(votes_issues,issue)
summarize(by_issues,count=n())
```

```
## # A tibble: 6 x 2
##   issue                                count
```

```
##   <chr>                                <int>
## 1 Arms control and disarmament        146581
## 2 Colonialism                         127027
## 3 Economic development                 68828
## 4 Human rights                        146441
## 5 Nuclear weapons and nuclear material 115266
## 6 Palestinian conflict                164531
```

```
by_issues <- group_by(votes_issues,country,issue)
summarize(by_issues,count=n())
```

```
## # A tibble: 1,194 x 3
## # Groups:   country [199]
##    country     issue                                count
##    <chr>       <chr>                                <int>
##  1 Afghanistan Arms control and disarmament           897
##  2 Afghanistan Colonialism                            898
##  3 Afghanistan Economic development                   440
##  4 Afghanistan Human rights                           904
##  5 Afghanistan Nuclear weapons and nuclear material   711
##  6 Afghanistan Palestinian conflict                   994
##  7 Albania     Arms control and disarmament           620
##  8 Albania     Colonialism                            717
##  9 Albania     Economic development                   306
## 10 Albania     Human rights                           735
## # … with 1,184 more rows
```

Con la función **separate()** del paquete **tidyr**, separo todos los votos por día, mes y año.

```
library(tidyr)

votes_per2 <- votes_per %>% separate(date,into=c("year","month","day"))
```

Se cuenta el numero de votos y la frecuencia de los votos positivos para cada para cada año y posteriormente para cada pais.

```
by_year <- votes_per2 %>%
  group_by(year) %>%
  summarize(total = n(),
            percent_yes = mean(vote == "yes"))
by_year
```

```
## # A tibble: 69 x 3
##    year  total percent_yes
##    <chr> <int>       <dbl>
##  1 1946   2143       0.573
##  2 1947   2039       0.569
##  3 1948   3454       0.400
##  4 1949   5700       0.425
##  5 1950   2911       0.497
##  6 1951    402       0.657
##  7 1952   4082       0.546
```

```
##  8 1953   1537       0.632
##  9 1954   1788       0.622
## 10 1955   2169       0.695
## # … with 59 more rows
```

```
by_country <- votes_per2 %>%
  group_by(country) %>%
  summarize(total = n(),
            percent_yes = mean(vote == "yes"))
by_country
```

```
## # A tibble: 200 x 3
##    country             total percent_yes
##    <chr>               <int>       <dbl>
##  1 Afghanistan          4972       0.842
##  2 Albania              3514       0.716
##  3 Algeria              4527       0.898
##  4 Andorra              1564       0.645
##  5 Angola               3075       0.922
##  6 Antigua and Barbuda  2658       0.919
##  7 Argentina            5361       0.779
##  8 Armenia              1629       0.759
##  9 Australia            5399       0.552
## 10 Austria              4939       0.633
## # … with 190 more rows
```

Elimino de la tabla **by_country** los países que han votado menos de 100 veces. Yo uso el comando de filtro. Se observa que solo se elimina un país: Zanzíbar.

```
by_countrymod <-by_country %>%
  arrange(percent_yes) %>%
  filter(total > 100)
length(by_country$country)
```

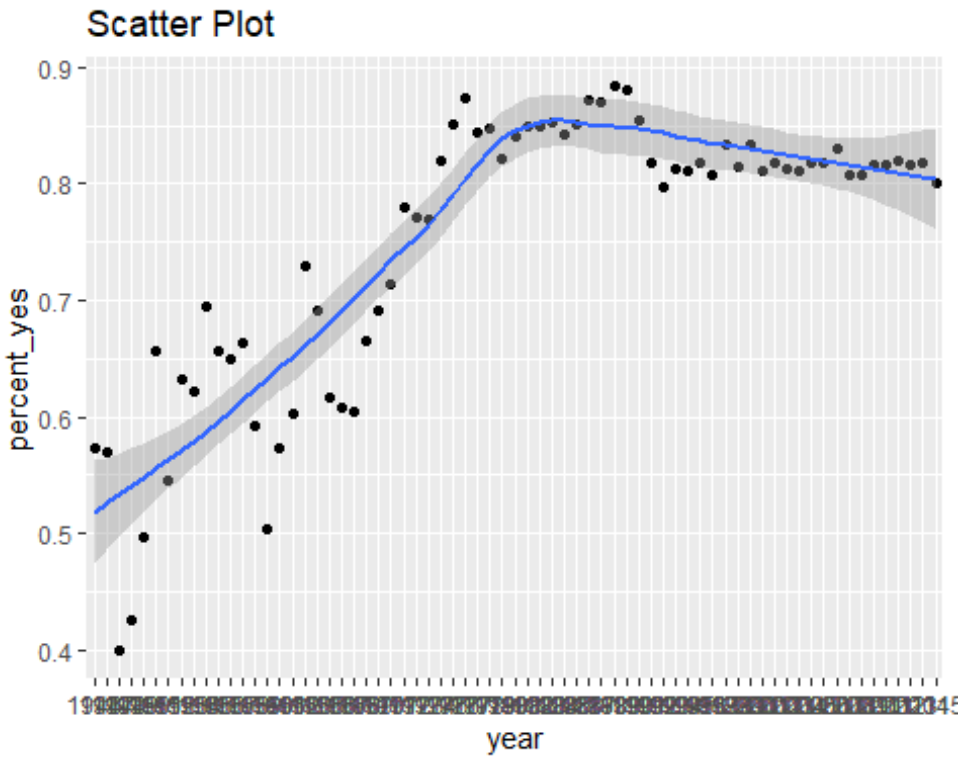
```
## [1] 200
```

```
length(by_countrymod$country)
```

```
## [1] 199
```

Con el package **ggplot2** muestro en un gráfico cómo la frecuencia de los votos positivos varía con el tiempo. Agrego, con la función **geom_smooth()**, una curva que se adapta a las observaciones para mostrar la tendencia.

```
library(ggplot2)

ggplot(by_year,aes(x=year, y=percent_yes, group=1 )) +
  geom_point() +
  geom_smooth() +
  ggtitle("Scatter Plot")
```



Agrupo la frecuencia de votos positivos de cada país dividido para cada año.

```
by_year_country <- votes_per2 %>%
  group_by(country, year) %>%
  summarize(total = n(),
            percent_yes = mean(vote == "yes"))
by_year_country
```

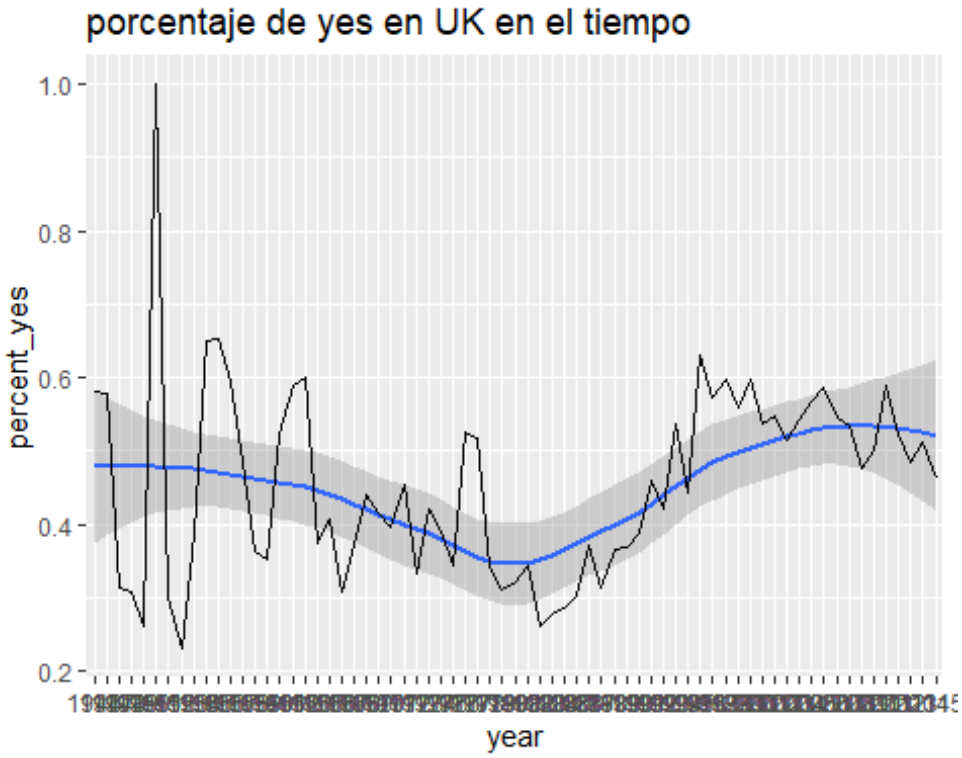
```
## # A tibble: 9,689 x 4
## # Groups:   country [200]
##    country     year  total percent_yes
##    <chr>       <chr> <int>       <dbl>
##  1 Afghanistan 1946     17       0.412
##  2 Afghanistan 1947     34       0.382
##  3 Afghanistan 1948     64       0.344
##  4 Afghanistan 1949     81       0.457
##  5 Afghanistan 1950     50       0.7
##  6 Afghanistan 1951      7       0.143
##  7 Afghanistan 1952     68       0.647
##  8 Afghanistan 1953     26       0.769
##  9 Afghanistan 1954     29       0.724
## 10 Afghanistan 1955     37       0.730
## # … with 9,679 more rows
```

Filtro la tabla recién creada para tener en cuenta solo los datos del Reino Unido. Ahora tengo una serie temporale para este país.

```r
UK_by_year <- by_year_country %>%
  filter(country == "United Kingdom of Great Britain and Northern Ireland")
UK_by_year
```

```
## # A tibble: 69 x 4
## # Groups:   country [1]
##    country                                                year  total percent_yes
##    <chr>                                                  <chr> <int>       <dbl>
##  1 United Kingdom of Great Britain and Northern Ireland 1946     43       0.581
##  2 United Kingdom of Great Britain and Northern Ireland 1947     38       0.579
##  3 United Kingdom of Great Britain and Northern Ireland 1948     64       0.312
##  4 United Kingdom of Great Britain and Northern Ireland 1949    104       0.308
##  5 United Kingdom of Great Britain and Northern Ireland 1950     50       0.26
##  6 United Kingdom of Great Britain and Northern Ireland 1951      7       1
##  7 United Kingdom of Great Britain and Northern Ireland 1952     70       0.3
##  8 United Kingdom of Great Britain and Northern Ireland 1953     26       0.231
##  9 United Kingdom of Great Britain and Northern Ireland 1954     31       0.387
## 10 United Kingdom of Great Britain and Northern Ireland 1955     37       0.649
## # … with 59 more rows
```

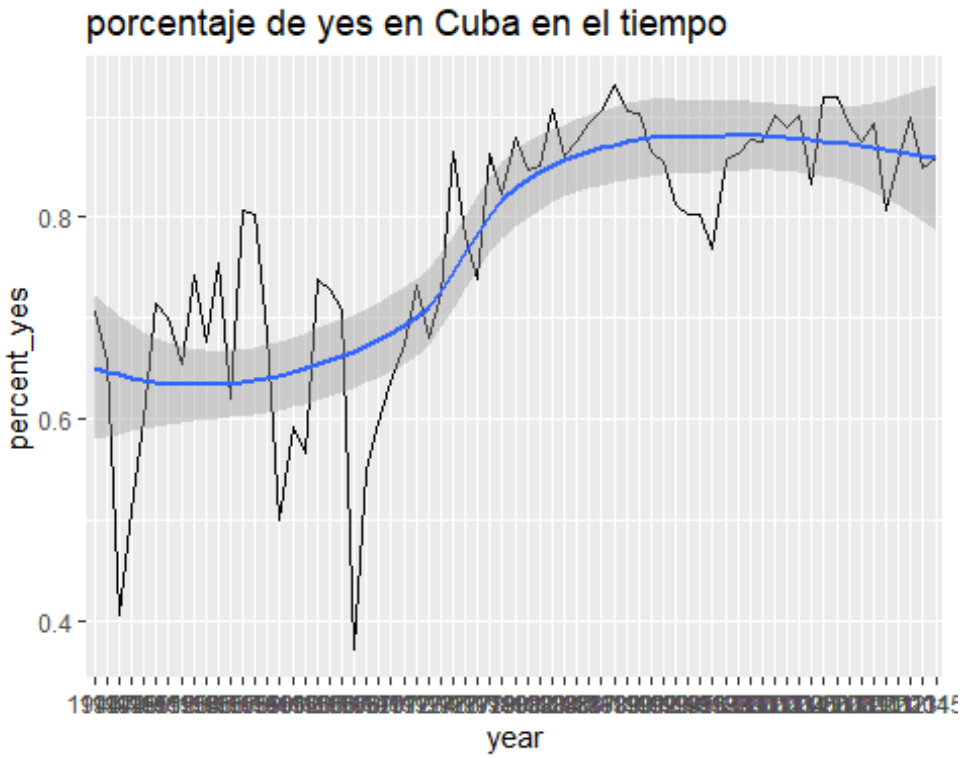
```r
ggplot(UK_by_year, aes(year, percent_yes,group=1)) + geom_smooth()+
  geom_line()+ggtitle("porcentaje de yes en UK en el tiempo")
```



Del mismo modo para Cuba

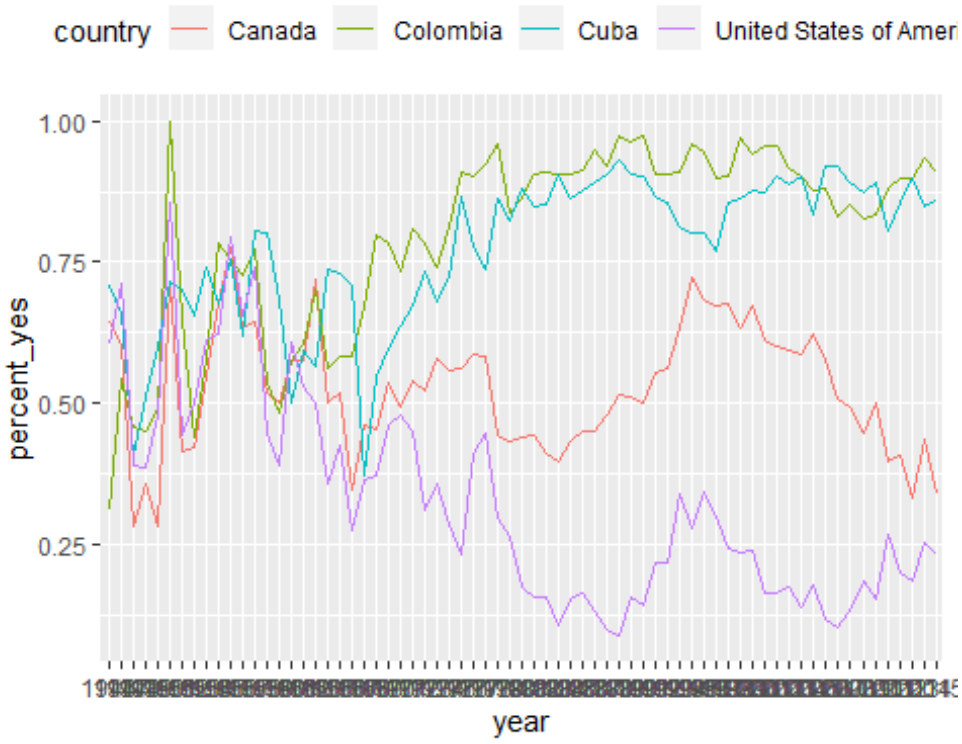
```r
cuba_by_year <- by_year_country %>%
  filter(country == "Cuba")
```

```
ggplot(cuba_by_year, aes(year, percent_yes,group=1)) +
  geom_line()+ geom_smooth() + ggtitle("porcentaje de yes en Cuba en el tiempo")
```

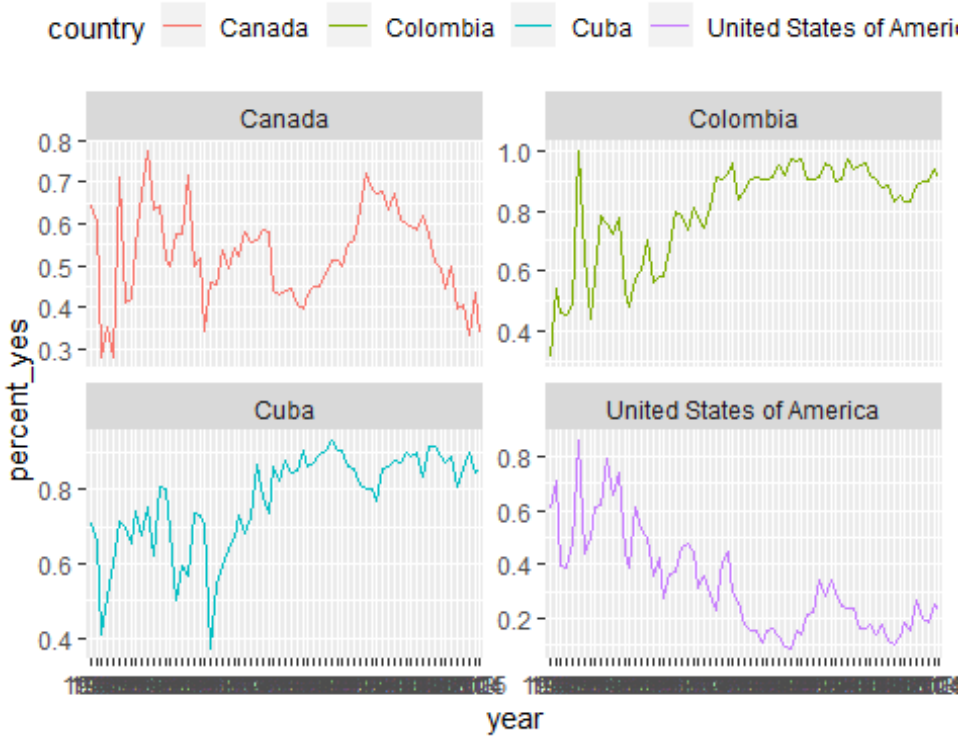


Ahora de la misma manera tomamos algunos países primero. Comparamos la evolución de la frecuencia de calificaciones positivas a largo plazo. Con el comando **facet_wrap** creo un gráfico para cada estado individual. Al colocar **scales**, escalo el gráfico en función de los valores asumidos por **percent_yes**.

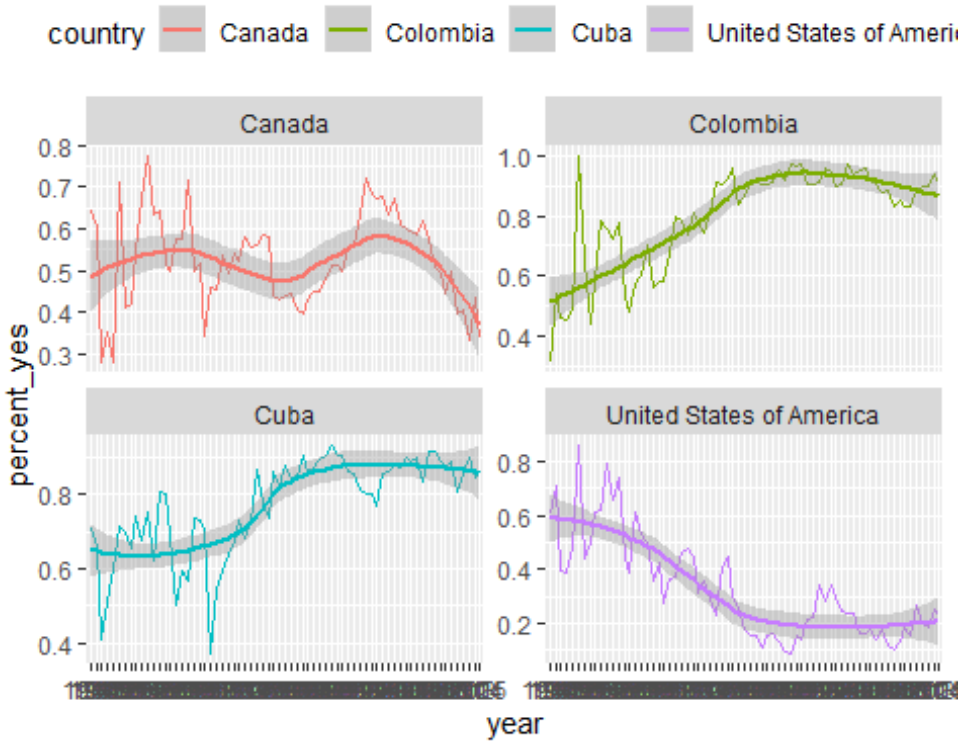
```
paises = c("United States of America","Canada","Cuba","Colombia")

filtered_4_countries <- by_year_country %>%
  filter(country %in% paises)

ggplot(filtered_4_countries, aes(year, percent_yes, color =
country,group=country)) +
  geom_line()+ theme(legend.position="top")
```




```
ggplot(filtered_4_countries, aes(year, percent_yes, color = country, group =
country)) +
  geom_line() +  facet_wrap(~ country,scales="free_y")+
theme(legend.position="top")
```

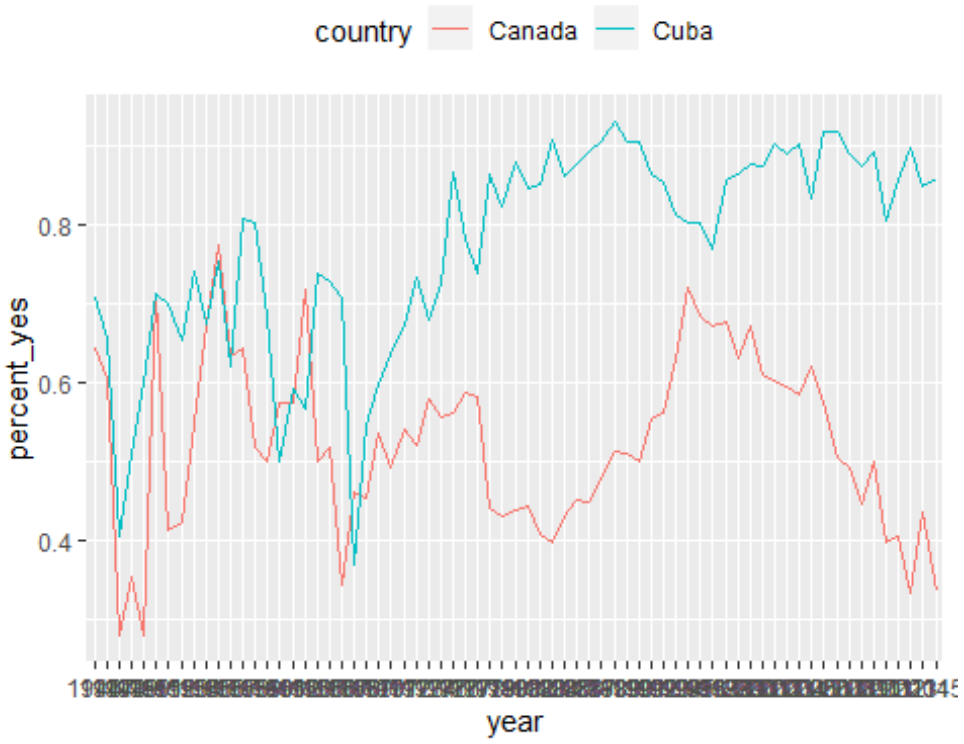


```r
ggplot(filtered_4_countries, aes(year, percent_yes, color = country, group =
country)) +
  geom_line() + geom_smooth() + facet_wrap(~ country,scales="free_y")+
theme(legend.position="top")
```

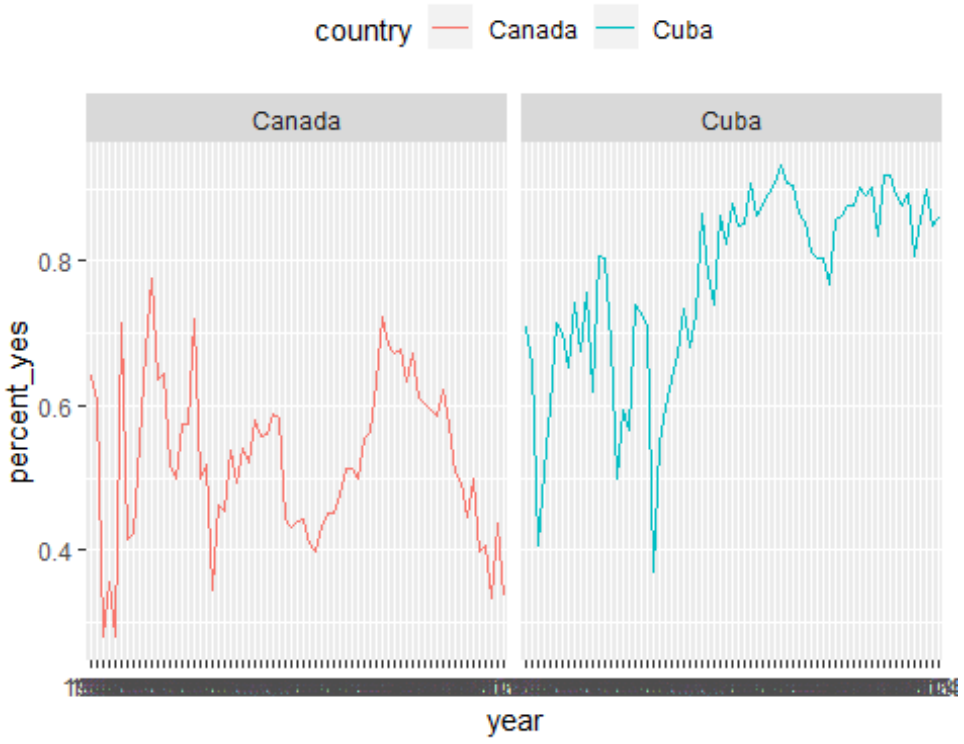


Repetimos lo mismo solo para dos países: Canadá y Cuba.

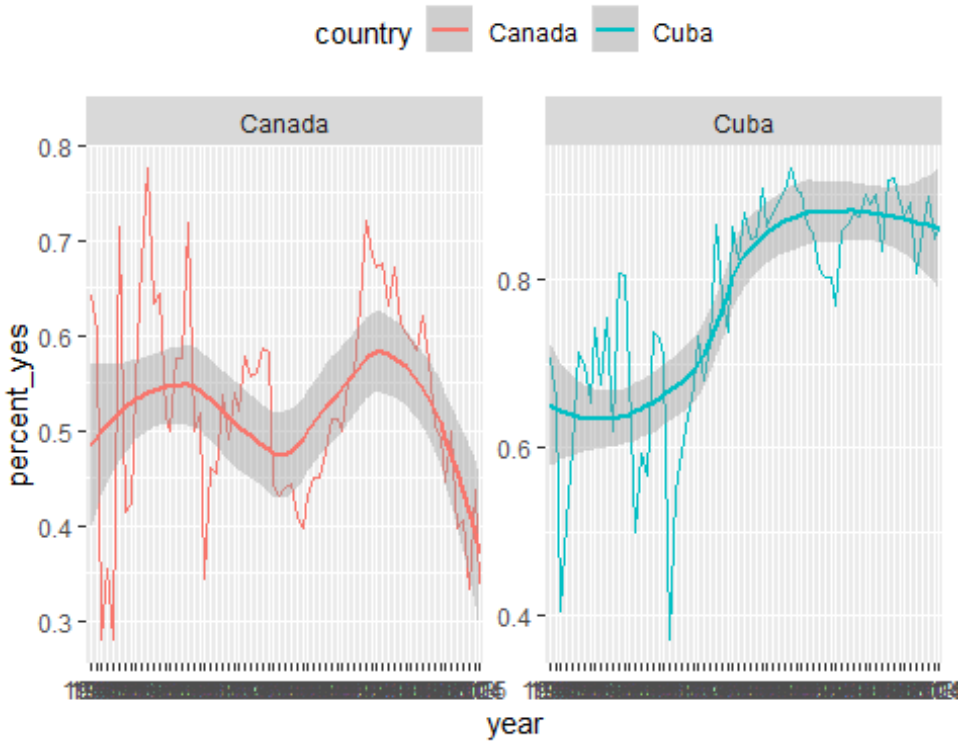
```r
paises2 = c("Canada","Cuba")
filtered_2_countries <- by_year_country %>%
  filter(country %in% paises2)

ggplot(filtered_2_countries, aes(year, percent_yes, color =
country,group=country)) +
  geom_line() + theme(legend.position="top")
```




```
ggplot(filtered_2_countries, aes(year, percent_yes, color = country, group =
country)) +
  geom_line() + facet_wrap(~ country)+ theme(legend.position="top")
```



```
ggplot(filtered_2_countries, aes(year, percent_yes, color = country, group =
country)) +
  geom_line() + facet_wrap(~ country,scales="free_y")+ geom_smooth()+
theme(legend.position="top")
```



¿Hay países que tengan un comportamiento similar a largo plazo?

Parte 2

El conjunto de datos bajo análisis tiene 200 países, para un total de 69 años de votación, desde 1946 hasta 2015. En 1951 solo hubo 402 votos, en contraste con los años anteriores (en 1950 2911) y posteriores (en 1952 4082). Sin embargo, algunos países se unieron a la ONU en los años siguientes y los datos no están completos. Por ejemplo, los datos de Suiza comenzaron en 2002, el año en que ingresó el país. El número de temas es 6, y son:

```
a<-un_roll_call_issues %>% group_by(issue)%>% summarize(total=n())
a$issue

## [1] "Arms control and disarmament"
## [2] "Colonialism"
## [3] "Economic development"
## [4] "Human rights"
## [5] "Nuclear weapons and nuclear material"
## [6] "Palestinian conflict"
```

Veamos el número de votos totales, positivos,abstenidos y negativos.

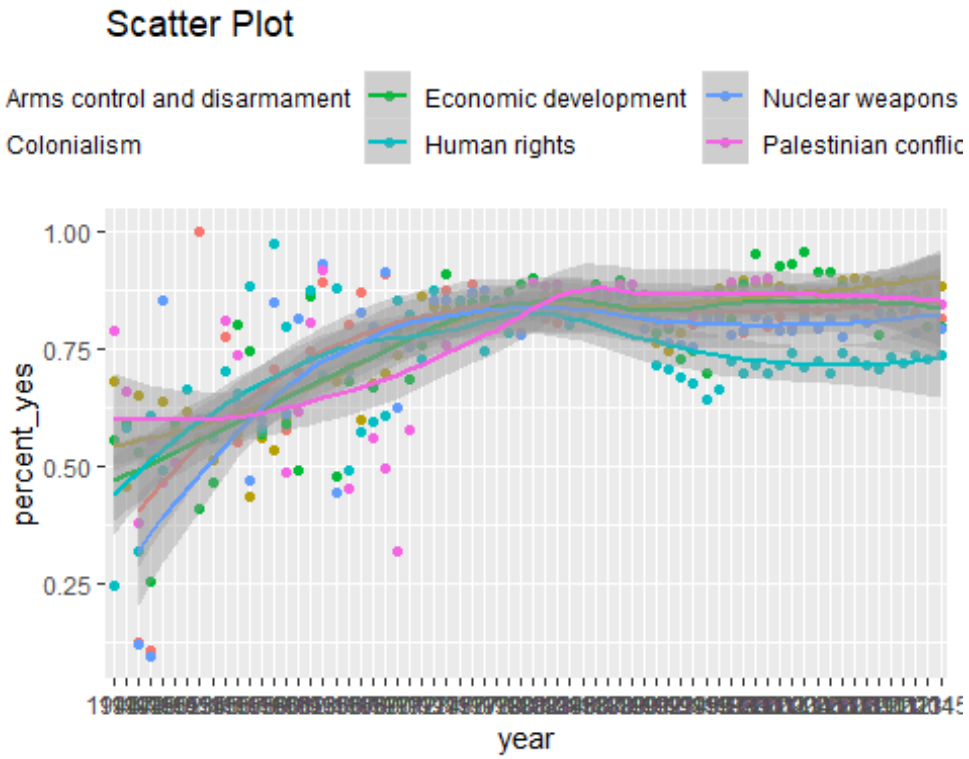
```
vt<-as.numeric(un_votes %>% count)
vy<-as.numeric(un_votes %>% filter(vote=="yes") %>% count)
va<-as.numeric(un_votes %>% filter(vote=="abstain") %>% count)
vn<-as.numeric(un_votes %>% filter(vote=="no") %>% count)
data.frame(Total=vt, Yes=vy, Abstain=va,No=vn)

##    Total    Yes Abstain    No
## 1 738764 588800   95099 54865
```

2.1 Análisis de series temporales

2.1.1 Cluster por temas

Ahora miro cómo el porcentaje de sí varía con el tiempo para los diferentes temas de votación. La tendencia es muy similar para todos los temas: el rápido crecimiento se vuelve casi constante.

```
## Warning: Column `rcid` has different attributes on LHS and RHS of join
```



Verifiquemos con el paquete **TSclust** si podemos agrupar los temas en grupos. Sin embargo, para este problema, no podemos confiar en las medidas clásicas de distancia entre observaciones (como el enlace único, el método de Ward ...) porque es necesario mantener la información temporal.

```
library(TSclust)
library(seriation)
```

Primero creo una matriz que contiene las observaciones para poder usar el comando **diss()**. De esta manera, tengo una matriz que para cada columna contiene el número de porcentaje de votos positivos para los temas individuales. Mientras que para cada fila tenemos las series de tiempo para el tema específico. Con unos pocos pasos obtengo la siguiente matriz.

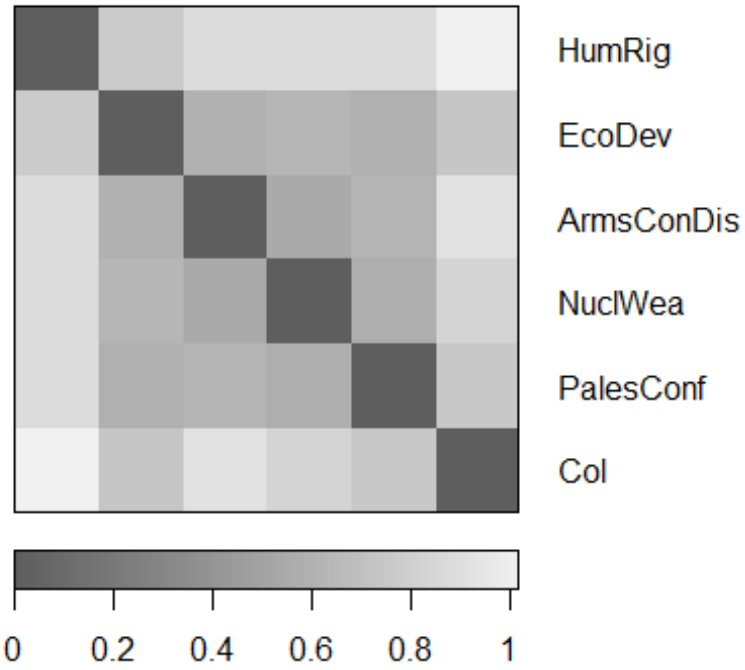
```
mar[1:5,1:5]
```

```
##                [,1]      [,2]      [,3]     [,4]      [,5]
## Col        0.6793194 0.4585859 0.6516691 0.512722 0.6380832
## EcoDev     0.0000000 0.0000000 0.0000000 0.000000 0.0000000
## HumRig     0.0000000 0.2448980 0.5809524 0.320197 0.6098901
```

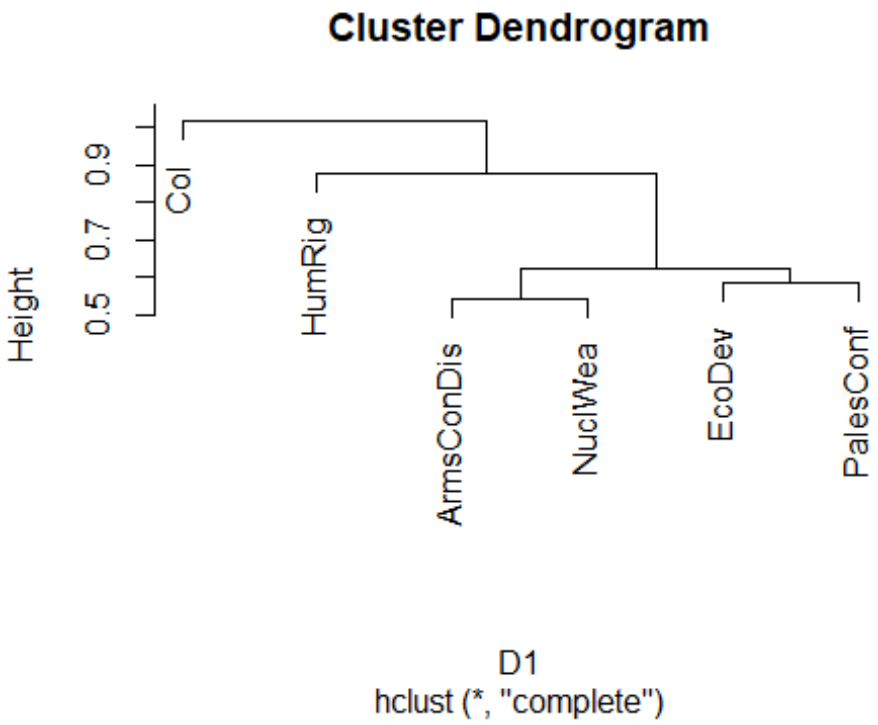
```
## PalesConf  0.0000000 0.0000000 0.0000000 0.000000 0.0000000
## ArmsConDis 0.0000000 0.0000000 0.0000000 0.000000 0.0000000
```

Con el comando **diss()** calculo el índice de disimilitud con la correlación de Pearson sobre el comportamiento temporal de la serie. De ella obtengo una matriz, cuanto menor es el valor entre dos series y más tienen un comportamiento similar. Con el comando **dissplot()** influyes visualmente en cuáles son los clústeres, y luego son visibles con **plot()**.

```
 D1 <- diss(mar, "COR")
dissplot(D1)
```

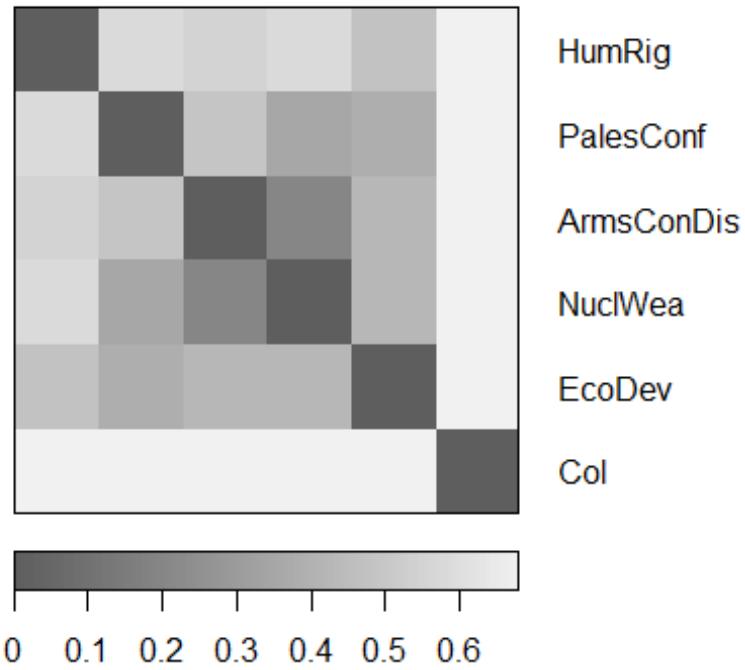



```
 C1 <- hclust(D1)
plot(C1)
```

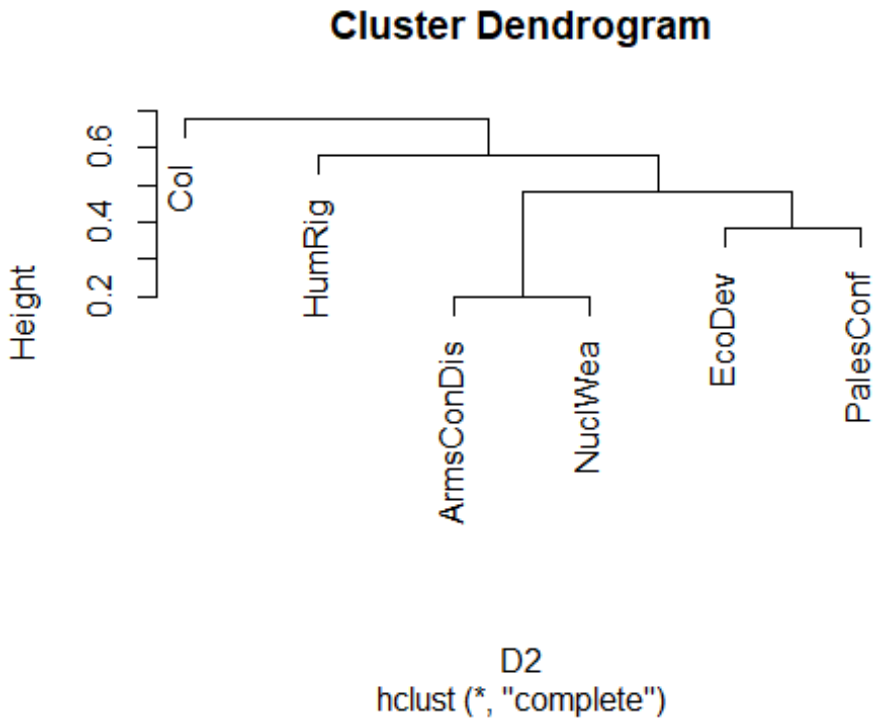


Para este problema, trato de usar la distancia de Frechet para series históricas. En este enlace puedes encontrar más información al respecto https://en.wikipedia.org/wiki/Fr%C3%A9chet_distance .

```
D2 <- diss(mar, "FRECHET")

dissplot(D2)
```




```
C2 <- hclust(D2)
plot(C2)
```

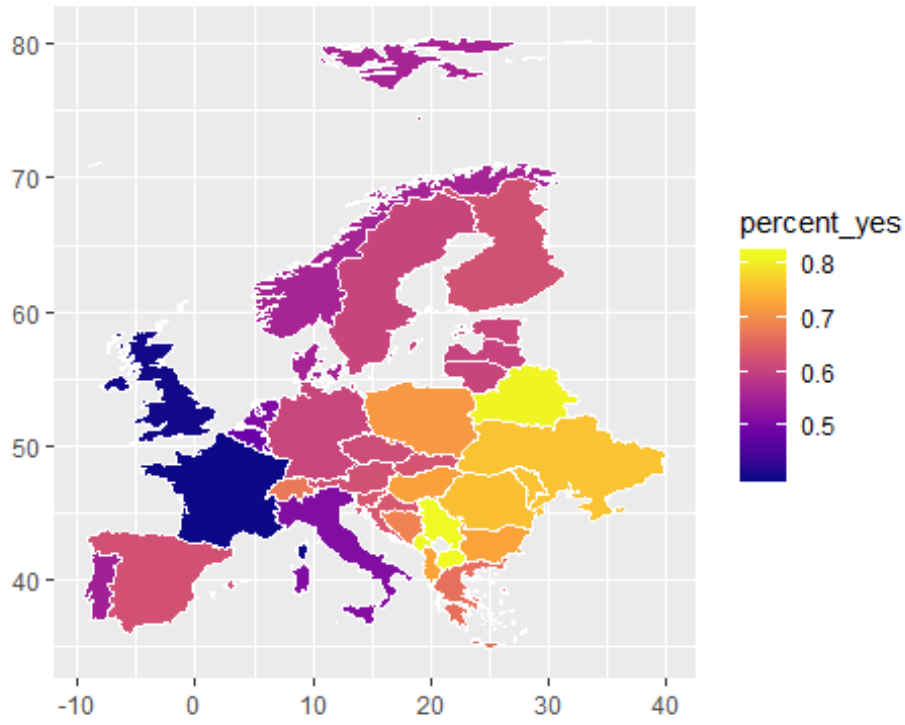


Ahora es interesante comparar los resultados de los dos métodos. Es decir, vea cómo, colocando el número de clústeres igual a 3, los dos métodos agrupan los temas de manera diferente.

```
groups <- cutree(C1, k=3)
groups

##        Col     EcoDev     HumRig  PalesConf ArmsConDis    NuclWea
##          1          2          3          2          2          2

groups <- cutree(C2, k=3)
groups

##        Col     EcoDev     HumRig  PalesConf ArmsConDis    NuclWea
##          1          2          3          2          2          2
```

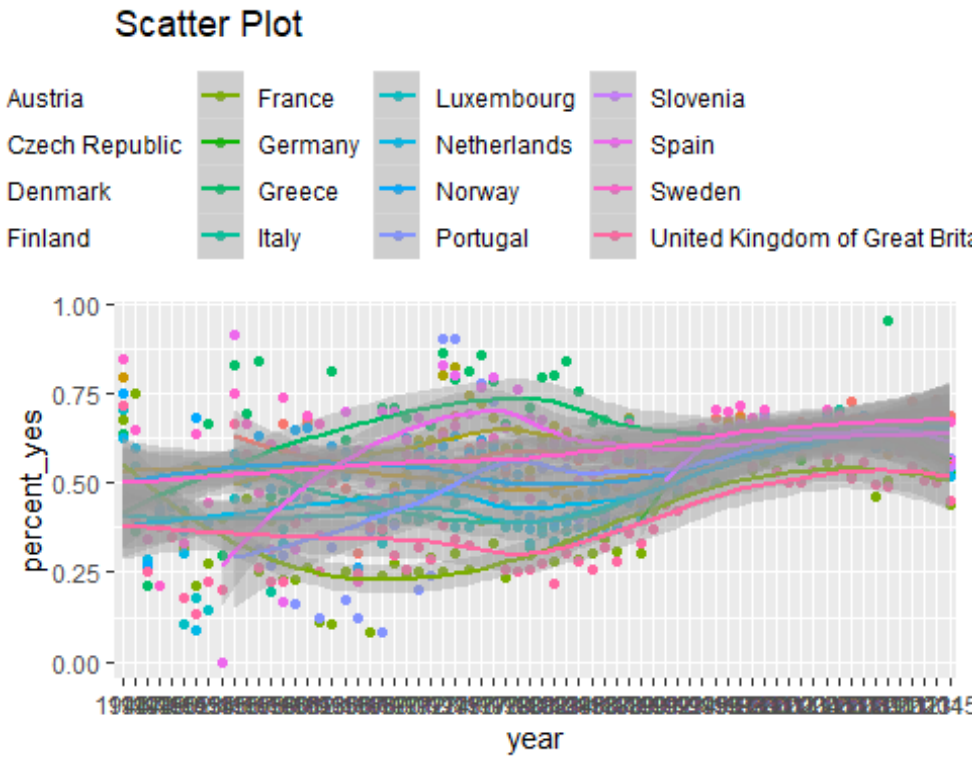
2.1.2 Cluster por pais europeo

Realizo un análisis de cluster para los principales países europeos teniendo en cuenta la evolución del porcentaje de votos positivos. Para algunos países, las observaciones iniciales son iguales a cero, porque no formaron parte de la ONU, por esos años distintos no habían votado. En el siguiente mapa puede ver el promedio de calificaciones positivas para todos los años en que los países han votado por cada nación.



El gráfico de series de tiempo para los principales países europeos es el siguiente. Los paises son : **Portugal**, **Spain**, **France**, **Germany**, **Austria**, **United Kingdom of Great Britain and Northern**

Ireland, **Netherlands**, **Denmark**, **Italy**, **Norway**, **Sweden**, **Finland**, **Slovenia**, **Czech Republic**, **Greece**y **Luxembourg**.

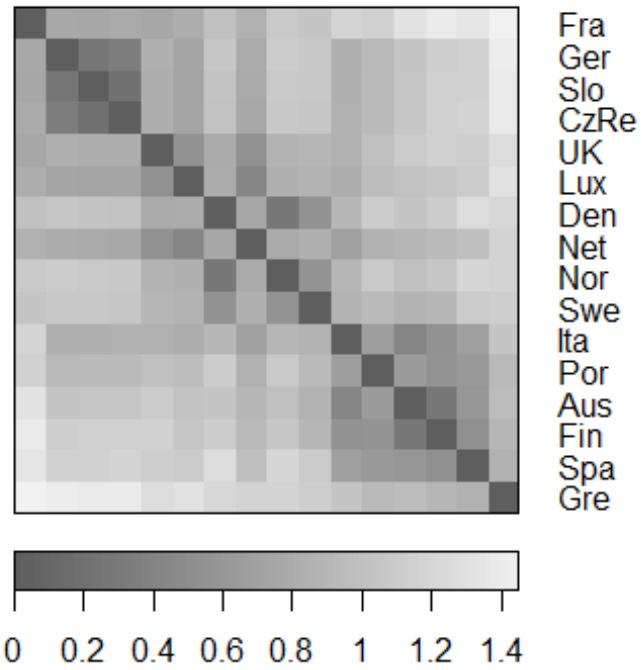


Obtengo la matriz de valores en orden cronológico, los primeros valores son los siguientes.

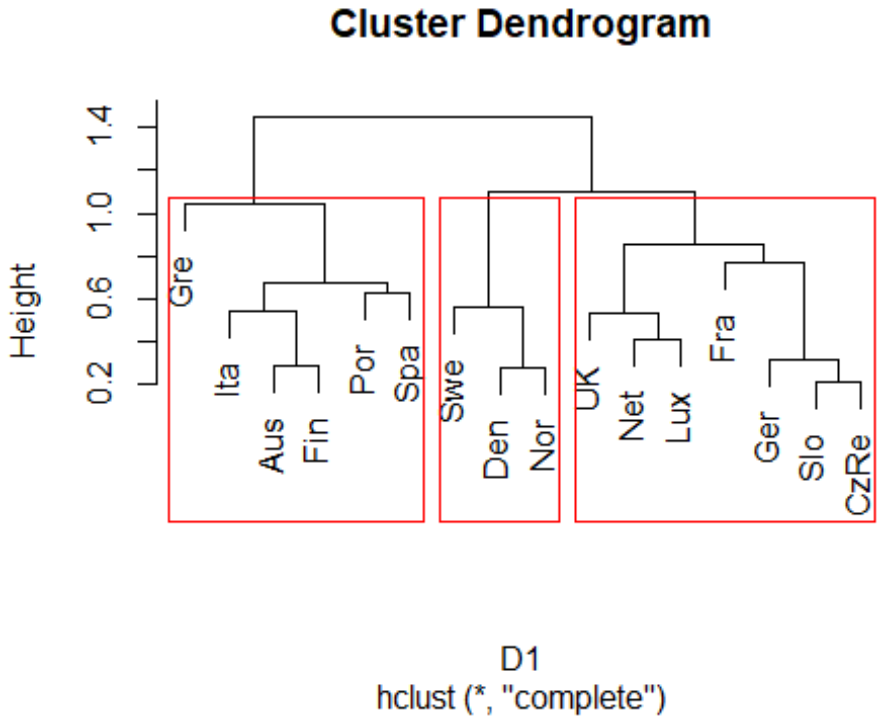
```
mar[1:5,1:5]

##     [,1] [,2]      [,3]      [,4]      [,5]
## Por 0.00 0.00 0.0000000 0.0000000 0.0000000
## Spa 0.00 0.00 0.0000000 0.0000000 0.0000000
## Fra 0.68 0.75 0.3461538 0.4177215 0.3478261
## Ger 0.00 0.00 0.0000000 0.0000000 0.0000000
## Aus 0.00 0.00 0.0000000 0.0000000 0.0000000
```

Usando la correlación de Pearson, tomamos los 3 grupos y verificamos qué estados están contenidos dentro.

```
 D1 <- diss(mar, "COR")
dissplot(D1)
```




```
 C1 <- hclust(D1)
plot(C1)
rect.hclust(C1, k=3)
```



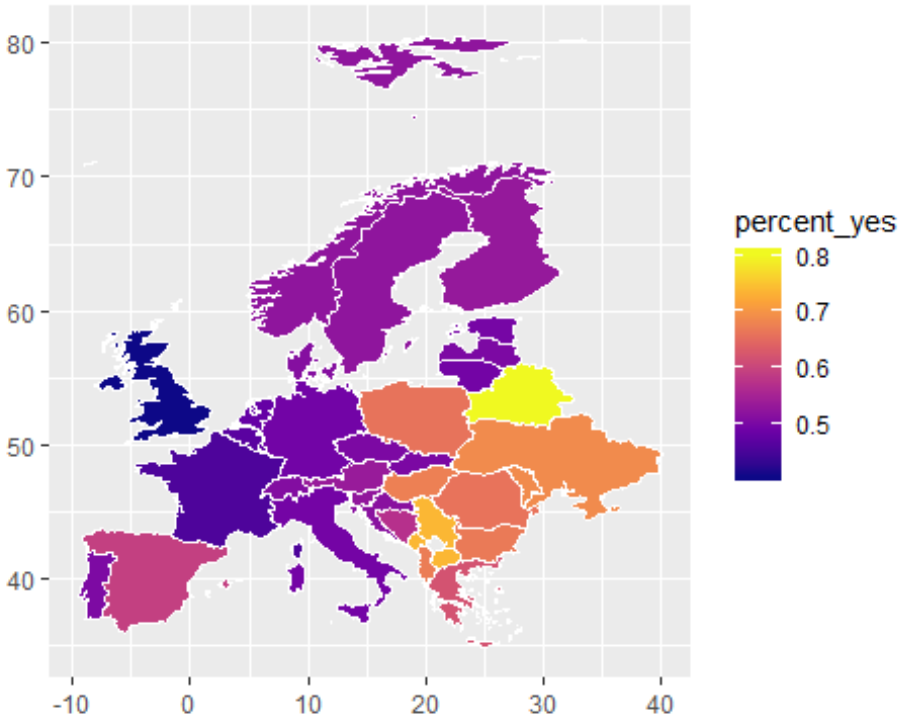
```
groups <- cutree(C1, k=3)
groups
```

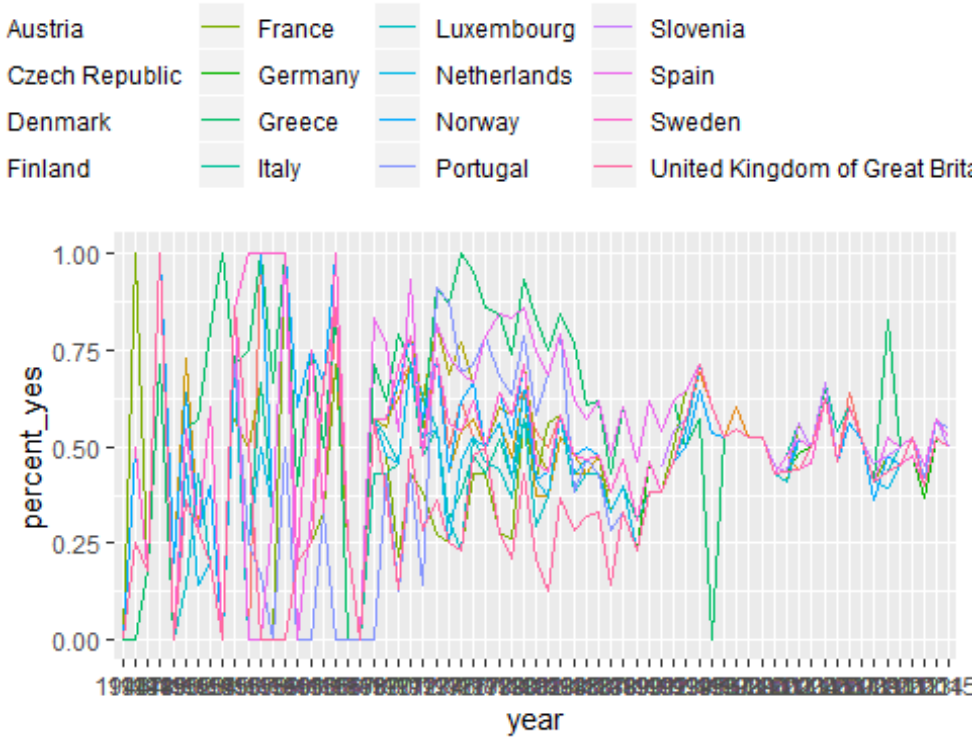
```
##  Por  Spa  Fra  Ger  Aus   UK  Net  Den  Ita  Nor  Swe  Fin  Slo CzRe  Gre  Lux
##    1    1    2    2    1    2    2    3    1    3    3    1    2    2    1    2
```

2.1.3 Cluster por pais europeo y tema

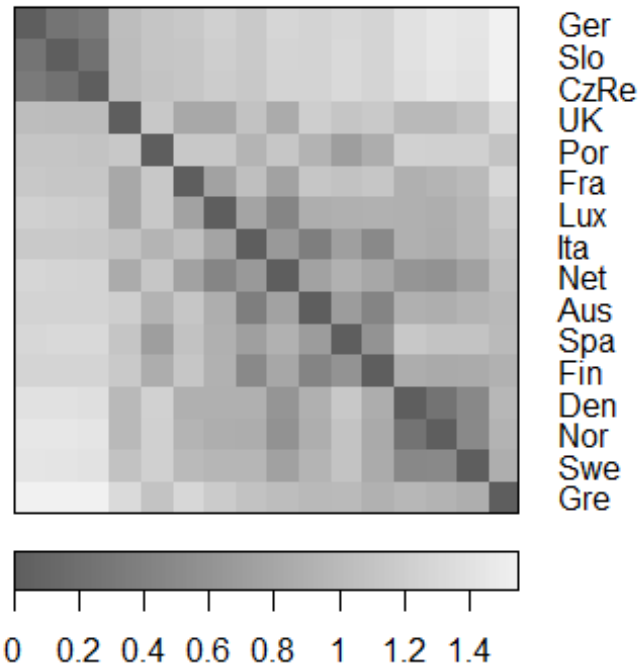
Ahora, hago el mismo proceso seleccionando un solo tema. en el primer caso los votos cuyo tema era los derechos humanos, mientras que en el segundo caso el colonialismo.

2.1.3.1 Tema: human right

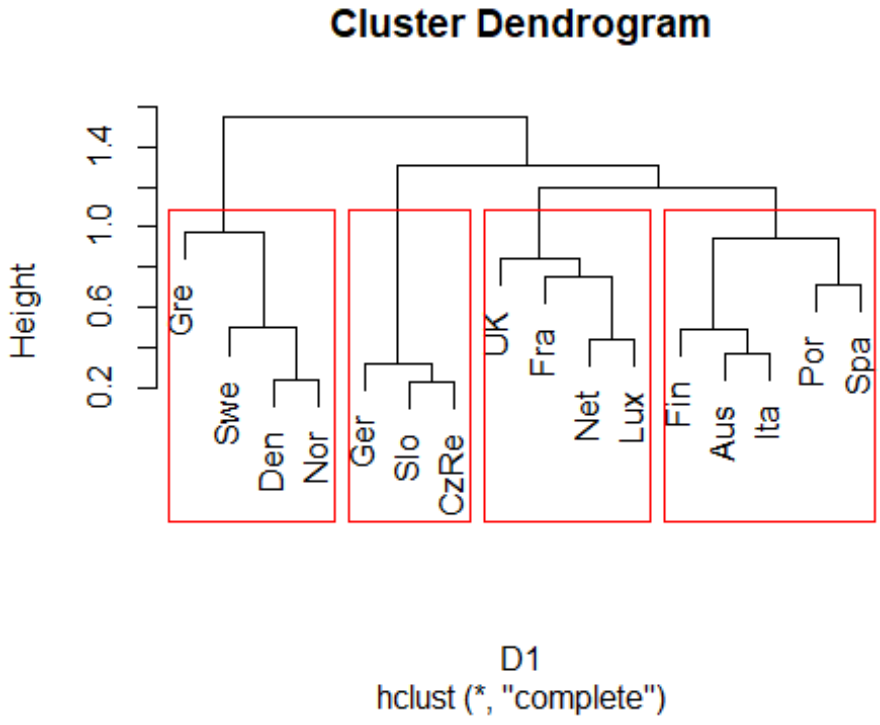





```
mar[1:5,1:5]

##     [,1] [,2] [,3]      [,4] [,5]
## Por    0    0    0 0.0000000    0
## Spa    0    0    0 0.0000000    0
## Fra    0    0    1 0.1818182    1
## Ger    0    0    0 0.0000000    0
## Aus    0    0    0 0.0000000    0

 D1 <- diss(mar, "COR")
dissplot(D1)
```



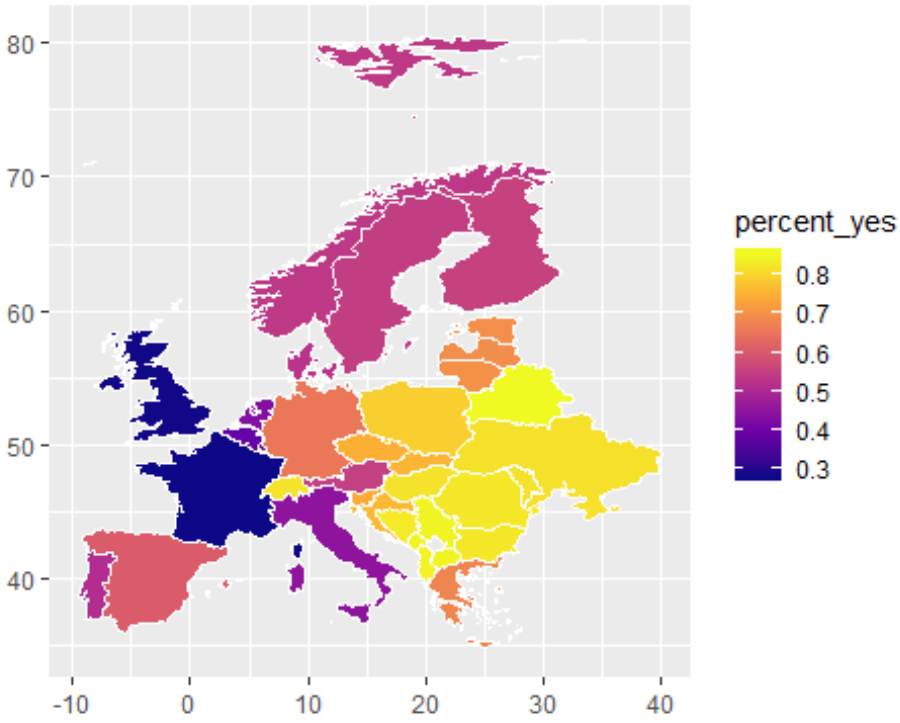

```
 C1 <- hclust(D1)
plot(C1)
rect.hclust(C1, k=4)
```

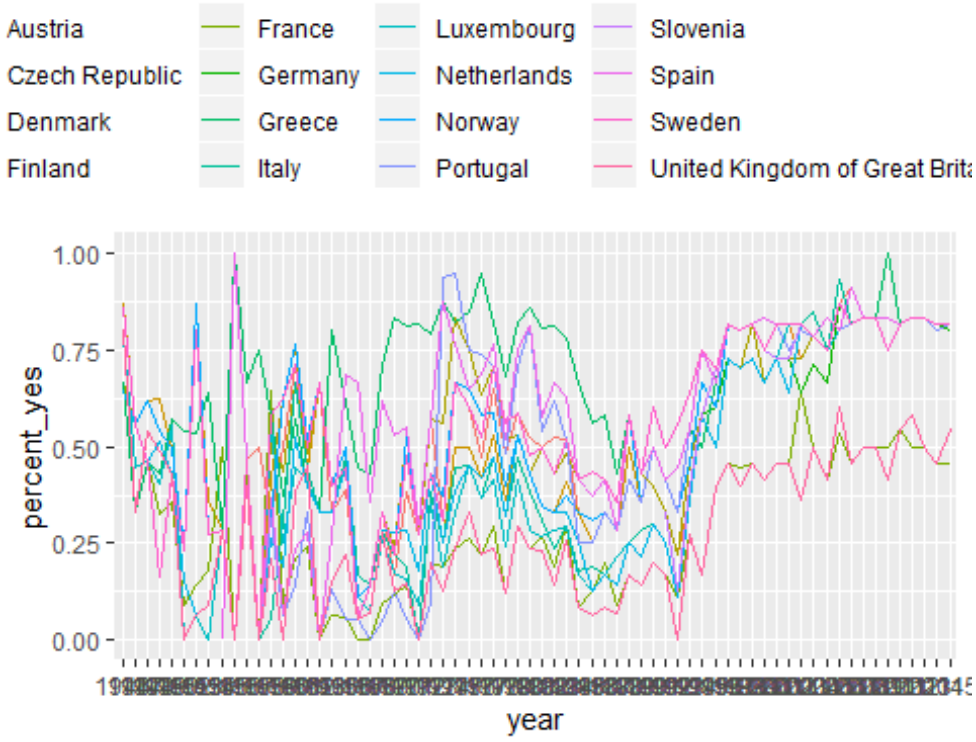


```
groups <- cutree(C1, k=4)
groups
```

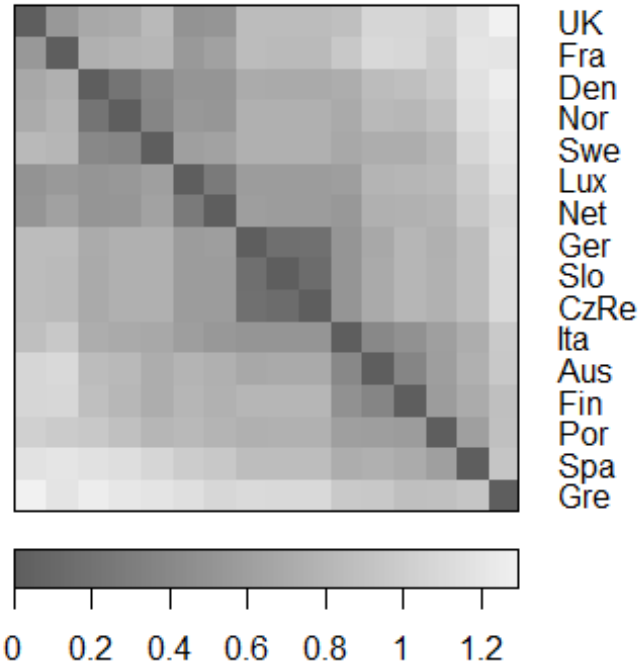
```
##  Por  Spa  Fra  Ger  Aus   UK  Net  Den  Ita  Nor  Swe  Fin  Slo CzRe  Gre  Lux
##    1    1    2    3    1    2    2    4    1    4    4    1    3    3    4    2
```

2.1.3.2 Tema: colonialism

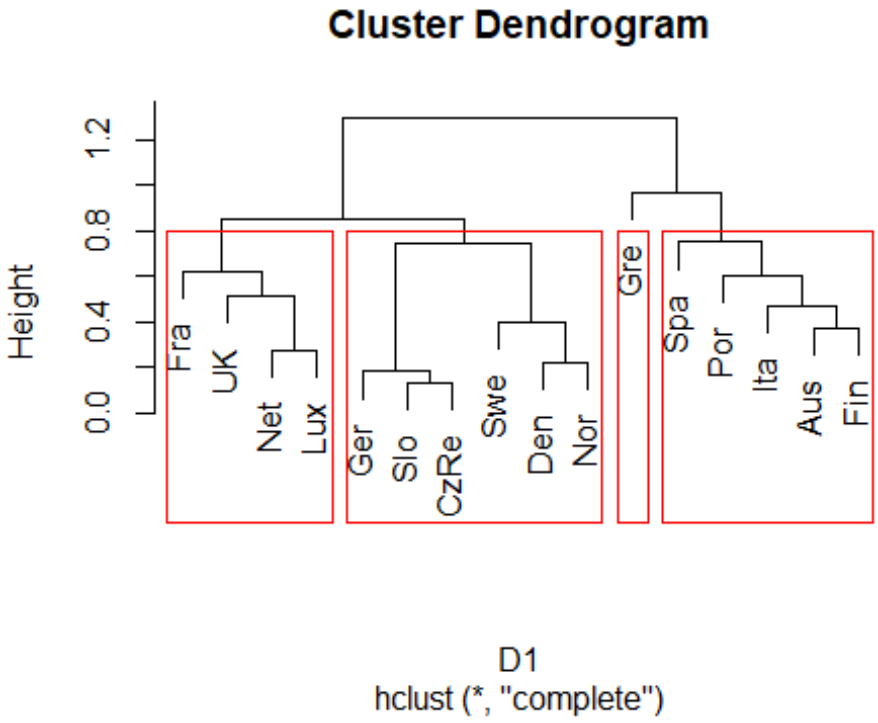





```
 D1 <- diss(mar, "COR")
dissplot(D1)
```



```
 C1 <- hclust(D1)
plot(C1)
rect.hclust(C1, k=4)
```




```
groups <- cutree(C1, k=4)
groups
```

```
##  Por  Spa  Fra  Ger  Aus   UK  Net  Den  Ita  Nor  Swe  Fin  Slo CzRe  Gre  Lux
##    1    1    2    3    1    2    2    3    1    3    3    1    3    3    4    2
```

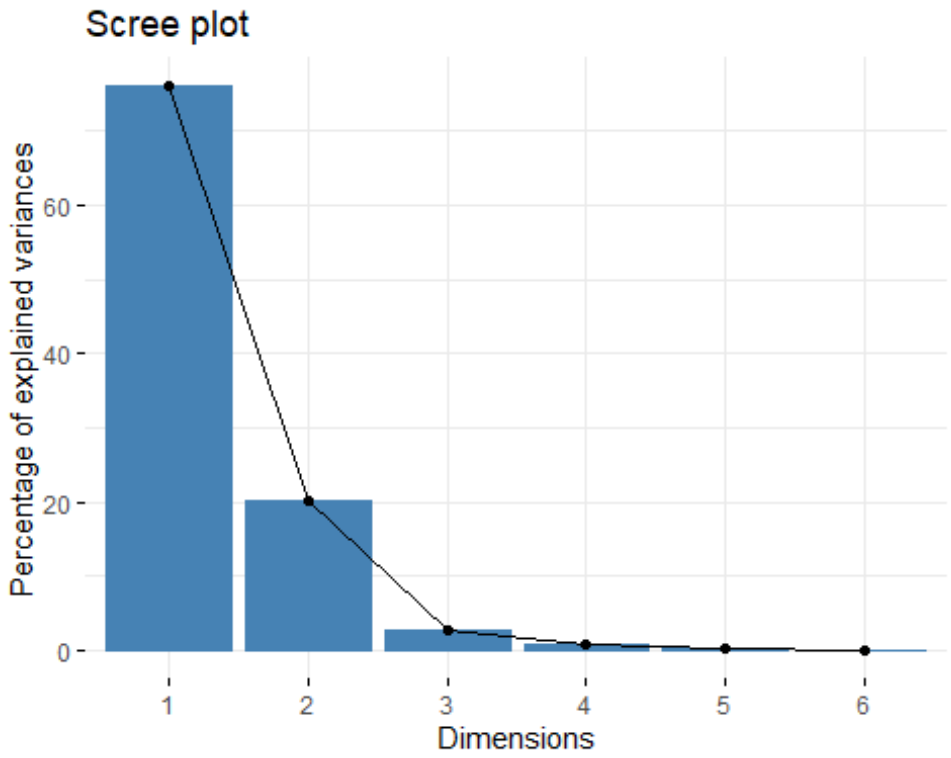
2.2 PCA por temas

Ahora llevo a cabo un análisis de PCA tomando como referencia los principales estados europeos y los porcentajes de votos positivos para los diversos temas como variables.

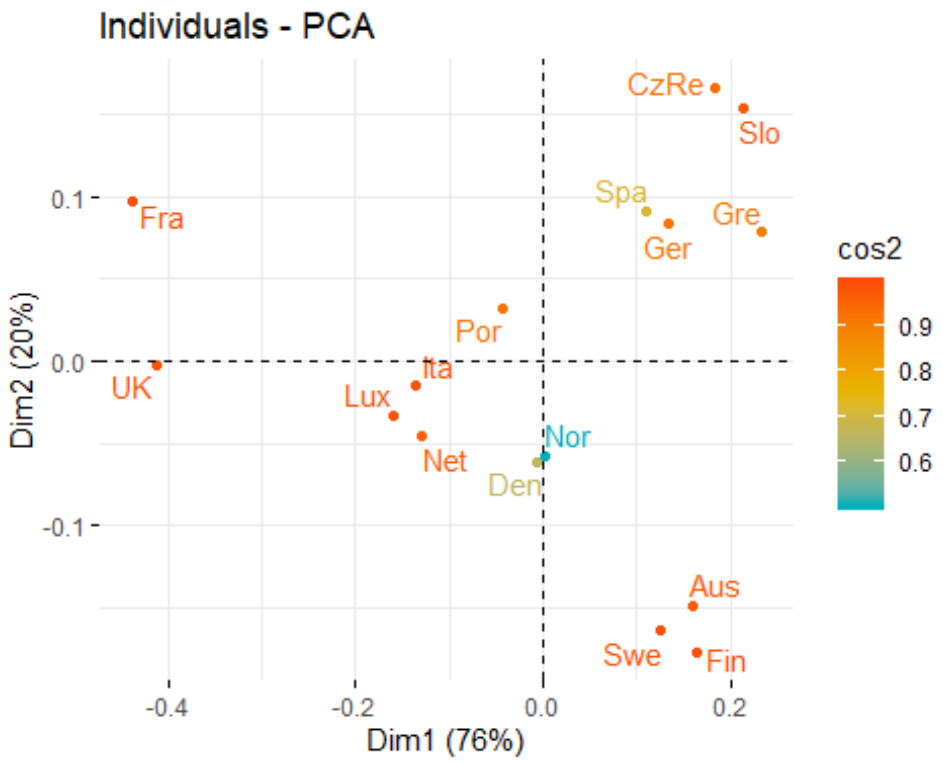
Para hacer esto, uso el package **factorextra**.

```
library(factoextra)
```

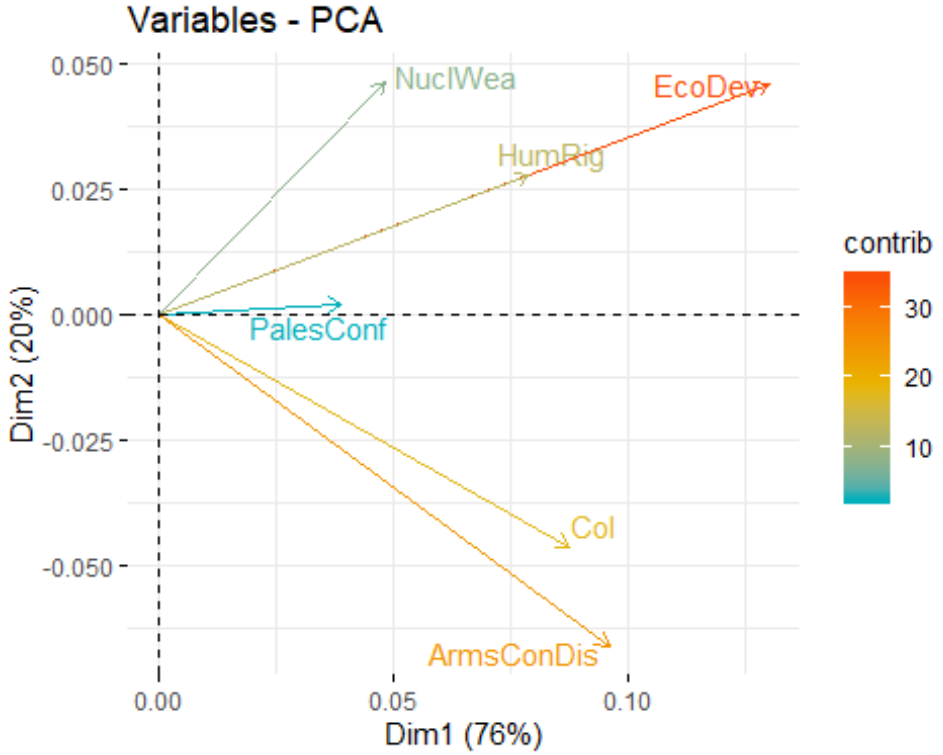
El scree plot (el valor de los eigenvalues) muestra que teniendo en cuenta los dos primeros PC capturamos aproximadamente el 95% de la información.



El siguiente gráfico muestra cómo las naciones se dispersan teniendo en cuenta solo los dos primeros componentes principales.



Aquí puede ver qué tipo de correlación entre las variables que hay: positivo si se encuentran en la misma parte de la gráfica, de otro modo negativo.



Para terminar el biplots, que resume los dos gráficos visto antes.