Procesamiento del lenguaje natural de corpus lingüísticos del dominio.

OTIC ANALYZER

Alexander Cole Mora | Rafael Herrero Álvarez | Raúl Martín Morales

Daniel Ramos Acosta | Pedro Ramos Rodríguez | Moisés Yanes Carballo

Índice

Introducción

Para la realización de esta práctica se nos pide utilizar alguna herramienta que nos permita procesar en lenguaje natural el corpus de un dominio sobre el turismo. En nuestro caso, trabajaremos con 3 lugares, Canarias, Londres y Nueva York. Con esto lugares, analizaremos la información sobre el clima, los lugares más visitados o emblemáticos y la procedencia de los turistas en estas zonas.

Obtendremos la información descargándola de distintas páginas webs. Todas ellas se encuentran en el anexo al final del documento. Todas esas páginas web las hemos descargado en formato PDF y las hemos convertido a txt utilizando la herramienta online PDF a TXT. Además, en el caso de la procedencia de los turistas, hemos obtenido dicha información en formato csv.

Todo el tratamiento de la información la realizamos con el lenguaje R, utilizando el programa R Studio. Con esto, generamos una serie de scripts para cada ámbito y obtenemos una información de manera gráfica. Para el clima y los lugares, esta información la sacamos con una nube de palabras, las cuales tendrán un color distintos según la cantidad de veces que se repitan en los distintos documentos. Para la procedencia, los gráficos cambian según los lugares, ya que en este caso hemos decidido crear otros distintos.

Los scripts usados se encuentran en los apartados siguientes. Los paquetes de R necesarios son los siguientes: `tm, wordcloud, lsa, lattice y stringr`. En todos los casos hemos realizado las siguientes modificaciones:

- Convertir el texto a minúscula.
- La eliminación de los signos de puntuación.
- La eliminación de los caracteres numéricos.
- Eliminar los espacios en blancos.
- Supresión de las palabras vacías en inglés.

1. Canarias

Para llevar a cabo el análisis del procesamiento del lenguaje en Canarias, se ha realizado un búsqueda en profundidad de diferentes documentos, con el fin de sacar algunas conclusiones interesantes. Estos documentos, tendrán relación con tres diferentes aspectos relacionados con el turismo:

Clima: Se han buscado ficheros que guarden relación con el clima de Canarias, en diferentes épocas e incluso comparaciones con otros lugares que compiten con las islas en el mismo sector.

Lugares: Se intenta destacar los lugares que mayor visita o interés sugieren entre los turistas que ya han visitado las islas o que puedan hacerlo en el futuro.

Procedencia: Se busca conseguir un resumen de los diferentes lugares de donde proceden los turistas.

Clima

En relación con el clima, hemos encontrado cinco diferentes documentos en los que se trata del tema en las Islas Canarias. Principalmente, la información encontrada tiene mayor porcentaje con las islas capitalinas, tanto Tenerife como Gran Canaria, pero también se incluye al resto de las islas.

Para realizar el análisis, hemos realizado un script en R en el cual mostramos en una nube de palabras, aquellas que más representadas están en los documentos. Es importante saber, que tratamos el tema de clima, puesto que muchas de las palabras que se muestran guardan una relación directa con este término.

En el script, cargamos los cinco ficheros TXT encontrados previamente convertidos a ese formato, pues en su origen, eran documentos en PDF. Luego procesamos los documentos, los pasamos a minúsculas, eliminamos los espacios en blanco, puntuaciones y palabras vacías del castellano. También se han eliminado otras

palabras que en el resultado no nos interesaba, como es el caso de Canarias o palabras raras que se han colado en los ficheros. El script es el siguiente:

```
library(tm)
library(wordcloud)

###FICHEROS DE CLIMA
#txt clima1
clima1                                                                          <-
readLines("C:/Users/alex_/Desktop/ScriptRCanarias/Clima1.txt",encoding="UTF-8")
clima1 = iconv(clima1,to="ASCII//TRANSLIT")
#txt clima2
clima2                                                                          <-
readLines("C:/Users/alex_/Desktop/ScriptRCanarias/Clima2.txt",encoding="UTF-8")
clima2 = iconv(clima2,to="ASCII//TRANSLIT")
#txt clima3
clima3                                                                          <-
readLines("C:/Users/alex_/Desktop/ScriptRCanarias/Clima3.txt",encoding="UTF-8")
clima3 = iconv(clima3,to="ASCII//TRANSLIT")
#txt clima4
clima4                                                                          <-
readLines("C:/Users/alex_/Desktop/ScriptRCanarias/Clima4.txt",encoding="UTF-8")
clima4 = iconv(clima4,to="ASCII//TRANSLIT")
#txt clima5
clima5                                                                          <-
readLines("C:/Users/alex_/Desktop/ScriptRCanarias/Clima5.txt",encoding="UTF-8")
clima5 = iconv(clima5,to="ASCII//TRANSLIT")

clima <- list(clima1,clima2,clima3,clima4,clima5)
corpus<-Corpus(VectorSource(clima))

d<-tm_map(corpus,content_transformer(tolower))
d<-tm_map(d,stripWhitespace)
d<-tm_map(d,removePunctuation)
d<-tm_map(d,removeNumbers)
d<-tm_map(d,removeWords,c("aao","dos","precipitacian","sharm","dalaman","aunque","canari
as","hacia","estacian","aerop","haciasanta","wwg","daa","izaaa","smn","asa","ello","esta
","cadigo","mas","daas","tac","aoc","rep","nao","salo","taonez","turquaa","omm","cruz",s
topwords("spanish")))

tdm<-TermDocumentMatrix(d)
m=as.matrix(tdm)

wf<-sort(rowSums(m),decreasing = TRUE)
dm<-data.frame(word=names(wf),freq=wf)

wordcloud(dm$word,    dm$freq,    min.freq    =    1,max.words=100,    random.order=FALSE,
colors=brewer.pal(8,"Dark2"))
```

El resultado de la nube de palabras es el siguiente:

Como vemos, encontramos palabras que tiene relación con clima: aire, temperatura, precipitaciones, climáticos, perturbaciones, zonas, invierno, verano, lluvias, húmedas y otras más.

Lugares

En cuanto a los lugares, hemos buscado seis diferentes documentos en donde se destacan los lugares más visitados y deseas de las islas. En este caso, encontramos información muy pareja de todas ellas.

El script realizado en R, es similar al anterior, cambiando claro está los documentos y las palabras que se han suprimido en el resultado de la nube de palabras.

El script es el siguiente:

```
library (tm)
library (wordcloud)
library (lsa)

###FICHEROS DE CLIMA
#txt clima1
Lugares1 <- readLines("C:/Users/Moi/Desktop/Lugares1.txt",encoding="UTF-8")
Lugares1 = iconv(Lugares1,to="ASCII//TRANSLIT")
#txt clima2
Lugares2 <- readLines("C:/Users/Moi/Desktop/Lugares2.txt",encoding="UTF-8")
Lugares2 = iconv(Lugares2,to="ASCII//TRANSLIT")
#txt clima3
Lugares3 <- readLines("C:/Users/Moi/Desktop/Lugares3.txt",encoding="UTF-8")
Lugares3 = iconv(Lugares3,to="ASCII//TRANSLIT")
#txt clima4
Lugares4 <- readLines("C:/Users/Moi/Desktop/Lugares4.txt",encoding="UTF-8")
Lugares4 = iconv(Lugares4,to="ASCII//TRANSLIT")
#txt clima5
Lugares5 <- readLines("C:/Users/Moi/Desktop/Lugares5.txt",encoding="UTF-8")
Lugares5 = iconv(Lugares5,to="ASCII//TRANSLIT")

Lugares6 <- readLines("C:/Users/Moi/Desktop/Lugares6.txt",encoding="UTF-8")
Lugares6 = iconv(Lugares6,to="ASCII//TRANSLIT")

Lugares <- list(Lugares1,Lugares2,Lugares3,Lugares4,Lugares5,Lugares6)
corpus<-Corpus(VectorSource(Lugares))

d<-tm_map(corpus,content_transformer(tolower))
d<-tm_map(d,stripWhitespace)
d<-tm_map(d,removePunctuation)
d<-tm_map(d,removeNumbers)
d<-tm_map(d,removeWords,c("aao","puedes","turastica","wwwtenerifeaccesibleorg","tambiacn
","dos","precipitacian","sharm","dalaman","aunque","canarias","hacia","estacian","aerop"
,"haciasanta","wwg","daa","izaaa","smn","asa","ello","esta","cadigo","mas","daas","tac",
"aoc","rep","nao","salo","taonez","turquaa","omm","cruz",stopwords("spanish")))

tdm<-TermDocumentMatrix(d)
m=as.matrix(tdm)

wf<-sort(rowSums(m),decreasing = TRUE)
dm<-data.frame(word=names(wf),freq=wf)

wordcloud(dm$word,    dm$freq,    min.freq    =    1,max.words=100,    random.order=FALSE,
colors=brewer.pal(8,"Dark2"))
```

El resultado de la nube de palabra es el siguiente:

Como vemos, encontramos lugares destacables como el Teide, iglesias, Adeje, Garachico, playas, senderos o incluso la isla de Tenerife entre otras.

Procedencia

En cuanto a la procedencia, lo que se ha hecho es buscar un documento que agrupara todas las islas canarias y en donde hubieran datos estadísticos acerca de la procedencia de los turistas. El documento obtenido en formato csv contiene datos de procedencia de turista que visitan las islas canarias entre los años 2009 y 2014.

Los resultados obtenidos arrojan que a medida que han ido pasando los años en todas las islas el número de turistas han ido aumentando considerablemente. Esto puede verse en las gráficas obtenidas a partir del script desarrollado con R.

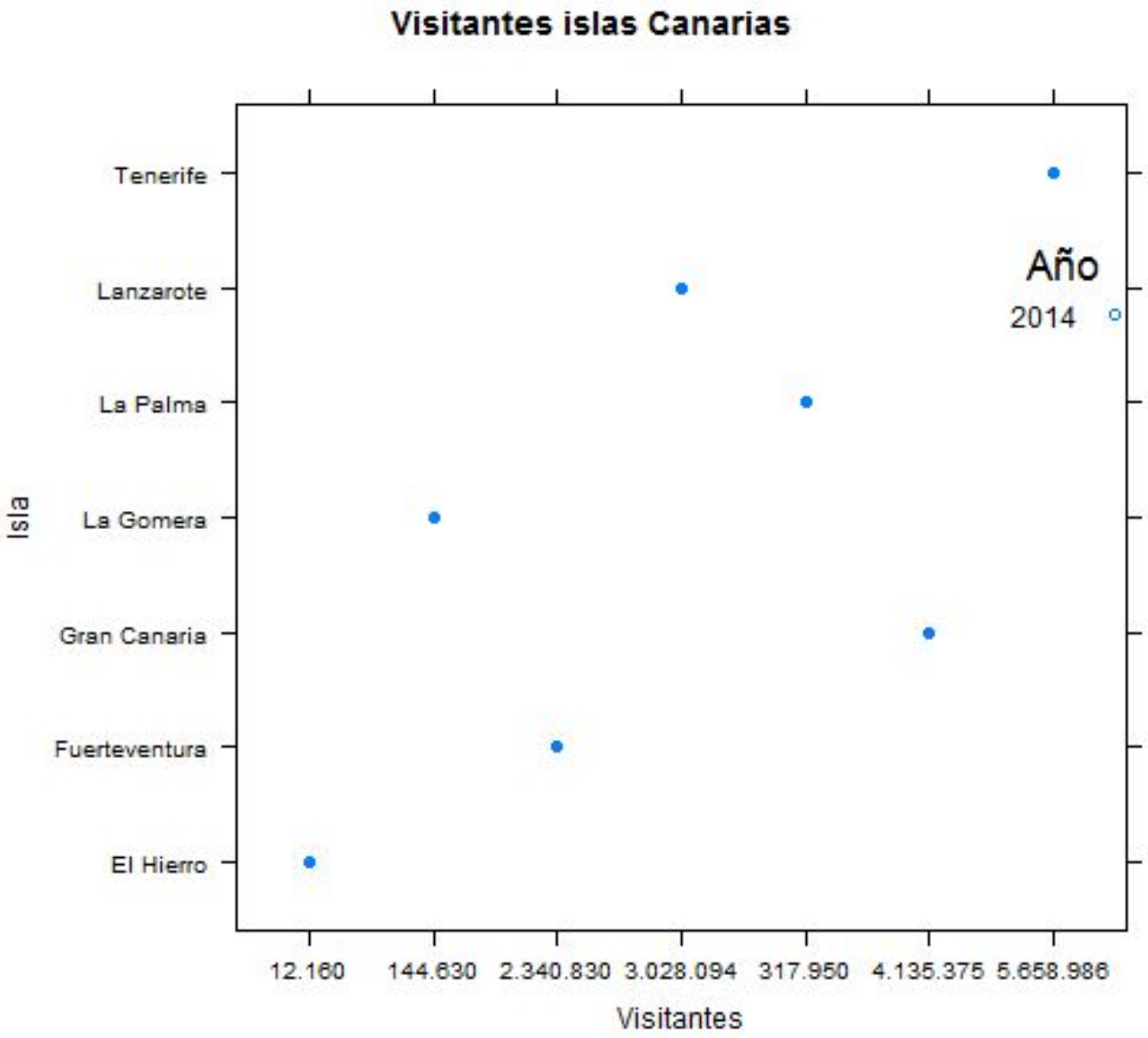
El script realizado en R sería el siguiente:

```
require("lattice")
require("stringr")
raiz <- setwd("~/")
```

```
normalizado <- normalizePath(raiz)
data <-
read.csv(paste(normalizado,"\\proyecto-final-TIO\\datos\\canarias\\visitantes_canaria.cs
v",sep=""),header = TRUE)
mydata <- data[c(1:7),]

plot1 <- xyplot(mydata$NOMBRE ~ mydata$X2014,
       main="Visitantes islas Canarias",type = "p",
       pch = 16 ,auto.key = list(x= 0.85, y=0.85, text= c("2014"),
                                 title="Año"),ylab = "Isla",xlab = "Visitantes")

plot2 <- xyplot(mydata$NOMBRE ~ mydata$X2013,
       main="Visitantes islas Canarias",type = "p",
       pch = 16 ,auto.key = list(x= 0.85, y=0.85, text= c("2013"),
                                 title="Año"),ylab = "Isla",xlab = "Visitantes")

plot3 <- xyplot(mydata$NOMBRE ~ mydata$X2012,
       main="Visitantes islas Canarias",type = "p",
       pch = 16 ,auto.key = list(x= 0.85, y=0.85, text= c("2012"),
                                 title="Año"),ylab = "Isla",xlab = "Visitantes")

plot4 <- xyplot(mydata$NOMBRE ~ mydata$X2011,
       main="Visitantes islas Canarias",type = "p",
       pch = 16 ,auto.key = list(x= 0.85, y=0.85, text= c("2011"),
                                 title="Año"),ylab = "Isla",xlab = "Visitantes")

plot5 <- xyplot(mydata$NOMBRE ~ mydata$X2010,
       main="Visitantes islas Canarias",type = "p",
       pch = 16 ,auto.key = list(x= 0.85, y=0.85, text= c("2010"),
                                 title="Año"),ylab = "Isla",xlab = "Visitantes")

plot6 <- xyplot(mydata$NOMBRE ~ mydata$X2009,
      main="Visitantes islas Canarias",type = "p",
      pch = 16 ,auto.key = list(x= 0.85, y=0.85, text= c("2009"),
                                 title="Año"),ylab = "Isla",xlab = "Visitantes")

for(i in 1:6){
  setwd("~/proyecto-final-TIO/datos/canarias/Graficos")
  archivo <- paste("xyplot",i,sep="")
  extension <- paste(archivo,".png",sep="")
  trellis.device(device="png", filename=extension,height=900,width=1024)
  crea_fichero <- paste("plot",i,sep="")
  plot=eval(parse(text=crea_fichero))
  print(plot)
  dev.off()}
```

El resultado de ejecutar dicho script sería el siguiente para el año 2009:

El resultado de ejecutar dicho script sería el siguiente para el año 2014:



2. Londres

Para la realización del análisis de procesamiento del lenguaje natural de la ciudad de Londres, se ha realizado una búsqueda exhaustiva sobre los datos más relevantes que se van a tratar en este proyecto. Estos datos son:

- **El clima**: para el procesamiento del lenguaje natural en este campo lo que se intenta obtener es el conjunto de palabras más representativa con respecto al clima de la ciudad de Londres.
- **Lugares**: con el análisis de los datos lugares se pretende obtener los lugares más visitados o más destacados de la ciudad de Londres.
- **Procedencia**: para el análisis de los datos referentes a la procedencia, se realiza un análisis estadístico de los turistas que más visitan la ciudad de Londres ordenados por país de procedencia.

Clima

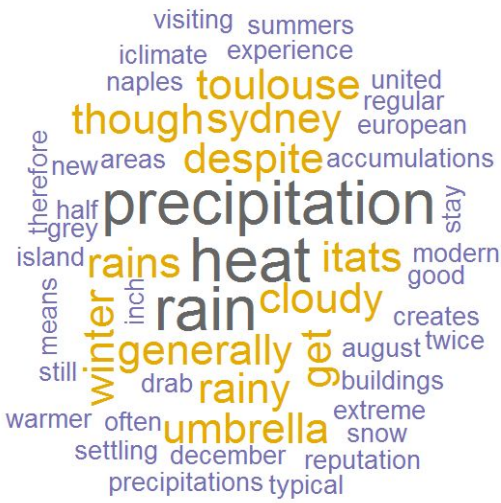
Para el tratamiento de los datos del clima se han encontrado 2 ficheros diferentes denominados “Clima2.txt” y “Clima2.txt”, en estos ficheros se encuentra información relevante con respecto al clima en la ciudad de Londres, la temporada donde hace mejor o peor tiempo,etc.

Para procesar los datos, se ha realizado el script “**ClimaLondres.R**”. Este script carga los dos ficheros .txt anteriormente indicados. Con estos ficheros, se realiza un corpus mediante el cual se realizará una serie de operaciones para limpiar el corpus anteriormente creado. El script resultante es:

```
library (tm)
library (wordcloud)
library (lsa)

#Cargamos ficheros
raiz <- setwd("~/")
texto1 <- readLines(paste(raiz,"/proyecto-final-TIO/datos/londres/Clima1.txt",sep =
""),encoding="UTF-8")
```

```
texto1 = iconv(texto1, to="ASCII//TRANSLIT")

texto2 <- readLines(paste(raiz,"/proyecto-final-TIO/datos/londres/Clima2.txt",sep =
""),encoding="UTF-8")
texto2 = iconv(texto2, to="ASCII//TRANSLIT")

#Unimos todos los ficheros bajo una única lista y creamos el corpus
docs <- list(texto1,texto2)
corpus <- Corpus(VectorSource(docs))

#Limpiamos el conjunto de los documentos de numeros, espacios en blanco, etc...
d <- tm_map(corpus, content_transformer(tolower)) #Lo convierte a minúsculas
d <- tm_map(d, removePunctuation) #Elimina signos de puntuación
d <- tm_map(d, removeNumbers) #Elimina los números
d <- tm_map(d, stripWhitespace) #Elimina los espacios en blanco
d <- tm_map(d, removeWords,
c("london","weather","climate","temperatures","temperature","year","months",
"usually","although","less","day","city","cities","likely","capital","normally",
"always","however","sometimes","throughout","idea","common",stopwords("english")))
#Elimina palabras vacías en inglés

#Creamos la matriz de términos
tdm <- TermDocumentMatrix(d)
m = as.matrix(tdm)

# conteo de palabras en orden decreciente
wf <- sort(rowSums(m),decreasing=TRUE)
# crea un data frame con las palabras y sus frecuencias
dm <- data.frame(word = names(wf), freq=wf)
# Frecuencia minima igual a 20

wordcloud(dm$word, dm$freq, min.freq = 1,
          max.words=50, random.order=FALSE, colors=brewer.pal(8,"Dark2"))
```

El resultado de la nube de palabras es el siguiente



Lugares

Para el tratamiento de los datos de los lugares se han encontrado 3 ficheros diferentes denominados “Lugares1.txt” , ”Lugares2.txt”, “Lugares3.txt”, en estos documentos aparecen los lugares más representativos y más visitados en la ciudad de Londres.

Para procesar los datos, se ha realizado el script “**script_Londres_lugares.R**”. Este script carga los dos ficheros .txt anteriormente indicados. Con estos ficheros, se realiza un corpus mediante el cual se realizará una serie de operaciones para limpiar el corpus anteriormente creado. El script resultante es:

```
library (tm)
library (wordcloud)
library (lsa)

#Cargamos ficheros
raiz <- setwd("~/")
texto1 <- readLines(paste(raiz,"/proyecto-final-TIO/datos/londres/Lugares1.txt",sep =
""),encoding="UTF-8")
texto1 = iconv(texto1, to="ASCII//TRANSLIT")

texto2 <- readLines(paste(raiz,"/proyecto-final-TIO/datos/londres/Lugares2.txt",sep =
""),encoding="UTF-8")
texto2 = iconv(texto2, to="ASCII//TRANSLIT")

texto3 <- readLines(paste(raiz,"/proyecto-final-TIO/datos/londres/Lugares3.txt",sep =
""),encoding="UTF-8")
texto3 = iconv(texto2, to="ASCII//TRANSLIT")
#Unimos todos los ficheros bajo una única lista y creamos el corpus
docs <- list(texto1,texto2,texto3)
corpus <- Corpus(VectorSource(docs))

#Limpiamos el conjunto de los documentos de numeros, espacios en blanco, etc...
d <- tm_map(corpus, content_transformer(tolower)) #Lo convierte a minúsculas
d <- tm_map(d, removePunctuation) #Elimina signos de puntuación
d <- tm_map(d, removeNumbers) #Elimina los números
d <- tm_map(d, stripWhitespace) #Elimina los espacios en blanco
d<-tm_map(d, removeWords,
c("london","londons","abbey","cast","count","end","fine","inside","mall","new","plan","r
ich","sir","stay","top","war","aboard","bell","child","cutty","entry","full","high","jus
t","map","now","port","roles","speech","tour","youll","years","yearold","years","written
","cocacola","share","visit","free","exploring","views","tickets","one","time","also","k
nown","see","pauls","two","take","big","home",stopwords("english"))) #Elimina palabras
vacías en inglés
```

```
#Creamos la matriz de términos
tdm <- TermDocumentMatrix(d)
m = as.matrix(tdm)

# conteo de palabras en orden decreciente
wf <- sort(rowSums(m),decreasing=TRUE)
# crea un data frame con las palabras y sus frecuencias
dm <- data.frame(word = names(wf), freq=wf)
# Frecuencia minima igual a 20

counts <- table (dm$word)

wordcloud(dm$word, dm$freq, min.freq = 1,
          max.words=50, random.order=FALSE, colors=brewer.pal(8,"Dark2"))
```

El resultado final de la nube de palabras es el siguiente:

Procedencia

Para el tratamiento de los datos de los turistas procedentes de los diferentes países, se ha utilizado un documento denominado "**visitantes.csv**". Este documento posee un análisis estadístico de los turistas que han visitado la ciudad de londres dividida por países de procedencia entre el año 2002 y el año 2015.

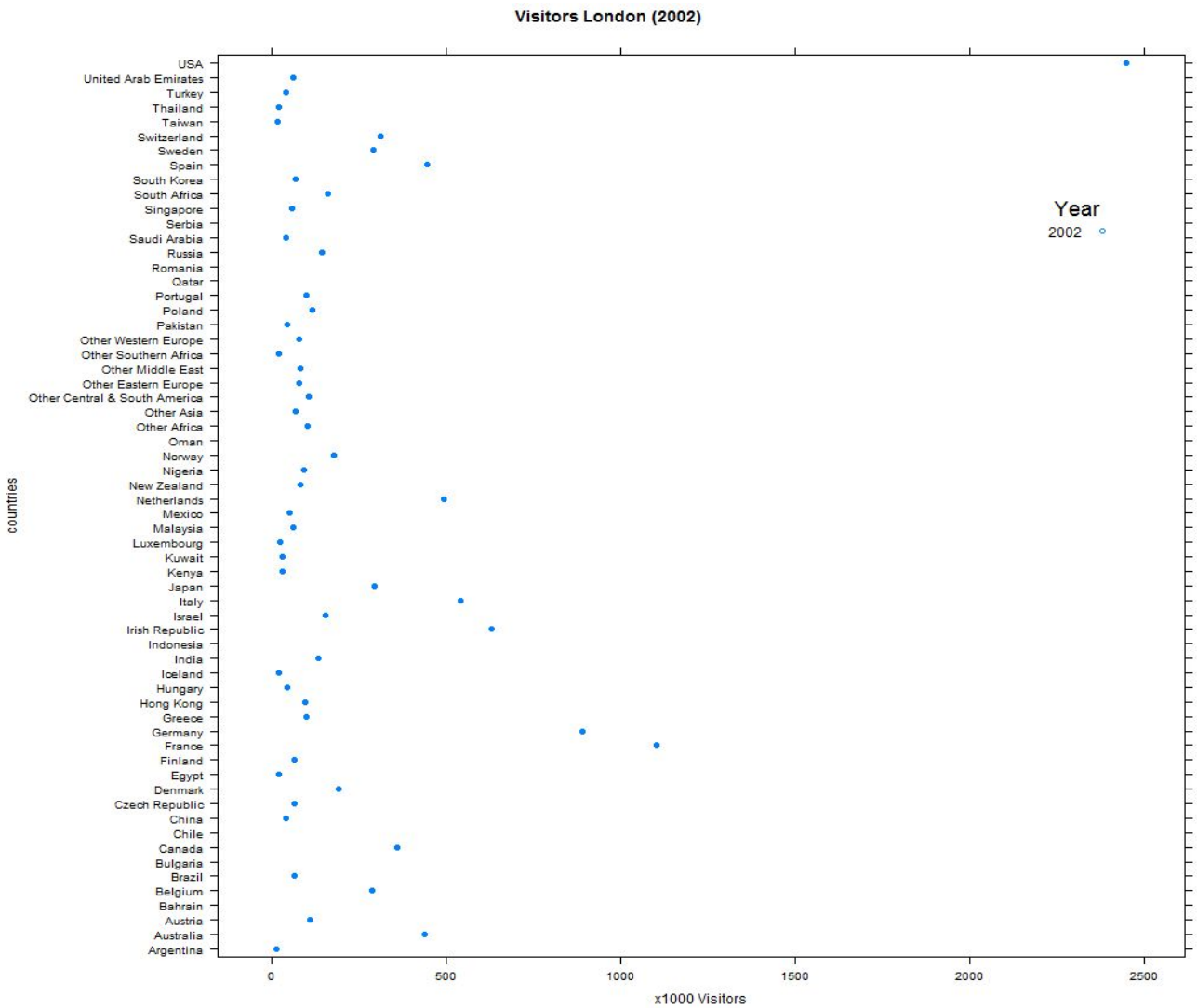
Para procesar los datos, se ha realizado el script "**script_graficosLondres.R**". Este script carga el fichero csv anteriormente indicado y realiza una serie de operaciones para generar un gráfico multivariable mediante la utilización de la función xyplot(). El script resultante es el siguiente:

```
require("lattice")
require("stringr")
raiz <- setwd("~/")
normalizado <- normalizePath(raiz)
data                                                                    <-
read.csv(paste(normalizado,"\\proyecto-final-TIO\\datos\\londres\\visitantes.csv",sep=""
),header = TRUE)
mydata <- data[c(1:62),]
plot1 <- xyplot(mydata$Country.of.origin ~ mydata$X2002,
       main="Visitors London (2002)",type = "p",
       pch = 16 ,auto.key = list(x= 0.85, y=0.85, text= c("2002"),
                                          title="Year"),ylab = "countries",xlab = "x1000
Visitors")

plot2 <- xyplot(mydata$Country.of.origin ~ mydata$X2003,
       main="Visitors London (2003)",type = "p",
       pch = 16 ,auto.key = list(x= 0.80, y=0.85, text= c("2003"),
                                          title="Year"),ylab = "countries",xlab = "x1000
Visitors")

plot3 <- xyplot(mydata$Country.of.origin ~ mydata$X2004,
                main="Visitors London (2004)",type = "p",
                pch = 16 ,auto.key = list(x= 0.80, y=0.85, text= c("2004"),
                                           title="Year"),ylab = "countries",xlab = "x1000
Visitors")

plot4 <- xyplot(mydata$Country.of.origin ~ mydata$X2005,
                main="Visitors London (2005)",type = "p",
                pch = 16 ,auto.key = list(x= 0.80, y=0.85, text= c("2005"),
                                           title="Year"),ylab = "countries",xlab = "x1000
Visitors")

plot5 <- xyplot(mydata$Country.of.origin ~ mydata$X2006,
```

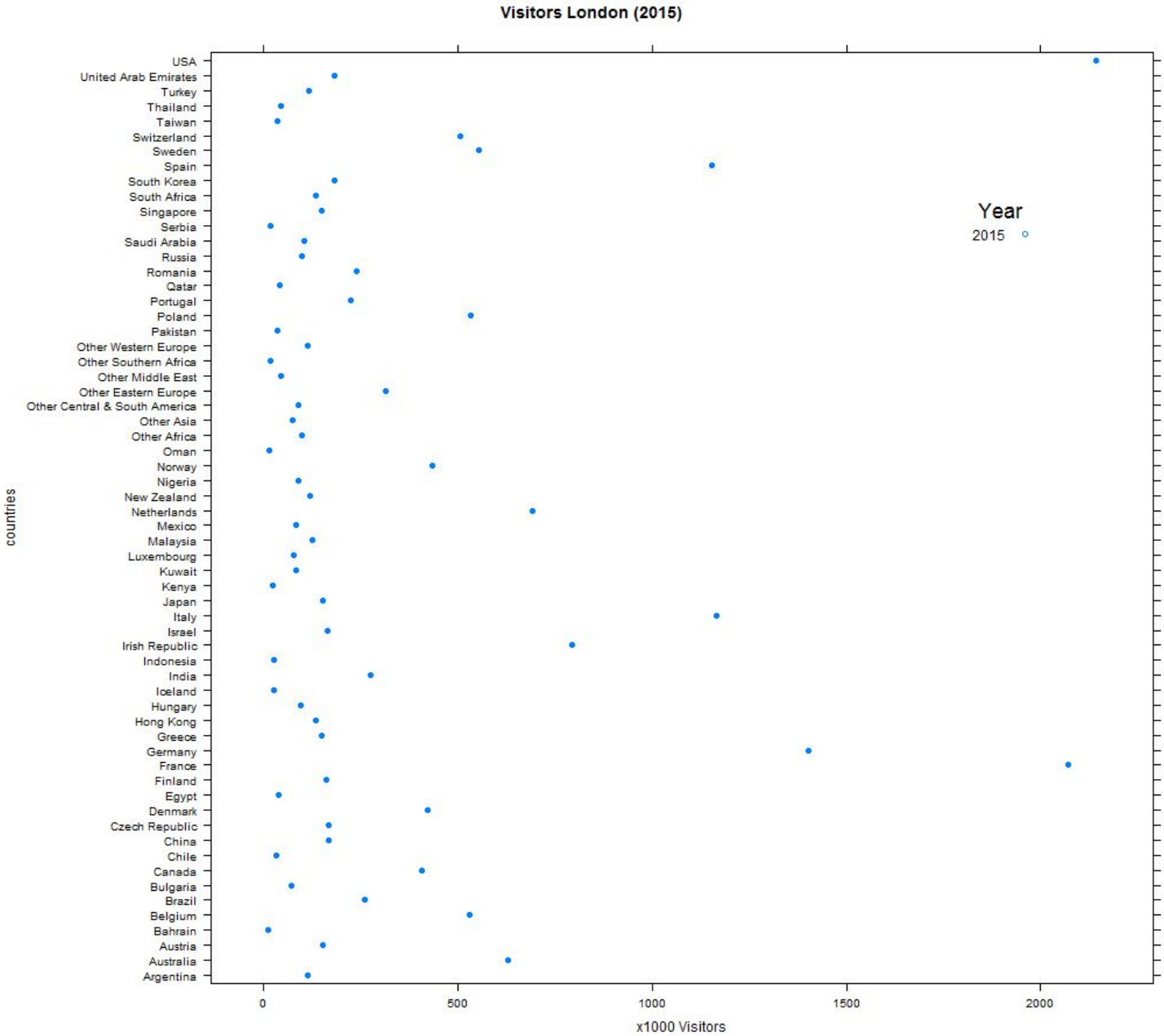
```
                main="Visitors London (2006)",type = "p",
                pch = 16 ,auto.key = list(x= 0.80, y=0.85, text= c("2006"),
                                          title="Year"),ylab = "countries",xlab = "x1000
Visitors")

plot6 <- xyplot(mydata$Country.of.origin ~ mydata$X2007,
                main="Visitors London (2007)",type = "p",
                pch = 16 ,auto.key = list(x= 0.80, y=0.85, text= c("2007"),
                                          title="Year"),ylab = "countries",xlab = "x1000
Visitors")

plot7 <- xyplot(mydata$Country.of.origin ~ mydata$X2008,
                main="Visitors London (2008)",type = "p",
                pch = 16 ,auto.key = list(x= 0.80, y=0.85, text= c("2008"),
                                          title="Year"),ylab = "countries",xlab = "x1000
Visitors")


plot8 <- xyplot(mydata$Country.of.origin ~ mydata$X2009,
                main="Visitors London (2009)",type = "p",
                pch = 16 ,auto.key = list(x= 0.80, y=0.85, text= c("2009"),
                                          title="Year"),ylab = "countries",xlab = "x1000
Visitors")

plot9 <- xyplot(mydata$Country.of.origin ~ mydata$X2010,
                main="Visitors London (2010)",type = "p",
                pch = 16 ,auto.key = list(x= 0.80, y=0.85, text= c("2010"),
                                          title="Year"),ylab = "countries",xlab = "x1000
Visitors")

plot10 <- xyplot(mydata$Country.of.origin ~ mydata$X2011,
                main="Visitors London (2011)",type = "p",
                pch = 16 ,auto.key = list(x= 0.80, y=0.85, text= c("2011"),
                                          title="Year"),ylab = "countries",xlab = "x1000
Visitors")

plot11 <- xyplot(mydata$Country.of.origin ~ mydata$X2012,
                main="Visitors London (2012)",type = "p",
                pch = 16 ,auto.key = list(x= 0.80, y=0.85, text= c("2012"),
                                          title="Year"),ylab = "countries",xlab = "x1000
Visitors")

plot12 <- xyplot(mydata$Country.of.origin ~ mydata$X2013,
                main="Visitors London (2013)",type = "p",
                pch = 16 ,auto.key = list(x= 0.80, y=0.85, text= c("2013"),
                                          title="Year"),ylab = "countries",xlab = "x1000
Visitors")

plot13 <- xyplot(mydata$Country.of.origin ~ mydata$X2014,
                main="Visitors London (2014)",type = "p",
                pch = 16 ,auto.key = list(x= 0.80, y=0.85, text= c("2014"),
                                          title="Year"),ylab = "countries",xlab = "x1000
```

```
Visitors")

plot14 <- xyplot(mydata$Country.of.origin ~ mydata$X2015,
                 main="Visitors London (2015)",type = "p",
                 pch = 16 ,auto.key = list(x= 0.80, y=0.85, text= c("2015"),
                                           title="Year"),ylab = "countries",xlab =
"x1000 Visitors")

for(i in 1:14){
  setwd("~/proyecto-final-TIO/datos/londres/Graficos")
  archivo <- paste("xyplot",i,sep="")
  extension <- paste(archivo,".png",sep="")
  trellis.device(device="png", filename=extension,height=900,width=1024)
  crea_fichero <- paste("plot",i,sep="")
  plot=eval(parse(text=crea_fichero))
  print(plot)
  dev.off()}
```

El resultado de uno de los gráficos para el año 2002 es el siguiente:



En el caso del año 2015, el resultado sería el siguiente:



3. Nueva York

Para la realización del análisis de procesamiento del lenguaje natural de Nueva York, se ha realizado una búsqueda exhaustiva sobre los datos que se van a tratar. Posteriormente los hemos analizado con R y hemos logrado obtener algunos conocimientos relevantes. Los veremos en los tres siguientes apartados.

Clima

Este script es básicamente una modificación del script que hemos realizado en las prácticas. Gracias a él, podemos ver cuales son las palabras de mayor relevancia en el documento. Lo esperable, es que tratándose de documentos relacionados con el clima, se hable de palabras como “sol” o “lluvia”.

```
library(tm)
library(lsa)
library(wordcloud)

rutaBase <- "D:\\Documentos\\Repos\\Clases\\tio\\proyecto"
ciudad <- "nuevayork"

getFile <- function (path) {
  file <- readLines(path, encoding="UTF-8")
  file = iconv(file, to="ASCII//TRANSLIT")
  return (file)}

join <- function (list) {
  return (paste(list, collapse = ''))}

getfiles <- function (rutaBase, ciudad, cantidad) {
  algo <- ""
  data <- 1:cantidad

  for (i in 1:cantidad) {
    rutaFinal <- paste(rutaBase, "/datos/", ciudad ,"/Clima", i ,".txt", sep="")
    tempData <- getFile(rutaFinal)
    data[i] = join(tempData)  }

  return (join(data))}

cleanCorpus <- function (corpus) {
  d <- tm_map(corpus, content_transformer(tolower))
  d <- tm_map(d, stripWhitespace)
  d <- tm_map(d, removePunctuation)
  d <- tm_map(d, removeWords, stopwords('english'))
   d <- tm_map(d, removeWords, c("will", "weathermodification", "also", "2025", "may",
```

```
"carbon", "systems", "however", "paper", "enemy", "aircraft", "can", "space", "new",
"operations", "week", "radio", "study", "natural"))
  return (d)}

data <- getfiles(rutaBase, ciudad, 2)
corpus <- Corpus(VectorSource(data))
d <- cleanCorpus(corpus)
tdm <- TermDocumentMatrix(d)
m = as.matrix(tdm)
wf <- sort(rowSums(m), decreasing=TRUE)
dm <- data.frame(word = names(wf), freq=wf)
findFreqTerms(tdm, lowfreq=20)
wordcloud(dm$word,   dm$freq,   min.freq   =   1,   max.words=50,   random.order=FALSE,
colors=brewer.pal(8,"Dark2"))
```

Al obtener la nube de palabras, podemos ver cómo la palabra más usada es “clima”, lo cual tiene sentido ya que los documentos hablan de eso precisamente, pero no nos es de mucha relevancia. De esta nube, podríamos sacar “fog”, “air”, “storm” y “cloud”. Se podría deducir que en Nueva York predomina un tiempo nublado y lluvioso la mayor parte del año.

Lugares

Para realizar el análisis de los lugares más visitados y característicos de Nueva York se han encontrado 5 documentos distintos. Para procesar los datos, se han cargado los ficheros .txt en el script "script_NYC_lugares.R". Después se han realizado unas transformaciones para eliminar datos irrelevantes. El script resultante es:

```
library (tm)
library (wordcloud)
library (lsa)

#Cargamos ficheros
texto1 <-
readLines("C:/Users/raul_/Desktop/proyecto-final-TIO-master/datos/nuevayork/Lugares1.txt
",encoding="UTF-8")
texto1 = iconv(texto1, to="ASCII//TRANSLIT")

texto2 <-
readLines("C:/Users/raul_/Desktop/proyecto-final-TIO-master/datos/nuevayork/Lugares2.txt
",encoding="UTF-8")
texto2 = iconv(texto2, to="ASCII//TRANSLIT")

texto3 <-
readLines("C:/Users/raul_/Desktop/proyecto-final-TIO-master/datos/nuevayork/Lugares3.txt
",encoding="UTF-8")
texto3 = iconv(texto2, to="ASCII//TRANSLIT")

texto4 <-
readLines("C:/Users/raul_/Desktop/proyecto-final-TIO-master/datos/nuevayork/Lugares4.txt
",encoding="UTF-8")
texto4 = iconv(texto2, to="ASCII//TRANSLIT")

texto5 <-
readLines("C:/Users/raul_/Desktop/proyecto-final-TIO-master/datos/nuevayork/Lugares5.txt
",encoding="UTF-8")
texto5 = iconv(texto2, to="ASCII//TRANSLIT")

#Unimos todos los ficheros bajo una única lista y creamos el corpus
docs <- list(texto1,texto2,texto3,texto4,texto5)
corpus <- Corpus(VectorSource(docs))

#Limpiamos el conjunto de los documentos de numeros, espacios en blanco, etc...
d <- tm_map(corpus, content_transformer(tolower)) #Lo convierte a minúsculas
d <- tm_map(d, removePunctuation) #Elimina signos de puntuación
d <- tm_map(d, removeNumbers) #Elimina los números
d <- tm_map(d, stripWhitespace) #Elimina los espacios en blanco
d <- tm_map(d, removeWords,
c("tel","top","world","ave","internationalist","one","experience","want","see","good","g
uide","wwwinternationalistcom","take","two","enjoy","youatll","may","itats","just","atmo
sphere","like","cusine","make","great","free","always","gps","many","also","place","admi
ssion","phone","get","best","grand","tickets","show","donatt","check","open","will","vis
```

```
it","day","activities","new","york","room","can","hours","closed","things","food","city"
,stopwords("english"))) #Elimina palabras vacías en inglés

#Creamos la matriz de términos
tdm <- TermDocumentMatrix(d)
m = as.matrix(tdm)

# conteo de palabras en orden decreciente
wf <- sort(rowSums(m),decreasing=TRUE)
# crea un data frame con las palabras y sus frecuencias
dm <- data.frame(word = names(wf), freq=wf)
# Frecuencia minima igual a 20

wordcloud(dm$word, dm$freq, min.freq = 1,
          max.words=50, random.order=FALSE, colors=brewer.pal(8,"Dark2"))
```

El resultado de la nube de palabras es el siguiente:

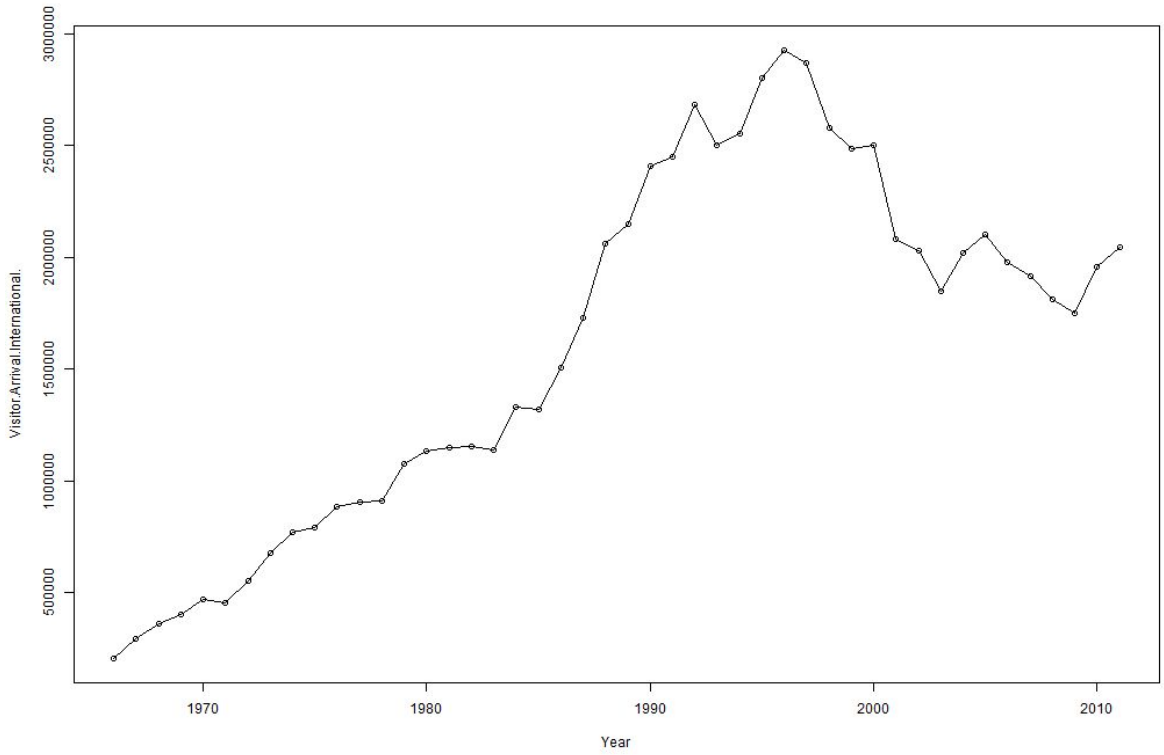
Procedencia

En este apartado analizamos quiénes y cómo llegan a la ciudad de Nueva York. Para ello hemos usado un fichero CSV obtenido desde la fuente de openData del gobierno de EEUU, en data.gov. Este script lo hemos tenido que modificar ligeramente respecto a los a los anteriores, ya que en nuestro caso no tenemos una imagen por cada

año, sino una imagen por cada tipo de dato, es decir, una para el total de visitantes, otra para visitantes extranjeros, etc.

En el script se lee el fichero CSV, y se crea una imagen por cada columna, sin contar con la columna de los años.

```
# Poner la ruta de donde está el repo
rutaBase <- "D:\\Documentos\\Repos\\Clases\\tio\\proyecto"

rutaCSV <- paste(rutaBase, "/datos/nuevayork/visitantes.csv", sep="")
rutaOutput <- paste(rutaBase, "/datos/nuevayork/Graficos/", sep="")

data <- read.csv(rutaCSV, header=TRUE)

for(i in 2:ncol(data)) {
  visitantes <- data[, c(1, i)]

  png(filename=paste(rutaOutput, i-1, ".png"),
      units="px",
      width=1024,
      height=700,
      pointsize=12,
      res=72)
  plot(visitantes,type="o")
  dev.off()
}
```



Una de las imágenes más interesantes es la número 3, que son las llegadas procedentes de otros países. Se puede observar cómo crece hasta alrededor de 1996, que empieza a caer al mismo ritmo con el que crecía antes. Una posible conclusión de esto, es que con el comienzo de la crisis la gente ha realizado menos viajes a Nueva York, y eso posible que hayan elegido un destino más económico.

4. Enlaces documentos

- Londres
  - Lugares:
    - http://travel.usnews.com/London_England/Things_To_Do/
    - http://www.visitlondon.com/things-to-do/sightseeing/london-attraction/top-ten-attractions
    - http://www.planetware.com/tourist-attractions-/london-eng-l-lon.htm
  - Clima:
    - https://www.londoncitybreak.com/climate
    - https://en.wikipedia.org/wiki/Climate_of_London
  - Procedencia (hasta 2015):
    - https://files.datapress.com/london/dataset/number-international-visitors-london/2016-07-25T10:33:39/international-visitors-london-raw.csv
    - http://files.londonandpartners.com/l-and-p/assets/our-insight-london-tourism-review-2014-15.pdf
- Nueva York
  - Lugares:
    - http://www.gocapetravel.com/resources/Top10GuidetoNewYorkCity.pdf
    - http://guides.tripomatic.com/download/tripomatic-free-city-guide-new-york-city.pdf
    - http://expediablog.co.uk/Expedia-New-York-Pocket-Guide.pdf
    - http://uk.complex.com/pop-culture/2013/06/best-summer-dates-in-nyc/fire-guns-at-west-side-rifle-and-pistol-range-and-shoot-mezcal-preferably-in-that-order
    - http://www.americaasyoulikeit.com/Public/Assets/User/files/brochures/AAYLI%20NY%20State%20brochure%20WEB.pdf
  - Clima:
    - https://business.weather.com/writable/documents/Retail/Holiday-eBook-print_FINAL.pdf
    - http://csat.au.af.mil/2025/volume3/vol3ch15.pdf
  - Procedencia
    - http://www.nytimes.com/2016/03/09/nyregion/record-number-of-tourists-visited-new-york-city-in-2015-and-more-are-expected-this-year.html

        - http://www.reuters.com/article/us-usa-newyork-tourism-idUSKBN0L61XM20150202
- Canarias
    - Lugares:
        - http://www.canarias.com/blog/curiosidades-de-canarias/?print=pdf
        - http://www.webtenerife.com/es/galeria-multimedia/folletos/lists/galeriafolletos/folleto%20tenerife%20en%20coche.pdf?iframe=true
        - http://www.todotenerife.es/assets/downloads/db1a973947.pdf
        - http://www.laguiadegrancanaria.com/archivos/guia_turistica_gran_canaria.pdf
        - http://senderosdelapalma.es/wp-content/uploads/mapasenderos.pdf
        - http://www.reservoirbirds.com/TripReports/RBTR_000002.pdf
    - Clima:
        - http://www.gobiernodecanarias.org/educacion/general/gestorglobal/DocsUp/parrafos/5324UD%202%20-%20Clima%20y%20vegetaci%C3%B3n%20de%20Canarias.pdf
        - http://www.elmejorclimadelmundo.com/files/estudioenelanuariodeestudiosatlanticos.pdf
        - http://www.divulgameteo.es/uploads/Caracter%C3%ADsticas-clima-Canarias.pdf
        - http://www.elmejorclimadelmundo.com/files/informedelauniversidaddelalaguna.pdf
        - http://editorial.dca.ulpgc.es/ftp/icaro/Anexos/2-%20CALOR/2-Clima/C.6.2-1%20Islas%20Canarias-Rasgos%20climaticos%20generales-INM.pdf
    - Procedencias:
        - http://www.ccelpa.org/informe-anual/IA2013/2013/09-2013.pdf
        - http://www.webtenerife.com/es/investigacion/situacion-turistica/informes-situacion-turistica/documents/balance%20de%20situaci%C3%B3n%20tur%C3%ADstica%202015.pdf