



# MEMORIA DE PRÁCTICAS

Jaime Sánchez Escudero  
Grado en Física  
Facultad de Ciencias  
UNIVERSIDAD DE VALLADOLID

# 1. OBJETIVO

El objetivo principal ha consistido en elaborar una base de datos, obtenida a partir de varias estaciones terrestres con el fin de obtener un mapa de climatología y ver periodicidades en los cambios del espesor óptico para aerosoles de 550nm de grosor a lo largo de un período de tiempo aproximado de diez años, permitiendo posteriormente comparar esos datos y periodicidades con los obtenidos por satélite para la misma zona.

# 2. DESARROLLO

## 2.1. Obtención de los datos

Para la creación de la base de datos (de la que se extraerá información posteriormente), ha sido necesario el manejo de dos programas, uno con el que descargaremos los datos de las diferentes estaciones para los rangos de tiempo deseados y también 'Rstudio', siendo éste último donde se realizará la mayor y más importante labor de organización y tratamiento de datos.

El primero de ellos, llamado "Aeronet Download Tool" (ADT) (figura 5), consiste en una interfaz simple donde podemos obtener los datos de las estaciones elegidas, estableciendo previamente unos ajustes que permitirán filtrar estos datos para únicamente descargar los deseados.

Para este estudio concreto, se toma todo el rango de fechas que comprende desde el 1 de enero de 2010 hasta el 31 de diciembre de 2020 con los datos diarios promedio, los cuales fueron medidos de día y con tipo de dato 'AOD20' (figura 5).

Las diferentes estaciones de las que han sido extraídos datos han sido "Agia Marina", "ÇUT-TEPAK", "İMS-METU-ERDEMLI", "Nicosiaz", "Pafos", situadas todas ellas en la isla de Chipre, a excepción de 'IMS', que se sitúa sobre la costa turca (figura 1).

Usando el programa 'Rstudio' (figura 6), mencionado anteriormente (Rstudio es un entorno de desarrollo que usa como lenguaje de programación R), podemos situar mediante coordenadas geográficas las diferentes estaciones utilizadas sobre un mapa, para conocer así su localización exacta.

Esto se puede lograr mediante el siguiente script:

```
library("maps")

Vcolor <- c('darkred', 'red', 'black', 'darkorange', 'blueviolet',
           'blue', 'dodgerblue', 'deeppink', 'forestgreen')
Vsim <- c(15,16,5,17,21,22,23,24,25,1,2,3,4,5)

fileOUT <- 'C:/Users/Usuario/Desktop/mapazoom.jpeg'
jpeg(fileOUT, width = 840, height = 720, quality = 100, bg='white')
fileAERONET<-paste('C:/Users/Usuario/Desktop/aeronet_locations.txt',
                  sep='')
aeronet <- read.table(fileAERONET,skip=1,header=T,sep=',')
```

```

sites <- c('AgiaMarina_Xyliatou', 'Pafos', 'IMS-METU-ERDEMLI',
          'CUT-TEPAK', 'Nicosia')

longitude <- rep(-360,length=length(sites))
latitude <- rep(-90,length=length(sites))

for(ns in 1:length(sites)){
  indSI <- which(sites[ns]==aeronet$Site_Name)
  lonSI <- aeronet[indSI,]$Longitude.decimal_degrees.
  latSI <- aeronet[indSI,]$Latitude.decimal_degrees.
  longitude[ns] <- lonSI
  latitude[ns] <- latSI
}

map(lwd=2,xlim=c(25,40),ylim=c(30,40),col='gray70')
points(longitude,latitude,type='p',pch=Vsim,lwd=2,col=Vcolor,
       cex=2,panel.first=grid(col='gray50'),xaxs='i',yaxs='i')
legend('bottomleft',sites,pch=Vsim,bty='n',cex = 1.8, col=Vcolor,
      text.col=Vcolor,text.font=1.5)

axis(1,at=seq(25,40,5))
axis(2,at=seq(30,40,5))
axis(3,at=seq(25,40,5),labels=NA)
axis(4,at=seq(30,40,5),labels=NA)

mtext(side=1,'Longitude (°E)',font=2,line=2)
mtext(side=2,'Latitude (°N)',font=2,line=2)

dev.off()

```

---

El funcionamiento del código es simple: Primero crea el archivo .jpeg (un formato de imagen) sobre el que después se plasmará el mapa y lee el archivo de texto 'aeronet-locations.txt' (figura 7), que contiene las coordenadas geográficas de todas las estaciones de la red Aeronet (Aerosol Robotic Network). Como sólo interesarán las cinco antes mencionadas, definimos un vector que contenga el nombre de éstas, de forma que el bucle principal del código obtenga únicamente las coordenadas de las estaciones buscadas. Una vez se obtiene esto, usando la librería 'maps', es posible generar un mapa y representar puntos sobre él con coordenadas geográficas. Entonces, el programa sobrescribirá el archivo de imagen creado al principio por el mapa con los puntos de las estaciones graficados, dando como salida el archivo siguiente:

Con estas cinco estaciones será posible visualizar posteriormente, para esa zona del mediterráneo, los cambios en el espesor óptico (AOD) para un aerosol de 550nm de grosor, así como su posible procedencia.

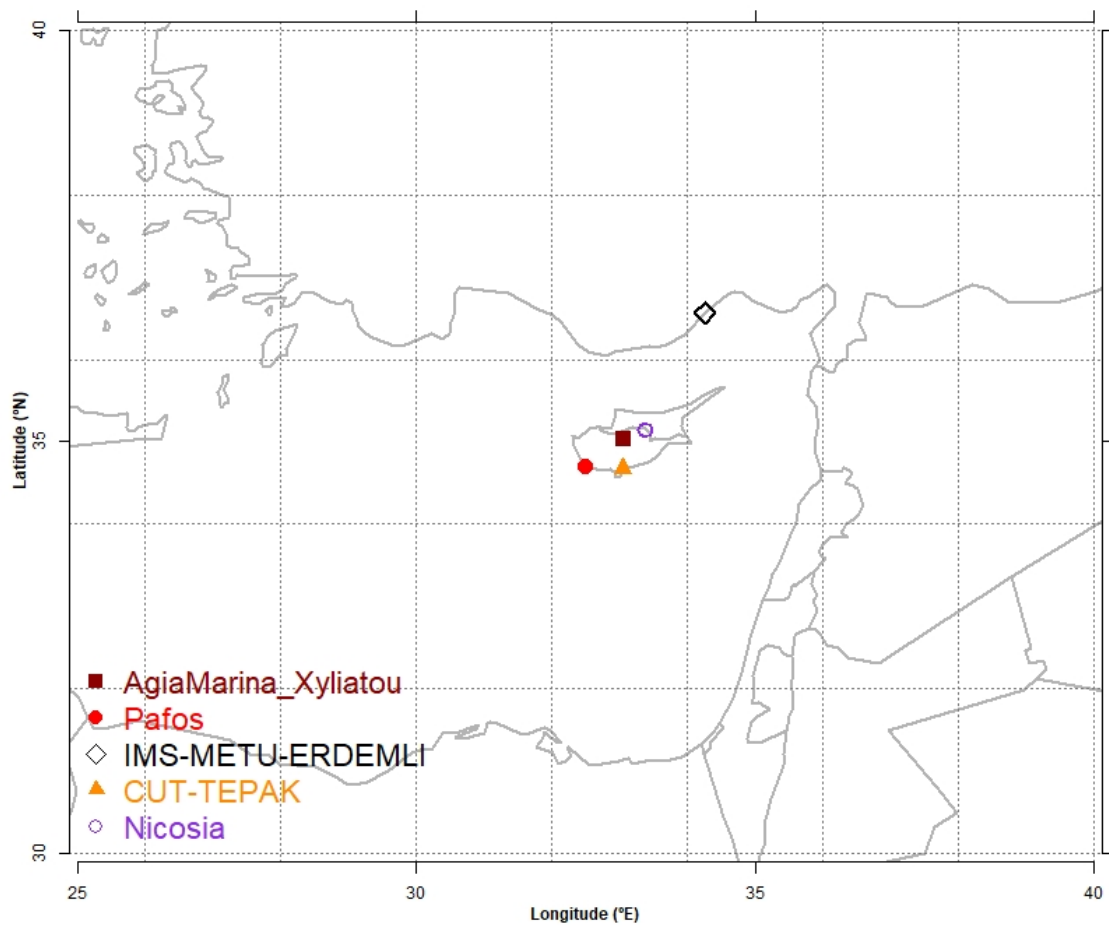


Figura 1: Localización de las diferentes estaciones.

Cabe mencionar que, en un primer momento, la estación 'IMS-METU-ERDEMLI' no entraba dentro del estudio por no situarse sobre la isla de Chipre. No obstante, tras estudiar su cercanía a la isla y dado que solucionaba una importante ausencia de datos entre 2013 y mediados de 2014 (principalmente), se decidió incluir para la generación de la base de datos.

## 2.2. Procesado de los datos

Una vez descargados los datos necesarios en formato '.csv' (archivo compatible con el programa 'microsoft excel' donde la información vendrá separada únicamente por comas (figura 8)) se hace necesario el tratamiento de estos archivos para organizar los nombres de las variables y sus respectivos valores.

Para esto, se ha elaborado el siguiente código en 'R' llamado 'fechas', que será el que permita tratar los datos en bruto de las 5 estaciones:

```
Agia <- 'C:/Users/Usuario/Desktop/Agia-Marina.csv'
CUT <- 'C:/Users/Usuario/Desktop/CUT-TEPAK.csv'
```

```

Pafos <- 'C:/Users/Usuario/Desktop/Pafos.csv'
Nicos <- 'C:/Users/Usuario/Desktop/Nicosia.csv'
IMS <- 'C:/Users/Usuario/Desktop/IMS-METU-ERDEMLI.csv'
fechas <- read.csv('C:/Users/Usuario/Desktop/550nm_completa.csv',
                  header = T, skip=0, sep=',', fill=T,
                  stringsAsFactors = FALSE) #lectura fechas

fechas[,c(2:14)] <- NA #generacion de columnas
colnames(fechas) <- c('fecha', 'Agia-Marina', 'CUT-TEPAK', 'Pafos',
                    'Nicosia', 'IMS-METU-ERDEMLI', 'exp. Agia',
                    'exp. CUT', 'exp. Pafos', 'exp. Nicos',
                    'exp. IMS', 'Media para 440nm',
                    'Media Exponente', '550nm')

#lectura csv####
Agia <- read.csv(Agia, header = T, skip=5, sep=',', fill=T,
                stringsAsFactors = FALSE)
CUT <- read.csv(CUT, header = T, skip=5, sep=',', fill=T,
                stringsAsFactors = FALSE)
Nicos <- read.csv(Nicos, header = T, skip=5, sep=',', fill=T,
                 stringsAsFactors = FALSE)
Pafos <- read.csv(Pafos, header = T, skip=5, sep=',', fill=T,
                 stringsAsFactors = FALSE)
IMS <- read.csv(IMS, header = T, skip=5, sep=',', fill=T,
               stringsAsFactors = FALSE)

#bucle Agia####
j <- 1
for (i in 1:nrow(fechas)) {
  if(j <= nrow(Agia) & fechas[i,1] == Agia[j,2]) {
    if (Agia[j,'AOD_440nm'] >= 0) {
      fechas[i,2] <- Agia[j,'AOD_440nm']
    }
    fechas[i,7] <- Agia[j,35]
    j <- j+1
  }
}

#bucle CUT-TEPAK####
j <- 1
for (i in 1:nrow(fechas)) {
  if(j <= nrow(CUT) & fechas[i,1] == CUT[j,2]) {
    if (CUT[j,'AOD_440nm'] >= 0) {
      fechas[i,3] <- CUT[j,'AOD_440nm']
    }
    fechas[i,8] <- CUT[j,35]
    j <- j+1
  }
}

```

```

}

#bucle Pafos####
j <- 1
for (i in 1:nrow(fechas)) {
  if(j <= nrow(Pafos) & fechas[i,1] == Pafos[j,2]) {
    if (Pafos[j,'AOD_440nm'] >= 0) {
      fechas[i,4] <- Pafos[j,'AOD_440nm']
    }
    fechas[i,9] <- Pafos[j,35]
    j <- j+1
  }
}

#bucle Nicosisa####
j <- 1
for (i in 1:nrow(fechas)) {
  if(j <= nrow(Nicos) & fechas[i,1] == Nicos[j,2]) {
    if (Nicos[j,'AOD_440nm'] >= 0) {
      fechas[i,5] <- Nicos[j,'AOD_440nm']
    }
    fechas[i,10] <- Nicos[j,35]
    j <- j+1
  }
}

#bucle IMS####
j <- 1
for (i in 1:nrow(fechas)) {
  if(j <= nrow(IMS) & fechas[i,1] == IMS[j,2]) {
    if (IMS[j,'AOD_440nm'] >= 0) {
      fechas[i,6] <- IMS[j,'AOD_440nm']
    }
    fechas[i,11] <- Agia[j,35]
    j <- j+1
  }
}

#media####
fechas[,12] <- apply(fechas[,c(2:6)], 1 , mean, na.rm = T)

#promedio expo. amstrong####
fechas[,13] <- apply(fechas[,c(7:11)], 1 , mean, na.rm = T)

#valor para 550nm####
for(i in 1:nrow(fechas)){
fechas[i,14]<-(fechas[i,12]*(0.55^(-fechas[i,13])))/(0.44^(-fechas[i,13]))
}

```

```

#escritura datos####
library(xlsx)

write.csv(fechas, file = 'C:/Users/Usuario/Desktop/resultado.csv',
          append = F, sep = ',', col.names = T, row.names = F)

write.xlsx(fechas, file = 'C:/Users/Usuario/Desktop/resultado.xlsx',
          showNA = F, append = F, col.names = T, row.names = F)

```

---

Para el posible funcionamiento del Script, serán necesarios varios archivos: uno que contenga todas las fechas diarias durante esos 10 años en formato "dd:mm:yyyy" (será este el generador del data frame principal ya que determinará el número total de filas necesarias, dando un total de 4018 días/filas) y otros 5 que serán los archivos .csv en bruto descargados directamente desde el programa ADT para las 5 estaciones (figura 8).

El código lee los 6 archivos .csv antes mencionados y crea el data frame donde van a ser introducidos todos los datos necesarios por columnas (es decir, una columna para cada variable, hasta un total de 14 necesarias).

Dichos datos serán las fechas totales (del primer archivo) y, de las fechas para las que hay medidas, el exponente de Armstrong a 440nm-870nm (AE) y el valor de AOD a 440nm para cada estación, así como el promedio del AE, el promedio del AOD a 440nm y el valor del AOD a 550nm (este último se obtendrá al final del Script mediante la 'Ley de Armstrong').

Hecho esto, se inician 5 bucles consecutivamente, uno para cada estación o archivo correspondiente. Estos bucles recorrerán, uno a uno, los valores del archivo con todas las fechas y los valores de los archivos de las estaciones, de forma que, cuando el valor de la fecha del archivo que contiene todas éstas coincida con la fecha del archivo de la estación de Aeronet, el programa escriba, para ese día coincidente, el valor del AOD a 440nm y el del AE (en caso de que dichos valores existan) en la columna y fila correspondientes, indicada también en el propio bucle.

Terminados los bucles, habrá 11 columnas con datos: una con las fechas totales, 5 con los valores de AOD a 440nm y otras 5 con los exponentes de Armstrong de cada estación. Ahora será posible promediar los AE y los valores del AOD a 440nm de cada lugar y, mediante la 'Ley de Armstrong' y de estos datos promediados, obtener para cada día el valor del AOD a 550nm, que será el importante en este caso concreto.

Esto se ha hecho de la siguiente forma:

$$\tau_{440nm} = \beta \lambda_{440nm}^{-\alpha}$$

$$\tau_{550nm} = \beta \lambda_{550nm}^{-\alpha}$$

En estas dos ecuaciones, el exponente  $\alpha$  es el mismo (AE) y  $\lambda$  es también conocida, ya que para el AOD 550nm es 0,55 y para el AOD 440nm es 0,44.

Sabiendo esto, es posible dividir ambas ecuaciones y despejar así  $\tau_{550nm}$ , que es el valor desconocido:

$$\tau_{550nm} = \tau_{440nm} \frac{0,55^{-\alpha}}{0,44^{-\alpha}}$$

Así, obtenemos los valores para el AOD a 550nm, que se escribe en una nueva columna del data frame antes mencionado.

Al final del código se crea un archivo .xlsx (hoja de cálculo de Excel) llamado 'resultado', donde se plasmará el data frame anterior completo (9).

Una vez los datos han sido correctamente procesados, organizados y calculados, conviene graficar las fechas para las que hay valores, ya que si tenemos periodos muy amplios sin datos, entonces los cálculos posteriores no serán tan buenos.

Se mostrarán en el mismo gráfico los valores de las 5 estaciones del AOD a 440nm, que son los que realmente nos permitirán ver donde hay datos o no puesto que el valor del AOD a 550nm fué obtenido a partir de ellos, además de dicho valor para 550nm.

Esto se ha hecho mediante el siguiente código:

```
#lectura datos####
fileIN <- 'C:/Users/Usuario/Desktop/resultado.csv'
datos <- read.csv(fileIN, skip=0, header=T, sep=',', fill=T,
                 stringsAsFactors = FALSE)

#fechas####
fechas <- strptime(paste(datos$fecha), "%d:%m:%Y", tz='GMT')

x1 <- as.POSIXct('2010-01-01', format="%Y-%m-%d")
x2 <- as.POSIXct('2020-12-31', format="%Y-%m-%d")

#creacion png####
imageOUT <- paste('C:/Users/Usuario/Desktop/resultado.png', sep='')
png(imageOUT, width=8, height=4, res=300, units='in')

par(bg="white", mar = c(3,3,2,2), mgp=c(0.5,0.4,0), tck=-0.02)

mycolors <- c('blue', 'red', 'green', 'orange', 'purple', 'black')

#plot o representaciones####
plota <- data.frame(fechas, datos$Agia.Marina, datos$CUT.TEPAK,
                  datos$Pafos, datos$Nicosia, datos$IMS,
                  datos$X550nm, stringsAsFactors=FALSE)
```



```

for(np in 1:(length(plota[1,])-1)){
  if(np>1) par(new=T)
  if(np==1){
    plot(plota$fechas,plota[(1+np)],pch=20, cex=0.5,
          col=mycolors[np],xlim=c(x1,x2), ylim=c(0,1.2),xlab='',
          ylab='')
  }
  if(np>1){
    plot(plota$fechas,plota[(1+np)],pch=20, cex=0.5,
          col=mycolors[np],xlim=c(x1,x2), ylim=c(0,1.2), xlab='',
          ylab='')
  }
}

#creacion del grafico####
axis(1, at=seq(x1, x2, "month"),labels = FALSE, tcl = -0.2)
axis(1, at=seq(x1, x2, "year"),labels = paste('20',c(10:20), sep=''))

axis(2, at=seq(0, 1.2, 0.1))
axis(2, at=seq(0, 1.2, 0.05),labels = FALSE, tcl = -0.2)

legend('topleft',c('Agia_440nm', 'CUT_440nm', 'Pafos_440nm',
  'Nicosia_440nm', 'IMS_440nm', 'AOD_550nm'),
  text.col=mycolors, cex=0.6, bty='n')

mtext('AOD',side=2,line=1.5)
mtext('Fecha (1 de enero de cada año)',side=1,line=1.5)

dev.off()

```

---

El código, leerá primero el archivo .xlsx sacado por 'fechas', concretamente las columnas 2 a 6 (datos de 440nm) y la columna 14 (valores para 550nm), y después creará el archivo de imagen donde será plasmado el gráfico con los datos.

Tras definir cómo será el gráfico, se representan los valores del AOD frente a la fecha y se guarda como archivo de imagen tipo .png:

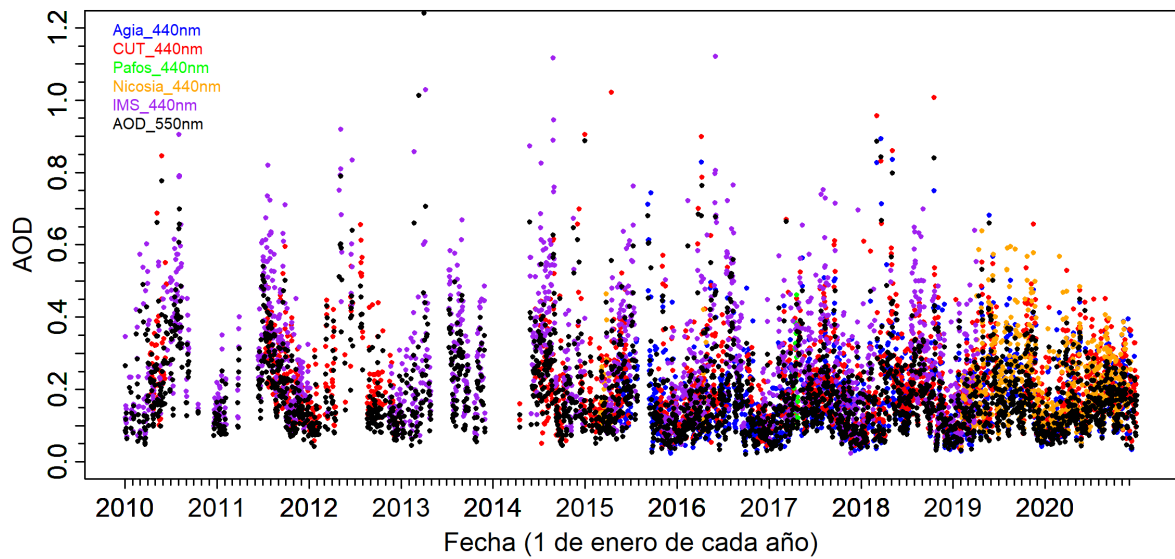


Figura 2: Salida .png del Script 'lectura AOD'.

Esta gráfica será útil para obtener posibles periodicidades de la presencia de este aerosol en la zona estudiada mediante Análisis de Fourier y también para compararlo con mediciones tomadas por satélite del mismo tipo de partícula.

Ya sólo quedará obtener el mapa de climatología y ver las fuentes de emisión de este tipo de aerosol.

Para ello, se crea el Script llamado 'mapa climatologia':

```
#lectura datos####
fileIN <- 'C:/Users/Usuario/Desktop/resultado.csv'
datos <- read.csv(fileIN, skip=0, header=T, sep=',', fill=T,
                 stringsAsFactors = FALSE)

#lectura fechas####
fechas <- read.csv('C:/Users/Usuario/Desktop/550nm_completa.csv',
                 header = T, skip=0, sep=',', fill=T,
                 stringsAsFactors = FALSE)

#lectura meses####
date <- as.Date(fechas[,1], '%d:%m:%Y')
mes <- as.numeric(format(date, '%m'))

#creacion matriz con valores mensuales####
estadisticosResumen <- matrix(0, ncol = 6, nrow = 12)
colnames(estadisticosResumen) <- c("Media", 'mediana', 'q1', 'q3',
                                'mínimo', 'máximo' )
```

```

rownames(estadisticosResumen) <- c("Enero", "Febrero", "Marzo",
                                   "abril", 'mayo', 'junio',
                                   'julio', 'agosto', 'septiembre',
                                   'octubre', 'noviembre',
                                   'diciembre')

#bucle que recorre todos los valores####
for (i in 1:12) {
  estadisticosResumen[i,1]<-mean(na.omit(datos$X550nm[mes == i]))
  estadisticosResumen[i,2]<-median(na.omit(datos$X550nm[mes == i]))
  estadisticosResumen[i,3]<-quantile(na.omit(datos$X550nm[mes == i]),
                                     probs = 0.25)
  estadisticosResumen[i,4]<-quantile(na.omit(datos$X550nm[mes == i]),
                                     probs = 0.75)
  estadisticosResumen[i,5]<-min(na.omit(datos$X550nm[mes == i]))
  estadisticosResumen[i,6]<-max(na.omit(datos$X550nm[mes == i]))
}

#grafica climatologia####
numeromes <- c(1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12)

meses <- paste('C:/Users/Usuario/Desktop/meses.jpeg',sep='')
jpeg(meses, width = 720, height = 720, quality = 100, bg = 'white')

plot(numeromes, estadisticosResumen[,1], xlim = c(1,12),
     ylim = c(0,1.3), type = "l", col = 1 , lwd = 3, pch = 8,
     xlab = 'mes', ylab = 'valor del AOD')

points(numeromes, estadisticosResumen[,2], col = 2, pch = 7, cex=2)

lines(numeromes, estadisticosResumen[,5], type = "l",
      col = rgb(1, 0.6, 0.8, alpha = 0.1), lwd = 0, pch = 20)
lines(numeromes, estadisticosResumen[,6], type = "l",
      col = rgb(1, 0.6, 0.8, alpha = 0.1) , lwd = 0.1, pch = 20)
polygon(c(numeromes, rev(numeromes)), c(estadisticosResumen[,3],
    rev(estadisticosResumen[,4])), col=rgb(1,0.6,0.8,alpha=0.5),
    lty = 0)

lines(numeromes, estadisticosResumen[,3], type = "l",
      col = rgb(0.6, 1, 0.4, alpha = 0.2), lwd = 0.1, pch = 20)
lines(numeromes, estadisticosResumen[,4], type = "l",
      col = rgb(0.6, 1, 0.4, alpha = 0.2), lwd = 0.1, pch = 20)

```

```

polygon(c(enumeromes, rev(enumeromes)), c(estadisticosResumen[,5],
      rev(estadisticosResumen[,6])),
      col = rgb(0.6, 1, 0.4, alpha = 0.3), lty = 0)

legend(x = 'topright', legend = c("diff. máx/mín.", "q1-q3",
  'media', 'mediana'), col = c(rgb(0.6, 1, 0.4, alpha = 0.2),
  rgb(1, 0.6, 0.8, alpha = 0.4),1,2), bty = "n",
  pch = c(19,19,20,7), cex = 1.8, y.intersp = 0.9)

dev.off()

library(xlsx)
write.xlsx(estadisticosResumen,
  file = 'C:/Users/Usuario/Desktop/valores mensuales.xlsx',
  showNA = F, append = F, col.names = T, row.names = T)

#gráfico AE-AOD####
AOD_550nm <- datos[,14]
AE <- datos[,13]

groups <- rep("Mezcla", 4018)
groups[AOD_550nm > 0.2 & AE < 0.6] <- "Desertico"
groups[AOD_550nm > 0.2 & AE > 1.3] <- "Biomasa"
groups[AOD_550nm < 0.2 & AE < 1] <- "Marino"
groups[AOD_550nm < 0.2 & AE > 1] <- "Continental"

procedenciasAOD<-paste('C:/Users/Usuario/Desktop/procedenciaAOD.jpeg',
  sep='')
jpeg(procedenciasAOD, width = 720, height = 720, quality = 100,
  bg = 'white')

plot(AOD_550nm, AE, pch = 20, col = as.factor(groups))

legend(x = 0.85, y = 1.1, legend = c("mezcla", "desértico",
  'urbano/biomasa/industrial', 'marino', 'continental/rural'),
  col = c(5,3,1,4,2), bty = "n", pch = 19, cex = 1.3,
  horiz = FALSE, title = 'procedencia:')

dev.off()

```

El Código empieza, como el anterior, abriendo el archivo .xlsx principal sacado por 'fechas' y, después de leer los datos, sólo obtendrá información sobre los meses (ya que para la realización del mapa de climatología interesa ver las variaciones mensuales y no tanto la variación del clima a lo largo de los años) creando un nuevo 'data frame' con los nombres de los meses y las funciones que queremos calcular. Estos serán la media mensual, mediana, la diferencia entre máximos y mínimos y también la diferencia entre el primer y tercer cuartil.

El programa calcula entonces las funciones anteriores y crea el mapa de climatología mostrado en la figura 3, así como el gráfico 4, que expone la procedencia u origen de las partículas según su valor del exponente de Amstrong en función del AOD a 550nm.

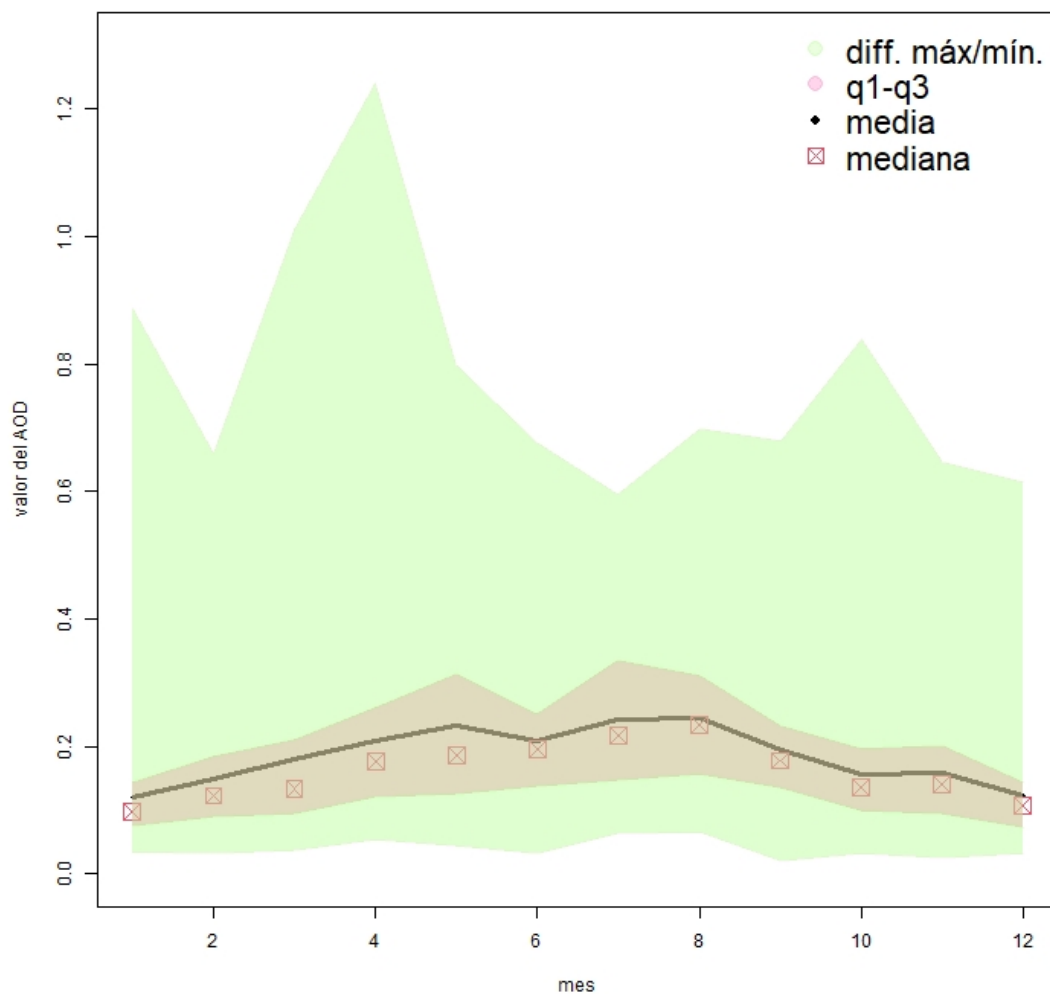


Figura 3: Mapa de climatología.

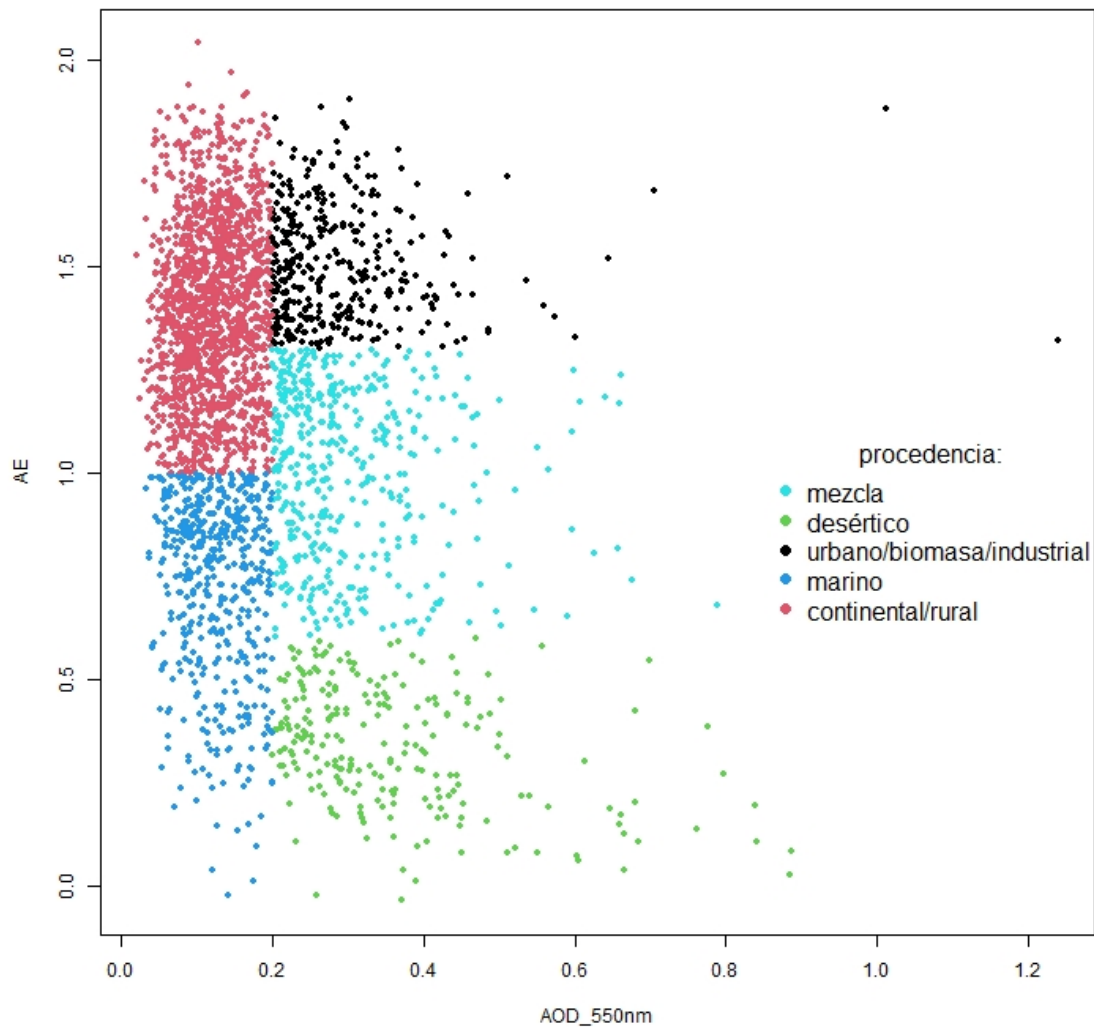


Figura 4: Clasificación en 5 tipos de aerosol en la región de Chipre.

Como dato adicional y relevante, es posible calcular el porcentaje de cada tipo de aerosol con un Script como el siguiente:

```
#lectura datos####
fileIN <- 'C:/Users/Usuario/Desktop/resultado.csv'
datos <- read.csv(fileIN, skip=0, header=T, sep=',', fill=T,
stringsAsFactors = FALSE)

sum(is.finite(datos[, "X550nm"])) -> total

AOD_550nm <- datos[,14]
AE <- datos[,13]

groups <- rep("Mezcla", 4018)
groups[AOD_550nm > 0.2 & AE < 0.6] <- "Desertico"
```

```

groups[AOD_550nm > 0.2 & AE > 1.3] <- "Biomasa"
groups[AOD_550nm < 0.2 & AE < 1] <- "Marino"
groups[AOD_550nm < 0.2 & AE > 1] <- "Continental"

# % Desertico ####
sum(is.na(groups[AOD_550nm > 0.2 & AE < 0.6])) -> q
length(groups[AOD_550nm > 0.2 & AE < 0.6]) -> w
x <- ((w-q)/total)*100

# % Biomasa ####
sum(is.na(groups[AOD_550nm > 0.2 & AE > 1.3])) -> e
length(groups[AOD_550nm > 0.2 & AE > 1.3]) -> r
y <- ((r-e)/total)*100

# % Marino ####
sum(is.na(groups[AOD_550nm < 0.2 & AE < 1])) -> s
length(groups[AOD_550nm < 0.2 & AE < 01]) -> d
z <- ((d-s)/total)*100

# % Continental ####
sum(is.na(groups[AOD_550nm < 0.2 & AE > 1])) -> a
length(groups[AOD_550nm < 0.2 & AE > 1]) -> f
t <- ((f-a)/total)*100

# % mezcla ####
m <- 100-(t+z+x+y)

```

---

Los porcentajes que se obtienen al ejecutar el código, son:

- 7.95 % para el Desértico
- 12.04 % para el procedente de biomasa
- 16.03 % para el Marino
- 49.23 % para el Continental
- 14.75 % para el Mezclado

que cuadran correctamente con lo visible en la gráfica 4.

### 3. ANEXO

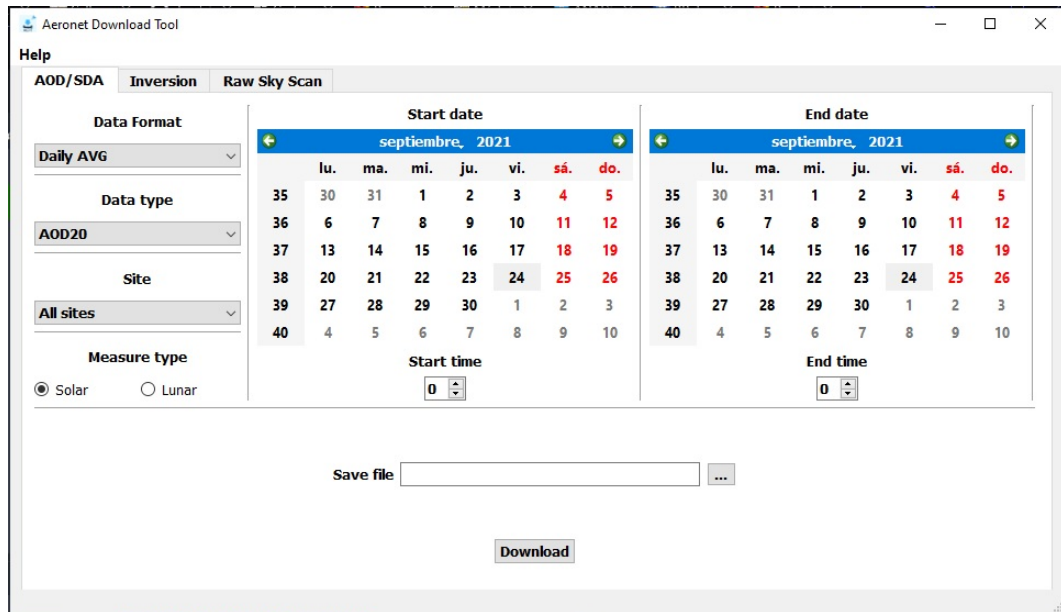


Figura 5: Interfaz del programa 'Aeronet Download Tool'

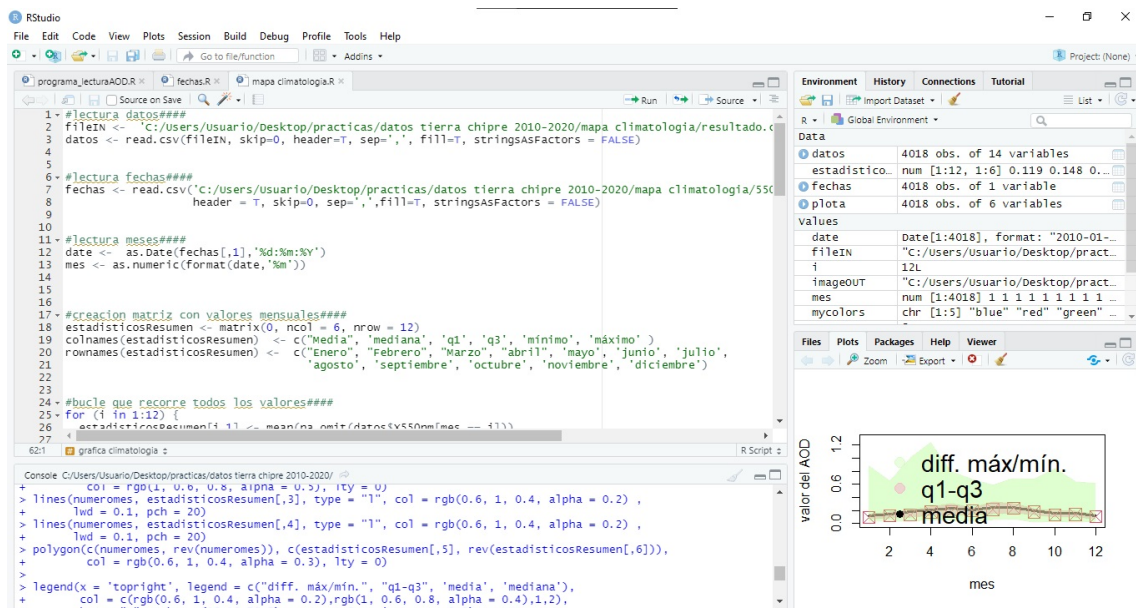


Figura 6: Interfaz principal del programa 'Rstudio'



```
Archivo Edición Formato Ver Ayuda
AERONET_Database_Site_List,Num=2,Date_Generated=12:04:2021
Site_Name,Longitude(decimal_degrees),Latitude(decimal_degree:
Cuiaba,-56.000000,-15.500000,250.000000
Alta_Floresta,-56.104453,-9.871339,277.000000
Jamari,-62.750000,-8.633333,100.000000
Tucson,-110.953003,32.233002,779.000000
GSFC,-76.839833,38.992500,87.000000
Kolfield,-74.476387,39.802223,50.000000
Porto_Nacional,-48.000000,-11.000000,210.000000
Brasilia,-47.900002,-15.917000,1100.000000
Harvard_Forest,-72.188000,42.532000,322.000000
Wallops,-75.474998,37.941666,10.000000
Hampton_Roads,-76.449997,36.783333,10.000000
Hagerstown,-77.724998,39.708332,200.000000
Hog_Island,-75.700000,37.420000,50.000000
Tukurui,-49.683331,-3.716666,100.000000
Arizona,-110.788002,32.442001,2791.000000
Santarem,-54.750000,-2.433333,70.000000
Ussuriysk,132.163500,43.700400,280.000000
Waskesiu,-106.083000,53.916698,550.000000
Ariquiums,-63.033332,-9.916667,80.000000
Colorado,-105.550003,40.450001,2500.000000
JamTown,-63.099998,-9.200000,50.000000
Key_Biscayne,-80.163330,25.731966,0.000000
Sevilleta,-106.885002,34.354721,1477.000000
Jornada,-106.516944,32.351111,1288.000000
CART_SITE,-97.486390,36.606670,318.000000
Cart_Site,-97.486390,36.606670,318.000000
NSA_YJP_BOREAS,-98.290001,55.903000,290.000000
FLIN_FLON,-101.690002,54.669998,305.000000
SSA_YJP_BOREAS,-104.650002,53.674999,490.000000
PRINCE_ALBERT,-105.699997,53.200001,425.000000
```

Línea 1, columna 1    100%    Windows (CRLF)    UTF-8

Figura 7: Extracto del archivo de texto 'aeronet-locations.txt'

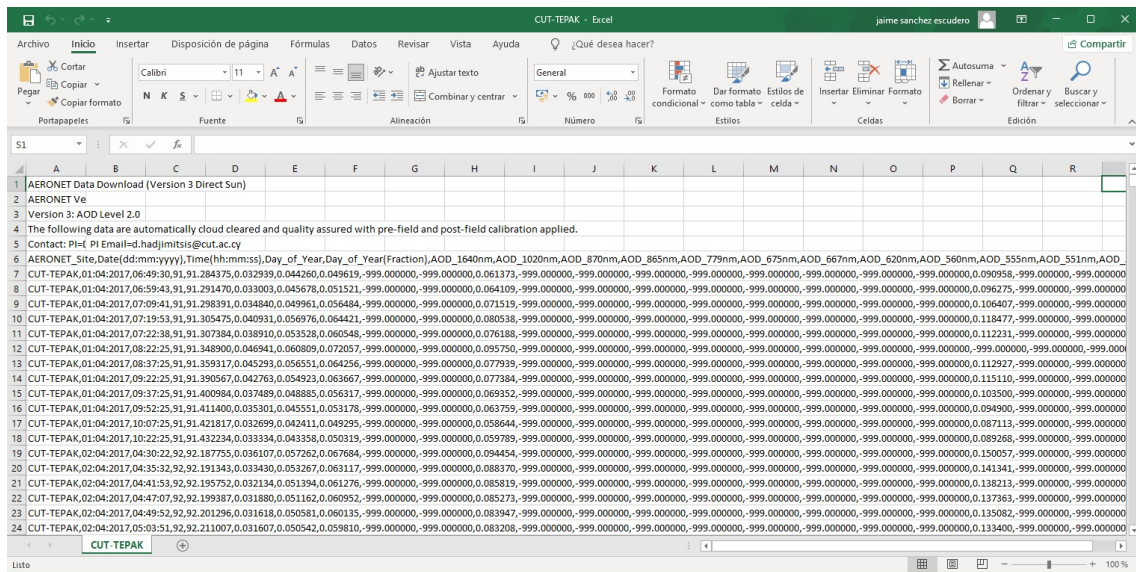


Figura 8: Ejemplo de la estructura de un archivo .csv

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	fecha	Agia-Marina	CUT-TEPAK	Pafos	Nicosia	IMS-METU-ERDEMLI	exp. Agia	exp. CUT	exp. Pafos	exp. Nicos	exp. IMS	Media para 440nm	Media Exponente	550nm
2	01-01-2010													
3	02-01-2010					0.345387					1.187163	0.345387	1.187163	0.265007403
4	03-01-2010					0.121323					1.596307	0.121323	1.596307	0.084965968
5	04-01-2010					0.112812					1.514991	0.112812	1.514991	0.080452122
6	05-01-2010					0.1544					1.267211	0.1544	1.267211	0.116370222
7	06-01-2010					0.10651					0.636029	0.10651	0.636029	0.0924172
8	07-01-2010					0.146379					0.367966	0.146379	0.367966	0.134840137
9	08-01-2010					0.069995					0.426588	0.069995	0.426588	0.063639443
10	09-01-2010													
11	10-01-2010													
12	11-01-2010													
13	12-01-2010													
14	13-01-2010													
15	14-01-2010													
16	15-01-2010													
17	16-01-2010													
18	17-01-2010													
19	18-01-2010													
20	19-01-2010					0.233964					0.602447	0.233964	0.602447	0.204534179
21	20-01-2010													
22	21-01-2010													
23	22-01-2010													
24	23-01-2010													
25	24-01-2010													
26	25-01-2010					0.119386					1.115205	0.119386	1.115205	0.093084822
27	26-01-2010					0.233666					0.884299	0.233666	0.884299	0.191821859
28	27-01-2010					0.14222					0.247478	0.14222	0.247478	0.134579087
29	28-01-2010													
30	29-01-2010													
31	30-01-2010													
32	31-01-2010													
33	01-02-2010													
34	02-02-2010													
35	03-02-2010					0.084087					0.236516	0.084087	0.236516	0.079764214
36	04-02-2010					0.127686					0.328066	0.127686	0.328066	0.118672585
37	05-02-2010					0.158327					0.133454	0.158327	0.133454	0.153681628
38	06-02-2010					0.135531					0.327855	0.135531	0.327855	0.125969733
39	07-02-2010													
40	08-02-2010													
41	09-02-2010													
42	10-02-2010													
43	11-02-2010													
44	12-02-2010													
45	13-02-2010													
46	14-02-2010					0.421402					0.801914	0.421402	0.801914	0.352357156
47	15-02-2010													

Figura 9: Fragmento del data frame en formato .xlsx usado en 'fechas'