

Adaptación al cambio climático en CyL - anexo

7.1_Climodiagramas

Irene Ruano Benito, Aitor Vázquez Veloso

16/06/2022

Consideraciones iniciales

Este script ha sido elaborado en: - SO: Windows - Encoding: UTF-8 - R: versión 4.2.0 -> IMPORTANTE, con versiones anteriores a 3.6.0 da problemas

Índice de contenidos

- Carga de librerías
- Carga de datos (periodo previo: 1960-1990)
- Procesado de datos (periodo previo: 1960-1990)
- Elaboración de climodiagramas (periodo previo: 1960-1990)
- Procesado de datos (periodo reciente: 1990-2020)
- Elaboración de climodiagramas (periodo reciente: 1990-2020)
- Elaboración de gráficos de temperatura y precipitación (periodo reciente: 1990-2020)

Carga de librerías

Como paso previo, es necesario instalar las librerías 'knitr' y 'rmarkdown' (requisito para utilizar RMarkdown). Si las tenemos instaladas, entonces cargamos las librerías necesarias y los datos con los que trabajaremos.

```
#### Instalación de librerías necesarias para utilizar RMarkdown ####  
install.packages('knitr')
```

```
## Installing package into '/home/aitor/R/x86_64-pc-linux-gnu-library/4.2'  
## (as 'lib' is unspecified)
```

```
install.packages('rmarkdown')
```

```
## Installing package into '/home/aitor/R/x86_64-pc-linux-gnu-library/4.2'  
## (as 'lib' is unspecified)
```

```
#### Carga de librerías (instalar previamente con install.packages('nombre_librería') si fuera necesari  
library('readxl') # carga de datos  
library('dplyr') # procesado de datos
```

```
##
```

```
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
```

```
##
```

```
## filter, lag
```

```
## The following objects are masked from 'package:base':
##
## intersect, setdiff, setequal, union

library('stringr') # climodiagramas
library('berryFunctions') # climodiagramas

##
## Attaching package: 'berryFunctions'

## The following object is masked from 'package:dplyr':
##
## between

library('ggplot2') # gráficos temperatura y precipitación
library('prettyunits') # gráficos temperatura y precipitación
```

Carga de datos (periodo previo: 1960-1990)

En este apartado se cargan los datos pluviométricos y termométricos del periodo previo (1960-1990) obtenidos del AEMET para todas las provincias de Castilla y León, así como los datos de las estaciones meteorológicas.

```
# - datos pluviométricos del periodo previo

# en este caso, dado que son muchos datos, el AEMET nos los ha proporcionado en dos documentos
pluv_previo <- "../datos_ejemplo/Resumen_precipitaciones_ponferrada-previo.xlsx"

Provincias <- c("León")

PluviometricosCyl <- NULL
hoja <- read_excel(pluv_previo, sheet = Provincias[1])
PluviometricosCyl <- rbind(PluviometricosCyl, hoja)

rm(pluv_previo, hoja) # eliminamos df temporales

# - datos termométricos del periodo previo
termometrico_previo <- "../datos_ejemplo/Resumen_temperaturas_ponferrada-previo.xlsx"

TermometricosCyl <- NULL
hoja <- read_excel(termometrico_previo, sheet = Provincias[1])
TermometricosCyl <- rbind(TermometricosCyl, hoja)

previo <- merge(PluviometricosCyl, TermometricosCyl) # unimos pluvio/termométricos
names(previo)[2] <- "ANHO" # renombramos la columna 'AÑO' omitiendo la ñ

rm(termometrico_previo, hoja, Provincias) # eliminamos df temporales

# - datos de las estaciones meteorológicas
estaciones_meteo <- read.csv("../datos_ejemplo/estacion_coord46.csv")
```

Procesado de datos (periodo previo)

En este apartado se calculan los datos medios de temperatura y precipitación mensuales para cada estación meteorológica, comprobando cuales tienen datos para todos los meses del año.

```
#### Calculamos las medias mensuales para cada estación meteorológica ####

previo <- previo[(previo$INDICATIVO %in% estaciones_meteo$estacionesantes), ] # filtro para estaciones

previo <- previo[!is.na(previo$TM_MES), ] # eliminamos filas vacías
previo_promedio <- NULL

for(i in unique(previo$INDICATIVO)){ # para cada estación meteorológica...
  estacion_i <- previo %>% filter(INDICATIVO == i)

  for(j in unique(estacion_i$MES)){ # para cada mes...
    estacion_ij <- estacion_i %>% filter(MES == j)

    Pmedia <- round(mean(estacion_ij$PMES77)/10, 3) # calculamos precipitación...
    Tmedia <- round(mean(estacion_ij$TM_MES)/10, 3) # y temperatura media mensual

    previo_promedio_ij <- as.data.frame(cbind(unique(estacion_ij$INDICATIVO),
                                              unique(estacion_ij$NOMBRE),
                                              unique(estacion_ij$MES),
                                              Pmedia,
                                              Tmedia))
    previo_promedio <- as.data.frame(rbind(previo_promedio, previo_promedio_ij))
  }
}

names(previo_promedio) <- c("INDICATIVO", "NOMBRE", "mes", "Pmedia", "Tmedia") # renombramos variables
previo_promedio$mes <- as.numeric(previo_promedio$mes)

#### Comprobamos qué estaciones tienen registros de datos para los 12 meses ####

estaciones_completas <- data.frame()
print('La siguiente lista de estaciones meteorológicas no tienen datos de los 12 meses:')

## [1] "La siguiente lista de estaciones meteorológicas no tienen datos de los 12 meses:"
for (i in unique(previo_promedio$INDICATIVO)){ # para cada estación meteorológica...
  estacion_i <- previo_promedio[previo_promedio$INDICATIVO==i, ]

  if (nrow(estacion_i) == 12){ # la conservamos si tiene datos de los 12 meses
    estaciones_completas <- rbind(estaciones_completas, estacion_i)
  } else{ # y la deseamos sino, mostrando cuáles
    print(unique(estacion_i$INDICATIVO))
  }
}

estaciones_completas <- estaciones_completas[order(estaciones_completas$mes), ] # ordenamos los datos
estaciones_completas <- estaciones_completas[!is.na(estaciones_completas$INDICATIVO), ] # eliminamos c

rm(estacion_i, estacion_ij, previo_promedio_ij, i, j, Pmedia, Tmedia) # eliminamos df temporales
```

Elaboración de climodiagramas (periodo previo: 1960-1990)

Con los datos previamente procesados se realizan los climodiagramas para las estaciones con datos completos del periodo previo (42 estaciones). *no se ha conseguido modificar la ruta para exportar los gráficos directamente a la carpeta 'datos'*

```
for (j in unique(estaciones_completas$INDICATIVO)){ # para cada estación...
  estacion <- filter(estaciones_completas, INDICATIVO == j) # selecciono la estación
  limite <- max(as.numeric(estacion$Pm))

  name <- paste('climo_', j, '_previo', '.png', sep = '') # nombre del gráfico
  png(name) # orden para exportar el gráfico

  if (limite > 100){ # si limite > 100, sin limite en el grafico para el eje y
    grafico <- climateGraph(as.numeric(estacion$Tmedia), as.numeric(estacion$Pm),
      main = "Previo (1960-1990)", units = c("\U{00B0}C", "mm"),
      labs = c('E', 'F', 'M', 'A', 'M', 'J', 'J', 'A', 'S', 'O', 'N', 'D'),
      mar = c(2, 3, 4, 3), textprop = 0, compress = TRUE)
    mtext(estacion$NOMBRE, side = 1, line = 1, at = 1.5, cex = 0.7, font = 3, col = "chocolate4")
  }
  else{ # si limite < 100, entonces el límite en el eje y = 100
    grafico <- climateGraph(as.numeric(estacion$Tmedia), as.numeric(estacion$Pm),
      main = "Previo (1960-1990)", units = c("\U{00B0}C", "mm"),
      labs = c('E', 'F', 'M', 'A', 'M', 'J', 'J', 'A', 'S', 'O', 'N', 'D'),
      mar = c(2, 3, 4, 3), ylim = c(0, 100/2), textprop=0)
    mtext(estacion$NOMBRE, side = 1, line = 1, at = 1.5, cex = 0.7, font = 3, col = "chocolate4")
  }

  dev.off() # Cerramos el dispositivo gráfico
}

rm(estacion, limite, name, grafico, j)
```

Carga de datos (periodo reciente: 1990-2020)

En este apartado se cargan los datos pluviométricos y termométricos del periodo reciente (1990-2020) obtenidos del AEMET para todas las provincias de Castilla y León.

```
# - datos pluviométricos del periodo reciente
pluv_reciente<-"../datos_ejemplo/Resumen_precipitaciones_ponferrada-reciente.xlsx"

Provincias <- c("Leon")

PluviometricosCyl <- NULL
hoja <- read_excel(pluv_reciente, sheet = Provincias[1])
PluviometricosCyl <- rbind(PluviometricosCyl, hoja)

# - datos termométricos del periodo reciente
term_reciente<-"../datos_ejemplo/Resumen_temperaturas_ponferrada-reciente.xlsx"

#TermometricosCyL <- NULL
hoja <- read_excel(term_reciente, sheet = Provincias[1])
TermometricosCyL <- rbind(TermometricosCyL, hoja)
```

```

reciente <- merge(PluviometricosCyL, TermometricosCyL) # unimos ambos archivos
names(reciente)[2]<- "ANHO" # renombrado sin caracteres especiales

rm(pluv_reciente, term_reciente, PluviometricosCyL, TermometricosCyL, hoja, Provincias) # eliminamos d.

```

Procesado de datos (periodo reciente: 1990-2020)

En este apartado se calculan los datos medios de temperatura y precipitación mensuales para cada estación meteorológica y quinquenio entre 1990 y 2020, comprobando cuales tienen datos para todos los meses del año.

Eliminamos estaciones meteorológicas muy próximas entre sí (se ha comprobado en QGIS con su posición)

```

quitar <- c("2235U", "2236", "2320E", "2450", "2519", "2121", "2532", "2916A", "2946X", "2946", "2440")
reciente <- reciente[!(reciente$INDICATIVO %in% quitar), ]

```

Calculamos los datos quinquenales para cada estación meteorológica

```

estacion_ID <- unique(reciente$INDICATIVO)
datos_5 <- data.frame()
periodos_recientes <- c(1991, 1996, 2001, 2006, 2011, 2016)

for (i in estacion_ID){ # para cada estación...

  estacion_i <- reciente[reciente$INDICATIVO == i, ]
  temp <- data.frame()

  for (l in periodos_recientes){ # para cada periodo...

    # se filtran los datos de estación y periodo
    inicio <- l
    fin <- l+4
    serie_temp <- filter(estacion_i, ANHO >= inicio, ANHO <= fin)

    Pmedia <- data.frame()
    Tmedia <- c()

    for (mes in 1:12){ # para cada mes...

      serie <- filter(serie_temp, MES == mes)

      # se calculan la precipitación y temperatura promedio del periodo
      Pm <- mean(serie$PMES77)/10
      Pm <- cbind(mes, Pm)
      Pmedia <- rbind(Pmedia, Pm)

      Tm <- mean(serie$TM_MES)/10
      Tmedia <- c(Tmedia, Tm)
    }

    # se almacenan los resultados de cada estación y periodo
    serie_total <- cbind(inicio, fin, Pmedia, Tmedia)
    temp <- rbind(temp, serie_total)
  }
}

```

```

# se almacenan los datos de todas las estaciones
previo <- cbind (estacion_i[1:nrow(temp), c(1, 4:10)], temp)
datos_5 <- rbind (datos_5, previo)
}

# se eliminan datos vacíos
datos_5 <- datos_5[!is.na(datos_5$INDICATIVO), ]
datos_5 <- datos_5[!is.na(datos_5$Pm), ]
datos_5 <- datos_5[!is.na(datos_5$Tmedia), ]

#### Comprobamos los años que tenemos completos (12 meses con datos)

estacion_IDs <- unique(datos_5$INDICATIVO)
estacion_5 <- data.frame()

for (i in estacion_IDs){ # para cada estación...

  estacion <- datos_5[datos_5$INDICATIVO == i, ]

  # independizamos las variables para tratarlas con mayor facilidad
  estID <- unique(estacion$INDICATIVO)
  nombre <- unique(estacion$NOMBRE)
  Xcoord <- unique (estacion$C_X)
  Ycoord <- unique (estacion$C_Y)
  Zcoord <- unique (estacion$ALTITUD)

  inicio <- min(estacion$inicio)
  final <- max(estacion$fin)
  nserie <- length(unique(estacion$inicio))
  nmeses <- length(estacion$mes)

  if (nserie == 6){
    ausencias <- "completo"
  } else{
    ausencias <- "falta"
  }

  if (nmeses == 72){
    nmeses <- "completo"
  } else{
    nmeses <- "falta"
  }

  falta <- 6 - nserie

  test_estacion <- data.frame(estID, nombre, Xcoord, Ycoord, Zcoord, inicio, final, nserie, nmeses, ausencias)
  estacion_5 <- rbind(estacion_5, test_estacion)
}

# filtramos las estaciones con datos completos
estacion_5comp <- filter(estacion_5, ausencias == "completo")

```

```

estacion_5comp <- filter(estacion_5comp, nmeses == "completo")
datos_5 <- datos_5[(datos_5$INDICATIVO %in% estacion_5comp$estID), ]

#### Depuración de datos - eliminando huecos vacíos

datos_NAs_Pm <- filter(datos_5, is.na(Pm))
datos_NAs_Tmedia <- filter(datos_5, is.na(Tmedia))

IDs_NAs_Pm <- datos_NAs_Pm$INDICATIVO
IDs_NAs_Tmedia <- datos_NAs_Tmedia$INDICATIVO

IDs_NAs_Pm <- subset(IDs_NAs_Pm, !duplicated(IDs_NAs_Pm))
IDs_NAs_Tmedia <- subset(IDs_NAs_Tmedia, !duplicated(IDs_NAs_Tmedia))

datos_5_finales <- filter(datos_5, !datos_5$INDICATIVO %in% IDs_NAs_Pm)
datos_5_finales <- filter(datos_5_finales, !datos_5_finales$INDICATIVO %in% IDs_NAs_Tmedia)

# eliminar variables temporales
rm(ausencias, estacion_ID, estacion_IDs, estID, falta, fin, final, i, IDs_NAs_Pm, IDs_NAs_Tmedia, inici
rm(datos_5, datos_NAs_Pm, datos_NAs_Tmedia, estacion, estacion_5, estacion_5comp, estacion_i, Pm, Pmedi

```

Elaboración de climodiagramas (periodo reciente: 1990-2020)

Con los datos previamente procesados se realizan los climodiagramas para las estaciones con datos completos del periodo reciente (42 estaciones). *no se ha conseguido modificar la ruta para exportar los gráficos directamente a la carpeta 'datos'*

```

# extraemos los identificadores de las estaciones meteorológicas
ids <- list()
ids <- datos_5_finales$INDICATIVO
ids <- subset(ids, !duplicated(ids))

for (j in ids){ # para cada estación...

  estacion <- filter(datos_5_finales, INDICATIVO == j) # selecciono la estación

  # creo el bucle para graficar los datos de cada estación
  for (i in periodos_recientes){ # para cada serie climática...

    name <- paste('climo_', j, '_', i, '.png', sep = '') # nombro la imagen con el ID de estación y el
    png(name) # orden para exportar el gráfico

    titulo <- str_c(as.character(i), "-", as.character(i+4)) # titulo del gráfico (periodo de estudio)
    temporal <- estacion[estacion$inicio == i, ]
    temp <- as.numeric(temporal$Tmedia)
    prec <- as.numeric(temporal$Pm)
    limite <- max(as.numeric(temporal$Pm)) # si limite > 100, sin limite en el grafico; si limite < 100

    if (limite > 100) {

      plot <- climateGraph(temp, prec, main = titulo, units = c("\U{00B0}C", "mm"), mar = c(2, 3, 4, 3),
        labs = c('E', 'F', 'M', 'A', 'M', 'J', 'J', 'A', 'S'),
      mtext(estacion$NOMBRE, side = 1, line = 1, at = 1.5, cex = 0.7, font = 3, col = "chocolate4")
    }
  }
}

```

```

    } else{

      plot <- climateGraph(temp, prec, main = titulo, units = c("\U{00B0}C", "mm"), mar = c(2, 3, 4, 3),
                           labs = c('E', 'F', 'M', 'A', 'M', 'J', 'J', 'A', 'S'),
                           mtext(estacion$NOMBRE, side = 1, line = 1, at = 1.5, cex = 0.7, font = 3, col = "chocolate4"))

    }

    dev.off() # Cerramos el dispositivo gráfico
  }
}

rm(estacion, temporal, i, ids, j, limite, name, periodos_recientes, plot, prec, temp, titulo)

```

Elaboración de gráficos de temperatura y precipitación (periodo reciente: 1990-2020)

Los datos anteriores se adaptan para obtener gráficos que muestran las diferencias en temperatura y precipitación entre el periodo previo y los quinquenios estudiados en el periodo reciente para cada estación meteorológica.

Adaptación de los datos a los gráficos finales

```

datos_5_finales$periodo_reciente <- str_c(as.character(datos_5_finales$inicio), "-", as.character(datos_5_finales$fin))

estaciones_completas$periodo <- str_c("1960-1990")
datos_5_finales$periodo <- ifelse(datos_5_finales$periodo_reciente == "1991-1995", "1991-1995",
                                  ifelse(datos_5_finales$periodo_reciente == "1996-2000", "1996-2000",
                                          ifelse(datos_5_finales$periodo_reciente == "2001-2005", "2001-2005",
                                                  ifelse(datos_5_finales$periodo_reciente == "2006-2010",
                                                          ifelse(datos_5_finales$periodo_reciente == "2011-2020",
                                                                  ifelse (datos_5_finales$periodo_reciente == "2016-2020", "2016-2020", "2011-2020"),
                                                                "2011-2020"),
                                                            "2006-2010"),
                                                    "2001-2005"),
                                              "1996-2000"),
                                  "1991-1995")

datos_5_finales <- datos_5_finales[, c(1:2, 11:13, 15)] # Seleccionamos solo las columnas que necesitamos
datos_5_finales <- datos_5_finales[(datos_5_finales$INDICATIVO %in% estaciones_meteo$estacionesantes), ]
estaciones_completas <- estaciones_completas[(estaciones_completas$INDICATIVO %in% estaciones_meteo$estacionesrecientes), ]

datos_5_finales <- rename(datos_5_finales, Pmedia = Pm)
clima_todo <- as.data.frame(rbind(datos_5_finales, estaciones_completas)) # Juntamos ambos data frames,

# Al filtrar lo cambia a caracter, por tanto, lo volvemos a poner a numerico, tanto Pm como Tmedia
clima_todo$Pmedia <- as.numeric(clima_todo$Pmedia)
clima_todo$Tmedia <- as.numeric(clima_todo$Tmedia)

# Ordenamos los datos segun el mes
clima_todo <- clima_todo[order(clima_todo$mes, clima_todo$periodo), ]
clima_todo$mes <- as.numeric(clima_todo$mes)

# Funcion para que el eje de la y muestre valores redondeados, la usamos en el gráfico
int_breaks <- function(x, n = 5) {
  l <- pretty(x, n)

```



```

  1[abs(1 %% 1) < .Machine$double.eps ^ 0.5]
}

estacionID <- unique(clima_todo$INDICATIVO) # códigos de estación

# Bucle para crear los graficos
for (i in estacionID){

  temp <- clima_todo[clima_todo$INDICATIVO == i,]
  nombre <- temp[1, 2]

  precipitacion = # gráfico de precipitación
  ggplot(temp, aes(x = mes, y = Pmedia, group = periodo, colour = periodo)) +
    geom_line(size = 1) +
    scale_color_manual(values = c("#000000", "#FFFF33", "#FFCC00", "#FF9900", "#FF6600", "#FF0000", "#990000")) +
    geom_point(size = 3, shape = 21, fill = "white") +
    scale_y_continuous(breaks = int_breaks) + # esto nos da error
    labs(x = "mes", y = "Precipitación acumulada media (mm)", colour="periodo", ) +
    scale_x_continuous(breaks = c(1:12),
                      labels = c('E', 'F', 'M', 'A', 'M', 'J', 'J', 'A', 'S', 'O', 'N', 'D')) +
    ggtitle(clima_todo[clima_todo$INDICATIVO == i, ]$NOMBRE) +
    theme(plot.title = element_text(hjust = 0.5), legend.title = element_text(hjust = 0.5))

  name <- paste('prec_', i, '.png', sep = '') # nombre del gráfico
  ggsave(name, precipitacion) # exportación

  temperatura = # gráfico de temperatura
  ggplot(temp, aes(x = mes, y = Tmedia, group = periodo, colour = periodo)) +
    geom_line(size = 1) +
    scale_color_manual(values = c("#000000", "#FFFF33", "#FFCC00", "#FF9900", "#FF6600", "#FF0000", "#990000")) +
    geom_point(size = 3, shape = 21, fill = "white") +
    scale_y_continuous(breaks = int_breaks) +
    scale_x_continuous(breaks = c(1:12),
                      labels = c('E', 'F', 'M', 'A', 'M', 'J', 'J', 'A', 'S', 'O', 'N', 'D')) +
    labs(x = "mes", y = "Temperatura media (°C)", colour = "periodo") +
    ggtitle(clima_todo[clima_todo$INDICATIVO == i, ]$NOMBRE) +
    theme(plot.title = element_text(hjust = 0.5), legend.title = element_text(hjust = 0.5))

  name <- paste('temp_', i, '.png', sep = '') # nombre del gráfico
  ggsave(name, temperatura) # exportación
}

```

```

## Saving 6.5 x 4.5 in image
## Saving 6.5 x 4.5 in image

```