Análisis de la evolución de la incidencia de la COVID-19 en España

Juan Matorras Díaz-Caneja

23/11/2020

Introducción

Este es un ejercicio básico de análisis de los datos de la incidencia de la COVID-19 en España a lo largo de 2020. Habiendo la cantidad de informes y herramientas para el análisis de los datos sobre la incidencia de la COVID-19 que ya existen, este documento no pretende aportar nada singularmente nuevo y su razón de ser no es otra que poner en práctica y profundizar por mi parte en el aprendizaje de las técnicas de análisis de datos y el lenguaje R que he iniciado en la segunda mitad de septiembre de 2020.

Los datos de partida son los publicados por el Gobierno de España en la web **datos.gob.es** a través del siguiente enlace: https://datos.gob.es/es/catalogo/e05070101-evolucion-de-enfermedad-por-el-coronavirus-covid-19.

El grueso del informe se centra sobre los totales en España agregando los datos disponibles por Comunidades Autónomas, aunque también se muestran información de las CCAA de Madrid y Cantabria. La razón de la selección de estas dos comunidades y no otras es simple y llanamente que nací y crecí en la última, mantenniendo allí vínculos familiares y de amistad, mientras que en la primera he pasado prácticamente la mitad de mi vida, sigo viviendo en ella y previsiblemente así seguirá siendo en los próximos años.

Proceso metodológico y software utilizado

El archivo de datos no ha sido sometido a ningún tipo de modificación o alteración previa y su manipulación en este análisis es el mínimo imprescindible para permitir el tratamiento de los datos y obtención de resultados.

```
DatosCompletos <- read.csv(file.path("data", "datos_ccaas.csv"))
## Descartamos el método de detección (PCR, antígenos,...)
DatosCCAAs <- DatosCompletos[, 1:3]
## Adecuación formato fechas para su correcta lectura
DatosCCAAs$fecha <- as.Date(DatosCCAAs$fecha, format = "%Y-%m-%d")
## Agrupación de información de CCAA para datos a nivel nacional
Datos <- DatosCCAAs %>% group_by(fecha) %>% summarise(num_casos = sum(num_casos),
                                                      .groups = 'drop')
```

La fecha y hora de descarga de los datos que han sido utilizados para las tablas y gráficos incluídos en este informe ha sido (aaaa-mm-dd hh:mm:ss): **2020-11-23 18:35:23**

El análisis se ha llevado a cabo utilizando el software libre para análisis estadístico **R**, versión 4.0.2. (1)

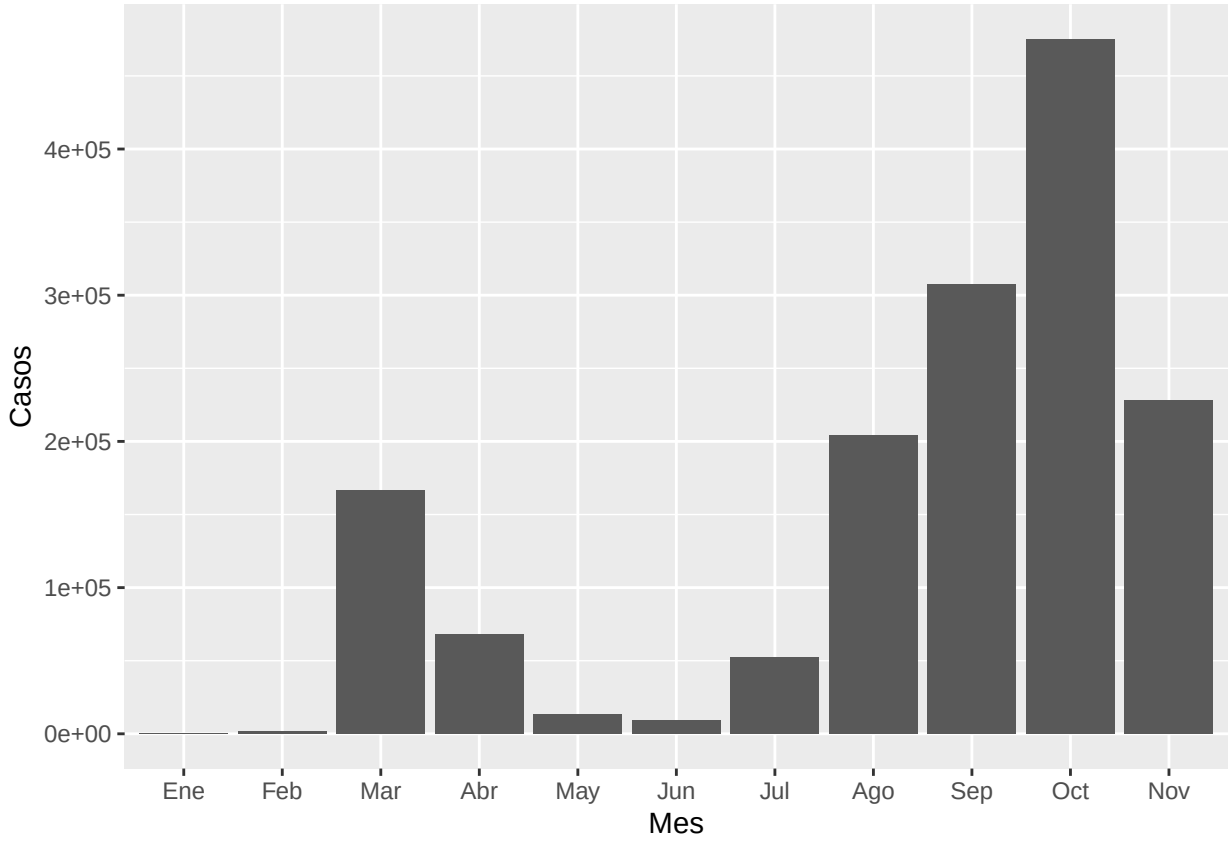
Se ha hecho uso también de los paquetes complementarios:

- **lubridate** para facilitar el manejo de fechas. (2)
- **knitr** para mejorar la apariencia de tablas. (3)
- **tydiverse** por los paquetes que incluye para ayudar en la extracción de la información y los gráficos mejorados de **ggplot2**. (4)

Incidencia mensual y número total de casos detectados desde el inicio de 2020

La evolución de número de casos notificados por meses se reflejan en la siguiene tabla y el gráfico que la acompaña:

```
CasosMensuales <- tapply(Datos$num_casos, month(Datos$fecha), sum)
TablaCasosMensuales <- data.frame(head(Meses, length(CasosMensuales)),
        format(CasosMensuales, big.mark = ".", decimal.mark = ","))
TotalCasosOrigen <- sum(Datos$num_casos)
PorcentajePoblacion <- paste(format(TotalCasosOrigen / poblESP *100, digits = 4,
    decimal.mark = ","), "%")
dfCasosMensuales <- data.frame(Mes = head(Meses, length(CasosMensuales)),
                               Casos =  CasosMensuales)
dfCasosMensuales$Mes <- factor(dfCasosMensuales$Mes, levels = dfCasosMensuales$Mes)
d <- ggplot(dfCasosMensuales, aes(Mes, Casos))
d + geom_col()
```




```
colnames(TablaCasosMensuales) <- c("Mes", "Nº casos mensuales")
kable(TablaCasosMensuales, align = "r")
```

Mes	Nº casos mensuales
Ene	69
Feb	1.743
Mar	166.742
Abr	67.775
May	13.249
Jun	9.353
Jul	52.094

Mes	Nº casos mensuales
Ago	203.838
Sep	307.459
Oct	475.257
Nov	227.994

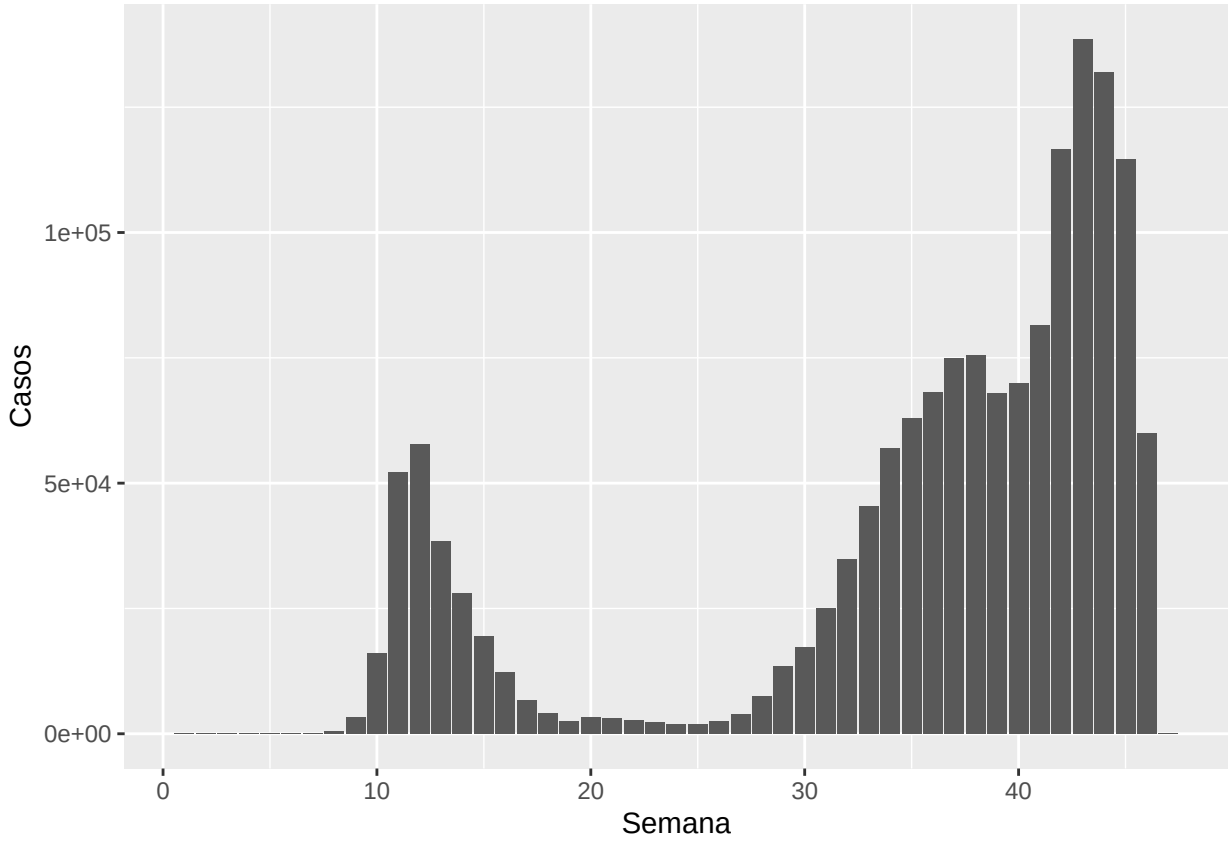
El número total de casos acumulados desde el 1 de enero de 2020 hasta la fecha indicada en el punto anterior según los datos oficiales disponibles en ese momento ascienden a un total de **1.525.573**.

Considerando una población en España de **47,33** millones de personas según los datos publicados por el INE (Instituto Nacional de Estadística) correspondientes al inicio del año 2020, el porcentaje de contagio de la población es del **3,223 %** hasta la fecha.

Incidencia semanal

- Evolución de número de casos identificados por semanas:

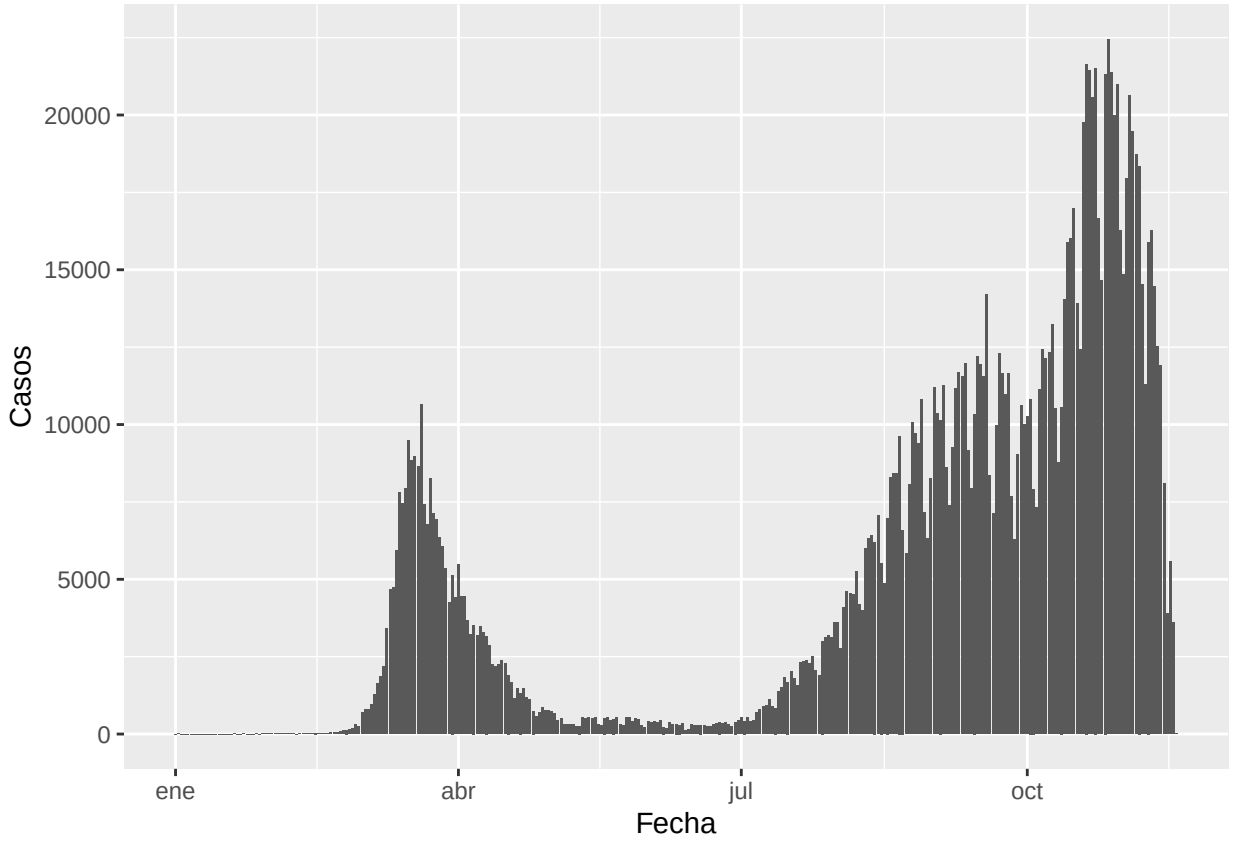
```
CasosSemanales <- tapply(Datos$num_casos, week(Datos$fecha), sum)
dfCasosSemanales <- data.frame(Semana = 1:length(CasosSemanales), Casos =
                                    CasosSemanales)
d <- ggplot(dfCasosSemanales, aes(Semana, Casos))
d + geom_col()
```



Incidencia diaria

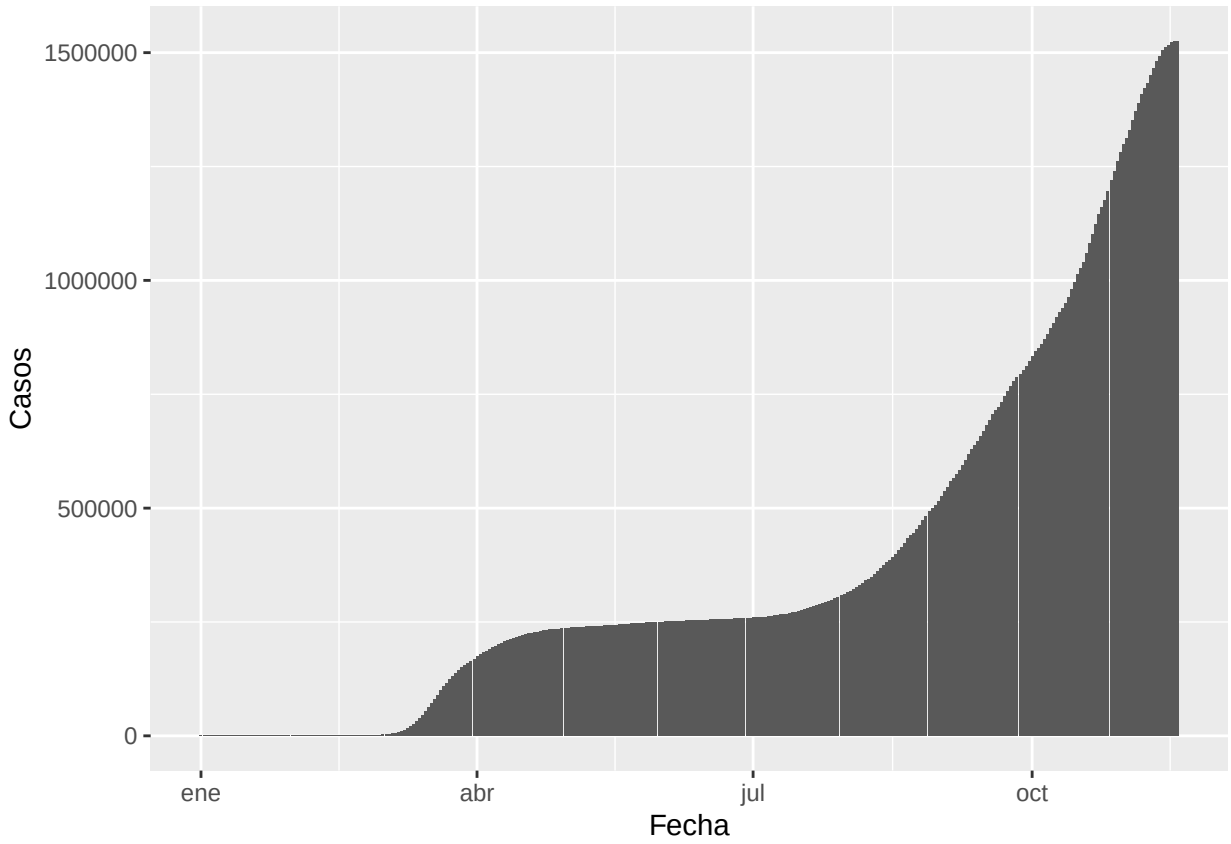
- Curva epidémica de los casos notificados por días:

```
dfCasosDiarios <- data.frame(Fecha = Datos$fecha, Casos = Datos$num_casos)
d <- ggplot(dfCasosDiarios, aes(Fecha, Casos))
d + geom_col()
```



- Gráfico de casos acumulados a origen por días:

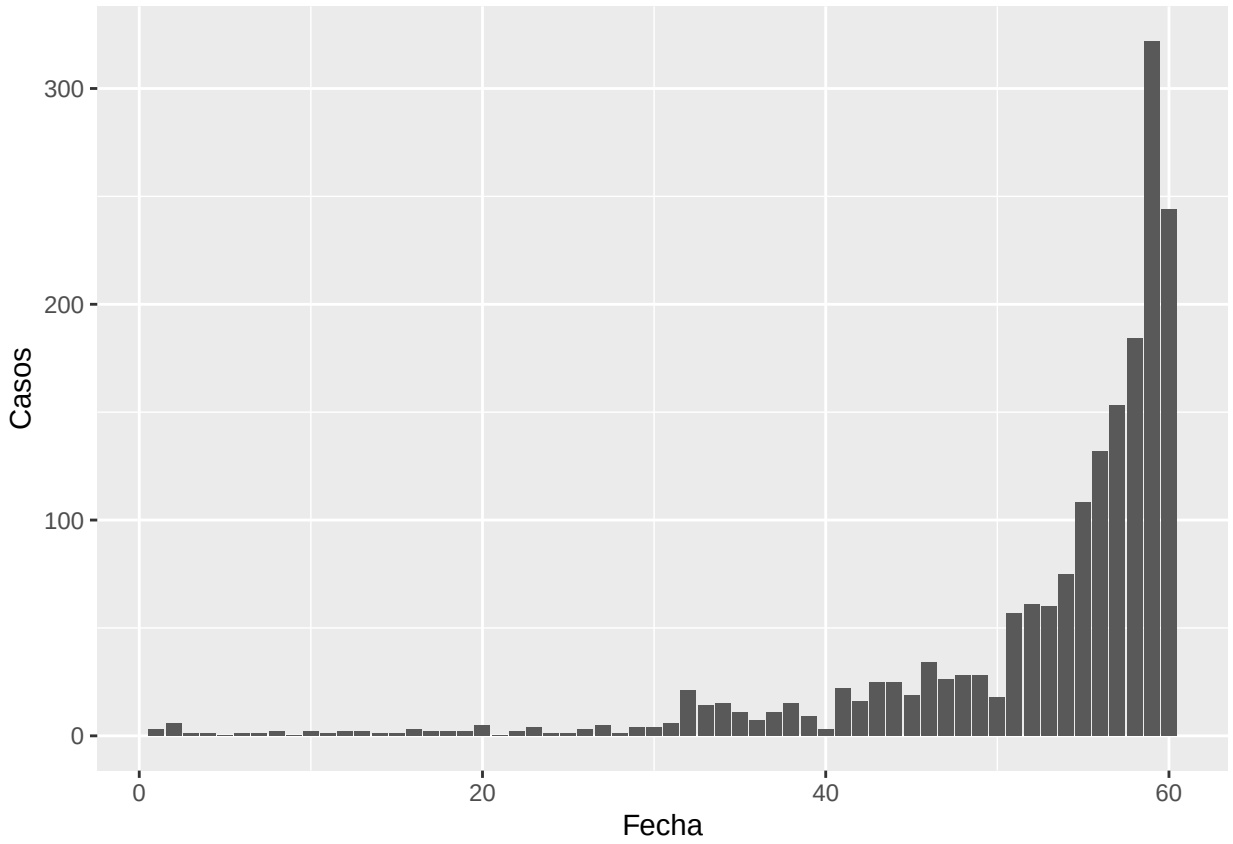
```
CasosDiariosAcumul <- cumsum(dfCasosDiarios$Casos)
dfCasosDiariosAcumul <- data.frame(Fecha = Datos$fecha, Casos = CasosDiariosAcumul)
d <- ggplot(dfCasosDiariosAcumul, aes(Fecha, Casos))
d + geom_col()
```



Detalle del número de casos en los dos primeros meses de 2020

- Evolución diaria del número de casos durante los dos primeros meses del año:

```
PeriodoEF <- seq.Date(
  from = as.Date("2020-01-01"),
  to = as.Date("2020-02-29"),
  by = "day")
DatosEF <- subset(Datos, fecha %in% PeriodoEF)
CasosDiariosEF <- tapply(DatosEF$num_casos, yday(DatosEF$fecha), sum)
dfCasosDiariosEF <- data.frame(Fecha = 1:length(CasosDiariosEF), Casos =  CasosDiariosEF)
TotalCasosEF <- sum(DatosEF$num_casos)
d <- ggplot(dfCasosDiariosEF, aes(Fecha, Casos))
d + geom_col()
```



Los casos reportados totales a lo largo de esos dos meses son **1.812**, si bien es claro que se produce una acusada inflexión en la pendiente de crecimiento a partir del día 50.

```
Periodo1_50 <- seq.Date(
  from = as.Date("2020-01-01"),
  to = as.Date("2020-02-19"),
  by = "day")
Periodo51_60 <- seq.Date(
  from = as.Date("2020-02-20"),
  to = as.Date("2020-02-29"),
  by = "day")
Datos1_50 <- subset(Datos, fecha %in% Periodo1_50)
Datos51_60 <- subset(Datos, fecha %in% Periodo51_60)
TotalCasos1_50 <- sum(Datos1_50$num_casos)
TotalCasos51_60 <- sum(Datos51_60$num_casos)
```

Siendo así que el desglose de número agregado de casos identificados en dichos primeros 50 días y los siguientes 10 días queda de la siguiente manera:

- Periodo 1-50: 416
- Periodo 51-60: 1.396

En 10 días se detectan **3,4** veces los casos que se habían producido en los 50 días anteriores.

Subsiguiente evolución durante la primera quincena de marzo

En este apartado analizamos cómo continúa desarrollándose la propagación de la pandemia a principos del mes de marzo, estableciendo por su relevancia en lo ocurrido en España durante esos días dos periodos de tiempo diferenciados, del 1 al 8 y del 9 al 15.

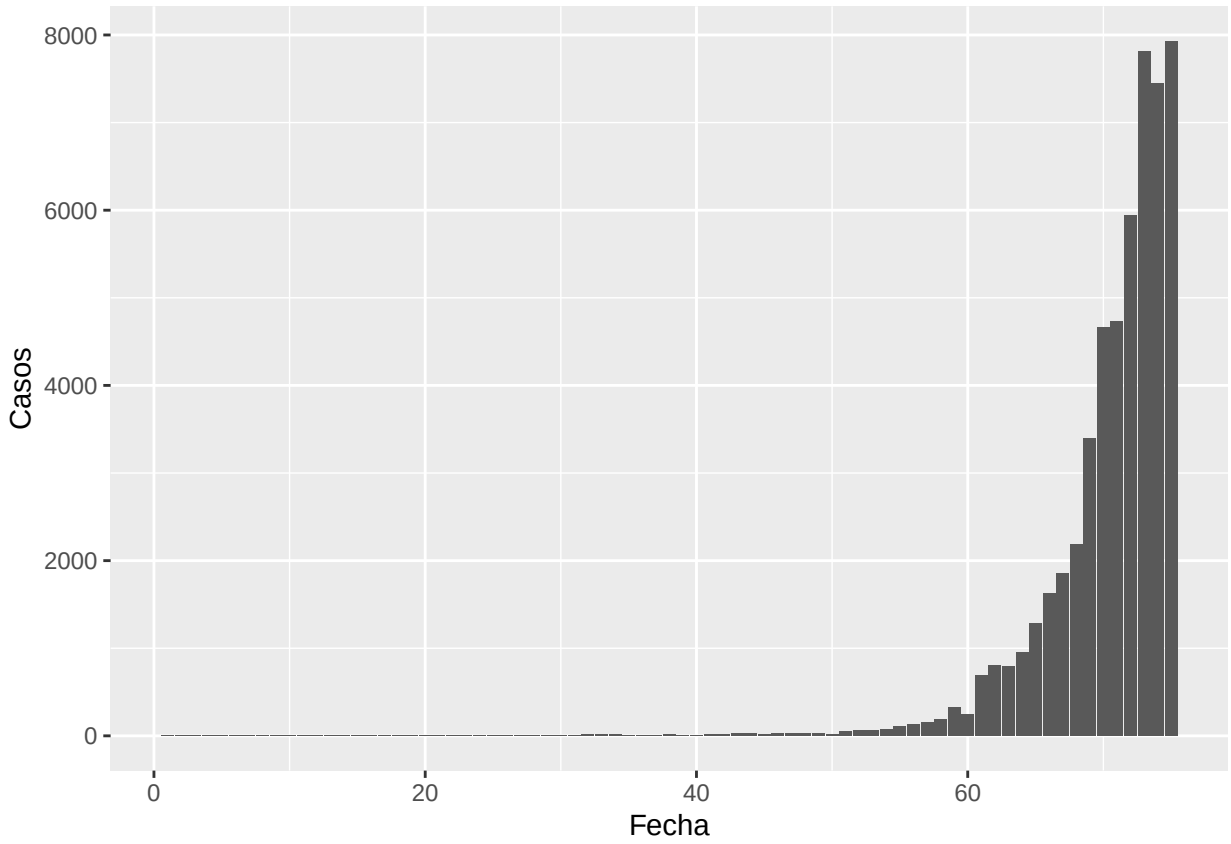
```
Periodo1_8mar <- seq.Date(
  from = as.Date("2020-03-01"),
  to = as.Date("2020-03-08"),
  by = "day")
Periodo9_15mar <- seq.Date(
  from = as.Date("2020-03-09"),
  to = as.Date("2020-03-15"),
  by = "day")
Datos1_8mar <- subset(Datos, fecha %in% Periodo1_8mar)
Datos9_15mar <- subset(Datos, fecha %in% Periodo9_15mar)
CasosDiarios1_8mar <- tapply(Datos1_8mar$num_casos, yday(Datos1_8mar$fecha), sum)
CasosDiarios9_15mar <- tapply(Datos9_15mar$num_casos, yday(Datos9_15mar$fecha), sum)
TotalCasos1_8mar <- sum(Datos1_8mar$num_casos)
TotalCasos9_15mar <- sum(Datos9_15mar$num_casos)
```

En los primeros ocho días de marzo la progresión diaria de nuevos casos siguió disparándose, resultando un total de **10.187** casos a añadir al total anterior, siendo éstos **5,6** veces los registrados a lo largo de todo enero y febrero.

Durante los siguientes siete días, del 9 al 15 de marzo, los casos a sumar fueron **41.933**, lo que supone **4,1** veces los notificados en los 8 primeros días del mes.

- Gráfico del número de casos diarios desde el 1 de enero hasta el 15 de marzo de 2020:

```
CasosDiariosHasta15mar <- c(CasosDiariosEF, CasosDiarios1_8mar, CasosDiarios9_15mar)
dfCasosDiariosHasta15mar <- data.frame(Fecha = 1:length(CasosDiariosHasta15mar), Casos =
                                       CasosDiariosHasta15mar)
d <- ggplot(dfCasosDiariosHasta15mar, aes(Fecha, Casos))
d + geom_col()
```



Incidencia acumulada por 100.000 habitantes en los 14 días previos a la declaración del estado de alarma del 14 de marzo

Pasemos ahora a calcular la incidencia acumulada por cada 100.000 habitantes en los 14 días previos a la declaración del estado de alarma que tuvo efecto

```
PeriodoPrevioEstadoAlarma <- seq.Date(
  from = as.Date("2020-02-29"),
  to = as.Date("2020-03-13"),
  by = "day")
DatosPrevioEstadoAlarma <- subset(Datos, fecha %in% PeriodoPrevioEstadoAlarma)
IncidAcumPrevioEstadoAlarma <- sum(DatosPrevioEstadoAlarma$num_casos)/poblESP*1E+05
```

Tomando esos 14 días previos, es decir, entre el 29 de febrero y el 13 de marzo, la incidencia acumulada por cada 100.000 habitantes, con el mismo dato de población presentado más arriba fue de **78** casos/100.000 hab.

Contrasta este valor de forma muy llamativa con los límites que se han estado manejando en la segunda ola de infecciones, donde se ha hablado de 200, 500 e incluso 1.000 casos/100.000 hab.

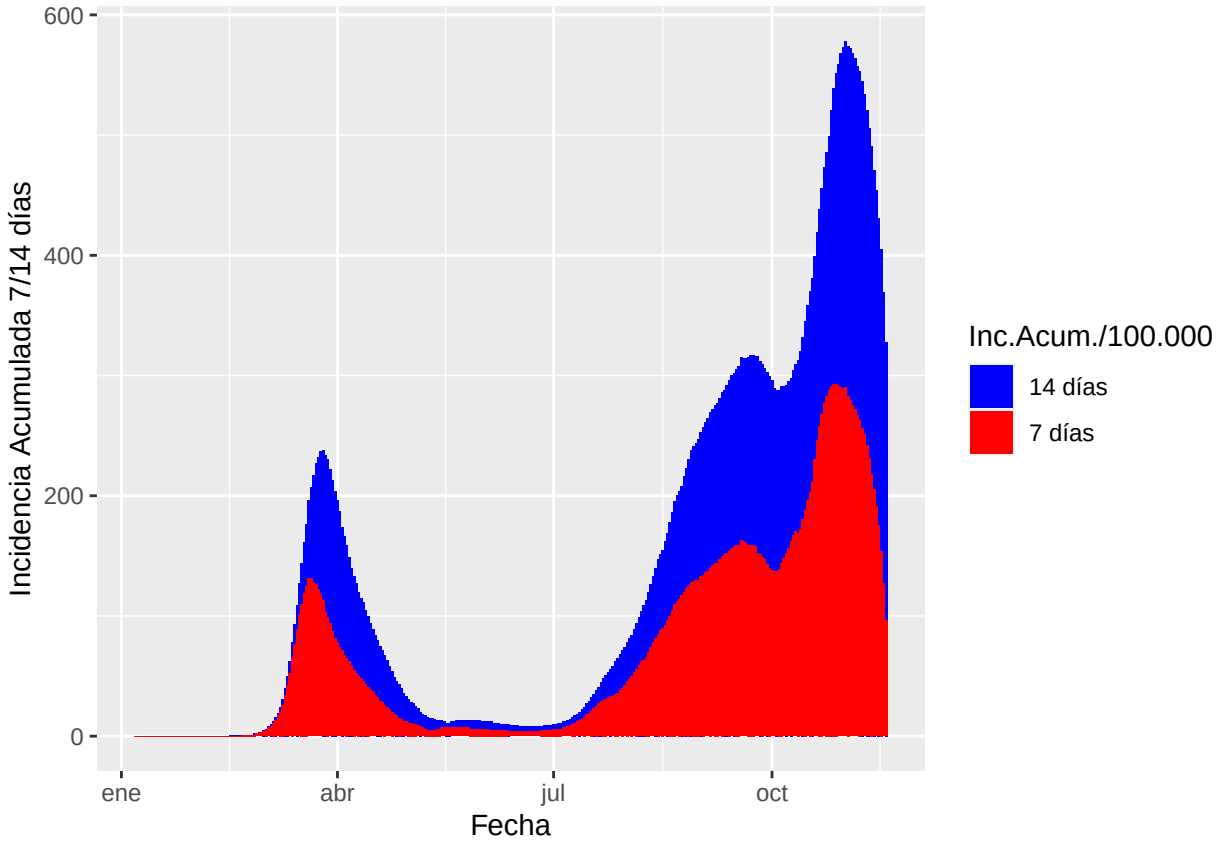
Evolución de la incidencia acumulada a lo largo de todo el año

En el siguiente gráfico se representan las incidencias acumuladas por cada 100.000 habitantes correspondientes a periodos de 14 y 7 días:

```
Fechas <- seq.Date(as.Date("2020-01-01"), by = "day", length.out =
                           length(dfCasosDiarios$Fecha))
UltFecha <- Fechas[length(Fechas)]
stopifnot(UltFecha == max(Datos$fecha))  ## Prueba interna consistencia datos
```

```r
## Inicialización de variables
InicioPeriodo14 <- as.Date("2020-01-01")
FinPeriodo14 <- InicioPeriodo14 + 13
IncidAcum14 <- data.frame(Fecha=as.Date(character()), IA14=integer())
IA14aux <- as.integer()
InicioPeriodo7 <- as.Date("2020-01-01")
FinPeriodo7 <- InicioPeriodo7 + 6
IncidAcum7 <- data.frame(Fecha=as.Date(character()), IA7=integer())
IA7aux <- as.integer()

## Incidencia acumulada 14 días
while (FinPeriodo14 <= UltFecha) {
        IA14aux <- 0
        Periodo14 <- seq.Date(
                from = InicioPeriodo14,
                to = FinPeriodo14,
                by = "day")
        DatosIA14 <- subset(Datos, fecha %in% Periodo14)
        IA14aux <- round(sum(DatosIA14$num_casos)/poblESP*1E+05, digits = 0)
        IncidAcum14 <- add_row(IncidAcum14, Fecha = FinPeriodo14, IA14 = IA14aux)
        InicioPeriodo14 <- InicioPeriodo14 + 1
        FinPeriodo14 <- FinPeriodo14 + 1
}

## Incidencia acumulada 7 días
while (FinPeriodo7 <= UltFecha) {
        IA7aux <- 0
        Periodo7 <- seq.Date(
                from = InicioPeriodo7,
                to = FinPeriodo7,
                by = "day")
        DatosIA7 <- subset(Datos, fecha %in% Periodo7)
        IA7aux <- round(sum(DatosIA7$num_casos)/poblESP*1E+05, digits = 0)
        IncidAcum7 <- add_row(IncidAcum7, Fecha = FinPeriodo7, IA7 = IA7aux)
        InicioPeriodo7 <- InicioPeriodo7 + 1
        FinPeriodo7 <- FinPeriodo7 + 1
}

## Gráficos de Incidencia acumulada por separado
## d <- ggplot(IncidAcum14, aes(Fecha, IA14)) + labs(y = "Incidencia Acumulada 14 días")
## d + geom_col()
## e <- ggplot(IncidAcum7, aes(Fecha, IA7)) + labs(y = "Incidencia Acumulada 7 días")
## e + geom_col()

## Gráficos de Incidencia Acumulada superpuestos
        ## Combinaión de ambos data frames en IncidAcum7_14
IncidAcum7_14 <- add_column(IncidAcum7, IA14=c(rep(0, 7), IncidAcum14$IA14))
                ## 7 primeros valores deberían ser NA, 0 para evitar aviso
g <- ggplot(IncidAcum7_14, aes(Fecha)) + labs(y = "Incidencia Acumulada 7/14 días")
g <- g + scale_fill_manual(name="Inc.Acum./100.000",
                           values = c("14 días" = "blue", "7 días" = "red"))
g <- g + geom_col(aes(y=IA14, fill="14 días"))
g <- g + geom_col(aes(y=IA7, fill="7 días"))
```

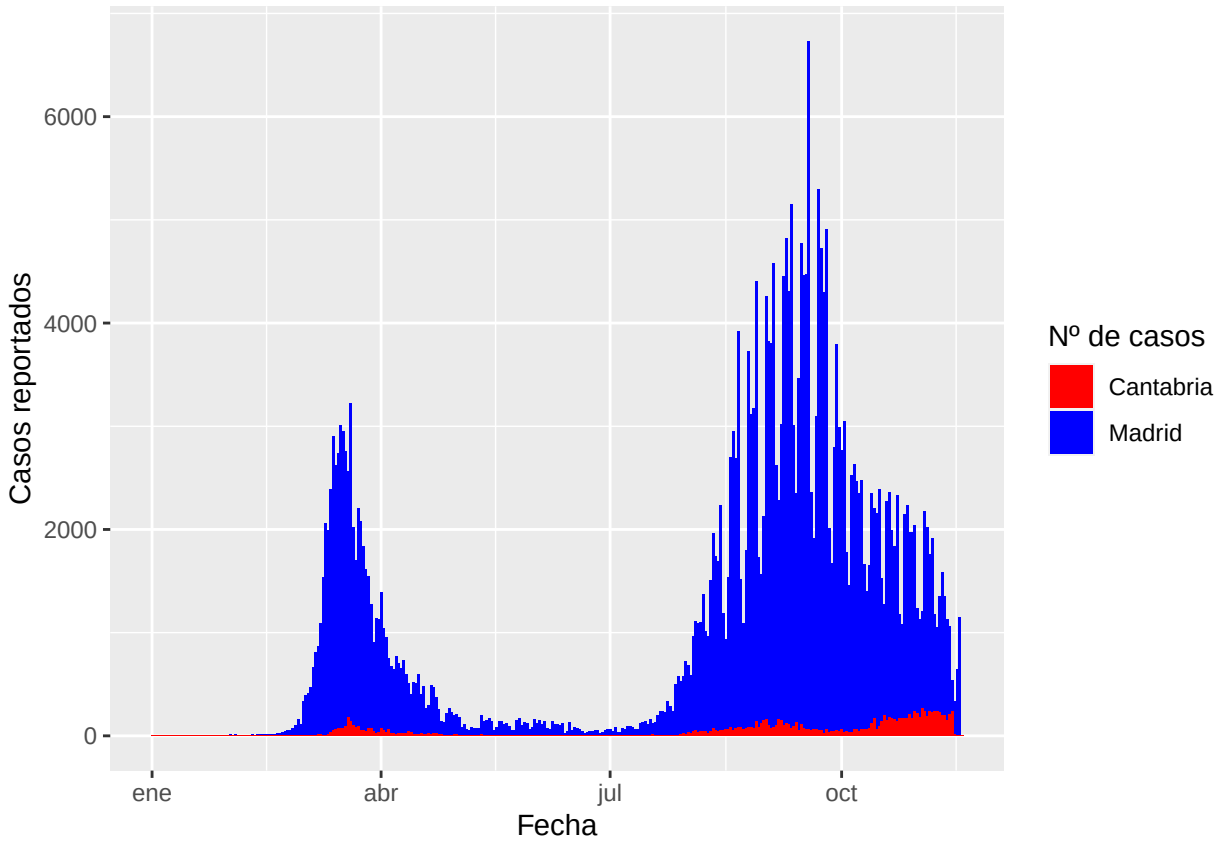
```
g
```



Comparación de la evolución del número de casos entre Cantabria y Madrid

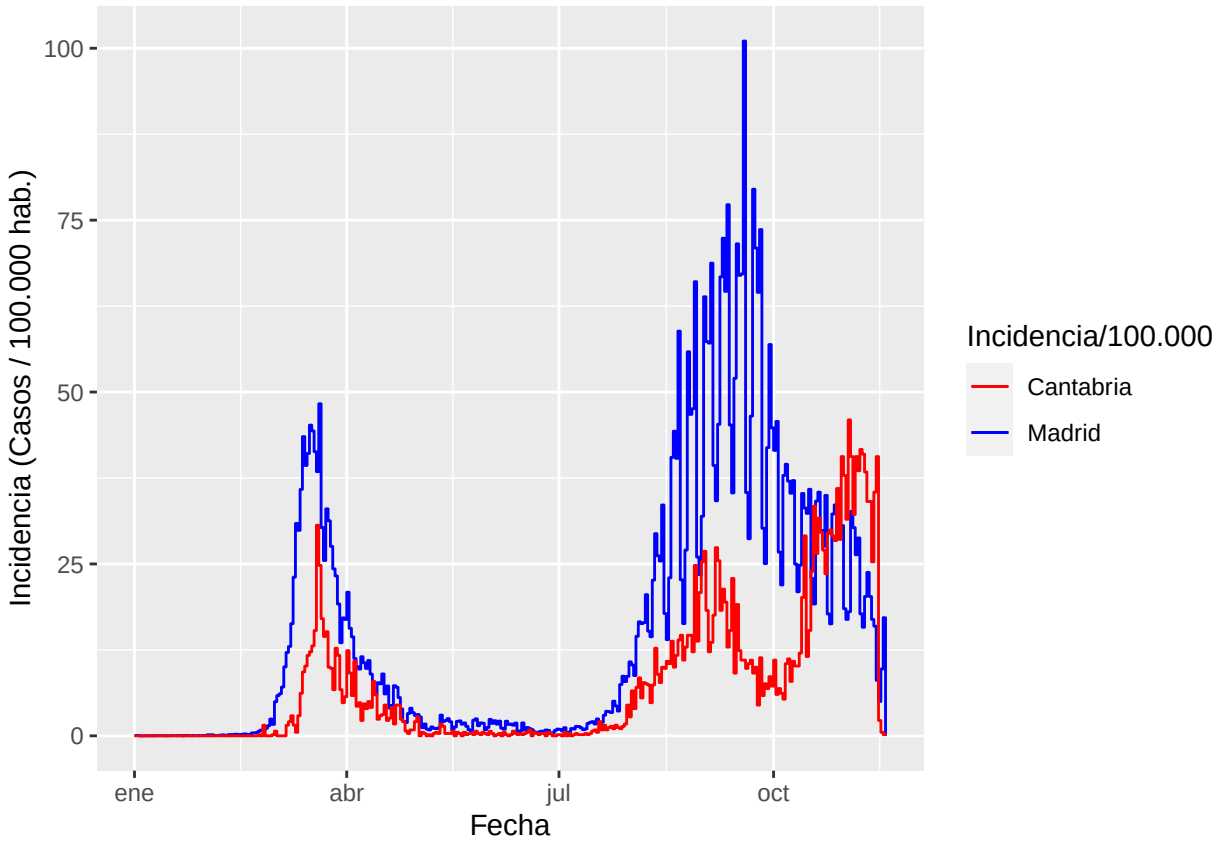
En el siguiente gráfico se compara la evolución de la enfermedad entre dos comunidades muy diferentes, Cantabria y la Comunidad Autónoma de Madrid:

```
## ISO CCAA: Cantabria = CB, Madrid = MD
CasosDiariosCantabria <- filter(DatosCCAAs, ccaa_iso == "CB")
CasosDiariosCantabria <- subset (CasosDiariosCantabria, select = -ccaa_iso)
colnames(CasosDiariosCantabria) <- c("Fecha", "Casos")
CasosDiariosMadrid <- filter(DatosCCAAs, ccaa_iso == "MD")
CasosDiariosMadrid <- subset (CasosDiariosMadrid, select = -ccaa_iso)
colnames(CasosDiariosMadrid) <- c("Fecha", "Casos")
CasosCantabriaMadrid <- add_column(CasosDiariosCantabria, CasosDiariosMadrid$Casos)
colnames(CasosCantabriaMadrid) <- c("Fecha", "CasosCANT", "CasosMAD")
g <- ggplot(CasosCantabriaMadrid, aes(Fecha)) + labs(y = "Casos reportados")
g <- g + scale_fill_manual(name="Nº de casos",
                           values = c("Madrid" = "blue", "Cantabria" = "red"))
g <- g + geom_col(aes(y=CasosMAD, fill="Madrid"))
g <- g + geom_col(aes(y=CasosCANT, fill="Cantabria"))
g
```



Como es lógico, los datos no son comparables en términos absolutos por la gran diferencia de población. Reprentamos ahora número de casos por cada 100.000 habitantes, con los datos de población en cada comunidad disponibles en el momento en el INE, que corresponden a 2019, reflejando los datos de Cantabria en color rojo y los de Madrid en azul:

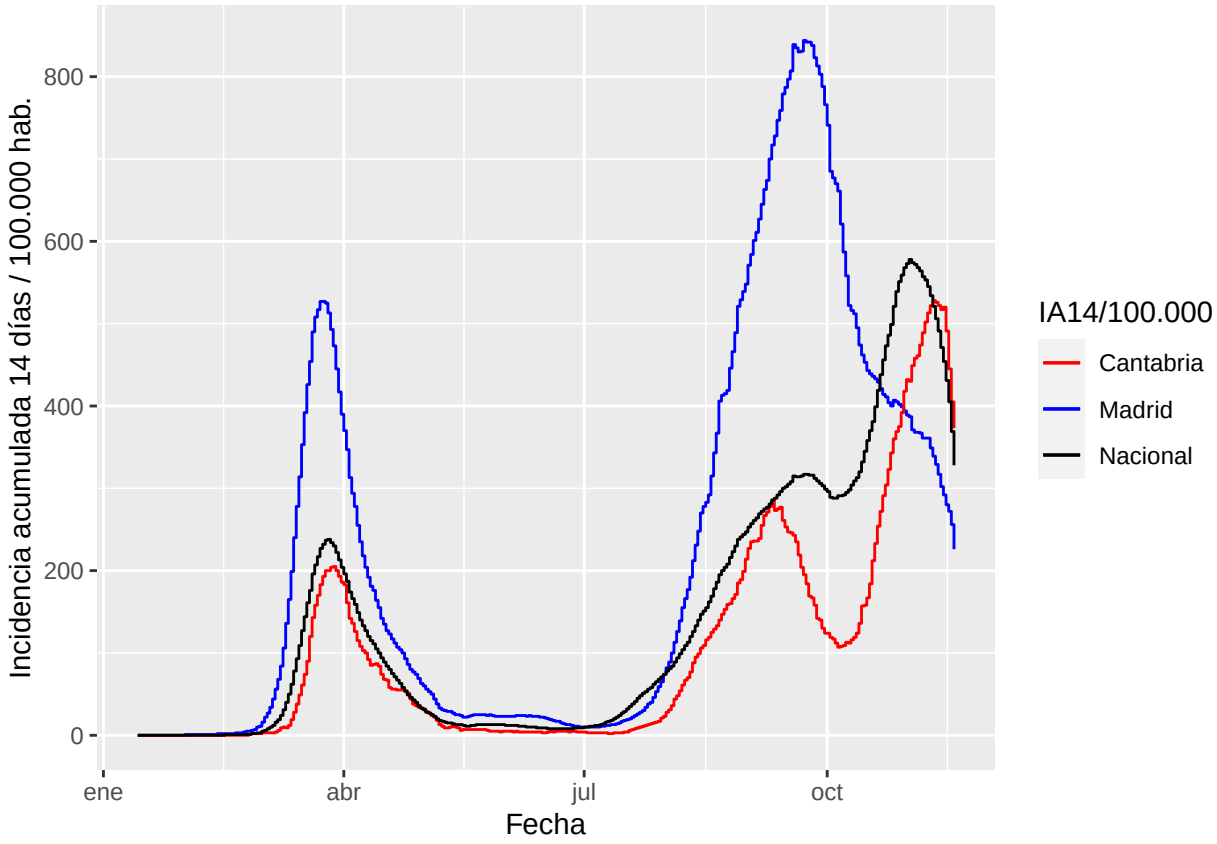
```
poblMAD <- 6.663E+06
poblCANT <- 581E+03
IncidenciaCantabriaMadrid <- data.frame(CasosCantabriaMadrid$Fecha,
                                        CasosDiariosCantabria$Casos/poblCANT*1E+05,
                                        CasosDiariosMadrid$Casos/poblMAD*1E+05)
colnames(IncidenciaCantabriaMadrid) <- c("Fecha", "IncidCANT", "IncidMAD")
g <- ggplot(IncidenciaCantabriaMadrid, aes(Fecha)) +
       labs(y = "Incidencia (Casos / 100.000 hab.)")
g <- g + scale_color_manual(name="Incidencia/100.000",
                            values = c("Madrid" = "blue", "Cantabria" = "red"))
g <- g + geom_step(aes(y=IncidMAD, colour="Madrid"))
g <- g + geom_step(aes(y=IncidCANT, colour="Cantabria"))
g <- g + theme(legend.justification = "center")
g <- g + theme(legend.position = "right")
g
```



Por completar la información comparativa entre ambas comunidades se adjunta también la incidencia acumulada en 14 días para ambas áreas geográficas, junto con la correspondiente al conjunto del territorio nacional:

```
InicioPeriodo14 <- as.Date("2020-01-01")
FinPeriodo14 <- InicioPeriodo14 + 13
IA14_CantabriaMadrid <- data.frame(Fecha=as.Date(character()), IA14_CANT=integer(),
                                   IA14_MAD=integer())
IA14aux <- as.integer()
while (FinPeriodo14 <= UltFecha) {
        IA14auxCANT <- 0
        IA14auxMAD <- 0
        Periodo14 <- seq.Date(
                from = InicioPeriodo14,
                to = FinPeriodo14,
                by = "day")
        DatosIA14_CantMad <- subset(IncidenciaCantabriaMadrid, Fecha %in% Periodo14)
        IA14auxCANT <- round(sum(DatosIA14_CantMad$IncidCANT), digits = 0)
        IA14auxMAD <- round(sum(DatosIA14_CantMad$IncidMAD), digits = 0)
        IA14_CantabriaMadrid <- add_row(IA14_CantabriaMadrid, Fecha = FinPeriodo14,
                                        IA14_CANT = IA14auxCANT, IA14_MAD = IA14auxMAD)
        InicioPeriodo14 <- InicioPeriodo14 + 1
        FinPeriodo14 <- FinPeriodo14 + 1
}
## Añadiendo columna con valores a nivel nacional
IA14_CantMadNac <- add_column(IA14_CantabriaMadrid, IA14_Nacional=IncidAcum14$IA14)
## Grádico
g <- ggplot(IA14_CantMadNac, aes(Fecha))
```

```
g <- g + labs(y = "Incidencia acumulada 14 días / 100.000 hab.")
g <- g + scale_color_manual(name = "IA14/100.000",
                            values = c("Madrid" = "blue", "Cantabria" =
                                                    "red", "Nacional" = "black"))
g <- g + geom_step(aes(y=IA14_MAD, color="Madrid"))
g <- g + geom_step(aes(y=IA14_CANT, color="Cantabria"))
g <- g + geom_step(aes(y=IA14_Nacional, color="Nacional"))
g <- g + theme(legend.justification = "center")
g <- g + theme(legend.position = "right")
g
```



.....

(1) R Core Team (2020). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL: https://www.R-project.org/

(2) Garrett Grolemund, Hadley Wickham (2011). Dates and Times Made Easy with lubridate. Journal of Statistical Software, 40(3), 1-25. URL: http://www.jstatsoft.org/v40/i03/

(3) Yihui Xie (2020). knitr: A General-Purpose Package for Dynamic Report Generation in R. R package version 1.30.

(4) Wickham et al., (2019). Welcome to the tidyverse. Journal of Open Source Software, 4(43), 1686, https://doi.org/10.21105/joss.01686