

# Taller 2: Estimación de $R(t)$ a partir de datos epidemiológicos

Agosto 12, 2016

## Introducción

Para este ejercicio, reproduciremos el análisis inicial del artículo “The epidemiology and transmissibility of Zika virus in Girardot and San Andres island, Colombia, September 2015 to January 2016”.

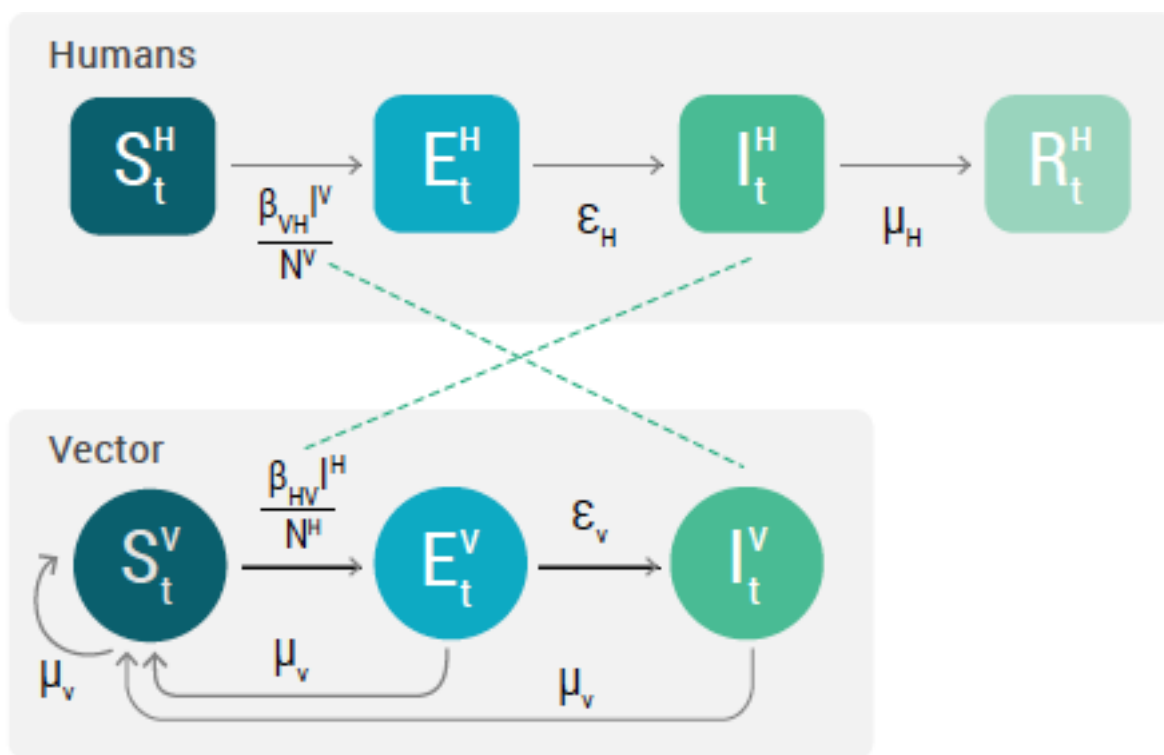


Figure 1:

Estas funciones se encuentran en el archivo `EstimationR`. El primer paso para este ejercicio es cargar este archivo. También es necesario cargar la librería “EpiEstim”.

```
library(EpiEstim) ## Libreria para estimar numeros reproductivos a partir de curvas epidemicas.
```

```
source("Taller.R") # Es necesario especificar el directorio donde se encuentra el archivo
```

## Epidemia de Zika Girardot y San Andres:

- La base de datos contiene las siguientes variables:
- DoS (Fecha de Inicio de sintomas)
- Date of Report (Fecha de notificacion)
- EW (Semana Epidemiologica)

- Age (Edad)
- Sex (Genero)

## Codigo paso a paso

### 1. Organizar el formato de las variables de tiempo

```
girardot.raw<-read.csv('Girardot_update.csv', sep=',', header=TRUE, as.is=TRUE)
colnames(girardot.raw)<-c('date_report_char', 'EW', 'year', 'age', 'uni_med', 'sex', 'pregnant',
                          'date_clinic_char', 'date_onset_char', 'case_type', 'date_birth_char')
girardot.raw$date_report<-as.Date(girardot.raw$date_report_char, '%m/%d/%Y')
girardot.raw$date_clinic<-as.Date(girardot.raw$date_clinic_char, '%m/%d/%Y')
girardot.raw$date_onset<-as.Date(girardot.raw$date_onset_char, '%m/%d/%Y')
girardot.raw$date_birth<-as.Date(girardot.raw$date_birth_char, '%m/%d/%Y')
girardot.raw$date_report_char<-girardot.raw$date_clinic_char<-girardot.raw$date_onset_char
<-girardot.raw$date_birth_char<-NULL
```

### 2. Codificar las variables en grupos

```
girardot.raw$gender <- as.numeric(girardot.raw$sex=='F')
girardot.raw$age_grp <- 1
girardot.raw$age_grp[girardot.raw$age>20 & girardot.raw$age<=50] <- 2
girardot.raw$age_grp[girardot.raw$age>50] <- 3
girardot.raw$child <- as.numeric(girardot.raw$age_grp==1)
girardot.raw$senior <- as.numeric(girardot.raw$age_grp==3)
girardot.raw$day <- as.numeric(girardot.raw$date_onset - min(girardot.raw$date_onset) + 1)
girardot.raw <- girardot.raw[order(girardot.raw$date_onset,
                                   girardot.raw$gender, girardot.raw$age_grp),]

girardot.raw$count <- 1
girardot <- aggregate(count~date_onset, data=girardot.raw, FUN='sum')
girardot$day <- as.numeric(girardot$date_onset - min(girardot$date_onset) + 1)
```

### 3. Grafica de curva epidemica y variables basicas.

```
#Epidemic curve
counts <- table(girardot.raw$date_onset)
barplot(counts, main="Epidemic curve Zika, Girardot",
        xlab="Date of onset of Symptoms")
```

```
#plot.epicurve(girardot, 'Girardot, Colombia')
girardot.strata <- aggregate(count~day+gender+child+senior, data=girardot.raw, FUN='sum')

demo.girardot.raw<-read.csv('pop_Girardot.csv', sep=',', header=TRUE, as.is=TRUE)
demo.girardot.raw$child<-as.numeric(demo.girardot.raw$agegrp %in%
c('0-4', '5-9', '10-14', '15-19'))
demo.girardot.raw$senior<-as.numeric(demo.girardot.raw$agegrp %in%
c('50-54', '55-59', '60-64', '65-69', '70-74', '75-79', '>80'))
demo.male <- aggregate(male~child+senior, data=demo.girardot.raw, FUN='sum')
colnames(demo.male) <- c('child', 'senior', 'total')
demo.male$gender <- 0
demo.female <- aggregate(female~child+senior, data=demo.girardot.raw, FUN='sum')
```

```
colnames(demo.female) <- c('child', 'senior', 'total')
demo.female$gender <- 1
demo.girardot <- rbind(demo.male, demo.female)
save(girardot, girardot.strata, demo.girardot, 'girardot.RData')
load('girardot.RData')
```

Describe la curva epidémica. Como es la distribución por género de los casos de Zika? Como es la distribución por edad? Cual es la media de edad? la mediana? Porque es importante saber esto? Hay diferencias por edad y género?

Para correr el modelo necesitamos estimar los parámetros y para eso vamos a usar la función `EstimationR`.

- Intervalo serial: ('DiscrSI') Calcula la distribución de intervalo serial asumiendo una media y la desviación estándar dada.

- Tiene tres parámetros:  $k$  (entero positivo en el cual se espera que este la distribución).
- $\mu$  (Número positivo que representa la media de la distribución Gamma).
- $\sigma$  (Número real dada la desviación estándar de la distribución Gamma).

```
DiscrSI<-function(k,mu,sigma)
{if(sigma<0){stop("sigma must be >=0.")}
a <- ((mu-1)/sigma)^2
b <- sigma^2/(mu-1)
CDFGamma<-function(k,a,b){return(pgamma(k,shape=a,scale=b))}
res <- k*CDFGamma(k,a,b)+(k-2)*CDFGamma(k-2,a,b)-2*(k-1)*CDFGamma(k-1,a,b)
res <- res+a*b*(2*CDFGamma(k-1,a+1,b)-CDFGamma(k-2,a+1,b)-CDFGamma(k,a+1,b))
res <- max(0,res)
return(res)}
```

- Infectividad General: ('OverallInfectivity'): Calcula la infectividad general, en cada periodo de tiempo y depende de los individuos infectados previamente.

```
OverallInfectivity <-function (I,SI.Distr)
{if(is.vector(I)==FALSE){stop("I must be a vector.")}
T<-length(I) for(i in 1:T){if(I[i]<0){stop("I must be a positive vector.")}}
if(is.vector(SI.Distr)==FALSE){stop("SI.Distr must be a vector.")} if(SI.Distr[1]!=0)
{stop("SI.Distr[1] needs to be 0.")}
if(length(SI.Distr)>1){for(i in 2:length(SI.Distr)){if(SI.Distr[i]<0)
{stop("SI.Distr must be a positive vector.")}}}
if(abs(sum(SI.Distr)-1)>0.01){stop("SI.Distr must sum to 1.")}
lambda <- vector()
lambda[2:length(I)] <- sapply(2:length(I), function(t) sum(SI.Distr[1:t]*I[t:1],na.rm=TRUE))
return(lambda)}
```

- Ahora si a estimar  $R$ : ('EstimateR'): Estima para cada intervalo de tiempo el número reproductivo instantáneo dada la incidencia de la curva epidémica y la distribución del intervalo serial.

```
fill.dates <- function(cases){cases.sort <- cases[order(cases$date_onset), ]tmp.date<-tmp.day<-NULL
for(i in 2:nrow(cases)){d <- as.numeric(cases.sort$date_onset[i]-cases.sort$date_onset[i-1])if(d>1)
{for(j in 1:(d-1)){tmp.date <- c(tmp.date, cases.sort$date_onset[i-1]+j)tmp.day
```

```

<- c(tmp.day, cases.sort$day[i-1]+j)}}}if(length(tmp.date) > 0){
data.tmp<-data.frame(tmp.date, rep(0,length(tmp.date)), tmp.day)
colnames(data.tmp) <- c('date_onset', 'count', 'day')
data.tmp$date_onset <- as.Date(data.tmp$date_onset, origin = "1970-01-01")
data.new <- rbind(cases, data.tmp)
data.new <- data.new[order(data.new$date_onset),]
return(data.new)} else return(cases.sort)}

fit.for.plot <- function(cases, start.time, stop.time, SI.mean, SI.sd, main.text)
{#cases<-girardot; start.time<-8:67; stop.time<-14:73; SI.mean<-c(10,15,20); SI.sd<-c(3,3,3); main.text<-
R.mean <- R.lower <- R.upper <- NULL for(i in 1:length(SI.mean))
{ fit<-EstimateR(cases$count, T.Start=start.time, T.End=stop.time, method=c("ParametricSI"),
Mean.SI=SI.mean[i], Std.SI=SI.sd[i], plot=FALSE) R.mean <- rbind(R.mean, fit$R[,3])
R.lower <- rbind(R.lower, fit$R[,5]) R.upper <- rbind(R.upper, fit$R[,11]) }
R.max <- max(R.upper, na.rm=TRUE)

#par(mfrow=c(2,1))
plot(stop.time, R.mean[1,], type='n', xlim=c(0,nrow(cases)), ylim=c(0, R.max),
xaxt='n', xlab='', ylab='R', main=main.text)
k<-which(!is.na(R.upper[1,]) & !is.na(R.lower[1,]))
polygon(c(rev(stop.time[k]), stop.time[k]), c(rev(R.upper[1,k]), R.lower[1,k])),
col = 'grey80', border = NA)
k<-which(!is.na(R.upper[2,]) & !is.na(R.lower[2,]))
polygon(c(rev(stop.time[k]), stop.time[k]), c(rev(R.upper[2,k]), R.lower[2,k])), col = 'grey80', border = NA)
k<-which(!is.na(R.upper[3,]) & !is.na(R.lower[3,]))
polygon(c(rev(stop.time[k]), stop.time[k]), c(rev(R.upper[3,k]), R.lower[3,k])),
col = 'grey80', border = NA)
lines(stop.time, R.mean[1,], lwd=1.5, lty=1, col='green')
lines(stop.time, R.mean[2,], lwd=1.5, lty=2, col='red')
lines(stop.time, R.mean[3,], lwd=1.5, lty=3, col='blue')
legend("topright", legend=paste('Mean SI: ', SI.mean, ' days', sep=''),lty=c(1,2,3),
lwd=c(2,2,2), col=c('green', 'red', 'blue'), text.width=30)
x<-1:nrow(cases)
x<-c(x-0.5, x+0.5)
k<-order(x)
x<-x[k]
y<-rep(cases$count,each=2)
plot(x,y, ylim=c(0, max(y)), type='l', lty=1, lwd=1.5, xlab='Day', ylab='Case Number')}

```

## 7. Ajuste del modelo

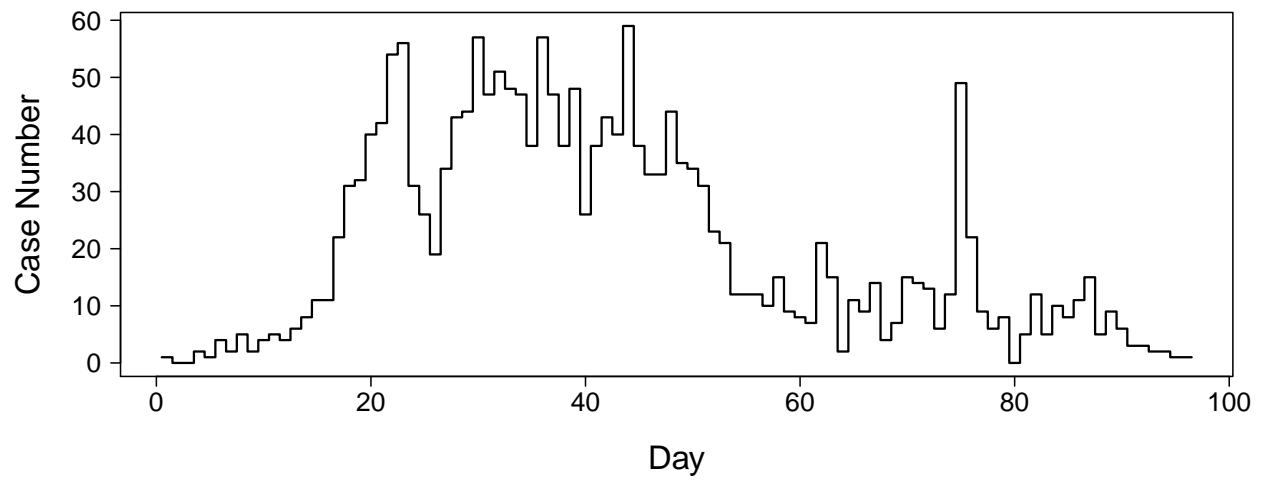
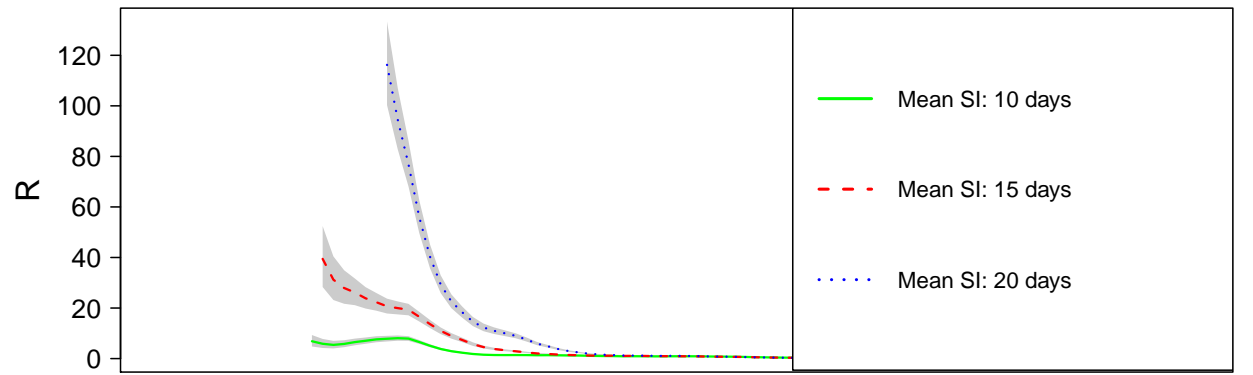
```

par(mfcol=c(2,2))
tmp <- fill.dates(girardot)
fit.for.plot(tmp, 8:84, 14:90, c(10,15,20), c(3,3,3), 'Girardot, Colombia')
tmp <- fill.dates(sanandres)
fit.for.plot(tmp, 8:134, 14:140, c(10,15,20), c(3,3,3), 'San Andres, Colombia')

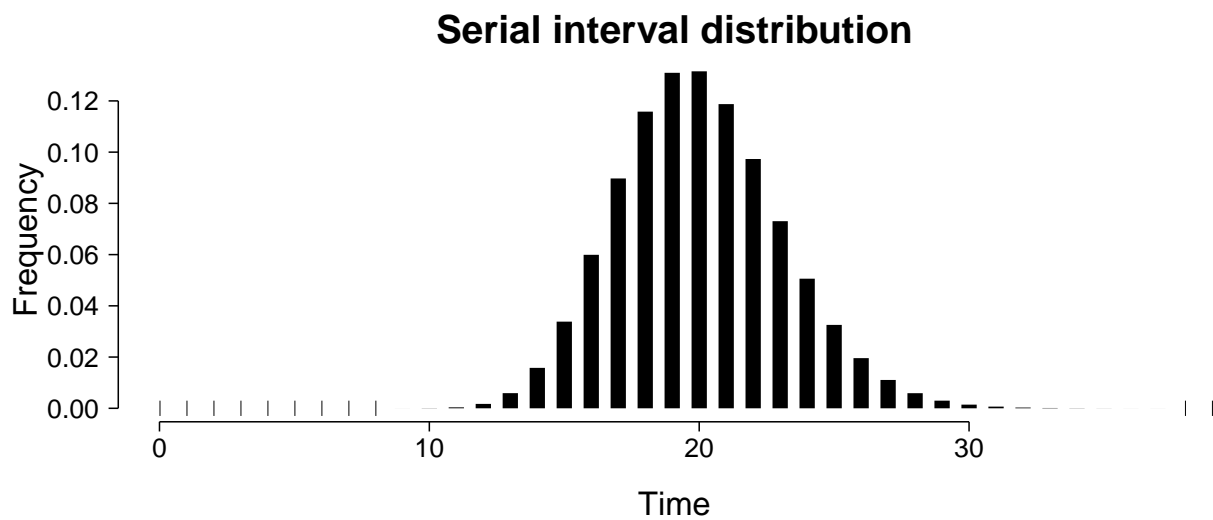
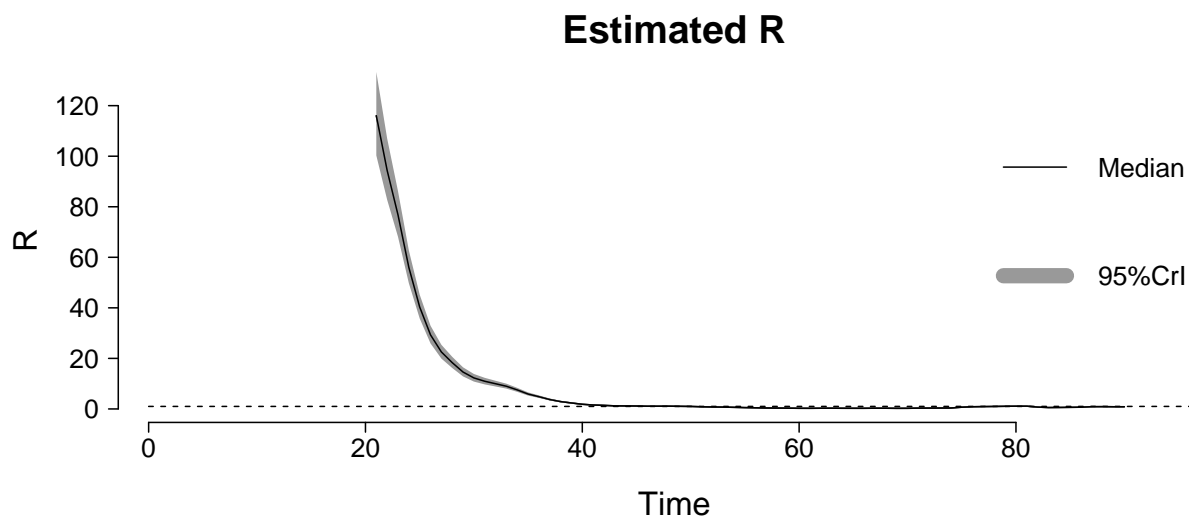
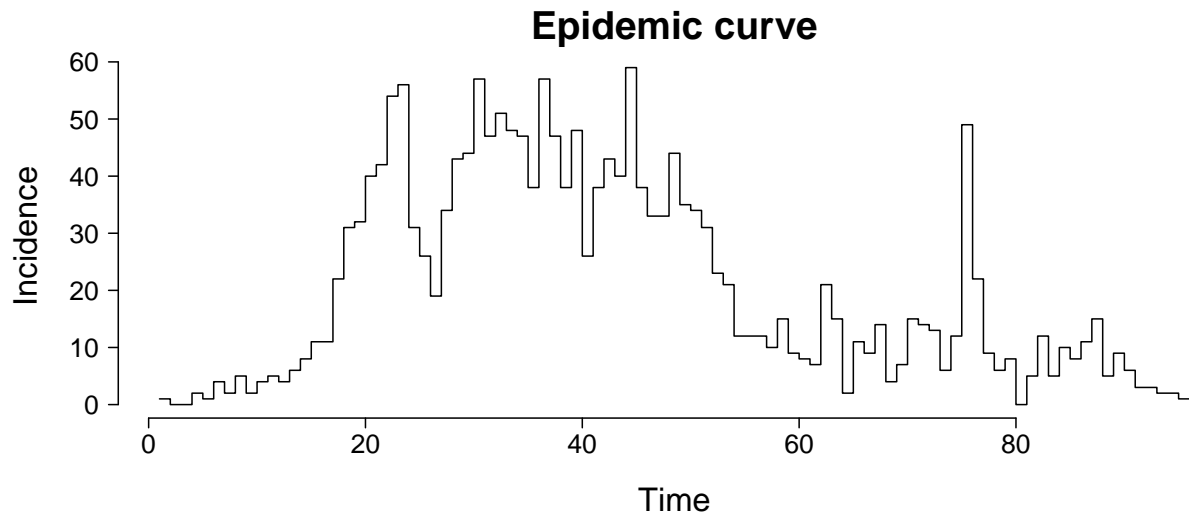
```

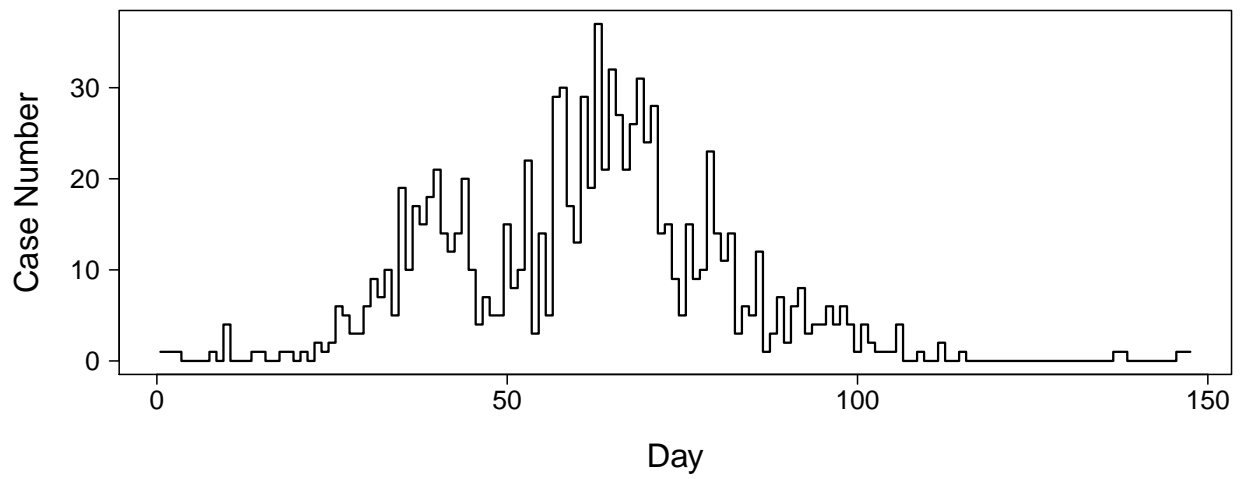
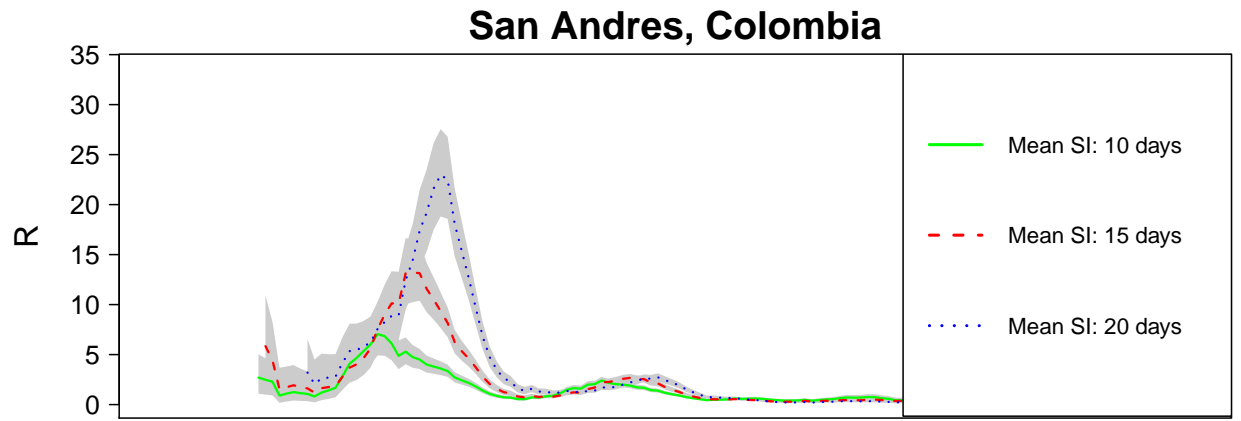
## 8. Comparemos las estimaciones y los resultados de Girardot y San Andres.

## Girardot, Colombia



Girardot:





San Andres:

