Estimate RSV onset and peak timing

Tutorial 2 for transition

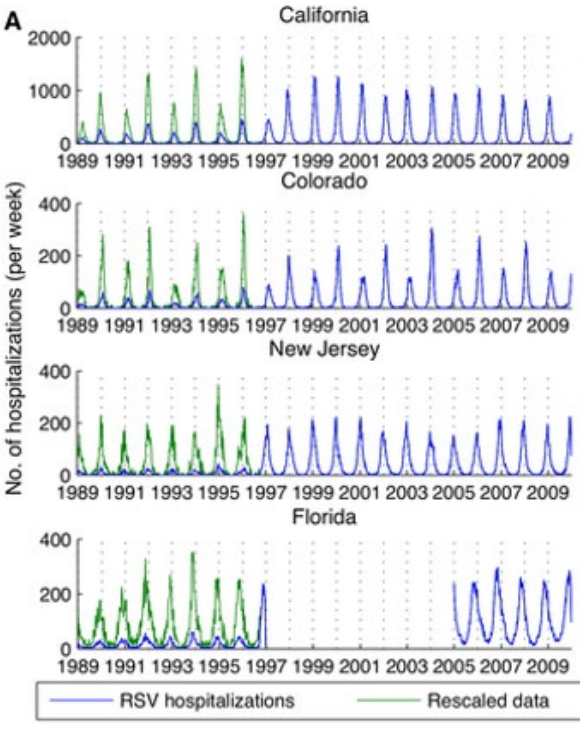
Gigi (Zhe Zheng)[1] Dan (Daniel Weinberger) Ginny (Virginia Pitzer)

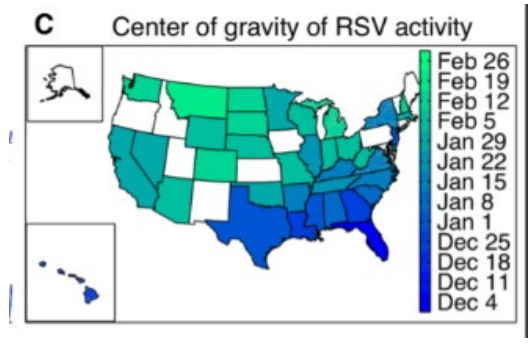
2023-01-04

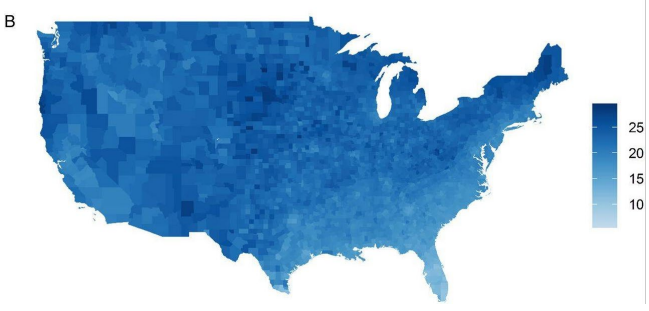
[1]zhe.zheng@yale.edu; zhe.zheng@aya.yale.edu; gigi.zhe.zheng@gmail.com

Background

Before the COVID-19 pandemic, the timing of RSV seasonal epidemics exhibited notable spatial patterns







Virginia E. Pitzer, 2015, PLoS Pathog
Daniel Weinberger, 2015, CID

Outline

- In section 1, we will first introduce how to find the peak timing of RSV epidemics using harmonic regression (given regular annual/biennial seasonality). We will then talk about identifying the onset of RSV epidemics using second derivative method, regardless of the seasonality of RSV.
- In section 2, we will learn how to use R to identify peak timing and onset of RSV epidemics.
- In section 3, we will incorporate the spatial component of RSV epidemics. We will first introduce the concept of spatial autocorrelation and then lean to use R to account for spatial autocorrelation.

References

Relevant readings:

- RSV onset timing at county level

- RSV peak timing on state level

- Comparing RSV onset timing before and during the COVID-19 pandemic

- RSV peak timing at ZIP code level and the drivers of RSV spread

- Assessment and optimization of respiratory syncytial virus prophylaxis in Connecticut, 1996–2013

- Disease outbreak outcome estimation using penalized splines

Harmonic regression to estimate the peak timing of RSV epidemics

Note: Most of the following materials came from Dan and Ginny's Lecture notes abd Harmonic Regression by NCSS[2].

Please check out:

- Dan's class: Public Health Surveillance
- Ginny's class: Quantitative Methods in Infectious Diseases

$$X_t = \mu + R\cos(2\pi ft - d) + e_t$$

Where
X_t is the time-series contains a periodic (cyclic) component.
μ is mean of the series.
R is the amplitude of seasonality.
$f = \frac{1}{period}$ is the frequency of the periodic.
d is the phase or horizontal offset.
e_t is the random error (noise) of the series.
t is the time step

[2] https://www.ncss.com/wp-content/themes/ncss/pdf/Procedures/NCSS/Harmonic_Regression.pdf

Pseudo-RSV data: Simulate time series with a 12 month period

Imagine this is RSV case data from 2 states, and we want to investigate the epidemic characteristics in these states and the lag between states.

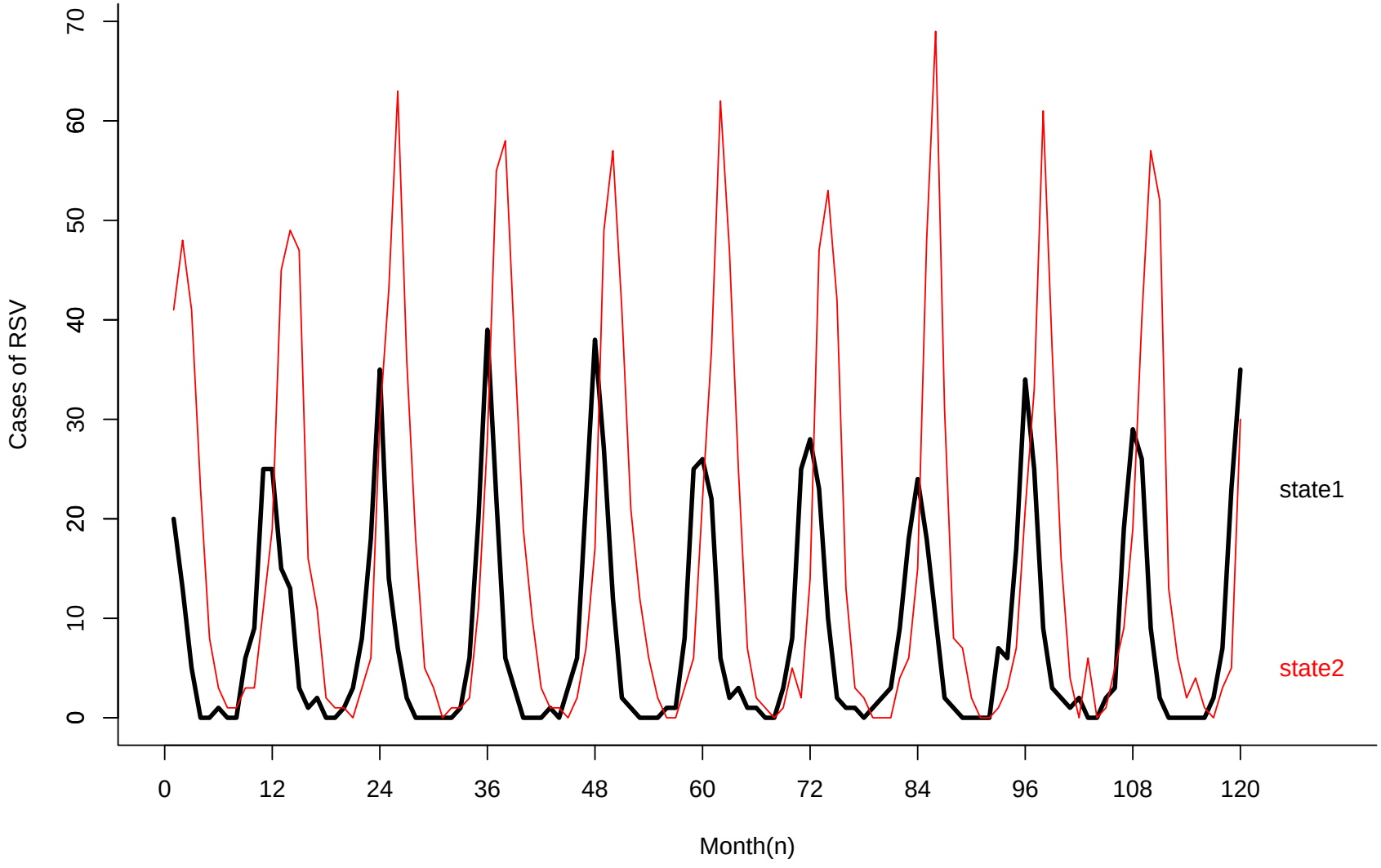
```
set.seed(123)
n=120 # 10 years
t <- seq(1,n)
amp1=2.5 # high amplitude
freq=1/12 # frequency = 1/period
amp2=2 # low amplitude

xt1a=amp1*cos(2*3.14159*t*freq)

#other series shifted by 2 months
xt2a=amp2*cos(2*3.14159*t*freq-1)

#Simulate some poisson count data
xt1=rpois(n,exp((1+xt1a)))
xt2=rpois(n,exp(2+xt2a))
```

The observed pseudo-RSV cases over time in two states



Investigate the epidemic characteristics of the pseudo-RSV time-serieses

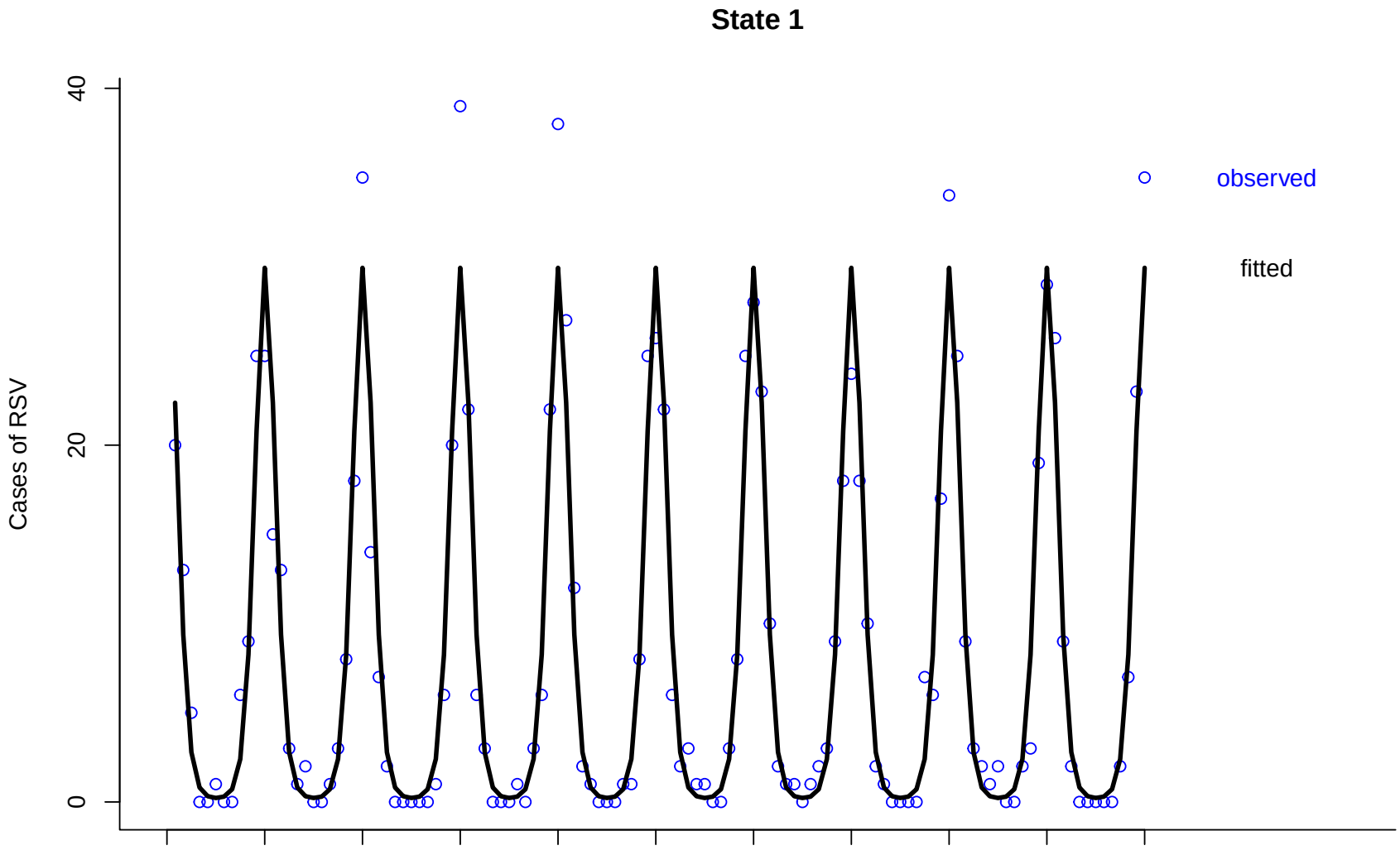
This is based on the prior knowledge that RSV has annual cycle in temperate regions and biannual cycle in high latitude regions. For other viruses or RSV circulation in other climate, you should consider using wavelet analysis to identify the periodicity first.

```
# Create the needed harmonic variables with
# 6 month, 12 month, and 24 month periodicities
t<-1:120
#Create harmonic variables
sin6=sin(2*pi*t/6)
cos6=cos(2*pi*t/6)
sin12=sin(2*pi*t/12)
cos12=cos(2*pi*t/12)
sin24=sin(2*pi*t/24)
cos24=cos(2*pi*t/24)
```

When you write the equations with sin and cos terms, you will need to include them both in an equation. For example, if the coefficient of cos12 is significant, you will need to include sin12 even though the coefficient is not significant in the summary.

Fit a simple poisson regression with 12 month period for state 1

```
fit1a <- glm(xt1~sin12+cos12, family='poisson')
pred1a<- fitted(fit1a)
#summary(fit1a)
```



Add in 24 month periodicity

```
fit2a <- glm(xt1~sin12+cos12+sin24+cos24,family='poisson')
pred2a<- fitted(fit2a)
summary(fit2a)
```

```
##
## Call:
## glm(formula = xt1 ~ sin12 + cos12 + sin24 + cos24, family = "poisson")
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.1406  -0.7825  -0.2894   0.3672   2.3956
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.94989    0.07873  12.065   <2e-16 ***
## sin12        0.07283    0.05737   1.269    0.204
## cos12        2.44820    0.09474  25.840   <2e-16 ***
## sin24        0.04088    0.09204   0.444    0.657
## cos24        0.04591    0.03407   1.348    0.178
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 1495.23  on 119  degrees of freedom
## Residual deviance:  101.72  on 115  degrees of freedom
## AIC: 436.78
##
## Number of Fisher Scoring iterations: 5
```

Add in 6 month periodicity

```
fit3a<-glm(xt1~sin12+cos12+sin24+cos24+sin6+cos6,family='poisson' )
pred3a<-fitted(fit3a)
summary(fit3a)
```

```
##
## Call:
## glm(formula = xt1 ~ sin12 + cos12 + sin24 + cos24 + sin6 + cos6,
##     family = "poisson")
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.0585  -0.8416  -0.2304   0.4924   2.5479
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  1.00529    0.09101  11.045   <2e-16 ***
## sin12        0.11920    0.09601   1.242    0.214
## cos12        2.32944    0.13971  16.673   <2e-16 ***
## sin24        0.04088    0.09204   0.444    0.657
## cos24        0.04591    0.03407   1.348    0.178
## sin6        -0.04481    0.07659  -0.585    0.558
## cos6         0.08943    0.07949   1.125    0.261
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 1495.23  on 119  degrees of freedom
## Residual deviance:  100.13  on 113  degrees of freedom
## AIC: 439.19
##
## Number of Fisher Scoring iterations: 5
```

Determine best model with AIC (smaller=better)

Winner is Model 1 (12 period). Also, when models have similar AIC score (within 2 points), we prefer a simpler model.

```
 AIC(fit1a)
```

```
## [1] 434.8372
```

```
 AIC(fit2a)
```

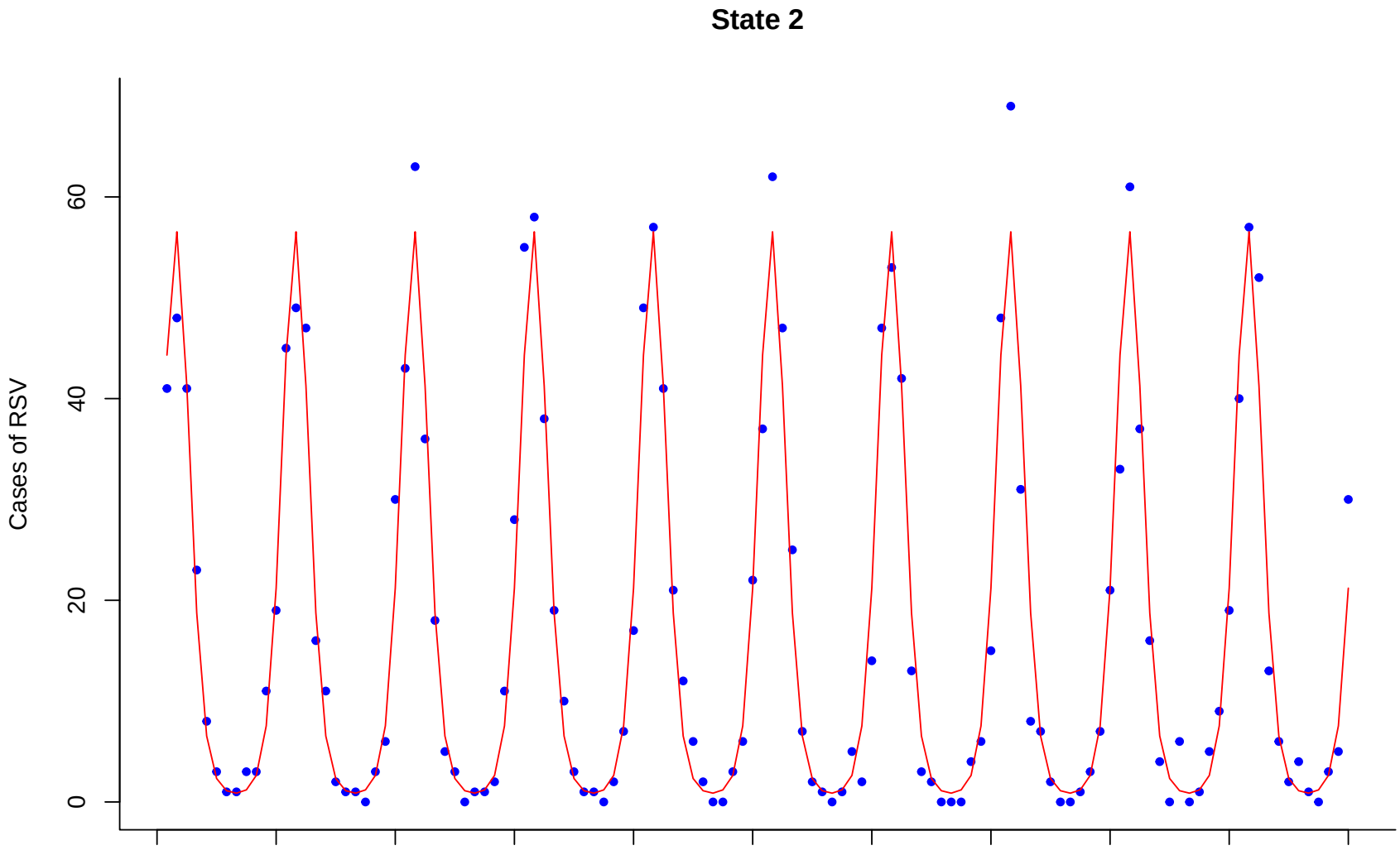
```
## [1] 436.7778
```

```
 AIC(fit3a)
```

```
## [1] 439.1851
```

Fit a simple poisson regression with 12 month period for state 2

```
fit1b <- glm(xt2~sin12+cos12, family='poisson')
pred1b<- fitted(fit1b)
#summary(fit1a)
```



Calculate amplitudes of the 12 periods in state1 and state2

```
beta_sin12_xt1<-coef(fit2a)['sin12']
beta_cos12_xt1<-coef(fit2a)['cos12']

amp12_xt1<-sqrt(beta_sin12_xt1^2+
                  beta_cos12_xt1^2)

amp12_xt1 #True value=2.5
```

```
##    sin12
## 2.449281
```

```
beta_sin12_xt2<-coef(fit1b)['sin12']
beta_cos12_xt2<-coef(fit1b)['cos12']

amp12_xt2<-sqrt(beta_sin12_xt2^2+
                  beta_cos12_xt2^2)

amp12_xt2 #True value=2
```

```
##    sin12
## 2.086493
```

Calculate phase of the 12 month period in state1 and state2

```
# Phase angle
phase12_xt1 <- -atan(beta_sin12_xt1/beta_cos12_xt1)
phase12_xt1
```

```
##       sin12
## -0.02973933
```

```
# True value = 0
phase12_xt2 <- -atan(beta_sin12_xt2/beta_cos12_xt2)
phase12_xt2
```

```
##     sin12
## -1.013021
```

```
# True value = 1
```

Calculate peak timing in month in state1 and state2

```
# Average peak timing
12*(1-phase12_xt1/(2*pi)) # True value = 0 month
```

```
##   sin12
## 12.0568
```

```
12*(1-phase12_xt2/(2*pi)) # True value = 2 month
```

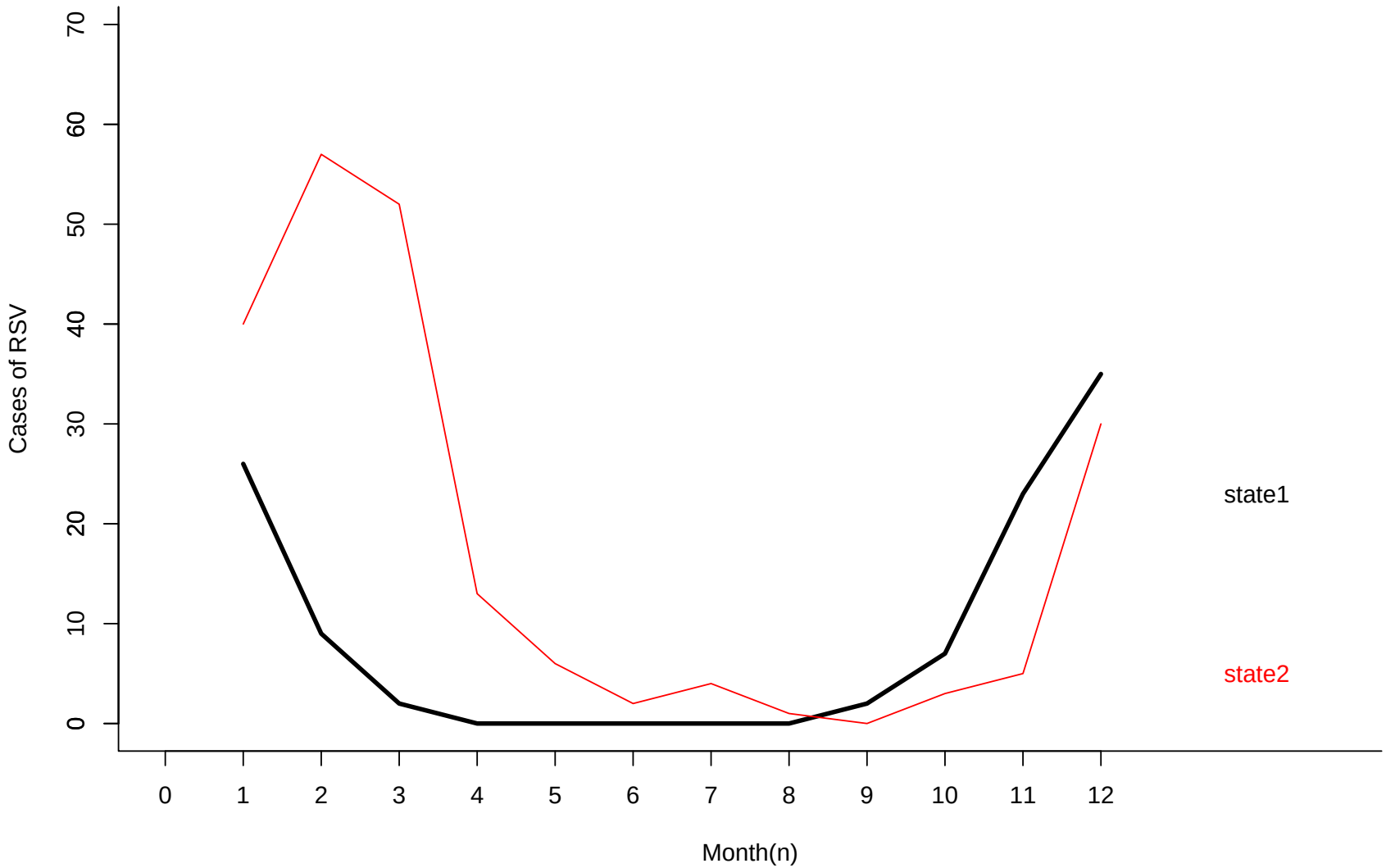
```
##    sin12
## 13.93473
```

Note: the period is 12 month - Therefore, peak timing of 12.06 month (year 1) equal to peak timing of 0.06 month (year 0)

- peak timing of 13.93 month (year 1) equals to 1.93 month (year 0)

Identify the onset of an epidemic using the second derivative method

When we look at the pseudo-RSV cases in the last 12 months, it is hard to determine when is the onset by eyes.



Fit a regression to the observed RSV cases

Because now we only observe part of the epidemic, we cannot use the harmonic regression to fit what we observe. Instead, we use p-spline regression, a generalized additive model. Please check out Smooth terms in GAM, P-splines in GAMs and Iris's work on p-spline inference for introduction.

```
require(mgcv)
```

```
## Loading required package: mgcv
```

```
## Loading required package: nlme
```

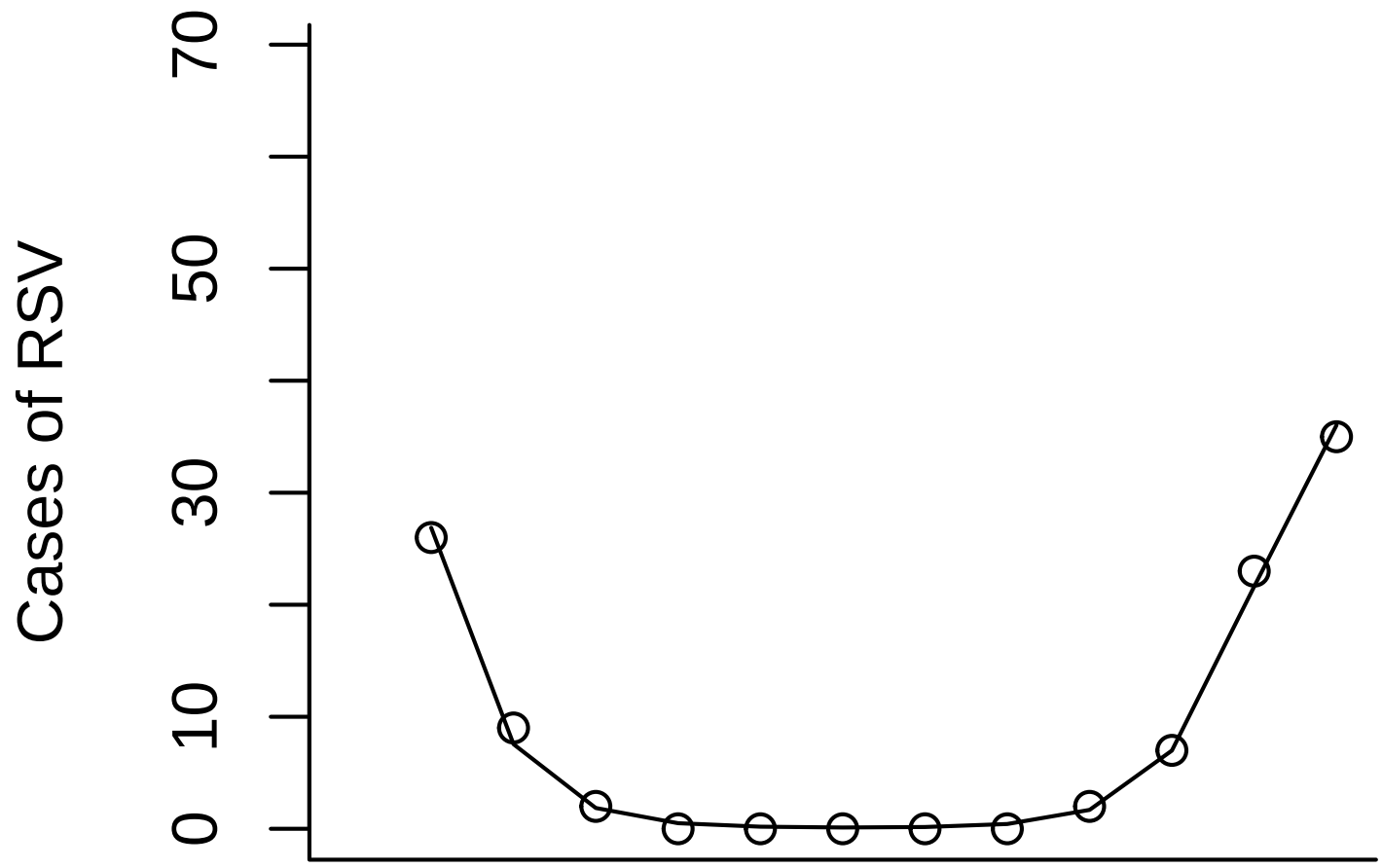
```
## This is mgcv 1.8-40. For overview type 'help("mgcv-package")'
```

```
irregular_model <- gam(cases ~ s(x=time, k=5, bs="ps"),
                       family=poisson, method="REML", data=data.
# k is knots
# The value for k set the upper limit on the
# wiggliness of the smooth function
# You can choose the value based on AIC scores as well
```

Comparing the model fit and what we observed

The dots are what we observed and the line is the model fit

```
plot(1:12, tail(xt1,12),type="p", xaxt="n",bty="l",
      ylab="Cases of RSV", xlab="Month(n)",
      ylim=range(c(xt1,xt2)),
     xlim=c(0,12))
lines(1:12,fitted(irregular_model))
```



Function to calculate the first and second derivatives

```
deriv <- function(x, y) diff(y) / diff(x) # function to calculat
middle_pts <- function(x) x[-1] - diff(x) / 2 # function to assi
```

```
#install.packages("pspline.inference")
library(pspline.inference)
t <- seq(0.5,12,0.01)
dtime <- seq(0.5,12,0.01)

# we use Iris's package to get the uncertainty interval
# of disease trajectory.
data=pspline.sample.timeseries(irregular_model, data.frame(time=

deriv.pred = data.frame(deriv=diff(data$cases)/diff(data$time),t
```