Data Mining de Series Temporales

Filtros FIR e IIR

Dr. Marcelo Risk

FCEN UBA

2020

Introducción

Filtros FIR e IIR

- Los filtros digitales se clasifican en dos tipos de acuerdo a la respuesta del filtro cuando a la entrada se presenta una señal impulso:
  - **FIR**: *Finite Impulse Response*, tienen respuesta lineal, son estables, no recursivos, lentos, requieren más coeficientes, son filtros que tienen solamente implementación digital; los filtros MA son FIR.
  - **IIR**: *Infinite Impulse Response*, respuesta no lineal, pueden ser inestables, recursivos, rápidos, requieren menos coeficientes, son filtros que modelizan filtros analógicos.

Filtros FIR

Filtros FIR

- Los filtros digitales **FIR** tienen la siguiente ecuación de recurrencia, en su formsa general:

$$y(n) = \sum_{k=0}^{N-1} h(k)x(n-k)$$

- Ejemplo: un filtro MA3 tiene un N=3:

$$MA3(n) = \sum_{k=0}^{N-1} h(k)x(n-k) = \frac{1}{3}x(n) + \frac{1}{3}x(n-1) + \frac{1}{3}x(n-2)$$

Filtros FIR

- Los coeficientes de los **FIR** deben cumplir con la suguiente condición, de forma de no amplificar ni atenuar la salida del mismo:

$$\sum_{k=0}^{N-1} h(k) = 1$$

- Ejemplo: un filtro MA3:

$$\sum_{k=0}^{N-1} h_{MA3}(k) = \frac{1}{3} + \frac{1}{3} + \frac{1}{3} = 1$$

Filtros FIR

- Los **FIR** pueden ser centrados:

$$y(n) = \sum_{k=M_1}^{M_2} h(k)x(n-k)$$

- Ejemplo: un filtro MA3 centrado:

$$MA3(n) = \sum_{k=-1}^{1} h(k)x(n-k) = \frac{1}{3}x(n+1) + \frac{1}{3}x(n) + \frac{1}{3}x(n-1)$$

Filtros FIR con distintos pesos

- Un filtro de tres coeficientes *FIR3*, denominado filtro de *Hanning*:

$$FIR3(n) = \sum_{k=0}^{2} h(k)x(n-k) = \frac{1}{4}x(n) + \frac{1}{2}x(n-1) + \frac{1}{4}x(n-2)$$

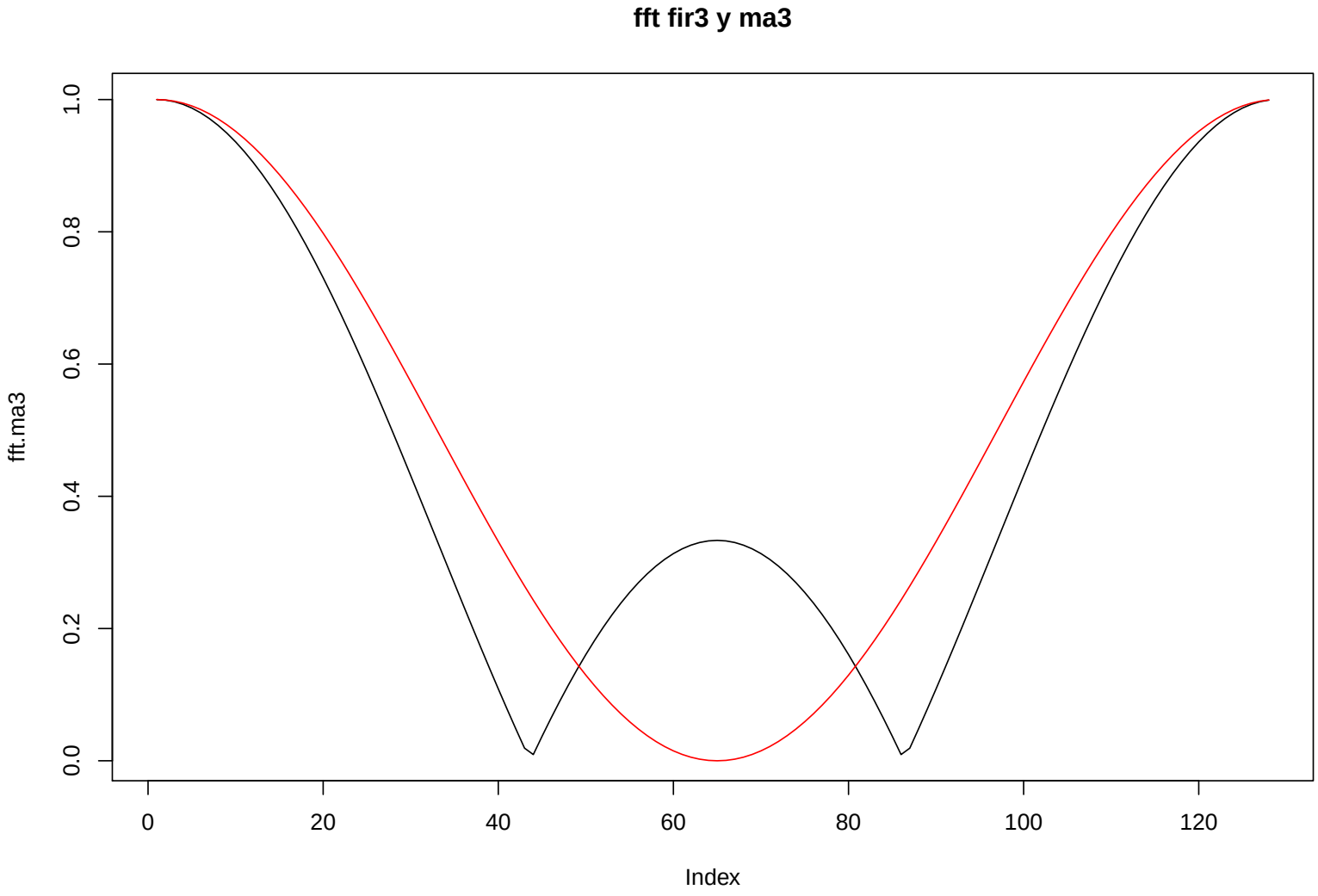
Ejemplo FIR3 de Hanning

```
N = 128
fir3 = rep(0,N)
fir3[1] = 1/4
fir3[2] = 1/2
fir3[3] = 1/4

ma3 = rep(0,N)
ma3[1:3] = 1/3

fft.fir3 = Mod(fft(fir3))
fft.ma3 = Mod(fft(ma3))
plot(fft.ma3,type='l',main='fft fir3 y ma3')
lines(fft.fir3,col='red')
```

Ejemplo FIR3 de Hanning



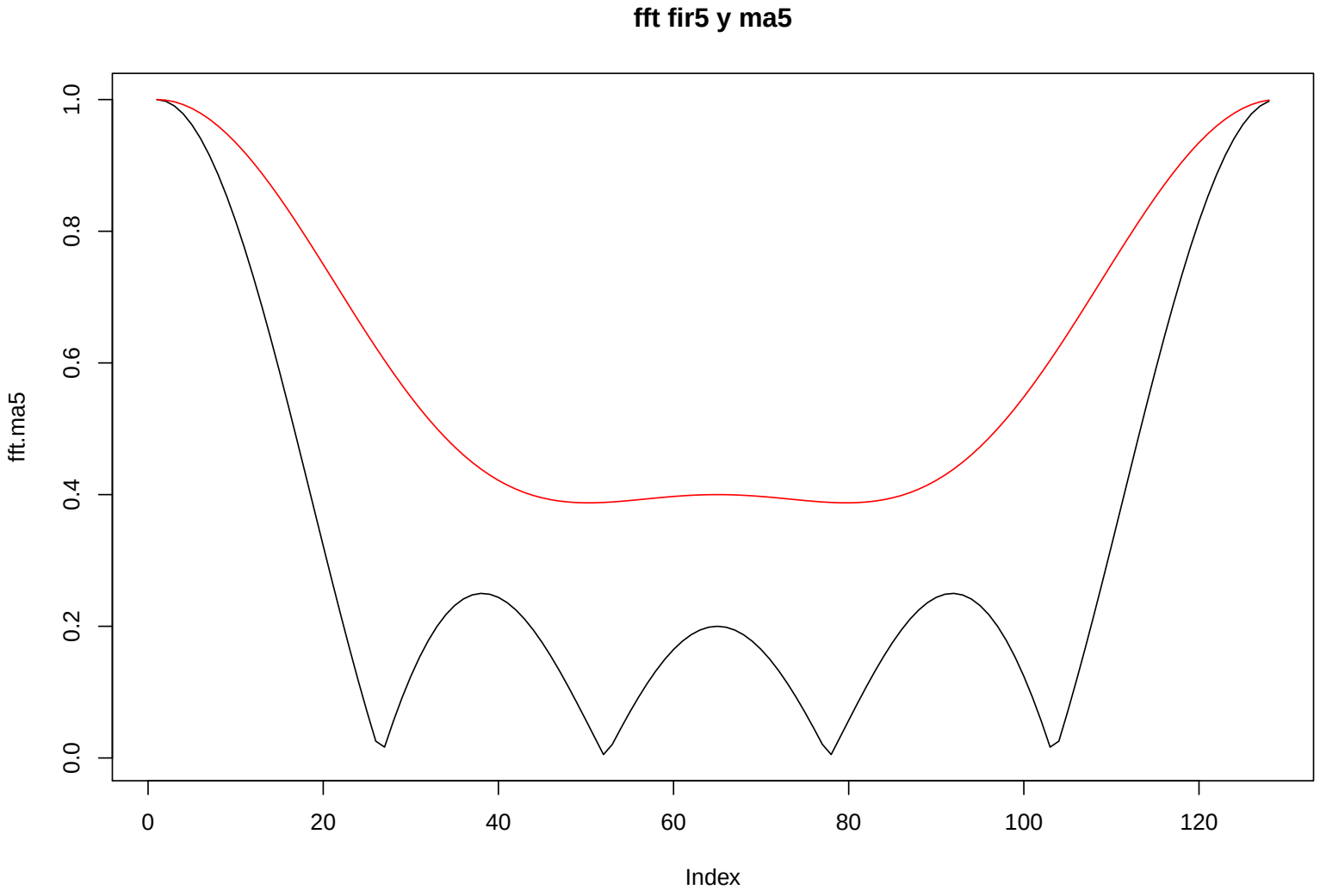
Ejemplo FIR5

```
N = 128
fir5 = rep(0,N)
fir5[1] = 0.05
fir5[2] = 0.15
fir5[3] = 0.6
fir5[4] = 0.15
fir5[5] = 0.05

ma5 = rep(0,N)
ma5[1:5] = 1/5

fft.fir5 = Mod(fft(fir5))
fft.ma5 = Mod(fft(ma5))
plot(fft.ma5,type='l',main='fft fir5 y ma5')
lines(fft.fir5,col='red')
```

Ejemplo FIR5



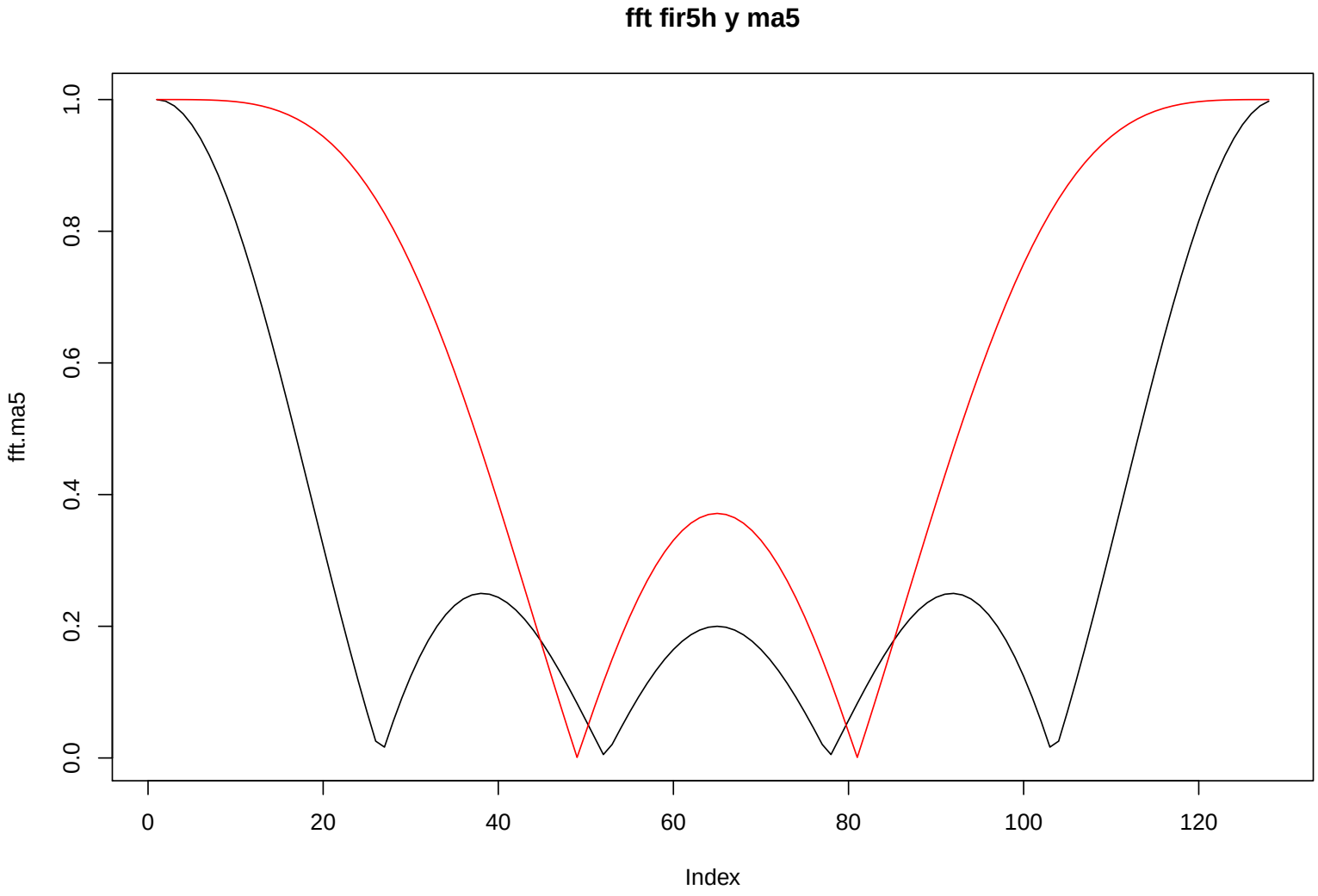
Ejemplo FIR5 de Hammimg

```
N = 128
fir5h = rep(0,N)
fir5h[1] = -3/35
fir5h[2] = 12/35
fir5h[3] = 17/35
fir5h[4] = 12/35
fir5h[5] = -3/35

ma5 = rep(0,N)
ma5[1:5] = 1/5

fft.fir5h = Mod(fft(fir5h))
fft.ma5 = Mod(fft(ma5))
plot(fft.ma5,type='l',main='fft fir5h y ma5')
lines(fft.fir5h,col='red')
```

Ejemplo FIR5 de Hammimg



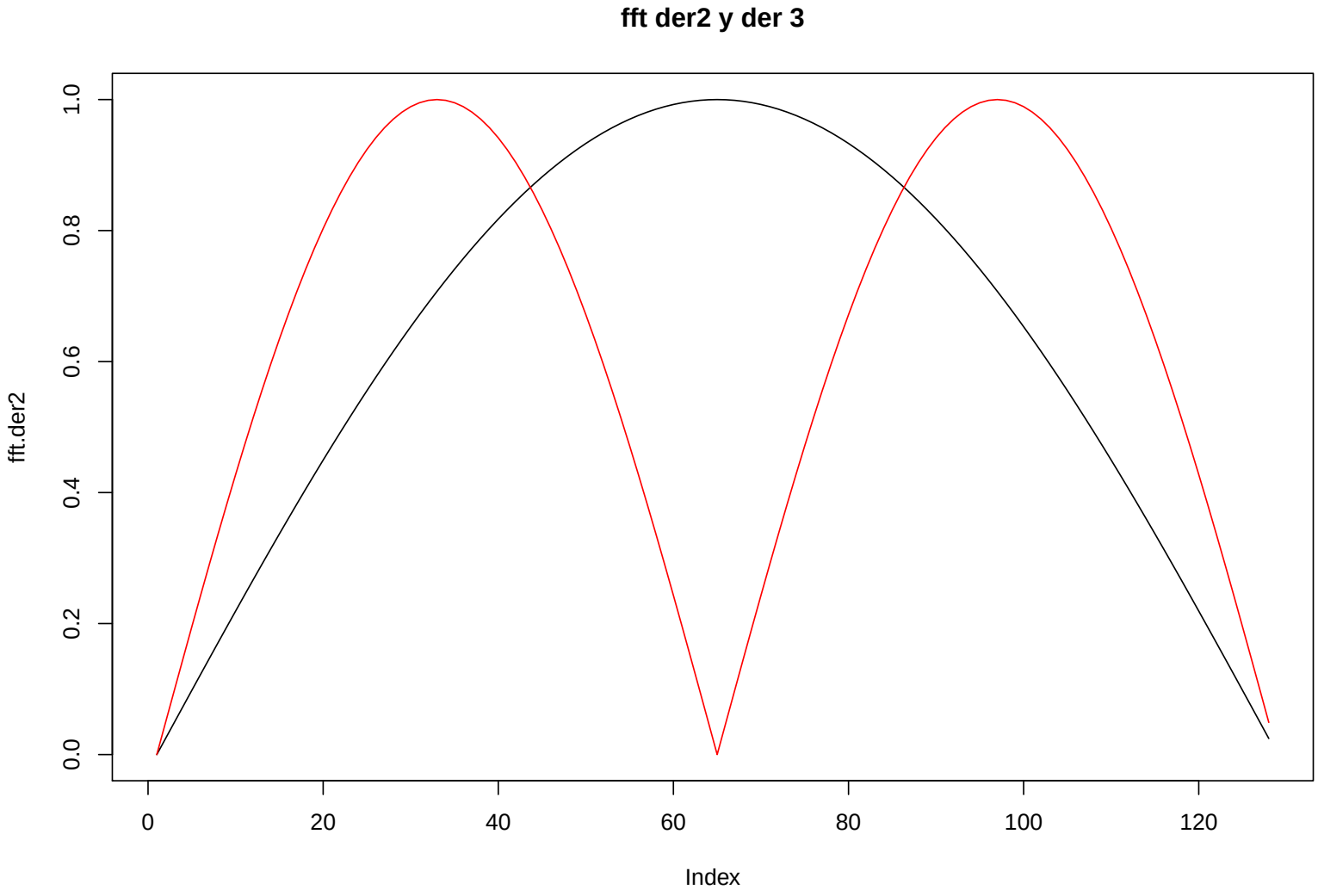
Ejemplo derivadas de 2 y 3 muestras

```
N = 128
der2 = rep(0,N)
der2[1] = 0.5
der2[2] = -0.5

der3 = rep(0,N)
der3[1] = 0.5
der3[3] = -0.5

fft.der2 = Mod(fft(der2))
fft.der3 = Mod(fft(der3))
plot(fft.der2,type='l',main='fft der2 y der3')
lines(fft.der3,col='red')
```

Ejemplo derivadas de 2 y 3 muestras

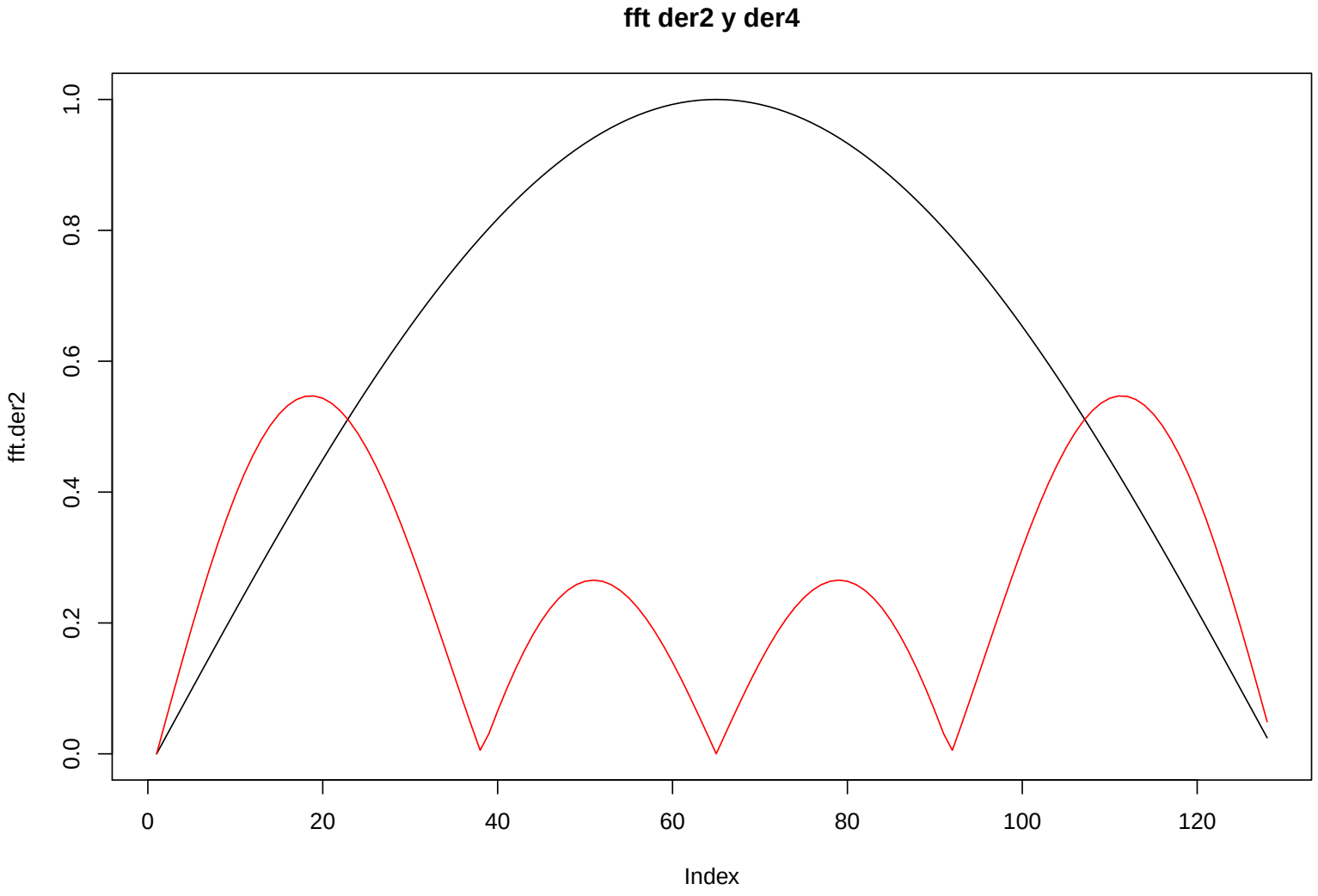


Ejemplo derivadas de 4 muestras

```
N = 128
der4 = rep(0,N)
der4[1] = 2/10
der4[2] = 1/10
der4[0] = 0
der4[4] = -1/10
der4[5] = -2/10

fft.der4 = Mod(fft(der4))
plot(fft.der2,type='l',main='fft der2 y der4')
lines(fft.der4,col='red')
```

Ejemplo derivadas de 4 muestras



Filtros IIR

Filtros IIR

- Los filtros digitales **IIR** tienen la siguiente ecuación de recurrencia, en su formsa general:

$$y(n) = \sum_{k=0}^{N} a_k x(n-k) - \sum_{k=1}^{M} b_k y(n-k)$$

- Ejemplo: un filtro IIR2:

$$IIR2(n) = 0.5x(n) - 0.5IIR2(n-1)$$

Ejemplo IIR2

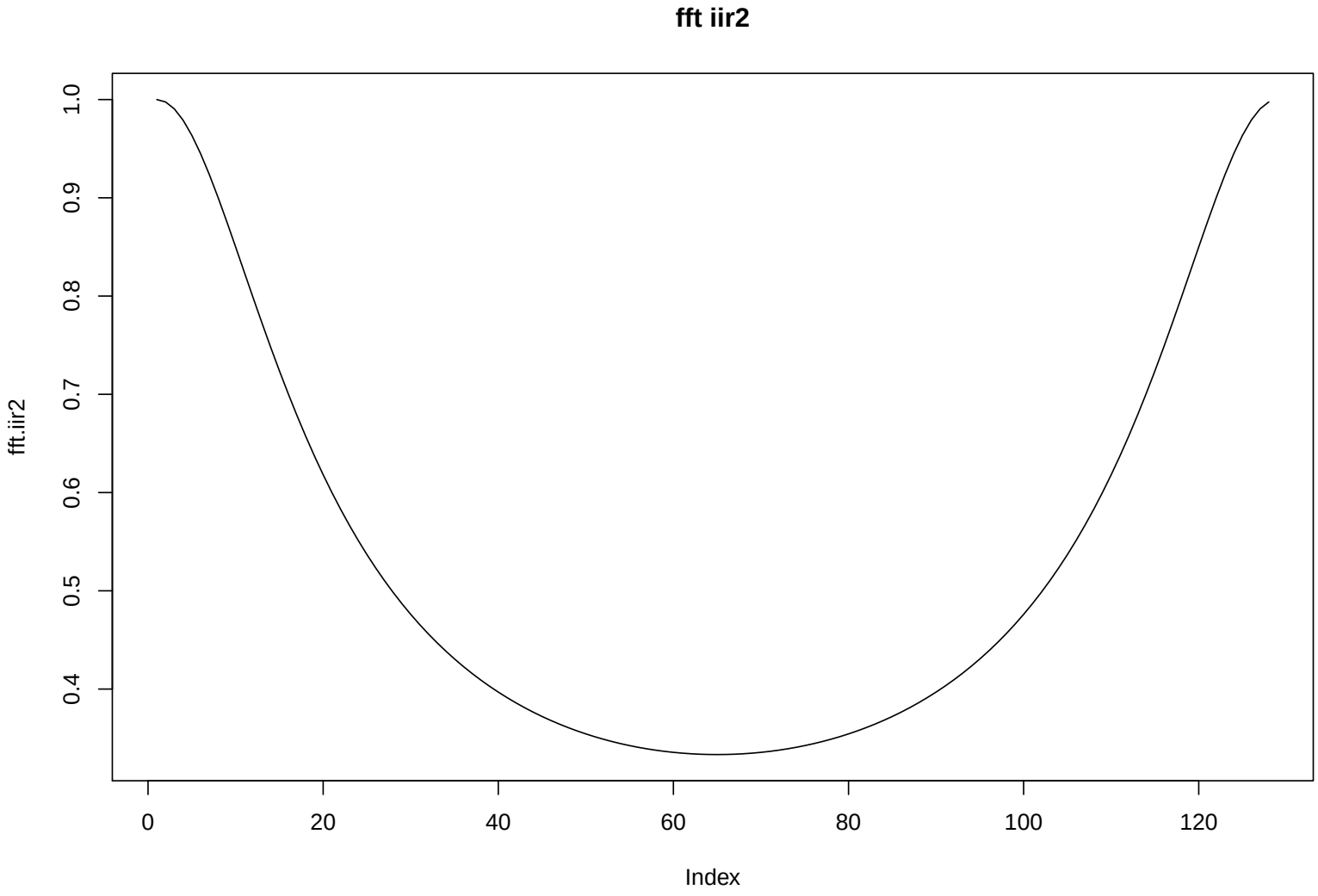
```
N = 128
x = rep(0,N)
x[2] = 1

iir2 = rep(0,N)
a = 0.5
b = -0.5

for(n in 2:N)
{
  iir2[n] = a*x[n] - b*iir2[n-1]
}

fft.iir2 = Mod(fft(iir2))
plot(fft.iir2,type='l',main='fft iir2')
```

Ejemplo IIR2



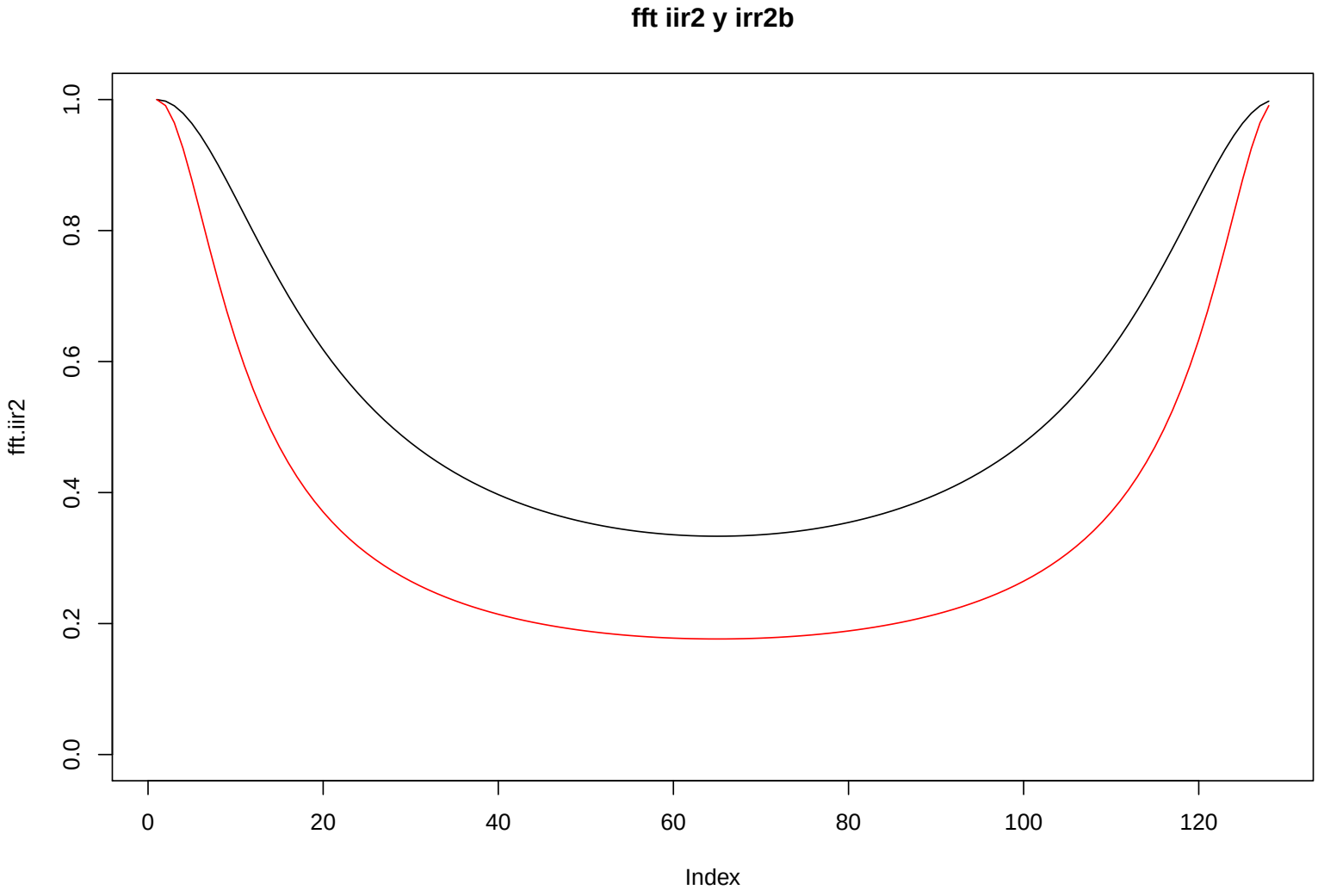
Ejemplo IIR2b

```
iir2b = rep(0,N)
a = 0.3
b = -0.7

for(n in 2:N)
{
  iir2b[n] = a*x[n] - b*iir2b[n-1]
}

fft.iir2b = Mod(fft(iir2b))
plot(fft.iir2,type='l',main='fft iir2 y irr2b',ylim = c(0,1))
lines(fft.iir2b,col='red')
```

Ejemplo IIR2b



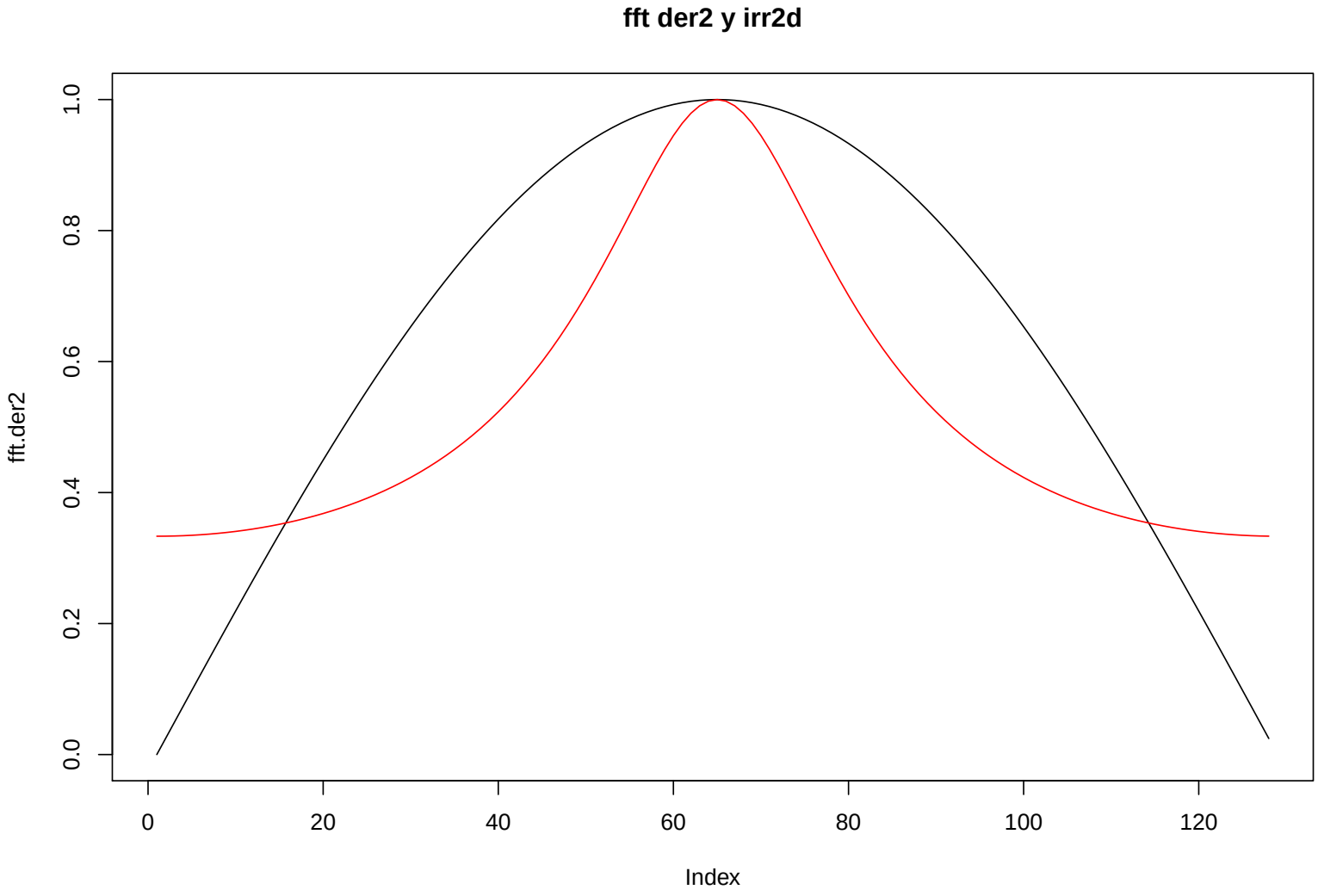
Ejemplo derivada con IIR

```
iir2d = rep(0,N)
a = 0.5
b = 0.5

for(n in 2:N)
{
  iir2d[n] = a*x[n] - b*iir2d[n-1]
}

fft.iir2d = Mod(fft(iir2d))
plot(fft.der2,type='l',main='fft der2 y irr2d',ylim = c(0,1))
lines(fft.iir2d,col='red')
```

Ejemplo derivada con IIR



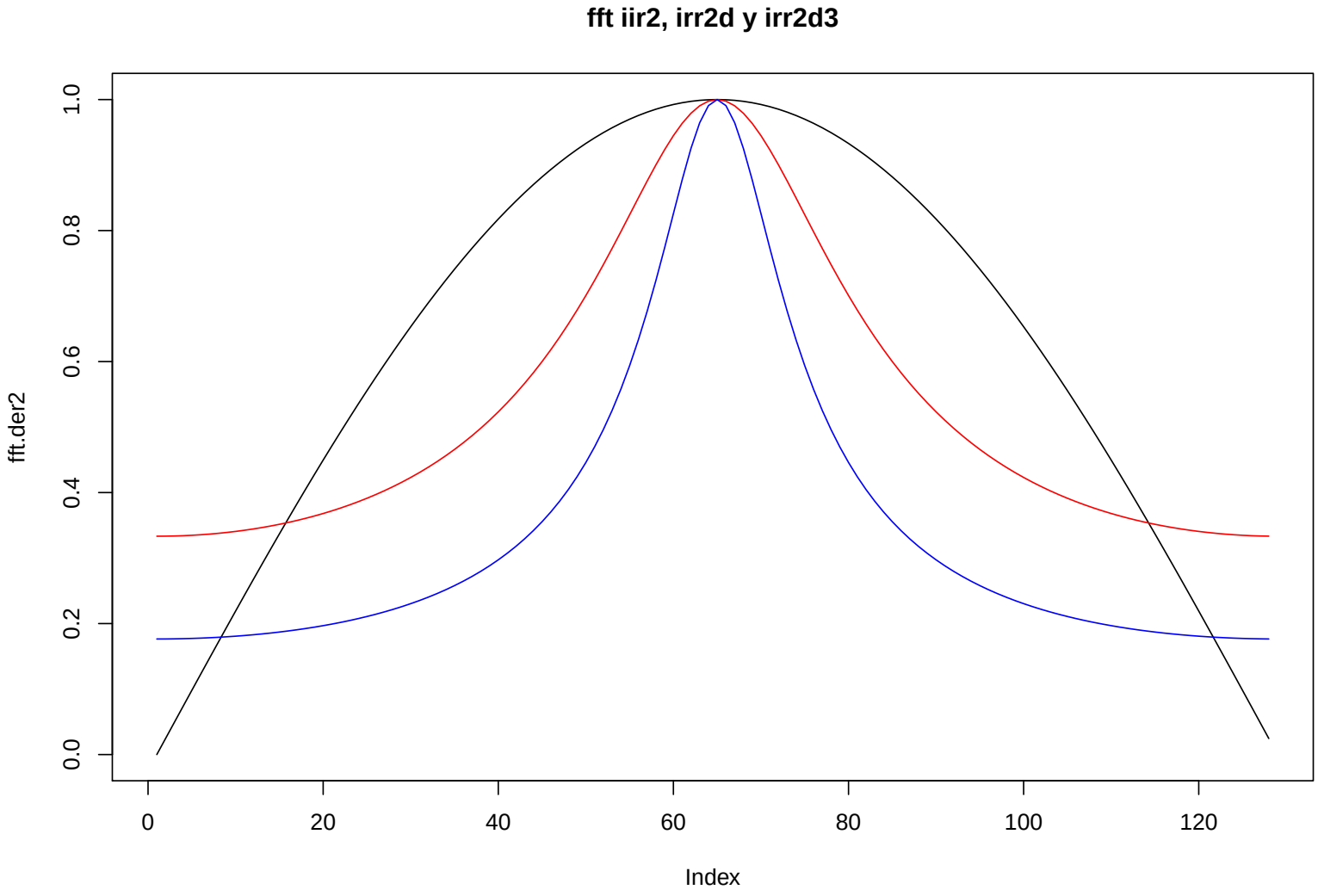
Ejemplo derivada con IIR

```
iir2d3 = rep(0,N)
a = 0.3
b = 0.7

for(n in 2:N)
{
  iir2d3[n] = a*x[n] - b*iir2d3[n-1]
}

fft.iir2d3 = Mod(fft(iir2d3))
plot(fft.der2,type='l',main='fft iir2, irr2d y irr2d3',ylim = c(
lines(fft.iir2d,col='red')
lines(fft.iir2d3,col='blue')
```

Ejemplo derivada con IIR



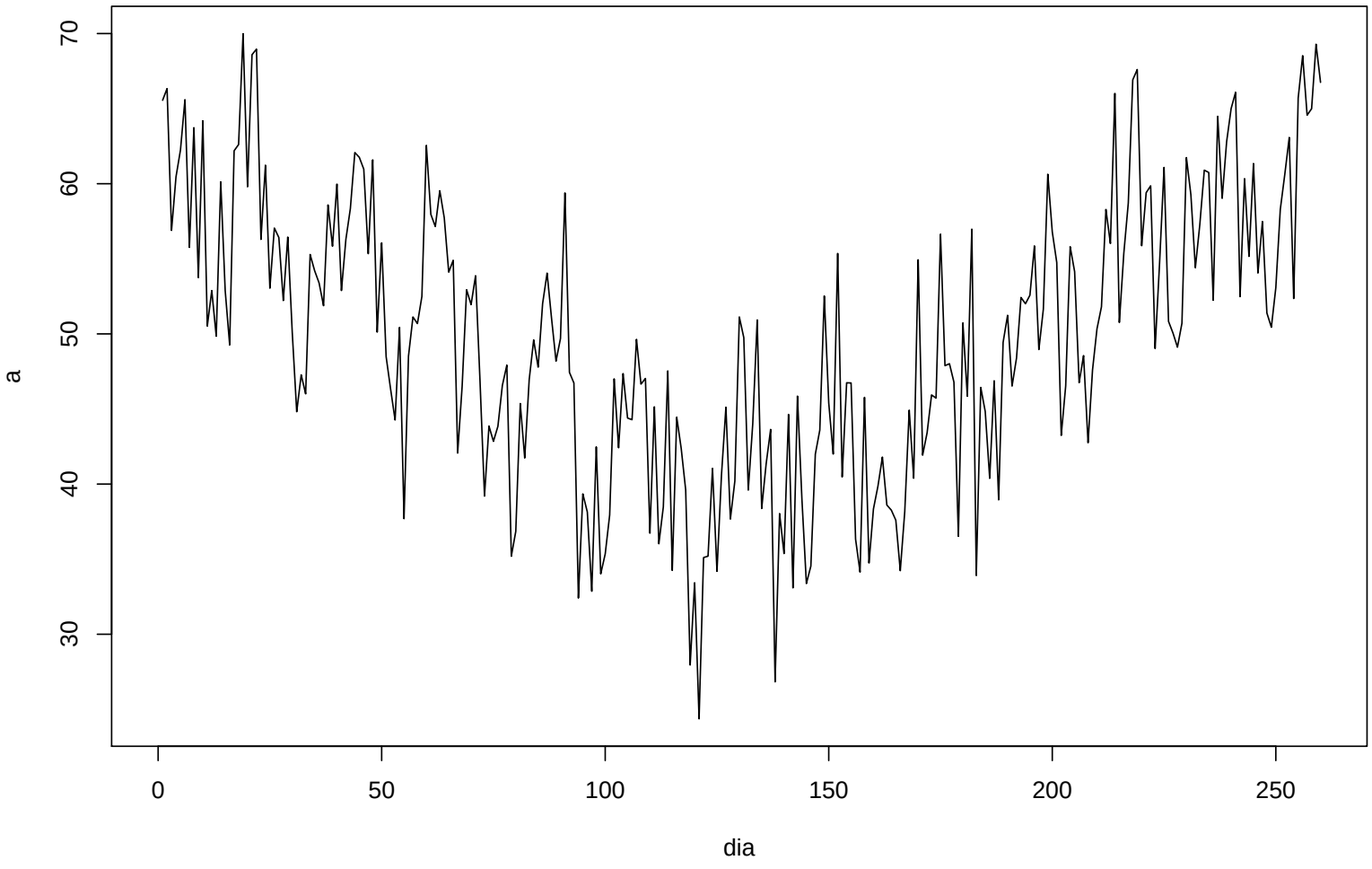
Aplicación de FIR e IIR

Aplicación de FIR e IIR

```
#dir = '/home/marcelo/Dropbox/PS_2020/'
dir = '/Users/marcelorisk/Dropbox/PS_2020/'
setwd(dir)
dato = read.csv('TS_1.csv')

op = par(mfrow = c(1,1))
plot(dato$t,dato$a,type='l',xlim=c(0,260),xlab='dia',ylab='a')
```

Aplicación de FIR e IIR

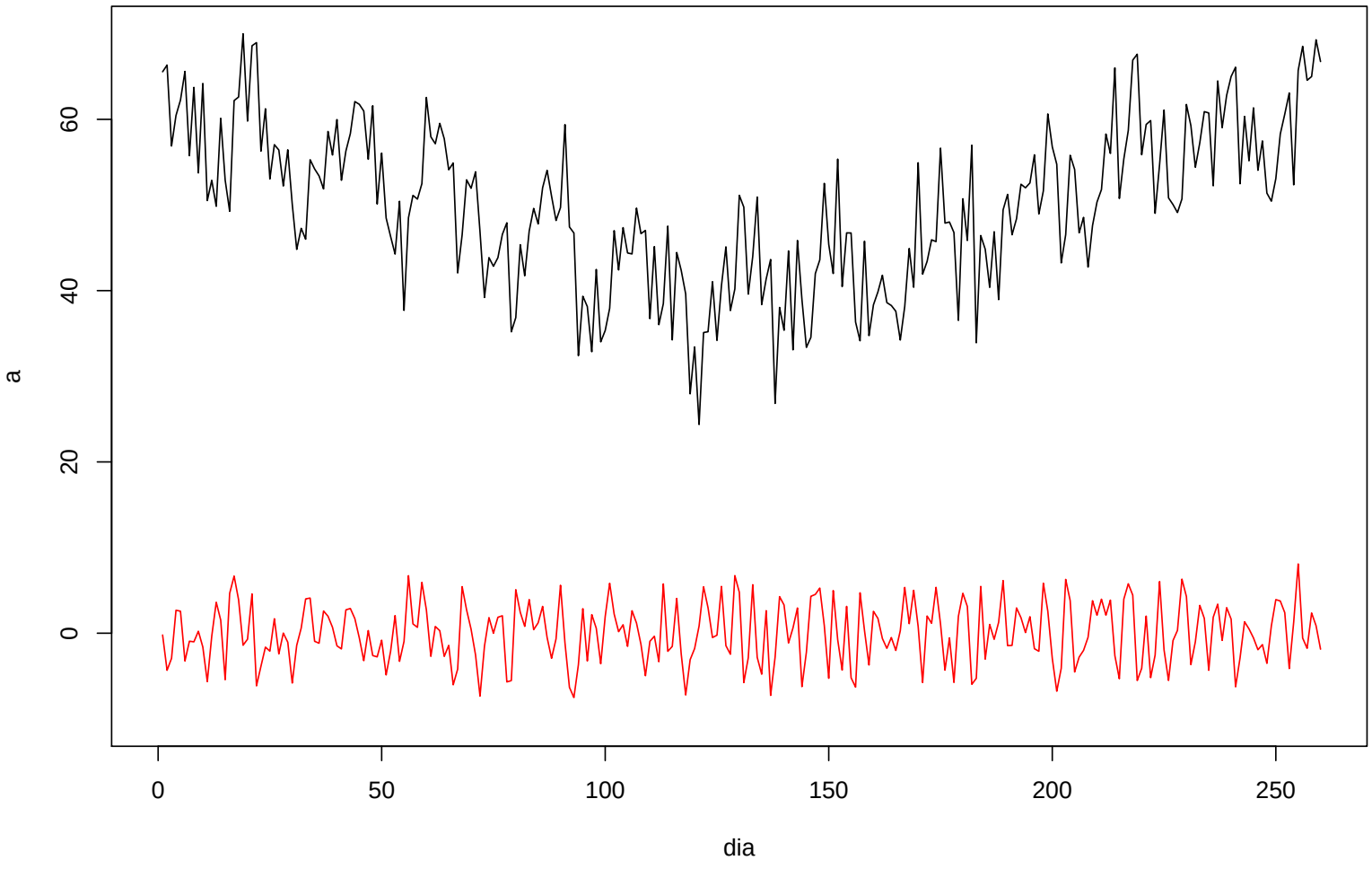


Aplicación derivada FIR de 3 muestras

```
der.a = filter(dato$a,c(0.5,0,-0.5),circular=TRUE)

op = par(mfrow = c(1,1))
plot(dato$t,dato$a,type='l',xlim=c(0,260),ylim=c(-10,70),xlab='d
lines(der.a,col='red')
```

Aplicación derivada FIR de 3 muestras

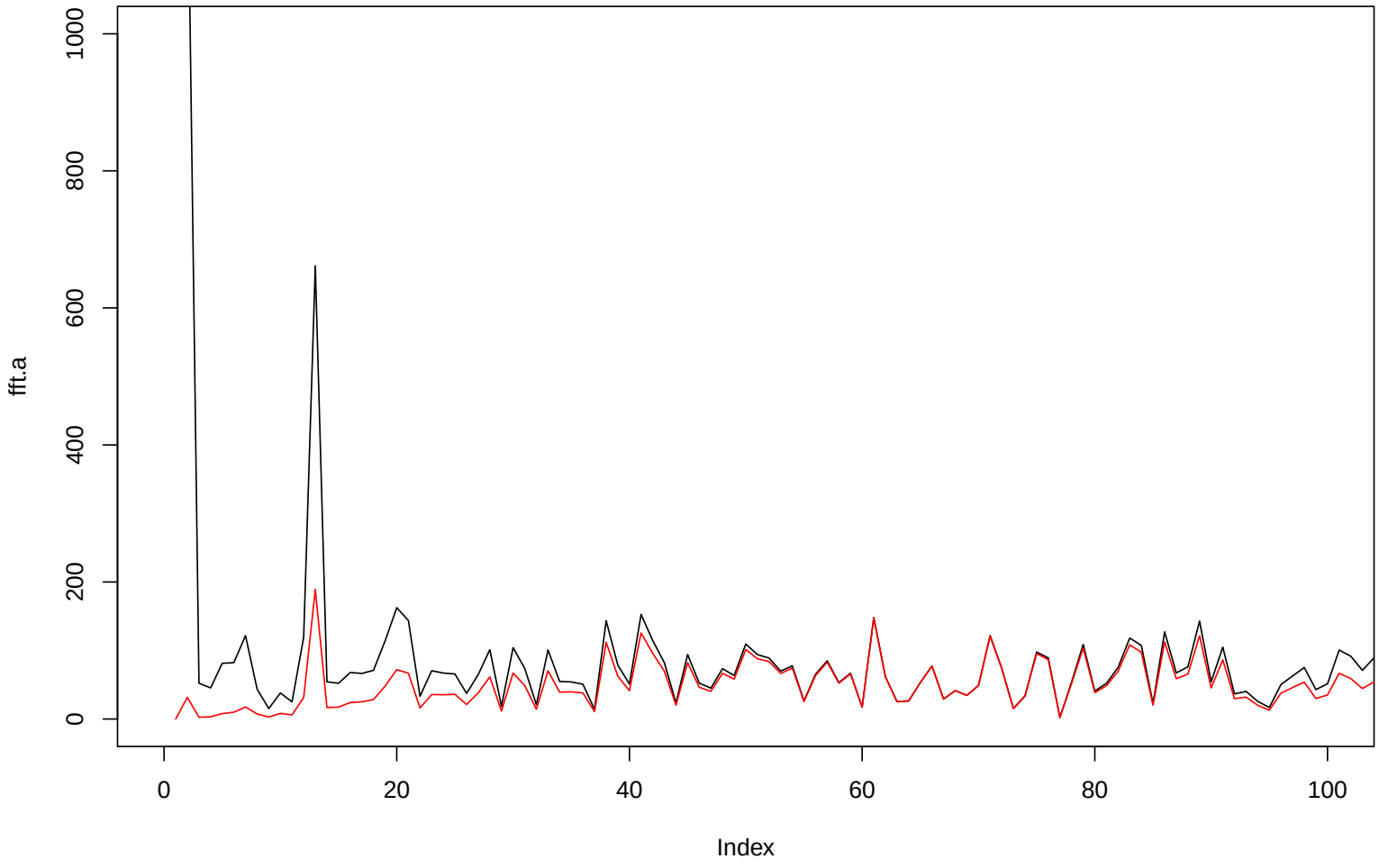


Aplicación derivada FIR de 3 muestras

```
fft.der.a = Mod(fft(der.a))
fft.a = Mod(fft(dato$a))

plot(fft.a,type='l',xlim = c(0,100),ylim = c(0,1000))
lines(fft.der.a,col='red')
```

Aplicación derivada FIR de 3 muestras

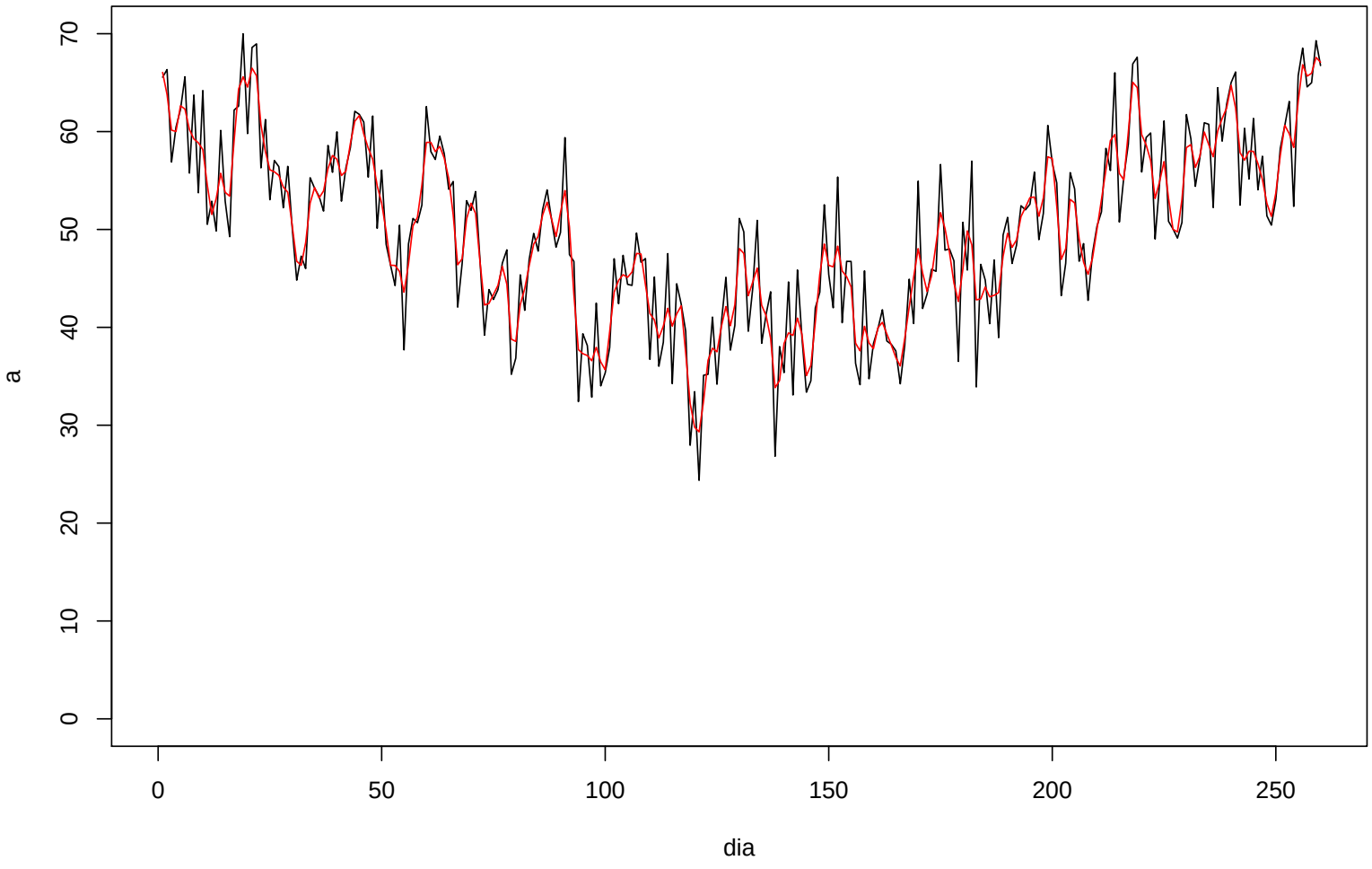


Aplicación FIR de Hanning

```
h3.a = filter(dato$a,c(0.25,0.5,0.25),circular=TRUE)

op = par(mfrow = c(1,1))
plot(dato$t,dato$a,type='l',xlim=c(0,260),ylim=c(0,70),xlab='dia
lines(h3.a,col='red')
```

Aplicación FIR de Hanning



Aplicación FIR de Hanning

```
fft.h3.a = Mod(fft(h3.a))
fft.a = Mod(fft(dato$a))

plot(fft.a,type='l',xlim = c(0,100),ylim = c(0,1000))
lines(fft.h3.a,col='red')
```

Aplicación FIR de Hanning