

Trabalho 4

Daniel Krügel

2023-11-25

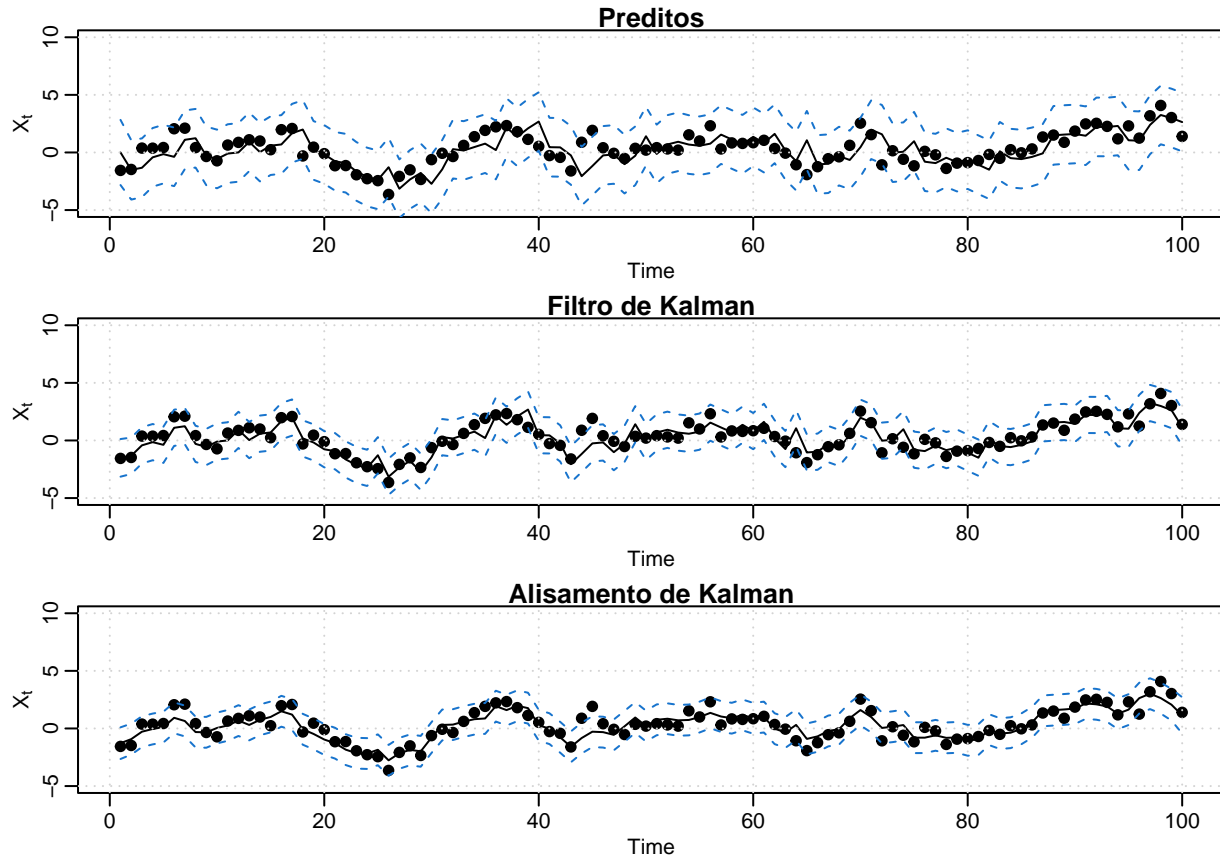
Questão 3

Gerando série de dados:

```
set.seed(123);  
x <- numeric(100)  
x[1] <- rnorm(1,0,2.78)  
  
for(i in 2:100){  
  x[i] <- x[i-1]*0.8 + rnorm(1)  
}  
  
v <- rnorm(100,0,1)  
y <- x + v  
  
num = 100
```

Filtro e alisamento:

```
ks = Ksmooth(y, A=1, mu0=0, Sigma0=1, Phi=1, sQ=1, sR=1)
```



Valores iniciais:

x_1	X_{0n}	P_{0n}
-1.558122	-0.6389023	0.618034

Tabela com as 10 primeiras observações:

Y	X_p	X_f	X_s	P_p	P_f	P_s
-2.2685289	0.0000000	-1.5123526	-1.2778046	2.000000	0.6666667	0.4721360
-1.2197916	-1.5123526	-1.3295020	-0.9259827	1.666667	0.6250000	0.4508497
0.1306762	-1.3295020	-0.4255822	-0.2803520	1.625000	0.6190476	0.4477441
0.0248602	-0.4255822	-0.1471269	-0.0457493	1.619048	0.6181818	0.4472910
-0.5244086	-0.1471269	-0.3803079	0.1182437	1.618182	0.6180556	0.4472249
2.0118053	-0.3803079	1.0981069	0.9248891	1.618056	0.6180371	0.4472152
1.3214781	1.0981069	1.2361580	0.6446182	1.618037	0.6180344	0.4472138
-1.2478971	1.2361580	-0.2990726	-0.3125125	1.618034	0.6180341	0.4472136
-0.7310435	-0.2990726	-0.5660453	-0.3342586	1.618034	0.6180340	0.4472136
0.1926811	-0.5660453	-0.0971266	0.0407803	1.618034	0.6180340	0.4472136

Questão 14

```
y = aggregate(unemp, nfrequency = 4, FUN = mean) # Fornecido pelo enunciado

num = length(y)
A = cbind(1,1,0,0)
# Função para calcular a verossimilhança
Linn =function(para){
  Phi = diag(0,4); Phi[1,1] = para[1]
  Phi[2,]=c(0,-1,-1,-1); Phi[3,]=c(0,1,0,0); Phi[4,]=c(0,0,1,0)
  cQ1 = para[2]; cQ2 = para[3] # raiz quadrada de q11 e q22
  cQ = diag(0,4); cQ[1,1]=cQ1; cQ[2,2]=cQ2
  cR = para[4] # raiz quadrada de r11
  kf = Kfilter(jj, A, mu0, Sigma0, Phi, cQ, cR)
  return(kf$like) }

# Parâmetros iniciais
mu0 = c(.7,0,0,0); Sigma0 = diag(.04,4)
init.par = c(1.03,.1,.1,.5) # Phi[1,1], 2 cQs e cR
# Estimação e resultados
est = optim(init.par, Linn,NULL, method='BFGS', hessian=TRUE, control=list(trace=1,REPORT=1))
```

```
## initial value 2.693644
## iter 2 value -0.853526
## iter 3 value -9.416505
## iter 4 value -10.241752
## iter 5 value -19.419809
## iter 6 value -30.441188
## iter 7 value -31.825543
## iter 8 value -32.248413
## iter 9 value -32.839918
## iter 10 value -33.019870
## iter 11 value -33.041749
## iter 12 value -33.050583
## iter 13 value -33.055492
## iter 14 value -33.078152
## iter 15 value -33.096870
## iter 16 value -33.098405
## iter 17 value -33.099018
## iter 18 value -33.099385
## iter 19 value -33.099498
## iter 19 value -33.099498
## final value -33.099498
## converged
```

```
SE = sqrt(diag(solve(est$hessian)))
u = cbind(estimate=est$par, SE)
rownames(u)=c('Phi11','sigw1','sigw2','sigv'); u
```

```
## estimate SE
## Phi11 1.0350847657 0.00253645
```

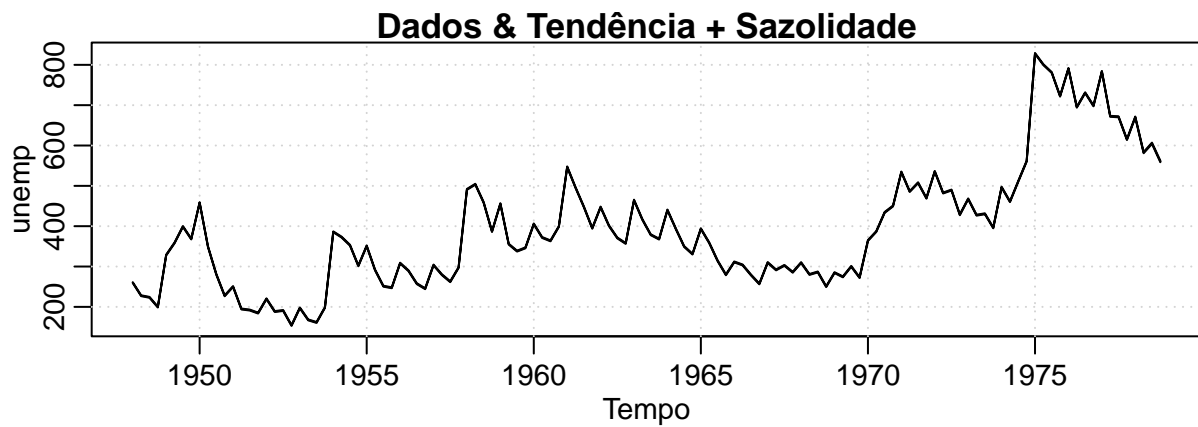
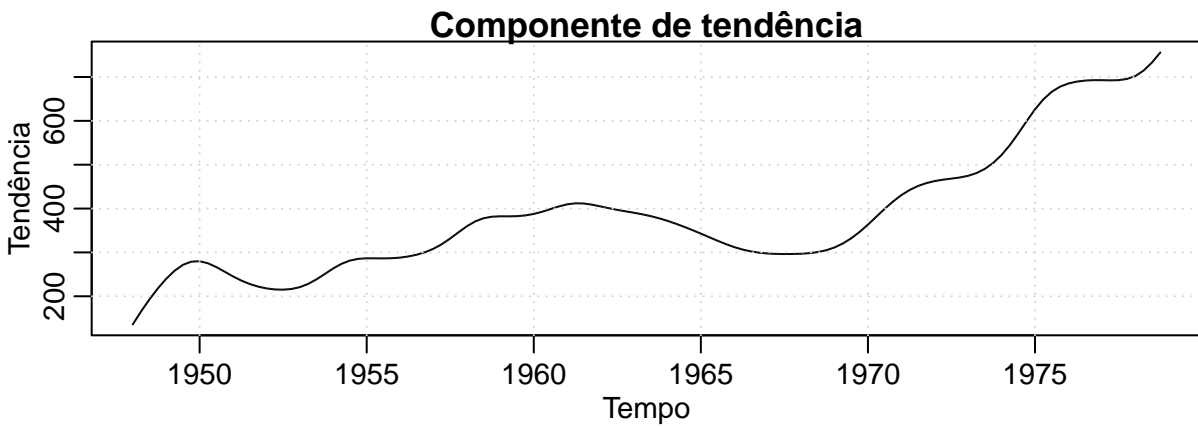
```
## sigw1 0.1397255477 0.02155155
## sigw2 0.2208782663 0.02376430
## sigv 0.0004655672 0.24174702
```

```
# Alisamento
Phi = diag(0,4); Phi[1,1] = est$par[1]
Phi[2,]=c(0,-1,-1,-1); Phi[3,]=c(0,1,0,0); Phi[4,]=c(0,0,1,0)
cQ1 = est$par[2]; cQ2 = est$par[3]
cQ = diag(1,4); cQ[1,1]=cQ1; cQ[2,2]=cQ2
cR = est$par[4]
ks = Ksmooth(y,A,mu0,Sigma0,Phi,cQ,cR)

# Gráficos
Tsm = ts(ks$Xs[1,,], start=1948, freq=4)
Ssm = ts(ks$Xs[2,,], start=1948, freq=4)
p1 = 3*sqrt(ks$Ps[1,1,]); p2 = 3*sqrt(ks$Ps[2,2,])
par(mfrow = c(2,1), mar=c(3,3,1,1), mgp=c(1.6,.6,0), cex=0.9, pch=19)
plot(Tsm, main='Componente de tendência', xlab="Tempo", ylab='Tendência')
xx = c(time(y), rev(time(y)))
yy = c(Tsm-p1, rev(Tsm+p1))
polygon(xx, yy, border=NA, col=gray(.5, alpha = .3))

grid()
plot(y, main='Dados & Tendência + Sazonalidade', xlab="Tempo",ylab='unemp')
xx = c(time(y), rev(time(y)) )
yy = c((Tsm+Ssm)-(p1+p2), rev((Tsm+Ssm)+(p1+p2)) )
polygon(xx, yy, border=NA, col=gray(.5, alpha = .3))
grid()

polygon(xx,yy, col=gray(.5, alpha = 0.4))
```



A linha de tendência apresenta uma estabilidade entre os anos de 55 e 70, porém apresentou um crescimento a partir da década de 70, olhando para o gráfico a baixo com dados de tendencia acrescentados ao padrão de sazonalidade conseguimos ver que o padrão o ciclo é de mais ou menos 1 ano, ou seja, temos um pico a cada ano.

Questão 23

```
set.seed(1) # Mudando a seed pois a seed 123 não achou convergencia no modelo

data <- polio

model <- depmix(data ~ 1, nstates = max(data), data= data.frame(polio), family=poisson())

fit01 <- fit(model)
```

```
## converged at iteration 340 with logLik: -224.6812
```

Como a função não salvou a matriz de transição nem os parâmetros, irei soltar a summary() inteira no próximo chunk

```
summary(fit01)
```

```
## Initial state probabilities model
## pr1 pr2 pr3 pr4 pr5 pr6 pr7 pr8 pr9 pr10 pr11 pr12 pr13 pr14
## 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0
##
## Transition matrix
## toS1 toS2 toS3 toS4 toS5 toS6 toS7 toS8 toS9 toS10 toS11 toS12
## fromS1 0 0 0 0 0 0.000 0.207 0.000 0.000 0.000 0.000 0 0.000
## fromS2 0 0 1 0 0 0.000 0.000 0.000 0.000 0.000 0.000 0 0.000
## fromS3 0 0 0 1 0 0.000 0.000 0.000 0.000 0.000 0.000 0 0.000
## fromS4 0 0 0 0 0 0.000 0.000 0.686 0.314 0.000 0.000 0 0.000
## fromS5 0 0 0 0 0 0.000 0.000 0.487 0.000 0.000 0.000 0 0.513
## fromS6 0 0 0 0 0 0.000 0.000 0.000 0.000 0.000 0.000 0 0.000
## fromS7 0 0 0 0 0 0.708 0.000 0.292 0.000 0.000 0.000 0 0.000
## fromS8 0 0 0 0 0 0.000 0.000 0.000 0.000 0.000 0.000 1 0.000
## fromS9 0 1 0 0 0 0.000 0.000 0.000 0.000 0.000 0.000 0 0.000
## fromS10 0 0 0 0 0 0.000 0.000 0.000 0.636 0.000 0.000 0 0.364
## fromS11 0 0 0 0 0 0.000 0.203 0.395 0.402 0.000 0.000 0 0.000
## fromS12 0 0 0 0 0 0.000 0.000 0.000 0.000 0.309 0.691 0 0.000
## fromS13 0 0 0 0 0 0.000 0.924 0.000 0.076 0.000 0.000 0 0.000
## fromS14 1 0 0 0 0 0.000 0.000 0.000 0.000 0.000 0.000 0 0.000
## toS13 toS14
## fromS1 0.000 0.793
## fromS2 0.000 0.000
## fromS3 0.000 0.000
## fromS4 0.000 0.000
## fromS5 0.000 0.000
## fromS6 0.971 0.029
## fromS7 0.000 0.000
## fromS8 0.000 0.000
## fromS9 0.000 0.000
## fromS10 0.000 0.000
## fromS11 0.000 0.000
## fromS12 0.000 0.000
## fromS13 0.000 0.000
## fromS14 0.000 0.000
##
## Response parameters
## Resp 1 : poisson
## Re1.(Intercept)
## St1 -62.519
## St2 0.288
## St3 1.675
## St4 2.336
## St5 1.248
## St6 -1.298
## St7 0.768
## St8 0.024
## St9 -128.099
## St10 0.110
## St11 -0.084
## St12 -156.877
```

## St13	0.109
## St14	0.000