

AR(1) & AR(2) processes

Hedibert Freitas Lopes

4/24/2017

AR(1) process

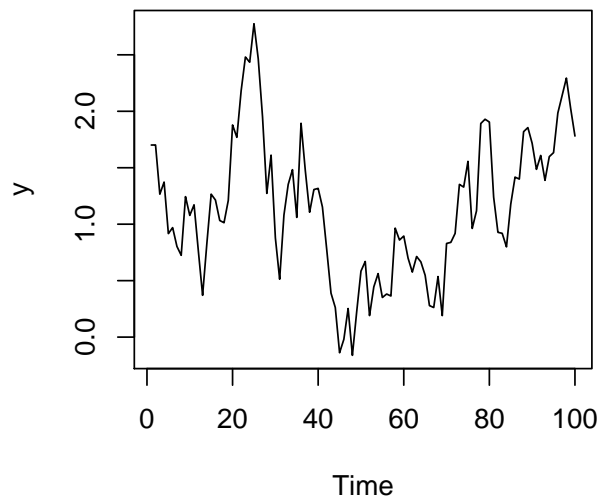
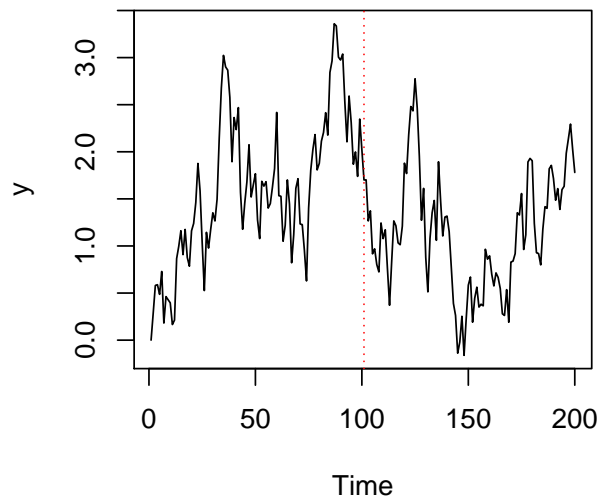
Let us start by simulating $n = 100$ observations from a Gaussian AR(1) process:

$$y_t = \alpha + \phi y_{t-1} + \epsilon_t \quad \epsilon_t \sim N(0, \sigma^2),$$

where the true values of the parameters are $\alpha = 0.1$, $\phi = 0.9$ and $\sigma^2 = 0.1$. Notice that I actually sample $2n$ observations and discard the first n ones.

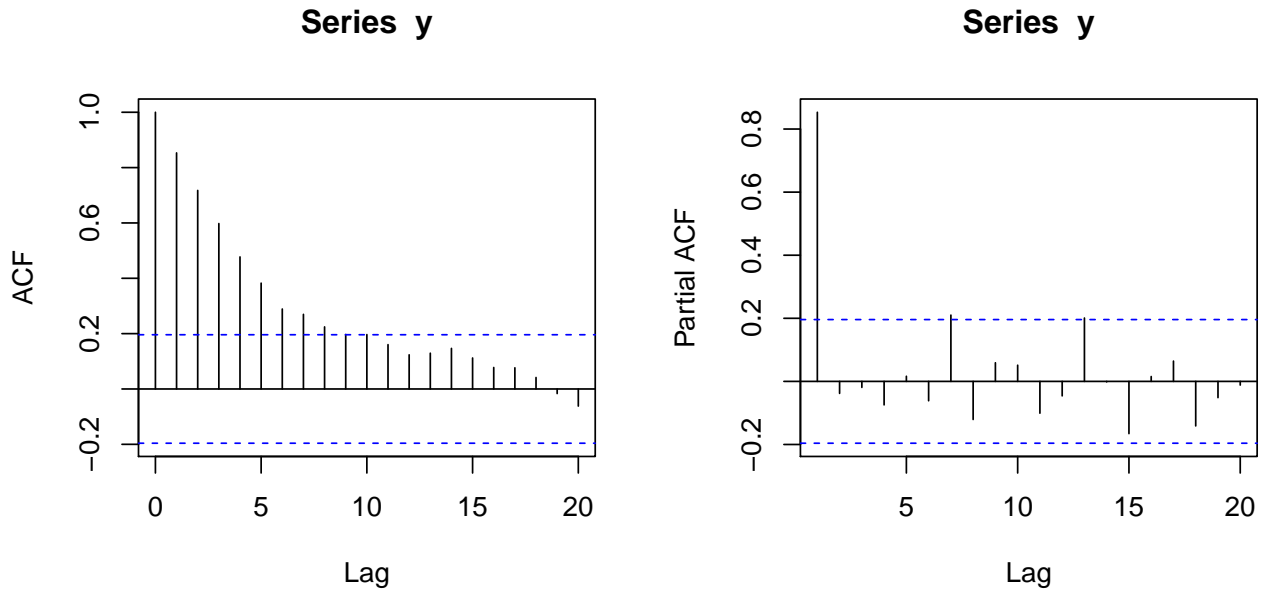
```
set.seed(12345)
n = 100
alpha = 0.1
phit=0.9
sig2 = 0.1
y = rep(0,2*n)
for (t in 2:(2*n))
  y[t]=rnorm(1,alpha+phit*y[t-1],sqrt(sig2))

par(mfrow=c(1,2))
ts.plot(y)
abline(v=101,col=2,lty=3)
y = y[(n+1):(2*n)]
ts.plot(y)
```



We also plot the autocorrelation function and the partial autocorrelation function based on y_1, \dots, y_n .

```
par(mfrow=c(1,2))
acf(y)
pacf(y)
```



Likelihood function

Assuming that α and σ^2 are known, we now compute the likelihood functions of ϕ based on y_1 alone and based on y_2, \dots, y_n . More precisely, when stationarity is assumed we know that $|\phi| < 1$ and the unconditional distribution of y_t (for any t) is $N(\alpha/(1-\phi), \sigma^2/(1-\phi^2))$. On the other hand, when stationarity is not required and/or imposed, we can assume that i) $y_1 \sim N(b, B)$ or that ii) $y_0 \sim N(b, B)$ and, consequently, $y_1|\phi \sim N(\alpha + \phi b, \phi^2 B + \sigma^2)$.

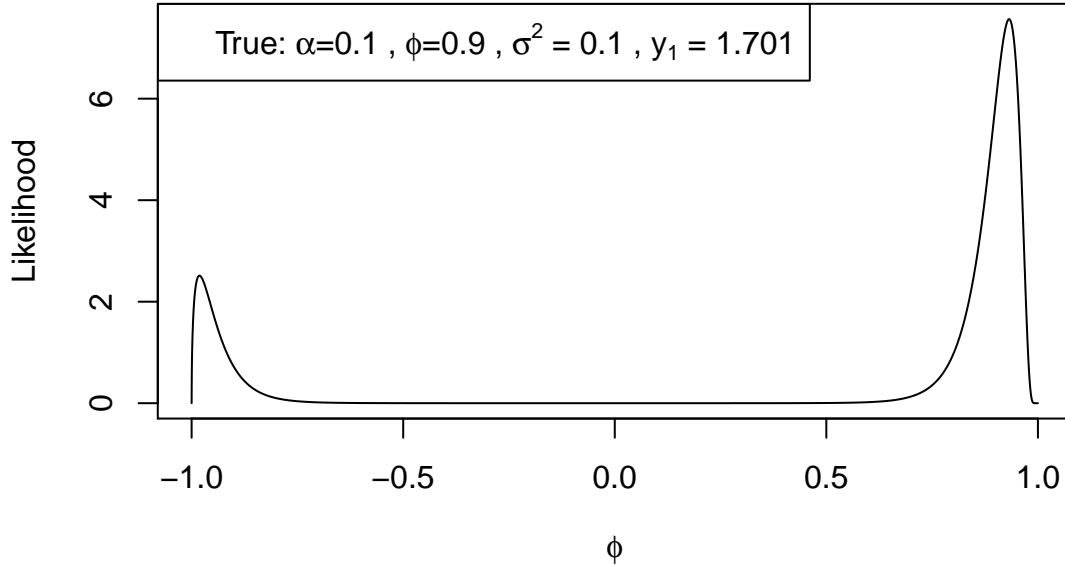
```

b = 0; B=1
N1 = 10000
phis1 = seq(-1,1,length=N1)
h = phis1[2]-phis1[1]
phis2 = seq(-3,3,by=h)
N2 = length(phis2)
like1 = rep(0,N1)
like2 = rep(0,N2)
like1n = rep(0,N1)
like2n = rep(0,N2)
for (i in 1:N1){
  like1[i] = dnorm(y[1],alpha/(1-phis1[i]),sqrt(sig2/(1-phis1[i]^2)))
  like1n[i] = prod(dnorm(y[2:n],phis1[i]*y[1:(n-1)],sqrt(sig2)))
}
for (i in 1:N2){
  like2[i] = dnorm(y[1],alpha+phis2[i]*b,sqrt(phis2[i]^2*B+sig2))
  like2n[i] = prod(dnorm(y[2:n],phis2[i]*y[1:(n-1)],sqrt(sig2)))
}
like1 = like1/sum(like1)/h
like2 = like2/sum(like2)/h
like1n = like1n/sum(like1n)/h
like2n = like2n/sum(like2n)/h
post1 = like1*like1n
post1 = post1/sum(post1)/h
post2 = like2*like2n
post2 = post2/sum(post2)/h

```

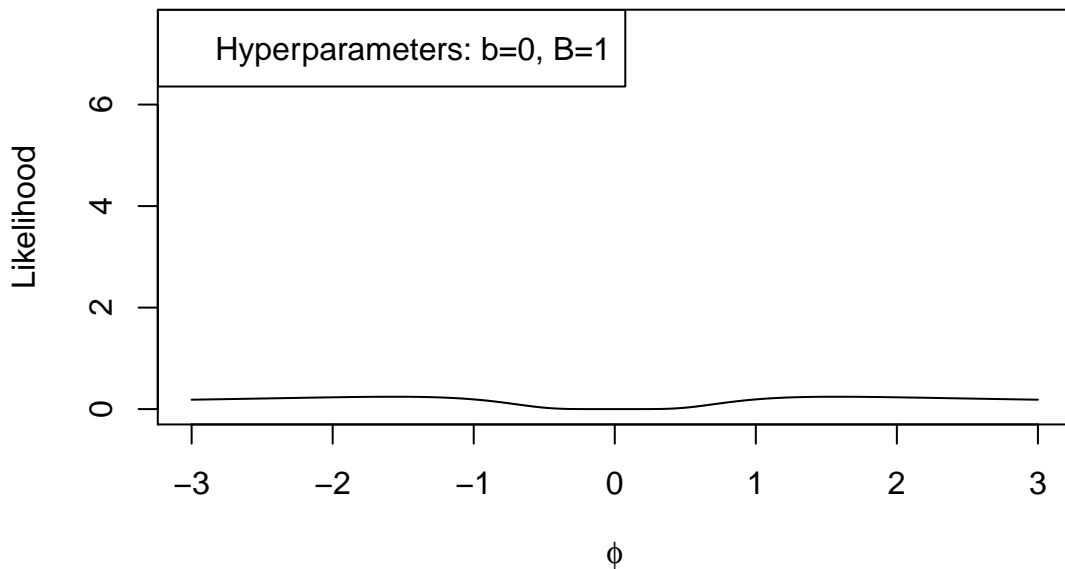
Here we show $L(\phi|y_1)$ when $y_1|\phi \sim N(\alpha/(1-\phi), \sigma^2/(1-\phi^2))$. Notice that the observed value $y_1 = 1.701$ makes the likelihood more peaked in the neighborhood of $\phi = 1$.

```
plot(phis1,like1,type="l",xlab=expression(phi),ylab="Likelihood",ylim=range(like1,like2))
legend("topleft",legend=substitute(paste("True: ",alpha,"=",a," ",phi,"=",b," ",sigma^2,"=",s2," ",y[1],"=",m),list(a=alpha,b=phit,s2=sig2,m=round(y[1],3))))
```



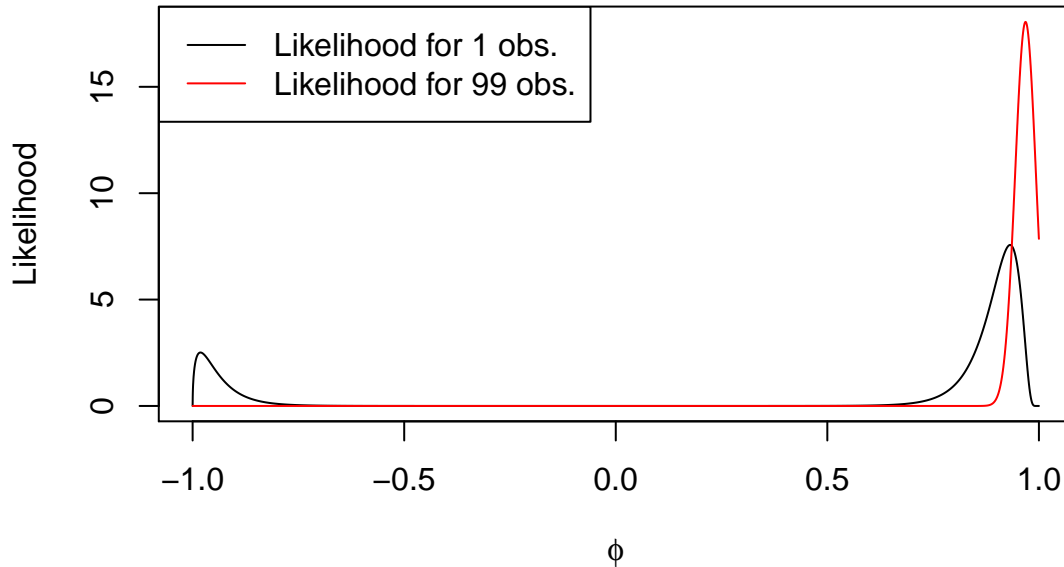
Here we show $L(\phi|y_1)$ when $y_1|\phi \sim N(\alpha + \phi b, \phi^2 B + \sigma^2)$. The likelihood is relatively flat in the real line.

```
plot(phis2,like2,type="l",xlab=expression(phi),ylab="Likelihood",ylim=range(like1,like2))
legend("topleft",legend=paste("Hyperparameters: b=",b," ",B=" ",B,sep=""))
```

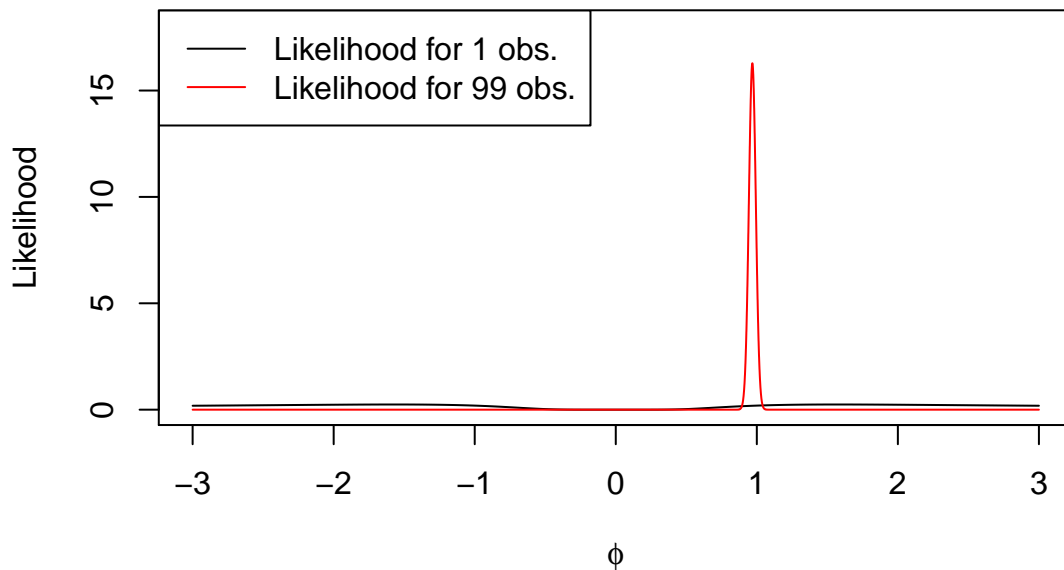


As expected, the likelihood of ϕ based on y_2, \dots, y_{100} is much more informative than the likelihood based on a single observation y_1 , regardless of the chosen distribution for $y_1|\phi$.

```
plot(phis1,like1,type="l",xlab=expression(phi),ylab="Likelihood",
     ylim=range(like1,like2,like1n,like2n))
lines(phis1,like1n,col=2)
legend("topleft",legend=c("Likelihood for 1 obs.",
                          paste("Likelihood for ",n-1," obs.",sep="")),col=1:2,lty=1)
```



```
plot(phis2,like2,type="l",xlab=expression(phi),ylab="Likelihood",
     ylim=range(like1,like2,like1n,like2n))
lines(phis2,like2n,col=2)
legend("topleft",legend=c("Likelihood for 1 obs.",
                          paste("Likelihood for ",n-1," obs.",sep="")),col=1:2,lty=1)
```



Finally, we combine $L(\phi|y_1)$ with $L(\phi|y_2, \dots, y_{100})$

```
plot(phis1, post1, type="l", xlab=expression(phi), ylab="Likelihood",  
     ylim=range(post1, post2), xlim=c(0.85, 1.05))  
lines(phis2, post2, col=2)  
abline(v=1, lty=3)  
legend("topleft", legend=c("Constrained", "Unconstrained"), col=1:2, lty=1)
```

