Gerar o modelo da página 20 da sebenta:

Uma imagem com texto, captura de ecrã, software, Ícone de computador

Descrição gerada automaticamente

> x=rnorm(200,0,1)

> y=rnorm(100,0,5)

> Z=c(x,y)

> plot.ts(Z)

# gerar 500 observações de um modelo AR(1), a=-0.5

ar1=arima.sim(list(ar=-0.5),n=500)

ar1=arima.sim(list(order=c(1,0,0),ar=-0.5),n=500)

grid=matrix(c(1,1,2,3),nrow=2,ncol=2,byrow=TRUE)

layout(grid)

plot(ar1, main="AR(1)", a=-0.5)

acf(ar1)

pacf(ar1)

#para estimar o parâmetro do AR(1)

Aj=arima(ar1,order=c(1,0,0))

Sem constante

Aj=arima(ar1,order=c(1,0,0),include.mean=F)

#######################################3

library(astsa)

sarima(serie1, p=0, d=1, q=3, P=2, D=0, Q=1, S=12, no.constant=T)

########################

ou

m=rnorm(500,0,1)

Y=filter(m,filter=c(-0.5),method=”recursive”)

grid=matrix(c(1,1,2,3),nrow=2,ncol=2,byrow=TRUE)

layout(grid)

plot.ts(Y[1:500])

acf(Y)

pacf(Y)

#######################################

fazer para, 0.8, -0.8, 0.2, -0.2

ar1=arima.sim(list(order=c(1,0,0),ar=0.8),n=500)

ar2=arima.sim(list(order=c(1,0,0),ar=-0.8),n=500)

ar3=arima.sim(list(order=c(1,0,0),ar=0.2),n=500)

ar4=arima.sim(list(order=c(1,0,0),ar=-0.2),n=500)

grid=matrix(c(1,2,3,4),nrow=2,ncol=2,byrow=TRUE)

layout(grid)

acf(ar1,lag.max=20)

acf(ar2, lag.max=20)

acf(ar3, lag.max=20)

acf(ar4, lag.max=20)

###SIMULAR AR(2)

Ar.sim=arima.sim(list(order=c(2,0,0),ar=c(0.6,-0.3)),n=100)

ar.sim<-arima.sim(model=list(ar=c(0.6,-0.3)),n=100)

grid=matrix(c(1,1,2,3),nrow=2,ncol=2,byrow=TRUE)

layout(grid)

ts.plot(ar.sim, main="Processo Autoregressivo",xlab="Tempo")

ar.acf<-acf(ar.sim,type="correlation",plot=T,ylab="FAC")

ar.pacf<-acf(ar.sim,type="partial",plot=T,ylab="FACP")

ou

Ar.sim=arima.sim(list(order=c(2,0,0),ar=c(0.6,-0.3)),n=100)

ar.sim<-arima.sim(model=list(ar=c(0.6,-0.3)),n=100)

ts.plot(ar.sim, main="Processo Autoregressivo",xlab="Tempo")

par(mfrow=c(1,2))

ar.acf<-acf(ar.sim,type="correlation",plot=T,ylab="FAC")

ar.pacf<-acf(ar.sim,type="partial",plot=T,ylab="FACP")

###SIMULAR MA(2)

ma.sim<-arima.sim(model=list(ma=c(-0.7,0.1)),n=100)

par(mfrow=c(1,1,2,3))

ts.plot(ma.sim, main="Processo Médias Móveis",xlab="Tempo")

ma.acf<-acf(ma.sim,type="correlation",plot=T,ylab="FAC")

ma.pacf<-acf(ma.sim,type="partial",plot=T,ylab="FACP")

###SIMULAR (ARMA) PRECISAS DA LIBRARY (“tseries”)

arma.sim<-arima.sim(model=list(ar=c(-0.7,0.1),ma=c(-0.5,-0.3)),n=100)

par(mfrow=c(1,1,2,3))

ts.plot(arma.sim, main="Processo Autoregressivo de Médias Móveis",xlab="Tempo")

arma.acf<-acf(arma.sim,type="correlation",plot=T,ylab="FAC")

arma.pacf<-acf(arma.sim,type="partial",plot=T,ylab="FACP")

ou

arma.sim<-arima.sim(model=list(order=c(2,0,0),ar=c(-0.7,0.1),n=100)

###SIMULAR (SARMA) PRECISAS DA LIBRARY (“tseries”)

Simular um ARIMA

set.seed(4)

Y=arima-sim(10000, model=list(order=c(2,1,0),ar=c(0.7,0.1))

arima(Y,order=c(2,1,0))

## observados vs estimados (original)

netPT.mod = ts(dados[1:204,7], start=c(2000,1), frequency=12)

est = exp(est.log)

res = netPT.mod - est

ts.plot(netPT.mod, main = "Série net Portugal")

lines(ts(est, start=c(2000,1), frequency=12), col="red")

&&&&&&&&&&&&&&&&&&&&&&&&&&&&

data(AirPassengers)

Y=AirPassengers

plot(Y)

mm=filter(Y,rep(1/3,3)) (NOTA: só podem usar números ímpares para a média usar o tempo t e um número de tempos para trás e o mesmo número de tempos para a frente)

mm

ts.plot(Y,main = "AirPassengers")

lines(mm, main = "Média Móvel", col="red")

ts.plot(Y,main = "AirPassengers")

lines(mm, main = "Média Móvel", col="red")

mm1=filter(Y,c(0.7,0.2,0.1),sides=1)

mm1

~~# gravar a figura anterior num ficheiro de formato pdf~~

~~dev.copy2pdf(device=x11,onefile=TRUE,file="ar1\_e\_fac\_amostral.pdf")~~

~~# gravar a mesma figura anterior num ficheiro de formato eps~~

~~# dev.copy2eps(device=x11,onefile=TRUE,file="ar1\_e\_fac\_amostral.eps")~~

~~# gravar a mesma figura anterior num ficheiro de formato png~~

~~png(file="ar1\_e\_fac\_amostral.png", width=700, height=350)~~

~~par(mfrow=c(1,2))~~

~~plot(ar1, main="AR(1), a=-0.5") # cronograma~~

~~acf(ar1) # FAC amostral~~

~~par(mfrow=c(1,1)) # volta ao 'default' de um gráfico por figura~~

~~dev.off()~~

~~# agora vamos simular um AR(2)~~

~~ar2=arima.sim(list(order=c(2,0,0),ar=c(-0.2,0.5)),n=200)~~

~~# e fazer o cronograma com um título "elaborado"~~

~~titulo=expression(AR(2) ~~ a["1"]==-0.2 ~~ a["2"]==0.5 ~ (raizes~reais~distintas))~~

~~plot(ar2,main = titulo)~~

~~# outro gráfico "elaborado": FAC teórica e amostral sobrepostas~~

~~# para um modelo AR(1), com a=0.5~~

~~parameters=-0.5 #define o valor de a~~

~~acfT=ARMAacf(ar=parameters,ma=0, lag.max=25) # calcula a FAC teórica~~

~~plot(acfT,type="h",xlab="lag",ylim=c(-1,1)) # gráfico da FAC teórica~~

~~abline(h=0) # e do eixo dos xx~~

~~# agora geramos 200 observações de um modelo AR(1) com a=0.5~~

~~pdq=c(1,0,0)~~

~~n=200~~

~~ar1=arima.sim(list(order=pdq,ar=parameters),n)~~

~~faca=acf(ar1, plot=F) # calcula a FAC amostral, sem fazer o gráfico~~

~~# acrescenta a FAC amostral no gráfico da FAC teórica~~

~~lines(faca$acf,type="p",pch=20,col="red")~~

~~# "bandas de confiança" para a FAC~~

~~U=2/sqrt(n)~~

~~L=-U~~

~~# acrescenta as bandas no gráfico~~

~~abline(h=c(0,L,U), lty=c(1,2,2), col=c(1,4,4))~~

~~# lty: tipo de linha. 1 solid, 2 dashed~~

~~# col: cor da linha~~