Capítulo 4

Exercício 04

Simule uma série a seu gosto,faça a análise descritiva, verifique se tem raiz unitária, construa um modelo e faca previsões.

Decidimos simular uma serie temporal, com um modelo ARIMA(1,1,1), usando o comando do R:

```
x <- arima.sim(n = 60, list(order = c(1,1,1), ar=c(0.5),ma=c(10)))
```

- n = 60.
- p = 1, auto regressivo com valor 0.5;
- d = 1, para que a série tenha tendência;
- q = 1, com médias móveis de ordem 1 e valor de 10 para o vetor;

Por fim, geramos então um modelo ARIMA(1,1,1)

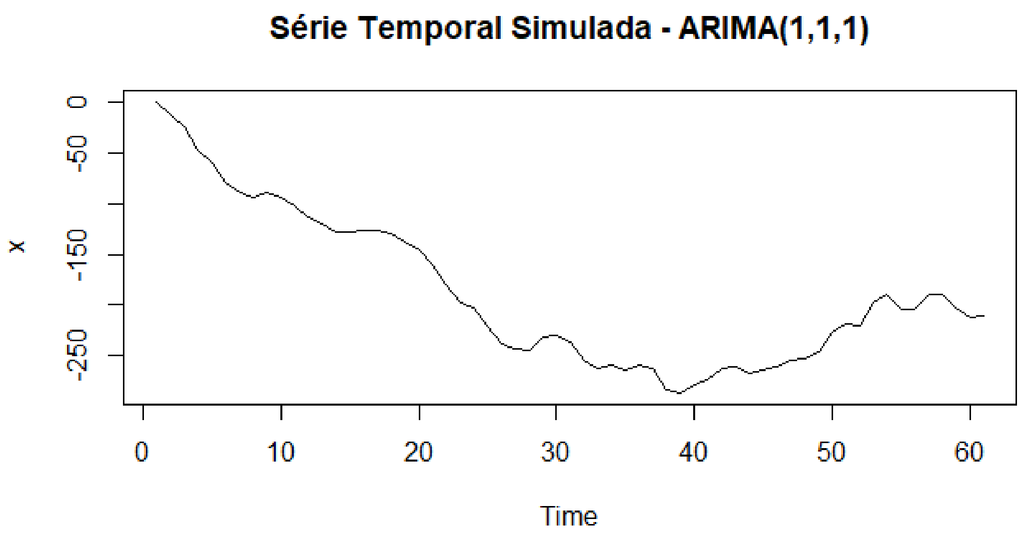

Figura 4.1: Gráfico da Série Temporal simulada

Observando a figura 4.1, podemos notar claramente a tendência que foi adicionada no momento da simulação. A variância não é constante, tendo assim presença de heterocedasticidade e não estacionariedade.

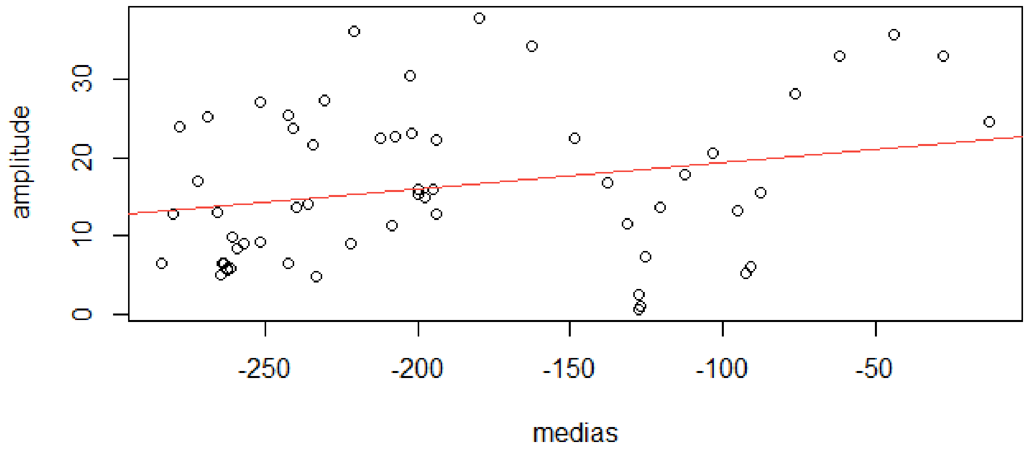

Figura 4.2: Gráfico de Heterocedasticidade

Pelo gráfico 4.2, podemos observar que a variância não aumenta conforme a média aumenta, portanto, não temos presença de heterocedasticidade na série.

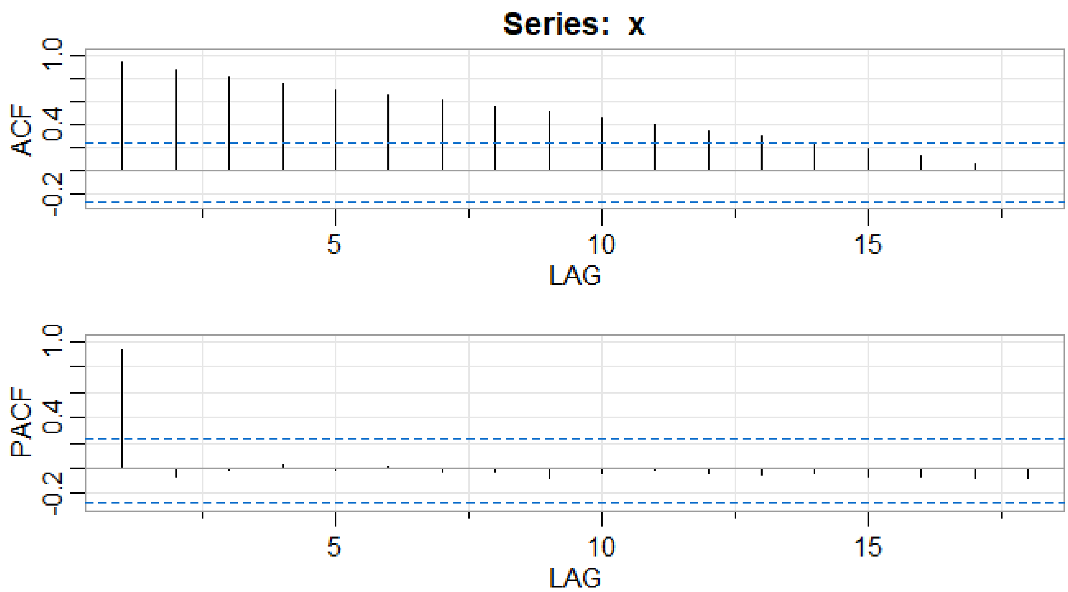

Figura 4.3: Gráficos das Funções de autocorrelação e de autocorrelação parcial do modelo

Observando o gráfico 4.3, nota-se que não temos a presença do lag 0, os lags ao decorrer do tempo vão se aproximando lentamente de zero, e demonstram a existência da tendência na série. Já para a autocorrelação parcial do modelo, o lag 1 é diferente de zero e se destaca dos demais, em seguida temos lags positivos e negativos sendo todos eles não significativos.

Verificando se tem raiz unitária

Para isso vamos usar o Teste Dickey-Fuller.

- Estabelecendo as hipóteses:

$$\begin{cases} H_0 : \textit{A série possui raiz unitária, ou seja, a série não é estacionária} \\ H_1 : \textit{A série não possui raiz unitária, ou seja, a série é estacionária} \end{cases}$$

- Resultados:
  Com a ajuda do R, através do comando adf.test(x,k=1), obtemos os seguintes resultados:

Tabela 4.1: Teste Dickey-Fuller

Dickey-Fuller	Lag order	p-value	alternative hypothesis
-1.2882	1	0.8627	stationary

- Conclusão:
  Como o *p-valor = 0.8627* é maior que $\alpha = 0.05$, não rejeitamos H_0, ou seja, há um nível de significância de 5%, temos indícios de que a série temporal simulada possui raiz unitária, ou seja, a serie não é estacionária

Modelo

Como os dados foram simulados considerando o modelo ARIMA(1,1,1), e pelas Funções de autocorrelação e de autocorrelação parcial do modelo, vamos verificar se o modelo ARIMA(1,1,1) seria adequado através do Gráfico de teste Ljung-Box.

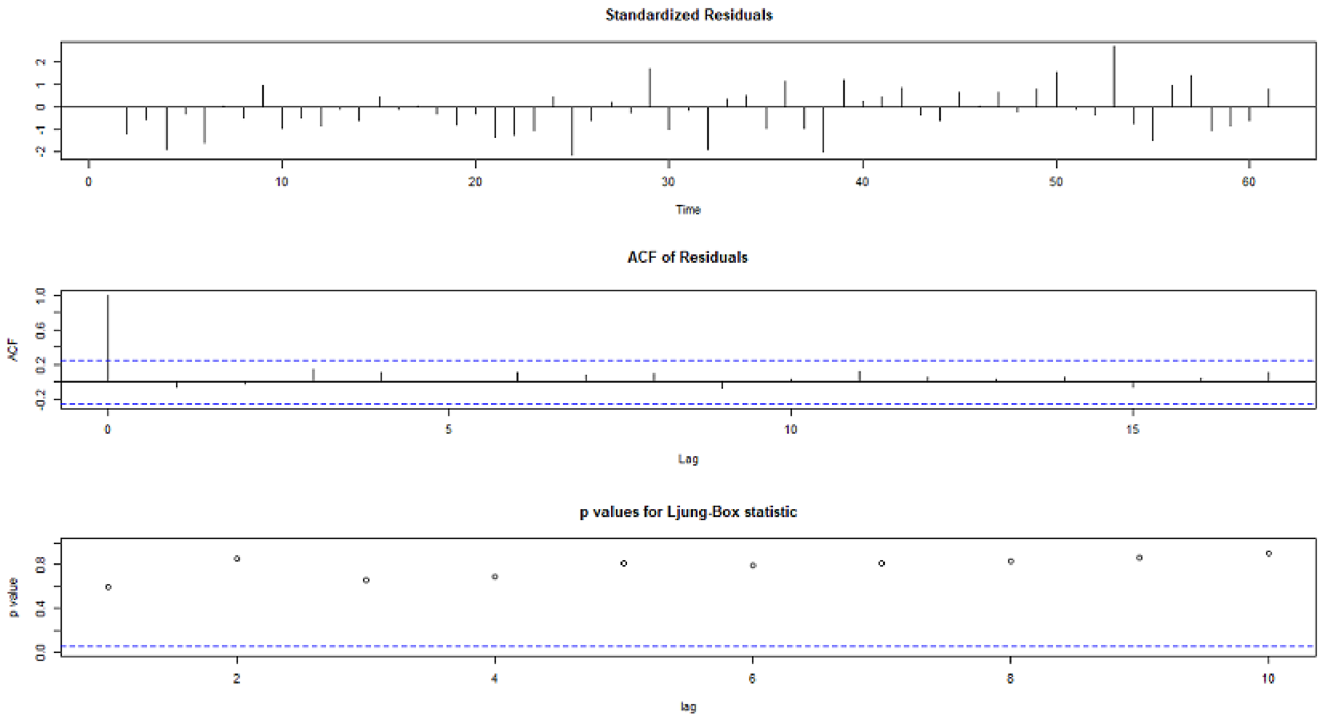

Figura 4.4: Gráfico do teste Ljung-Box para o ajuste ARIMA(1,1,1)

No primeiro gráfico temos os resíduos padronizados, note que, a princípio não vemos nenhum padrão neles, indicando aleatoriedade nos resíduos.

No segundo gráfico temos a função de Auto-Correlação dos resíduos, onde no ponto 0 ele vale 1 e nos demais pontos essas correlações são muito pequenas e concentradas dentro do intervalo limitado pela linha azul, então os meus resíduos são um ruído branco.

No terceiro gráfico, temos o teste Ljung-Box, que testa se os resíduos são ruido branco, como todos os pontos (cada ponto é um valor-p) estão bem distantes da linha azul que representa o nível de significância $\alpha = 0.05$, Portanto, para todos os p-valores apresentados no terceiro gráfico, não rejeitamos H_0 : os resíduos são ruido branco, ou seja, temos evidencias de que os resíduos são ruido branco

Logo, como o ajuste ARIMA(1,1,1) apresentou fortes resultados que, de fato os resíduos são ruido branco, e possuem aleatoriedade, podemos assim concluir que para a nossa série temporal o ajuste com o modelo ARIMA(1,1,1) é adequado.

Previsões

Para realizamos as previsões do modelo, optamos por gerar 10 previsões de tempos a frente, através do comando sarima.for(x,10,1,1,1), obtemos que :

A previsão iniciou-se no tempo 62 e terminou no tempo 71, e os 10 valores previstos são:

-209.5854, -212.7171, -216.2112 -219.7350, -223.2613, -226.7878 -230.3144, -233.8409, -237.3674, -240.8940, e respectivamente o erro padrão associado a cada previsão é: 8.76806, 16.38648, 21.72277, 26.00422, 29.67560, 32.94039, 35.90958, 38.65134, 41.21110, 43.62091.

A seguir para uma melhor visualização temos o Gráfico da série com a previsão do modelo:

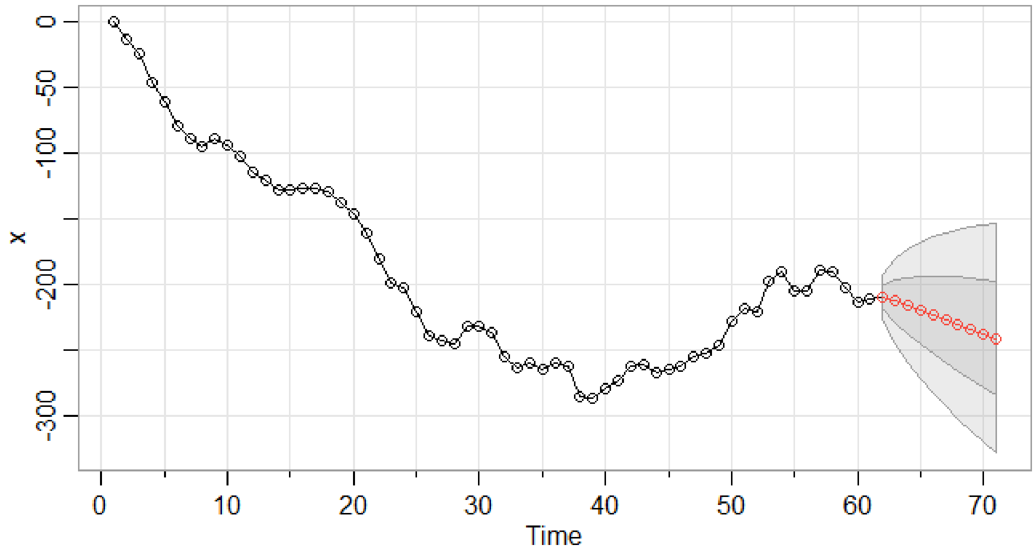

Figura 4.5: Gráfico da série com a previsão do modelo

Como podemos notar, na Figura 4.5, em vermelho estão as previsões obtidas pelo modelo ARIMA(1,1,1), e em cinza são os intervalos de confiança. Como vimos no primeiro tempo previsto temos a previsão de -209.5854 com o erro padrão sendo 8.76806, já no ultimo tempo previsto temos a previsão de -240.8940 com o erro padrão sendo 43.62091, ou seja, conforme tentamos prever tempos mais distantes maior é o nosso erro padrão e a incerteza sobre a previsão.

Algoritmo usado na resolução da questão 4:

```
set.seed(11)
##### Simulando a serie
x <- arima.sim(n = 60, list(order = c(1,1,1), ar=c(0.5),ma=c(10)))

#### Analise descritiva
ts.plot(x, main = "Série Temporal Simulada - ARIMA(1,1,1)")

# Verificando Tendencia
library(randtests)
runs.test(x)
```

```
# Verificando Heterocedasticidade
med.var<-function(x,k)
{N<-length(x)
x.m<-rep(0,(N-k))
x.r<-rep(0,(N-k))
for (i in 1:(N-k)) x.m[i]<-mean(x[i:(i+k)])
for (i in 1:(N-k)) x.r[i]<-max(x[i:(i+k)])-min(x[i:(i+k)])
plot(x.m,x.r,xlab="medias",ylab="amplitude")
aa1<-lm(x.r~x.m)
abline(aa1$coef[1],aa1$coef[2],col=2)
summary(aa1)
}

med.var(x,2) #variancia

library(astsa)
acf2(x)

# Fazendo ajuste do modelo

ajust <- arima(x,order = c(1,1,1))
ajust
tsdiag(ajust)


##### Testes de raiz unitaria
# Se não tem tendencia, eu não tenho raiz unitaria
library(tseries)
#adf.test(x,k=p)

teste_tseries <-adf.test(x,k=1)
teste_tseries


##### estimando valores
library(astsa)
sarima.for(x,10,1,1,1)
```