

Prova 2 resolvida

Mikael Marin Coletto

Índice

1 Questão 1

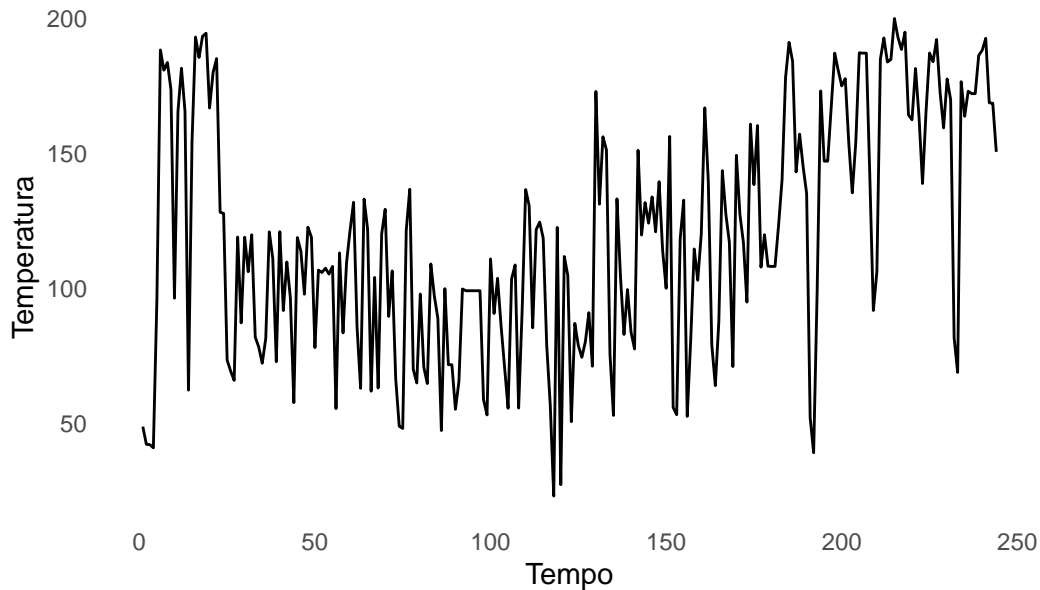
Diga se a série a ser estudada é estacionária utilizando os métodos desenvolvidos em sala de aula, caso a série seja não estacionária qual o procedimento a ser adotado;

```
## Lendo dados
df_temperatura <- readxl::read_xls(paste0(data_path, "Temperatura dos moldes.xls"))

## Transformando em série temporal
ts_temperatura <- ts(df_temperatura$T1, frequency = 1)

## Plotando série
ggplot() +
  geom_line(aes(x = time(ts_temperatura), y = ts_temperatura)) +
  labs(title = "Série temporal da temperatura do molde T1",
       x = "Tempo",
       y = "Temperatura") +
  theme_minimal() +
  theme(panel.grid.major = element_blank(),
        panel.grid.minor = element_blank())
```

Série temporal da temperatura do molde T1



1.1 Investigação visual

Olhando para a série, a série não parece ser estacionária, parece haver um crescimento após o tempo ~125.

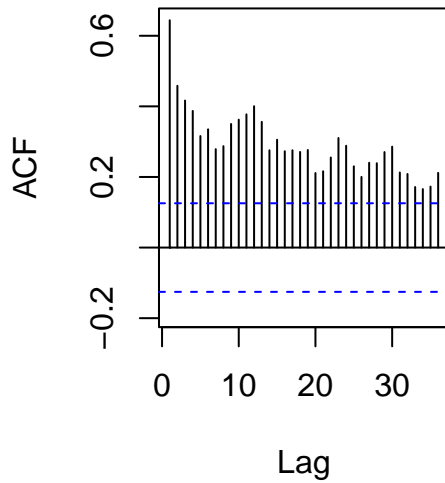
1.2 Olhando para a ACF e PACF

```
par(mfrow = c(1, 2))  
## Autocorrelação dos dados T1  
forecast::Acf(ts_temperatura, type = "correlation", main = "Autocorrelação para T1", lag.max
```

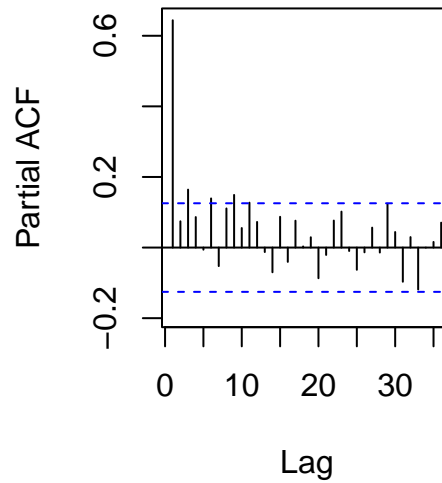
```
Registered S3 method overwritten by 'quantmod':  
  method      from  
as.zoo.data.frame zoo
```

```
## Autocorrelação parcial dos dados T1  
forecast::Acf(ts_temperatura, type = "partial", main = "Autocorrelação parcial para T1", lag
```

Autocorrelação para T1



Autocorrelação parcial para



Pela inspeção visual da ACF, vemos que a lag 12 não decai até 0, indicando que a série pode não ser estacionária, além disso vemos um decaimento ao longo do tempo, indicando que a série parece ter um componente autoregressivo. Na PACF vemos um pico no primeiro lag e uma queda mais brusca, indicando um novo modelo AR(1).

1.3 Teste de estacionariedade

```
library(tseries)
## Teste de estacionariedade
### ADF (Augmented Dickey-Fuller) -> H0 diz que é tem uma raiz unitário, ou seja, tem uma di.
stat_test_adf <- adf.test(ts_temperatura)
# stat_test_adf
### KPSS (Kwiatkowski-Phillip-Schmidt-Shin-> H0 diz que a série não é estacionária
stat_test_kpss <- kpss.test(ts_temperatura)
# stat_test_kpss
```

O teste de Dickey-Fuller aumentado (ADF) é um teste de raiz unitária para uma amostra de uma série temporal. A hipótese nula do teste é que a série temporal possui uma raiz unitária, o que indica que a série não é estacionária. A hipótese alternativa (rejeitando a hipótese nula) é que a série temporal é estacionária. Rodando o teste nos dados da temperatura do molde

T1, temos um resultado de p-valor 0.016, ou seja, **rejeitamos a hipótese nula e aceitamos a hipótese alternativa** de que a **série temporal é estacionária**.

Já o teste de Kwiatkowski-Phillips-Schmidt-Shin (KPSS) é outro teste que verifica raízes unitárias em tendência, e tem sua hipótese nula que a série é não estacionária, e a hipótese alternativa que a série é estacionária. Rodando o teste nos dados da temperatura do molde T1, temos um resultado de p-valor 0.01, ou seja, **rejeitamos a hipótese nula e aceitamos a hipótese alternativa** de que a **série temporal não é estacionária**.

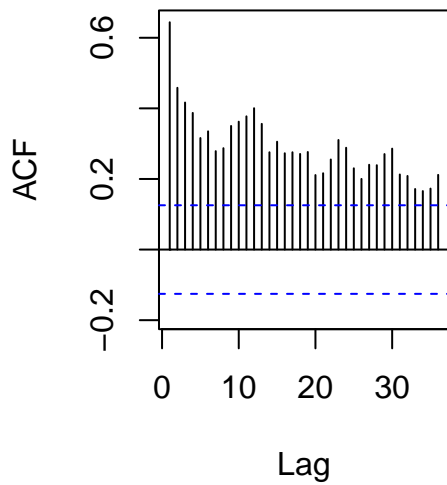
2 Questão 2 e 3

2) Determine a função geradora da série temporal e o provável modelo; 3) Estime no mínimo dois modelos concorrentes que você consideraria importante para os seus estudos futuros com os respectivos parâmetros;

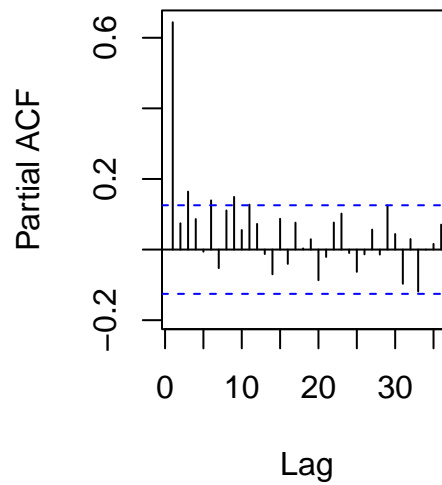
```
## ACF e PACF
par(mfrow = c(1, 2))
# acf(ts_temperatura, main = "ACF")
# pacf(ts_temperatura, main = "PACF")
## Autocorrelação dos dados T1
forecast::Acf(ts_temperatura, type = "correlation", main = "Autocorrelação para T1", lag.max

## Autocorrelação parcial dos dados T1
forecast::Acf(ts_temperatura, type = "partial", main = "Autocorrelação parcial para T1", lag
```

Autocorrelação para T1



Autocorrelação parcial para



Observando a função de autocorrelação (ACF) podemos observar um pico no primeiro lag, e uma pequena queda, porém se mantém relativamente alta e com todos os pontos acima da nossa linha de significância, também não observamos um decaimento até 0 nem mesmo no lag 25 observado, ou seja, precisaremos testar sazonalidade ou diferença.

Já olhando a função de autocorrelação parcial (PACF) observamos um pico no primeiro lag e uma queda mais brusca, onde os demais pontos estão quase todos dentro do limite da linha de significância, indicando um modelo de autoregressão AR(1).

Combinando as duas análises, iremos testar um modelo ARIMA(1, 1, 0).

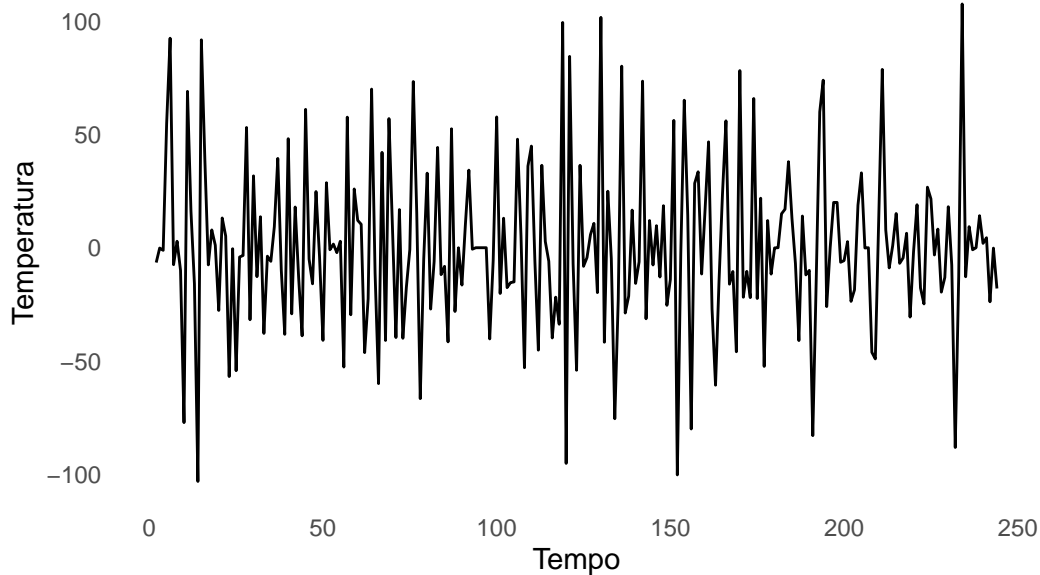
2.1 Fazendo a primeira diferença

```
## Diferenciando a série
ts_temperatura_diff <- diff(ts_temperatura)

## Plotando série diferenciada
ggplot() +
  geom_line(aes(x = time(ts_temperatura_diff), y = ts_temperatura_diff)) +
  labs(title = "Série temporal da temperatura do molde T1 diferenciada",
       x = "Tempo",
       y = "Temperatura") +
```

```
theme_minimal() +  
theme(panel.grid.major = element_blank(),  
      panel.grid.minor = element_blank())
```

Série temporal da temperatura do molde T1 diferenciada



E agora faremos o teste de estacionariedade (Augmented Dickey–Fuller Test) na série diferenciada.

```
stat_test_diff_adf <- adf.test(ts_temperatura_diff)  
# stat_test_diff  
stat_test_diff_kpss <- kpss.test(ts_temperatura_diff)  
# stat_test_diff_kpss
```

Rodando o teste ADF nos dados da temperatura com a primeira diferença, temos um resultado de p-valor 0.01 para o teste ADF e 0.1 para o teste KPSS, ambos os testes agora indicam que a série após a primeira diferença se tornou estacionária.

2.2 Os modelos

Os testes anteriores indicaram uma diferença, e a primeira diferença parece ter tornado a série um pouco mais estável. Além disso, pelo comportamento do gráfico de autocorrelação e autocorrelação parcial parecem haver indícios de um modelo com componente regressivo, o nosso primeiro teste então será um modelo ARIMA de ordem 1, 1, 0.

2.2.1 Modelo 1: ARIMA(1, 1, 0)

```
model_arima_1 <- forecast::Arima(ts_temperatura, order = c(1, 1, 0))

## Coeficientes do modelo
coef_model_arima_1 <- lmtest::coeftest(model_arima_1)
```

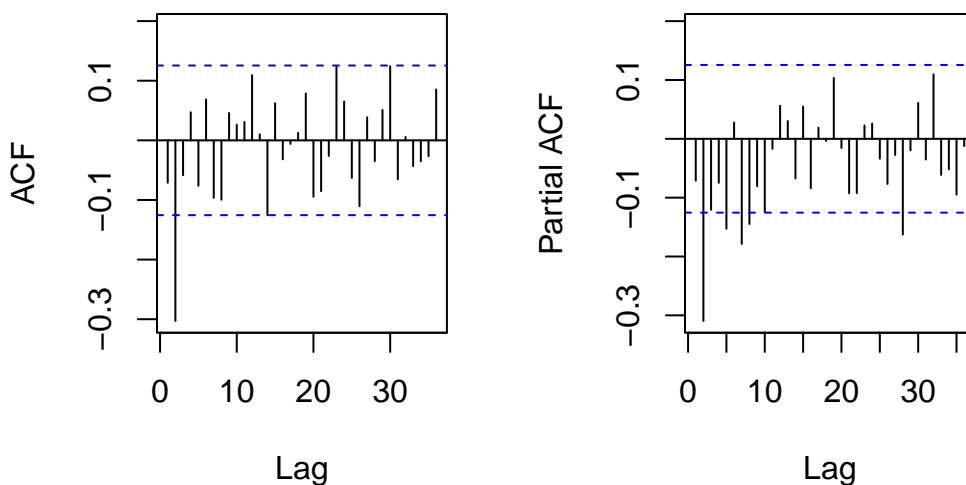
Analisando o modelo ARIMA(1, 1, 0) temos um coeficiente para o componente autoregressivo de -0.246, um valor dentro do considerado válido e um p-valor de 0, indicando que parece sim existir um componente autoregressivo de ordem 1 e que a nossa diferença parece ter contribuído. Vamos analisar a seguir os gráficos de autocorrelação e autocorrelação parcial dos resíduos. O

```
## ACF e PACF dos resíduos
par(mfrow = c(1, 2))

## Autocorrelação dos resíduos
forecast::Acf(model_arima_1$residuals, type = "correlation", main = "Autocorrelação dos resíduos")

## Autocorrelação parcial dos resíduos
forecast::Acf(model_arima_1$residuals, type = "partial", main = "Autocorrelação parcial dos resíduos")
```

Autocorrelação dos resíduos autocorrelação parcial dos res



Após a análise dos resíduos, verificamos que o modelo ARIMA(1, 1, 0) não foi capaz de capturar toda a estrutura da série, pois ainda há autocorrelações significativas nos resíduos. Vamos tentar um modelo ARIMA(1, 1, 1).

2.2.2 Modelo 2: ARIMA(1, 1, 1)

```
model_arima_2 <- forecast::Arima(ts_temperatura, order = c(1, 1, 1))

## Coeficientes do modelo
coef_model_arima_2 <- lmtest::coeftest(model_arima_2)
coef_model_arima_2
```

z test of coefficients:

	Estimate	Std. Error	z value	Pr(> z)
ar1	0.437803	0.067107	6.5239	0.00000000006848 ***
ma1	-0.931568	0.025106	-37.1047	< 0.0000000000000022 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

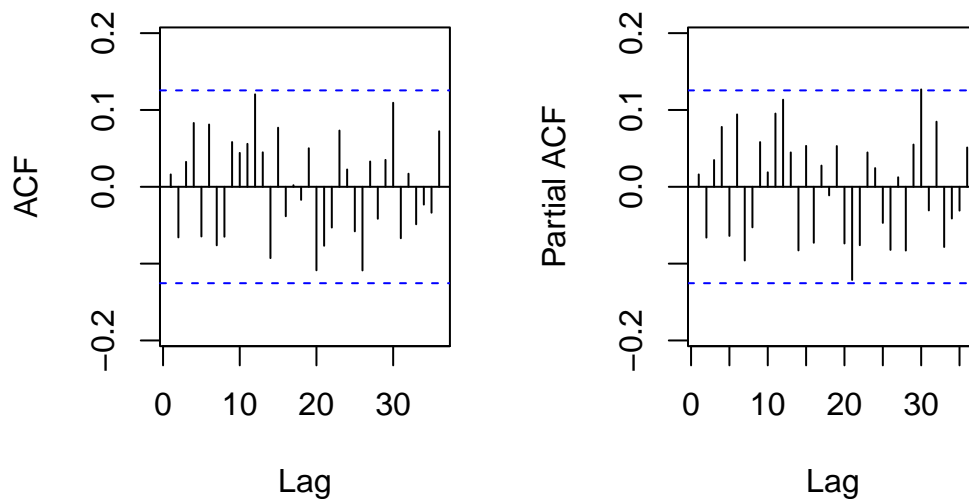
Aqui novamente parecemos ter um modelo adequado, investigando os estimadores estão ambos entre valores -1 e 1, e o p-valor de ambos foi bastante baixo, respectivamente para o componente autoregressivo e para o componente de médias móveis tivemos o estimador de 0.4378029 e -0.931568 e para o valor p tivemos 0 e 0, indicando que o modelo ARIMA(1, 1, 1) é adequado. Vamos analisar os resíduos.

```
## ACF e PACF dos resíduos
par(mfrow = c(1, 2))

## Autocorrelação dos resíduos
forecast::Acf(model_arima_2$residuals, type = "correlation", main = "Autocorrelação dos resíduos")

## Autocorrelação parcial dos resíduos
forecast::Acf(model_arima_2$residuals, type = "partial", main = "Autocorrelação parcial dos resíduos")
```


Autocorrelação dos resíduosautocorrelação parcial dos res



Agora sim já temos os resíduos mais controlados e dentro da nossa linha de significância, indicando que o modelo ARIMA(1, 1, 1) já pode ser usado como um primeiro modelo para estimar essa série.

Agora em sequência iremos trabalhar com modelos que tenham mais componentes autoregressivos e mais componentes de médias móveis e ver como se comporta a nossa estatística de desempenho do modelo (AIC e BIC).

Como comparação, temos AIC e BIC para o modelo ARIMA (1, 1, 1):

```
## AIC e BIC do modelo ARIMA(1, 1, 1)
aic_bic_arima_2 <- c(AIC(model_arima_2), BIC(model_arima_2))
```

AIC = 2384.783612 e BIC = 2395.2627963.

2.2.3 Modelo 3: ARIMA (2, 1, 1)

```
model_arima_3 <- forecast::Arima(ts_temperatura, order = c(2, 1, 1))

## Coeficientes do modelo
coef_model_arima_3 <- lmtest::coeftest(model_arima_3)
coef_model_arima_3
```

z test of coefficients:

	Estimate	Std. Error	z value	Pr(> z)
ar1	0.450888	0.069978	6.4433	0.0000000001169 ***
ar2	-0.055989	0.068687	-0.8151	0.415
ma1	-0.923984	0.029840	-30.9646	< 0.00000000000000022 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Ao avaliarmos o p-valor do segundo componente de autoregressão temos um p-valor de 0.4149925 indicando que ele não é significativo para o modelo, portanto não prosseguiremos com as seguintes análises e com este modelo.

2.2.4 Modelo 4: ARIMA (1, 1, 2)

```
model_arima_4 <- forecast::Arima(ts_temperatura, order = c(1, 1, 2))  
  
## Coeficientes do modelo  
coef_model_arima_4 <- lmtest::coeftest(model_arima_4)  
coef_model_arima_4
```

z test of coefficients:

	Estimate	Std. Error	z value	Pr(> z)
ar1	0.29497	0.17014	1.7337	0.08297 .
ma1	-0.76329	0.17498	-4.3621	0.00001288 ***
ma2	-0.14774	0.14938	-0.9890	0.32264

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

A adição do segundo componente de médias móveis também prejudicou os demais componentes do modelo, portanto também não prosseguiremos com as análises para este caso.

2.2.5 Modelo 5: ARIMA (0, 1, 2)

```

model_arima_5 <- forecast::Arima(ts_temperatura, order = c(0, 1, 2))

## Coeficientes do modelo
coef_model_arima_5 <- lmtest::coeftest(model_arima_5)
coef_model_arima_5

```

z test of coefficients:

	Estimate	Std. Error	z value	Pr(> z)
ma1	-0.504421	0.057123	-8.8304	< 0.00000000000000022 ***
ma2	-0.344221	0.058074	-5.9273	0.000000003079 ***

 Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

## AIC e BIC do modelo ARIMA(0, 1, 2)
aic_bic_arima_5 <- c(AIC(model_arima_5), BIC(model_arima_5))
# aic_bic_arima_5

```

Outro modelo testado que também teve bons resultados foi o modelo ARIMA(0, 1, 2), com estimativas do coeficiente de -0.504421 e -0.344221 e para o valor p tivemos 0 e 0 para o componente de médias móveis 1 e 0.000 para o componente de médias móveis 2, indicando que ambos são significativos para o modelo. AIC = 2386.8010583 e BIC = 2397.2802427. Vamos então analisar seus resíduos.

```

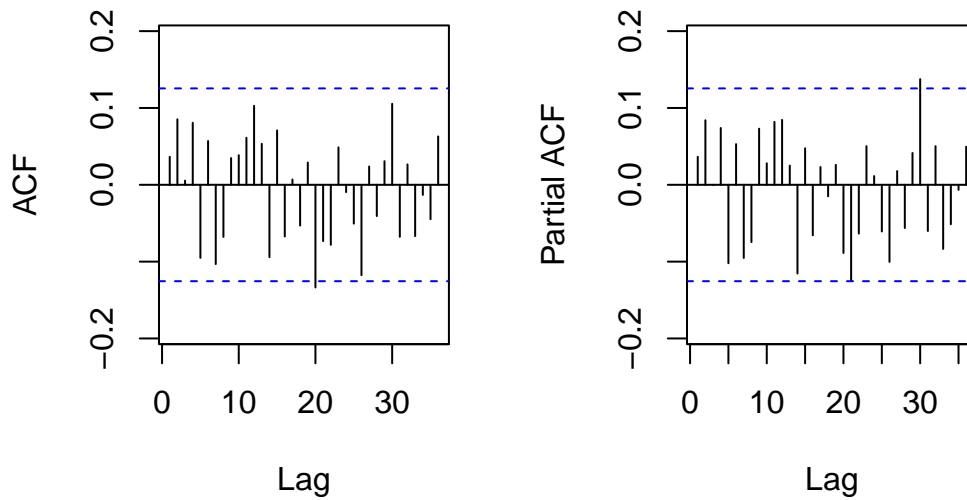
## ACF e PACF dos resíduos
par(mfrow = c(1, 2))

## Autocorrelação dos resíduos
forecast::Acf(model_arima_5$residuals, type = "correlation", main = "Autocorrelação dos resíduos")

## Autocorrelação parcial dos resíduos
forecast::Acf(model_arima_5$residuals, type = "partial", main = "Autocorrelação parcial dos resíduos")

```

Autocorrelação dos resíduos autocorrelação parcial dos res



Os resíduos também se comportaram bem, tendo poucos pontos fora dos limites de significância, indicando que o modelo $ARIMA(0, 1, 2)$ é adequado para a série temporal da temperatura do molde T1.

2.2.6 Escolhendo o modelo

Os modelos que tivemos melhores resultados foram o modelo 2 ($ARIMA(1, 1, 1)$) e o modelo 4 ($ARIMA(0, 1, 2)$), ambos com componentes significativos e com AIC e BIC menores que os demais modelos testados. Portanto, para a série temporal da temperatura do molde T1, os modelos $ARIMA(1, 1, 1)$ e $ARIMA(0, 1, 2)$ são os mais indicados. Olhando pelas métricas de avaliação do quanto o modelo está sendo explicado os valores são bem próximos, porém como temos um AIC e BIC melhores no primeiro modelo, optaremos pelo $ARIMA(1, 1, 1)$.

2.2.7 Questão 4

Estime os resíduos dos modelos concorrentes e verifique se eles possuem característica de ruído branco;

Já analisamos os resíduos de ambos os modelos, e ambos parecem adequados para a série temporal da temperatura do molde T1.

2.2.8 Questão 5

Com o intuito de validar o modelo encontrado para futuras previsões quais os critérios que você utilizaria? Calcule-os;

Como já temos dois modelos que se comportaram bem nos testes de significância dos parâmetros e estão dentro do esperado, podemos avaliar graficamente as previsões dos modelos nos dados atuais e observar qual parece se adequar melhor. Outra questão que também poderíamos avaliar, caso trabalhássemos na área, seria utilizar o que faria mais sentido de acordo com as nossas hipóteses anteriores ou experiências anteriores, ou ainda buscando outros estudos na literatura que pudessem corroborar com a escolha do modelo.

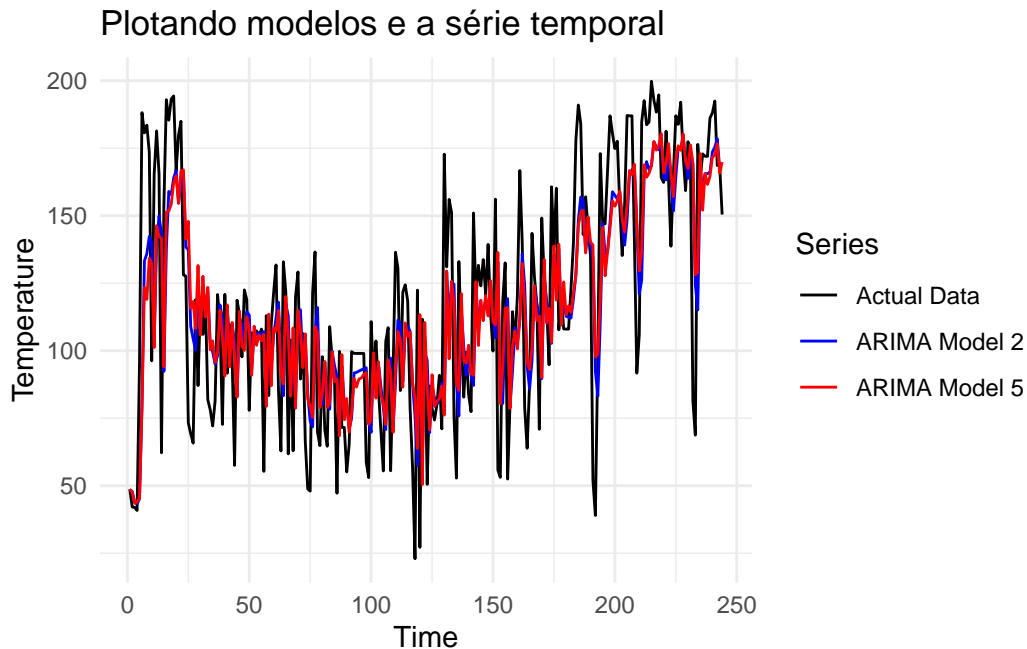
```
## Valores previstos dos modelos
fitted_values_2 <- fitted(model_arima_2)
fitted_values_5 <- fitted(model_arima_5)

## Transformando tudo em um dataframe
df <- data.frame(
  time = time(ts_temperatura),
  actual = as.numeric(ts_temperatura),
  model2 = as.numeric(fitted_values_2),
  model5 = as.numeric(fitted_values_5)
)

# Ajustando dataframe
df_long <- df %>%
  tidyr::pivot_longer(
    cols = c(actual, model2, model5),
    names_to = "series",
    values_to = "value"
  )

## Plotando modelos mais série temporal
ggplot(df_long, aes(x = time, y = value, color = series)) +
  geom_line() +
  theme_minimal() +
  labs(title = "Plotando modelos e a série temporal",
    x = "Time",
    y = "Temperature",
    color = "Series") +
  scale_color_manual(values = c("actual" = "black",
                                "model2" = "blue",
                                "model5" = "red"),
```

```
labels = c("Actual Data",
           "ARIMA Model 2",
           "ARIMA Model 5"))
```



2.2.9 Questão 6

Realize previsões 3 passos à frente do melhor modelo encontrado.

Para realizar a previsão 3 passos a frente, usaremos o modelo ARIMA (1, 1, 1) que foi o escolhido para a série temporal da temperatura do molde T1.

```
## Previsão a frente
previsao <- forecast::forecast(model_arima_2, h = 3)

## Plotando previsão
# plot(forecast_values)

# Criar um gráfico da previsão
# Primeiro criar um dataframe com os dados originais e previstos
dados_originais <- length(ts_temperatura)
tempo_futuro <- seq(from = dados_originais + 1, length.out = 3)
```

```

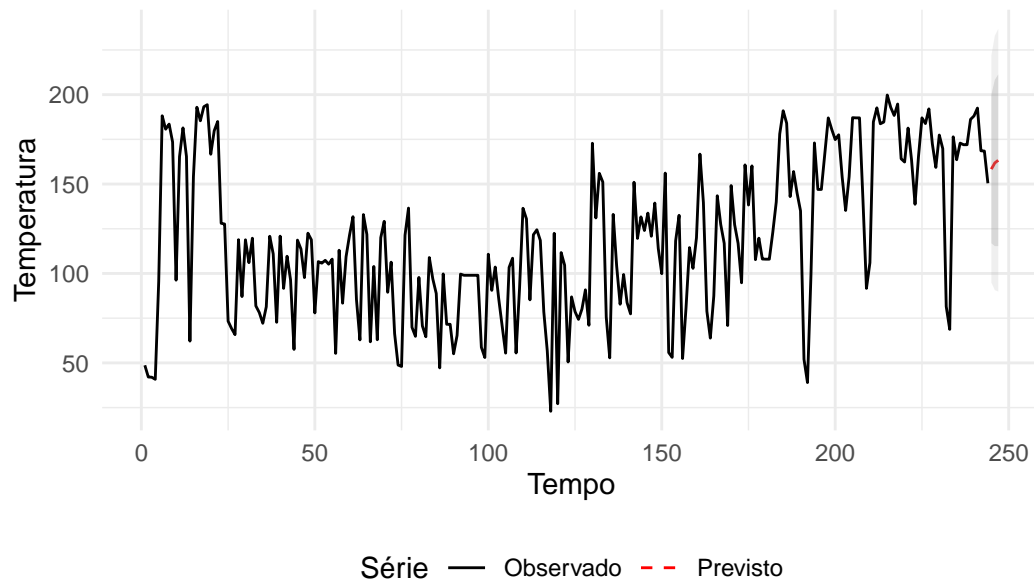
df_previsao <- data.frame(
  time = c(time(ts_temperatura), time(ts_temperatura)[dados_originais] + seq(1:3)/frequency(
  valor = c(as.numeric(ts_temperatura), as.numeric(previsao$mean)),
  tipo = c(rep("Observado", dados_originais), rep("Previsto", 3))
)

# Adicionar intervalos de confiança
df_ic <- data.frame(
  time = time(ts_temperatura)[dados_originais] + seq(1:3)/frequency(ts_temperatura),
  lower80 = as.numeric(previsao$lower[, "80%"]),
  upper80 = as.numeric(previsao$upper[, "80%"]),
  lower95 = as.numeric(previsao$lower[, "95%"]),
  upper95 = as.numeric(previsao$upper[, "95%"])
)

ggplot() +
  # Dados originais e previsão pontual
  geom_line(data = df_previsao,
    aes(x = time, y = valor, color = tipo, linetype = tipo)) +
  # Intervalo de confiança de 95%
  geom_ribbon(data = df_ic,
    aes(x = time, ymin = lower95, ymax = upper95),
    fill = "grey70", alpha = 0.2) +
  # Intervalo de confiança de 80%
  geom_ribbon(data = df_ic,
    aes(x = time, ymin = lower80, ymax = upper80),
    fill = "grey50", alpha = 0.2) +
  # Personalização do gráfico
  theme_minimal() +
  labs(title = "Previsão ARIMA - 3 Passos à Frente",
    x = "Tempo",
    y = "Temperatura",
    color = "Série",
    linetype = "Série") +
  scale_color_manual(values = c("Observado" = "black", "Previsto" = "red")) +
  scale_linetype_manual(values = c("Observado" = "solid", "Previsto" = "dashed")) +
  theme(legend.position = "bottom")

```

Previsão ARIMA – 3 Passos à Frente



No gráfico podemos ver:

- Os dados originais em preto;
- A previsão pontual em vermelho;
- O intervalo de confiança de 95% em cinza claro;
- O intervalo de confiança de 80% em cinza escuro.