

Análise de Séries Temporais - Relatório Final

Universidade Federal do Ceará
Centro de Ciências
Departamento de Estatística e Matemática Aplicada
Bacharelado em Estatística

Antônio Arthur Silva de Lima

06 de junho de 2024

Sumário

Introdução	2
Objetivos	2
Escolha da série	2
Dados de treino e teste	3
Modelagem	4
Modelo de Holt-Winters	4
Ajuste do modelo aditivo	5
Ajuste do modelo multiplicativo	8
Modelos ARIMA	10
Primeiro modelo sugerido	12
Segundo modelo sugerido	14
Terceiro modelo sugerido	14
Modelos alternativos	14

Introdução

Séries temporais são sequências de dados ordenados no tempo ou em outra dimensão (espaço, por exemplo), e sua análise envolve técnicas estatísticas capazes de modelar e extrair padrões significativos do conjunto de observações, a fim de realizar previsões que possam ser úteis no contexto envolvido. Exemplos práticos e muito importantes do tema, são cotações diárias de ações, o índice pluviométrico de dada região, o número de casos de dada doença no país ao longo de certo intervalo, dentre vários outros.

Objetivos

O presente trabalho busca aplicar técnicas e análises específicas a uma série temporal que apresente as componentes de tendência e sazonalidade, que seja recente, preferencialmente brasileira, e contenha ao menos 100 observações; como parte dos requisitos para aprovação na disciplina.

Escolha da série

A **Pesquisa Mensal do Comércio (PMC)**, realizada pelo **Instituto Brasileiro de Geografia e Estatística (IBGE)**, é um dos pilares para compreender o panorama econômico do país e/ou região, de forma a produzir indicadores que acompanham a conjuntura do comércio varejista. De maneira a realizar um estudo real, voltado aos indicadores econômicos do estado do Ceará, a série escolhida para análise é intitulada *Índice de volume de vendas no varejo - total - CE*, que contempla os índices de 2000 a 2022, obtidos como parte da **PMC** nos referidos anos. A série pode ser obtida no domínio público [Sistema Gerenciador de Séries Temporais](#). O gráfico da série é apresentado na Figura 1.

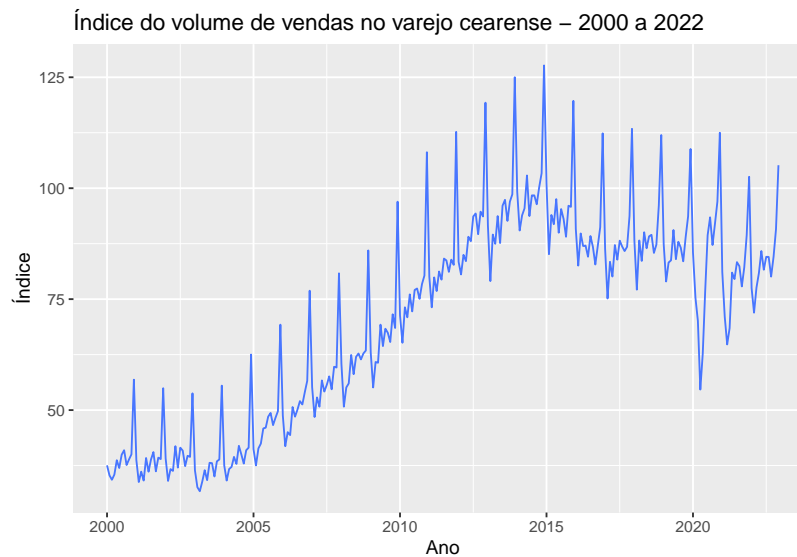


Figura 1: Gráfico da série

A partir do gráfico, percebe-se que a série é claramente não estacionária, contendo uma tendência de crescimento no geral — com alguns decréscimos entre 2015 e 2020 — e também, sazonalidade, com picos nos últimos meses do ano, baixas nos primeiros meses, e altas e baixas nos meses intermediários. Ambas as componentes não aparentam ser determinísticas, já que variam junto com a série, apesar da componente sazonal ter comportamento mais previsível.

Podemos ver tais propriedades de maneira mais específica a partir da decomposição da série, como mostra a Figura 2.

o que faz a decomposição?

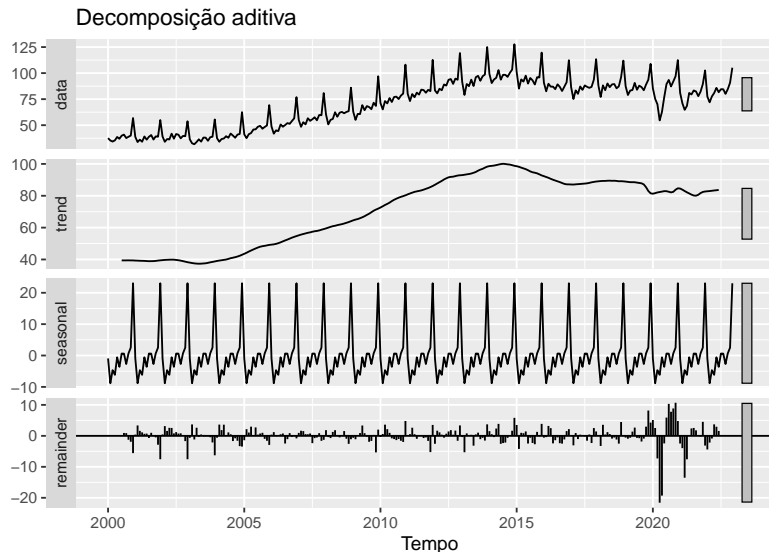


Figura 2: Decomposição clássica

Em um primeiro momento, pode-se inferir que tal comportamento deva-se a presença ou ausência de datas festivas ao longo dos meses, como natal e réveillon em dezembro (altos índices); carnaval, semana santa e festas juninas entre fevereiro e julho. Também é possível destacar a queda acentuada do índice no início de 2020, que foi marcado pela pandemia de Covid-19, e as seguintes altas e baixas, entre 2021 — ainda na pandemia — e 2022, que podem ter sido decorrentes da liberação do programa Auxílio Brasil, como forma de alavancar a economia naquele período.

Dados de treino e teste

Em muitos algoritmos de *machine learning*, costuma-se dividir os dados observados em dados de treino e dados de teste, visando **avaliar** e **validar** o(s) modelo(s) proposto(s). Geralmente, o tamanho desses conjuntos fica em torno de 70% e 30% para o treino e teste, respectivamente, podendo variar de acordo com o problema e também com a quantidade de dados disponíveis.

Na análise de séries temporais, não parece razoável seguir com uma divisão de 70%/30%, visto que, dessa forma, perde-se muita informação, uma vez que temos dados dispostos de forma contínua no tempo. Além disso, o modelo pode ser comprometido se a série tiver mudanças bruscas ou variações fortes no conjunto de teste, pois o mesmo será “cortado” do treinamento do modelo, e assim, previsões podem ser menos realistas. Também deve-se considerar o problema intrínseco de previsões em séries temporais: quanto maior o horizonte de previsão, menos acurada será a mesma.

Assim, parece mais razoável utilizar uma parcela de dados maior para o conjunto de treino — principalmente quando dispomos de muitas observações — pois desta forma, não excluimos os pontos mais recentes da série, e o modelo pode captar mais facilmente os padrões.

Finalmente, iremos dividir nossos dados da seguinte maneira:

- **Treino:** de 2000 a 2019 (240 observações, o que corresponde a aproximadamente 87% da série),
- **Teste:** de 2020 a 2022 (36 observações, o que corresponde a aproximadamente 13% da série).

Utilizando o R, a divisão é feita da seguinte maneira:

```
train = head(dados_ts, 240)
test = tail(dados_ts, -240)
```

onde *train* é o conjunto treino, *test*, o conjunto teste; e *dados_ts* a série no formato *time series* do R.

Modelagem

A partir das informações que já extraímos, podemos agora começar a propor modelos capazes de modelar o tipo de comportamento apresentado pela série. Como vimos que a mesma é não estacionária, isto é, possui tendência e sazonalidade, parece razoável utilizar:

1. Modelos de Holt-Winters (suavização exponencial);
2. Modelos ARIMA.

Modelo de Holt-Winters Este é um dos métodos de suavização exponencial

Modelos de suavização exponencial estão entre os métodos mais eficazes para entender o comportamento de séries temporais. Previsões com base nesse método são, de maneira resumida, médias ponderadas de observações passadas, onde os pesos decaem exponencialmente quanto mais antiga for a observação, ou seja, é uma abordagem que prioriza as observações mais recentes.

Dentro dessa classe de modelos, há três abordagens distintas, que cobrem grande parte de casos, sendo elas:

1. Suavização Exponencial Simples,
2. Suavização Exponencial com Tendência,
3. Suavização Exponencial com Sazonalidade.

O primeiro método é utilizado quando não há tendência e sazonalidade de forma clara, mas não necessariamente uma série estacionária. Uma maneira de representar tal modelo é através de uma componente, chamada comumente de *l*, que denota o nível da série. Assim, pode-se dividir uma equação maior em duas outras equações, sendo elas:

$$\text{Equação de previsão : } \hat{y}_{t+h|t} = l_t$$

$$\text{Equação de suavização : } l_t = \alpha y_t + (1 - \alpha)l_{t-1}$$

onde $0 \leq \alpha \leq 1$ é o parâmetro de suavização do nível, estimado posteriormente por mínimos quadrados, junto a l_0 . Um problema ao utilizar tal método, é que ele nos fornece previsões exatamente iguais ao último nível da série, sendo portanto, um método *naïve*.

No segundo método, além da componente de nível, também está presente a tendência, representada por *b*. Desta forma, temos 3 equações, sendo elas:

$$\text{Equação de previsão : } \hat{y}_{t+h|t} = l_t + hb_t$$

$$\text{Equação do nível : } l_t = \alpha y_t + (1 - \alpha)(l_{t-1} + b_{t-1})$$

$$\text{Equação da tendência : } b_t = \beta(l_t - l_{t-1}) + (1 - \beta)b_{t-1}$$

onde $0 \leq \beta \leq 1$ é o parâmetro de suavização da tendência, também estimado por mínimos quadrados.

Por fim, o último método também leva em consideração séries que tenham sazonalidade, adicionando então, uma componente sazonal denotada por *s*, e também um parâmetro de suavização desta componente, denotado

por γ , estando este entre 0 e 1. Além disso, considera duas formas de como a componente sazonal pode entrar no modelo: de maneira aditiva ou multiplicativa.

O método aditivo é usado nos casos em que as variações sazonais são praticamente constantes, isto é, sem muitas mudanças (o que parece ser o caso da série escolhida), enquanto o método multiplicativo é preferível para situações onde as variações sazonais mudam proporcionalmente ao nível da série.

Deste modo, as equações para ambas as variações do modelo são dadas da seguinte maneira:

- Método **aditivo**

$$\begin{aligned}\hat{y}_{t+h|t} &= l_t + hb_t + s_{t+h-m(k+1)} \\ l_t &= \alpha(y_t - s_{t-m}) + (1 - \alpha)(l_{t-1} + b_{t-1}) \\ b_t &= \beta(l_t - l_{t-1}) + (1 - \beta)b_{t-1} \\ s_t &= \gamma(y_t - l_{t-1} - b_{t-1}) + (1 - \gamma)s_{t-m}\end{aligned}$$

onde m é a periodicidade sazonal (para séries mensais, $m = 12$, para séries quaternais, $m = 4$, por exemplo), e k é dado por $\lceil \frac{h-1}{m} \rceil$.

- Método **multiplicativo**

$$\begin{aligned}\hat{y}_{t+h|t} &= (l_t + hb_t)s_{t+h-m(k+1)} \\ l_t &= \alpha \frac{y_t}{s_{t-m}} + (1 - \alpha)(l_{t-1} + b_{t-1}) \\ b_t &= \beta(l_t - l_{t-1}) + (1 - \beta)b_{t-1} \\ s_t &= \gamma \frac{y_t}{l_{t-1} + b_{t-1}} + (1 - \gamma)s_{t-m}\end{aligned}$$

Finalmente, podemos ajustar os modelos de Holt-Winters aditivo e multiplicativo nos dados de treino da nossa série, realizar previsões, e ver a qualidade do ajuste de cada modelo em relação aos dados de teste, comparando as duas abordagens ao final da modelagem.

Por quê unicamente será ajustado o método de suavização exponencial de Holt-Winters?

Ajuste do modelo aditivo

No R, ajustamos os dados de treino ao modelo aditivo a partir da função `HoltWinters()`, presente no pacote `stats`, da seguinte forma:

```
hw_a = HoltWinters(train)
```

A função irá automaticamente estimar os parâmetros de suavização e as componentes iniciais. Os parâmetros estimados foram:

$\hat{\alpha}$	$\hat{\beta}$	$\hat{\gamma}$
0.470853	0.0493485	0.7640633

Com o modelo pronto, podemos agora realizar uma previsão de 3 anos à frente (36 meses), utilizando a função `forecast()`, do pacote `forecast`, da seguinte maneira:

```
hw_a_f = forecast(hw_a, h = 36)
```

Podemos então, comparar os resultados previstos com os dados do conjunto de teste, como mostra a Figura 3.

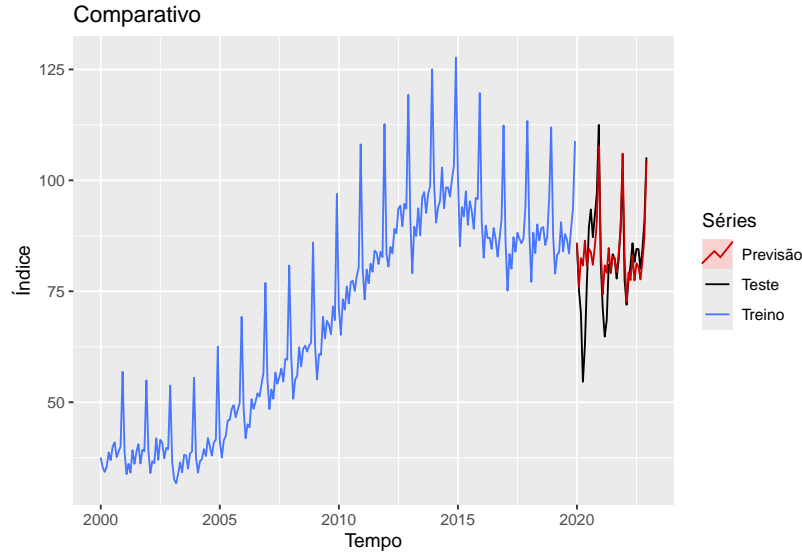


Figura 3: Previsto x Real no modelo aditivo

É possível notar que o modelo foi menos assertivo no ano de 2020, dado que neste ano ocorreu uma mudança brusca no índice, decorrente da queda na economia causada pela pandemia, enquanto os anos de 2021 e 2022 tiveram uma assertividade maior, onde o modelo foi capaz de captar os padrões sazonais no índice e pontos de mudança repentina.

Para validar numericamente o modelo, podemos utilizar algumas medidas capazes de nortear a precisão e acuracidade das previsões, obtendo-as a partir da função `accuracy()`, também do pacote `forecast`:

```
accuracy(hw_a_f, x = test)
```

A função nos retorna os seguintes indicadores:

	ME	RMSE	MAE	MPE	MAPE	MASE	ACF1	Theil's U
Treino	-0.0181999	2.019458	1.532047	0.0128043	2.257447	0.3393039	0.0001888	—
Teste	-1.3987664	7.861042	5.150156	-2.9077920	7.085141	1.1406105	0.7091539	0.9961126

De modo resumido, tais indicadores representam:

- **ME**: o erro médio, dado por $\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)$
- **RMSE**: raiz quadrada do erro quadrático médio, dado por $\sqrt{\frac{1}{n} \sum_{i=1}^n (\hat{y}_i - y_i)^2}$
- **MAE**: erro absoluto médio, dado por $\frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i|$
- **MPE**: erro percentual médio, dado por $\frac{100\%}{n} \sum_{i=1}^n \frac{y_i - \hat{y}_i}{y_i}$

- **MAPE**: erro absoluto percentual médio, dado por $\frac{100\%}{n} \sum_{i=1}^n \left| \frac{y_i - \hat{y}_i}{y_i} \right|$
- **MASE**: erro absoluto escalonado médio, dado por $\frac{1}{n} \sum_{i=1}^n \frac{|y_i - \hat{y}_i|}{\frac{1}{n-1} \sum_{i=2}^n |y_i - y_{i-1}|}$
- **ACF1**: função de autocorrelação no lag 1, dada por $\frac{\sum_{t=1}^{T-1} (y_t - \bar{y})(y_{t+1} - \bar{y})}{\sum_{t=1}^T (y_t - \bar{y})^2}$
- **Theil's U**: estatística U do tipo U2, que mede a qualidade da previsão em relação ao conjunto de teste, dada por
$$\sqrt{\frac{\sum_{i=1}^n \left[\frac{\hat{Y}_i - Y_i}{Y_{i-1}} \right]^2}{\sum_{i=1}^n \left[\frac{Y_i - Y_{i-1}}{Y_{i-1}} \right]^2}}$$

Analisando de maneira conjunta (não é recomendado analisar as métricas de forma individual), é possível concluir que o modelo consegue ter um bom ajuste aos dados, visto que praticamente todos os erros são considerados baixos, especialmente a função de autocorrelação dos resíduos no conjunto de treino. Dentre as medidas acima, uma bastante utilizada e difundida é o **MAPE**, pois entrega uma visão geral e rápida sobre a performance do modelo. No nosso caso, vemos que este valor é de aproximadamente 7,1% no conjunto de teste, reforçando a qualidade do ajuste. Geralmente, valores do **MAPE** abaixo de 10% indicam boa performance. Além disso, a estatística U fica abaixo de 1, indicando que as previsões encontradas pelo modelo são melhores que o método *naive*, isto é, melhores do que um simples chute.

Também devemos checar os resíduos e o comportamento da função de autocorrelação ao longo dos lags, esperando que a mesma seja baixa e dentro do esperado. Isso pode ser visto de maneira conjunta, graficamente, através da função `checkresiduals()` do pacote *forecast*:

O que seria baixa e dentro do esperado?

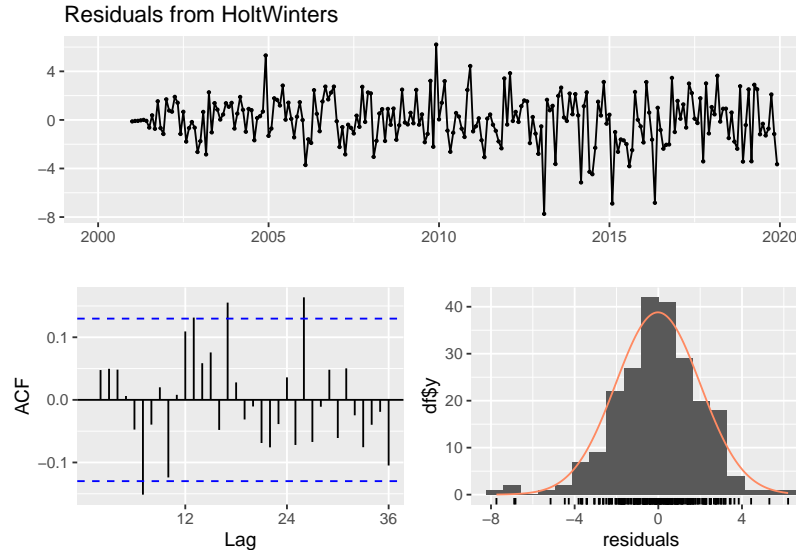


Figura 4: Análise Residual - modelo aditivo

Assim, temos que os resíduos são aproximadamente simétricos em torno do 0, e a função de autocorrelação, com exceção de alguns pontos, tem o comportamento esperado. **Qual seria o comportamento esperado?**

A função `checkresiduals()` também é capaz de retornar o resultado do teste de *Ljung-Box* para autocorrelação, onde H_0 assume que os resíduos são independentes: **Satisfaz este comportamento?**

A hipótese do teste é de independência?

```
##
##  Ljung-Box test
##
## data:  Residuals from HoltWinters
## Q* = 31.588, df = 24, p-value = 0.1375
##
## Model df: 0.   Total lags used: 24
```

Portanto, a um nível de significância de 5%, não temos evidências para rejeitar a hipótese nula de independência entre os resíduos, o que finaliza a validação do nosso modelo aditivo. **O modelo satisfaz todos os supostos?** Esta é a hipótese nula do teste?

Ajuste do modelo multiplicativo

No R, ajustamos os dados de treino ao modelo multiplicativo também a partir da função `HoltWinters()`, presente no pacote *stats*, da seguinte forma:

```
hw_m = HoltWinters(train, seasonal = 'mult')
```

Assim como anteriormente, a função irá automaticamente estimar os parâmetros de suavização e as componentes iniciais. A estimação resultou nos seguintes números:

$\hat{\alpha}$	$\hat{\beta}$	$\hat{\gamma}$
0.4464225	0.0371133	0.7661509

Note que os valores estimados foram próximos ao modelo aditivo, sendo a maior diferença entre as componentes de nível em cada modelo.

Com os valores estimados em mãos, realizamos então a previsão do índice para os anos de 2020 a 2022 (36 meses):

```
hw_m_f = forecast(hw_m, h = 36)
```

Finalmente, iremos comparar graficamente os dados previstos com os dados reais do conjunto de teste, a partir da Figura 5:

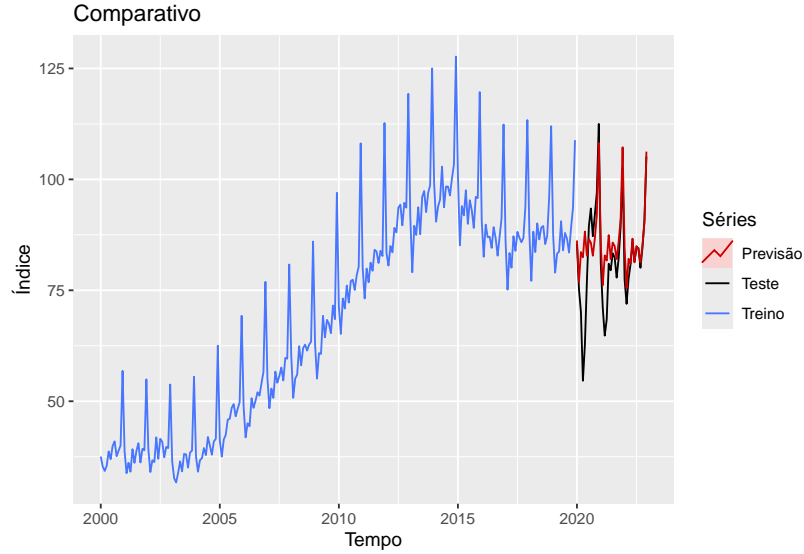


Figura 5: Previsto x Real no modelo multiplicativo

Assim como no modelo aditivo, o modelo multiplicativo não foi capaz de prever a queda brusca do índice em 2020, porém, conseguiu abstrair de maneira competente os padrões dos meses seguintes. O resultado da previsão foi bastante similar ao do primeiro modelo, sendo a diferença mais clara entre os meses intermediários de 2021 (que foi melhor no modelo aditivo) e de 2022 (melhor no modelo multiplicativo).

Para confirmar nossas inferências, iremos utilizar as mesmas métricas de avaliação do primeiro modelo, como mostra a tabela abaixo:

	ME	RMSE	MAE	MPE	MAPE	MASE	ACF1	Theil's U
Treino	-0.0666109	2.110593	1.595995	0.0739974	2.298064	0.3534667	0.0328520	—
Teste	-3.6752537	8.425691	5.393043	-5.7349928	7.556641	1.1944027	0.6956143	1.079249

É notório que os indicadores estão muito próximos aos do primeiro modelo, porém, ainda são um pouco maiores, especialmente a função de autocorrelação no conjunto de treino, que foi da ordem de 100 vezes maior no modelo multiplicativo, apesar de ter sido levemente maior no conjunto de teste no primeiro modelo. Outras medidas que confirmam a inferioridade do modelo multiplicativo são o **MPE**, que foi a medida que apresentou a maior diferença absoluta entre os dois modelos no conjunto de teste; o **MAPE**, sendo aproximadamente meio ponto percentual maior na segunda abordagem; e por fim, a estatística U, a qual indica que a qualidade das previsões realizadas por esse método são inferiores a métodos *naive*.

Também é necessário realizar, assim como no primeiro modelo, os gráficos para entender o comportamento dos resíduos e função de autocorrelação:

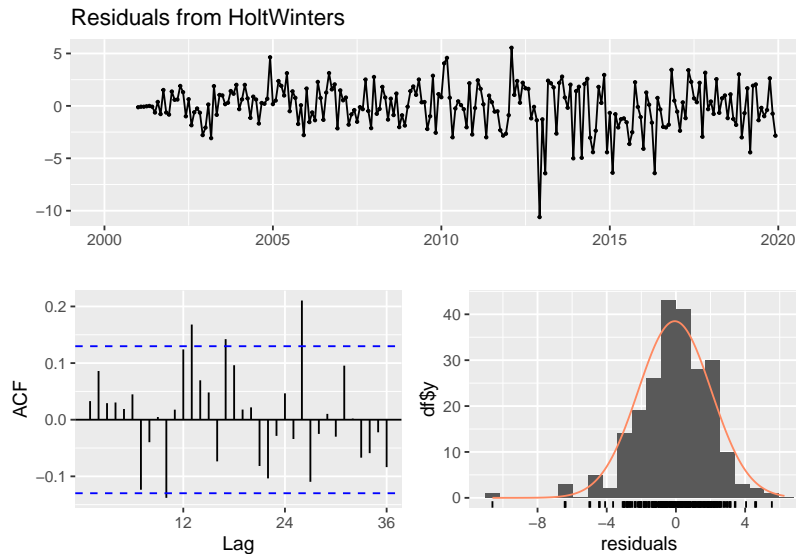


Figura 6: Análise Residual - modelo multiplicativo

Assim, é fácil perceber que os resíduos são levemente assimétricos à esquerda, e que a função de autocorrelação possui um comportamento esperado inferior ao do modelo aditivo, o que pode ser comprovado também pelo teste de *Ljung-Box*:

```
##
## Ljung-Box test
##
## data: Residuals from HoltWinters
## Q* = 38.062, df = 24, p-value = 0.03417
##
## Model df: 0. Total lags used: 24
```

Logo, a um nível de 5% de significância, temos evidências suficientes para rejeitar a hipótese de que os resíduos sejam independentes entre si.

Então, concluímos que o modelo aditivo teve o melhor ajuste, sendo este preferível para modelar a série histórica do índice de volume de vendas no varejo ~~cearense~~, dentro a classe de modelos de suavização exponencial.

Por fim, com os modelos já validados, pode-se ajustar agora o primeiro método à série completa, utilizando os parâmetros estimados no conjunto treino, e em seguida, realizar uma previsão de, por exemplo, 12 meses, o que corresponde aos índices de 2023 (não inclusos no conjunto de dados).

Modelos ARIMA

A classe dos *Modelos Autorregressivos Integrados de Médias Móveis* é bastante ampla e muito utilizada na análise de séries temporais, entregando em geral, previsões com boa qualidade. Esse tipo de modelo, diferente dos métodos de suavização exponencial, tenta descrever os dados por meio do comportamento da sua autocorrelação, ao invés de descrever componentes de tendência e sazonalidade.

Um método muito comum e recomendado ao se trabalhar com modelos ARIMA é o de *Box-Jenkins*, que consiste basicamente em quatro etapas, sendo elas

1. Identificação: onde o objetivo é entender o comportamento da série através de gráficos, realizar transformações caso necessário, e por fim, sugerir um modelo;
2. Estimação: com o modelo escolhido, realizar a estimação dos parâmetros utilizando máxima verossimilhança, mínimos quadrados ou uma combinação de métodos;
3. Validação: fazer o diagnóstico do modelo — por meio de análise residual e testes estatísticos — para verificar a qualidade do ajuste;
4. Previsão: caso o modelo seja validado, realizar a previsão h períodos à frente.

Este processo pode ser iterado várias vezes, até que se encontre um modelo satisfatório.

Aplicando a metodologia na série proposta no trabalho, e sabendo que a mesma não atende às condições iniciais para utilização de modelos ARIMA, podemos realizar transformações, mais especificamente de duas formas: ajustar as componentes sazonal e de tendência, ou realizar diferenças, objetivando encontrar estacionariedade.

Realizar transformações não faz referência a subtrair as componentes estimadas através da decomposição

A primeira abordagem pode ser feita subtraindo da série original, as componentes estimadas através da decomposição. A segunda abordagem, por sua vez, poder ser feita simplesmente realizando uma diferenciação simples (que geralmente é suficiente), ou primeiro realizar uma diferenciação sazonal e em seguida uma diferenciação simples. Iremos considerar a segunda abordagem.

Por quê?

Como nossa série possui forte sazonalidade, iremos primeiro realizar uma diferenciação sazonal, isto é, tomar diferenças com base em um lag de 12 meses. Desta forma, perdemos as primeiras 12 observações.

Melhorar a escrita

```
Zt_diff_SZ = diff(dados_ts, lag = 12)
```

Podemos então verificar os gráficos da série transformada, da autocorrelação e autocorrelação parcial, como mostra a Figura 7.

Por quê considerar primeiro uma diferença sazonal no lugar de uma diferença usual?

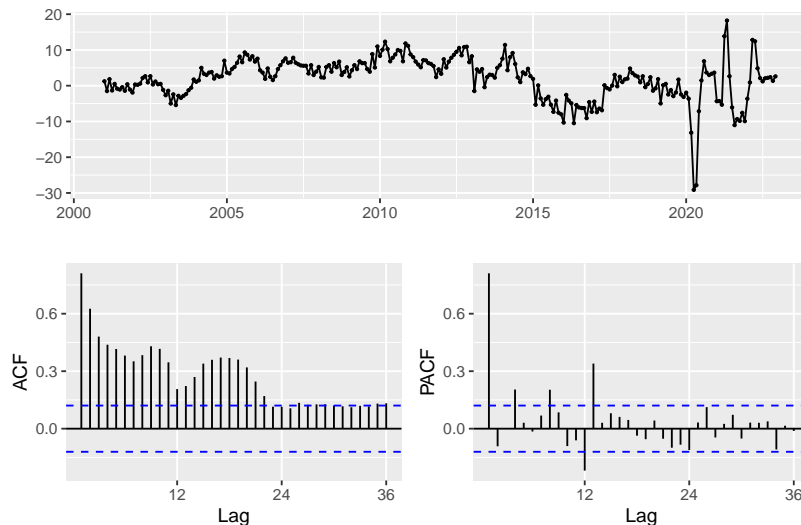
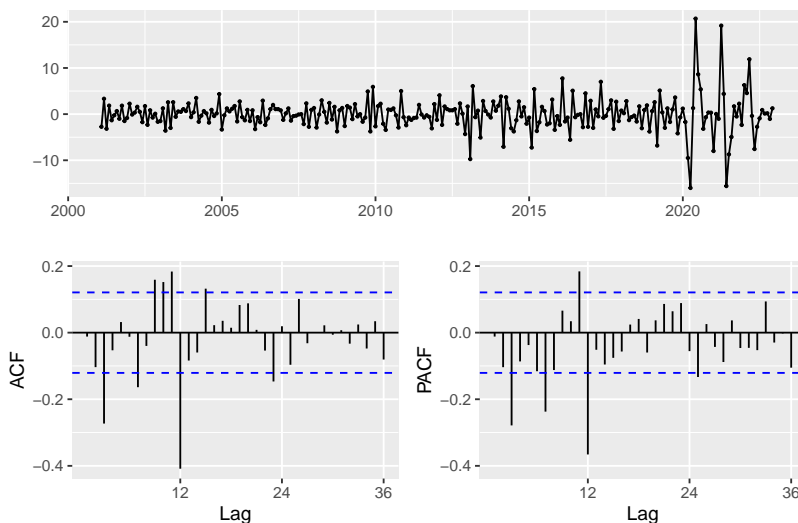


Figura 7: Verificação de estacionariedade

Note que a diferenciação sazonal ainda não torna a série estacionária. Isso pode ser solucionado tomando mais uma diferença, mas desta vez, na forma simples:

```
Zt_diff_SM = diff(Zt_diff_SZ)
```

Verificamos novamente o comportamento da série transformada através dos gráficos.



Por quê?

Com isso, aparentemente a série tornou-se estacionária. Para validar essa hipótese, podemos realizar 2 testes estatísticos, sendo eles o teste aumentado de *Dickey-Fuller* e o teste de *Kwiatkowski-Phillips-Schmidt-Shin* (KPSS). Ambos os testes estão presentes no pacote *tseries*.

```
##  
## Augmented Dickey-Fuller Test  
##  
## data: Zt_diff_SM  
## Dickey-Fuller = -9.9526, Lag order = 6, p-value = 0.01  
## alternative hypothesis: stationary
```

O teste aumentado de Dickey-Fuller nos aponta que há evidências suficientes para rejeitar a hipótese de não estacionariedade (H_0 : série não estacionária).

```
##  
## KPSS Test for Level Stationarity  
##  
## data: Zt_diff_SM  
## KPSS Level = 0.019199, Truncation lag parameter = 5, p-value = 0.1
```

O teste KPSS também indica que os dados sejam estacionários. Desta forma, já validando a nova série como sendo estacionária, podemos então olhar para o gráfico das funções de autocorrelação e autocorrelação parcial, e com base neles, sugerir alguns modelos iniciais, comparando uns com os outros e verificando a performance de cada um.

Primeiro modelo sugerido

Qual é a figura das funções de autocorrelação?

Analisando as funções de autocorrelação, nota-se que ambas aparentam ser curvas sinusoidais, com correlações mais significativas no lag 12. Nota-se também, em ambos os gráficos, aproximadamente 4 ou 5 lags significativos. Além disso, é fácil perceber que as funções decaem para 0 após o 12º lag. Daí, uma primeira sugestão de modelo poderia ser o $ARIMA(12, 1, 12)(0, 1, 0)_{12}$, onde $p = 12$; $d = 1$; $q = 12$. Como foi realizada uma diferença sazonal, também deve-se utilizar $D = 1$, que corresponde à parte sazonal.

No R, isto pode ser realizado da seguinte maneira:

Desta forma está considerando que todos os lags até o lag 12 são significativos, é isto mesmo? Deve apresentar as fac e facp

```
m0_fit = Arima(dados_ts, order = c(12,1,12), seasonal = c(0,1,0))
```

Os coeficientes estimados e outros valores, como a variância dos estimadores, a log-verossimilhança e o critério da informação de Akaike (AIC) são:

```
## Series: dados_ts
## ARIMA(12,1,12)(0,1,0)[12]
##
## Coefficients:
##      ar1      ar2      ar3      ar4      ar5      ar6      ar7      ar8
##      0.1403 -0.1646 -0.2759 -0.0665 -0.2651 -0.0518 -0.2863 -0.1375
## s.e.  0.1074  0.1247  0.1338  0.1647  0.1272  0.1200  0.0997  0.1344
##      ar9      ar10     ar11     ar12      ma1      ma2      ma3      ma4
##      -0.1382  0.0269  0.0191 -0.1814 -0.2951  0.0412  0.0519 -0.0198
## s.e.  0.1617  0.1465  0.1104  0.1161  0.0999  0.1265  0.1351  0.1764
##      ma5      ma6      ma7      ma8      ma9      ma10     ma11     ma12
##      0.2046 -0.0969  0.1222  0.0812  0.1500 -0.1414  0.0088 -0.4868
## s.e.  0.1303  0.1207  0.0939  0.1433  0.1793  0.1764  0.1105  0.1162
##
## sigma^2 = 8.231: log likelihood = -643.32
## AIC=1336.65  AICc=1342.13  BIC=1425.95
```

Todos os parâmetros são significativos?

Afim de validar a primeira proposta de modelo, iremos novamente estudar o comportamento dos resíduos, como mostra a Figura 8.

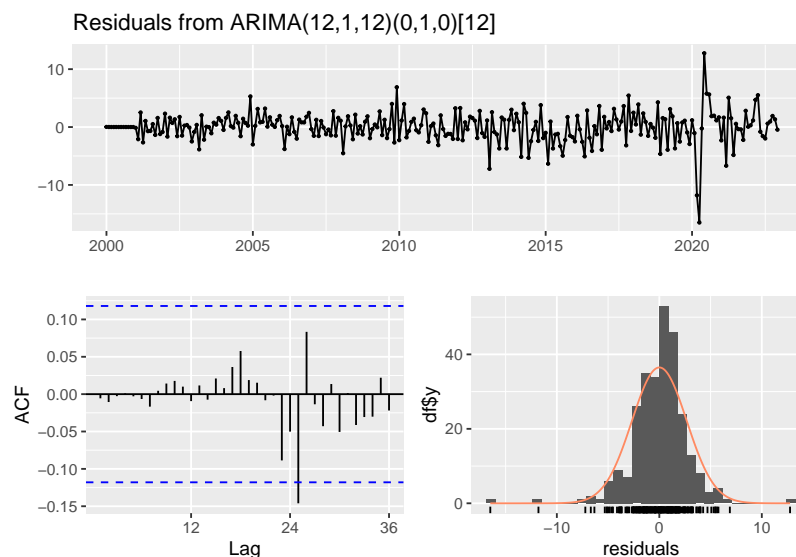


Figura 8: Análise residual

Com isso, nota-se que os resíduos são aproximadamente simétricos ao redor do 0, e só possuem um lag significativo, ainda que com fraca autocorrelação (≈ -0.15).

Podemos também validar o modelo através do teste *Ljung-Box*:

```
##
## Ljung-Box test
##
## data: Residuals from ARIMA(12,1,12)(0,1,0)[12]
## Q* = 13.989, df = 3, p-value = 0.00292
##
## Model df: 24. Total lags used: 27
```

O teste acaba rejeitando a hipótese nula de independência entre os resíduos. ~~Entretanto~~, é muito comum que isso aconteça em vários contextos, não significando necessariamente que o modelo seja ruim. Perceba que os gráficos residuais apresentaram bons resultados, indicando boa qualidade de ajuste. ~~Entretanto~~ **o que significa "bons resultados"?**

Por fim, podemos avaliar a performance do modelo aplicando-o a dados de treino e teste, e verificar algumas métricas de avaliação, assim como nos modelos de Holt-Winters anteriormente. Para isso, aplicamos o modelo ajustado no conjunto treino, realizamos uma previsão de 36 meses, e em seguida, chamamos a função `accuracy()`, obtendo eventualmente os números a seguir.

	ME	RMSE	MAE	MPE	MAPE	MASE	ACF1	Theil's U
Treino	-0.0234018	1.748774	1.303742	0.0084043	1.906071	0.2887411	-0.0500014	—
Teste	-2.2131430	7.791811	4.932066	-3.8699356	6.870279	1.0923096	0.6972654	0.9996533

Com isso, concluímos que o modelo teve um ótimo ajuste, já que fica clara a baixa taxa de erro tanto no conjunto de treino quanto no de teste, menor inclusive que os erros dos modelos de suavização exponencial.

Segundo modelo sugerido

Terceiro modelo sugerido

Modelos alternativos