Universidade Federal de São Carlos
Centro de Ciências Exatas e de tecnologia
Departamento de Estatística

Um Estudo sobre a Provisão de Consórcio com Séries Temporais e *Machine Learning*

Thais Maíra Gonçalves de Lima

Trabalho de Conclusão de Curso

Thais Maíra Gonçalves de Lima

Um Estudo sobre a Provisão de Consórcio com Séries Temporais e *Machine Learning*

Este exemplar corresponde à redação final do trabalho de conclusão de curso devidamente corrigido e defendido por Thais Maíra Gonçalves de Lima e aprovado pela banca examinadora.

São Carlos, 01 de Maio de 2022.

Banca Examinadora

- Maria Silvia de Assis Moura
- Felipe Marangoni
- Francisco Rojas

Dedicatória

Dedico este trabalho para Deus, para a minha orientadora Maria Silvia, para minha família, avós, parentes, amigos e para todas as pessoas que procuram entender e estudar um pouco mais a metodologia de *Machine Learning* dentro de Séries Temporais e/ou que buscam informações sobre o consórcio no Brasil.

Agradecimentos

Agradeço primeiramente a Deus, por estar comigo todos os dias de minha trajetória e pela oportunidade de alcançar essa conquista; à minha orientadora Maria Silvia por todo empenho, dedicação, paciência e ensinamento durante a realização dessa pesquisa; à empresa Stepwise por disponibilizar os dados para estudo; aos meus avós que sempre cuidaram de mim, quando preciso; a minha mãe, Tatiana de Souza Lima, meu pai, Bernardino Gonçalves de Lima, minha irmã, Ana Beatriz Gonçalves de Lima e meu namorado, Murilo Leandro Araujo que sempre me apoiaram a persistir no alcance dos meus sonhos e me motivaram à jamais desistir dos meus objetivos e; todos os parentes e amigos que torceram por mim para a conclusão dessa jornada, o nome de vocês eu guardo dentro do meu coração com muito amor e carinho, muito obrigada por tudo.

Resumo

O setor de consórcio de uma determinada empresa tem como funcionamento a divisão de seus clientes em grupos que iniciam e finalizam ao mesmo tempo o pagamento de parcelas de um valor combinado para a obtenção de um veículo próprio. A aquisição desse automóvel é realizada por sorteios que acontecem em qualquer momento entre o pagamento das parcelas.

A provisão é um capital que a empresa necessita manter para o caso em que houver desistência dos clientes desse grupo do pagamento dessas prestações. Se no encerramento do grupo a empresa não conseguir obter o valor que havia determinado, outro setor da mesma organização suprirá o necessário com o empréstimo que será essencial para o grupo. Logo, quanto maior a inadimplência do grupo, maior será o valor de empréstimo que a empresa deverá ressarcir no consórcio.

Para encontrar modelos ideais que consigam estimar a previsão necessária da provisão e o valor do empréstimo em relação ao tempo, foram utilizadas duas das modelagens usuais de séries temporais: *ARIMA* e *Alisamento Exponencial - Holt Winters*, que mais se aproximou dos valores previstos; e uma aplicação de *Booosting - Machine Learning*.
Palavras-chave: *Alisamento Exponencial, ARIMA, consórcio, empréstimo, Machine Learning, previsão, provisão.*

Abstract

The consortium sector of a particular company works by dividing its customers into groups that start and finish at the same time the payment of installments of a combined amount to obtain their own vehicle. The acquisition of this car is carried out by drawing lots that take place at any time between the payment of the installments.

The provision is a capital that the company needs to maintain in the event that customers in this group withdraw from paying these installments. If, at the end of the group, the company is unable to obtain the amount it had determined, another sector of the same organization will supply the necessary with the loan that will be essential for the group. Therefore, the greater the group's default, the greater the loan amount that the company must reimburse in the consortium.

In order to find ideal models that can estimate the necessary forecast of the provision and the value of the loan over time, two of the usual time series models were used: ARIMA and Exponential Smoothing - Holt Winters, which more approached the predicted values; and a Booosting - Machine Learning application.

Keywords: *Exponential Smoothing, ARIMA, consortium, loan, Machine Learning, forecast, provision.*

Sumário

Lista de Tabelas

Lista de Figuras

Capítulo 1

Introdução

A empresa em estudo é um conglomerado dos setores consórcio, banco e varejo. O enfoque escolhido deste trabalho são os dados fornecidos pelo setor de consórcio de veículos, entretanto entender sobre a organização desse setor é essencial para melhor compreensão do trabalho.

De acordo com Montes (2015), o consórcio vem do latim "consortiu", significa associação, participação e comunidade de bens. Antigamente, no direito grego e romano, embora não existia esse termo em específico, eram criadas algumas associações entre negociadores que se assemelhavam ao comportamento do consórcio atualmente.

O consórcio de empresas aparece pela primeira vez na legislação brasileira em 1960, em que as instituições financeiras são autorizadas a operar no mercado financeiro de capitais para o fim especial de colocar títulos ou valores mobiliários no mercado. A partir desse momento, o sistema de consórcios passa a ser observado como um meio vantajoso no sistema brasileiro, pois realiza a associação de empresários, mediante a assinatura de um contrato, que objetiva realizar um determinado empreendimento. O local em que mais se pode ver a atuação dos consórcios é nas rodovias (Montes, 2015).

Na empresa em estudo, o consórcio funciona da seguinte maneira: a empresa seleciona um determinado grupo de clientes que combinam o pagamento parcelado do valor total entre eles, com o objetivo de serem sorteados em algum momento do prazo definido para obterem um veículo equivalente ao valor final que será pago. Cada cliente, pode ter mais de um acordo como esse que é denominado como "cotas" e a organização precisa se planejar em relação a todo esse período com o cliente.

Caso acontecer algum imprevisto e um dos clientes do grupo não conseguir dar prosseguimento ao pagamento dessas parcelas, a empresa deve se planejar para ter esse valor

em mãos. Essa função é denominada Provisão e será estudada nesse trabalho. A Provisão analisa grupo à grupo com sua respectiva previsão de venda e possui como objetivo entender o quanto a empresa está vendendo, denominada como *visão carteira*.

Se todos do grupo de consórcio pagassem corretamente as parcelas conforme o combinado, as cotas contemplariam os cotistas e no encerramento do grupo, a empresa teria recebido o valor certo referente a esse grupo. Como isso não é sempre que acontece, o setor banco dessa empresa realiza um empréstimo para esse grupo com o intuito dele atingir o valor determinado anteriormente, essa informação está descrita como empréstimo no conjunto de dados. Os inadimplentes serão cobrados posteriormente a respeito dos valores não recebidos pela empresa.

Receive (2021) destaca que LGD se refere a *Loss Given Default* (LGD) — ou perda dada à inadimplência, em português que significa uma taxa de default,referente ao valor efetivamente perdido na operação pelo descumprimento contratual permanente por parte do devedor. Logo na Provisão pode ocorrer exatamente isso, pois existe a possibilidade da perda do dinheiro ou do cliente no começo, meio ou final do acordo, gerando uma curva de inadimplência, percepção de risco ou o cancelamento que são fatores de riscos que atrapalham uma boa previsão financeira.

A previsão do empréstimo e da provisão, utilizando esses dados fornecidos pela empresa, auxilia na obtenção de um possível e preciso resultado de maneira prévia, possibilitando um planejamento e preparo de forma antecipada, caso algum cliente não consiga pagar conforme o combinado. E essa técnica não está presente somente agora, Hyndman e Athanasopoulos (2018) relatam a existência desse termo desde o ano de 700 a.C. na Bíblia, em que era utilizado como sinal de inspiração divina.

Neste trabalho, serão apresentados possíveis modelos que se ajustam aos dados, bem como suas respectivas análises de diagnósticos e será escolhido qual modelo possui a melhor predição aos dados. Depois, as previsões pelo modelo escolhido serão utilizadas para conseguir identificar a necessidade de um capital abaixo ou acima do esperado pela empresa de consórcio. Esses resultados são influenciados por clientes bons ou maus pagadores.

Para manter o nível de qualidade da previsão e reproduzí-la de maneira otimizada e equilibrada, será necessário a aplicação de modelos de Séries Temporais, utilizando as metodologias *ARIMA* e *Alisamento Exponencial* e da inovação do *Machine Learning*, apresentando a abordagem *XGBoost*.

Izbicki e dos Santos (2020) relatam que o *Machine Learning* nasceu na década de 60

como um campo da inteligência artificial para aprender padrões com base em dados. O *Machine Learning* é baseado em um algoritmo de regressão que captura os padrões que existem em todo o conjunto de dados (Pavlyshenko (2018)).

Ao longo de todo este trabalho serão apresentados modelos de diferentes abordagens estatísticas que consigam predizer, de maneira eficiente, o capital ideal que a empresa de consórcio necessita para se manter funcionando e com boa performance. Para cada técnica estudada, serão mostrados os requisitos para sua utilização e se realmente é a mais adequada, quando comparada aos demais.

O trabalho está estruturado da seguinte maneira, no Capítulo 2 é apresentado a definição de previsão e séries temporais, além das metodologias, *ARIMA* e *Alisamento Exponencial*, bem como a técnica *XGBoost*. No Capitulo 3 têm-se a análise de dados e os resultados obtidos das diferentes técnicas para as variáveis **Provisao_Total** e **Emprestimo**, contando também com a comparação entre as metodologias. No capitulo 4, apresenta-se a conclusão dessa pesquisa.

Capítulo 2

Material e Métodos

2.1 Previsão

A previsão é um procedimento estatístico muito utilizado nos negócios, auxiliando nas decisões sobre a programação de produção, transporte e pessoal, e fornece um guia para o planejamento estratégico de longo prazo. É válido destacar que ela é definida como uma previsão do futuro com a maior precisão possível(Hyndman e Athanasopoulos (2018)). Com esse contexto, nota-se que é necessário a utilização da previsão no conjunto de dados da empresa de consórcio, pois a partir disso é possível garantir uma tomada de decisão em muitas áreas dessa organização.

Nielsen (2019) destaca a ocorrência da urgência em desenvolver análises de previsões que podem ser realizadas de forma eficiente. Esses esforços de modelagem de previsão visam um bom desempenho que descrevem seus modelos como um aprendizado profundo em paralelo com o desenvolvimento de modelos estatísticos tradicionais, investigando como esses podem ser melhorados. Logo, procura-se um modelo de alto nível de qualidade para as previsões e ao mesmo tempo com uma performance adequada.

De acordo com Hyndman e Athanasopoulos (2021), o processo de previsão se divide em cinco passos básicos, que são:

- **Passo 1: Definição de problemas** - Definir o problema cuidadosamente requer uma compreensão da forma como as previsões serão usadas, quem requer as previsões e como a função de previsão se encaixa dentro da organização: coleta de dados, manutenção de bancos de dados e uso das mesmas para o planejamento futuro;

- **Passo 2: Coleta de informações** - Existem pelo menos dois tipos de informações

necessárias: dados estatísticos e o conhecimento das pessoas que coletam os dados e utilizam as previsões. Dados antigos serão menos úteis devido a mudanças estruturais no sistema que estão sendo previstas e o ideal é optar sempre por dados mais recentes;

- **Passo 3: Análise preliminar (exploratória)** - O ideal é sempre realizar gráficos com os dados e identificar se: há padrões consistentes? Existe uma tendência significativa? A sazonalidade é importante? Há evidências da presença de ciclos de negócios? Há *outliers* nos dados que precisam ser explicados por aqueles com conhecimento especializado? Quão fortes são as relações entre as variáveis disponíveis para análise? Essa visualização é essencial para dar prosseguimento a uma previsão adequada;

- **Passo 4: Escolha e ajuste de modelos** - O melhor modelo de uso depende da disponibilidade de dados históricos, da força das relações entre a variável de previsão e quaisquer variáveis explicativas, e da forma como as previsões devem ser utilizadas. É comum comparar dois ou três modelos em potencial. Cada modelo é, em si, uma construção artificial que se baseia em um conjunto de suposições (explícitas e implícitas) e geralmente envolve um ou mais parâmetros que devem ser estimados usando os dados históricos conhecidos;

- **Passo 5: Usar e avaliar um modelo de previsão** - Uma vez que um modelo tenha sido selecionado e seus parâmetros estimados, o modelo é usado para fazer previsões. O desempenho do modelo pode ser avaliado antes e após disponibilizar parte dos dados para avaliá-lo.

2.2 Séries Temporais

Uma série temporal é um conjunto de observações obtidas sequencialmente no tempo, que possui como característica central a presença de dependência ou correlação temporal entre elas, dada por:

$$\left\{Y_t, t = 1, 2, ..., T\right\}, \tag{2.1}$$

em que a variável de interesse Y pode ser univariada ou multivariada (Costa (2015)).

Uma série temporal pode ser vista como uma realização de um processo estocástico, $\left\{Y_t, t \in I\right\}$ onde em cada $t \in I$, Y_t é uma variável aleatória definida num espaço de probabilidades $(\Omega, \mathcal{A}, \mathcal{P})$. O conjunto de índices temporais pode ser discreto $I = \left\{1, 2, ..., T\right\}$ ou contínuo $I = \left\{t : t_1 < t < t_2\right\}$ dando origem a processos em tempo discreto ou contínuo, respectivamente.

2.2.1 Modelo $ARIMA$

Segundo Hyndman e Athanasopoulos (2021), as seguintes definições são essenciais para prosseguimento do modelo:

- **Tendência:** Tendência é o aumento ou diminuição do curto, médio ou longo prazo nos valores Y_t da série, quando ela passa de uma tendência crescente para uma tendência decrescente;

- **Sazonalidade:** Ocorre quando uma série temporal é afetada por fatores, como a época do ano ou o dia da semana; acontece sempre em um período de comprimento fixo e conhecido. Nielsen (2019) ressalta que qualquer tipo de comportamento recorrente da frequência é um exemplo de sazonalidade;

- **Ruído branco:** Uma sequência de variáveis aleatórias que devem possuir média zero ,variância constante (homoscedasticidade) e as observações não correlacionadas;

- **Estacionariedade:** Uma série temporal *estacionária* é aquela cujas propriedades estatísticas são μ_t e σ_t, para qualquer t . Assim, séries com *tendências* ou com *sazonalidade*, não são estacionárias.

- **Diferenciação:** É um procedimento para tornar uma série temporal não estacionária em estacionária, isto é, calcular as diferenças entre observações consecutivas que pode ajudar a estabilizar a média de uma série temporal, removendo mudanças no nível de uma série temporal e, portanto, eliminando ou reduzindo *tendência* ou *sazonalidade* se for feita uma diferença sazonal;

- **Gráficos:** Os gráficos ACF e $PACF$ das Funções de Autocorrelação e de Autocorrelação Parcial mostram as autocorrelações entre Y_t e Y_{t-k} para os diferentes valores de k. Caso Y_t e Y_{t-1} estiverem correlacionados, então Y_{t-1} e Y_{t-2} também devem ser

correlacionados ou podem estar correlacionados, simplesmente porque ambos estão conectados a Y_{t-1}.

Para auxiliar na modelagem $ARIMA$, os gráficos ACF e $PACF$ podem ser utilizados de forma complementar, para determinar valores apropriados para p e q.

O gráfico ACF é útil para identificar séries temporais estacionárias, caso contrário, o ACF decairá para zero, de maneira relativamente rápida, enquanto o ACF de uma série não estacionária diminui lentamente.

Usualmente, uma série é considerada *Ruído Branco* se 95% dos valores da ACF estiverem dentro de $\frac{\pm 2}{\sqrt{T}}$, em que T é o comprimento da série temporal. É comum traçar esses limites em um gráfico do ACF (linhas tracejadas em azul/vermelho). Se um ou mais valores estiverem fora dos limites delimitados, ou mais de 5%, então a série provavelmente não é definida como ruído branco.

O alisamento exponencial e os modelos $ARIMA$ são as duas abordagens mais utilizadas para a previsão de séries temporais e fornecem metodologias complementares ao problema. Os modelos de suavização exponencial são baseados em uma descrição da tendência e sazonalidade nos dados (Hyndman e Athanasopoulos, 2021).

De acordo com Morettin e Toloi (2006) os modelos $ARIMA$ são úteis para descrever o comportamento de séries em que os erros observados são autocorrelacionados e influenciam na evolução do processo. Existem três classes de processos que podem ser descritos pelo modelo $ARIMA$:

1. *Processos lineares estacionários*, passíveis de representação na forma:

$$Y_t - \mu = \epsilon_t + \psi_1\epsilon_{t-1} + \psi_2\epsilon_{t-2} + ... = \sum_{k=0}^{\infty} \psi_k\epsilon_{t-k}, \psi_0 = 1, \tag{2.2}$$

tal que ϵ_t é um ruído branco, $\mu = E(Y_t) = \epsilon_t, \psi_1, \psi_2, ...$ é uma sequência de parâmetros, tal que $\sum_{k=0}^{\infty} \psi_k^2 < \infty$.

Existem três casos particulares do modelo 2.2, que são:

(a) Processo auto-regressivo de ordem $p = AR(P)$;

(b) Processo de médias móveis de ordem $q = MA(q)$;

(c) Processo auto-regressivo e de médias móveis de ordens p e q: $ARMA(p, q)$.

2. *Processos Lineares não-estacionários homogêneos.* Constituem uma generalização dos processos lineares estacionários que supõem que o mecanismo gerador da série produz erros autocorrelacionados e que as séries sejam não estacionárias em nível e/ou em inclinação. Estas séries podem tornar-se estacionárias por meio de um número finito (geralmente um ou dois) de diferenças (Morettin e Toloi, 2006);

3. *Processos de Memória Longa.* São processos estacionários que possuem uma função de autocorrelação com decaimento muito lento (hiperbólico) e cuja análise necessitará de uma diferença fracionária $(0 < d < 0,5)$(Morettin e Toloi, 2006).

Morettin e Toloi (2006) mencionam também que estes processos são descritos de maneira adequada pelos chamados modelos autorregressivos integrados e de médias móveis de ordem p, d e q : $ARIMA(p, d, q)$, que podem ser generalizados pela inclusão de um operador sazonal.

De acordo com Hyndman e Athanasopoulos (2021), os modelos $ARIMA$ também são capazes de modelar uma ampla gama de dados sazonais. Portanto, um modelo $ARIMA$ sazonal é caracterizado por incluir termos sazonais adicionais aos modelos $ARIMA$, ficando definidos como:

$$ARIMA(p, d, q)(P, D, Q)_m, \tag{2.3}$$

sendo (p, d, q) a parte não sazonal do modelo, (P, D, Q) a parte sazonal do modelo e m representando o período sazonal.

Critérios de Seleção de Modelos

Os critérios de seleção de modelos são utilizados para indicar, a partir de diferentes fórmulas, qual o modelo mais adequado a se escolher. Para estudo desse trabalho, serão escolhidos três critérios de seleção que foram explicados por Hyndman e Athanasopoulos (2021).

O Critério de Informação de Akaike é um método intimamente relacionado, definimos como:

$$AIC = T^* \log\left(\frac{SSE}{T^*}\right) + 2(k + 2), \tag{2.4}$$

sendo T^* o número de observações usadas para estimação e k (preditores no modelo).

Diferentes pacotes de computador usam definições ligeiramente diferentes para o AIC, porém a ideia principal é avaliar o ajuste do modelo pela medida SSE, que é a Soma de Quadrado dos Resíduos, e com o número de parâmetros que precisam ser estimados.

O Critério mede a qualidade do modelo e o valor mínimo do AIC representa maior qualidade e/ou simplicidade para o modelo de previsão, sendo definido como o mais adequado.

O *Critério de Informação Corrigido* de *Akaike* para pequenos valores de T^*, o AIC tende a selecionar muitos preditores e, portanto, uma versão corrigida por viés do AIC foi desenvolvida:

$$AIC_c = AIC + \frac{2(k+2)(k+3)}{T^* - k - 3}. \tag{2.5}$$

Da mesma forma que AIC, o AIC_c deve ser minimizado.

O *Critério de Informação Bayesiana de Schwarz* também denominado como BIC, possui a seguinte fórmula:

$$BIC = T^* \log\left(\frac{SSE}{T^*}\right) + (k+2)\log(T^*). \tag{2.6}$$

Assim como no AIC, o BIC tem a intenção de dar o melhor modelo. Diferente do AIC, o BIC penaliza o número de parâmetros mais fortemente. Esse critério calcula a densidade a posteriori dos parâmetros do modelo ajustado e verifica sua função log-verossimilhança maximizada em relação ao número de parâmetros do modelo.

Passo a Passo para Ajustes de Modelos $ARIMA$

Quando é desejado realizar uma previsão através dos modelos $ARIMA/SARIMA$ com séries temporais, é recomendado a utilização desses passos (Hyndman e Athanasopoulos, 2021):

1. Realize gráficos e identifique quaisquer observações incomuns;
2. Caso haja necessidade, transforme os dados para estabilizar a variância;
3. Se a série não for estacionária, tome as primeiras diferenças dos dados até que a série seja estacionária;
4. Examine o modelo $ACF/PACF$: Um modelo $ARIMA(p, d, 0)$ ou $ARIMA(0, d, q)$ é apropriado?

5. Experimente os modelos escolhidos e use o AIC_c para identificar o melhor modelo;

6. Verifique os resíduos do seu modelo escolhido, plotando o ACF dos resíduos e realizando testes dos resíduos. Se eles não se parecerem com *ruído branco*, tente um modelo modificado;

7. Uma vez que os resíduos pareçam ruído branco, calcule as previsões.

2.2.2 Alisamento Exponencial

De acordo com Burba (2018), o Alisamento Exponencial é um dos métodos clássicos de previsão de maior sucesso. Em sua forma básica, pode ser denominado como:

$$\hat{Y}_{t+h|t} = \alpha Y_t + \alpha(1-\alpha)Y_{t-1} + \alpha(1-\alpha)^2 Y_{t-2}..., \tag{2.7}$$

com $0 < \alpha < 1$, sendo $Y_{t+h|t}$ o valor no tempo Y_{t+h} dado que o valor Y_t é conhecido.

As previsões são semelhantes a uma média ponderada das observações anteriores e os pesos correspondentes diminuem exponencialmente à medida que volta-se no tempo.

Método de Holt - Tendência Linear

Com o intuito de conseguir utilizar a aplicação de Alisamento Exponencial com tendência, foi desenvolvido por *Holt* o método de tendência linear, definido como (Hyndman e Athanasopoulos, 2021):

$$\text{Equação de Previsão: } \hat{Y}_{t+h|t} = \ell_t + hb_t, \tag{2.8}$$

$$\text{Equação de Nível: } \ell_t = \alpha Y_t + (1-\alpha)(\ell_{t-1} + b_{t-1}), \tag{2.9}$$

$$\text{Equação de Tendência: } b_t = \beta^*(\ell_t - \ell_{t-1}) + (1-\beta^*)b_{t-1}, \tag{2.10}$$

em que ℓ_t denota uma estimativa do nível da série no momento t, b_t denota uma estimativa da tendência (inclinação) da série no momento t, α é o parâmetro de suavização para o nível, $0 \leq \alpha \leq 1$ e β^* é o parâmetro de suavização da tendência, $0 \leq \beta^* \leq 1$.

A equação de nível mostra que ℓ_t é uma média ponderada da observação Y_t e a previsão de treinamento dá um passo à frente para o tempo t. A equação de tendência mostra que

b_t é uma média ponderada da tendência estimada no momento t.

A função de previsão não é mais plana, mas possui tendência. A previsão *h-step-ahead* é igual ao último nível estimado mais h vezes o último valor de tendência estimado. As previsões são uma função linear de h.

Método de Holt-Winters - Sazonalidade

Hyndman e Athanasopoulos (2021) relataram o método de *Holt* e *Winters* como a extensão do método de *Holt* para conseguir capturar também a sazonalidade. O método sazonal *Holt-Winters* aborda a equação de previsão juntamente com três equações de suavização (nível ℓ_t, tendência b_t e componente sazonal s_t, respectivamente α, β^* e γ). O m é denotado como período da sazonalidade de acordo com os dados utilizados. Então,

$$\hat{Y}_{t+h|t} = (\ell_t + hb_t)s_{t+h-m(k+1)}; \tag{2.11}$$

$$\ell_t = \alpha\frac{Y_t}{s_{t-m}} + (1-\alpha)(\ell_{t-1} + b_{t-1}); \tag{2.12}$$

$$b_t = \beta^*(\ell_t - \ell_{t-1}) + (1-\beta^*)b_{t-1}; \tag{2.13}$$

$$s_t = \gamma\frac{Y_t}{(\ell_{t-1} + b_{t-1})} + (1-\gamma)s_{t-m}. \tag{2.14}$$

Os termos que definem o nível e a tendência, seguem da mesma definição da Seção 2.2.2, enquanto que o termo do componente sazonal é definido como s_t com $\gamma = \gamma^*(1-\alpha)$. A restrição usual do parâmetro é $0 \leq \gamma^* \leq 1$, que se traduz em $0 \leq \gamma \leq 1-\alpha$.

Em caso de dúvidas sobre os parâmetros definidos acima, veja mais sobre o significado de cada símbolo matemático na Seção A.

Existem duas variações neste método que diferem na natureza do componente sazonal. O método aditivo é preferido quando as variações sazonais são aproximadamente constantes ao longo da série, enquanto o método multiplicativo é preferido quando as variações sazonais estão mudando proporcionalmente ao nível da série. Com o método multiplicativo, o componente sazonal é expresso em termos relativos (percentuais) e a série é ajustada sazonalmente dividindo-se pelo componente sazonal. Dentro de cada período, o componente sazonal será somado a aproximadamente m (para saber mais a respeito,

procure em Hyndman e Athanasopoulos (2021)).

2.3 *Ensemble Learning - XGBoost*

De acordo com Rocca (2019), um dos métodos de *Machine Learning* é o *Ensemble Learning*, o qual consiste em combinar um conjunto de modelos para fornecer melhores predições e serem mais precisos que um único modelo, fornecendo um ajuste mais robusto e com maior acurácia.

O conceito de *Ensemble Learning*, também chamado de aprendizado por agrupamento, se baseia na ideia de tentar reduzir o viés e/ou a variância a partir da combinação de diversos modelos de predição mais simples (chamados de *weak learners*), treinados para solucionar um mesmo problema, e produzir a partir desses um modelo agrupado mais complexo (*strong learners*), o qual fornece melhores resultados.

Nos métodos sequenciais a ideia é ajustar os modelos iterativamente de forma que o treino do modelo em um determinado passo dependa dos modelos ajustados nos passos anteriores. *Boosting* é uma das abordagens mais conhecidas neste caso e produz um modelo *ensemble* que geralmente apresenta um menor viés do que os modelos individuais que o compõe (Rocca, 2019).

Para maximizar o desempenho do preditor final, o *Boosting* treina iterativamente novos modelos sempre com base nas observações que os ajustes anteriores tiveram mais dificuldade, tornando a predição mais resistente a viés. Em sequência, atualiza-se o modelo para dar prioridade as predições com maior acurácia nas observações da base de teste.

Normalmente, é escolhida a árvore de decisão de baixa profundidade para fazer parte do preditor final, pois o ideal para os métodos *Boosting* é escolher como base para agregação um modelo simples com alto viés e baixa variância.

No geral, para realizar a técnica *Boosting* os passos são:

- Encaixar um *weak learner*, combinação de diversos modelos de predição mais simples;
- Agregar ao modelo *Ensemble*;
- Atualizar o conjunto de dados de treinamento;
- Identificar os pontos fortes e fracos do modelo atual de *Ensemble* para encaixar o próximo modelo base;

- Repetir o processo novamente.

Para entender melhor esse processo, verifique a Figura 2.1:

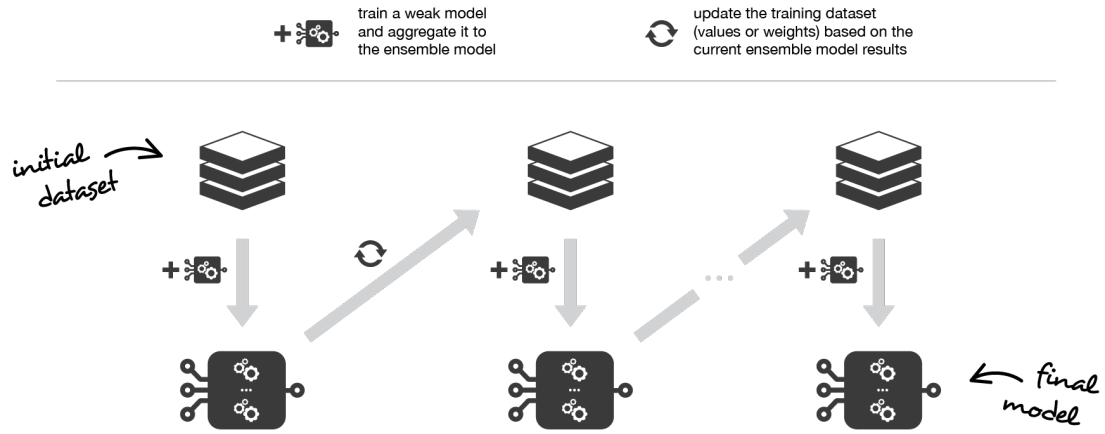

Figura 2.1: Ilustração do método *Boosting* (Rocca, 2019).

A maneira como é realizado a sequência e o modo como os modelos são agregados consiste na definição dos diferentes métodos de *Boosting*, em que cada qual possui um formato diferente de estruturar o modelo, entretanto possuem o mesmo intuito teórico (Neves, 2019).

Para compreender melhor uma das técnicas *Boosting*, será abordada a metodologia *XGBoost* com mais detalhes e explicações em Medeiros (2021).

XGBoost é o nome de uma biblioteca que implementa um modelo *ensemble* de classificação e regressão, baseado em árvores de decisão agregadas pelo *boosting*.

A árvore de decisão é um modelo voltado para classificação ou regressão, baseado na execução de sucessivas partições binárias de uma amostra, buscando a constituição de subamostras internamente homogêneas. Cada subamostra particionada recebe o nome de nó e cada resultado final identificado recebe o nome de folha.

A construção de uma árvore de decisão é feita com a criação de uma árvore grande e complexa e sua posterior poda, visando evitar seu sobreajuste ao conjunto de dados utilizado.

O *XGBoost* é um *ensemble* que utiliza o *boosting* em sua construção, uma vez que uma determinada árvore inicialmente construída vai sendo sequencialmente aprimorada por meio do *boosting*, a partir de novas árvores construídas pautadas na lógica de uma Floresta Aleatória (Medeiros, 2021).

Capítulo 3

Análises e Resultados

3.1 Banco de dados

O conjunto de dados analisado contém datas e valores de empréstimo e de provisões ativas, encerradas e totais de uma determinada empresa de consórcio de veículos e é composto por 55 observações, ordenadas por mês e ano, entre o período de Janeiro de 2017 até Julho de 2021 e contém cinco variáveis, que são:

1. **DtBase** - são as datas compostas por meses e anos dos valores totais de provisões e empréstimos realizados naquele mês e ano específico;

2. **Provisao_Ativos** - é o total em reais da provisão ativa da empresa de consórcio de veículos em um determinado mês e ano dos grupos que estão ativos no momento;

3. **Provisao_Encerrados** - é o total em reais da provisão que já foi encerrada pela empresa de consórcio de veículos em um respectivo mês e ano, ou seja, grupos que foram encerrados;

4. **Provisao_Total** - é o total em reais da provisão total da empresa de consórcio de veículos, levando em consideração as duas provisões anteriores (ativas e encerradas), naquele mês e ano específico;

5. **Emprestimo** - é o total em reais do empréstimo realizado pela empresa de consórcio de veículos para suprir as provisões em um determinado mês e ano. O mesmo é realizado pelo setor banco da empresa e auxilia no cálculo dos grupos encerrados.

Tabela 3.1: Dados de empréstimo e provisões ativas, encerradas e totais de uma empresa de consórcio de veículos.

DtBase	**Provisao_Ativos**	**Provisao_Encerrados**	**Provisao_Total**	**Emprestimo**
201701	26099047	28245487	54344534	44722761
201702	26283606	28866341	55149947	45964986
201703	27443346	28900357	56343703	45710932
⋮	⋮	⋮	⋮	⋮
202105	22321053	38925814	61246868	69460979
202106	23105518	28677323	51782842	69427694
202107	21885163	30825215	52710378	57534332

Para a realização das análises desta primeira parte do trabalho, não foram necessárias as variáveis **Provisao_Ativos** e **Provisao_Encerrados**, pois a **Provisao_Total** engloba esses outros tipos de provisões.

Todos os métodos descritos no Capítulo 2 foram realizados no *software R* pela plataforma *RStudio* explicada por Carlini (2021). Com os resultados obtidos, foi realizada toda análise necessária para encontrar a previsão com um modelo adequado para séries temporais.

A variável **DtBase** contém valores mensais em um período de quatro anos e meio, ou seja, o equivalente a 55 meses consecutivos. Essa informação é importante para análise das séries que serão estudadas em relação ao tempo.

A princípio, o objetivo é estimar a provisão total necessária para os meses consecutivos, além de analisar a informação fornecida para prever a quantidade ideal de dinheiro nos próximos meses para um bom funcionamento e organização da empresa. A variável resposta analisada dos dados que foram apresentados é a **Provisao_Total**.

3.2 Modelagem e Previsão da *Provisao_Total*

3.2.1 Análise Descritiva

A Figura 3.1 apresenta a série temporal da provisão total da empresa em um período de 55 meses consecutivos, quatro anos e seis meses:

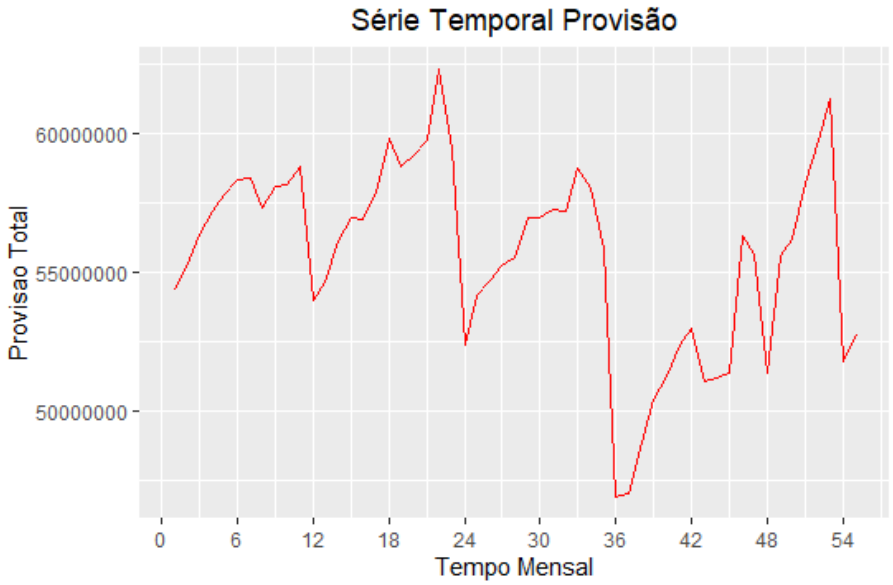

Figura 3.1: Série temporal da provisão total da empresa de consórcio de veículos.

Identifica-se um comportamento característico a cada período de 12 meses, uma sazonalidade (definido na Seção 2.2.1). Observa-se também que os valores da variável **Provisao_Total** nos primeiros 24 meses (2017 e 2018) foram semelhantes, exceto pelo seu aumento no final do segundo ano. Porém, no final do terceiro ano (2019), houve um decaimento dos valores da variável e nos anos consecutivos (2020 e 2021), os valores tiveram alguns picos altos ou baixos, mas somente entre os últimos meses da análise (2021) foram atingidos os maiores valores.

Como os dados são bem recentes, a queda no final do ano de 2019 e a diminuição da série no ano de 2020 pode ter ocorrido devido à Pandemia da COVID-19 que ocasionou uma redução nos gastos da população no geral, reduzindo assim a provisão total necessária para bom funcionamento da empresa.

Algumas medidas resumo são mostradas na tabela 3.2, juntamente com os gráficos (Figura 3.2).

Tabela 3.2: Medidas resumo da série **Provisao_Total** da empresa de consórcio.

Mínimo	**1º Quartil**	**Mediana**	**Média**	**3º Quartil**	**Máximo**
46882790	53495662	56315177	55659862	58121512	62333131

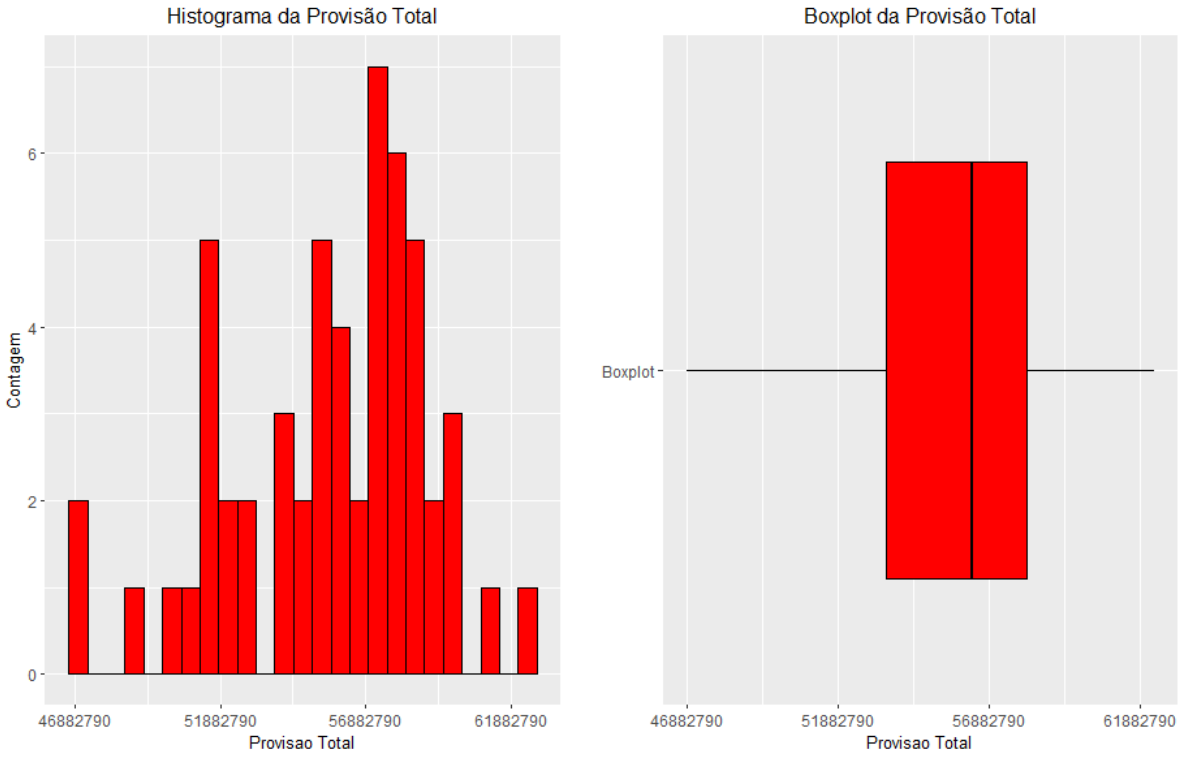

Figura 3.2: Histograma e *Boxplot* da variável **Provisao_Total** da empresa de consórcio.

Nota-se que a variável em estudo da empresa de consórcio se encontra entre um intervalo de mais de 46 milhões de reais até aproximadamente 63 milhões. Não existem pontos *outliers* presentes. A média e a mediana do valor da série são próximas, se diferenciando por menos de um milhão de reais, porém isso não significa que a série da provisão total realmente apresente uma distribuição normal. Verifique que as observações estão localizadas mais à direita do gráfico, notando assim uma assimetria à esquerda.

Para obter melhores resultados adiante, seria interessante que a série **Provisao_Total** apresentasse um comportamento de distribuição normal em suas observações, entretanto foi escolhido trabalhar com as variáveis em seu formato original com o intuito de que todos possam entender melhor os passos realizados até a previsão dos próximos meses.

Variância Constante

Para identificar se a variável **Provisao_Total** tem variância constante, dado que não é possível notar com clareza somente observando a Figura 3.1, será realizado o gráfico de Amplitude e Média. Veja a Figura 3.3:

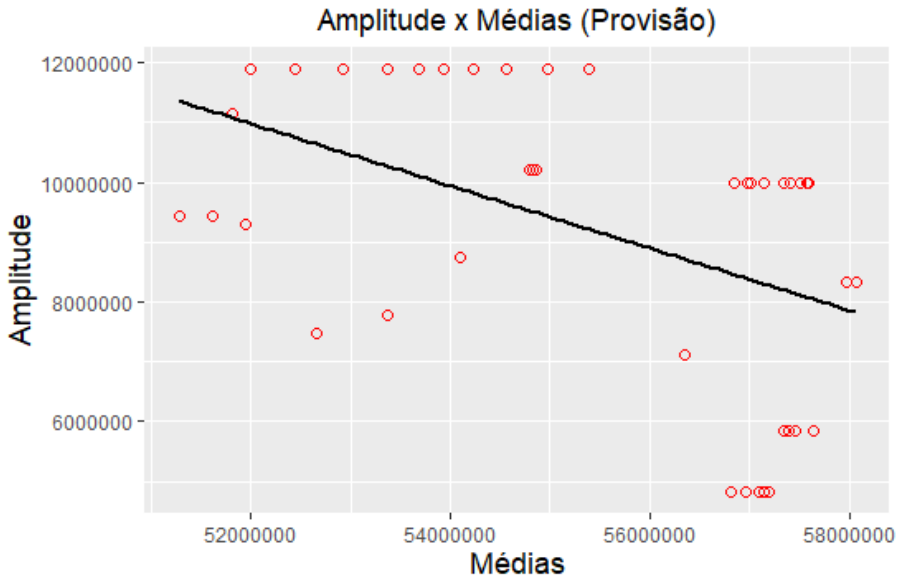

Figura 3.3: Gráfico de Amplitude e Média da variável em estudo da empresa de consórcio.

Verifique que a média não varia linearmente com a amplitude. As observações da série estão bem dispersas e não acompanham o comportamento da reta constante com uma inclinação decrescente, então não existe um efeito de variância na série de provisão total. Algumas transformações na série **Provisao_Total** também poderiam auxiliar para esse caso de variância constante, porque assim a provisão total se ajustaria caso a variância tivesse efeito.

Sazonalidade

Para apresentar a sazonalidade da série **Provisao_Total** é necessário recordar as definições de Hyndman e Athanasopoulos (2021) e Nielsen (2019) e visualizar os gráficos ACF e $PACF$ (Seção 2.2.1). Observe a Figura 3.4:

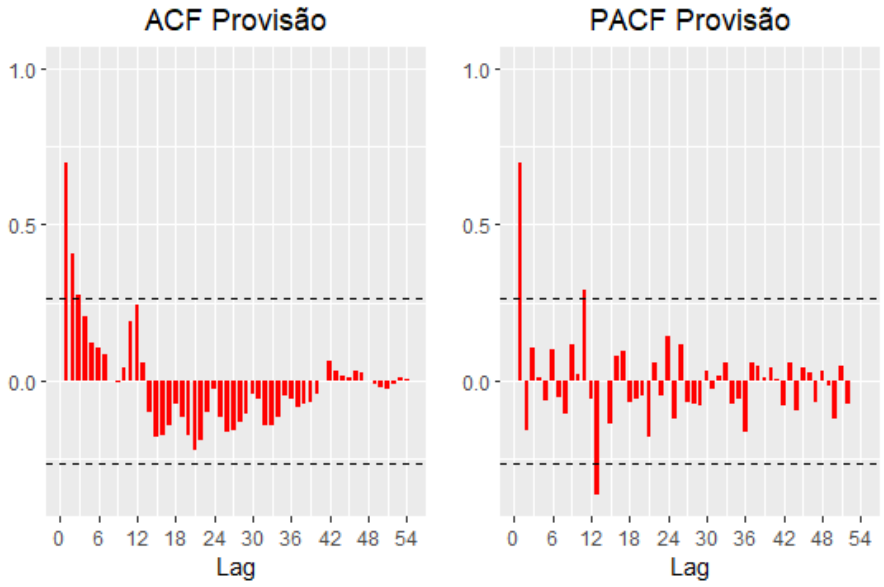

Figura 3.4: Gráfico ACF e $PACF$ da provisão total da empresa de consórcio.

Quando observa-se ambos os gráficos, o valor do PAC do *lag* 12, além da banda de confianca, sugere sazonalidade, pois se sobressaiu em relação ao limite estabelecido pelo gráfico. Logo próximo ao *lag* 12, existem dois pontos que ultrapassam o limite delimitado, então a série da provisão total apresenta sazonalidade de ordem 12.

Estacionariedade

Para avaliar a Estacionariedade definida anteriormente de Hyndman e Athanasopoulos (2021), será apresentado um Teste de Hipóteses (*Augmented Dickey-Fuller Test*).

Seja $\{Y_t\}_{t\in\mathbb{Z}}$ um processo univariado autorregressivo de ordem 1 ($AR(1)$), definido como:

$$Y_t = \phi Y_{t-1} + \epsilon_t, \tag{3.1}$$

sendo $\{\epsilon_t\}_{t\in\mathbb{Z}}$ um processo ruído branco com $E(\epsilon_t) = 0$ e $\sigma^2 > 0$ (considerando um nível de significância com $\alpha = 5\%$):

H_0: $\phi = 1$, o processo $\{Y_t\}_{t\in\mathbb{Z}}$ não é estacionário e conhecido como passeio aleatório;

H_1: $|\phi| < 1$, o processo $\{Y_t\}_{t\in\mathbb{Z}}$ é estacionário.

Como o nível de significância de $\alpha = 5\%$ e o valor-$p = 0,2824$, isto é, valor-$p > 0,05$, logo o H_0 não é rejeitado, indicando assim que os dados não são estacionários. Com essa afirmação, o passo três da Seção 2.2.1 precisará ser realizado.

Mais detalhes a respeito desse teste realizado se encontra em Pereira (2010).

Diferença

Como mencionado anteriormente, com o intuito de preparar as observações da variável **Provisao_Total** para elaboração do modelo, será feito a parte de diferenças para concluir o passo três de uma previsão adequada com o modelo $ARIMA$ da Seção 2.2.1.

Primeiramente, verifique a Figura 3.5 e perceba a série da variável estudada, utilizando a diferença de tamanho 1.

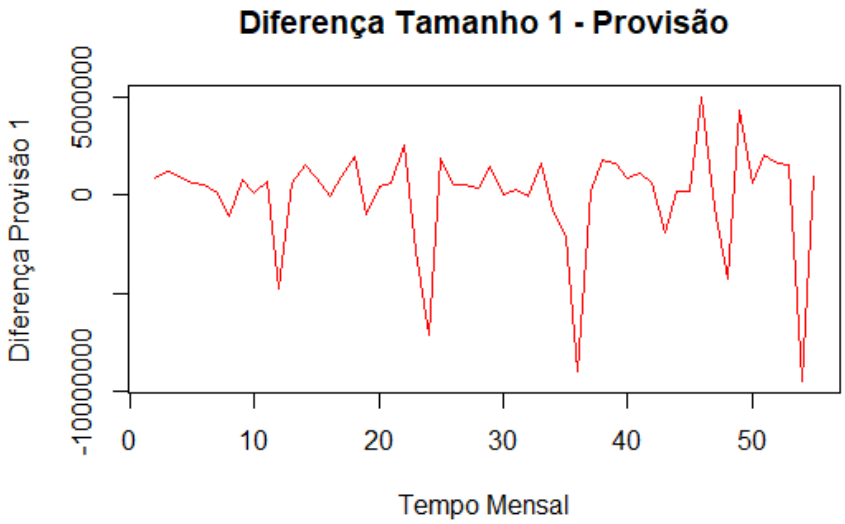

Figura 3.5: Gráfico da diferença de tamanho 1 da série.

Nota-se com a Figura 3.5 um comportamento característico dentre o período comentado acima de 12 meses. Além disso, já é possível identificar estacionariedade na série da provisão total e para comprovar isso, foi realizado novamente o Teste de Estacionariedade (*Augmented Dickey-Fuller Test*) com valor-$p = 0,0136$, (valor-$p \leq 0.05$), então a série **Provisao_Total** é estacionária.

Para identificar essa sazonalidade, veja a Figura 3.6 que irá destacar o período definido como 12 meses.

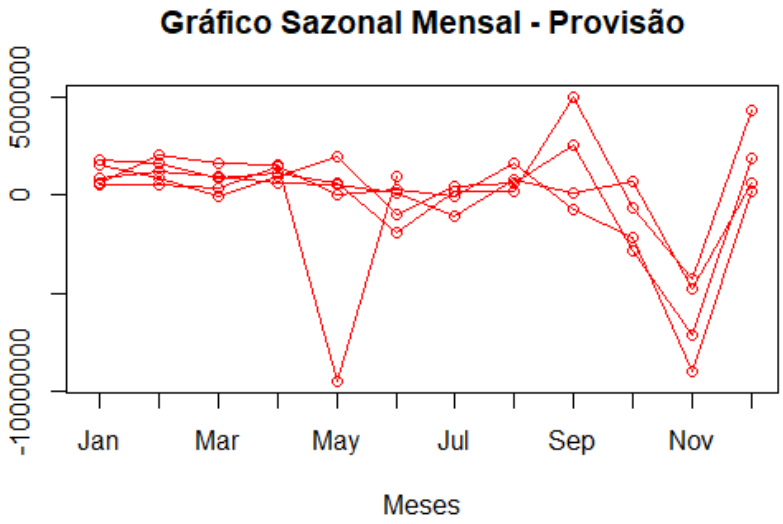

Figura 3.6: Gráfico Sazonal mensal contendo as informações da provisão total.

Observe que houve uma queda no mês de Maio e outras no mês de Novembro. Em Outubro, a variável em estudo começou a apresentar um valor menor em relação as demais e a queda acaba ocorrendo no mês de Novembro. Nos meses de Setembro e Dezembro, há um valor maior que os outros meses.

Novamente, a Pandemia da doença *COVID-19* pode ter ocasionado essa queda mais brusca no mês de Maio de 2020, comparado com os demais, é o mês que mais chama atenção.

Para verificar qual seria um modelo $ARMA$ sugerido para a série de Diferença da **Provisao_Total** analisou-se os gráficos ACF e $PACF$ da Figura 3.7, porém agora com a presença da diferença de tamanho 1.

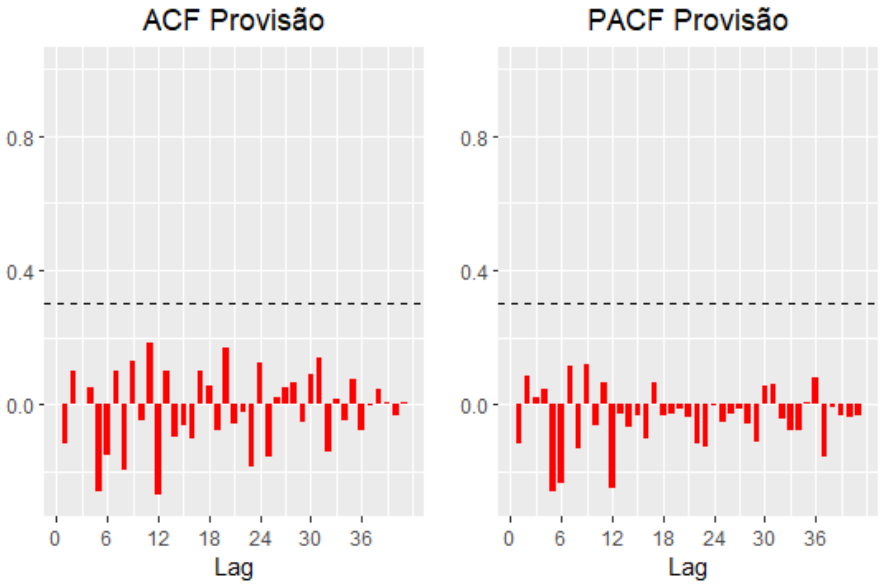

Figura 3.7: Gráfico ACF e $PACF$ da série **Provisao_Total**.

Observe que não há valor significativo de ACF ou $PACF$, para qualquer *lag*. Portanto, o modelo $ARMA$ sugerido para **Provisao_Total** é $ARIMA(0,1,0)$.

3.2.2 $ARIMA$

Como o modelo aplicado será o $SARIMA$, foram elaborados testes com dois modelos para a série, em um caso utilizando a ordem do $ARIMA$ selecionado pelo *RStudio* e no outro alterando p e q com a mesma sazonalidade.

Observe os dois modelos:

$$SARIMA(0,1,0)(0,1,1)_{12} \tag{3.2}$$

e

$$SARIMA(1,1,1)(0,1,1)_{12}. \tag{3.3}$$

Quando ambos os modelos são analisados separadamente, observa-se os seguintes resultados nas Tabelas 3.3 e 3.4:

Tabela 3.3: Resultados do Modelo 3.2.

	Estimate	SE	valor-t	valor-p
$sma1$	-0,5132	0,2246	-2,2805	0,0276

Note que na Tabela 3.3, existe apenas um coeficiente (Modelo 3.2) que é significativo, pois o seu valor-$p = 0,0276 \leq 0,05$. Logo, levando em consideração um nível de

significância com $\alpha = 5\%$, conclui-se que *sma1* é significativo.

Tabela 3.4: Resultados do Modelo 3.3.

	Estimate	SE	valor-t	valor-p
$ar1$	-0,6890	0,4619	-1,4919	0,1438
$ma1$	0,6012	0,4673	1,2864	0,2059
$sma1$	-0,5031	0,2272	-2,2141	0,0327

Entretanto, na Tabela 3.4 existem mais coeficientes e somente o mesmo coeficiente é significativo, com um valor-$p = 0,0327$, no restante deles o valor-$p > 0,05$, identificando assim que para esse caso eles não são significativos no modelo, podendo desconsiderar o mesmo. A análise terá prosseguimento para averiguação dessa pré conclusão.

Seguindo a explicação enunciada sobre seleção de modelos na Seção 2.2.1, serão verificados AIC, AIC_C e BIC na Tabela 3.5:

Tabela 3.5: Resultados de seleção do Modelo 3.2 e 3.3 da série de provisão total.

	AIC	AIC_C	BIC
Modelo 3.2	32,2305	32,2329	32,3133
Modelo 3.3	32,3155	32,3306	32,4810

Novamente, de acordo com os resultados de seleção de modelo, identifica-se que o Modelo 3.2 é o mais adequado para ajuste dos dados. Note que os valores só se diferenciam com as casas decimais, porém esse modelo já havia indicado ser melhor que o Modelo 3.3 anteriormente.

Por último, será apresentado uma série de gráficos dos resíduos de cada modelo e avaliando todos os pontos estudados: resíduos, resultados do modelo e da seleção, o modelo mais adequado será escolhido e levado em consideração para as próximas análises.

Veja a Figura 3.8 que irá mostrar um conjunto de gráficos dos resíduos do Modelo 3.2:

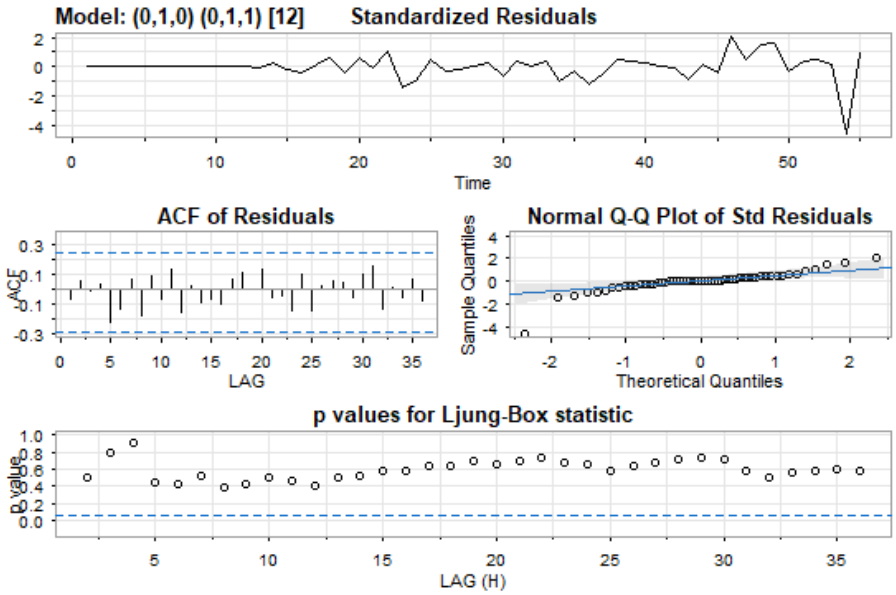

Figura 3.8: Resíduos do Modelo 3.2 da série **Provisao_Total**.

Observe que na Figura 3.8 os resíduos do gráfico ACF, apresentam um comportamento de ruído branco, pois estão dispersos dentro da banda de confiança. No gráfico Normal *qqplot*, verifica-se que os resíduos estão espalhados ao redor da reta, logo eles apresentam um comportamento de normalidade.

O teste de *Ljung-Box* identifica se existe autocorrelação na série se os resíduos do modelo não são autocorrelacionados, sendo esta a hipótese nula. Para saber mais informações a respeito de *Ljung-Box*, leia mais em Ferreira *et al.* (2015).

Na Figura 3.8, quando observados os p - valores do teste de *Ljung-Box* para *lags* até 36, note que os pontos estão acima do limite tracejado, logo não rejeita-se H_0, notando assim que os resíduos não são autocorrelacionados. Portanto, o modelo apresenta um comportamento ideal para a previsão em relação aos resíduos.

Note agora a Figura 3.9 que mostra um conjunto de gráficos dos resíduos do Modelo 3.3:

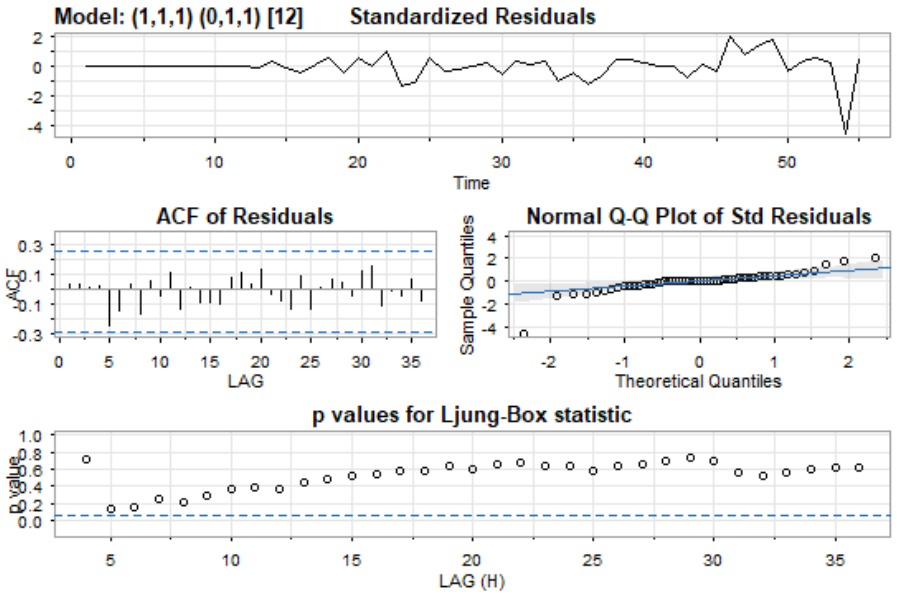

Figura 3.9: Resíduos do Modelo 3.3 da provisão total.

Analisando os gráficos dos resíduos na Figura 3.9, percebe-se que as análises são semelhantes ao da Figura 3.8, isto é, apresenta um comportamento adequado em relação aos resíduos. Um ponto em destaque, é que ao verificar os resíduos do teste de *Ljung-Box* com o valor-p, nota-se que as observações estão mais próximas do limite delimitado pelo gráfico do que o modelo anterior, porém mesmo com esse fato, identifica-se que a hipótese H_0 não é rejeitada, podendo salientar então que os resíduos não são autocorrelacionados (Ferreira *et al.*, 2015).

Diante de todas as análises para identificar o melhor modelo, concluí-se que o mais adequado para a variável **Provisao_Total** é o Modelo 3.2, indicado pelo *Software R-Studio*. Note que com essa escolha, os itens quatro, cinco e seis dos passos para a realização de uma previsão com modelo $ARIMA$ foram concluídos, porém será feito também uma detecção de *outliers* para verificar se os resíduos possuem algum valor dessa maneira e assim elaborar uma tratativa do modelo para cumprir os passos descritos na Seção 2.2.1.

Os *outliers* são caracterizados como observações discordantes ou atípicas em séries temporais e sua presença é responsável por afetar procedimentos da análise e prejudicar as estimativas dos parâmetros do modelo ajustado e as possíveis previsões (Almeida *et al.*, 2007). Com o intuito de detectar essas observações nos resíduos, serão verificadas se existem esses *outliers* denominados como AO - *outliers* aditivos que afeta a série apenas no momento T que aparece e o IO - *outliers* inovadores (intervenção) que prejudica as observações seguintes da sua análise. Para mais informações a respeito das diferentes

detectações de *outliers*, veja mais em Almeida *et al.* (2007).

Para a identificação de *outliers* presentes nos resíduos do Modelo 3.2 da variável em estudo, foram utilizadas duas funções do *Software R-Studio*: *detectAO* e *detectIO* (disponíveis no pacote TSA) respectivamente, mostra se existem *outliers* aditivos e/ou se existem *outliers* inovadores.

Verificando ambas as funções para a série, não houve *outliers* aditivos, entretanto foram destacadas três observações para os *outliers* de intervenção, que são as posições 24, 36 e 54 dos resíduos, isto é, a série pode ser afetada a partir da posição 24 das observações. Observe como fica o coeficiente do Modelo 3.2 na Tabela 3.6, quando esses *outliers* são tratados:

Tabela 3.6: Coeficientes do Modelo 3.2 com a tratativa dos *outliers* para a série de provisão total.

	sma1	**IO24**	**IO36**	**IO54**
	-0.9501	-2386383	-4379332	-10243624
s.e.	0.1782	1450386	1389939	1251195

Note agora os coeficientes do Modelo 3.2 juntamente com a tratativa das observações *outliers*, esses coeficientes são valores negativos que variam de dois até aproximadamente dez milhões. Com os *outliers* tratados, observe novamente o comportamento dos resíduos na Figura 3.10, para verificar se eles estão prontos para a previsão:

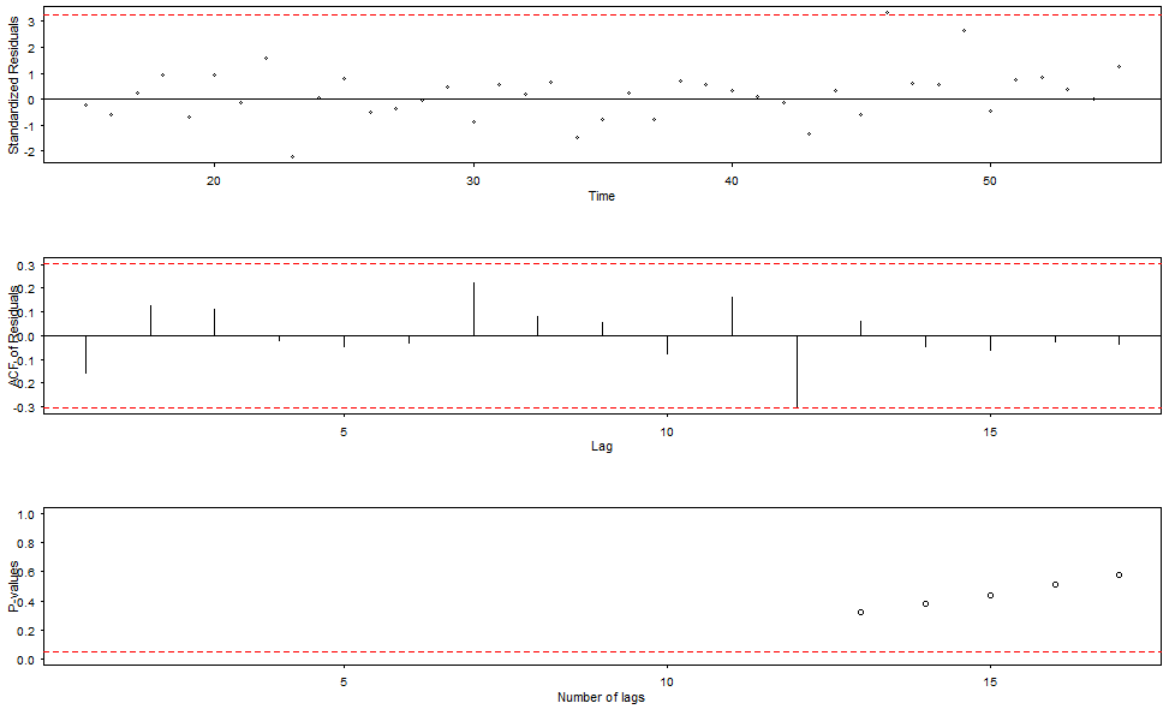

Figura 3.10: Resíduos do Modelo 3.2 da série **Provisao_Total** com a tratativa dos *outliers*.

Na Figura 3.10, percebe-se que os resíduos estão seguindo um comportamento de ruído branco para a provisão total. No primeiro e segundo gráfico verifica-se que os dados estão dispersos ao redor da reta 0 e nenhum ultrapassa as linhas delimitadas, isto é, prova-se que as observações são um ruído branco. Por fim, observa-se que os pontos do teste de *Ljung-Box* com o valor-p estão acima da reta delimitada, identificando assim que a hipótese nula não é rejeitada, isto é, os resíduos não são autocorrelacionados (Ferreira *et al.*, 2015), portanto os resíduos possuem um comportamento adequado para uma previsão.

Previsão

Agora que o modelo e os resíduos estão adequados será realizada a previsão do modelo $SARIMA(0,1,0)(0,1,1)_{12}$ escolhido (Modelo 3.2 com a tratativa dos *outliers*) que é o último item para os passos de uma previsão. Com o intuito de conseguir prever qual serão os valores da variável **Provisao_Total** para os próximos 12 meses de previsão em relação aos valores obtidos, verifique na Tabela 3.7 os valores da variável em estudo acompanhados de seus intervalos de confiança:

Tabela 3.7: Valores de previsão com 12 meses da série com limites inferiores e superiores do intervalo de confiança.

Ano/Mês	**Limite Inferior**	**Provisão Total Prevista**	**Limite Superior**
2021/08	49303790	52673999	56044208
2021/09	48599201	53365396	58131591
2021/10	49865061	55702434	61539807
2021/11	47691949	54432366	61172783
2021/12	41245552	48781567	56317582
2022/01	42305534	50560825	58816116
2022/02	42640969	51553209	60465449
2022/03	43505252	53029233	62553214
2022/04	43891929	53990661	64089394
2022/05	44592896	55235386	65877876
2022/06	43077908	54237692	65397477
2022/07	39995544	51649683	63303823

De acordo com a Tabela 3.7, a previsão para o próximo ano da série de provisão total são valores entre 48 e aproximadamente 56 milhões de reais. O menor valor da série **Provisao_Total** está previsto para Dezembro desse mesmo ano (2021) e o maior valor em Outubro de 2021. No ano consecutivo, os valores ultrapassam os 50 milhões, chegando até aproximadamente 55 milhões de reais. Portanto, percebe-se que com a previsão em 2022, a provisão total da empresa de consórcio tende a estabilizar próximo a concentração observada no *boxplot* da variável **Provisao_Total**, isto é, valores mais concentrados entre aproximadamente 53 e 58 milhões de reais, retirando assim todo impacto causado pela Pandemia.

Para conseguir identificar melhor essa previsão, comparada aos valores anteriores da variável em estudo, veja a Figura 3.11:

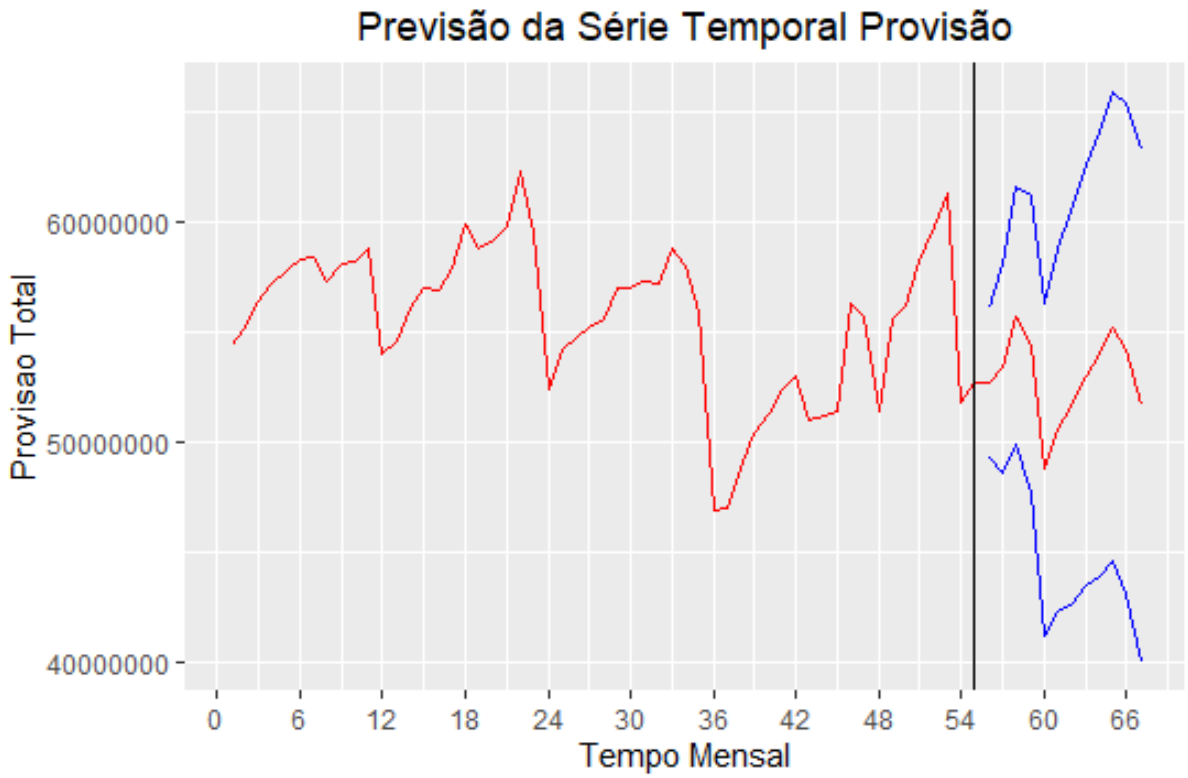

Figura 3.11: Previsão de 12 meses com limites inferiores e superiores da série em estudo.

Observe na Figura 3.11 para a série **Provisao_Total** que o intervalo de confiança é grande, podendo estar entre, aproximadamente, 39 e 66 milhões de reais. Mesmo se tratando desses intervalos, mas os limites inferiores chegam a ficar abaixo da observação 36.

Os códigos utilizados no *RStudio* se encontram na Seção B.

3.2.3 *Holt-Winters*

Como explicado na Seção 2.2.2, a metodologia de *Holt-Winters* é uma outra técnica para identificar a previsão da variável estudada. Com o intuito de utilizar outra aplicação, foi escolhido esse método que aborda os estudos de séries temporais e futuramente compará-lo com outras técnicas, principalmente a realizada com o modelo $ARIMA$.

Os passos para uma previsão adequada desse método é bem próximo ao utilizado de Hyndman e Athanasopoulos (2021), com algumas particularidades em relação ao modelo, porém com a apresentação de 12 meses de previsão e do gráfico. As análises descritivas da série serão reaproveitadas para estudo dessa metodologia.

O Método Multiplicativo de *Holt-Winters* aborda a equação de previsão em conjunto com três equações de suavização que envolve nível, tendência e componente sazonal (res-

pectivamente α, β^* e γ), realizando as Fórmulas 2.12, 2.13 e 2.14, chega-se nos seguintes resultados da Tabela 3.8:

Tabela 3.8: Resultados das Fórmulas 2.12, 2.13 e 2.14 de *Holt-Winters*.

Equação	**Resultado**
α	0,7218
β^*	0
γ	1

Observe na Tabela 3.8 que a equação β^* é zerada para a série de provisão total, neste caso, nota-se que para essa metodologia não foi considerada nenhuma tendência. O valor do componente sazonal é 1 e o do nível é 0,7218 e esses resultados serão utilizados para encontrar a Equação 2.11.

Veja agora a Figura 3.12 da série **Provisao_Total** em preto e a série de acordo com *Holt-Winters* em vermelho:

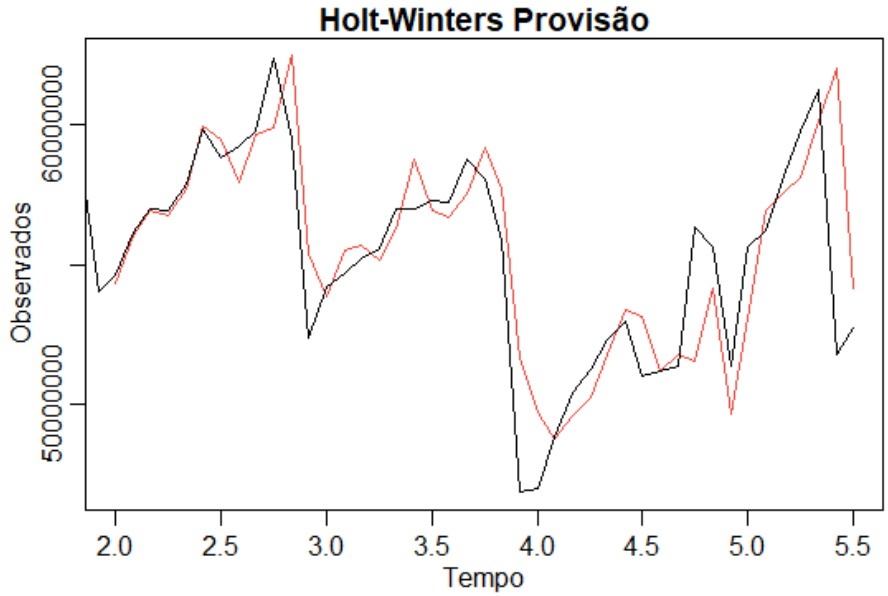

Figura 3.12: Provisão Total com a série realizada no método de *Holt-Winters* para verificação.

Ambas possuem um comportamento semelhante, acompanhando em momentos de picos ou quedas e pra esse caso, as observações não são as iniciais.

Utilizando o conceito de *Holt-Winters* foi realizada a previsão dos próximos 12 meses da variável em estudo. Verifique a Tabela 3.9:

Tabela 3.9: Valores de previsão com 12 meses da série pelo método de *Holt-Winters* com limites inferiores e superiores do intervalo de confiança.

Ano/Mês	**Limite Inferior**	**Provisão Total Prevista**	**Limite Superior**
2021/08	48848553	53302675	57756798
2021/09	48296461	53804339	59312216
2021/10	48846241	55310178	61774116
2021/11	46525013	53599768	60674522
2021/12	41041412	48297654	55553895
2022/01	42396385	50582382	58768379
2022/02	42696114	51613227	60530339
2022/03	43352935	53021278	62689620
2022/04	43143434	53373192	63602949
2022/05	43169867	53978723	64787579
2022/06	40927837	51808122	62688407
2022/07	42971926	53786915	64601905

Na Tabela 3.9 é possível identificar que, de acordo com a previsão realizada por essa metodologia, a série da provisão total nos próximos meses terá um valor entre aproximadamente 48 e 55 milhões, sendo a menor série **Provisao_Total** em Dezembro de 2021 e a maior em Outubro de 2021. O valor mais alto é praticamente a média da provisão total observada, logo esses resultados apresentaram um comportamento um pouco mais baixo em relação a variável **Provisao_Total**.

Os diferentes intervalos destacados se encontram entre aproximadamente 40 e 64 milhões. Ou seja, são intervalos bem grandes e ainda maiores do que os valores mínimo e máximo mostrados da variável em estudo.

Para entender como esses valores estão distribuídos na série, verifique a Figura 3.13:

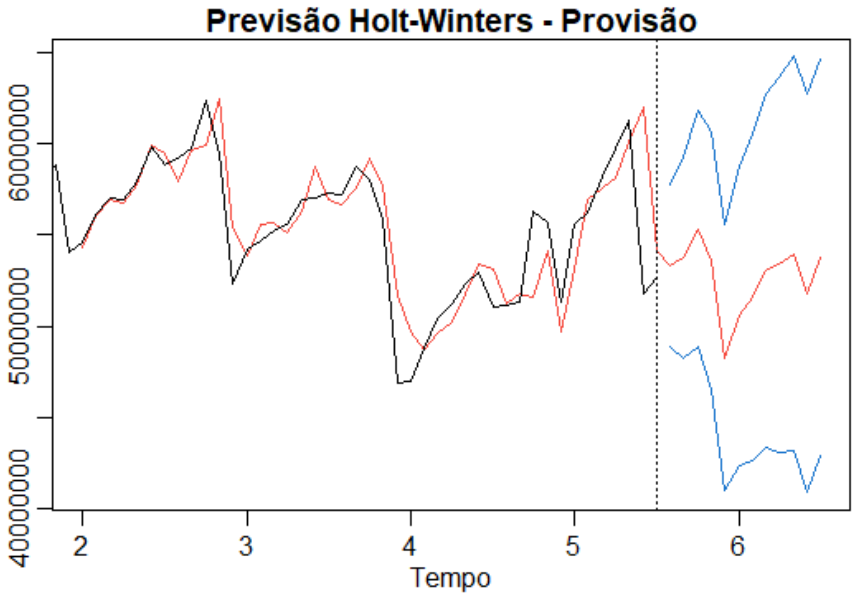

Figura 3.13: Gráfico contendo a série da provisão total unida ao seu planejamento de *Holt-Winters* para os próximos 12 meses.

Note com a Figura 3.13 que a série **Provisao_Total** contendo a previsão para o próximo ano, possui um comportamento mais baixo em relação as observações anteriores e possui uma queda no final do ano de 2021 (Dezembro) que já havia ocorrido nos dois anos anteriores. Os intervalos são bem grandes e capazes de abranger os momentos em que os valores da provisão total podem aumentar ou diminuir de uma vez.

3.2.4 *Extreme Gradient Boosting*

Como já foram realizadas as análises que envolvem a modelagem e a previsão para as metodologias *ARIMA* e *Alisamento Exponencial - Holt Winters*, será abordado neste momento a aplicação do modelo *Extreme Gradient Boosting*, chamado também de *XGBoost*, explicado na Seção 2.3.

Caracterizado por representar uma das várias técnicas de *Machine Learning*, esse modelo foi realizado com o intuito de comparação com outras metodologias aplicadas propriamente para séries temporais.

Seguindo os passos analisados em Alice (2020), primeiramente foi necessário preparar a base para aplicação da metodologia de *XGBoost*. A princípio, como essa análise será realizada somente para a variável que aborda provisão, então foram utilizadas do banco as variáveis: **DtBase** e **Provisao_Total**.

É válido destacar que a variável **DtBase** contém, respectivamente, o ano e o mês de cada uma das provisões. Como é necessário a utilização de covariáveis para realizar a metodologia de *XGBoost* e como é levado em consideração somente a data e a variável resposta por se tratar de uma série temporal, então a data será separada em duas variáveis, sendo elas definidas como *ano* e *mes* para cada uma das observações.

Dessa maneira, a variável **DtBase** será decomposta da seguinte maneira na Tabela 3.10:

Tabela 3.10: Exemplo da decomposição da variável **DtBase**.

DtBase	**ano**	**mes**
201701	2017	01
201702	2017	02
201703	2017	03
⋮	⋮	⋮
202205	2022	05
202206	2022	06
202207	2022	07

Com a Tabela 3.10 verifica-se alguns exemplos da decomposição da variável **DtBase**. Note que as últimas três variáveis foram adicionadas já considerando a previsão de 12 meses para frente, isto é, iniciando em Agosto de 2021 até Julho de 2022.

Considerando as 12 observações adicionadas, isto é, o *ano* e o *mes* de previsão do próximo ano e adicionando informação vazia para a variável **Provisao_Total**, então o

total da base passa a ser de 67 observações.

Como a maioria dos modelos realizados com as técnicas de *Machine Learning* necessitam da divisão do conjunto de dados em treinamento e teste, o mesmo foi feito para esse banco, contudo, por se tratar de séries temporais, existem algumas modificações.

Ao contrário de considerar uma porcentagem da divisão dos dados e defini-los como treino e teste, esses serão divididos em treinos e valores preditos, pois os últimos são aqueles que deseja-se encontrar sua respectiva previsão, ou seja, 12 meses para frente.

Dado que, para as outras metodologias utilizadas nas Seções 3.2.2 e 3.2.3 foram utilizadas a previsão equivalente ao próximo ano do conjunto de dados, o mesmo foi realizado para a metologia de *XGBoost*.

Portanto, a base foi dividida em duas partes definindo uma semente fixa, utilizando uma divisão de aproximadamente 85% para treino e 15% para valores preditos:

- **Treino:** Foram consideradas as primeiras 55 observações, idênticas ao conjunto de dados original e contém informações entre as datas de Janeiro de 2017 até Julho de 2021;

- **Preditos:** O banco de dados que compõem o conjunto dos preditos são as 12 últimas observações, isto é, da 56ª até a 67ª observação; são as datas em que seus valores serão previstos, de Agosto de 2021 até Julho de 2022.

Iniciando com a modelagem do *XGBoost*, foi realizado a preparação dos parâmetros de controle do treinamento que é a base que foi denominada como **Treino**, para isso foi utilizada a função pronta do *RStudio* chamada *trainControl* do pacote *caret* para controlar os nuances computacionais da base de treino.

Para essa função foram escolhidos e definidos na escolha desse modelo, os seguintes argumentos:

- *method:* o método escolhido para ser utilizado é o "cv" que significa *cross-validation*, validação cruzada, por se tratar de um estudo de séries temporais, é o mais adequado para a ocasião;

- *number:* é o número de iterações de reamostragem e foi utilizado cinco;

- *allowParallel:* pergunta se a função pode utilizar um *backend* paralelo carregado e disponível e a resposta é sim para esse modelo;

- *verboseIter:* pergunta se a função deve criar uma lógica para aparecer os códigos da base de treino e nesse caso, o interesse é o modelo em si, então a resposta é falsa;

- *returnData:* pergunta se é necessário salvar os dados e isso não é necessário dada a análise.

Os outros argumentos dessa função não foram mencionados, porque não foram utilizados no momento de reproduzir o *trainControl*.

Agora, com o intuito de mostrar os hiperparâmetros da validação cruzada, foi reproduzida a função do *RStudio* cujo nome é *expand.grid* que cria um conjunto de dados com as combinações possíveis e como o interesse são os hiperparâmetros, logo é criado uma lista com o que é necessário para o modelo. Nesse caso, a função também foi realizada para a base **Treino** e a lista definida dentro dessa função, contém as seguintes informações (Medeiros, 2021):

- *nrounds:* é o número de vezes e pra esse caso foi definido um vetor com 100 e 200;

- *max_depth:* é a profundidade considerada da árvore e nesse modelo foram escolhidas a profundidade de tamanho 10, 15 e 20;

- *colsample_bytree:* é a proporção de subamostra de colunas ao construir cada árvore, para esse modelo foi escolhido o 1;

- *eta:* a taxa escolhida como eta que caracteriza o *learning* - aprendizagem foi definido como 0,1;

- *gamma:* é a redução de perda mínima e no caso é definido como 0;

- *min_child_weight:* soma mínima de peso da instância, quando necessário, nesse caso foi considerado 1;

- *subsample:* é a proporção de subamostra das instâncias da base de treino, considerada também como 1.

Lembrando que esses valores escolhidos foram baseados na cronologia de criação de um modelo *XGBoost* para Séries Temporais de Alice (2020).

Quando os “tunning” são criados, o modelo já pode ser realizado, utilizando a função *train* do *RStudio* (Kuhn, 2008). Para essa aplicação em *XGBoost* os parâmetros usados nesse modelo que interliga o *Machine Learning* com Séries Temporais são:

- *provisao_estendida* . - A variável resposta com suas respectivas covariáveis, *ano* e *mes*, isto é, o modelo de interesse;
- *data* - São os dados utilizados de Treinamento, exceto a variável *Provisao_Total*;
- *method* - O método escolhido para essa análise é o “xgbTree” que verifica o desempenho e escolhe o melhor modelo com árvore e os “tunning” do *XGBoost*;
- *trControl* - Utiliza o que já foi feito anteriormente na função denominada como *trainControl*;
- *tuneGrid* - Utiliza também o que já foi feito anteriormente na função *expand.grid*, que foi denominada como *xgb_grid*;
- *nthread* - Controla o número de *thread* usados na construção do modelo, para construir árvores de forma paralela, e para esse modelo, foi escolhido somente uma ligação.

O modelo de *XGBoost* que possui o melhor *bestTune*, ou seja, a performance mais adequada para esse conjunto de dados é mostrado na Tabela 3.11:

Tabela 3.11: Resultados de Parâmetros e Performances do Modelo *XGBoost* para a variável *Provisao_Total*.

nrounds	*max_depth*	*eta*	*gamma*	*colsample_bytree*	*min_child_weight*	*subsample*
100	15	0,1	0	1	1	1

Como definidos nessa própria seção 3.2.4, os valores referentes a *eta, gamma, colsample_bytree, min_child_weight* e *subsample* continuam os mesmos, porém o melhor *nrounds* selecionado foi o 100, então esse é o número que encontra a melhor performance do modelo e a profundidade da árvore escolhida para o melhor modelo foi 15 que era a média entre as duas outras opções de profundidade.

Com o intuito de identificar os intervalos de confiança de acordo com o modelo de *XGboost* para a variável *Provisao_Total*, foi criada uma matriz vazia de 12 linhas e 100 colunas. Cada linha representa a previsão de 12 meses e as colunas indicam 100 observações em cada uma. Então, com os mesmos parâmetros do modelo, foram gerados 100 previsões *XGboost* para cada mês de previsão. Foram considerados os valores mínimos de cada mês como sendo o intervalo inferior e os valores máximos como sendo o superior. A média indica o valor original da previsão.

Observe na Tabela 3.12 a previsão para o próximo ano da provisão total, incluindo seus respectivos intervalos de confiança:

Tabela 3.12: Valores de previsão com 12 meses da série pelo método de *XGBoost* com limites inferiores e superiores definidos como intervalo de confiança.

Ano/Mês	**Limite Inferior**	**Provisão Total Prevista**	**Limite Superior**
2021/08	52658748	52666446	52710068
2021/09	52854060	52861758	52905380
2021/10	53246076	53253774	53297396
2021/11	53158068	53165766	53209388
2021/12	50474744	50482443	50526068
2022/01	55567524	55574479	55613888
2022/02	56208256	56217460	56217460
2022/03	58171384	58174794	58194116
2022/04	59690064	59701399	59765628
2022/05	61051196	61080373	61245712
2022/06	51769776	51771724	51782760
2022/07	52658748	52658748	52710068

Com a Tabela 3.12 verifica-se com a previsão de *XGBoost* que a série da provisão total no próximo ano terá valores entre aproximadamente 50 e 61 milhões, sendo o menor valor de **Provisao_Total** em Dezembro de 2021 e a maior em Maio de 2022. Nota-se que diferente das outras metodologias, ambos os limites estão mais próximos do valor real da variável em estudo.

Para visualizar esses valores de maneira gráfica, verifique a Figura 3.14:

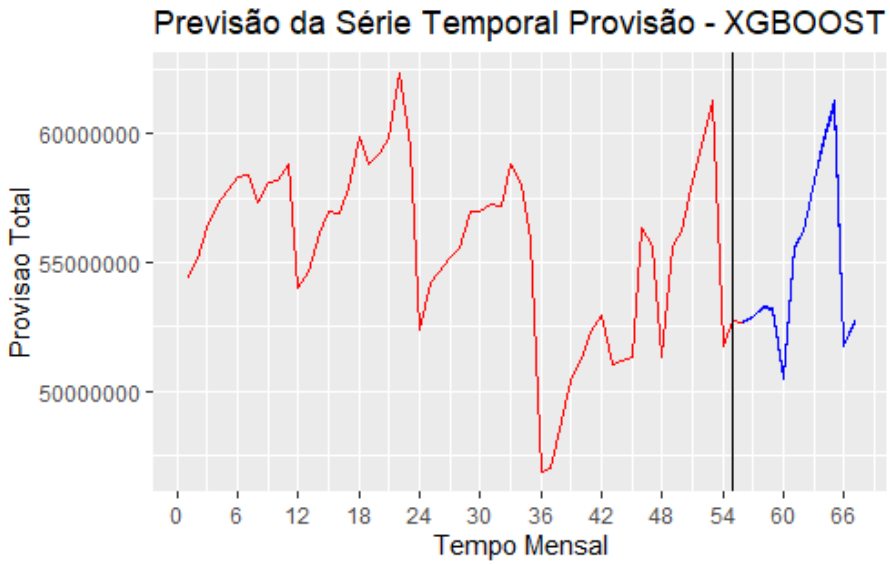

Figura 3.14: Previsão da série Provisão Total com a metodologia *XGBoost*.

Note na Figura 3.14 que, como os intervalos e os valores reais de previsão são bem próximos, não é possível distinguir na imagem as séries de ambos os limites e a da previsão. Porém, é possível ver o comportamento do próximo ano, inicialmente existe uma diminuição dos valores, que possui o menor valor em Dezembro e depois esses valores se elevam, com pico em Maio, finalizando em aproximadamente 52 milhões.

É interessante destacar que o comportamento do próximo ano está semelhante ao das últimas observações entre os meses 48 e 54. Isso pode ter ocorrido, devido ao tamanho do conjunto de dados analisados ser pequeno, fazendo com que o modelo *XGBoost* não tenha uma performance ideal quanto teria para um base maior de observações.

3.3 Correlação Cruzada entre as variáveis

Como as análises da variável **Provisao_Total** foram realizadas, nota-se o interesse de um complemento da pesquisa sobre a empresa de consórcio em verificar comportamentos, então foi pensado na possibilidade de prever o quanto de empréstimo a organização irá precisar nos próximos meses, baseado no valor das respectivas provisões.

Logo, com o intuito de analisar não somente uma, mas duas variáveis, primeiramente será verificado se elas possuem correlação cruzada para entender melhor o comportamento de ambas em conjunto.

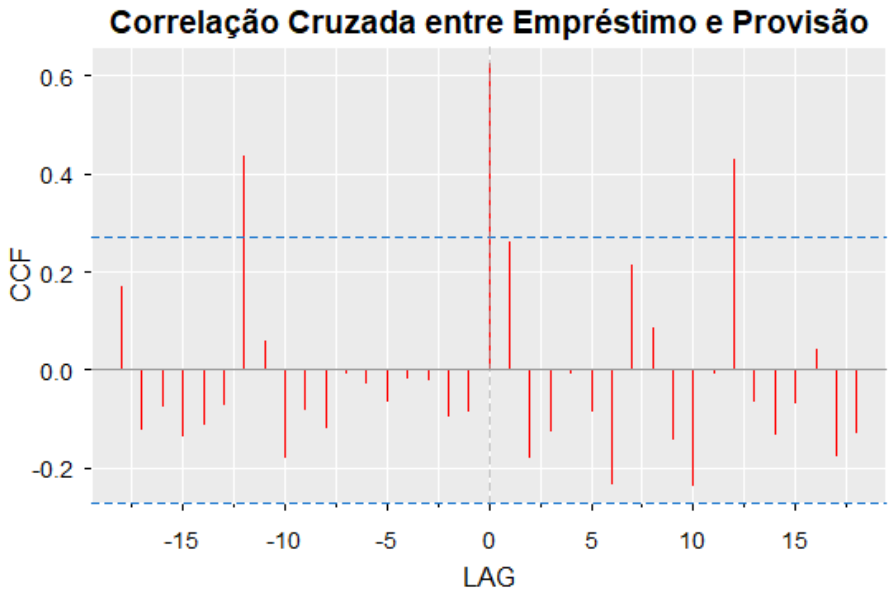

Figura 3.15: Correlação cruzada entre as variáveis **Emprestimo** e **Provisao_Total**.

A Figura 3.15 apresenta a correlação entre as séries de empréstimo e da provisão total, note que há um deslocamento demarcado entre os *lags* -15 e 15 e realiza-se uma comparação de maneira contemporânea e espontânea, ambas caminham juntas e possuem sazonalidade. Para calcular a correlação é realizado a fórmula $corr[X(t + lag), Y_t]$ e os pontos delimitados informam se existe ou não correlação entre ambas. Veja que há uma correlação no *lag* de tamanho 12 entre ambas.

Mediante essa análise, observe que existe a possibilidade de ser estudado a respeito das duas variáveis, então agora que as abordagens da variável **Provisao_Total** foram finalizadas, foi realizado todo procedimento novamente para a variável **Emprestimo** (com previsão dos meses consecutivos). As análises irão permanecer de formas separadas.

3.4 Modelagem e Previsão da Variável *Emprestimo*

3.4.1 Análise Descritiva

Essa variável foi escolhida com o interesse e a curiosidade a respeito da previsão do valor de empréstimo que será necessário para as necessidades da empresa de consórcio, podendo ser considerada como uma segunda variável resposta que torna-se dependente da provisão total. De acordo com essa escolha, primeiramente foram realizadas análises descritivas com o objetivo de conhecer e entender a série temporal da Seção 2.1.

Observe a Figura 3.16 que apresenta a série temporal da variável **Emprestimo** reali-

zado pela empresa em um período de 55 meses (quatro anos e seis meses):

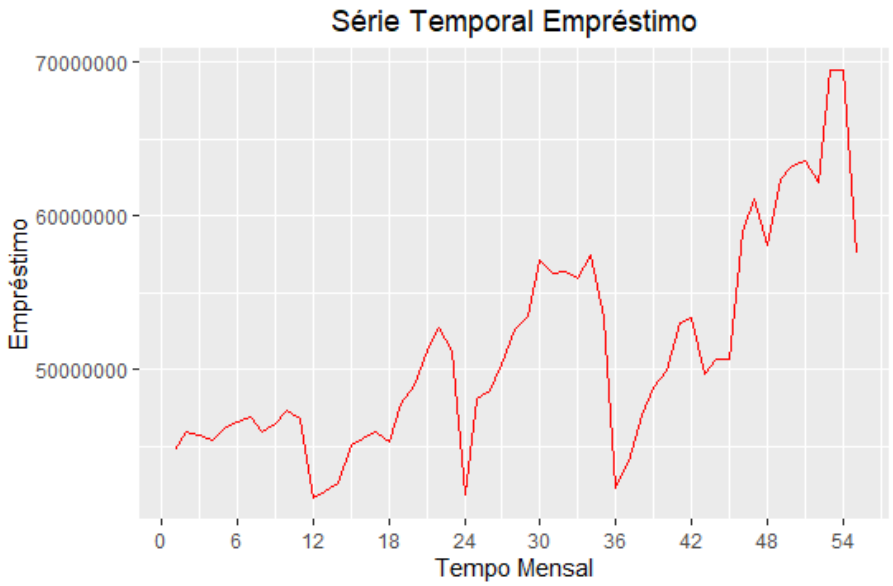

Figura 3.16: Série temporal da variável *Emprestimo* da empresa de consórcio de veículos.

Verifique na Figura 3.1 que existe um comportamento específico em cada período com duração de 12 meses, identificando assim uma sazonalidade (Seção 2.2.1) na série de tamanho 12. Note que os valores da série de empréstimo no primeiro ano (2017) foram os menores em relação aos demais, a partir desse ano a série **Emprestimo** passa a ficar cada vez maior, principalmente nos últimos dois anos de observações (2020 e 2021), decaindo apenas em cada fechamento de sazonalidade, isto é, nas observações 12, 24 e 36.

As observações analisadas pelo empréstimo são recentes, fazendo com que o aumento no valor da variável **Emprestimo** a partir do ano de 2019 (após o 24º mês) possa estar relacionado com o impacto causado pela Pandemia da COVID-19, assim como ocorreu com a provisão total. É possível que os clientes tiveram imprevistos com relação a doença e não conseguiram finalizar o consórcio, exigindo assim um valor maior para a empresa comparado aos anos anteriores.

Para compreender a variável em estudo num todo, verifique algumas das medidas resumo na Tabela 3.13 da série e gráficos (Figura 3.17) que mostram esses resultados.

Tabela 3.13: Medidas resumo da série de empréstimo da empresa de consórcio.

Mínimo	**1º Quartil**	**Mediana**	**Média**	**3º Quartil**	**Máximo**
41622279	45967540	49725092	51162757	56101631	69460979

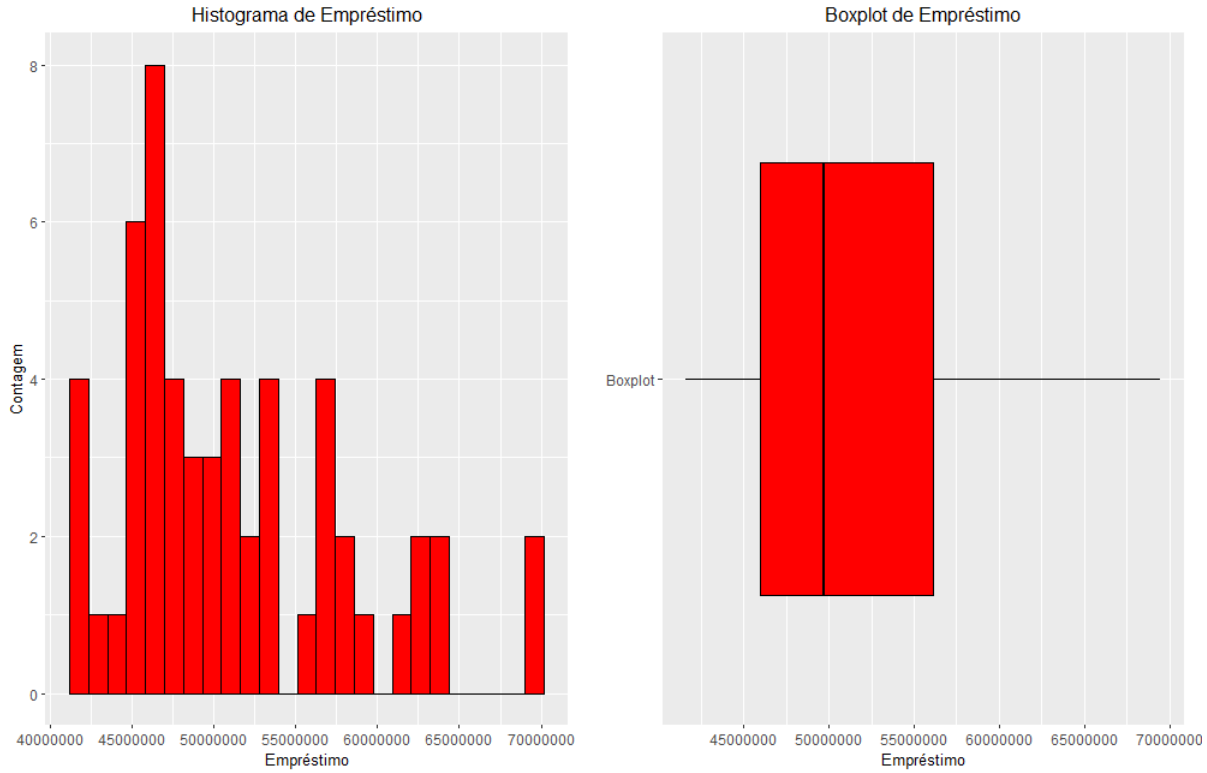

Figura 3.17: Histograma e *Boxplot* da série **Emprestimo** realizado pela empresa de consórcio.

Observe a Tabela 3.13 e a Figura 3.17 que possui dois gráficos respectivamente, Histograma e *Boxplot*, o comportamento do empréstimo da empresa de consórcio que atua entre um intervalo de mais de 41 milhões de reais até aproximadamente 69 milhões. Não existem pontos *outliers* observados e a variabilidade está presente entre aproximadamente 46 e 56 milhões de reais, ou seja, possui um intervalo de mais de dez milhões de reais, podendo ser considerada um pouco alta. A média e a mediana da variável **Emprestimo** se diferenciam em um pouco menos de 2 milhões entre uma e outra, porém percebe-se uma distância em relação ao meio do gráfico de *Boxplot*. Observe que as observações estão mais localizadas à esquerda da Figura 3.17, notando assim uma assimetria à direita.

Lembrando que foi escolhido trabalhar com as variáveis em seu formato original para que todos possam entender melhor os passos realizados até a previsão dos próximos meses.

Note que, com a análise anterior realizada a respeito da variável em estudo, foram feitos

os passos um e dois da previsão de um modelo $ARIMA$, como definido em Hyndman e Athanasopoulos (2021).

Variância Constante

Com o intuito de verificar se a série possui variância constante, pois não é possível identificar facilmente na Figura 3.16, foi feito o gráfico de Amplitude e Média. Veja a Figura 3.18:

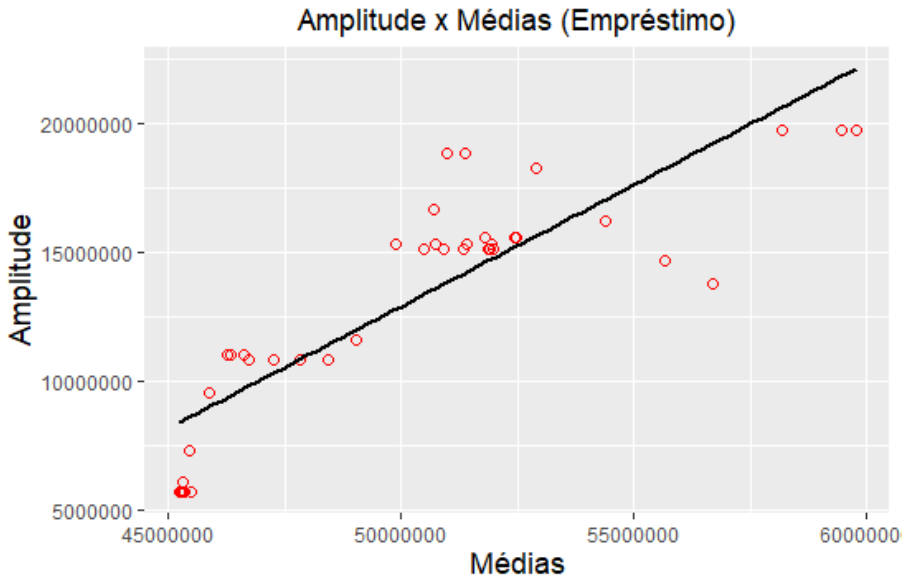

Figura 3.18: Gráfico de Amplitude e Média da série de empréstimo da empresa de consórcio.

Note que a média aparenta variar linearmente com a amplitude, porém veja que algumas observações não estão sobrepostas na reta. Os pontos estão próximos da reta e em alguns casos as médias são maiores ou menores em relação a linearidade da amplitude. O comportamento da reta é constante e possui uma inclinação crescente, como as observações não estão exatamente sobre a reta delimitada, logo não existe um efeito de variância na série **Emprestimo**. A transformação do empréstimo poderia auxiliar nesse caso de variância constante.

Sazonalidade

Veja a Figura 3.19 para verificar a sazonalidade da variável **Emprestimo**:

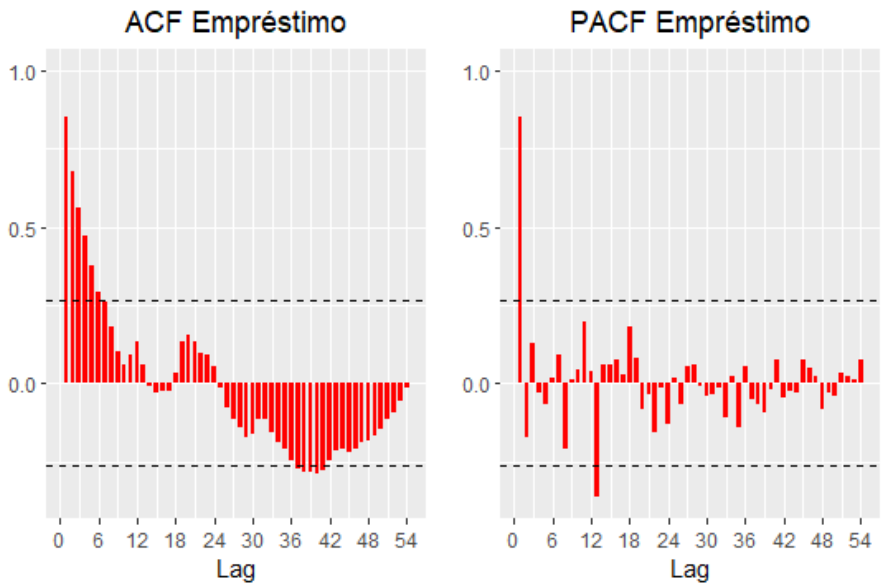

Figura 3.19: Gráfico ACF e $PACF$ da variável em estudo da empresa de consórcio.

Veja que em ambos os gráficos existe um padrão característico a cada 12 meses para afirmar que tem de fato uma sazonalidade na série. O *lag* 12 ultrapassa o limite estabelecido pelo gráfico, identificando assim que a série do empréstimo não apresenta ruído branco.

Estacionariedade

Realiza-se o Teste de Hipóteses (*Augmented Dickey-Fuller Test*) definido em 3.1 que mostra se a série **Emprestimo** possui observações estacionárias ou não (considerando um nível de significância com $\alpha = 5\%$):

H_0: $\phi = 1$, o processo $\{Y_t\}_{t\in\mathbb{Z}}$ não é estacionário e conhecido como passeio aleatório;

H_1: $|\phi| < 1$, o processo $\{Y_t\}_{t\in\mathbb{Z}}$ é estacionário.

Como o nível de significância de $\alpha = 5\%$ e o valor-$p = 0,3016$, isto é, valor-$p > 0,05$, logo o H_0 não é rejeitado, indicando assim que os dados não são estacionários. Com essa afirmação, o passo três da Seção 2.2.1 precisou ser realizado.

Diferença

Para preparar as observações da variável **Emprestimo**, foram realizadas diferenças com o intuito de concluir com o passo três a realização de uma previsão adequada do modelo $ARIMA$ (Hyndman e Athanasopoulos, 2021).

Note a Figura 3.20 e veja a série da variável em estudo, só que dessa vez, utilizando a diferença de tamanho 1.

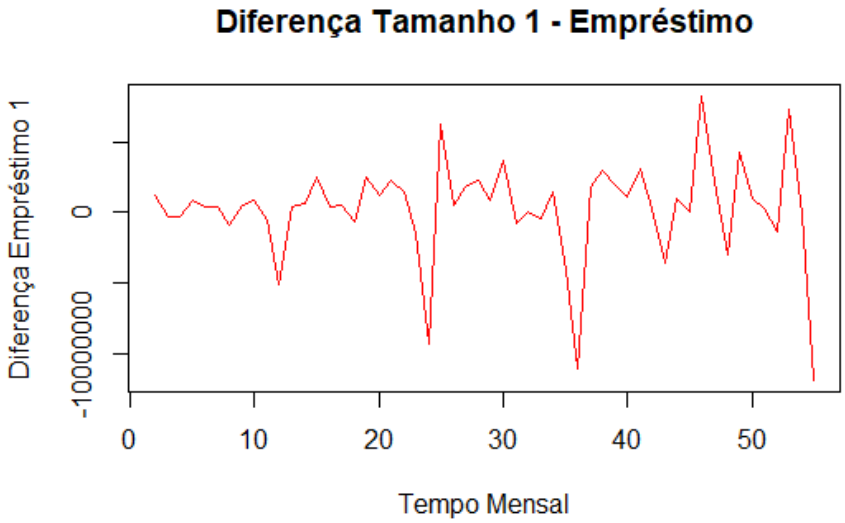

Figura 3.20: Gráfico da diferença de tamanho 1 da série estudada.

Observe na Figura 3.5 um comportamento específico dentre o período de 12 meses, como mencionado a respeito da sazonalidade anteriormente. Entretanto, para os últimos meses, esse comportamento fica mais característico em um intervalo de seis meses (alteração que pode ter acontecido em decorrência da Pandemia). Verifique que é notável perceber estacionariedade na série de empréstimo e para comprovar isso, foi realizado novamente o Teste de Estacionariedade (*Augmented Dickey-Fuller Test*) e como o nível de significância é de $\alpha = 5\%$ e o valor-$p = 0,0302$ (valor-$p \leq 0,05$), logo a série **Emprestimo** passa a possuir observações estacionárias, pois rejeita-se H_0.

Para identificar a sazonalidade referida, note a Figura 3.21 que destaca o período definido como 12 meses.

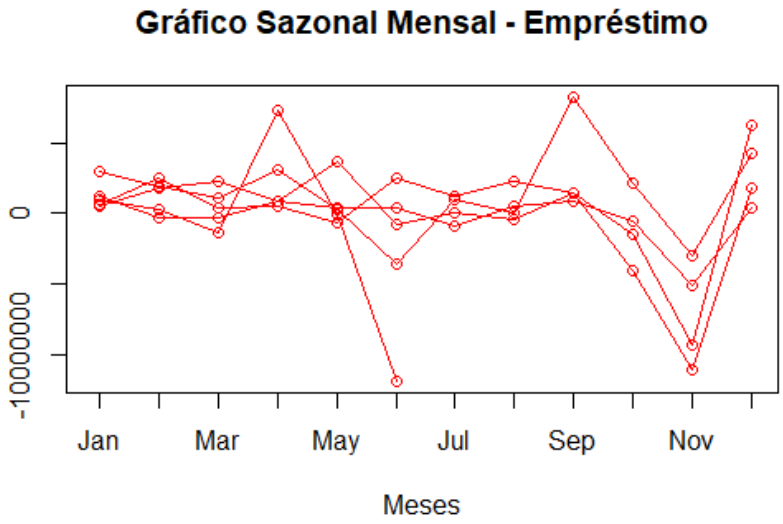

Figura 3.21: Gráfico Sazonal mensal contendo as informações do empréstimo.

Quando a diferença é aplicada na variável **Emprestimo**, identifica-se que a tendência é retirada da variável em estudo, como mostrado no teste de estacionariedade, então o gráfico de sazonalidade mensal da Figura 3.21 mostra um comportamento mais estável dentre os meses, exceto no mês de Abril, Junho, Setembro, Outubro, Novembro e Dezembro (como é abordado o comportamento mensal, sabe-se que a sazonalidade é de ordem 12). Verifique que sem a tendência houve uma queda nos meses de Junho e Outubro e outras no mês de Novembro. Em Outubro, a série começou a apresentar um comportamento menor em relação as demais e a queda acaba ocorrendo no mês de Novembro. Nos meses de Junho, Setembro e Dezembro, há um aumento da série de empréstimo em relação aos outros meses. A Pandemia pode ter ocasionado esses momentos de picos e quedas identificados.

Para identificar se a série **Emprestimo** mostra um ruído branco após aplicar a diferença de tamanho 1, será apresentado novamente o gráfico ACF e $PACF$ da Figura 3.22, mas agora contendo a aplicação da diferença.

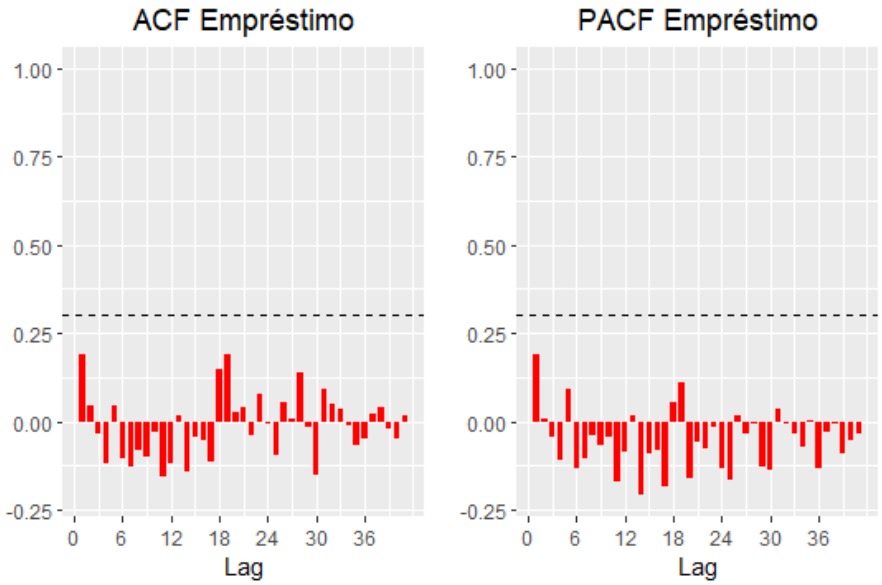

Figura 3.22: Gráfico ACF e $PACF$ do empréstimo.

Veja que, com a aplicação da diferença de tamanho 1 na série da Figura 3.22, as observações não apresentam nenhum período ou comportamento específico e próximo entre si, permanecendo mais dispersas em torno da reta 0. Também nota-se que nenhum *lag* ultrapassa o limite delimitado pelo gráfico, provando assim que a variável **Emprestimo** definitivamente passa a conter uma série de ruído branco.

Com a realização da diferença nas observações da variável em estudo, é possível concluir o item três dos passos para a previsão de um modelo ARIMA e continuar os próximos procedimentos.

3.4.2 $ARIMA$

Para encontrar o modelo $ARIMA$ mais adequado para esse conjunto de dados, foi realizada a função do *RStudio* denominada como *auto.arima*. Com essa detecção, o modelo $ARIMA$ identificado como o melhor para a variável em estudo é o $ARIMA(0,1,0)$.

Como o modelo aplicado será o $SARIMA$, foram realizados testes com três modelos para a série, contendo em todos o $ARIMA$ selecionado pelo *RStudio* e se diferenciando na sazonalidade.

Observe os três modelos estudados:

$$SARIMA(0,1,0)(0,0,0)_{12}; \tag{3.4}$$

$$SARIMA(0,1,0)(0,1,1)_{12} \tag{3.5}$$

e

$$SARIMA(0,1,0)(1,1,0)_{12}. \tag{3.6}$$

Para mais detalhes sobre os Modelos ajustado, verifique as Tabelas 3.14, 3.15 e 3.16:

Tabela 3.14: Resultados do Modelo 3.4

	Estimados	SE	valor-t	valor-p
Constante	237251,3	483734,4	0,4905	0,6258

Observe que na Tabela 3.14, tem apenas constante (Modelo 3.4) que não é significativa, pois o seu valor-$p = 0,6258 > 0,05$. Logo, levando em consideração um nível de significância com $\alpha = 5\%$, conclui-se que a constante não é significativa.

Tabela 3.15: Resultados do Modelo 3.5

	Estimados	SE	valor-t	valor-p
$sma1$	-0,2585	0,2188	-1,1814	0,2442

A Tabela 3.15 também apresenta um coeficiente não significativo, pois possui um valor-$p = 0,2442$, isto é, valor-$p > 0,05$. Logo a sazonalidade em $MA(1)$ não é significativa.

Tabela 3.16: Resultados do Modelo 3.6

	Estimados	SE	valor-t	valor-p
$sar1$	-0,2379	0,2027	-1,1738	0,2472

Note que na Tabela 3.16 também é apresentado um coeficiente não significativo, pois possui um valor-$p = 0,2472$, ou seja, valor-$p > 0,05$. Logo a sazonalidade em $AR(1)$ não é significativa.

Seguindo a explicação enunciada sobre seleção de modelos na Seção 2.2.1, serão verificados AIC, AIC_C e BIC na Tabela 3.17:

Tabela 3.17: Resultados de seleção do Modelo 3.4, 3.5 e 3.6 da série de empréstimo.

	AIC	AIC_C	BIC
Modelo 3.4	33,0792	33,0806	33,1528
Modelo 3.5	32,8666	32,8690	32,9494
Modelo 3.6	32,8684	32,8708	32,9512

Observando os resultados obtidos pela seleção do modelo, nota-se que o Modelo 3.5 é o mais adequado para ajuste dos dados. Veja que os valores só se diferenciam na segunda casa decimal de um para o outro ou em até uma unidade. Com essa seleção, pode-se verificar que o Modelo 3.4 é o que menos se aplica aos dados.

Verifique a Figura 3.23 que irá mostrar um conjunto de gráficos dos resíduos do Modelo 3.4:

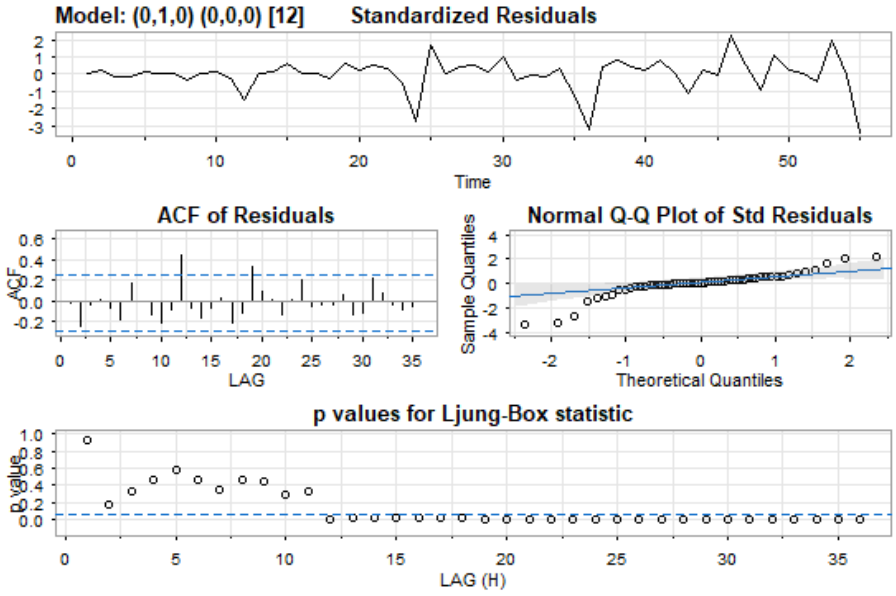

Figura 3.23: Resíduos do Modelo 3.4 da série **Emprestimo**.

Verifique que na Figura 3.23 os resíduos do gráfico ACF não apresentam um comportamento de ruído branco, pois mesmo estando dispersos ao redor da reta 0 existem dois pontos que ultrapassam os limites delimitados. No gráfico Normal *qqplot*, observe que a maioria dos resíduos estão sobrepostos sobre a reta, porém os primeiros e os últimos pontos, se afastam um pouco da reta, logo não se pode afirmar que eles apresentam um comportamento de normalidade. Quando observados os resíduos com teste de *Ljung-Box* nota-se que eles rejeitam H_0, isto é, possuem auotocorrelação (Ferreira *et al.*, 2015). Portanto, a princípio os resíduos do Modelo 3.4 não são tão adequados para os dados.

Note agora a Figura 3.24 que apresenta um conjunto de gráficos dos resíduos do Modelo

3.5:

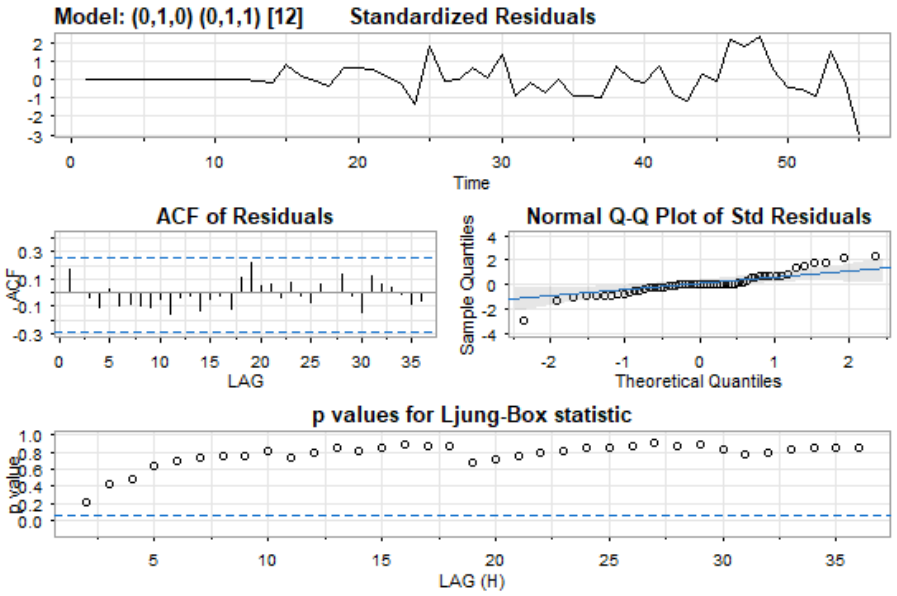

Figura 3.24: Resíduos do Modelo 3.5 do empréstimo.

Analisando os gráficos dos resíduos na Figura 3.24, verifica-se que as análises são diferentes da Figura 3.23, ou seja, mostra que os resíduos estão mais próximos de serem um ruído branco, o gráfico ACF apresenta um comportamento de ruído branco, pois nenhum *lag* ultrapassa as linhas delimitadas e o gráfico Normal *qqplot* indica que a maioria dos pontos estão sobrepostos na reta. Um ponto em destaque é que o teste de *Ljung-Box* não rejeita H_0 e os pontos estão distantes do limite estabelecido, logo os resíduos não são autocorrelacionados (Ferreira *et al.*, 2015). Com isso, destaca-se que o Modelo 3.5 pode ser utilizado como previsão.

Veja agora a Figura 3.25 que irá mostrar os resíduos do Modelo 3.6:

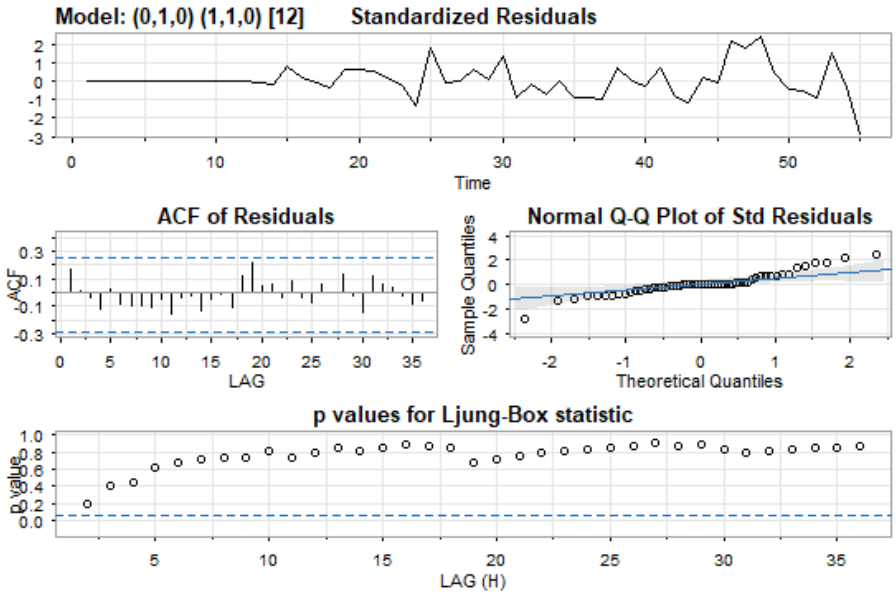

Figura 3.25: Resíduos do Modelo 3.6 da variável **Emprestimo**.

Na Figura 3.25 os gráficos dos resíduos estão adequados de acordo com o que é esperado. O *ACF* dos resíduos estão apresentando um comportamento de ruído branco, sem observações além dos limites estabelecidos. No gráfico Normal *qqplot*, nota-se que os pontos estão bem próximos da reta. No último gráfico, percebe-se que o teste de *Ljung-Box* não rejeita H_0, indicando assim que os pontos não estão autocorrelacionados (Ferreira *et al.*, 2015). Portanto, o Modelo 3.6 apresenta um comportamento adequado em relação aos resíduos.

Com todas as análises para identificar o melhor modelo, concluí-se que o mais adequado para a variável em estudo é o Modelo 3.5, indicado pelo *Software R-Studio*, pois os resíduos estão adequados e na seleção de modelos apresenta os menores valores em relação aos outros modelos. Então com essa escolha, os itens quatro, cinco e seis dos passos para a realização de uma previsão com modelo ARIMA foram concluídos, mas foi realizado uma detecção dos *outliers* para verificar se os resíduos possuem esse valor e elaborar uma tratativa do modelo.

Para a identificação dos *outliers* presentes nos resíduos do Modelo 3.5 da variável em estudo, foram utilizadas duas funções do *Software R-Studio*: *detectAO* e *detectIO* (pertencentes ao pacote *TSA*) respectivamente, mostra se existem *outliers* aditivos e/ou se existem *outliers* inovadores (intervenção).

Observando ambas as funções para a série, foi detectado o *outlier* aditivo da posição 55, logo esse ponto afeta os resíduos e três observações para os *outliers* de intervenção,

que são as posições 24, 36 e 55 dos resíduos, identificando assim que após a posição 24 as observações são afetadas (veja mais em Almeida *et al.* (2007)). Veja como ficam os coeficientes do Modelo 3.5 na Tabela 3.18, quando esses *outliers* são tratados:

Tabela 3.18: Coeficientes do Modelo 3.5 com a tratativa dos *outliers* para a série de empréstimo.

	sma1	**IO24**	**IO36**	**IO54**
	-0,7641	-3516498	-6305320	-11201890
s.e.		1420272		1798815

Note agora os coeficientes do Modelo 3.5 com a tratativa das observações *outliers* e com algumas informações faltantes. Entretanto, com os *outliers* tratados, observe novamente o comportamento dos resíduos na Figura 3.26, para verificar se eles estão adequados:

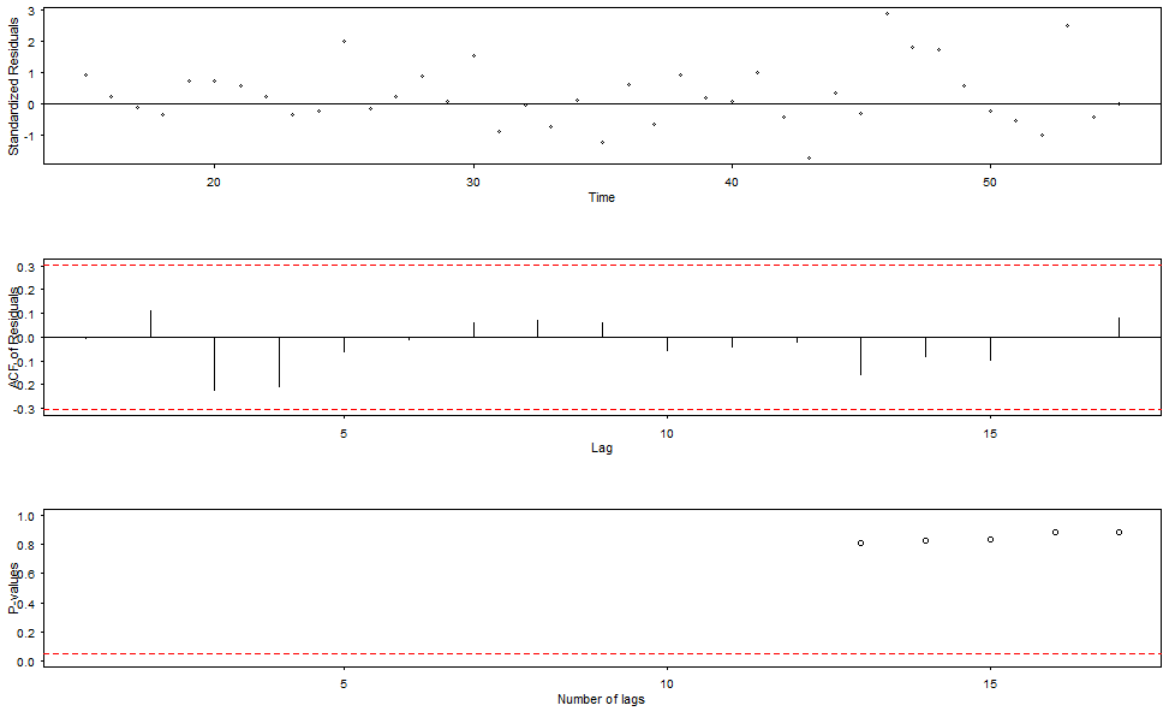

Figura 3.26: Resíduos do Modelo 3.5 da série **Emprestimo** com a tratativa dos *outliers*.

Na Figura 3.26, nota-se que os resíduos estão adequados para o empréstimo. Nos dois primeiros gráficos observa-se que os dados estão dispersos ao redor da reta 0 e nenhum ultrapassa as linhas delimitadas, ou seja, as observações passam a ser parte de um ruído branco. Por fim, observa-se que no último gráfico, o teste de *Ljung-Box* não rejeita H_0, logo os resíduos não têm autocorrelação e possuem um comportamento adequado para um modelo.

Mesmo o Modelo 3.5 não apresentando um $sma1$ significativo, o seu comportamento em relação aos resíduos são adequados e, quando realizada a tratativa dos *outliers*, possui também um resultado coerente com o que é esperado. Além disso, contém os menores valores, quando comparado com os métodos de seleção de outros modelos. Logo, é o modelo mais indicado para prosseguir com a análise. Outro motivo interessante para a escolha desse modelo, foi o fato do ajuste da série provisão total possuir a mesma ordem de modelo $SARIMA(1,1,1)(0,1,1)_{12}$.

Previsão

Como o ajuste da variável **Emprestimo** já foi feito, foi feita a previsão do Modelo $ARIMA$ escolhido (Modelo 3.5 com a tratativa dos *outliers*) que é o último item para os passos de uma previsão. Para conseguir prever qual serão os valores da variável em estudo necessários para os próximos meses, foi escolhido o número de 12 meses, o equivalente a um ano de previsão em relação aos valores obtidos anteriormente para visualizar melhor qual será o comportamento da série.

Observe na Tabela 3.19 os valores da série de empréstimo acompanhados de seus intervalos de confiança:

Tabela 3.19: Valores de previsão com 12 meses da série **Emprestimo** com limites inferiores e superiores do intervalo de confiança.

Ano/Mês	Limite Inferior	Empréstimo Previsto	Limite Superior
2021/08	63782803	68460833	73138863
2021/09	62162937	68778670	75394404
2021/10	65280430	73383016	81485602
2021/11	63700227	73056287	82412347
2021/12	57792420	68252813	78713206
2022/01	58726449	70185235	81644021
2022/02	59190977	71564769	83938561
2022/03	59376599	72602244	85827889
2022/04	58508399	72534256	86560112
2022/05	62159739	76942553	91725367
2022/06	61978853	77481709	92984565
2022/07	59796942	75987850	92178758

Na Tabela 3.19, percebe-se que a previsão para o próximo ano do empréstimo são valores entre 68 e aproximadamente 77 milhões de reais. O menor valor da variável **Emprestimo** está previsto para Dezembro desse mesmo ano (2021) e o maior valor em Junho de 2022. No ano de 2022, todos os valores ultrapassam os 70 milhões, sendo os

últimos três meses (Maio, Junho e Julho) os maiores empréstimos previstos. Então, nota-se que os valores resultantes da previsão do empréstimo são bem maiores em relação aos valores observados anteriormente, pois o valor mínimo da previsão de 68 milhões é próximo do ponto máximo observado.

Quando comparados os valores colocados dentre os intervalos, verifica-se que os valores se encontram entre o mínimo de aproximadamente 58 milhões e um valor máximo de aproximadamente 93 milhões, isto é, a previsão possui um intervalo maior em relação aos valores observados e além disso contém valores bem maiores.

Com o intuito de conseguir visualizar melhor essa previsão, comparada com os valores anteriores da variável em estudo, note agora a Figura 3.27:

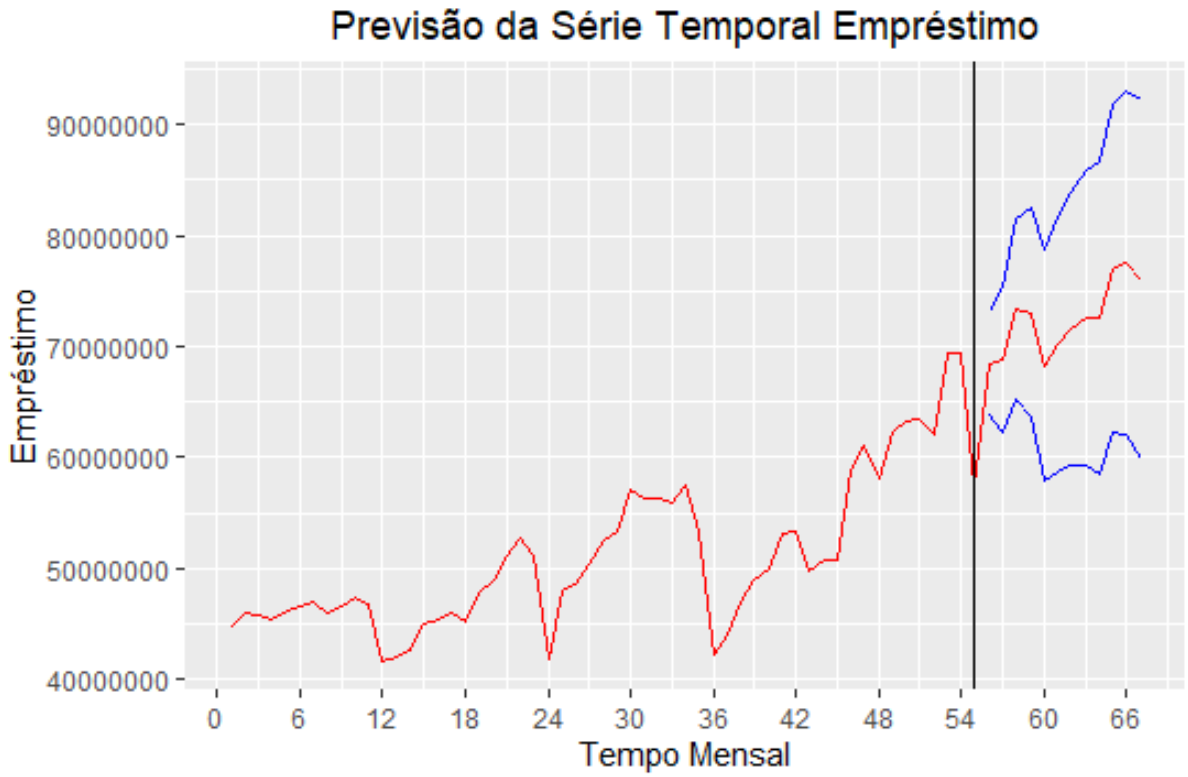

Figura 3.27: Previsão de 12 meses com limites inferiores e superiores da série.

Veja na Figura 3.27 que pelo comportamento da série de empréstimo comparada a previsão, como mencionado anteriormente, nota-se que os valores previstos possuem comportamentos maiores em relação aos observados. O intervalo de confiança inferior, tem um comportamento estável em relação a série **Emprestimo** observado e o limite superior do intervalo, está ainda maior do que os valores da previsão.

Com a realização desse processo, os passos do estudo de uma previsão com o modelo $ARIMA$ foi concluído e os resultados foram apresentados, portanto obtém-se uma análise

de previsão completa do empréstimo. Os comandos para realização desse processo, se encontra na Seção B.

3.4.3 *Holt-Winters*

Com o intuito de utilizar outra aplicação foi escolhido a metodologia de *Holt-Winters*, porque esse método aborda os estudos de séries temporais e futuramente é possível compará-lo com outras técnicas, principalmente a realizada com o modelo $ARIMA$.

Os passos para uma previsão adequada desse método é bem próximo ao utilizado de Hyndman e Athanasopoulos (2021), com algumas particularidades em relação ao modelo, porém com a apresentação de 12 meses de previsão e do gráfico. As análises descritivas da série serão reaproveitadas para estudo dessa metodologia.

O Método Multiplicativo de *Holt-Winters* aborda a equação de previsão em conjunto com três equações de suavização que envolve nível, tendência e componente sazonal (respectivamente α, β^* e γ). De acordo com a variável **Emprestimo** e realizando as Fórmulas 2.12, 2.13 e 2.14, chega-se nos seguintes resultados da Tabela 3.20:

Tabela 3.20: Resultados das Fórmulas 2.12, 2.13 e 2.14 de *Holt-Winters* da variável em estudo.

Equação	**Resultado**
α	0,6694
β^*	0
γ	1

Observe na Tabela 3.20 que a equação β^* é zerada para a série, neste caso percebe-se que para essa metodologia não foi considerada nenhuma tendência. O valor do componente sazonal é 1 e o do nível é 0,6694 e esses resultados serão utilizados para encontrar a Equação 2.11.

Note agora a Figura 3.28 da série de empréstimo em preto e a série de acordo com *Holt-Winters* em vermelho:

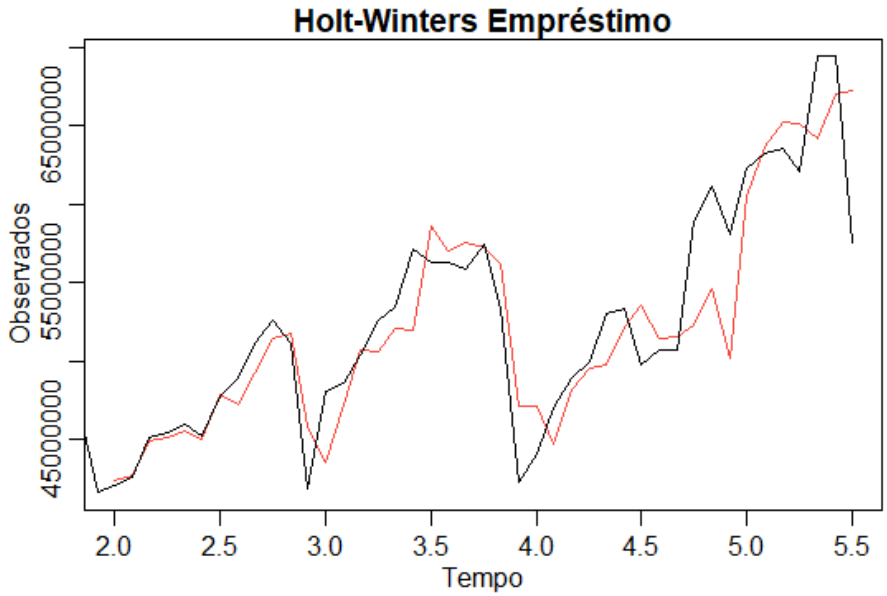

Figura 3.28: Série **Emprestimo** com método de *Holt-Winters* para verificação.

Note que a Figura 3.28 mostra o empréstimo (em preto) e a série realizada com o ajuste do modelo de *Holt-Winters* (vermelho). Veja que ambas possuem um comportamento semelhante, acompanhando em momentos de picos ou quedas e pra esse caso, as observações não são as iniciais. Porém, na posição entre 4 e 4,5, o ajuste de *Holt-Winters* permanece abaixo enquanto os observados são maiores.

Com o conceito de *Holt-Winters* será feita a previsão dos próximos 12 meses da variável **Emprestimo**. Verifique a Tabela 3.21:

Tabela 3.21: Valores de previsão com 12 meses da variável em estudo pelo método de *Holt-Winters* com limites inferiores e superiores do intervalo de confiança.

Ano/Mês	Limite Inferior	Empréstimo Previsto	Limite Superior
2021/08	56101521	62512750	68923978
2021/09	55518162	63258095	70998028
2021/10	58583521	67729085	76874648
2021/11	55135941	64987439	74838937
2021/12	46074259	55866712	65659166
2022/01	47692637	58726642	69760647
2022/02	47876274	59864999	71853724
2022/03	48291075	61226637	74162198
2022/04	48056919	61762343	75467768
2022/05	50420823	65455085	80489348
2022/06	48593340	63900583	79207826
2022/07	45286932	58609368	71931803

Na Tabela 3.21 nota-se com a previsão realizada por essa metodologia que a série em

estudo nos próximos meses terá um valor entre aproximadamente 56 e 68 milhões, sendo a menor observação da série de empréstimo em Dezembro de 2021 e a maior em Outubro de 2021. O valor mais alto é praticamente o máximo destacado nos valores observados da série **Emprestimo**, portanto esses resultados apresentaram um comportamento um pouco mais alto em relação ao empréstimo.

Os diferentes intervalos destacados se encontram entre aproximadamente 45 e 80 milhões. Isto é, são intervalos bem grandes e o valor máximo mostrado é maior que o valor da variável **Emprestimo**.

Para entender como esses valores estão distribuídos na série de previsão, veja a Figura 3.29:

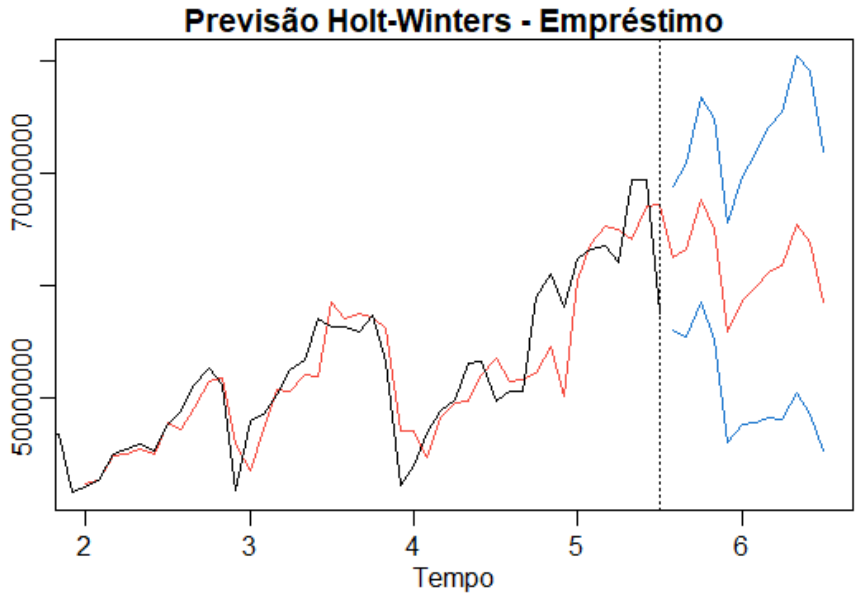

Figura 3.29: Gráfico contendo a série da variável em estudo junto com a previsão de *Holt-Winters* para os próximos 12 meses.

Note com a Figura 3.29 que a série contendo a previsão do empréstimo para o próximo ano, possui um comportamento estável e ao mesmo tempo um pouco maior em relação as observações anteriores, além disso contém uma queda no final do ano de 2021 (Dezembro), assim como costuma acontecer em fechamento de semestres. Os intervalos são grandes e capazes de abranger os momentos em que os valores da série **Emprestimo** aumentarem ou diminuírem.

3.4.4 *Extreme Gradient Boosting*

Para realizar a metodologia de *XGBoost* na variável *Emprestimo* foram utilizados os mesmos procedimentos iniciais da Seção 3.2.4 até a produção do modelo (Alice, 2020). Logo, os mesmos parâmetros escolhidos para a variável *Provisao_Total* também foram usados em *Emprestimo*.

O modelo *XGBoost* com o melhor *bestTune* do empréstimo é apresentado na Tabela 3.22.

Tabela 3.22: Resultados de Parâmetros e Performances do Modelo *XGBoost* para a variável *Emprestimo*.

nrounds	*max_depth*	*eta*	*gamma*	*colsample_bytree*	*min_child_weight*	*subsample*
200	15	0,1	0	1	1	1

Note na Tabela 3.22 que o que diferencia o empréstimo da performance de melhor modelo em relação a provisão total apresentada na Seção 3.2.4 é o *nrounds* ser equivalente a 200, logo esse é o número que encontra a melhor performance pra essa variável. A profundidade da árvore e os outros parâmetros são semelhantes aos apresentados na *Provisao_Total*.

Assim como o procedimento criado na provisão total para encontrar os intervalos de confiança dos valores previstos, mostrados na Seção 3.2.4, também foi realizado o mesmo em empréstimo e a previsão para o próximo ano acompanhada de seus respectivos intervalos de confiança é apresentado na Tabela 3.23.

Tabela 3.23: Valores de previsão com 12 meses da série pelo método de *XGBoost* com limites inferiores e superiores definidos como intervalo de confiança.

Ano/Mês	Limite Inferior	Provisão Total Prevista	Limite Superior
2021/08	57507840	57524484	57534272
2021/09	57507840	57523583	57532848
2021/10	57510016	57542119	57562116
2021/11	58279624	58432097	58521644
2021/12	57553092	57589264	57610508
2022/01	62319736	62323158	62325168
2022/02	63232284	63246608	63255020
2022/03	63453380	63501986	63530532
2022/04	62127912	62131537	62137708
2022/05	69356760	69421995	69460312
2022/06	69356760	69401407	69427640
2022/07	57507840	57524470	57534248

Na Tabela 3.23 observa-se com a previsão de *XGBoost* que a série do empréstimo nos próximos 12 meses terá valores entre aproximadamente 57 e 69 milhões com o menor valor de **Emprestimo** em Setembro de 2021 e o maior em Maio de 2022. Novamente os limites se encontram mais próximos do valor original da série estudada.

Com o intuito de entender o comportamento da previsão em conjunto com a série, verifique a Figura 3.30.

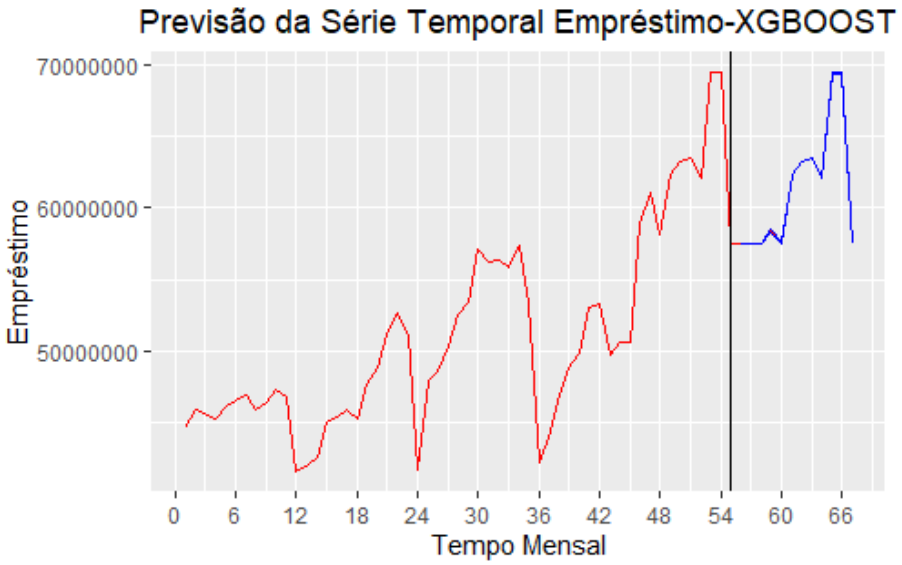

Figura 3.30: Previsão da série de Empréstimo com a metodologia *XGBoost*.

É possível observar na Figura 3.30 que, como os limites estão próximos do valor previsto, logo não é trivial verificar a performance da previsão e dos limites, mas sim, de uma maneira geral. Nos primeiros meses da previsão, os empréstimos permaneceram

mais estáveis entre 57 e 58 milhões, porém esses valores se elevaram a partir de 2022, repetindo o mesmo comportamento que ocorreu no começo do ano de 2021.

A mesma justificativa comentada na Seção 3.2.4 a respeito da performance da metodologia *XGBoost* também ocorre para a variável empréstimo. Então, é provável que devido ao número de observações, o método não tenha sido tão efetivo assim, por repetir a mesma performance que ocorreu antes na série.

3.5 Comparação entre as Metodologias aplicadas nos Dados

Com o intuito de entender quais foram as melhores metodologias para cada variável em estudo do conjunto de dados da empresa de consórcio foi feita uma comparação entre os modelos, verificando se a previsão está adequada ou próxima do valor real.

Logo, foram utilizadas as últimas 12 observações da série de cada variável como valores originais e retiradas do banco de dados para serem usadas na comparação.

Então foi realizada todas as metodologias novamente, com os mesmos modelos e parâmetros encontrados, porém agora com um banco de dados com apenas 43 observações, isto é, a retirada de 12 observações da base completa.

Depois que todos os métodos são elaborados de novo, principalmente o da previsão de um ano, observa-se a série da base completa com os valores originais em comparação com as previsões do modelo *ARIMA, Alisamento Exponencial* e *XGBoost*.

Para comparação é utilizada a técnica de soma das 12 observações de previsão do erro absoluto que calcula a diferença entre o valor original e o previsto e a soma do erro relativo que é o erro absoluto dividido pelo valor real de interesse.

3.5.1 Comparação na Variável *Provisao_Total*

Elaborando o procedimento descrito nessa Seção 3.5, note a Figura 3.31 que contém as séries compostas pelas observações originais e as previsões do modelo *ARIMA, Alisamento Exponencial* e *XGBoost*.

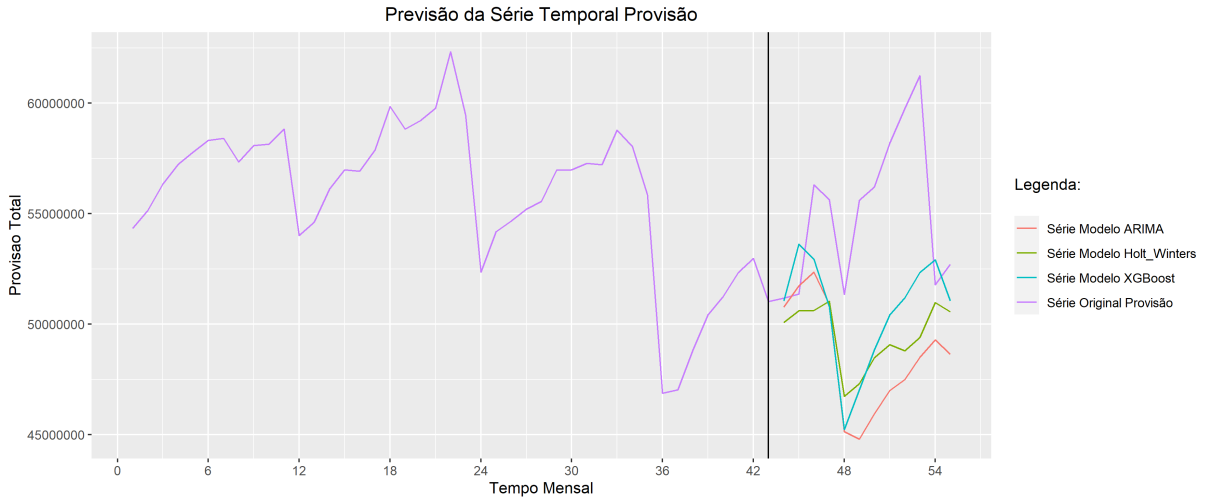

Figura 3.31: Série original em conjunto com as previsões da variável *Provisao_Total* com todas as metodologias vistas.

Veja que na Figura 3.31 a série contendo as observações originais da variável em estudo em roxo, indica que os valores reais estão bem mais elevados do que os previstos, porém isso provavelmente deve ter ocorrido, devido as circunstâncias da Pandemia no país.

A previsão do modelo *ARIMA* em vermelho aparenta estar mais distante das observações originais em relação as outras previsões. Entretanto, o mesmo comportamento de aumento e diminuição da provisão também é previsto de maneira correta, ele só está mais abaixo em relação ao valor real.

A série de previsão da metodologia *Holt Winters* na cor verde e a metodologia de *XGBoost* na cor azul são as que estão mais próximas da provisão total original. No caso, a previsão *XGBoost* é a que realiza um comportamento semelhante dos aumentos e diminuições dos valores da provisão, enquanto a previsão *Holt Winters* não se assemelha tanto nos momentos de pico ou de baixas dos valores de provisão.

Para entender melhor qual metodologia foi a mais efetiva, observe na Figura 3.24 os valores referentes aos erros absolutos e relativos.

Tabela 3.24: Erros absolutos e Relativos das metodologias estudadas.

	Erro Absoluto	**Erro Relativo**
Metodologia *ARIMA*	78761900	17,2098
Metodologia *Holt Winters*	67674341	14,7871
Metodologia *XGBoost*	53893936	11,776

Verifique na Tabela 3.24 que os erros absolutos variam em uma diferença entre aproximadamente 53 e 78 milhões de reais, logo o menor erro absoluto apresentado é o da

Metodologia *XGBoost*.

Os erros relativos estão entre aproximadamente 11 e 17 pontos, portanto a melhor metodologia seria a que possui o menor erro que novamente é a do modelo *XGBoost*.

Então, o melhor modelo que prevê a série de *Provisao_Total* é o de *XGBoost*, porém destaca-se os pontos comentados na Seção 3.2.4 sobre a performance da metodologia em relação ao número de observações do banco de dados. Caso isso for levado em consideração, utilizando Séries Temporais, o melhor método seria o de *Holt Winters*, pois é o que apresenta ambos os erros medianos.

3.5.2 Comparação na Variável *Emprestimo*

Realizando os mesmos procedimentos feitos na variável *Provisao_Total* na Seção 3.5.1, veja agora a Figura 3.32 que apresenta todas as previsões e as observações originais da variável *Emprestimo*.

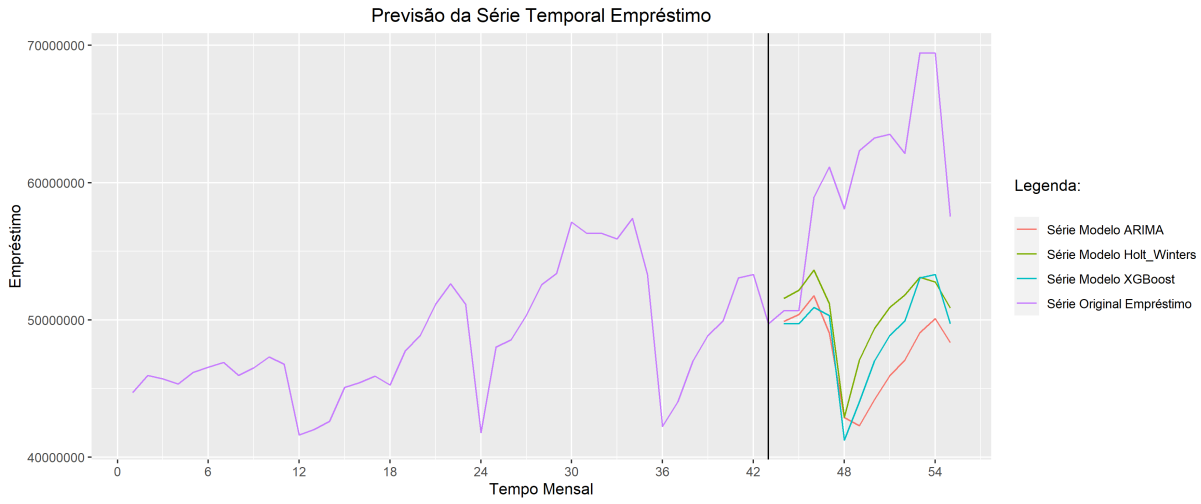

Figura 3.32: Série original em conjunto com as previsões da variável *Emprestimo* com todas as metodologias vistas.

Observe na Figura 3.32 que os valores de empréstimo, na série em roxo, aumentam muito nas últimas 12 observações e essa elevação nenhuma previsão consegue capturar. Esse aumento pode ter ocorrido em detrimento da Pandemia, logo os modelos de previsão não foram capaz de se aproximar tanto desse comportamento.

A previsão do modelo *ARIMA* em vermelho faz um comportamento de elevação dos empréstimos, depois decai e aumenta novamente. O mesmo acontece com as previsões de *Holt Winters* em verde e de *XGBoost* em azul. Portanto, pela Figura 3.32 não é possível distinguir ao certo a previsão que mais se aproxima dos dados originais.

Note agora com a Tabela 3.25 os erros absolutos e relativos de cada metodologia:

Tabela 3.25: Erros absolutos e Relativos das metodologias estudadas.

	Erro Absoluto	**Erro Relativo**
Metodologia *ARIMA*	156090829	31,1989
Metodologia *Holt Winters*	119736696	23,9325
Metodologia *XGBoost*	139263026	27,8354

Verifique na Tabela 3.25 que diferente da análise anterior na Seção 3.5.1, os valores estavam com erros absolutos abaixo dos 100 milhões, para esse caso, os erros absolutos se encontram aproximadamente entre 119 e 156 milhões em empréstimo. Portanto, quando se observa o erro absoluto, a melhor metodologia seria a de *Holt Winters* para essa variável, pois apresenta o menor valor comparada as outras técnicas utilizadas.

Os pontos de erros relativos apresentados estão entre aproximadamente 27 e 31. Logo, a melhor metodologia verificando o erro relativo também é de *Holt Winters*.

Então, no caso da variável *Emprestimo* a melhor metodologia para o conjunto de dados é a de *Holt Winters*.

Possivelmente com a influência da Pandemia no Brasil nos anos de 2020 e 2021, as observações das variáveis analisadas de provisão total e de empréstimo sofreram grandes alterações em relação ao seu comportamento gradual.

Isso implica ressaltar que as previsões de todas as metodologias, quando comparadas com seus respectivos valores originais, não foram tão efetivas e corretas, porém capturaram os momentos com valores mais altos e mais baixos de ambas as variáveis.

Quando analisada a metodologia da *Provisao_Total*, identifica-se que a mais adequada como modelo e previsão é o *XGBoost*, entretanto existe a preocupação com a repetição das últimas observações e a performance em relação ao tamanho total do conjunto de dados. Em segundo lugar existe o método de *Alisamento Exponencial - Holt Winters* que possui seus erros relativo e absoluto mais baixo que o do modelo *ARIMA*.

Para a variável *Emprestimo* a melhor metodologia em ambos os erros é o de *Holt Winters* que sua série de previsão se aproxima mais dos valores originais de empréstimo dos últimos 12 meses da base analisada.

No geral, a metodologia de *ARIMA* não foi eficiente para esse conjunto de dados da empresa de consórcio e o que conseguiu explicar de maneira mais adequada ambas as

variáveis foi a metodologia de *Alisamento Exponencial*, pois foi a mais indicada para o empréstimo e uma das melhores na provisão total. Portanto, a melhor previsão para o próximo ano do empréstimo e da provisão total é o apresentado na metodologia de *Holt Winters*.

Capítulo 4

Conclusão

Primeiramente foram realizados todos os passos para elaborar uma previsão de um modelo $ARIMA$, seguindo a estrutura da Seção 2.2.1 para a variável **Provisao_Total**. Depois da apresentação desses resultados, foi aplicado a previsão com a metodologia de *Holt-Winters* da Seção 2.2.2 e a metodlogia da *XGBoost* da Seção 3.2.4, afim de estudar três métodos diferentes e analisar os resultados obtidos para ambos os casos.

Com o intuito de fazer um estudo para mais uma variável, foram escolhidos os dados do **Emprestimo** e verificado a correlação entre o mesmo com a **Provisao_Total** para identificar se seria possível dar prosseguimento a análise dessa maneira. Neste caso, foi determinado que realizar essas análises seria interessante e o estudo foi prosseguido de forma separada entre as variáveis (Seção 3.3).

Para dar prosseguimento as análises, foi realizado a previsão de um modelo $ARIMA$ para a variável **Emprestimo**, seguindo a ordem dos passos definidos na Seção 2.2.1. Após obter os resultados, foi aplicada a metodologia de *Holt-Winters* e a de *XGBoost*. Logo, os mesmos procedimentos elaborados na variável **Provisao_Total**, também foram feitos para essa variável, visando um trabalho mais completo.

Nota-se que a metodologia $ARIMA$ não foi eficiente para esse conjunto de dados da empresa de consórcio, quando comparado os erros relativos e absolutos e o que conseguiu explicar de maneira mais adequada ambas as variáveis foi a metodologia de *Alisamento Exponencial*, pois foi a mais indicada para o empréstimo e uma das melhores na provisão total, acertando mais nos momentos de previsão.

Embora o *XGBoost* tenha apresentado o melhor resultado de erro na variável **Provisao_Total**, essa metodologia não obteve performance na análise dos dados, pois de acordo com Medeiros (2021), a modelagem de *XGBoost* é mais apropriada para bases grandes e

no caso a análise foi feita com poucas observações.

Portanto, para a determinada empresa de consórcio, a previsão mais indicada é a de *Alisamento Exponencial*, pois estará utilizando Séries Temporais e acertará mais em seus valores.

Como existem resultados para a previsão da variáveis **Provisao_Total**, a empresa pode se atentar aos valores previstos para prevenir o LGD e auxiliar a função administradora da organização na saúde financeira dos grupos atuais e futuros e no momento adequado da entrega do veículo.

Para um estudo futuro, seria interessante a análise de aspectos contábeis na série para identificar outros fatores que possam influenciar em seu comportamento, bem como verificar a visão carteira do banco de dados, com a inserção de covariáveis explicativas da série.

Como a Pandemia influenciou nos resultados obtidos do modelo e da previsão, poderia também pensar em possibilidades de realizar uma tratativa desse contexto e criar covariáveis para uma análise complementar como a *baixa contábil de crédito* ou *atraso completo de 180 dias* para tentar diminuir o prejuízo final ou recuperá-lo.

Questões contábeis também poderiam ser realizadas com estratégicas de estatística bayesiana, pois aumenta a volatividade para ajustes e previsões.

Uma outra sugestão é realizar os modelos por segmentação, dividindo assim cada um por setores da empresa como o de banco, consórcio e varejo.

Esse trabalho acrescentou muito na minha formação por entender melhor o mercado de Consórcio e principalmente para trabalhar mais as metodologias de Séries Temporais e de *Machine Learning* que são estudadas na graduação.

Apêndice A

Símbolos Matemáticos

- Y_t é uma variável aleatória definida num espaço de probabilidade $(\Omega, \mathcal{A}, \mathcal{P})$, também denominada como série temporal estacionária;
- I é o conjunto de índices temporais, sendo discreto ou contínuo;
- t é o tempo definido na série temporal, podendo ser de 1 até T que é o comprimento total da série;
- L é a verossimilhança dos dados;
- p, d e q são os parâmetros do modelo ARIMA;
- ℓ_t denota uma estimativa do nível da série no momento t;
- b_t é uma estimativa da tendência (inclinação) da série no momento t;
- α e o parâmetro de suavização para o nível, $0 \leq \alpha \leq 1$;
- β^* e o parâmetro de suavização da tendência, $0 \leq \beta^* \leq 1$;
- s_t é o componente sazonal do modelo que é composto por γ, sendo $\gamma = \gamma^*(1-\alpha)$, a restrição usual do parâmetro é $0 \leq \gamma^* \leq 1$, que se traduz em $0 \leq \gamma \leq 1-\alpha$;
- m é o período da sazonalidade de acordo com os dados utilizados;
- ϵ_t é o ruído branco, definido na Seção 2.2.1;
- ψ é o parâmetro que compõem o modelo, sendo $\psi_0 = 1$;
- T^* é o número de observações usadas para estimativa;

- SSE - Soma de Quadrados dos Resíduos, obtida na análise de um modelo;
- σ^2 é a variância dos resíduos;
- h é o termo utilizado como função linear de *Holt-Winters*.

Apêndice B

Códigos para Modelagem e Previsão

```
########################################
#                                      #
#                                      #
#              CÓDIGOS TG              #
#                                      #
#                                      #
########################################

library(readr)
library(ggplot2)
library(dplyr)
library(patchwork)
library(randtests)
library(ggcorrplot)
library(cowplot)
library(fpp2)
library(TSA)
library(tseries)
library(lmtest)
library(reshape)
library(astsa)
library(GGally)
library(forecast)
```

```
library(xgboost)
library(caret)

load("C:/Users/thais/OneDrive/Documentos/TG/Dadostg.RData")

# Provisao_Total
dados_tg$Provisao_Total = as.numeric(dados_tg$Provisao_Total)

dados_tg$Tempo_Mensal = as.numeric(1:nrow(dados_tg))

(options(scipen = 999))

Meses = dados_tg$Tempo_Mensal

#Gráfico da Séries Temporais de Provisão

dados_tg %>% ggplot()+
  geom_line(aes(x = Tempo_Mensal, y = Provisao_Total),color="red")+
  labs(y = "Provisao Total", x = "Tempo Mensal",
       title = "Série Temporal Provisão")+
  theme(plot.title = element_text(hjust = 0.5)) +
  scale_x_continuous(breaks = seq(0, 55, 6))

#Gráfico da Séries Temporais de Empréstimo

dados_tg %>% ggplot()+
  geom_line(aes(x = Tempo_Mensal, y = Emprestimo),color="red")+
  labs(y = "Empréstimo", x = "Tempo Mensal",
       title = "Série Temporal Empréstimo")+
  theme(plot.title = element_text(hjust = 0.5)) +
  scale_x_continuous(breaks = seq(0, 55, 6))

g1 = dados_tg %>% ggplot() +
```

```
  geom_histogram(aes(x = Provisao_Total),color="black",fill="red",
                 bins = 25)+
  labs(x = "Provisao Total", y = "Contagem",
       title = "Histograma da Provisão Total")+
  theme(plot.title = element_text(hjust = 0.5),
        title = element_text(size = 10))+
  scale_x_continuous(breaks = seq(46882790,62333131,5000000))

# Gráfico box plot da variável resposta
g2 = dados_tg %>% ggplot() +geom_boxplot(aes(x=Provisao_Total, y= "Boxplot")
                                         ,color="black",
                                         fill="red")+
  scale_x_continuous(breaks = seq(46882790,62333131,5000000))+
  labs(x = "Provisao Total", y = "",
       title = "Boxplot da Provisão Total")+
  theme(plot.title = element_text(hjust = 0.5),
        title = element_text(size = 10))

g1+g2

#medidas resumo
summary(dados_tg$Provisao_Total)

summary(dados_tg$Emprestimo)

# Gráfico da variável empréstimo

e1 = dados_tg %>% ggplot() +
  geom_histogram(aes(x = Emprestimo),color="black",fill="red",bins = 25)+
  labs(x = "Empréstimo", y = "Contagem",
       title = "Histograma de Empréstimo")+
  theme(plot.title = element_text(hjust = 0.5),
        title = element_text(size = 10))+
```

```
  scale_x_continuous(breaks = seq(40000000,70000000,5000000))

# Gráfico box plot da variávelempréstimo
e2 = dados_tg %>% ggplot() +geom_boxplot(aes(x=Emprestimo, y= "Boxplot"),
                                         color="black",
                                         fill="red")+
  scale_x_continuous(breaks = seq(40000000,70000000,5000000))+
  labs(x = "Empréstimo", y = "",title = "Boxplot de Empréstimo")+
  theme(plot.title = element_text(hjust = 0.5),
        title = element_text(size = 10))

e1+e2


# fazendo grafico media x amplitude (verificar variância constante)
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)])

  data <- data.frame(x.m, x.r)

  ggplot(data , aes(x=x.m, y = x.r,color="red")) +
    geom_point(col = "red",fill = "red" , size = 2 , pch = 1 ) +
    labs(x = "Médias" , y = "Amplitude",
         title = "Amplitude x Médias (Empréstimo)") +

    theme(axis.title.x = element_text(size=rel(1.2) , angle = 0 ) ,
          axis.title.y = element_text (size=rel(1.2) , angle = 90)) +
```

```
    geom_smooth(method="lm" , se = FALSE, col="black")+
    theme(plot.title = element_text(hjust = 0.5))
}

#med.var(dados_tg$Provisao_Total,6)
med.var(dados_tg$Provisao_Total,12)
med.var(dados_tg$Emprestimo,12)

# gráfico de sazonalidade pacote - função do ggplot

ggplot.corr <- function(data, lag.max = 24, ci = 0.95,
                        large.sample.size = TRUE, horizontal = TRUE,...) {

  require(ggplot2)
  require(dplyr)
  require(cowplot)

  if(horizontal == TRUE) {numofrow <- 1} else {numofrow <- 2}

  list.acf <- acf(data, lag.max = lag.max,
                  type = "correlation", plot = FALSE)
  N <- as.numeric(list.acf$n.used)
  df1 <- data.frame(lag = list.acf$lag, acf = list.acf$acf)
  df1$lag.acf <- dplyr::lag(df1$acf, default = 0)
  df1$lag.acf[2] <- 0
  df1$lag.acf.cumsum <- cumsum((df1$lag.acf)^2)
  df1$acfstd <- sqrt(1/N * (1 + 2 * df1$lag.acf.cumsum))
  df1$acfstd[1] <- 0
  df1 <- select(df1, lag, acf, acfstd)

  list.pacf <- acf(data, lag.max = lag.max, type = "partial",
                   plot = FALSE)
  df2 <- data.frame(lag = list.pacf$lag,pacf = list.pacf$acf)
```

```
df2$pacfstd <- sqrt(1/N)

if(large.sample.size == TRUE) {
  plot.acf <- ggplot(data = df1, aes( x = lag, y = acf)) +
    geom_area(aes(x = lag, y = qnorm((1+ci)/2)*acfstd), fill = "red") +
    geom_area(aes(x = lag, y = -qnorm((1+ci)/2)*acfstd),
              fill = "red") +
    geom_col(fill = "red", width = 0.7) +
    scale_x_continuous(breaks = seq(0,max(df1$lag),6)) +
    scale_y_continuous(name = element_blank(),
                       limits = c(min(df1$acf,df2$pacf),1)) +
    ggtitle("ACF") +
    theme(plot.title = element_text(hjust = 0.5))+
    labs(x = "Lag")

  plot.pacf <- ggplot(data = df2, aes(x = lag, y = pacf)) +
    geom_area(aes(x = lag, y = qnorm((1+ci)/2)*pacfstd),
              fill = "red") +
    geom_area(aes(x = lag, y = -qnorm((1+ci)/2)*pacfstd),
              fill = "red") +
    geom_col(fill = "red", width = 0.7) +
    scale_x_continuous(breaks = seq(0,max(df2$lag, na.rm = TRUE),6)) +
    scale_y_continuous(name = element_blank(),
                       limits = c(min(df1$acf,df2$pacf),1)) +
    ggtitle("PACF") +
    theme(plot.title = element_text(hjust = 0.5))+
    labs(x = "Lag")
}
else {
  plot.acf <- ggplot(data = df1, aes( x = lag, y = acf)) +
    geom_col(fill = "red", width = 0.7) +
    geom_hline(yintercept = qnorm((1+ci)/2)/sqrt(N),
               colour = "black",
```

```
                linetype = "dashed") +
    geom_hline(yintercept = - qnorm((1+ci)/2)/sqrt(N),
                colour = "black",
                linetype = "dashed") +
    scale_x_continuous(breaks = seq(0,max(df1$lag),6)) +
    scale_y_continuous(name = element_blank(),
                       limits = c(min(df1$acf,df2$pacf),1)) +
    ggtitle("ACF Empréstimo") +
    theme(plot.title = element_text(hjust = 0.5))+
    labs(x = "Lag")


  plot.pacf <- ggplot(data = df2, aes(x = lag, y = pacf)) +
    geom_col(fill = "red", width = 0.7) +
    geom_hline(yintercept = qnorm((1+ci)/2)/sqrt(N),
                colour = "black",
                linetype = "dashed") +
    geom_hline(yintercept = - qnorm((1+ci)/2)/sqrt(N),
                colour = "black",
                linetype = "dashed") +
    scale_x_continuous(breaks = seq(0,max(df2$lag, na.rm = TRUE),6)) +
    scale_y_continuous(name = element_blank(),
                       limits = c(min(df1$acf,df2$pacf),1)) +
    ggtitle("PACF Empréstimo") +
    theme(plot.title = element_text(hjust = 0.5))+
    labs(x = "Lag")
  }
  cowplot::plot_grid(plot.acf, plot.pacf, nrow = numofrow)
}

ggplot.corr(data = dados_tg$Provisao_Total, lag.max = 55,
            ci= 0.95, large.sample.size = FALSE, horizontal = TRUE)
```

```
ggplot.corr(data = dados_tg$Emprestimo, lag.max = 55, ci= 0.95,
            large.sample.size = FALSE, horizontal = TRUE)


#testando a estacionariedade provisão
adf.test(dados_tg$Provisao_Total)
#p-valor = 0.2824, logo a série não é estacionária

#testando a estacionariedade empréstimo
adf.test(diff(ts(dados_tg$Emprestimo)))
#p-valor = 0.3016, logo a série não é estacionária


#aplicando diferença tamanho 1 na provisão total

plot(diff(ts(dados_tg$Provisao_Total)),col="red",pch=16,
     main = "Diferença Tamanho 1 - Provisão",
     xlab="Tempo Mensal",ylab = "Diferença Provisão 1")

#aplicando diferença tamanho 1 no empréstimo
plot(diff(ts(dados_tg$Emprestimo)),col="red",pch=16,
     main = "Diferença Tamanho 1 - Empréstimo",
     xlab="Tempo Mensal",ylab = "Diferença Empréstimo 1")

#correlação cruzada das séries
ccf(diff(ts(dados_tg$Emprestimo)),diff(ts(dados_tg$Provisao_Total)))
ccf2(diff(ts(dados_tg$Emprestimo)),diff(ts(dados_tg$Provisao_Total)),
     main = "Correlação Cruzada entre Empréstimo e Provisão",
     col="red",gg=T)

#season plot sem tendência provisão
seasonplot(diff(ts(dados_tg$Provisao_Total)),12,col="red",
```

```
           main = "Gráfico Sazonal Mensal - Provisão",
           xlab = "Meses") #escolher esse

#season plot sem tendência empréstimo
seasonplot(diff(ts(dados_tg$Emprestimo)),12,col="red",
           main = "Gráfico Sazonal Mensal - Empréstimo",
           xlab = "Meses") #escolher esse

# notando que agora não existe sazonalidade provisão


ggplot.corr(data = diff(diff(ts(dados_tg$Provisao_Total)),12),
            lag.max = 55, ci= 0.95, large.sample.size = FALSE,
            horizontal = TRUE)

#notando que agora não existe sazonalidade empréstimo
ggplot.corr(data = diff(diff(ts(dados_tg$Emprestimo)),12),
            lag.max = 55, ci= 0.95, large.sample.size = FALSE,
            horizontal = TRUE)

############## construção dos modelos provisão ####################
auto.arima(ts(dados_tg$Provisao_Total))

modelo3 = arima(ts(dados_tg$Provisao_Total),order = c(0,1,0),
                seasonal = c(0,1,1))
coeftest(modelo3)
tsdiag(modelo3)

modelo5 = arima(ts(dados_tg$Provisao_Total),order = c(1,1,1),
                seasonal = c(0,1,1))
coeftest(modelo5)
tsdiag(modelo5)
```

```
#sarima(ts,p,d,q,P,D,Q) #fazer para os modelos 3 e 5, calcula o aic e bic

#modelo 3

modelo3_sar = sarima(ts(dados_tg$Provisao_Total),0,1,0,0,1,1,12)
modelo3_sar

modelo5_sar = sarima(ts(dados_tg$Provisao_Total),1,1,1,0,1,1,12)
modelo5_sar

#modelo 3 é o melhor
Provisao = ts(dados_tg$Provisao_Total)
sarima.for(Provisao,12,0,1,0,0,1,1,12,plot.all = F)
#bic e aic e análise do modelo

# tratando outliers
detectAO(modelo3) #aditivo
detectIO(modelo3) #intervenção

modelo3_a= arimax(dados_tg$Provisao_Total,order=c(0,1,0),
                  seasonal=list(order=c(0,1,1),period=12),
                  io=c(24,36,54))
modelo3_a
tsdiag(modelo3_a)

### Tentando encontrar a previsão do melhor modelo

#criando matriz de dados

io_24 = c(rep(0,23),1,rep(0,(55-24)))
io_24_12 = c(rep(0,23),1,rep(0,(55-24)),rep(0,12))
io_36 = c(rep(0,35),1,rep(0,(55-36)))
```

```
io_36_12 = c(rep(0,35),1,rep(0,(55-36)),rep(0,12))
io_54 = c(rep(0,53),1,0)
io_54_12 = c(rep(0,53),1,0,rep(0,12))

io_s = data.frame(io_24,io_36,io_54)
io_s_12 = data.frame(io_24_12,io_36_12,io_54_12)

modelo_3a_testetot<-arimax(dados_tg$Provisao_Total,
                           order=c(0,1,0),seasonal=list(order=c(0,1,1),
                                                        period=12),
                           xreg = data.frame(io_s))

ap = predict(modelo_3a_testetot,n.ahead = 12,newxreg = data.frame(io_s_12))

#intervalo de confiança

ls = as.list(1:12)
li = as.list(1:12)

for (i in 1:12){
  ls[i] = ap$pred[i]+ap$se[i]*1.96

  li[i] = ap$pred[i]-ap$se[i]*1.96
}

#valores da previsão 12 meses pra frente

provisao_prev = c(dados_tg$Provisao_Total,ap$pred[1:12])
Tempo_mensal_prov_prev = c(dados_tg$Tempo_Mensal,56:67)
provisao_cont_li = c(rep(NA,55),li[[1]],li[[2]],li[[3]],li[[4]],
                     li[[5]],li[[6]],
                     li[[7]],li[[8]],li[[9]],li[[10]],li[[11]],li[[12]])
provisao_cont_ls = c(rep(NA,55),ls[[1]],ls[[2]],ls[[3]],ls[[4]],
```

```
                    ls[[5]],ls[[6]],
                    ls[[7]],ls[[8]],ls[[9]],ls[[10]],ls[[11]],ls[[12]])

graf_prov_prev = data.frame(provisao_prev,Tempo_mensal_prov_prev,
                            provisao_cont_li,
                            provisao_cont_ls)



graf_prov_prev %>% ggplot()+
  geom_line(aes(x = Tempo_mensal_prov_prev, y = provisao_prev),
            color="red")+
  labs(y = "Provisao Total", x = "Tempo Mensal",
       title = "Previsão da Série Temporal Provisão")+
  theme(plot.title = element_text(hjust = 0.5)) +
  scale_x_continuous(breaks = seq(0,67 , 6)) +
  geom_vline(xintercept = 55)+
  geom_line(aes(x = Tempo_mensal_prov_prev, y = provisao_cont_li),
            color="blue")+
  geom_line(aes(x = Tempo_mensal_prov_prev, y = provisao_cont_ls),
            color="blue")


############## construção dos modelos empréstimo ####################

auto.arima(ts(dados_tg$Emprestimo))

modeloe1 = arima(ts(dados_tg$Emprestimo), order = c(0,1,0))
modeloe1


modeloe2 = arima(ts(dados_tg$Emprestimo),order = c(0,1,0),
                 seasonal = c(0,1,1))
```

```
modeloe2

modeloe4 = arima(ts(dados_tg$Emprestimo),order = c(0,1,0),
                 seasonal = c(1,1,0))
modeloe4


modeloe1_sar = sarima(ts(dados_tg$Emprestimo),0,1,0,0,0,0,12)
modeloe1_sar

modeloe2_sar = sarima(ts(dados_tg$Emprestimo),0,1,0,0,1,1,12)
modeloe2_sar

modeloe4_sar = sarima(ts(dados_tg$Emprestimo),0,1,0,1,1,0,12)
modeloe4_sar

#modelo e2 é o melhor
Emprestimo = ts(dados_tg$Emprestimo)
sarima.for(Emprestimo,12,0,1,0,0,1,1,12,plot.all = F)
#bic e aic e análise do modelo

# tratando outliers
detectAO(modeloe2)
detectIO(modeloe2)

modeloe2_a = arimax(dados_tg$Emprestimo,order=c(0,1,0),
                    seasonal=list(order=c(0,1,1),period=12),io=c(24,36,55))
modeloe2_a
tsdiag(modeloe2_a)

#calculando a previsão

io_55 = c(rep(0,54),1)
```

```
io_55_12 = c(rep(0,54),1,rep(0,12))

io_s_emp = data.frame(io_24,io_36,io_55)
io_s_12_emp = data.frame(io_24_12,io_36_12,io_55_12)

modelo_2a_testetot<-arimax(dados_tg$Emprestimo,order=c(0,1,0),
                           seasonal=list(order=c(0,1,1),period=12),
                           xreg = data.frame(io_s_emp))

ap_emp = predict(modelo_2a_testetot,n.ahead = 12,
                 newxreg = data.frame(io_s_12_emp))

#valores da previsão 12 meses pra frente

ls_e = as.list(1:12)
li_e = as.list(1:12)

for (i in 1:12){
  ls_e[i] = ap_emp$pred[i]+ap_emp$se[i]*1.96

  li_e[i] = ap_emp$pred[i]-ap_emp$se[i]*1.96
}

empr_prev = c(dados_tg$Emprestimo,ap_emp$pred[1:12])
Tempo_mensal_empr_prev = c(dados_tg$Tempo_Mensal,56:67)
empr_cont_li = c(rep(NA,55),li_e[[1]],li_e[[2]],li_e[[3]],
                 li_e[[4]],li_e[[5]],li_e[[6]],
                 li_e[[7]],li_e[[8]],li_e[[9]],li_e[[10]],
                 li_e[[11]],li_e[[12]])
empr_cont_ls = c(rep(NA,55),ls_e[[1]],ls_e[[2]],ls_e[[3]],
                 ls_e[[4]],ls_e[[5]],ls_e[[6]],
                 ls_e[[7]],ls_e[[8]],ls_e[[9]],ls_e[[10]],
                 ls_e[[11]],ls_e[[12]])
```

```
graf_empr_prev = data.frame(empr_prev,Tempo_mensal_empr_prev,empr_cont_li,
                            empr_cont_ls)

graf_empr_prev %>% ggplot()+
  geom_line(aes(x = Tempo_mensal_empr_prev, y = empr_prev),
            color="red")+
  labs(y = "Empréstimo", x = "Tempo Mensal",
       title = "Previsão da Série Temporal Empréstimo")+
  theme(plot.title = element_text(hjust = 0.5)) +
  scale_x_continuous(breaks = seq(0,67 , 6)) + geom_vline(xintercept = 55)+
  geom_line(aes(x = Tempo_mensal_prov_prev, y = empr_cont_li),color="blue")+
  geom_line(aes(x = Tempo_mensal_prov_prev, y = empr_cont_ls),color="blue")

################### alisamento exponencial provisao ########################

alis12 = HoltWinters(ts(dados_tg$Provisao_Total,frequency = 12),
                     seasonal = "mult")

const12 = c(alis12$alpha,alis12$beta,alis12$gamma)
const12

coef12 = coefficients(alis12)
coef12

#valores ajustados e componentes

ajust12 = alis12$fitted
ajust12
nivel12     = alis12$fitted[1,2]
tendencia12 = alis12$fitted[1,3]
sazonal12   = alis12$fitted[1,4]
```

```
verif12 = (nivel12 + tendencia12) * sazonal12
verif12 #igual o x chapeu

#equações
dim12 = dim(alis12$fitted)
dim12 #usa t11

#gráficos das séries
par(mfrow = c(1,1))

#plot(alis6)
plot(alis12,main = "Holt-Winters Provisão",ylab = "Observados",
     xlab="Tempo",col = "black")

#nível, tendência e sazonalidade
z12 = ts(cbind(alis12$fitted[,2],alis12$fitted[,3],
               alis12$fitted[,4]),start=c(1,1),freq=12)
plot(z12,main = "Holt-Winters Provisão",ylab = "Observados",
     xlab="Tempo")

#previsão com 12 meses
p12 = predict(alis12, n.ahead=12, prediction.interval = T)
p12
plot(alis12, p12,main = "Previsão Holt-Winters - Provisão",
     ylab = "",xlab="Tempo") #para 12 meses para a frente

################## alisamento exponencial emprestimo ######################

alis_empr_12 = HoltWinters(ts(dados_tg$Emprestimo,frequency = 12),
                           seasonal = "mult")

const_empr_12 = c(alis_empr_12$alpha,alis_empr_12$beta,alis_empr_12$gamma)
```

```
const_empr_12

coef_empr_12 = coefficients(alis_empr_12)
coef_empr_12

ajust_empr_12 = alis_empr_12$fitted
ajust_empr_12


#equações
dim_empr_12 = dim(alis_empr_12$fitted)
dim_empr_12 #usa t11


plot(alis_empr_12,main = "Holt-Winters Empréstimo",
     ylab = "Observados",xlab="Tempo",col = "black")

z12_empr = ts(cbind(alis_empr_12$fitted[,2],alis_empr_12$fitted[,3],
                    alis_empr_12$fitted[,4]),start=c(1,1),freq=12)
plot(z12_empr,main = "Holt-Winters Empréstimo",ylab = "Observados",
     xlab="Tempo")

#previsão com 12 meses
p12_empr = predict(alis_empr_12, n.ahead=12, prediction.interval = T)
p12_empr
plot(alis_empr_12, p12_empr,
     main = "Previsão Holt-Winters - Empréstimo",
     ylab = "",xlab="Tempo") #para 12 meses para a frente

# Fazendo a An?lise para a vari?vel Provis?o

set.seed(25)
```

```
dados_provisao = dados_tg %>% select(DtBase,Provisao_Total)

# Gerando valores para realizar a previs?o de 12 meses para frente

add_dados = rep(NA,12)

data_estendida = c(dados_provisao$DtBase,seq(202108,202112,1),seq(202201,202207,1))

provisao_estendida = c(dados_provisao$Provisao_Total,add_dados)

dados_provisao_estendido = cbind(data_estendida,provisao_estendida)

dados_provisao_estendido = as.data.frame(dados_provisao_estendido)

# Separando os dados em colunas de m?s e ano

dados_provisao_estendido$ano = substr(dados_provisao_estendido$data_estendida,1,4)
dados_provisao_estendido$mes = ifelse(
substr(dados_provisao_estendido$data_estendida,5,6)=="10"|
      substr(dados_provisao_estendido$data_estendida,5,6)=="11"|
      substr(dados_provisao_estendido$data_estendida,5,6)=="12",
      substr(dados_provisao_estendido$data_estendida,5,6),
      substr(dados_provisao_estendido$data_estendida,6,6))

dados_provisao_estendido$ano = as.numeric(dados_provisao_estendido$ano)
dados_provisao_estendido$mes = as.numeric(dados_provisao_estendido$mes)

# Separando os dados em bases de treino e teste

treino = dados_provisao_estendido[1:55,]
pred = dados_provisao_estendido[56:67,]

# Espa?o de par?metros para o ajuste do modelo
```

```
xgb_trcontrol <- caret::trainControl(
  method = "cv",
  number = 5,
  allowParallel = TRUE,
  verboseIter = FALSE,
  returnData = FALSE
)

xgb_grid <- base::expand.grid(
  list(
    nrounds = c(100, 200),
    max_depth = c(10, 15, 20), # maximum depth of a tree
    colsample_bytree = seq(0.5), # subsample ratio of columns when
    #construction each tree
    eta = 0.1, # learning rate
    gamma = 0, # minimum loss reduction
    min_child_weight = 1,  # minimum sum of instance weight (hessian)
    #needed ina child
    subsample = 1 # subsample ratio of the training instances
  ))

#Fazendo o Modelo

model_1 <- train(provisao_estendida~., data =treino[,-1],
                 method = "xgbTree", trControl = xgb_trcontrol,
                 tuneGrid = xgb_grid,nthread = 1)

# melhores valores de hiperpar?metros
model_1$bestTune
summary(model_1)
# previsao
x_pred_data = as.data.frame(pred %>%
```

```
                              dplyr::select(mes,ano))

xgb_pred <- model_1 %>% stats::predict(x_pred_data)
xgb_pred



# Fazendo a An?lise para a vari?vel Empr?stimo

dados_emprestimo = dados_tg %>% select(DtBase,Emprestimo)

# Gerando valores para realizar a previs?o de 12 meses para frente

#add_dados = rep(NA,12)

data_estendida_emp = c(dados_emprestimo$DtBase,seq(202108,202112,1),
                       seq(202201,202207,1))

emprestimo_estendido = c(dados_emprestimo$Emprestimo,add_dados)

dados_emprestimo_estendido = cbind(data_estendida_emp,
                                   emprestimo_estendido)

dados_emprestimo_estendido = as.data.frame(dados_emprestimo_estendido)

# Separando os dados em colunas de m?s e ano

dados_emprestimo_estendido$ano = substr(
dados_emprestimo_estendido$data_estendida_emp,1,4)
dados_emprestimo_estendido$mes = ifelse(
  substr(dados_emprestimo_estendido$data_estendida_emp,5,6)=="10"|
        substr(dados_emprestimo_estendido$data_estendida_emp,5,6)=="11"|
        substr(dados_emprestimo_estendido$data_estendida_emp,5,6)=="12",
```

```
        substr(dados_emprestimo_estendido$data_estendida_emp,5,6),
        substr(dados_emprestimo_estendido$data_estendida_emp,6,6))

dados_emprestimo_estendido$ano = as.numeric(dados_emprestimo_estendido$ano)
dados_emprestimo_estendido$mes = as.numeric(dados_emprestimo_estendido$mes)

# Separando os dados em bases de treino e teste

treino_emp = dados_emprestimo_estendido[1:55,]
pred_emp = dados_emprestimo_estendido[56:67,]

# Espa?o de par?metros para o ajuste do modelo

xgb_trcontrol <- caret::trainControl(
  method = "cv",
  number = 5,
  allowParallel = TRUE,
  verboseIter = FALSE,
  returnData = FALSE
)

xgb_grid <- base::expand.grid(
  list(
    nrounds = c(100, 200),
    max_depth = c(10, 15, 20), # maximum depth of a tree
    colsample_bytree = seq(0.5), # subsample ratio of columns when
    #construction each tree
    eta = 0.1, # learning rate
    gamma = 0, # minimum loss reduction
    min_child_weight = 1,  # minimum sum of instance weight (hessian)
    #needed ina child
    subsample = 1 # subsample ratio of the training instances
  ))
```

```
#Fazendo o Modelo

model_1_emp <- train(emprestimo_estendido~., data =treino_emp[,-1],
                     method = "xgbTree", trControl = xgb_trcontrol,
                     tuneGrid = xgb_grid,nthread = 1)

# melhores valores de hiperpar?metros
model_1_emp$bestTune

# previsao
x_pred_data_emp = as.data.frame(pred_emp %>%
                                  dplyr::select(mes,ano))

xgb_pred_emp <- model_1_emp %>% stats::predict(x_pred_data_emp)
xgb_pred_emp

# Testando o intervalo de confian?a para a vari?vel provisao

predictions <- data.frame(matrix(NA,12,100))

library(xgboost)

##come up with 100 unique seed values that you can reproduce
set.seed(1500)
seeds <- runif(1200,1,1200)

for (i in 1:100){

  # set.seed(seeds[i])
  xgb_model <- train(provisao_estendida~., data =treino[,-1],
                     method = "xgbTree", trControl = xgb_trcontrol,
                     tuneGrid = xgb_grid,nthread = 1, set.seed(seeds[i]))
```

```
  x_pred_data = as.data.frame(pred %>%
                                dplyr::select(mes,ano))
  predictions[,i] <- xgb_model %>% stats::predict(x_pred_data)

}
predictions

#Transformando em banco de dados

dados_preditos = t(predictions)

summary(dados_preditos)

# Testando o intervalo de confian?a para a vari?vel empr?stimo

predictions_emp <- data.frame(matrix(NA,12,100))

library(xgboost)

##come up with 100 unique seed values that you can reproduce
set.seed(1500)
seeds <- runif(1200,1,1200)

for (i in 1:100){

  # set.seed(seeds[i])
  xgb_model_emp <- train(emprestimo_estendido~., data =treino_emp[,-1],
                         method = "xgbTree", trControl = xgb_trcontrol,
                         tuneGrid = xgb_grid,nthread = 1,
                         set.seed(seeds[i]))
  x_pred_data_emp = as.data.frame(pred_emp %>%
                                    dplyr::select(mes,ano))
  predictions_emp[,i] <- xgb_model_emp %>%
```

```
    stats::predict(x_pred_data_emp)

}
predictions_emp

#Transformando em banco de dados

dados_preditos_emp = t(predictions_emp)

summary(dados_preditos_emp)

# Gr?fico de previs?o do XGBoost para a vari?vel provis?o

val_min_pro = c(52658748,52854060,53246076,53158068,50474744,
                55567524,56208256,
                58171384,59690064,61051196,51769776,52658748)
val_pre_pro = c(52666446,52861758,53253774,53165766,50482443,
                55574479,56217460,
                58174794,59701399,61080373,51771724,52658748)
val_max_pro = c(52710068,52905380,53297396,53209388,50526068,
                55613888,56217460,
                58194116,59765628,61245712,51782760,52710068 )

Tempo_pro_xgb_y = c(dados_tg$Provisao_Total,val_pre_pro)
Tempo_pro_xgb_x = c(1:67)
Val_min_pro_xgb = c(rep(NA,55), val_min_pro)
Val_max_pro_xgb = c(rep(NA,55), val_max_pro)

graf_prov_prev_xgb = data.frame(Tempo_pro_xgb_x,Tempo_pro_xgb_y,
                                Val_min_pro_xgb,
                                Val_max_pro_xgb)

(options(scipen = 999))
```

```
graf_prov_prev_xgb %>% ggplot()+
  geom_line(aes(x = Tempo_pro_xgb_x, y = Tempo_pro_xgb_y),color="red")+
  labs(y = "Provisao Total", x = "Tempo Mensal",
       title = "Previs?o da S?rie Temporal Provis?o - XGBOOST")+
  theme(plot.title = element_text(hjust = 0.5)) +
  scale_x_continuous(breaks = seq(0,67 , 6)) +
  geom_vline(xintercept = 55)+
  geom_line(aes(x = Tempo_pro_xgb_x, y = Val_min_pro_xgb),color="blue")+
  geom_line(aes(x = Tempo_pro_xgb_x, y = Val_max_pro_xgb),color="blue")



# Gr?fico de previs?o do XGBoost para a vari?vel emprestimo

val_min_emp = c(57507840,57507840,57510016,58279624,57553092,
                62319736,63232284,
                63453380,62127912,69356760,69356760,57507840)
val_pre_emp = c(57524484,57523583,57542119,58432097,57589264,
                62323158,63246608,
                63501986,62131537,69421995,69401407,57524470)
val_max_emp = c(57534272,57532848,57562116,58521644,57610508,
                62325168,63255020,
                63530532,62137708,69460312,69427640,57534248)

Tempo_emp_xgb_y = c(dados_tg$Emprestimo,val_pre_emp)
Tempo_emp_xgb_x = c(1:67)
Val_min_emp_xgb = c(rep(NA,55), val_min_emp)
Val_max_emp_xgb = c(rep(NA,55), val_max_emp)

graf_empr_prev_xgb = data.frame(Tempo_emp_xgb_x,Tempo_emp_xgb_y,
                                Val_min_emp_xgb,
                                Val_max_emp_xgb)
```

```
(options(scipen = 999))

graf_empr_prev_xgb %>% ggplot()+
  geom_line(aes(x = Tempo_emp_xgb_x, y = Tempo_emp_xgb_y),color="red")+
  labs(y = "Empr?stimo", x = "Tempo Mensal",
       title = "Previs?o da S?rie Temporal Empr?stimo-XGBOOST")+
  theme(plot.title = element_text(hjust = 0.5)) +
  scale_x_continuous(breaks = seq(0,67 , 6)) +
  geom_vline(xintercept = 55)+
  geom_line(aes(x = Tempo_emp_xgb_x, y = Val_min_emp_xgb),color="blue")+
  geom_line(aes(x = Tempo_emp_xgb_x, y = Val_max_emp_xgb),color="blue")




# Previsao separado em treino e teste

# Foi utilizada a divis?o de 80% treino e 20% teste
#(totalizando aproximadamente as 11 observa??es finais)

# Trabalhando com a vari?vel Provis?o

treino_provi = dados_tg %>% filter(DtBase<202008)
teste_provi = dados_tg %>% filter(DtBase>202007)

# Fazendo a previs?o com o modelo arima

############# constru??o dos modelos provis?o ####################

modelo3 = arima(ts(treino_provi$Provisao_Total),
                order = c(0,1,0), seasonal = c(0,1,1))
coeftest(modelo3)
```

```
tsdiag(modelo3)

#modelo 3
modelo3_sar = sarima(ts(treino_provi$Provisao_Total),0,1,0,0,1,1,12)
modelo3_sar

#modelo 3 ? o melhor
Provisao = ts(treino_provi$Provisao_Total)
sarima.for(Provisao,12,0,1,0,0,1,1,12,plot.all = F)

# tratando outliers
detectAO(modelo3) #aditivo
detectIO(modelo3) #interven??o

modelo3_a= arimax(treino_provi$Provisao_Total,
                  order=c(0,1,0),
                  seasonal=list(order=c(0,1,1),period=12),
                  io=c(24,36))
modelo3_a
tsdiag(modelo3_a)

io_24 = c(rep(0,23),1,rep(0,(43-24)))
io_24_12 = c(rep(0,23),1,rep(0,(43-24)),rep(0,12))
io_36 = c(rep(0,35),1,rep(0,(43-36)))
io_36_12 = c(rep(0,35),1,rep(0,(43-36)),rep(0,12))

io_s = data.frame(io_24,io_36)
io_s_12 = data.frame(io_24_12,io_36_12)

modelo_3a_testetot<-arimax(treino_provi$Provisao_Total,
                           order=c(0,1,0),seasonal=list(order=c(0,1,1),period=12),
                           xreg = data.frame(io_s))
```

```
ap = predict(modelo_3a_testetot,n.ahead = 12,
             newxreg = data.frame(io_s_12))

#intervalo de confian?a

ls = as.list(1:12)
li = as.list(1:12)

for (i in 1:12){
  ls[i] = ap$pred[i]+ap$se[i]*1.96

  li[i] = ap$pred[i]-ap$se[i]*1.96
}

#valores da previs?o 12 meses pra frente

provisao_prev = c(treino_provi$Provisao_Total,ap$pred[1:12])
Tempo_mensal_prov_prev = c(1:55)
provisao_cont_li = c(rep(NA,43),li[[1]],li[[2]],li[[3]],
                     li[[4]],li[[5]],li[[6]],
                     li[[7]],li[[8]],li[[9]],li[[10]],li[[11]],li[[12]])
provisao_cont_ls = c(rep(NA,43),ls[[1]],ls[[2]],ls[[3]],
                     ls[[4]],ls[[5]],ls[[6]],
                     ls[[7]],ls[[8]],ls[[9]],ls[[10]],ls[[11]],ls[[12]])
teste_dist_provi = c(rep(NA,43),teste_provi$Provisao_Total)

provisao_geral = dados_tg$Provisao_Total

graf_prov_prev = data.frame(provisao_prev,Tempo_mensal_prov_prev,
                            provisao_cont_li,
                            provisao_cont_ls,teste_dist_provi,
                            provisao_geral,prev_prov_holt,
                            provisao_prev_prev,
```

```
                              prev_prov_xgb)
provisao_prev_prev = c(rep(NA,43),ap$pred[1:12])
################## alisamento exponencial provisao ########################

alis12 = HoltWinters(ts(treino_provi$Provisao_Total,
                        frequency = 12),seasonal = "mult")

const12 = c(alis12$alpha,alis12$beta,alis12$gamma)
const12

coef12 = coefficients(alis12)
coef12

ajust12 = alis12$fitted
ajust12
nivel12     = alis12$fitted[1,2]
tendencia12 = alis12$fitted[1,3]
sazonal12   = alis12$fitted[1,4]
verif12 = (nivel12 + tendencia12) * sazonal12
verif12
dim12 = dim(alis12$fitted)
dim12

#gr?ficos das s?ries
par(mfrow = c(1,1))

#plot(alis6)
plot(alis12,main = "Holt-Winters Provis?o",
     ylab = "Observados",xlab="Tempo",col = "black")

z12 = ts(cbind(alis12$fitted[,2],alis12$fitted[,3],
               alis12$fitted[,4]),start=c(1,1),freq=12)
plot(z12,main = "Holt-Winters Provis?o",ylab = "Observados",xlab="Tempo")
```

```
#previs?o com 12 meses
p12 = predict(alis12, n.ahead=12, prediction.interval = T)
p12
plot(alis12, p12,main = "Previs?o Holt-Winters - Provis?o",
     ylab = "",xlab="Tempo") #para 12 meses para a frente

prev_prov_holt = c(rep(NA,43),50077300,
                   50603645,
                   50621689,
                   51036260,
                   46740413,
                   47319860,
                   48484031,
                   49079017,
                   48796670,
                   49398938,
                   50988844,
                   50565796)

# Teste

comp_prov = graf_prov_prev %>% ggplot(aes(x = Tempo_mensal_prov_prev,
                                          y = provisao_geral,
                                          col="S?rie Original Provis?o"),color="red")+
  geom_line()+
  labs(y = "Provisao Total", x = "Tempo Mensal",
       title = "Previs?o da S?rie Temporal Provis?o",
       color = "Legenda:\n")+
  theme(plot.title = element_text(hjust = 0.5)) +
  scale_x_continuous(breaks = seq(0,67 , 6)) +
  geom_vline(xintercept = 43)+
  geom_line(aes(y = provisao_prev_prev, col="S?rie Modelo ARIMA",
```

```
                color="green"))+
  geom_line(aes(y = prev_prov_holt, col="S?rie Modelo Holt_Winters",
                color="red"))+
  geom_line(aes(y = prev_prov_xgb, col="S?rie Modelo XGBoost",
                color="orange"))

ggsave(comp_prov,filename = "comp_prov.png",
       path = "C:/Users/thais/OneDrive/Documentos/TG/TG2/Imagens_TG2",
       width = 30,
       height = 12.5,
       units = "cm")

# Fazendo a An?lise para a vari?vel Provis?o XGBoost

set.seed(25)

dados_provisao_test = treino_provi %>% select(DtBase,Provisao_Total)

# Gerando valores para realizar a previs?o de 12 meses para frente

add_dados_test = rep(NA,12)

data_estendida_test = c(dados_provisao_test$DtBase,
                        seq(202008,202012,1),seq(202101,202107,1))

provisao_estendida_test = c(dados_provisao_test$Provisao_Total,
                            add_dados_test)

dados_provisao_estendido_test = cbind(data_estendida_test,
                                      provisao_estendida_test)

dados_provisao_estendido_test = as.data.frame(dados_provisao_estendido_test)
```

```
# Separando os dados em colunas de m?s e ano

dados_provisao_estendido_test$ano = substr(
dados_provisao_estendido_test$data_estendida_test,
                                          1,4)
dados_provisao_estendido_test$mes = ifelse(
  substr(dados_provisao_estendido_test$data_estendida_test,5,6)=="10"|
        substr(dados_provisao_estendido_test$data_estendida_test,5,6)=="11"|
        substr(dados_provisao_estendido_test$data_estendida_test,5,6)=="12",
        substr(dados_provisao_estendido_test$data_estendida_test,5,6),
        substr(dados_provisao_estendido_test$data_estendida_test,6,6))

dados_provisao_estendido_test$ano = as.numeric(dados_provisao_estendido_test$ano)
dados_provisao_estendido_test$mes = as.numeric(dados_provisao_estendido_test$mes)

# Separando os dados em bases de treino e teste

treino_test = dados_provisao_estendido_test[1:43,]
pred_test = dados_provisao_estendido_test[44:55,]

# Espa?o de par?metros para o ajuste do modelo

xgb_trcontrol <- caret::trainControl(
  method = "cv",
  number = 5,
  allowParallel = TRUE,
  verboseIter = FALSE,
  returnData = FALSE
)

xgb_grid <- base::expand.grid(
  list(
    nrounds = c(100, 200),
```

```
    max_depth = c(10, 15, 20), # maximum depth of a tree
    colsample_bytree = seq(0.5), # subsample ratio of columns when
    #construction each tree
    eta = 0.1, # learning rate
    gamma = 0, # minimum loss reduction
    min_child_weight = 1,  # minimum sum of instance weight (hessian)
    #needed ina child
    subsample = 1 # subsample ratio of the training instances
  ))

#Fazendo o Modelo

model_1_test <- train(provisao_estendida_test~., data =treino_test[,-1],
                      method = "xgbTree", trControl = xgb_trcontrol,
                      tuneGrid = xgb_grid,nthread = 1)

# melhores valores de hiperpar?metros
model_1_test$bestTune

# previsao
x_pred_data_test = as.data.frame(pred_test %>%
                                   dplyr::select(mes,ano))

xgb_pred_test <- model_1_test %>% stats::predict(x_pred_data_test)
xgb_pred_test

prev_prov_xgb = c(rep(NA,43),xgb_pred_test)

#calculando o erro absoluto de cada um

provisao_erro = teste_provi$Provisao_Total

erro_absoluto_provisao_arima = abs(sum(provisao_erro-ap$pred[1:12]))
```

```
erro_relativo_provisao_arima = sum(erro_absoluto_provisao_arima/provisao_erro)

erro_absoluto_provisao_holt = abs(sum(provisao_erro-prev_prov_holt[44:55]))
erro_relativo_provisao_holt = sum(erro_absoluto_provisao_holt/provisao_erro)

erro_absoluto_provisao_xgb = abs(sum(provisao_erro-xgb_pred_test))
erro_relativo_provisao_xgb = sum(erro_absoluto_provisao_xgb/provisao_erro)



#### 'Previsao para empr?stimo ###-----------------------------------------------------

auto.arima(ts(treino_provi$Emprestimo))

modeloe2 = arima(ts(treino_provi$Emprestimo),order = c(0,1,0),
                 seasonal = c(0,1,1))
modeloe2

modeloe2_sar = sarima(ts(treino_provi$Emprestimo),0,1,0,0,1,1,12)
#melhor at? o momento
modeloe2_sar

Emprestimo = ts(treino_provi$Emprestimo)
sarima.for(Emprestimo,12,0,1,0,0,1,1,12,plot.all = F)

# tratando outliers
detectAO(modeloe2)
detectIO(modeloe2)

modeloe2_a = arimax(treino_provi$Emprestimo,order=c(0,1,0),
                    seasonal=list(order=c(0,1,1),period=12),io=c(24,36))
modeloe2_a
tsdiag(modeloe2_a)
```

```
#calculando a previs?o


io_s_emp = data.frame(io_24,io_36)
io_s_12_emp = data.frame(io_24_12,io_36_12)

modelo_2a_testetot<-arimax(treino_provi$Emprestimo,
                           order=c(0,1,0),seasonal=list(order=c(0,1,1),
                                                        period=12),
                           xreg = data.frame(io_s_emp))

ap_emp = predict(modelo_2a_testetot,n.ahead = 12,
                 newxreg = data.frame(io_s_12_emp))

#valores da previs?o 12 meses pra frente

ls_e = as.list(1:12)
li_e = as.list(1:12)

for (i in 1:12){
  ls_e[i] = ap_emp$pred[i]+ap_emp$se[i]*1.96

  li_e[i] = ap_emp$pred[i]-ap_emp$se[i]*1.96
}

empr_prev = c(treino_provi$Emprestimo,ap_emp$pred[1:12])
Tempo_mensal_empr_prev = c(1:55)
empr_cont_li = c(rep(NA,43),li_e[[1]],li_e[[2]],li_e[[3]],
                 li_e[[4]],li_e[[5]],li_e[[6]],
                 li_e[[7]],li_e[[8]],li_e[[9]],li_e[[10]],li_e[[11]],
                 li_e[[12]])
empr_cont_ls = c(rep(NA,43),ls_e[[1]],ls_e[[2]],ls_e[[3]],ls_e[[4]],
```

```
                ls_e[[5]],ls_e[[6]],
                ls_e[[7]],ls_e[[8]],ls_e[[9]],ls_e[[10]],ls_e[[11]],
                ls_e[[12]])
empr_prev_prev = c(rep(NA,43),ap_emp$pred[1:12])
#-------------------------------------------------------------------------------
################## alisamento exponencial emprestimo ######################

alis_empr_12 = HoltWinters(ts(treino_provi$Emprestimo,frequency = 12),
                           seasonal = "mult")

const_empr_12 = c(alis_empr_12$alpha,alis_empr_12$beta,
                  alis_empr_12$gamma)
const_empr_12

coef_empr_12 = coefficients(alis_empr_12)
coef_empr_12

ajust_empr_12 = alis_empr_12$fitted
ajust_empr_12



#equa??es
dim_empr_12 = dim(alis_empr_12$fitted)
dim_empr_12 #usa t11



plot(alis_empr_12,main = "Holt-Winters Empr?stimo",
     ylab = "Observados",xlab="Tempo",col = "black")

z12_empr = ts(cbind(alis_empr_12$fitted[,2],alis_empr_12$fitted[,3],
                    alis_empr_12$fitted[,4]),start=c(1,1),freq=12)
plot(z12_empr,main = "Holt-Winters Empr?stimo",ylab = "Observados",
     xlab="Tempo")
```

```
#previs?o com 12 meses
p12_empr = predict(alis_empr_12, n.ahead=12, prediction.interval = T)
p12_empr
plot(alis_empr_12, p12_empr,
     main = "Previs?o Holt-Winters - Empr?stimo",ylab = "",xlab="Tempo")

#para 12 meses para a frente

pre_emp_holt = c(rep(NA,43),51562758,
                 52178804,
                 53632169,
                 51201291,
                 42918331,
                 47073187,
                 49383096,
                 50918418,
                 51826718,
                 53124374,
                 52758033,50859639)

# XGBoost do empr?stimo

set.seed(1500)

dados_emprestimo_test = treino_provi %>% select(DtBase,Emprestimo)

# Gerando valores para realizar a previs?o de 12 meses para frente

#add_dados = rep(NA,12)

data_estendida_emp_test = c(dados_emprestimo_test$DtBase,
                            seq(202008,202012,1),seq(202101,202107,1))
```

```
emprestimo_estendido_test = c(dados_emprestimo_test$Emprestimo,
                              add_dados_test)

dados_emprestimo_estendido_test = cbind(data_estendida_emp_test,
                                        emprestimo_estendido_test)

dados_emprestimo_estendido_test = as.data.frame(dados_emprestimo_estendido_test)

# Separando os dados em colunas de m?s e ano

dados_emprestimo_estendido_test$ano = substr(
dados_emprestimo_estendido_test$data_estendida_emp_test,1,4)
dados_emprestimo_estendido_test$mes = ifelse(
  substr(dados_emprestimo_estendido_test$data_estendida_emp_test,5,6)=="10"|
          substr(dados_emprestimo_estendido_test$data_estendida_emp_test,5,6)=="11"|
          substr(dados_emprestimo_estendido_test$data_estendida_emp_test,5,6)=="12",
          substr(dados_emprestimo_estendido_test$data_estendida_emp_test,5,6),
          substr(dados_emprestimo_estendido_test$data_estendida_emp_test,6,6))

dados_emprestimo_estendido_test$ano = as.numeric(dados_emprestimo_estendido_test$ano)
dados_emprestimo_estendido_test$mes = as.numeric(dados_emprestimo_estendido_test$mes)

# Separando os dados em bases de treino e teste

treino_emp_test = dados_emprestimo_estendido_test[1:43,]
pred_emp_test = dados_emprestimo_estendido_test[44:55,]

# Espa?o de par?metros para o ajuste do modelo

xgb_trcontrol <- caret::trainControl(
  method = "cv",
  number = 5,
```

```
  allowParallel = TRUE,
  verboseIter = FALSE,
  returnData = FALSE
)

xgb_grid <- base::expand.grid(
  list(
    nrounds = c(100, 200),
    max_depth = c(10, 15, 20), # maximum depth of a tree
    colsample_bytree = seq(0.5), # subsample ratio of columns when
    #construction each tree
    eta = 0.1, # learning rate
    gamma = 0, # minimum loss reduction
    min_child_weight = 1,  # minimum sum of instance weight (hessian)
    #needed ina child
    subsample = 1 # subsample ratio of the training instances
  ))

#Fazendo o Modelo

model_1_emp_test <- train(emprestimo_estendido_test~.,
                          data =treino_emp_test[,-1],
                          method = "xgbTree", trControl = xgb_trcontrol,
                          tuneGrid = xgb_grid,nthread = 1)

# melhores valores de hiperpar?metros
model_1_emp_test$bestTune

# previsao
x_pred_data_emp_test = as.data.frame(pred_emp_test %>%
                                       dplyr::select(mes,ano))

xgb_pred_emp_test <- model_1_emp_test %>%
```

```
  stats::predict(x_pred_data_emp_test)
xgb_pred_emp_test_EMP = c(rep(NA,43),xgb_pred_emp_test)

emprestimo_erro = teste_provi$Emprestimo

emprestimo_geral = dados_tg$Emprestimo

graf_EMP_prev = data.frame(Tempo_mensal_prov_prev,emprestimo_geral,
                           empr_prev_prev,pre_emp_holt,
                           xgb_pred_emp_test_EMP)

comp_emp = graf_EMP_prev %>% ggplot(aes(x = Tempo_mensal_prov_prev,
                                        y = emprestimo_geral,
                                        col="S?rie Original Empr?stimo"),color="red")+
  geom_line()+
  labs(y = "Empr?stimo", x = "Tempo Mensal",
       title = "Previs?o da S?rie Temporal Empr?stimo",
       color = "Legenda:\n")+
  theme(plot.title = element_text(hjust = 0.5)) +
  scale_x_continuous(breaks = seq(0,67 , 6)) +
  geom_vline(xintercept = 43)+
  geom_line(aes(y = empr_prev_prev, col="S?rie Modelo ARIMA",
                color="green"))+
  geom_line(aes(y = pre_emp_holt, col="S?rie Modelo Holt_Winters",
                color="red"))+
  geom_line(aes(y = xgb_pred_emp_test_EMP,
                col="S?rie Modelo XGBoost",color="orange"))

ggsave(comp_emp,filename = "comp_emp.png",
       path = "C:/Users/thais/OneDrive/Documentos/TG/TG2/Imagens_TG2",
       width = 30,
       height = 12.5,
       units = "cm")
```

```
erro_absoluto_emprestimo_arima = abs(sum(emprestimo_erro-ap_emp$pred[1:12]))
erro_relativo_emprestimo_arima = sum(erro_absoluto_emprestimo_arima/emprestimo_erro

erro_absoluto_emprestimo_holt = abs(sum(emprestimo_erro-pre_emp_holt[44:55]))
erro_relativo_emprestimo_holt = sum(erro_absoluto_emprestimo_holt/emprestimo_erro)

erro_absoluto_emprestimo_xgb = abs(sum(emprestimo_erro-xgb_pred_emp_test))
erro_relativo_emprestimo_xgb = sum(erro_absoluto_emprestimo_xgb/emprestimo_erro)
```

Referências Bibliográficas

Alice (2020). Introdução ao r: primeiros passos com o rstudio. Disponível em: < `http://datasideoflife.com/?p=1009`. Acessado em: 19 mar 2022.

Almeida, A. Q., Raisen, V. A., Depizzol, D. B. e dos Santos, R. (2007). Modelagem de *Outliers* em séries temporais de vazão máxima do rio jucu, domingos martins, es. *XVII Simpósio Brasileiro de Recursos Hídricos*.

Burba, D. (2018). An overview of time series forecasting models.

Carlini, L. (2021). Introdução ao r: primeiros passos com o rstudio.

Costa, G. F. (2015). Previsão multi-passos em séries temporais: Estratégias clássicas e de aprendizagem automática. *Tese (Mestrado em Anĺise de Dados e Sistemas de Apoio á Decisão)*.

Ferreira, P. C., Junior, J. L. G. e Mattos, D. M. (2015). X-13arima-seats com r: Um estudo de caso para a produção industrial brasileira. *Instituto Brasileiro de Economia*.

Hyndman, R. J. e Athanasopoulos, G. (2018). *Forecasting: principles and practice*. OTexts: Melbourne, Australia. OTexts.com/fpp2. Accessed on 31/08/2021.

Hyndman, R. J. e Athanasopoulos, G. (2021). *Forecasting: principles and practice*. OTexts: Melbourne, Australia. OTexts.com/fpp3. Accessed on 20/10/2021.

Izbicki, R. e dos Santos, T. M. (2020). *Aprendizado de máquina: uma abordagem estatística*. ISBN 978-65-00-02410-4.

Kuhn, M. (2008). Building predictive models in r using the caret package. *Journal of statistical software*, **28**, 1–26.

Medeiros, A. S. V. d. (2021). *Estudo sobre o uso de análise técnica e XGBoost em operações de day-trade*. B.S. thesis, Universidade Federal do Rio Grande do Norte.

Montes, P. S. (2015). Aspectos jurídicos dos consórcios no brasil. *Artigo científico. Disponível: http://www. franca. unesp. br/artigos2008/priscila% 20silva% 20montes. pdf. Acesso em*, **10**(11).

Morettin, P. A. e Toloi, C. M. C. (2006). *Análise de Séries Temporais*. Egard Blusher, São Paulo. ISBN 978-85-212-0389-6.

Neves, E. C. (2019). Disponível em: < `https://medium.com/turing-talks/turing-talks-24-modelos-de-predi\%C3\%A7\%C3\%A3o-ensemble-learning-aa02ce01afda`.

Nielsen, A. (2019). *Practical Time Series Analysis*. ISBN 9781492041658.

Pavlyshenko, B. M. (2018). Machine-learning models for sales time series forecasting. *Using Stacking Approaches for Machine Learning Models*.

Pereira, M. B. (2010). Teste de dickey-fuller robusto baseado nos ranks para séries temporais com observações atípicas.

Receive (2021). Loss given defaut. Disponível em: < `http://datasideoflife.com/?p=1009`. Acessado em: 01 mai 2022.

Rocca, J. (2019). Ensemble methods: bagging, boosting and stacking. *medium-towards data science*.