Otimização da capacidade de previsão de demanda em uma empresa do setor alimentício por meio de modelagem estatística

Trabalho de Diplomação em Engenharia Física II

Universidade Federal do Rio Grande do Sul - Instituto de Física - Escola de Engenharia

Gabriel Isoton De David

Professor Orientador: Gustavo De Medeiros Azevedo

SUMÁRIO

1 INTRODUÇÃO

A evolução da ciência fez com que o poder preditivo do ser humano tivesse uma evolução drástica nos últimos séculos, desde prever o movimento das marés e dos astros antigamente, evoluindo até o atual momento em que a previsão é uma ferramenta crucial na engenharia e no mundo dos negócios. Ela permite o planejamento eficiente, a tomada de decisões informadas, a otimização de recursos e processos, a gestão de riscos e o desenvolvimento de novas tecnologias. Além disso, na engenharia ela é muito utilizada para prever a vida útil e o tempo de falha de máquinas e componentes eletrônicos, assim como na redução de ruído e melhoria da qualidade de processamento de sinais de áudio, imagens, comunicação e sensoriamento. A utilização de modelos estatísticos de previsão contribui para a eficiência e o sucesso dos projetos, garantindo uma alocação adequada de recursos, a mitigação de riscos e a melhoria contínua das estratégias e processos para o gerenciamento e administração de companhias, desde o planejamento de sua produção e estoque, até o investimento de capital da empresa.

A previsão de demanda especificamente, é um aspecto crucial para o sucesso de qualquer empresa que dependa de uma cadeia de suprimentos eficiente. Esta previsão permite que a empresa estime a quantidade de insumos e serviços que é necessária para atender às necessidades futuras dos clientes e planejar sua cadeia produtiva, reduzindo o risco do chamado *Bullwhip Effect* (**??**).

O *Bullwhip Effect* (Efeito chicote em português), é um fenômeno que ocorre quando há variações na demanda ao longo da cadeia de suprimentos que se amplificam conforme se aproximam dos fornecedores. Ou seja, pequenas flutuações na demanda do consumidor podem resultar em grandes variações nos estoques dos varejistas, levando a problemas como excesso ou falta de produtos para os fabricantes, o que pode acarretar no aumento do custo de insumos e estocagem, bem como a perda de eficiência produtiva.

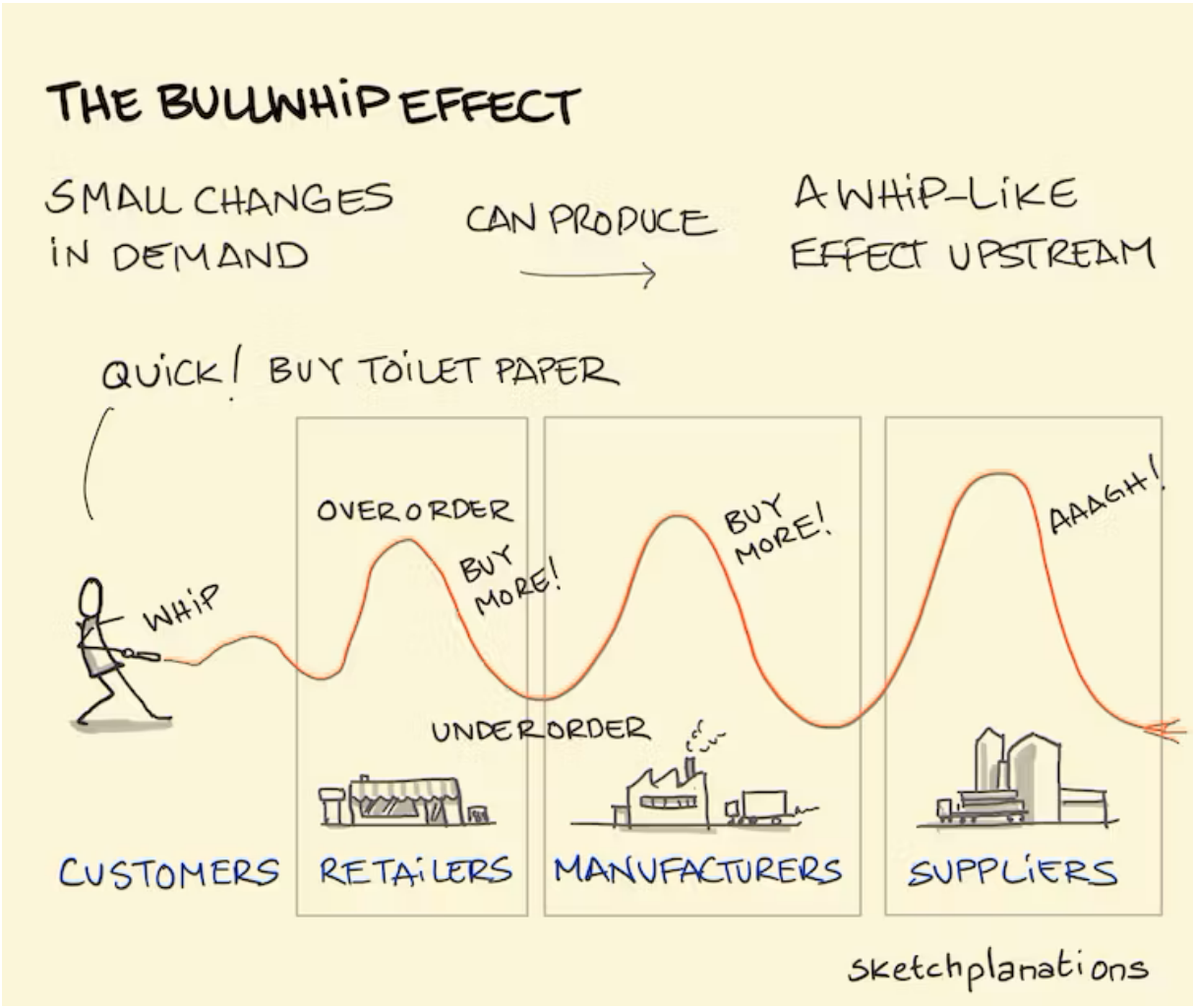


Figura 1 – BullWhip Effect (fonte: (**??**))

No setor alimentício, a demanda é influenciada por vários fatores, como sazonalidade, mudanças no comportamento do consumidor, oscilações do preço próprio e da concorrência que podem agravar o *bullwhip effect*. A capacidade de prever a demanda futura com precisão é, portanto, ainda mais crítica para garantir que a empresa possa entregar o produto certo na hora certa para seus clientes.

Este trabalho, realizado em conjunto com a empresa Aprix (**??**), tem como objetivo explorar a otimização da capacidade preditiva de demanda, através de modelos matemáticos, em um software de precificação desenvolvido para uma empresa do setor alimentício. Serão utilizadas técnicas avançadas de análise de dados para aprimorar a precisão do modelo de previsão de demanda existente, permitindo uma gestão mais eficiente da cadeia de suprimentos desse cliente, e trazendo melhores insumos para uma tomada de decisão de preços em seus produtos.

2 REFERENCIAL TEÓRICO

Nessa seção serão apresentados e discutidos os principais conceitos presentes na literatura sobre os temas relacionados à previsão de demanda e modelagem estatística, que serão necessários para o desenvolvimento do projeto.

2.1 Previsão

Como citado no livro (**??**):

"A necessidade de previsão está aumentando à medida que a gestão tenta diminuir sua dependência do acaso e se torna mais científica ao lidar com seu ambiente. Como cada área de uma organização está relacionada a todas as outras, uma previsão boa ou ruim pode afetar toda a organização"

A capacidade de prever eventos sempre foi muito almejada na história da humanidade nos mais diversos âmbitos, desde negócios e finanças até ciências sociais e físicas. Quando conseguimos identificar padrões é possível tomar medidas possíveis para mitigar possíveis riscos ou identificar oportunidades futuras.

Segundo (**??**) existem três principais motivos para que métodos de previsão sejam utilizados nas decisões gerenciais de grandes empresas para que elas atinjam seus objetivos:

1. Planejamento: O uso eficiente de recursos requer o planejamento de produção, transporte, dinheiro, mão de obra, entre outros. Previsões do nível de demanda por produto, material, trabalho, financiamento ou serviço são um input essencial para esse planejamento.

2. Adquirir recursos: O tempo necessário para adquirir matérias-primas, contratar pessoal ou comprar máquinas e equipamentos pode variar de alguns dias a vários anos. A previsão é necessária para determinar os requisitos futuros de recursos.

3. Determinar os requisitos de recursos: todas as organizações devem determinar quais recursos desejam ter a longo prazo. Tais decisões dependem de oportunidades de mercado, fatores ambientais e do desenvolvimento interno de recursos financeiros, humanos, de produto e tecnológicos. Essas determinações exigem boas previsões e gestores capazes de interpretar as previsões e tomar decisões apropriadas.

2.2 Variáveis

Cada conjunto de dados tem suas próprias características e complexidades, por isso será realizada uma análise cuidadosa dos dados disponíveis, envolvendo: identificar valores ausentes, *outliers* e quais variáveis são mais relevantes para a previsão.

Além disso, é importante garantir que o conjunto de dados seja grande o suficiente para treinar o modelo e validar as previsões. Um erro comum nesse tipo de análise é o *overfitting*,

que ocorre quando o modelo se ajusta demais procurando padrões no conjunto de dados de treinamento, e acaba encontrando padrões, inclusive, no seu erro intrínseco associado. Isso torna o modelo excessivamente complexo e não generalizando bem para novos dados que forem inseridos. Portanto, é importante encontrar um equilíbrio entre a complexidade do modelo e sua capacidade de previsão para novos dados que forem adicionados (**??**).

Variáveis podem ser classificadas como quantitativas ou categóricas, variáveis quantitativas são aquelas que estão associadas a um valor, como por exemplo, a altura de uma pessoa, enquanto as categóricas são aquelas utilizadas para caracterizar algo dentre *n* possibilidades, tais como o estado matrimonial de uma pessoa.

Problemas com respostas categóricas são chamados de problemas de classificação, enquanto problemas que envolvem respostas quantitativas são chamados de problemas de regressão (**??**), como é o caso do problema abordado nesse trabalho.

2.2.1 Elasticidade

A elasticidade entre demanda e preço é uma medida que quantifica a sensibilidade da quantidade demandada de um bem ou serviço às mudanças em seu preço. Ela descreve o quanto a demanda (quantidade procurada) de um produto se altera em resposta a variações no preço desse produto. A fórmula geral para calcular a elasticidade preço da demanda é:

$$\text{Elasticidade Preço da Demanda} = \frac{\%\text{ de Variação na Quantidade Demandada}}{\%\text{ de Variação no Preço}} \tag{1}$$

A elasticidade é frequentemente expressa como um número negativo, pois reflete a relação inversa típica entre preço e quantidade demandada: quando o preço aumenta, a quantidade demandada tende a diminuir, e vice-versa. No entanto, o valor absoluto da elasticidade é importante, pois indica a magnitude da resposta da demanda às mudanças de preço

2.3 Modelos de previsão

Físicos têm utilizado técnicas de modelagem há muito tempo, ajustando equações matemáticas, ou modelos estatísticos, aos dados experimentais para descrever fenômenos físicos e as relações entre as variáveis presentes naquele experimento. Na física, essa técnica é chamada de *fitting*, o seu objetivo é encontrar os parâmetros ideais de uma curva que descreva e minimize a diferença entre os valores observados e os valores previstos pelo modelo, a técnica mais utilizada para esse propósito é o método de mínimos quadrados. Esse método busca minimizar a soma dos quadrados das diferenças entre os valores previstos pelo modelo e os dados observados.

Na ciência de dados, o processo de *fitting* é semelhante, os cientistas de dados utilizam algoritmos e técnicas estatísticas para ajustar modelos aos dados e encontrar os melhores parâmetros que descrevam os padrões ou relacionamentos presentes no histórico de dados.

Podemos dizer que, o *fitting* utilzado pelos físicos há tanto tempo é o precursor da atual ciência de dados, visto que, ambos os campos compartilham o objetivo comum de extrair conhecimento útil dos dados por meio da modelagem e ajuste de modelos estatísticos.

Segundo (**??**), a premissa básica que se deve ter em mente quando realiza-se uma previsão é a de que existe um padrão nos dados disponíveis e que ele provavelmente continuará no futuro, portanto, a capacidade de uma técnica específica de modelagem estatística fornecer uma boa previsão depende, em grande parte, de combinar o padrão dos dados com a técnica que melhor possa lidar com ele.

Quando desejamos realizar a previsão de dados qualitativos existem dois grandes tipos de modelização, os modelos explicativos e os modelos de séries temporais:

2.3.1 Modelos explicativos

Os modelos explicativos tem a premissa de que a variável a ser prevista (dependente) pode ser explicada por uma ou mais variáveis independentes, como por exemplo, a relação entre o tempo de estudo dos alunos e o resultado deles em uma prova. Para quantificar essa relação é preciso entender o conceito de correlação.

Correlação é uma medida estatística que indica a força e a direção da relação entre duas variáveis. Esse coeficiente varia de -1 a 1, indicando uma correlação negativa, neutra ou positiva. Um coeficiente de correlação de -1 indica uma correlação perfeitamente negativa, enquanto um coeficiente de correlação de 1 indica uma correlação perfeitamente positiva. Um coeficiente de correlação de 0 indica uma correlação neutra entre as variáveis.

No exemplo citado, podemos afirmar que a correlação entre o tempo de estudo e o resultado obtido nas provas é positiva, dado que a nota da avaliação (variável dependente), está diretamente ligada ao tempo que o aluno se dedicou aos estudos (variável independente).

O objetivo dos modelos explicativos é descobrir como e quanto essas variáveis estão correlacionadas, ou seja, quanto as mudanças que ocorrem nos dados de entrada influenciarão os dados de saída e usar essa informação para ajudar na previsão da variável dependente (**??**). Ou como no exemplo, quanto o tempo de estudo está relacionado a nota obtida pelo aluno, com um modelo preciso o suficiente, seria possível determinar exatamente quantas horas de estudo são necessárias para que um aluno tire nota 9 na avaliação.

Dentro dessa categoria de modelos explicativos, os mais utilizados são:

1. **Regressão Linear:** é um modelo que estabelece uma relação linear entre uma variável dependente e uma variável independente. Analisa qual a correlação entre as duas variáveis.

2. **Regressão Múltipla:** é uma extensão do modelo de regressão linear, é utilizado quando há mais de uma variável independente que pode influenciar a variável dependente. Exemplo: influência do tempo de estudo e horas dormidas na nota da avaliação.

3. **Modelos de Árvore de Decisão:** são modelos que utilizam uma estrutura de árvore para identificar as variáveis mais importantes na previsão do comportamento de uma variável independente e tomar decisões com base nesses resultados. É amplamente utilizado em problemas de classificação e previsão.

4. **Redes Neurais Artificiais:** são modelos que utilizados para identificar padrões em grandes conjuntos de dados e prever o comportamento de uma variável em problemas complexos.

2.3.2 Modelos de séries temporais

Modelos de séries temporais são ferramentas estatísticas utilizadas para analisar e prever o comportamento de conjuntos de dados sequenciais ao longo do tempo. Essa previsão se baseia nos valores passados daquela variável e o objetivo desses métodos é tentar prever o futuro utilizando os padrões de comportamento histórico dos dados.

Na categoria de modelos de séries temporais, os mais utilizados são:

1. **ARIMA (*Autoregressive Integrated Moving Average*):** é um modelo estatístico que busca prever o comportamento de uma variável ao longo do tempo, levando em consideração padrões e tendências históricas. É amplamente utilizado em finanças, economia e outras áreas que lidam com dados de séries temporais.

2. **Modelos de Suavização Exponencial:** são modelos que buscam prever o comportamento de uma variável, levando em consideração a média móvel ponderada dos valores históricos. É utilizado em problemas de previsão de curto prazo.

3. **SARIMA (*Seasonal Autoregressive Integrated Moving Average*):** é uma extensão do modelo ARIMA que leva em consideração a sazonalidade dos dados. É utilizado em problemas que envolvem dados periódicos, como vendas de produtos sazonais tais como frutas ou dados climáticos.

4. **ARIMAX (*Autoregressive Inegrated Moving Average with eXogenous inputs*):** este modelo combina a análise de séries temporais ARIMA com a análise de regressão linear adicionando a análise de variáveis independentes no método de previsão.

5. **SARIMAX (*Seasonal Autoregressive Inegrated Moving Average with eXogenous inputs*):** este modelo combina a análise de séries temporais sazonais SARIMA com a análise de regressão linear adicionando variáveis independentes no método de previsão.

2.4 Métodos de avaliação

Para avaliar o desempenho de um modelo estatístico de previsão é necessário quantificar quão próxima é a resposta da previsão do modelo do valor real da observação, ou seja, precisamos medir o erro relativo associado à essa previsão.

2.4.1 Erro Quadrático Médio

Para realizar essa análise existem diversos métodos estatísticos, no caso das regressões lineares, o método mais comum é o Erro Quadrático Médio (MSE) que fornece a dispersão dos erros absolutos da previsão, cuja fórmula é:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \widehat{Y}_i)^2 \tag{2}$$

Na qual,

- Y_i: É o valor real para a observação i. Representa os valores reais da variável de interesse que está sendo prevista.
- $\widehat{Y}_i$: É o valor previsto para a observação i. Representa as previsões feitas pelo modelo para a variável de interesse.
- n: É o número total de observações. Representa o tamanho da amostra, ou seja, a quantidade de observações disponíveis para calcular o MSE.

Um ponto a ser considerado, é o de que a unidade de medida do MSE é o quadrado da unidade de medida original, o que pode ser contraintuitivo na hora da interpretação do resultado. Para mitigar esse fator podemos extrair a raiz quadrada do MSE, essa nova medida de erro leva o nome de Raiz do Erro Médio Quadrático (RMSE).

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(Y_i - \widehat{Y}_i)^2} \tag{3}$$

Ambas essas abordagens de avaliação tem algumas características específicas. Por se tratarem de erros quadráticos, eles são mais sensíveis à valores extremos na previsão, quanto maior o erro da previsão, mais penalizado ele será no cálculo do MSE e do RMSE, o que faz com que esses métodos se tornem extremamente sensíveis a *outliers*, que segundo a definição de (**??**) são: *"Observações que diferem substancialmente do valor esperado dado um modelo da situação. Outliers podem ser identificados de forma subjetiva ou por desvios estatisticamente significativos"*.

Outra fonte de erro de interpretação que pode ocorrer da utilização desses métodos é a escala das variáveis envolvidas nos modelos, por isso é importante normalizar as variáveis para evitar distorções nessas métricas

2.4.2 Erro Percentual Absoluto Médio

Um método comumente utilizado na literatura para avaliar a assertividade de um modelo, sem necessitar que as varáveis sejam normalizadas, é o Erro Percentual Absoluto Médio (MAPE).

O MAPE é uma medida relativa do erro médio absoluto, expresso como uma porcentagem em relação aos valores reais observados. É útil para avaliar a precisão percentual da previsão de diferentes modelos ou séries de dados sem depender da unidade de medida. Ele é calculado como a média dos erros absolutos percentuais entre os valores reais e previstos, sua fórmula é:

$$MAPE = \frac{100}{n} \sum_{i=1}^{n} \left| \frac{Y_i - \hat{Y}_i}{Y_i} \right| \tag{4}$$

Onde:

- Y_i é o valor real da variável de interesse para a observação i.
- $\hat{Y}_i$ é o valor previsto da variável de interesse para a observação i.
- n é o número total de observações.

Por causa da divisão, o cálculo do MAPE pode apresentar problemas quando os valores reais observados (Y_i) forem próximos de zero.

2.4.3 Coeficiente de determinação (R^2)

O coeficiente de determinação, representado como R^2, é uma métrica estatística usada para avaliar o quão bem um modelo de regressão se ajusta aos dados. Ele varia de 0 a 1 e indica a proporção da variação na variável dependente que é explicada pelas variáveis independentes do modelo. Um valor de R^2 igual a 1 indica um ajuste perfeito do modelo aos dados, enquanto um valor de 0 indica que o modelo não explica nenhuma variação nos dados. Em geral, quanto mais próximo R^2 estiver de 1, melhor o modelo se ajusta aos dados, mas é importante considerar outras métricas e avaliar o contexto do problema ao interpretar o R^2.

$$R^2 = 1 - \frac{SSR}{SST} \tag{5}$$

Onde:

- R^2: é o coeficiente de determinação, que representa a proporção da variabilidade explicada pelas variáveis independentes em um modelo de regressão.
- SSR: é a Soma dos Quadrados dos Resíduos, que representa a quantidade de variabilidade não explicada pelo modelo.

- SST: é a Soma Total dos Quadrados, que representa a variabilidade total na variável dependente. Ela representa a soma dos quadrados das diferenças entre cada valor observado e a média dos valores observados.

3 METODOLOGIA

Segundo um estudo realizado por (**??**), existem sete etapas básicas no processo de previsão, que são:

1. Determinar o uso da previsão;
2. Selecionar o item a ser previsto;
3. Determinar o horizonte de tempo da previsão;
4. Selecionar o(s) método(s) de previsão;
5. Coletar os dados necessários para fazer a previsão;
6. Fazer a previsão;
7. Validar e implementar o resultado.

O problema dessa metodologia é o fato de que a determinação dos modelos de previsão são definidos antes da coleta e análise dos dados de entrada disponíveis, por isso, a abordagem que será utilizada nesse trabalho é a levantada por (**??**), que separa o processo de previsão em 5 etapas ao invés de 7:

1. Definição do problema;
2. Coleta de informações;
3. Análise exploratória preliminar;
4. Escolha e ajuste dos modelos;
5. Uso e avaliação dos modelos;

Com essa metodologia, a escolha dos modelos é intrinsicamente baseada nas características encontradas nos dados que servirão de entrada para a previsão, o que faz mais sentido do que o método anterior.

Como a definição do problema e o objetivo da previsão a ser realizada já foram explicitados anteriormente, essa seção é dedicada à discussão do estudo de caso da situação atual do

problema, abrangendo as quatro etapas restantes descritas acima. Iniciando pelo conjunto de dados disponível para a análise, até a seleção dos modelos de previsão que serão testados. Incluindo uma revisão da metodologia atual e da sua capacidade preditiva, e as principais abordagens alternativas que podem ser utilizadas para tornar a previsão mais assertiva.

3.1 Conjunto de dados

A base de dados utilizada é composta por 15 colunas e 341.040 linhas, contendo informações sobre as vendas mensais de cada produto de uma companhia do setor alimentício desde o dia primeiro de janeiro de 2019 até o dia primeiro de março de 2023. Sendo que cada linha dessa base representa a venda mensal realizada pela companhia para um produto específico, com o seguinte formato:

	cod_prod	uf	data	quantidade	custo_fixo_medio	custo_variavel_medio	custo_frete	preco_medio	quantidade_conc	preco_medio_conc	valor	valor_conc	preco_nielsen	preco_scanntech	preco_ratio
0	10016	CE	01/01/2019	5340.0	0.000000	0.000000	0.0	106.111124	0.0	NaN	566633.40	0.0	NaN	NaN	NaN
1	10018	BA	01/01/2019	940.0	0.000000	0.000000	0.0	55.126596	0.0	NaN	51819.00	0.0	NaN	NaN	NaN
2	10018	CE	01/01/2019	1706.0	0.000000	0.000000	0.0	57.682415	0.0	NaN	98406.20	0.0	NaN	NaN	NaN
3	10018	ES	01/01/2019	695.0	0.000000	0.000000	0.0	45.990000	0.0	NaN	31963.05	0.0	NaN	NaN	NaN
4	10018	PB	01/01/2019	10700.0	0.000000	0.000000	0.0	58.547925	0.0	NaN	626462.80	0.0	NaN	NaN	NaN
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
341034	3247211	EX	01/03/2023	65200.0	0.201612	0.630600	0.0	1.108250	0.0	NaN	72257.90	0.0	NaN	NaN	NaN
341035	3247212	EX	01/03/2023	29000.0	0.183656	0.677325	0.0	1.108250	0.0	NaN	32139.25	0.0	NaN	NaN	NaN
341036	3247213	EX	01/03/2023	25000.0	0.203206	0.808258	0.0	1.108250	0.0	NaN	27706.25	0.0	NaN	NaN	NaN
341037	3247215	EX	01/03/2023	21000.0	0.214408	0.674993	0.0	1.100440	0.0	NaN	23109.25	0.0	NaN	NaN	NaN
341038	3247221	EX	01/03/2023	81460.0	0.370660	1.528390	0.0	2.101495	0.0	NaN	171187.80	0.0	NaN	NaN	NaN

341039 rows × 15 columns

Figura 2 – Base de dados disponibilizada pela companhia

Inicialmente podemos perceber, que uma grande quantidade das informações dispostas são nulas ou NaN (*not a number*). Mas essas informações serão melhores exploradas em sequência.

As colunas presentes na base de dados nos apresentam as seguintes informações:

1. **Código do SKU**: Código da Unidade de Manutenção de Estoque (*Stock Keeping Unit*) . É o número de série de um produto específico.
2. **Unidade Federativa**: A região no país onde ocorreu o volume de vendas.
3. **Data**: A data em que ocorreu o volume de vendas.
4. **Quantidade**: O volume de vendas do SKU (demanda em Kg).
5. **Custo fixo médio**: O custo fixo do SKU.
6. **Custo variável médio**: O custo variável do SKU.
7. **Custo de transporte**: O custo do frete do SKU.
8. **Preço médio**: O preço médio desse produto em uma região e data específicas.

9. **Vendas dos concorrentes**: Para alguns SKUs e datas existem dados sobre o volume de vendas dos concorrentes do fabricante em produtos semelhantes.

10. **Preço médio dos concorrentes**: O preço médio dos concorrentes desse produto no ponto de venda.

11. **Custo total**: A soma de custo fixo, custo variável e custo de transporte.

12. **Valor**: Faturamento bruto total, ou seja, quantidade multiplicada pelo preço médio.

13. **Valor da Concorrência**: Faturamento bruto total da concorrência, ou seja, venda dos concorrentes multiplicada pelo preço médio da concorrência.

14. **Preço Nielsen**: um dos fornecedores de dados da companhia, mostra o preço coletado para aquele produto no ponto de venda.

15. **Preço Scanntech**: outro dos fornecedores de dados da companhia, mostra o preço coletado para aquele produto no ponto de venda.

Dentre as variáveis disponíveis listadas acima, como foi citado na seção 2.2, podemos dividir as variáveis em dois grupos, sendo o "Código do SKU"e a "Unidade Federativa"variáveis categóricas, enquanto todas as outras são variáveis quantitativas.

O interesse desse trabalho é analisar alternativas que podem melhorar a capacidade preditiva do volume de vendas da companhia. Como mencionado anteriormente no item 2.3, essa previsão pode ser realizada através da análise da correlação entre essas variáveis e a que desejamos prever, ou, com uma análise dos padrões da série histórica da própria variável de interesse.

3.2 Análise exploratória inicial dos dados

Utilizando a plataforma Google Colaboratory (**??**), a linguagem de programação Python (**??**), a bilbioteca Pandas (**??**) foi realizada a análise exploratória inicial no *dataset* disponível.

Analisando as variáveis categóricas do conjuto de dados (Código do SKU e Unidades Federativas), constatou-se que o conjunto de dados possui 1559 SKUs diferentes vendidos pela companhia, sendo que as vendas são divididas entre 27 unidades federativas brasileiras, mais a exportação.

Além disso, realizou-se uma análise preliminar relacionando os SKUs mais vendidos da companhia com o percentual de vendas representado por eles no último mês, essa análise constatou que os 100 principais SKUs da companhia representam 53,04% do volume de venda da empresa, e que 90% do volume de vendas são realizadas pelos 520 SKUs mais vendidos, ou seja, os outros 1039 SKUs presentes nessa base de dados são responsáveis por apenas 10% do volume de vendas do negócio. Esse tipo de informação é relevante para delimitar o escopo

de trabalho que deverá ser realizado, mostrando que não é necessário ter modelos específicos para todos os SKUs para ter um impacto positivo e significativo na previsão de demanda da companhia.

Os códigos utilizados para essa análise foram:

```python
import pandas as pd

# Carregar os dados em um DataFrame do Pandas
data = pd.read_csv("C:/Users/dedav/Desktop/UFRGS/Ideias TCC/TCC - Gustavo Azevedo/dados2023-04-19.csv",sep=";")

# Converter a coluna de datas para o tipo de dado `datetime`
data['data'] = pd.to_datetime(data['data'])

# Filtrar o dataset para obter apenas os dados do último mês
ultimo_mes = data[data['data'].dt.month == data['data'].max().month]

# Agrupar os dados filtrados pelo código do produto e calcular a soma do volume vendido para cada produto
grupo_produto = ultimo_mes.groupby('cod_prod')['quantidade'].sum()

# Ordenar os produtos com base na soma do volume vendido em ordem decrescente
produtos_mais_vendidos = grupo_produto.sort_values(ascending=False)

# Calcular a soma do volume dos produtos mais vendidos
soma_top = produtos_mais_vendidos.head(100).sum()

# Calcular o percentual de vendas do último mês representado por eles
percentual_vendas = (soma_top / grupo_produto.sum()) * 100

# Imprimir o resultado
print(f"Os 100 produtos mais vendidos representam {percentual_vendas:.2f}% das vendas do último mês.")
```

```
Os 100 produtos mais vendidos representam 53.04% das vendas do último mês.
```

Figura 3 – Código: percentual de vendas representado pelos 100 principais SKUs)

```python
# Ordenar os produtos com base na soma do volume vendido em ordem decrescente
produtos_mais_vendidos = grupo_produto.sort_values(ascending=False)

# Calcular o percentual acumulado de vendas para cada produto
percentual_acumulado = produtos_mais_vendidos.cumsum() / grupo_produto.sum()

# Encontrar o número de produtos necessários para atingir o % de vendas
num_produtos_percent = (percentual_acumulado <= 0.9).sum()

# Imprimir o resultado
print(f"{num_produtos_percent} produtos representam 90% das vendas da companhia.")
```

```
520 produtos representam 90% das vendas da companhia.
```

Figura 4 – Código: número de SKUs que representam 90% das vendas do último mês

3.2.1 Variável Quantidade

Entender a natureza da variável que se deseja prever é fundamental na escolha dos modelos de previsão que serão testados, por isso foi realizada uma análise gráfica da variável que representa o volume de vendas da companhia em forma de histograma, podendo-se obter mais informações sobre o formato da distribuição dessa variável, bem como identificar anomalias e *outliers* nos dados de entrada. Essa etapa é importante pois permite a identificação da simetria da distribuição e nos indica a necessidade de utilizar técnicas de transformação nos dados. Com esse intuito, foi realizada uma representação gráfica do histograma do volume de vendas de cada SKU ao longo dos quatro anos abrangidos pela análise, o resultado encontrado foi o seguinte:

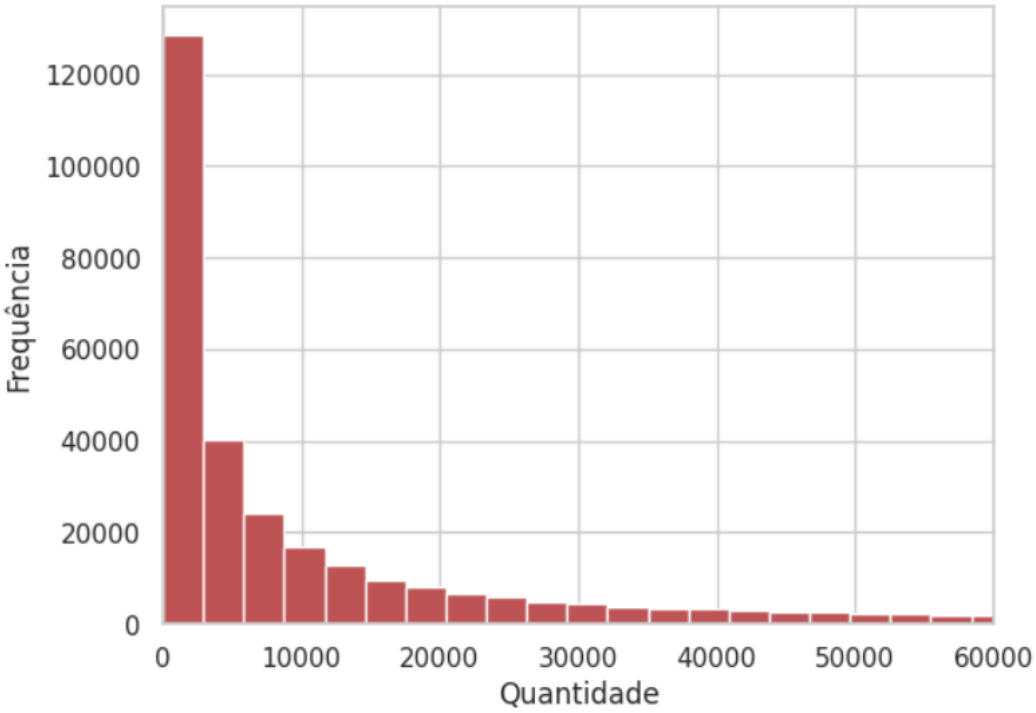


Figura 5 – Histograma do histórico de vendas mensal até 60 toneladas

Podemos observar que o primeiro grupo tem a maioria das observações, seguido por um grupo com um menor número de observações com um grande volume de vendas e, por fim, alguns dados com enormes quantidades vendidas (na casa das 60 toneladas). Filtrando o histograma para até 10 toneladas de venda mensal, temos:

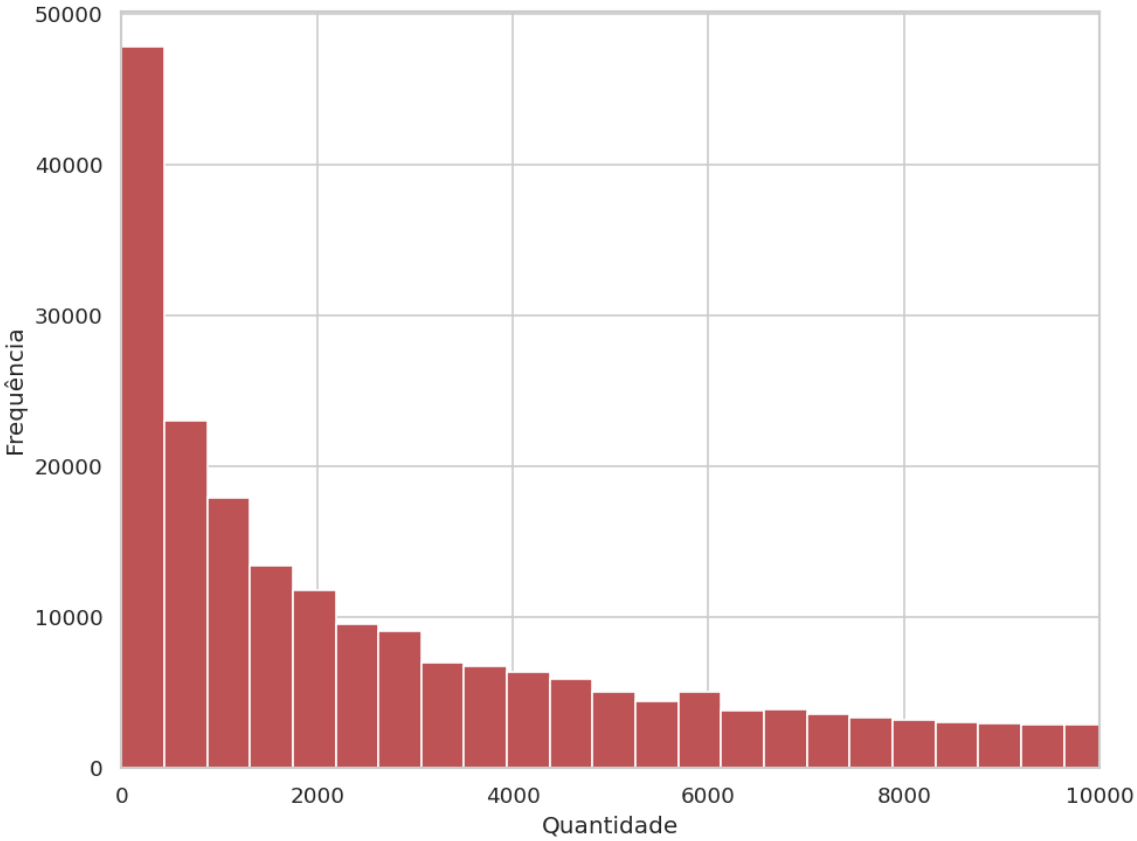


Figura 6 – Histograma do histórico de vendas mensal até 10 toneladas

Realizando mais um filtro na primeira faixa de dados, até 4 toneladas temos:

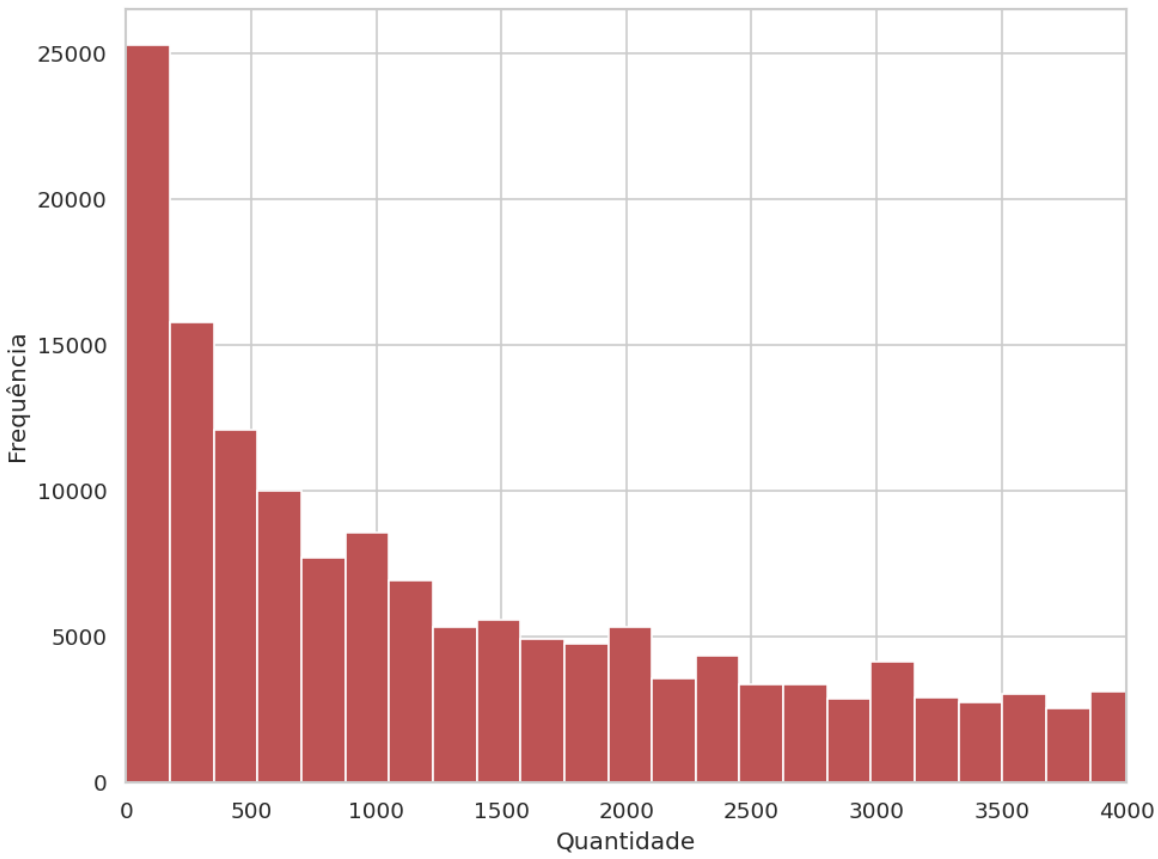


Figura 7 – Histograma do histórico de vendas mensal até 4 toneladas

É possível perceber que a maior parte das vendas mensais realizadas por produto na companhia são na faixa de até 1000 Kg.

Como a diferença entre os grupos é drástica e na ordem exponencial, aplicamos o logarítmo nos dados para uma análise mais aprofundada. Graficando o histograma novamente depois de realizar essa transformação, podemos observar a distribuição abaixo:

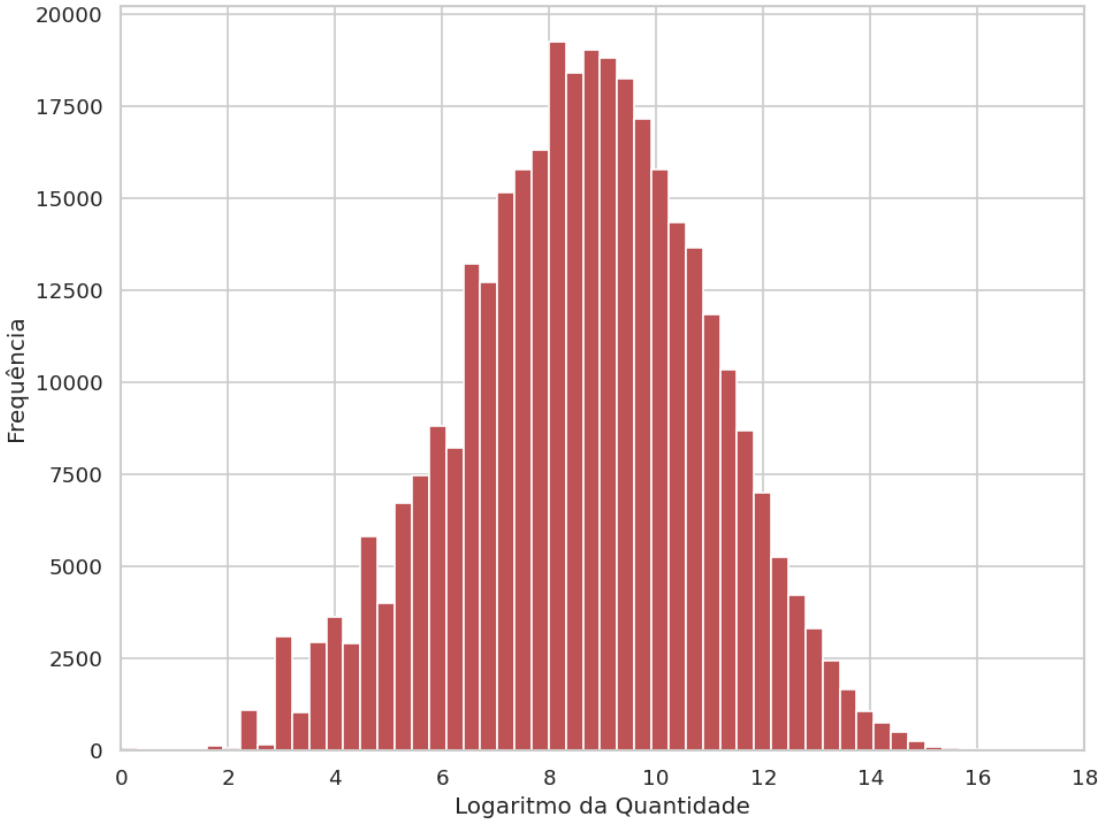


Figura 8 – Logarítmo do histórico de vendas

Utilizando as informações de média e variância dos dados depois da transformação logarítmica, com os valores 8.62 e 2.3, respectivamente, é possível graficar o histograma novamente, acompanhado da sua distribuição normal equivalente, o resultado ficou:

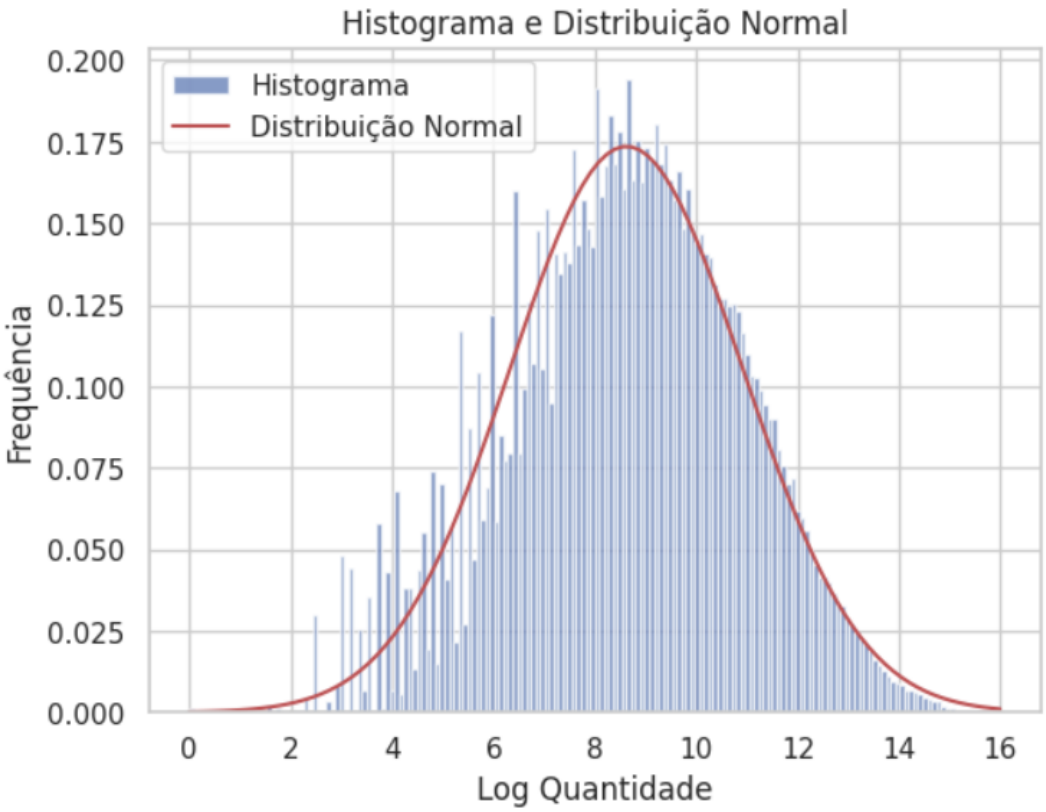


Figura 9 – Logarítmo do histórico de vendas e Distribuição normal

3.2.2 Variável Preço

O preço é a principal variável a ser considerada ao prever a demanda de um produto devido à sua influência direta sobre o comportamento do consumidor. Uma mudança no preço pode afetar a decisão de compra dos consumidores de forma imediata e perceptível: um preço mais alto geralmente leva a uma redução na demanda, enquanto um preço mais baixo tende a aumentá-la. Além disso, o preço muitas vezes reflete a percepção de valor do produto para os consumidores, afetando sua disposição em adquiri-lo. A distribuição dos dados de preço do *dataset* inicial tem o seguinte formato:

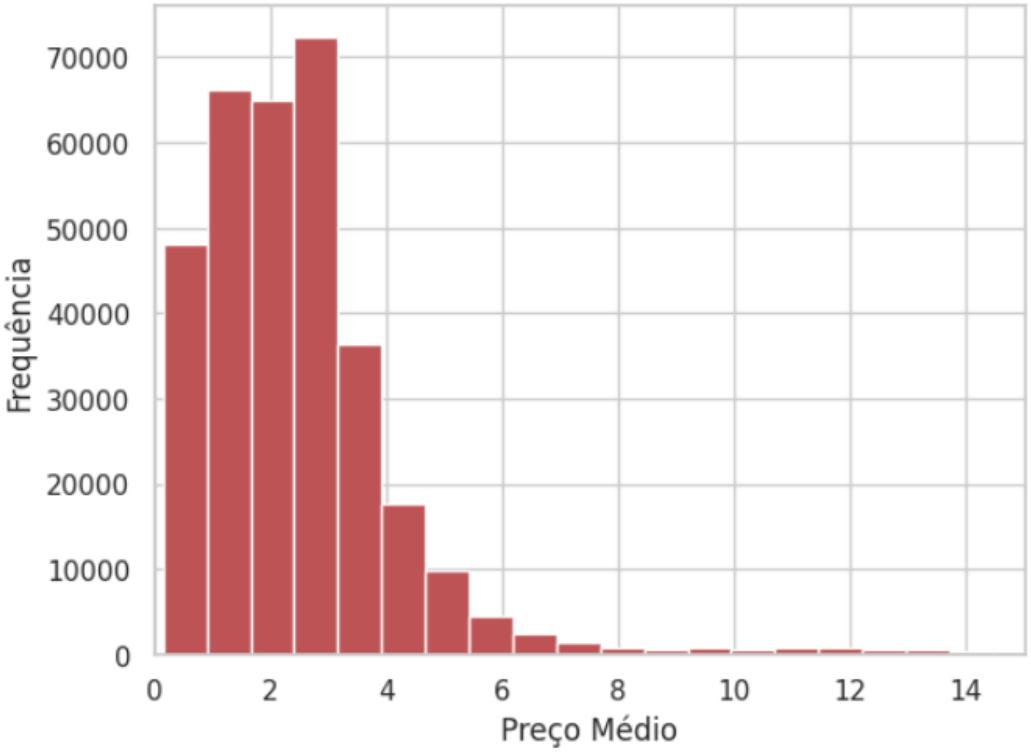


Figura 10 – Distribuição do preço

É possível perceber que a maior parte dos preços de venda dos produtos da companhia estão concentrados na faixa entre um e quatro reais.

3.3 Pré processamento dos dados

Após a análise exploratória, foi realizado o pré-processamento dos dados. Isso envolveu uma série de verificações e modificações no *dataset* inicial para melhorar sua futura manipulação e utilização para a modelagem estatística, tais como:

1. **Retirar dados de venda no exterior;**
2. **Agregar os dados de venda a nível Brasil**: agrupar a venda de cada produto para todos os estados, resultando em um dado de venda mensal por SKU parz todo Brasil
3. **Retirar produtos que saíram de linha:** para isso, foram excluídos do *dataset* todos os produtos que não foram vendidos no mês anterior, esses produtos foram considerados como descontinuados. Dos 1559 SKUs contidos no histórico, sobraram 770.
4. **Ponderação pela quantidade vendida:** para atribuir pesos diferentes aos dados mais relevantes, multiplicamos as variáveis pela sua respectiva quantidade vendida e dividimos pela quantidade vendida total daquele produto, isso foi relizado para as seguintes variáveis:
    - Preço médio;

- Custo total;
- Preço médio da concorrência.

5. **Criação de nova variável auxiliar:** utilizando os dados de Preço Médio e Custo Total, calculamos a Margem Média do SKU.

6. **Criação de nova variável auxiliar:** utilizando o preço praticado no mês M e o preço praticado no mês M-1, calculamos uma nova variável chamada de Diferença de Preço, que é a diferença percentual entre o preço nesses dois meses.

7. **Seleção das colunas de interesse**

Depois dessas modificações, o *dataset* final ficou com o seguinte formato:

	cod_prod	data	quantidade	preco_medio	custo_variavel_medio	preco_medio_conc	custo_total	diferenca_preco	margem_media
0	10016	2019-01-01	5340.0	106.111124	0.000000	NaN	0.000000	NaN	106.111124
1	10016	2019-02-01	4480.0	105.000000	0.000000	NaN	0.000000	-0.010471	105.000000
2	10016	2019-03-01	3980.0	104.472161	0.000000	NaN	0.000000	-0.005027	104.472161
3	10016	2019-04-01	4640.0	103.990000	0.000000	NaN	0.000000	-0.004615	103.990000
4	10016	2019-05-01	4480.0	103.990000	0.000000	NaN	0.000000	0.000000	103.990000
...	...	...	...	...	...	...	...	...	...
24795	547585	2022-11-01	151866.0	2.850904	1.201199	NaN	1.611560	0.019858	1.239344
24796	547585	2022-12-01	145182.0	2.780935	1.247631	NaN	1.624922	-0.024543	1.156013
24797	547585	2023-01-01	98080.0	2.724030	1.044352	NaN	1.425055	-0.020462	1.298976
24798	547585	2023-02-01	88512.0	2.760986	1.224215	NaN	1.559236	0.013567	1.201750
24799	547585	2023-03-01	98461.0	2.750676	1.228751	NaN	1.584655	-0.003734	1.166020

Figura 11 – Dados processados

É possível perceber que existe uma grande falta de dados na base disponibilizada pela companhia, principalmente referente aos dados de custo e preço médio da concorrência. Das 24800 linhas que compõem a tabela, 12320 delas tem NaN (*not a number*) como preço médio da concorrência, e 2600 linhas com o custo total do SKU zerados.

Quando os dados de entrada se aproximam de uma distribuição normal é um forte indicador de que modelos paramétricos poderão ser utilizados. Modelos paramétricos são um tipo de modelo estatístico que assume uma forma específica e bem definida para a distribuição subjacente dos dados. Esses modelos são caracterizados por um conjunto fixo de parâmetros que determinam a forma da distribuição e descrevem suas propriedades estatísticas, tais como a média e a variância.

Como base do estudo da modelagem estatística foi selecionado um dos SKUs com maior quantidade de vendas da companhia, seu código é 21101 e se refere a um pacote de espaguete de 500G . Esse SKU foi escolhido devido a sua relevância em volume de vendas da companhia e, por consequência, ser um dos SKUs que possui grande parte dos seus dados de custo e preço da concorrência completos no *dataset*.

Para esse SKU específico graficamos seu histórico de vendas, preço médio praticado, custo total e preço da concorrência. Esse histórico tem o seguinte formato:

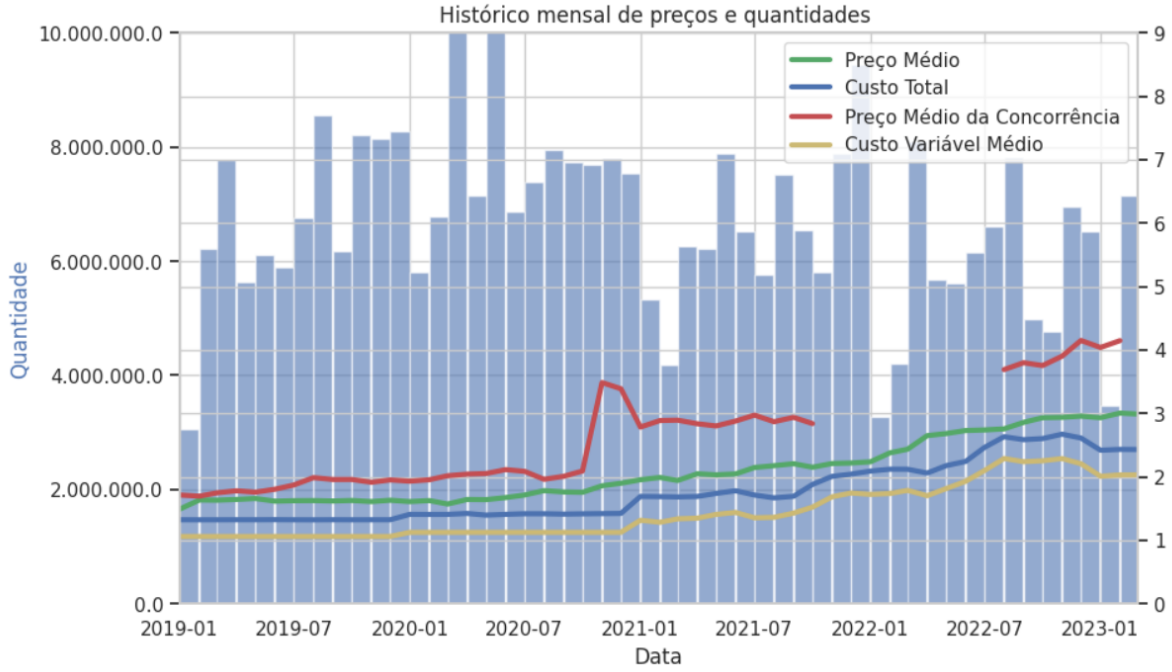


Figura 12 – Histórico SKU 21101

É possível perceber um *gap* nos dados de preço da concorrência para esse SKU no ano de 2022. Essa mesma falta de dados foi observada nos 10 SKUs mais vendidos da companhia. A conclusão que se chega é que houve algum problema com o fornecimento de dados do ponto de venda nesse período.

É interessante ressaltar a grande difereça que se tem entre o Preço Médio e o Preço da Concorrência. Isso se deve ao fato de que os dados que temos de preço da concorrência são os preços que o consumidor final paga pelo produto, enquanto os dados de preço do produto próprio que possuímos são os preços praticados pela fabricante para os revendedores (mercados e atacados) que vão adicionar sua margem de lucro em cima e revender o produto ao consumidor final.

Da mesma forma que para o conjunto de dados completo realizou-se a análise da distribuição dos histogramas de quantidade, preço médio e custo total desse SKU específico. Os resultados estão apresetados abaixo:

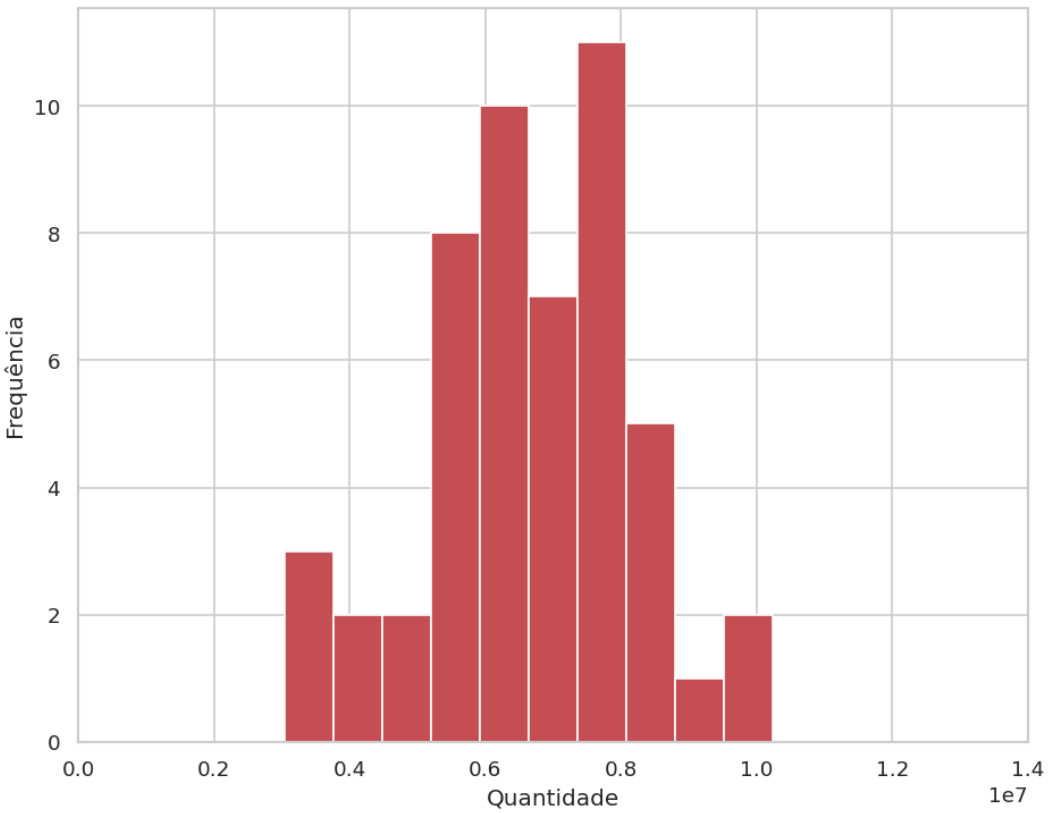


Figura 13 – Distribuição da quantidade: SKU 21101

A quantidade vendida mensalmente desse produto variou entre 3 e 10 mil toneladas por mês.

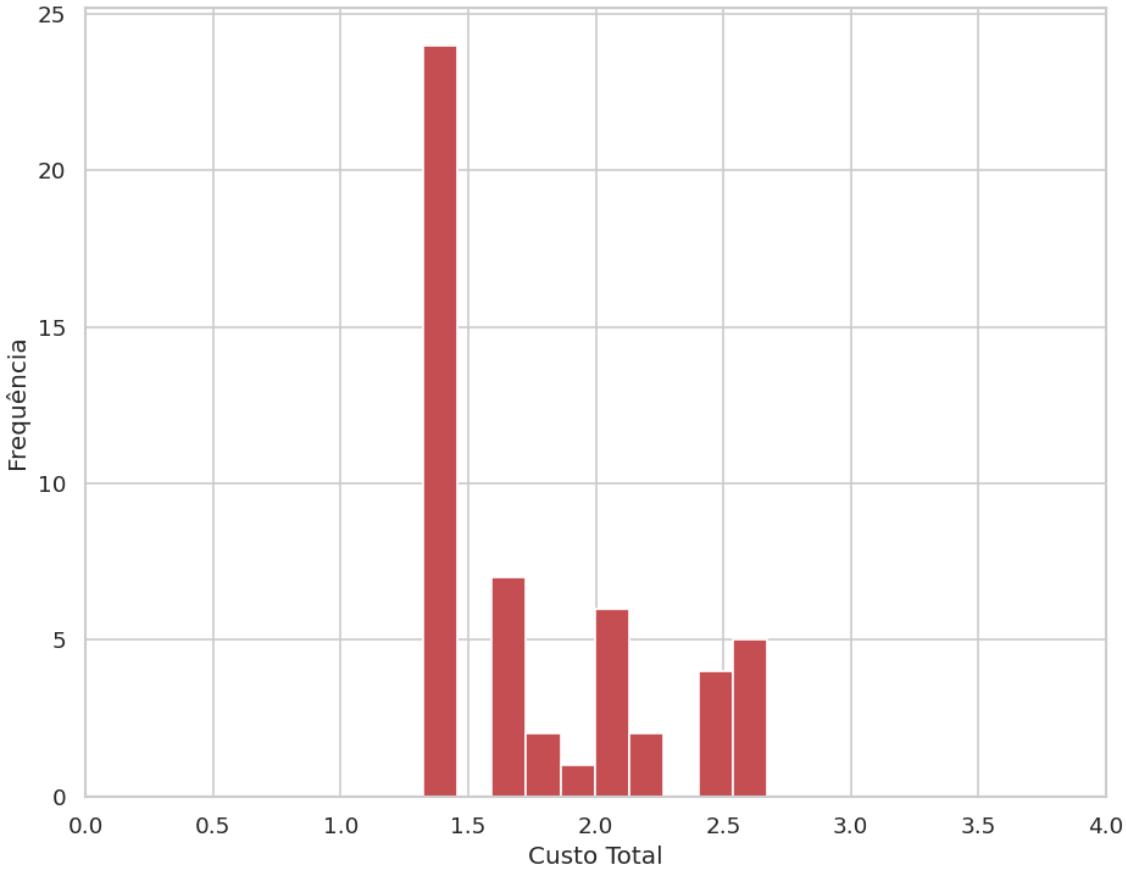


Figura 14 – Distribuição do Custo Total: SKU 21101

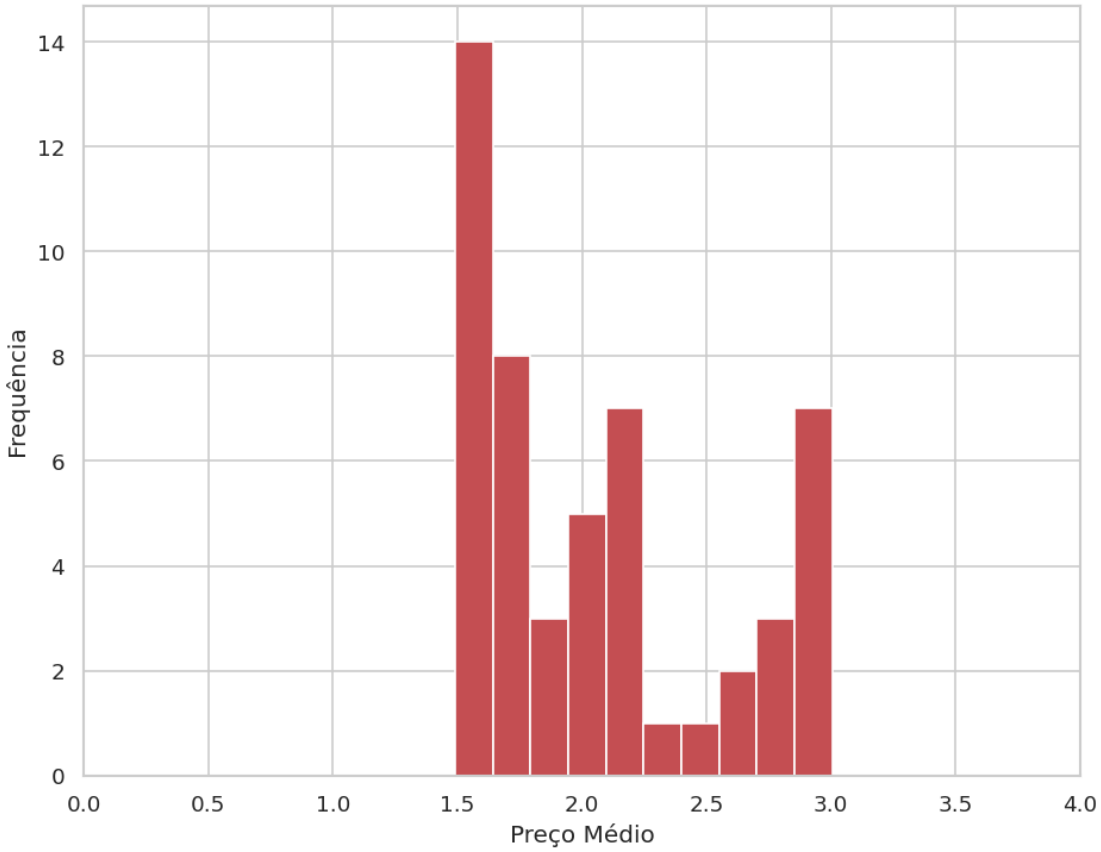


Figura 15 – Distribuição do preço médio: SKU 21101

O preço médio vendido do produto ficou entre R$ 1,50 e R$ 3,00 no histórico. Como esperado o Custo e o Preço Médio tem uma distribuição parecida por estarem muito ligados entre si.

3.4 Modelagem estatística

A aplicação da modelagem estatística foi realizada utilizando a biblioteca *statsmodels* (**??**) do Python. Essa biblioteca oferece um conjunto de recursos e funcionalidades estatísticas que nos permitem realizar análises detalhadas dos dados, desde estimar modelos de regressão e séries temporais, ajustá-los aos dados e testar hipóteses.

Para os modelos ARIMAX testados separamos o *dataset* em dados de treino e dados de teste. Utilizamos 80% dos dados para treinar o modelo para a previsão e 20% dos dados sendo previstos (dados de teste), ou seja, os dados de treino vão de Janeiro de 2019 até Maio de 2022 (29 meses), e os dados de teste começam em Junho de 2022 e vão até Março de 2023 (10 meses).

3.4.1 Processo de seleção de varáveis

Para determinar as principais variáveis que serão utilizadas nos modelos estatísticos podemos fazer uma análise da correlação entre cada uma das colunas do *dataset* (**??**).

	cod_prod	quantidade	preco_medio	custo_variavel_medio	preco_medio_conc	custo_total	diferenca_preco	margem_media
cod_prod	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
quantidade	NaN	1.000000	-0.275236	-0.248686	-0.247187	-0.273230	-0.229659	-0.090493
preco_medio	NaN	-0.275236	1.000000	0.968448	0.931600	0.973480	0.132567	0.405620
custo_variavel_medio	NaN	-0.248686	0.968448	1.000000	0.876127	0.997686	0.065576	0.173389
preco_medio_conc	NaN	-0.247187	0.931600	0.876127	1.000000	0.892214	0.129262	0.408940
custo_total	NaN	-0.273230	0.973480	0.997686	0.892214	1.000000	0.073390	0.185756
diferenca_preco	NaN	-0.229659	0.132567	0.065576	0.129262	0.073390	1.000000	0.275680
margem_media	NaN	-0.090493	0.405620	0.173389	0.408940	0.185756	0.275680	1.000000

Figura 16 – Correlação entre as variáveis

Como esperado, as variáveis tem correlação negativa com a quantidade, isso ocorre porque existe uma relação inversa entre o aumento do preço de um produto e sua demanda. Quando o preço aumenta a tendência é que o consumo e as vendas daquele produto diminuam. Além disso, existe uma forte correlação entre diversas variáveis por elas estarem direta ou indiretamente relacionadas, tais como preço e custo.

Em *datasets* pequenos, como é o caso do que está sendo utilizado, é importante tomar cuidado com a adição de variáveis aos modelos e sua colinearidade, que se trata da correlação entre uma ou mais variáveis que estão sendo utulizadas no modelo (**??**). Para uma amostra menor de dados, a probabilidade de *overfitting* aumenta muito conforme adicionamos variáveis, não necessariamente um maior número de variáveis gera melhores previsões.

3.4.2 Regressão Linear

Por ser um dos modelos mais simples que podemos implementar, utilizaremos o modelo de Regressão Linear como o modelo base para comparação desse estudo. Foi realizada uma regressão linear entre a Quantidade e o Preço Médio, que são nossas variáveis de maior interesse.

Um modelo de previsão utilizando regressão linear é uma abordagem estatística para estimar valores futuros com base em padrões observados nos dados históricos. No contexto da análise de vendas, podemos criar um modelo que prevê a quantidade vendida de um produto com base no preço médio praticado desse produto.

A fórmula do modelo de previsão com regressão linear é semelhante à equação geral da regressão linear:

$$y = mx + b \tag{6}$$

Onde:

- y: é a variável dependente (variável que queremos prever);
- x: é a variável independente (variável de entrada ou característica);

- m: é o coeficiente angular (inclinação da reta de regressão);
- b: é o coeficiente linear (intercepta no eixo y).

É importante lembrar que um modelo de previsão é uma simplificação da realidade e pressupõe que a relação linear entre as variáveis seja uma boa aproximação. Além disso, a qualidade das previsões depende da qualidade dos dados usados para treinar o modelo e da validade da suposição de linearidade.

3.4.3 Regressão Multilinear

Na regressão multilinear, o objetivo é encontrar a melhor combinação linear das variáveis independentes que se ajuste aos dados e preveja a variável dependente de forma mais precisa possível. Cada variável independente possui um coeficiente de regressão associado, que indica a contribuição relativa dess variável na previsão, mantendo as outras constantes (**??**). No caso do problema em questão, é utilizada para encontrar a melhor combinação de coeficientes para as variáveis de custo, preço do SKU e que expliquem o volume de vendas daquele SKU. O preço praticado pela concorrência não foi considerado na análise devido a falta de dados mencionada anteriormente.

A fórmula geral para um modelo de regressão multilinear é:

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_n X_n + \varepsilon \tag{7}$$

Onde:

- Y é a variável dependente que estamos tentando prever, em nosso caso a demanda;
- $X_1, X_2, \ldots, X_n$ são as variáveis independentes (preço praticado, preço da concorrência e custo);
- $\beta_0, \beta_1, \beta_2, \ldots, \beta_n$ são os coeficientes de regressão que quantificam o impacto das variáveis independentes na variável dependente;
- ε representa o termo de erro, que captura as discrepâncias entre a previsão do modelo e os valores reais da variável dependente.

Para estimar os coeficientes de regressão, é comum utilizar o mesmo método utilizado no *fitting*, o método dos mínimos quadrados, que busca minimizar a soma dos quadrados dos resíduos (diferenças entre os valores reais e as previsões do modelo).

No modelo implementado atualmente para esse cliente, o sMAPE (erro absoluto simétrico médio em termos percentuais), calculado pela última vez em fevereiro de 2023 para esse modelo em utilização foi de 28,9%.

3.4.4 ARIMAX

O modelo ARIMAX é uma extensão do modelo ARIMA, que incorpora variáveis exógenas na previsão. Enquanto o modelo ARIMA é adequado para a previsão de séries temporais com uma só variável, o modelo ARIMAX é especialmente útil quando existem fatores externos que podem impactar a série temporal e não estão sendo capturados pelas próprias observações passadas. Variáveis independentes (exógenas) que podem influenciar a série temporal que estamos tentando prever, são incluidas no modelo como regressores adicionas, melhorando a precisão das previsões do ARIMA.

O modelo ARIMAX possui 3 parâmetros (*p,d,q*) que são utilizados na hora de definir o modelo, um para cada fator da previsão de séries temporais. Detalhando melhor o significado das componentes que dão nome ao modelo ARIMAX, temos:

- **Auto Regressivo (fator *p*):** modela a dependência linear entre os valores passados e presentes da série temporal. Ele usa as observações anteriores da série temporal como entradas para estimar os coeficientes de regressão. O fator p é número de valores passados que serão utilizados para prever o próximo valor da série temporal, quanto maior for o p, maior é a dependência dos valores passados na previsão. Para escolher o valor de p são utilizadas funções de autocorrelação (ACF) e funções de autocorrelação parcial (PACF)
- **Diferenciação Integrada (fator *d*):** torna a série temporal estacionária. Estacionariedade é uma propriedade desejável em séries temporais, onde a média e a variância dos dados não mudam com o tempo. A diferenciação é realizada subtraindo-se o valor atual da série temporal pelo valor da série temporal em um passo de tempo anterior. O fator d indica o número de vezes que a série temporal será diferenciada para torná-la estacionária. Se já for estacionária, então $d = 0$. Para escolher o valor de d, é comum utilizar testes estatísticos como Dickey-Fuller, que avalia a estacionaridade da série.
- **Média Móvel (fator *q*)**: modela a dependência linear entre os erros de previsão passados e presentes, estima os coeficientes de regressão com base nos erros residuais das previsões anteriores. O fator q indica o número de erros passados que serão utilizados para corrigir a previsão atual. Para escolher o valor de q, é comum utilizar as funções ACF e PACF dos erros de previsão.
- **Variáveis exógenas:** são variáveis independentes que não fazem parte da série temporal de interesse, mas podem ter influência sobre ela, são utilizadas para modelar a dependência linear entre as variáveis exógenas e a série temporal utilizada.

No contexto do projeto, é altamente relevante utilizar esse modelo, pois além do histórico de demanda, temos acesso a três variáveis independentes no *dataset* inicial, todas com uma influência direta ou indireta na demanda de um produto, sendo elas: o preço aplicado, o custo de produção do SKU e o preço aplicado pelos concorrentes em produtos similares.

O modelo ARIMAX é representado pela fórmula geral:

$$Y_t = c+\phi_1 Y_{t-1}+\phi_2 Y_{t-2}+\ldots+\phi_p Y_{t-p}+\theta_1 \varepsilon_{t-1}+\theta_2 \varepsilon_{t-2}+\ldots+\theta_q \varepsilon_{t-q}+\beta_1 X_{1,t}+\beta_2 X_{2,t}+\ldots+\beta_k X_{k,t}+\varepsilon(t) \tag{8}$$

Onde:

- Y_t é a variável de interesse que estamos tentando prever no tempo t.
- c é uma constante ou o termo de interceptação do modelo.
- $\phi_1, \phi_2, \ldots, \phi_p$ são os coeficientes dos termos autorregressivos que representam a relação entre os valores passados de Y e Y_t.
- $Y_{t-1}, Y_{t-2}, \ldots, Y_{t-p}$ são os valores passados de Y.
- $\theta_1, \theta_2, \ldots, \theta_q$ são os coeficientes dos termos de média móvel que representam a relação entre os erros passados (ε) e Y_t.
- $\varepsilon_{t-1}, \varepsilon_{t-2}, \ldots, \varepsilon_{t-q}$ são os erros passados do modelo.
- $X_{1t}, X_{2t}, \ldots, X_{kt}$ são os valores das variáveis exógenas no tempo t.
- $\beta_1, \beta_2, \ldots, \beta_k$ são os coeficientes das variáveis exógenas $X_{1t}, X_{2t}, \ldots, X_{kt}$ que representam a relação entre as variáveis exógenas e Y_t.
- $\varepsilon(t)$ é o erro atual do modelo.

O objetivo do modelo ARIMAX é encontrar os valores adequados para os coeficientes ϕ, θ, e β que minimizem o erro entre os valores observados e previstos de Y.

Dentro do modelo ARIMAX foi utilizada uma técnica de seleção para determinar quais os melhores parâmetros *p,d,q* para a previsão. O nome dessa técnica é Akaike Information Criterion (AIC), que são valores associados a modelos estatísticos e utilizados para avaliar a adequação desses modelos aos dados observados através da análise da verossimilhança. Essa metodologia analisou o resultado dos AICs para as combinações lineares entre os parâmetros *p,d,q* com o *p* (auto regressivo), indo de 1 a 3, o *d* (diferenciação integrada) indo de 0 a 2, e o *q* (média móvel) indo de 0 a 2. Quanto maior for o AIC resutante melhor é a adequação daquele modelo para previsão dos dados, analisando os resultados foram definidos 3 modelos relevantes para comparação, os modelos ARIMAX(1,0,1), ARIMAX(3,2,2), ARIMAX(2,0,1)

3.5 Análise dos modelos

Para realizar a análise e comparação dos modelos foi implementada uma função no código em Python que calcula e armazena as métricas mais relevantes para análise dos modelos

como citado em 2.4. Que são o RMSE, o MAPE e o coeficiente de determinação R^2 tanto para os dados de treinamento do modelo como para os de teste e previsão. O código desenvolvido para essa função é o seguinte:

```
def abbreviate_var_name(var_name):
    abv_name = "".join(
        [f"{word[0]}" for word in re.split(" |_", var_name.strip()) if word[0] != "("]
    )
    return abv_name

def print_and_save_metrics(y_train, y_pred_train, y_pred_test, x_train, model, model_name, cv=5):
    """
    Métricas para acompanhar o desempenho dos modelos

    MAPE: Erro Médio Absoluto Percentual (Mean Absolute Percentage Error)
    RMSE: Raiz do Erro Quadrático Médio (Root Mean Squared Error)
    R²: Coeficiente de Determinação (quantidade da variância explicada pelo modelo)
    """
    train_rmse = np.sqrt(mean_squared_error(y_train, y_pred_train))
    train_mape = mean_absolute_percentage_error(y_train, y_pred_train)
    train_r2 = r2_score(y_train, y_pred_train)
    print(f'O RMSE do treinamento é de: { train_rmse:,.2f}')
    print(f'O MAPE do treinamento é de: { train_mape:.2%}')
    print(f'O R² do treinamento é de: { train_r2:.2f}\n')

    test_rmse = np.sqrt(mean_squared_error(y_test, y_pred_test))
    test_mape = mean_absolute_percentage_error(y_test, y_pred_test)
    test_r2 = r2_score(y_test, y_pred_test)
    print(f'O RMSE do teste é de: { test_rmse:,.2f}')
    print(f'O MAPE do teste é de: { test_mape:.2%}')
    print(f'O R² do teste é de: { test_r2:.2f}\n')

    results_dict = {"Model": model_name,
                    "MAPE (train)": train_mape,
                    "MAPE (test)": test_mape,
                    "R2 (train)": train_r2,
                    "R2 (test)": test_r2,
                    "model": model}
```

Figura 17 – Código de cálculo das métricas

Para as análises dessas métricas nos dados de treinamento, desconsideramos os 3 pontos iniciais, esses pontos podem ser desconsiderados porque na fase inicial do treinamento os modelos ainda não tem suficientes que baseiem seus resultados e apenas contaminam a análise.

Além do cálculo de métricas realizado, serão comparados gráficos de dispersão(*scatter*) entre a quantidade real no eixo x, e a quantidade prevista no eixo y, sendo a reta traçada em 45 graus, a previsão realizada, quanto mais os pontos do gráfico se aproximarem dessa reta melhor está a previsçao do modelo.

Para completar a análise, a própria bilbioteca *statsmodels* possui uma funcionalidade chamada *model.summary()* utilizada para apresentar o sumário geral do modelo implementado, nesse sumário são apresentadas diversas informações gerais, tais como algumas informações gerais dos modelos, o R^2, os *p-values* das variáveis e o resultado de alguns testes estatísticos.

3.5.1 Teste de premissas de linearidade

Testar as premissas da regressão ajuda a validar se o modelo linear é apropriado para os dados em questão. Se as premissas não forem satisfeitas, isso pode indicar que o modelo não é adequado para descrever o relacionamento entre as variáveis independentes e dependentes (**??**).

Depois dos modelos treinados e as previsões realizadas, foi aplicado o teste das premissas da regressão linear com a próprio biblioteca *statsmodels* (**??**). Esse teste envolve a verificação de:

1. **Linearidade entre as variáveis:** a relação entre as variáveis independentes e a variável dependente deve ser linear.

2. **Normalidade da distribuição dos erros:** através do teste de Kolmogorov-Smirnov que compara a função de distribuição acumulativa da amostra com a função de distribuição acumulativa teórica da distribuição normal. Se o *p-value* calculado for maior que 5% significa que é possíel se afirmar com 95% de de confiança que a distribuição é normal (**??**).

3. **Independência dos erros:** o teste de Durbin-Watson identifica a correlação entre os resíduos de um modelo. O resultado do teste assume valores entre 0 e 4. Valores de coeficiente próximos a 2 indicam que não há evidência de autocorrelaçõ entre os resíduos (**??**).

4. **Homocedasticidade:** a dispersão dos erros deve ser uniforme ao longo de todas as observações, e é verificada através do teste de Breusch-Pagan. Se o *p-value* associado ao teste for menor que 0,05, você pode concluir que há evidência estatística de heterocedasticidade nos resíduos. Se o *p-value* for maior, não há evidência estatística para rejeitar a homocedasticidade (**??**).

5. **Ausência de multicolinearidade:** não deve haver alta correlação entre as variáveis independentes, pois isso pode dificultar a interpretação dos coeficientes, pode ser analisada olhando o gráfico de correlação entre as variáveis apresentado em 16, valores de correlação maiores que 0,7 entre as variáveis independentes pode ser um forte indício de multicolinearidade.

4 RESULTADOS

4.1 Regressão Linear

Para a Regressão Linear utilizamos os dados de Preço Médio como variável preditora e a Quantidade como variável a ser predita. Essa divisão e treinamento dos modelos foi realizada utilizando o código abaixo:

```python
from sklearn import metrics, preprocessing, model_selection, ensemble
from sklearn.linear_model import LinearRegression, Ridge
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error, mean_absolute_percentage_error, r2_score

# Separando as variáveis independentes (X) da variável dependente (y)

x = sku_21101.sort_values("data")[['preco_medio']].iloc[1:]
y = sku_21101.sort_values("data")['quantidade'].iloc[1:]


# Criando o modelo de regressão linear
model = LinearRegression()

# Treinando o modelo com os dados de treinamento
model.fit(x_train, y_train)

# Fazendo previsões com o conjunto de teste
y_pred_train = model.predict(x_train)


model.predict(x)
```

Figura 18 – Código de treinamento do modelo de Regressão Linear

Para uma melhor visualização dos resultados foi realizada a comparação do Volume Real vendido no período, com a previsão realizada, o código utilizado e o resultado estão apresentados abaixo.

```python
def plot_real_vs_prediction(y_pred, y_train, y_test=None):
    """ Visualizar volume histórico previsto vs realizado """

    fig, ax = plt.subplots(figsize=(8,5))
    ax.scatter(y_train.index, y_train, s=10, label='Dados', zorder=99)
    ax.plot(y.index, y, lw=1.5, label='Volume Real')
    ax.plot(y.index, y_pred, lw=2, color='red', label='Volume Predito')
    ax.set_ylabel('Quantidade')
    ax.set_xlabel('Tempo')
    ax.set_title("Quantidade vs. Predito")
    ax.legend()

y_pred = model.predict(x)

plot_real_vs_prediction(y_pred, y_train, y_test)
```

Figura 19 – Código utilizado para comparação Real X Previsto da Regressão Linear

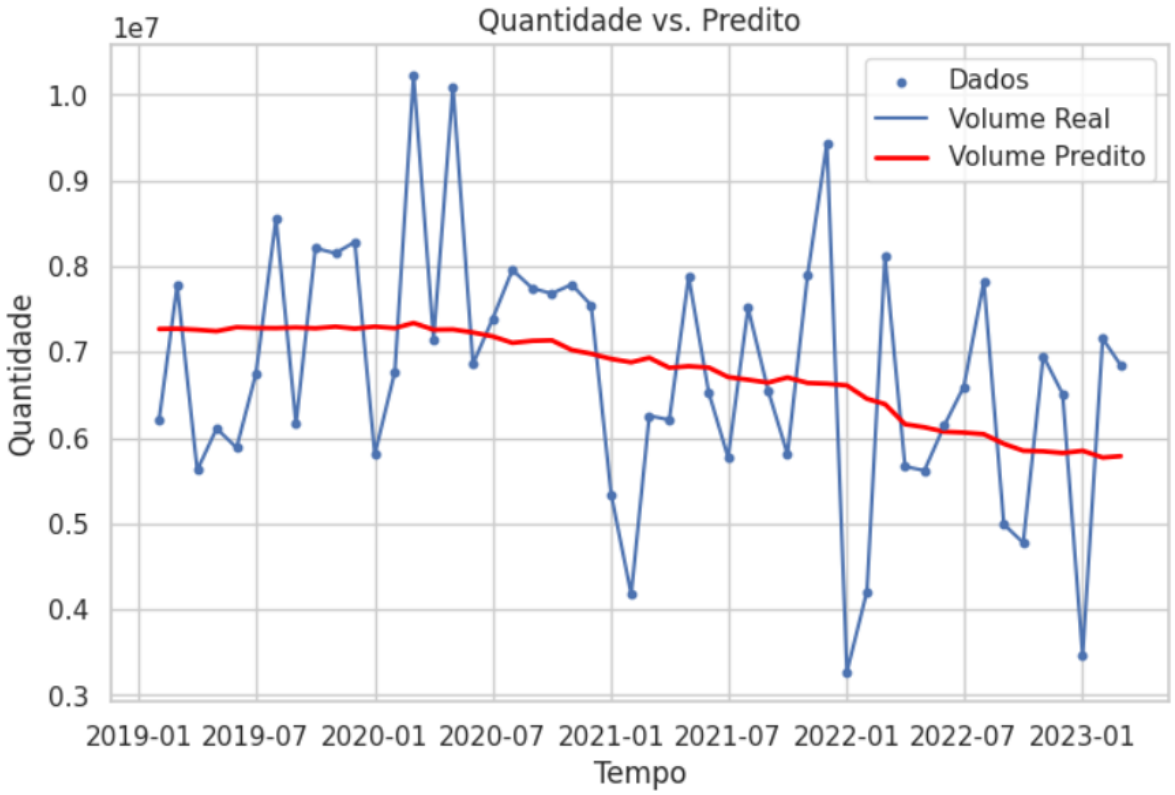


Figura 20 – Comparação Real versus Previsto da Regressão Linear

Adicionando a variável preditora ao gráfico para comparação, temos:

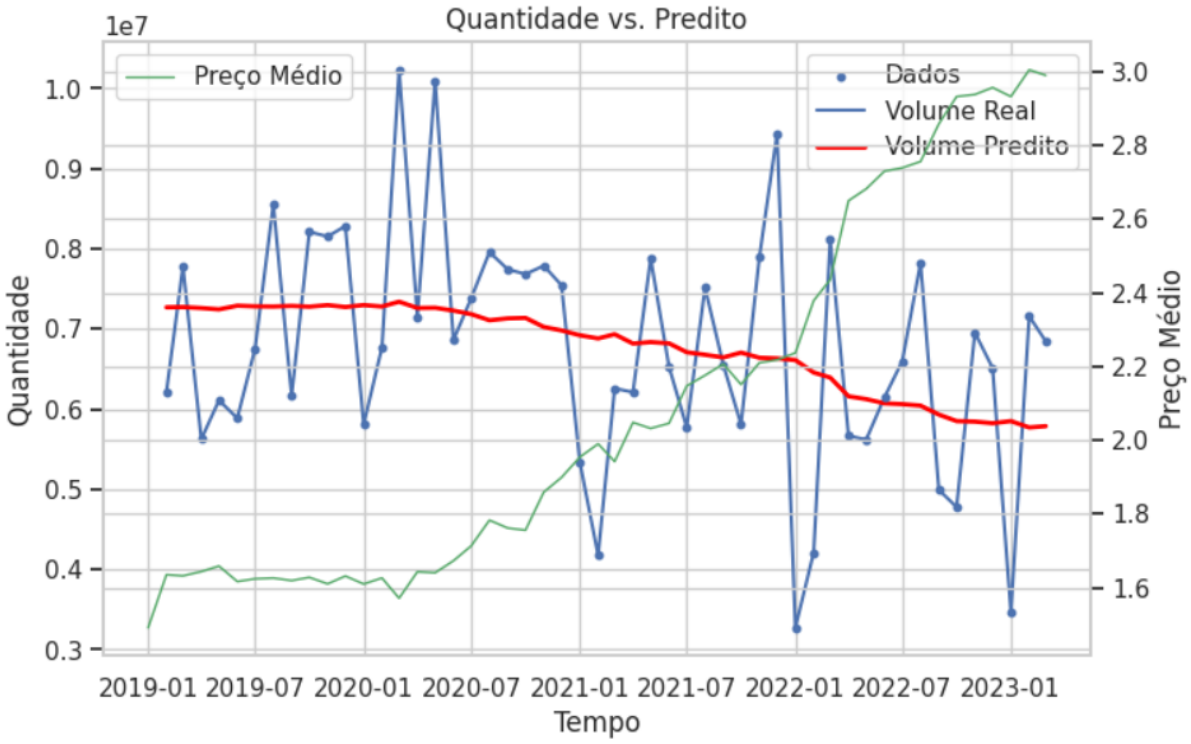


Figura 21 – Comparação Real versus Previsto da Regressão Linear com Preço Médio

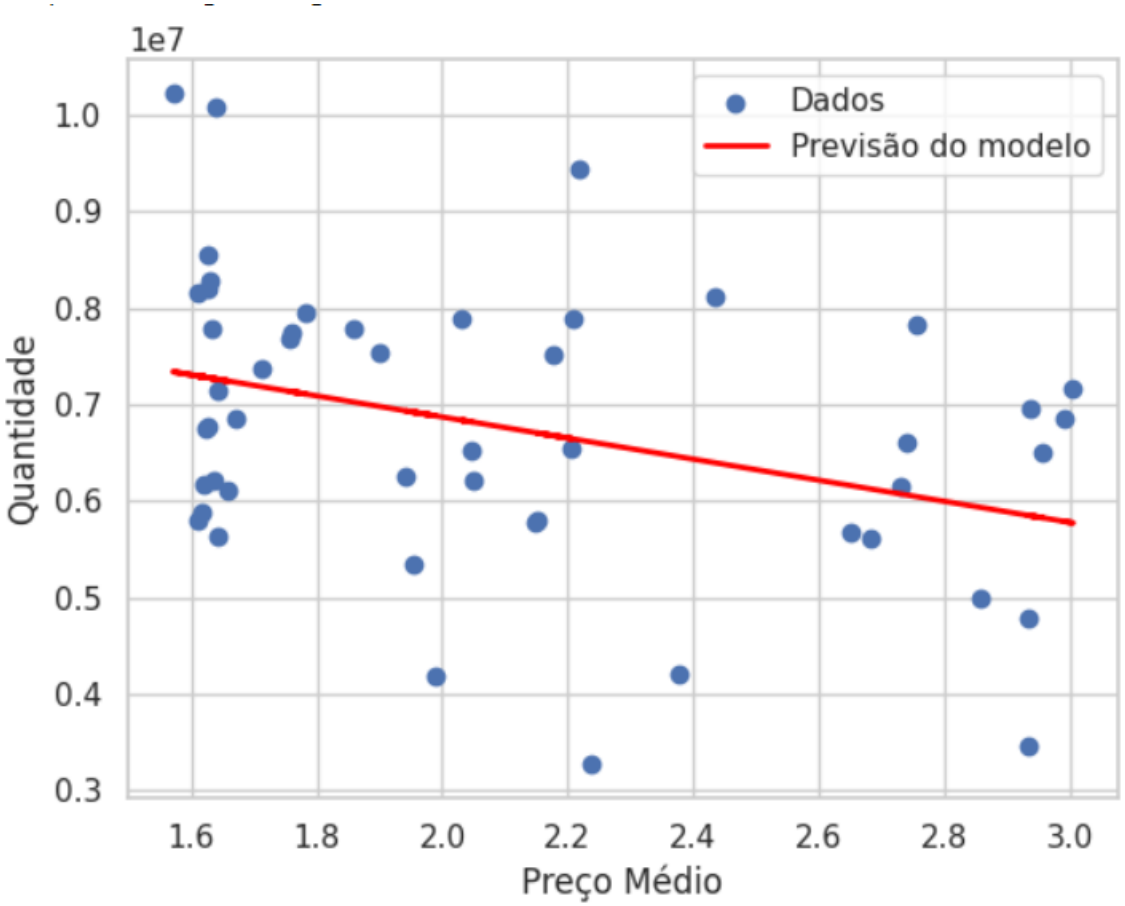


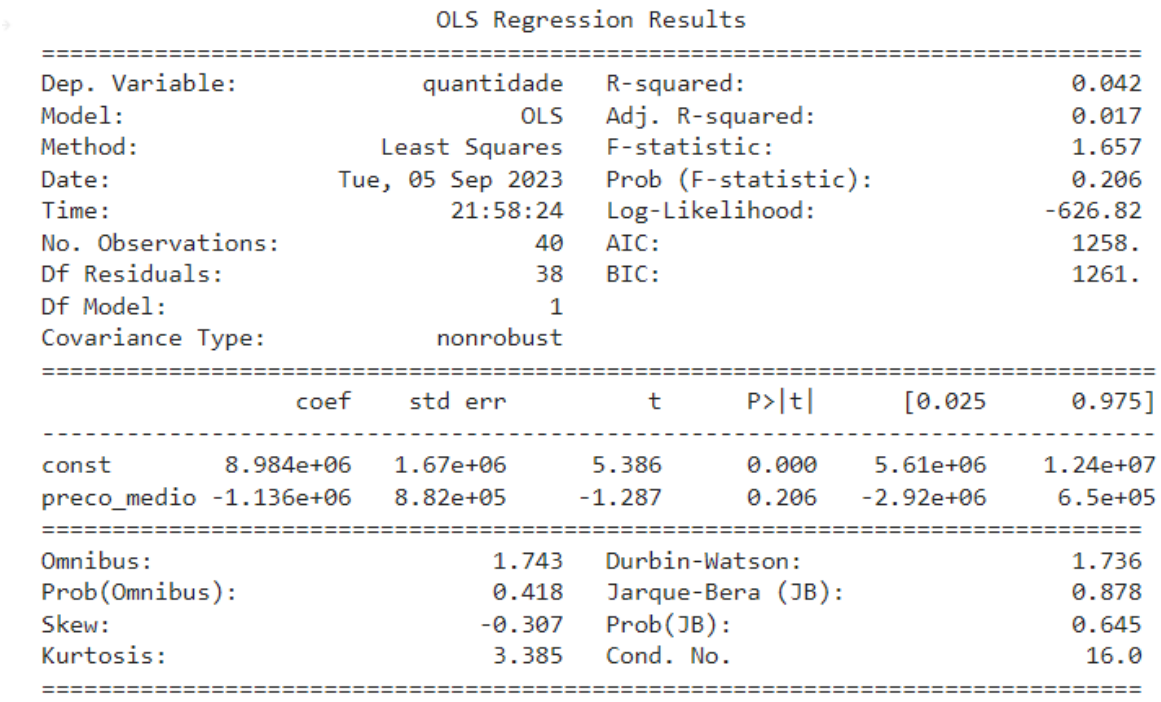
Figura 22 – Regressão Linear

Analisando o gráfico é possível perceber que existe uma relação linear entre o comportamento das variáveis de preço e custo. Mesmo que modesta, essa realação inversa está de acordo com a premissa que se tem sobre a elasticidade. Em que uma variação positiva no preço acarreta em uma queda de demanda de consumo.

As métricas desse modelo foram:

- **RMSE do modelo**: 1368771,22 Kg
- **MAPE do modelo**: 18.65%
- **R^2 do modelo**: 0,13

O sumário do modelo de Regressão Linear teve os seguintes resultados:



OLS Regression Results

Dep. Variable:	quantidade	R-squared:	0.042
Model:	OLS	Adj. R-squared:	0.017
Method:	Least Squares	F-statistic:	1.657
Date:	Tue, 05 Sep 2023	Prob (F-statistic):	0.206
Time:	21:58:24	Log-Likelihood:	-626.82
No. Observations:	40	AIC:	1258.
Df Residuals:	38	BIC:	1261.
Df Model:	1		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	8.984e+06	1.67e+06	5.386	0.000	5.61e+06	1.24e+07
preco_medio	-1.136e+06	8.82e+05	-1.287	0.206	-2.92e+06	6.5e+05

Omnibus:	1.743	Durbin-Watson:	1.736
Prob(Omnibus):	0.418	Jarque-Bera (JB):	0.878
Skew:	-0.307	Prob(JB):	0.645
Kurtosis:	3.385	Cond. No.	16.0

Figura 23 – Sumário do modelo de Regressão Linear

4.1.1 Testando as premissas de linearidade

4.1.1.1 Normalidade dos erros

Para garantir a linearidade da regressão é necessário testar se os erros seguem uma distribuição normal (gaussiana), ou seja, a distribuição dos resíduos deve ser aproximadamente simétrica e centrada em torno de zero.

Para verificar essa hipótese foram criadas as variáveis de erro do treinamento e erro do teste. Esse erro foi calculado diminuindo o valor real do valor predito em cada ponto da análise e tem o seguinte histórico:

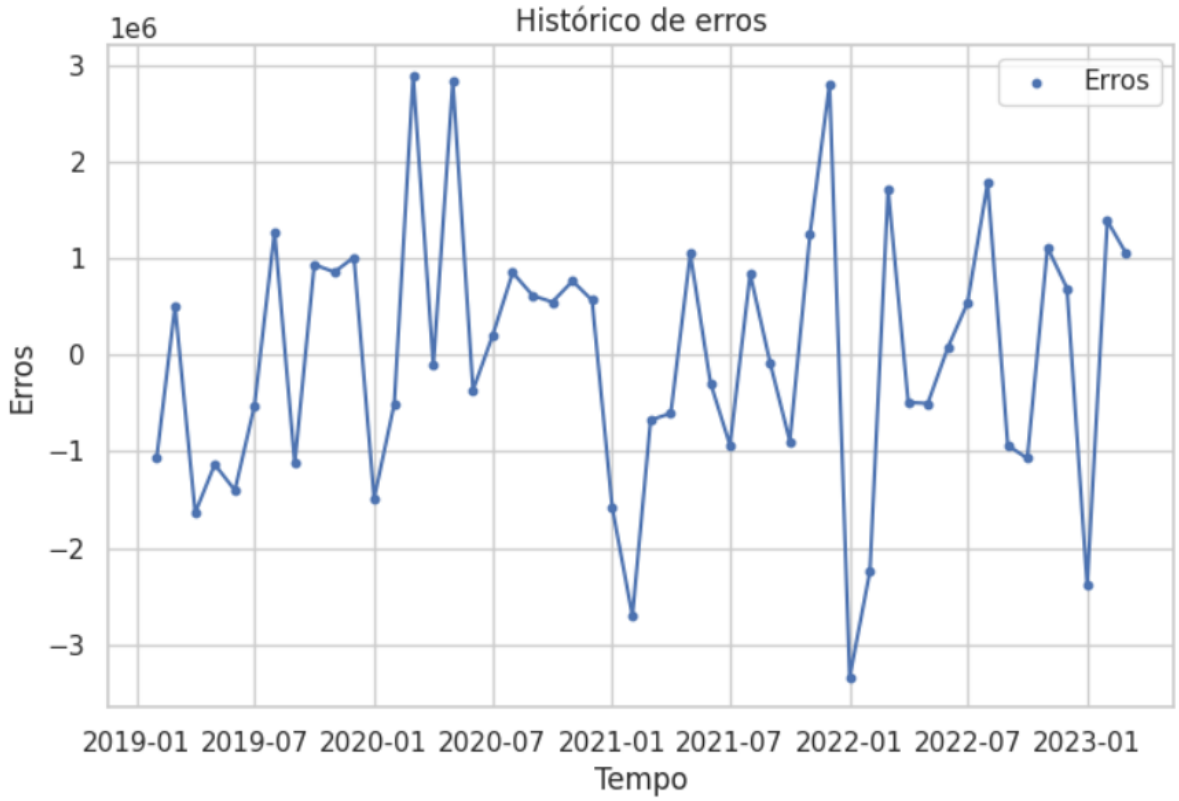


Figura 24 – Histórico dos erros da Regressão Linear

Concatenando essas informações em um histograma, temos:

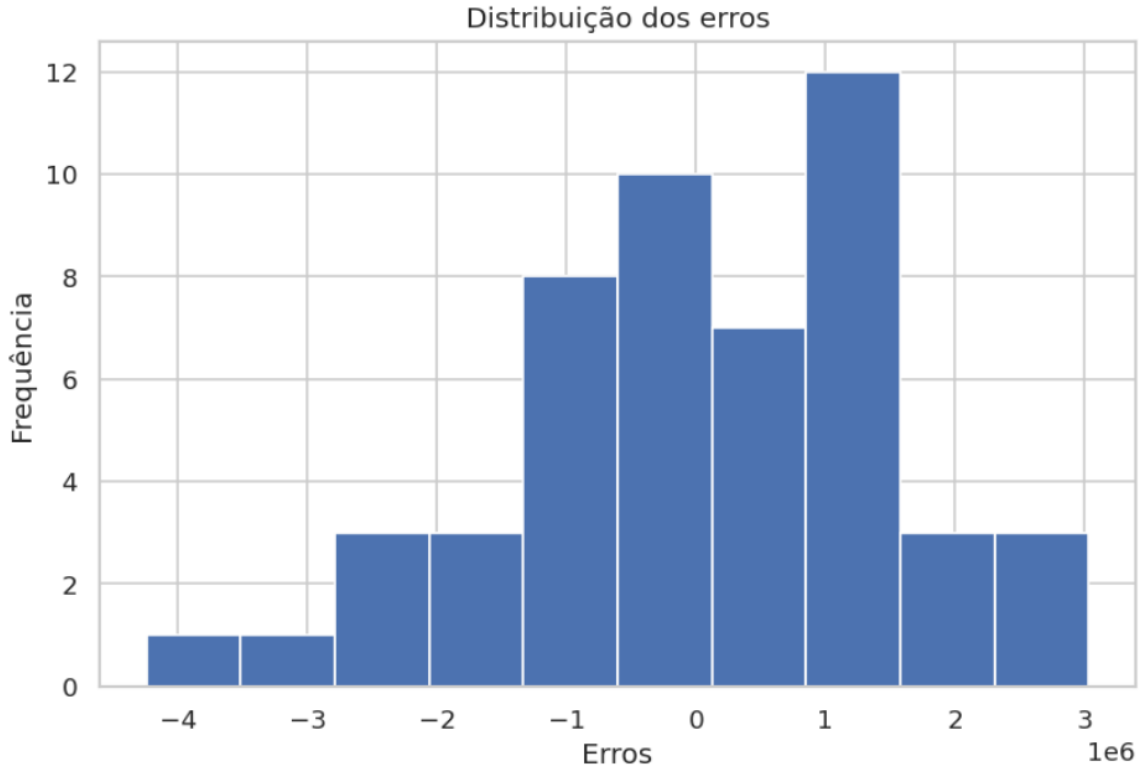


Figura 25 – Dispersão de erros da Regressão Linear

Como temos relativamente pouco histórico de dados, não é possível determinar a normalidade da distribuição apenas visualmente, por isso foi realizado um teste de Kolmogorov-Smirnov para realizar essa determinação. A normalidade da distribuição foi comprovada com um *p-value* dos erros com valor de 0,23.

4.1.1.2 Independência dos erros

O teste de Durbin-Watson realizado para a Regressão Linear, teve como resultado um coeficiente de 1,736. Isso nos mostra que existe uma grande probabilidade dos erros não terem correlação entre si.

4.1.1.3 Homocedasticidade

Através do teste de Breusch-Pagan encontramos um *p-value* como sendo 0.79, muito maior do que 0,05, com isso podemos concluir que não há evidência estatística para rejeitar a homocedasticidade do modelo.

Com esses resultados é possível afirmar que modelos lineares podem ser utilizados para previsão do SKU em questão.

4.2 Regressão Multilinear

4.2.1 Regressão Multilinear com 5 variáveis

Nessa regressão foram utilizadas todas as variáveis independentes à disposição, o código utilizado para implementar o modelo multilinear foi o seguinte,:

```
from sklearn import metrics, preprocessing, model_selection, ensemble
from sklearn.linear_model import LinearRegression, Ridge
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error, mean_absolute_percentage_error, r2_score
import statsmodels.api as sm

# Separando as variáveis independentes (X) da variável dependente (y)
x = sku_21101[[
                    'preco_medio',
                    'custo_variavel_medio',
                    'custo_total',
                    #'preco_medio_conc',
                    'diferenca_preco',
                    'margem_media'
                ]].iloc[1:]
y = sku_21101.sort_values("data")['quantidade'].iloc[1:]


# Criando o modelo de regressão linear
model = LinearRegression()

# Treinando o modelo com os dados de treinamento
model.fit(x_train, y_train)

# Fazendo previsões com o conjunto de teste
y_pred_train = model.predict(x_train)


model.predict(x)
```

Figura 26 – Código de treinamento do modelo de Regressão Multilinear

Graficando a quantidade real vendida no período com o previsto, temos:

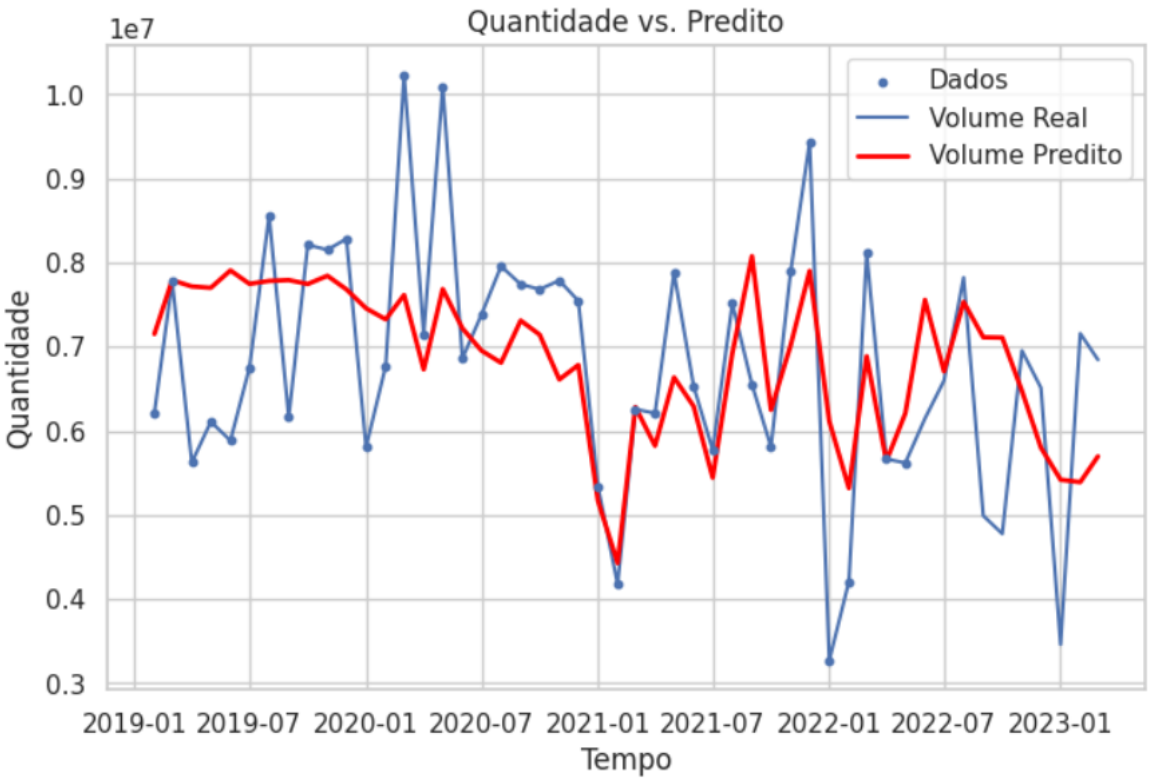


Figura 27 – Comparação Real versus Previsto da Regressão Multilinear com 5 variáveis

O sumário do modelo de Regressão Multilinear teve os seguintes resultados

OLS Regression Results

Dep. Variable:	quantidade	R-squared:	0.358
Model:	OLS	Adj. R-squared:	0.285
Method:	Least Squares	F-statistic:	4.889
Date:	Tue, 05 Sep 2023	Prob (F-statistic):	0.00308
Time:	23:44:33	Log-Likelihood:	-615.98
No. Observations:	40	AIC:	1242.
Df Residuals:	35	BIC:	1250.
Df Model:	4		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	1.294e+07	1.68e+06	7.703	0.000	9.53e+06	1.64e+07
preco_medio	-8.453e+06	2.52e+06	-3.351	0.002	-1.36e+07	-3.33e+06
custo_variavel_medio	2.639e+07	7.94e+06	3.325	0.002	1.03e+07	4.25e+07
custo_total	-1.652e+07	4.72e+06	-3.498	0.001	-2.61e+07	-6.93e+06
diferenca_preco	-6.574e+06	7.82e+06	-0.841	0.406	-2.24e+07	9.3e+06
margem_media	8.063e+06	2.69e+06	3.002	0.005	2.61e+06	1.35e+07

Omnibus:	0.586	Durbin-Watson:	1.699
Prob(Omnibus):	0.746	Jarque-Bera (JB):	0.393
Skew:	-0.239	Prob(JB):	0.822
Kurtosis:	2.912	Cond. No.	2.24e+16

Figura 28 – Sumário do modelo de Regressão Multilinear com 5 variáveis

Por fim, as métricas desse modelo ficaram:

- **RMSE do modelo**: 1181324,04 Kg
- **MAPE do modelo**: 14,53%
- **R^2 do modelo**: 0,36

4.2.2 Regressão Multilinear com 2 variáveis

Como comentado em 3.4.1, no caso da regressão multilinear, precisamos tomar cuidado com o *overfitting* gerado pela quantidade de variáveis e encontrar a melhor combinação delas para realizar a previsão.

Dito isso, foram testadas diversas combinações entre as variáveis independentes e, dessa forma, foi possível melhorar a assertividade desse modelo utilizando as seguintes variáveis para a previsão "Diferença de Preço"e "Custo Variável Médio". Os novos resultados ficaram:

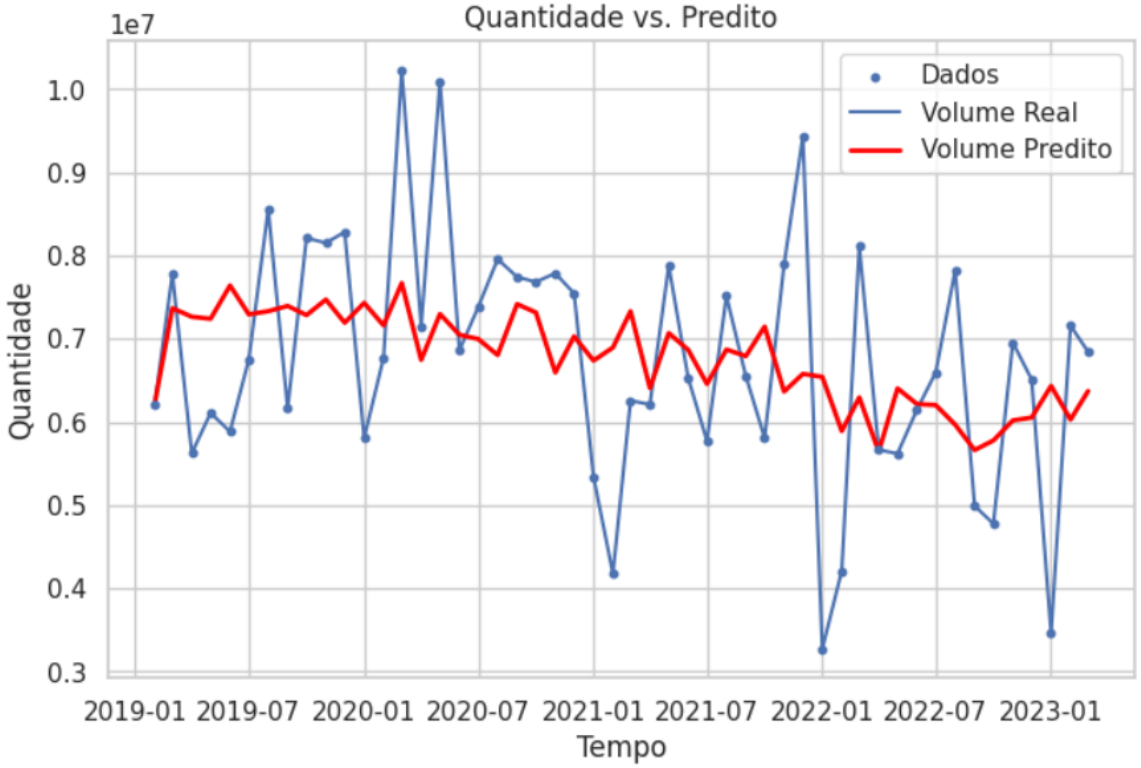
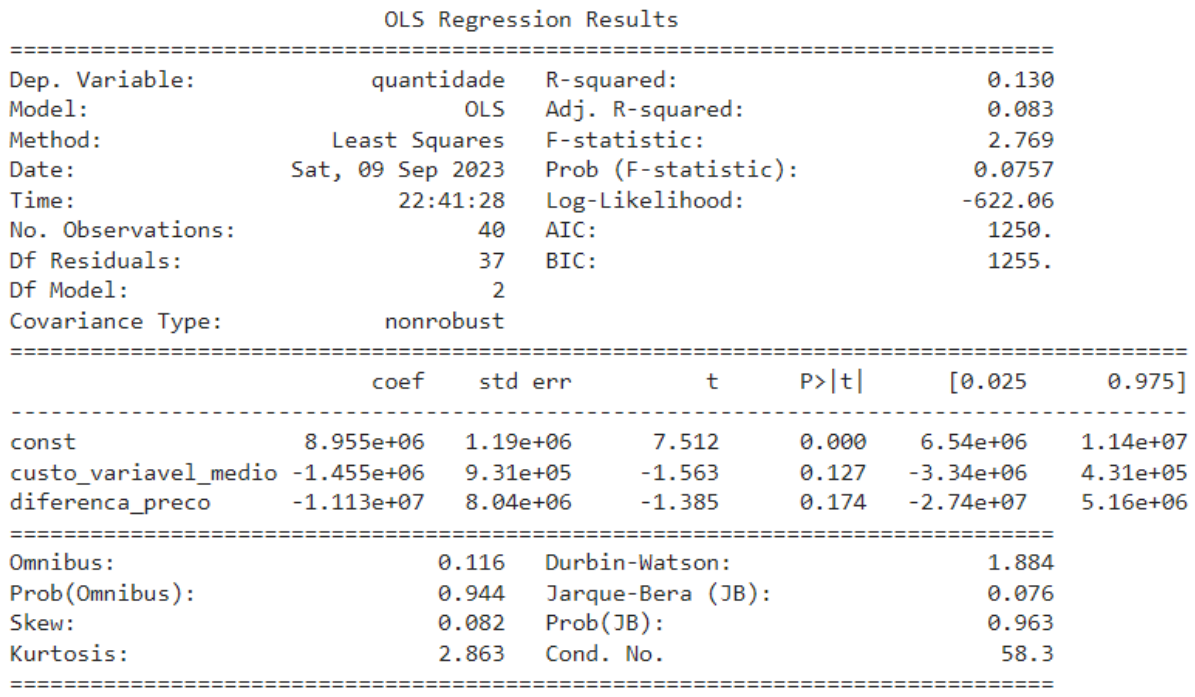


Figura 29 – Comparação Real versus Previsto Multilinear com 2 variáveis

Percebemos como os dados previstos pela regressão multilinear estão menos dispersos do que da regressão linear, principalmente quando falamos dos pontos mais extremos do gráfico.



OLS Regression Results

Dep. Variable:	quantidade	R-squared:	0.130
Model:	OLS	Adj. R-squared:	0.083
Method:	Least Squares	F-statistic:	2.769
Date:	Sat, 09 Sep 2023	Prob (F-statistic):	0.0757
Time:	22:41:28	Log-Likelihood:	-622.06
No. Observations:	40	AIC:	1250.
Df Residuals:	37	BIC:	1255.
Df Model:	2		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	8.955e+06	1.19e+06	7.512	0.000	6.54e+06	1.14e+07
custo_variavel_medio	-1.455e+06	9.31e+05	-1.563	0.127	-3.34e+06	4.31e+05
diferenca_preco	-1.113e+07	8.04e+06	-1.385	0.174	-2.74e+07	5.16e+06

Omnibus:	0.116	Durbin-Watson:	1.884
Prob(Omnibus):	0.944	Jarque-Bera (JB):	0.076
Skew:	0.082	Prob(JB):	0.963
Kurtosis:	2.863	Cond. No.	58.3

Figura 30 – Sumário do modelo de Regressão Multilinear com 2 variáveis

As métricas do modelo com essas variáveis ficaram:

- **RMSE do modelo**: 1358460,18 Kg
- **MAPE do modelo**: 17,97%
- **R² do modelo**:0,14

4.3 ARIMAX

O mesmo teste de combinação de variáveis do modelo multilinear foi usado para o ARIMAX, e as variáveis exógenas que obtiveram os melhores resultados continuaram sendo o "Diferença de Preço"e "Custo Variável", o código utilizado para implementar o modelo ARIMAX foi:

```
# Atribuir preço como variável preditora 'x'" e volume (demanda) como variável de resposta 'y'
x = sku_21101[[
                    'preco_medio',
                    'custo_variavel_medio',
                    'custo_total',
                    #'preco_medio_conc',
                    #'diferenca_preco',
                    'margem_media'
                ]].iloc[1:]
y = sku_21101.sort_values("data")['quantidade'].iloc[1:]

# Fitar o modelo aos dados (fit = treinar)
order = (1, 0, 1)

split_point = int(x.shape[0] * 0.8)
x_train = x.iloc[:split_point]
x_test = x.iloc[split_point:]
y_train = y.iloc[:split_point]
y_test = y.iloc[split_point:]

x_train.index = pd.DatetimeIndex(x_train.index).to_period('M')
y_train.index = pd.DatetimeIndex(y_train.index).to_period('M')
x_test.index = pd.DatetimeIndex(x_test.index).to_period('M')
y_test.index = pd.DatetimeIndex(y_test.index).to_period('M')
x.index = pd.DatetimeIndex(x.index).to_period('M')
y.index = pd.DatetimeIndex(y.index).to_period('M')

model = ARIMA(
    endog=y_train,
    exog=x_train,
    order=order,)

model = model.fit()

# Com o modelo treinado, vamos prever os dados nunca antes vistos
y_pred_train = model.predict(endog=y_train, exog=x_train).to_frame("Quantidade Prevista")
y_pred_test = model.forecast(x_test.shape[0],exog=x_test).to_frame("Quantidade Prevista")
y_pred = y_pred_train.append(y_pred_test)

y.to_frame().join(y_pred).plot(figsize=(8,5), ylim=[0,12000000])

model.summary()

print(y.to_frame().join(y_pred))
```

Figura 31 – Código do modelo ARIMAX

Nesse código, os valores atribuídos à variável *order* são respectivamente os valores (p,d,q) do modelo, nas próximas seções serão apresentados diferentes resultados para esse modelo variando esses valores.

4.3.1 ARIMAX (1,0,1)

Graficando a quantidade real vendida no período com o previsto, temos:

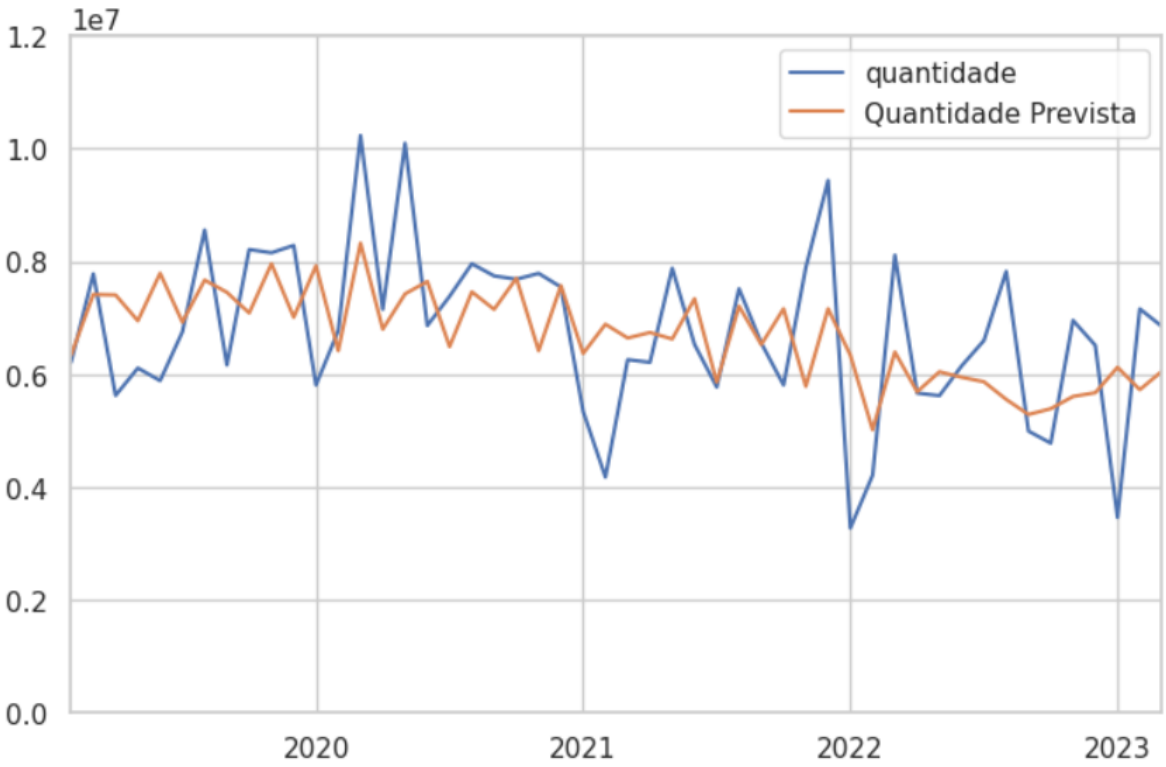


Figura 32 – Comparação Real verus Previsto modelo ARIMAX (1,0,1)

Por fim o sumário do modelo é:

SARIMAX Results

Dep. Variable:	quantidade	No. Observations:	40
Model:	ARIMA(1, 0, 1)	Log Likelihood	-620.273
Date:	Sat, 09 Sep 2023	AIC	1252.547
Time:	23:04:46	BIC	1262.680
Sample:	02-28-2019	HQIC	1256.211
	- 05-31-2022		
Covariance Type:	opg		

	coef	std err	z	P>|z|	[0.025	0.975]
const	8.955e+06	1.36e+06	6.573	0.000	6.29e+06	1.16e+07
custo_variavel_medio	-1.455e+06	1e+06	-1.450	0.147	-3.42e+06	5.12e+05
diferenca_preco	-1.113e+07	9.59e+04	-116.100	0.000	-1.13e+07	-1.09e+07
ar.L1	-0.6984	0.247	-2.827	0.005	-1.183	-0.214
ma.L1	0.9332	0.129	7.242	0.000	0.681	1.186
sigma2	1.898e+12	0.740	2.56e+12	0.000	1.9e+12	1.9e+12

Ljung-Box (L1) (Q):	0.19	Jarque-Bera (JB):	0.27
Prob(Q):	0.67	Prob(JB):	0.87
Heteroskedasticity (H):	1.58	Skew:	-0.18
Prob(H) (two-sided):	0.42	Kurtosis:	2.82

Figura 33 – Sumário do modelo ARIMAX (1,0,1)

- **RMSE do treinamento**: 1324462,01 Kg

- **MAPE do treinamento**: 16,43%
- **R² do treinamento**: 0,22
- **RMSE do teste**: 1359341,39 Kg
- **MAPE do teste**: 20,28%

4.3.2 ARIMAX (2,0,1)

Graficando a quantidade real vendida no período com o previsto, temos:

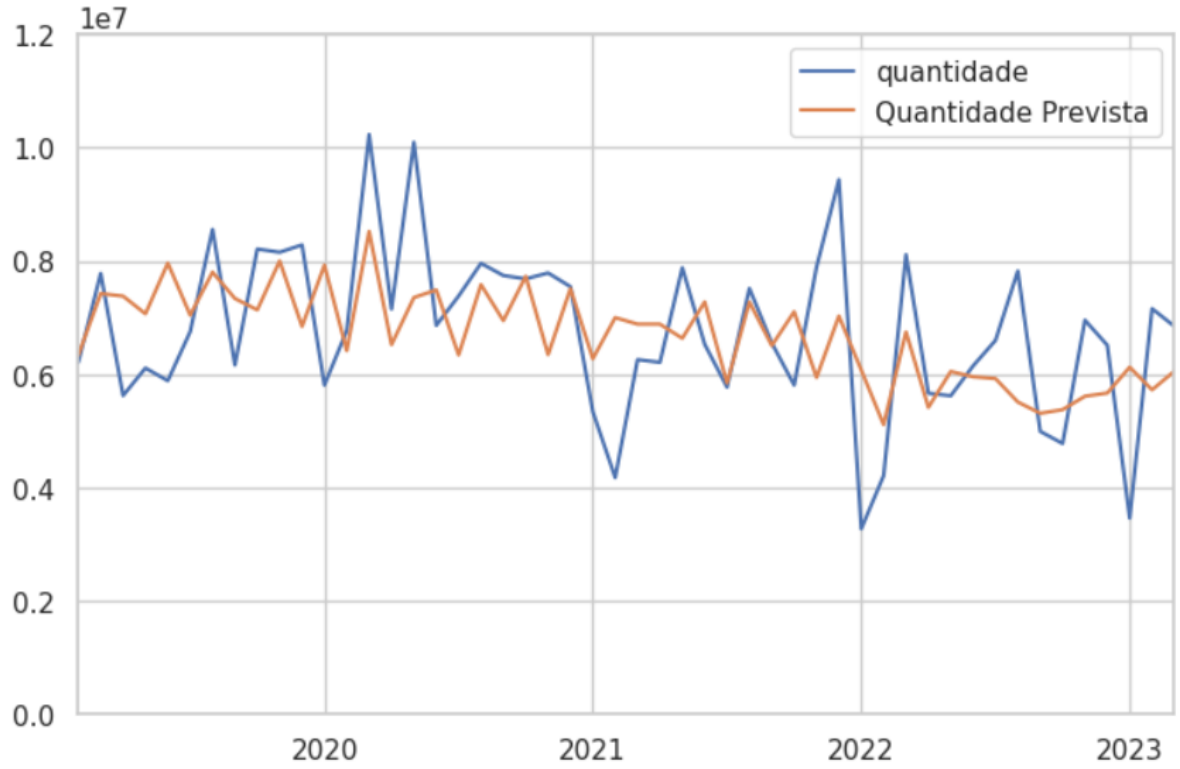


Figura 34 – Comparação Real verus Previsto modelo ARIMAX (2,0,1)

Por fim o sumário do modelo é:

SARIMAX Results

Dep. Variable:	quantidade	No. Observations:	40
Model:	ARIMA(2, 0, 1)	Log Likelihood	-619.963
Date:	Sat, 09 Sep 2023	AIC	1253.926
Time:	23:31:48	BIC	1265.748
Sample:	02-28-2019 - 05-31-2022	HQIC	1258.200
Covariance Type:	opg		

	coef	std err	z	P>|z|	[0.025	0.975]
const	8.955e+06	1.21e+06	7.427	0.000	6.59e+06	1.13e+07
custo_variavel_medio	-1.455e+06	8.95e+05	-1.627	0.104	-3.21e+06	2.98e+05
diferenca_preco	-1.113e+07	8.27e+04	-134.638	0.000	-1.13e+07	-1.1e+07
ar.L1	-0.7357	0.234	-3.144	0.002	-1.194	-0.277
ar.L2	-0.1160	0.168	-0.692	0.489	-0.445	0.213
ma.L1	0.9005	0.135	6.660	0.000	0.636	1.166
sigma2	1.802e+12	0.663	2.72e+12	0.000	1.8e+12	1.8e+12

Ljung-Box (L1) (Q):	0.00	Jarque-Bera (JB):	0.26
Prob(Q):	0.95	Prob(JB):	0.88
Heteroskedasticity (H):	1.37	Skew:	-0.16
Prob(H) (two-sided):	0.58	Kurtosis:	2.77

Figura 35 – Sumário do modelo ARIMAX (2,0,1)

E as métricas calculadas foram:

- **RMSE do treinamento**: 1317264,16 Kg
- **MAPE do treinamento**: 16,61%
- **R² do treinamento**: 0,23
- **RMSE do teste**: 1363407,17 Kg
- **MAPE do teste**: 20,26%

4.3.3 ARIMAX (3,0,2)

Graficando a quantidade real vendida no período com o previsto, temos:

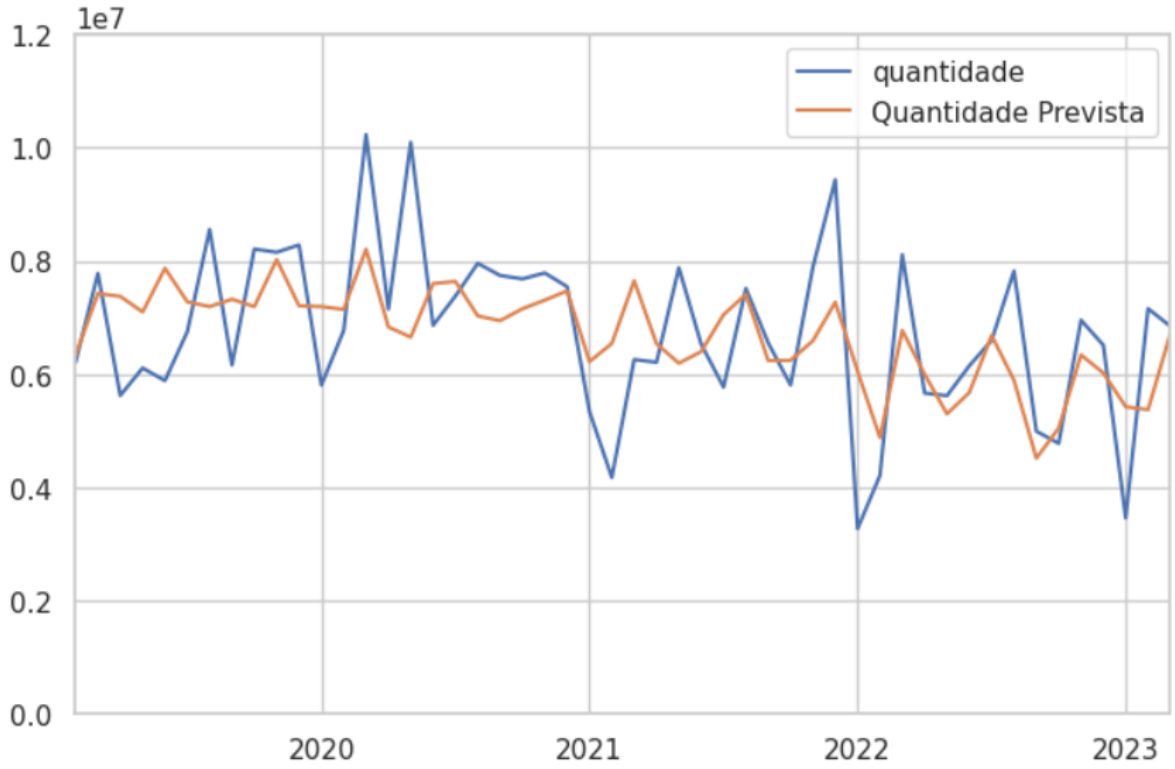


Figura 36 – Comparação Real verus Previsto modelo ARIMAX (3,0,2)

O sumário do modelo é:

SARIMAX Results

Dep. Variable:	quantidade	No. Observations:	40
Model:	ARIMA(3, 0, 2)	Log Likelihood	-619.218
Date:	Sat, 09 Sep 2023	AIC	1256.436
Time:	23:17:04	BIC	1271.636
Sample:	02-28-2019 - 05-31-2022	HQIC	1261.932
Covariance Type:	opg		

	coef	std err	z	P>|z|	[0.025	0.975]
const	8.955e+06	9.8e+04	91.383	0.000	8.76e+06	9.15e+06
custo_variavel_medio	-1.455e+06	1.32e+05	-11.019	0.000	-1.71e+06	-1.2e+06
diferenca_preco	-1.113e+07	148.635	-7.49e+04	0.000	-1.11e+07	-1.11e+07
ar.L1	0.2144	0.228	0.939	0.348	-0.233	0.662
ar.L2	-0.9684	0.175	-5.524	0.000	-1.312	-0.625
ar.L3	0.1827	0.259	0.705	0.481	-0.325	0.691
ma.L1	-0.1539	0.346	-0.445	0.657	-0.833	0.525
ma.L2	0.9674	0.313	3.089	0.002	0.354	1.581
sigma2	1.783e+12	0.000	5.23e+15	0.000	1.78e+12	1.78e+12

Ljung-Box (L1) (Q):	0.02	Jarque-Bera (JB):	0.30
Prob(Q):	0.90	Prob(JB):	0.86
Heteroskedasticity (H):	1.48	Skew:	0.18
Prob(H) (two-sided):	0.49	Kurtosis:	3.23

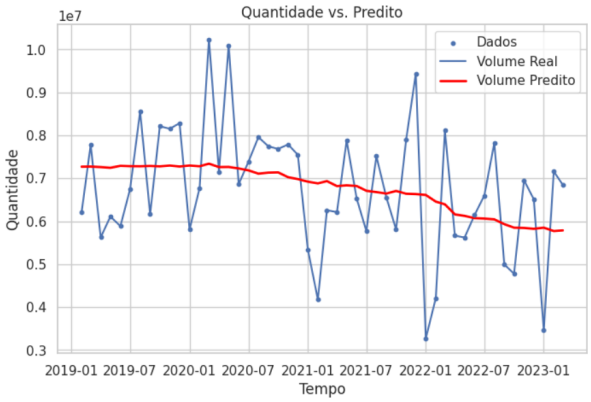
Figura 37 – Sumário do modelo ARIMAX (3,0,2)

E por fim, as métricas calculadas foram:

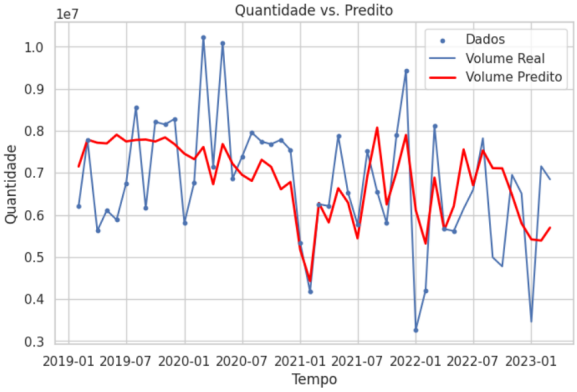
- **RMSE do treinamento**: 1285019,21 Kg
- **MAPE do treinamento**: 15,82%
- **R^2 do treinamento**: 0,27
- **RMSE do teste**: 1092382,51 Kg
- **MAPE do teste**: 14,87%

4.4 Comparação dos resultados

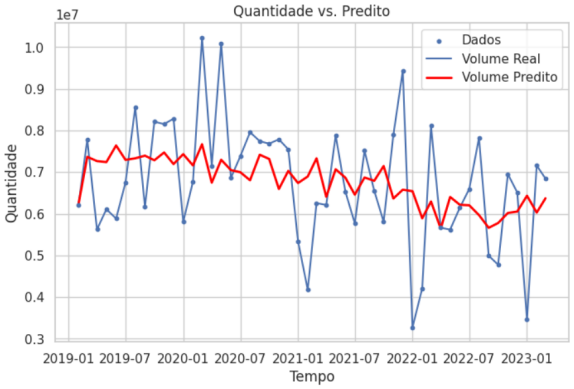
Agregando os gráficos mais relevantes de todos os modelos testados, temos



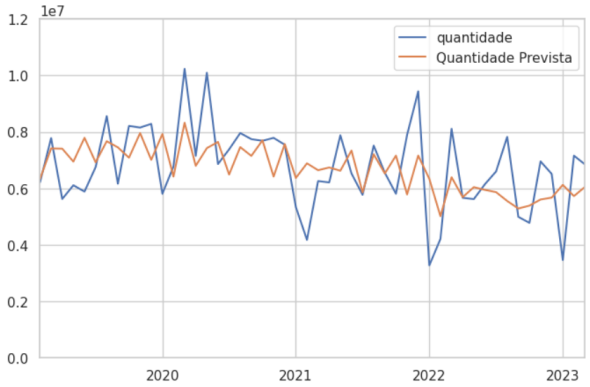
(a) Regressão Linear



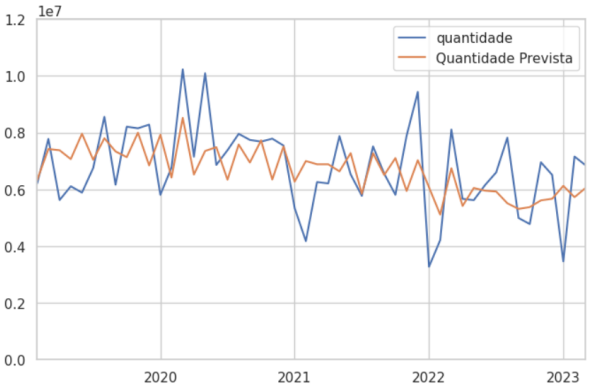
(b) Regressão Multilinear com 5 variáveis



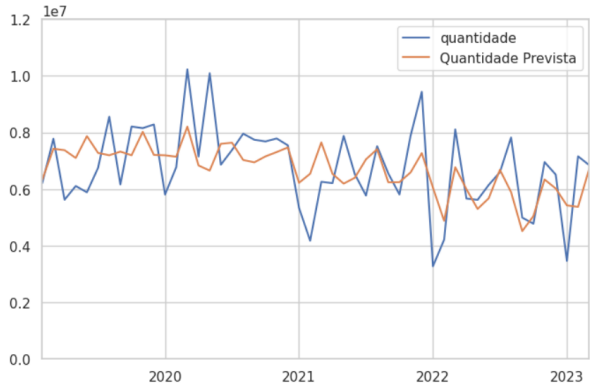
(c) Regressão Multilinear com 2 variáveis



(d) ARIMAX (1,0,1)



(e) ARIMAX (2,0,1)



(f) ARIMAX (3,0,2)

Figura 38 – Comparação dos modelos

É interessante ressaltar como fica claro a evolução dos modelos com o tempo, inicialmente no período de treino, o modelo ainda tem poucos dados para realizar a previsão e ainda não encontra padrões nos dados, com o passar do tempo o modelo está mais treinado e melhora sua previsão muito mais próximo dos dados reais. A partir de maio de 2022, quando o modelo entra em sua fase de testes inicialmente a previsão ainda está próxima do realizado, o que é um ponto positivo visto que, muitas vezes, a empresa busca planejar sua produção um ou dois meses à frente. Em seguida, o acúmulo de erros da previsão faz com que a assertividade dos modelos diminua a partir daquele ponto, afinal, assim como em toda as áreas, quanto mais passos a frente quisermos prever resultados menos precisa será essa previsão.

Modelo	RMSE (Kg)	MAPE	R^2
Regressão Linear	1368771,22	18,85%	0,13
Regressão Multilinear 5 variáveis	1181324,04	14,53%	0,36
Regressão Multilinear 2 variáveis	1358460,18	17,97%	0,14
ARIMAX (1,0,1)	1324462,01	16,43%	0,22
ARIMAX (2,0,1)	1317264,16	16,61%	0,23
ARIMAX (3,0,2)	1285019,21	15,82%	0,27

Tabela 1 – Comparação dos Resultados dos Dados de Treinamento

Modelo	RMSE (Kg)	MAPE
ARIMAX (1,0,1)	1359341,39	20,28%
ARIMAX (2,0,1)	1363407,17	20,26%
ARIMAX (3,0,2)	1092382,51	14,87%

Tabela 2 – Comparação dos Resultados dos Dados de Teste

4.4.1 Principais considerações

- **Teste vs. Treinamento:** A diferença no desempenho entre os conjuntos de treinamento e teste pode ser um forte sinal de *overfitting*.
- **Melhor Desempenho Geral:** com base nos resultados fornecidos, o modelo ARIMAX (3,0,2) teve, de longe, o melhor desempenho geral. Ele teve o melhor R^2 nos testes, o que indica que foi o modelo que melhor se adaptou aos dados, e o MAPE indicando que o modelo está fazendo previsões muito mais precisas em comparação com os outros.

Acredita-se que os principais entraves para atingir melhores resultados com o projeto foram a baixa quantidade de dados, com a granularidade mensal das vendas, obteve-se apenas 40 pontos de dados para treinamento e previsão dos modelos, possivelmente ao longo do tempo com uma maior quantidade de dados os modelos conseguiriam encontrar mais padrões no histórico e, por consequência, melhorar sua assertividade. Como principais melhorias que poderiam ser realizadas na análise, temos:

- Aumentar o tamanho da amostra: Como mencionado anteriormente, um tamanho de amostra pequeno pode ser uma limitação. Com uma maior base de dados seria possível melhorar a capacidade dos modelos de capturar padrões mais robustos.
- Explorar outros modelos: Além dos modelos testados, poderiam ser testados outros modelos não lineares, ou modelos de redes neurais.
- Análise da validação cruzada: avaliar o desempenho dos modelos de forma mais robusta para identificar se o *overfitting* está ocorrendo.

5 CONCLUSÃO

Como apresentado anteriormente, objetivo inicial desse projeto era: melhorar a capacidade de previsão da demanda do cliente de alguma das seguintes formas:

1. **Melhoria do modelo atual**: através de uma aperfeiçoamento no modelo de Regressão Multilinear, com um melhor tratamento dos dados iniciais, tanto das variáveis independentes, quanto das independentes, assim como na forma de calcular os coeficientes da regressão.

2. **Implementação de um modelo alternativo**: baseado na revisão bibliográfica realizada, o modelo com maior potencial para aumentar a assertividade da previsão é o ARIMAX. Isso ocorre pois ele alia uma boa identificação de padrões em séries temporais com a inclusão de variáveis externas que melhoram o ajuste do modelo. Em caso de disponibilidade de tempo excedente no cronograma, serão realizados testes adicionais em outros modelos, tais como os Modelos Lineares Generalizados e/ou modelos de redes neurais.

É possível afirmar que esse objetivo foi cumprido com ressalvas. Atualmente o modelo global implementado para todos os produtos da companhia, utilizando o modelo multilinear, tem um MAPE de 27,8%, com esse projeto nós conseguimos reduzir esse valor para um dos produtos mais vendidos da companhia, além disso implementamos um modelo alternativo que também obteve um resultado consideravelmente melhor que o existente.

Como ressalvas podemos apontar o fato de que em uma possível expansão da análise para mais SKUs, iríamos enfrentar maiores problemas de falta de dados para produtos menos relevantes, assim como, provavelmente, encontraríamos produtos com distribuições nas quais modelos lineares não poderiam ser aplicados.

Para vias da consolidação dos conhecimentos e ferramentas aprendidas durante a graduação em Engenharia Fisica, com possível aplicação futura no mercado, acredita-se que o projeto obteve sucesso em sua excução.

6 CRONOGRAMA

Para desenvolvimento do projeto foi estipulado, e seguido, um cronograma de trabalho detalhado viabilizando sua execução até o fim do semestre, na metade de setembro de 2023. Segue abaixo esse cronograma:

Junho	**Tarefas**
Semana 1	Elaboração do relatório do TCC I
Semana 2	Preparação da apresentação do TCC I
Semana 3	Revisão do relatório e apresentação
Semana 4	Refinar análise inicial dos dados e pré-processamento
Julho	
Semana 1	Preparar dados e dividir em conjunto de treinamento e teste
Semana 2	Implementar modelo de Regressão Multilinear e avaliar desempenho
Semana 3	Implementar o modelo ARIMAX
Semana 4	Teste e otimização do modelo
Agosto	
Semana 1	Comparar modelos e selecionar o melhor para otimização
Semana 2	Interpretar resultados e documentar metodologia
Semana 3	Elaboração relatório do TCC II
Semana 4	Preparação da apresentação do TCC II
Setembro	
Semana 1	Revisão do relatório e apresentação
Semana 2	Apresentação do projeto

Tabela 3 – Cronograma de Execução do Projeto

REFERÊNCIAS

AL-ZU'BI, Z.; HEIZER, J.; RENDER, B. *Operations Management*. [S.l.: s.n.], 2013. ISBN 9781447903031.

APRIX. 2020. Disponível em: <https://www.aprix.com.br/home>.

ARMSTRONG, J. The forecasting dictionary. *Principles of Forecasting: A Handbook for Researchers and Practitioners*, 01 2001.

COLABORATOY. 2023. Disponível em: <https://colab.research.google.com/>.

HOSHMAND, A. R. *Business and Economic Forecasting for the Information Age: A Practical Approach*. London: Greenwood Publishing Group, Inc., 2002.

JAMES, G. et al. *An Introduction to Statistical Learning: with Applications in R*. Springer, 2013. Disponível em: <https://faculty.marshall.usc.edu/gareth-james/ISL/>.

MAKRIDAKIS, S.; WHEELWRIGHT, S.; MCGEE, V. E. *Forecasting: Methods and Applications*. [S.l.: s.n.], 1979.

PANDASLIBRARY. 2023. Disponível em: <https://pandas.pydata.org/docs/user_guide/index.html>.

PYTHON. 2023. Disponível em: <https://www.python.org/>.

SEABOLD, S.; PERKTOLD, J. statsmodels: Econometric and statistical modeling with python. In: *9th Python in Science Conference*. [s.n.]. Disponível em: <https://www.statsmodels.org/stable/generated/statsmodels.stats.diagnostic.kstest_exponential.html#statsmodels-stats-diagnostic-kstest-exponential>.

SEABOLD, S.; PERKTOLD, J. statsmodels: Econometric and statistical modeling with python. In: *9th Python in Science Conference*. [s.n.]. Disponível em: <https://www.statsmodels.org/stable/generated/statsmodels.stats.stattools.durbin_watson.html#statsmodels.stats.stattools.durbin_watson>.

SEABOLD, S.; PERKTOLD, J. statsmodels: Econometric and statistical modeling with python. In: *9th Python in Science Conference*. [s.n.], 2010. Disponível em: <https://www.statsmodels.org/stable/diagnostic.html#normality-and-distribution-tests>.

SEABOLD, S.; PERKTOLD, J. statsmodels: Econometric and statistical modeling with python. In: *9th Python in Science Conference*. [s.n.], 2010. Disponível em: <https://www.statsmodels.org/stable/generated/statsmodels.stats.diagnostic.het_breuschpagan.html>.

SKETCHPLANATIONS. *The Bullwhip Effect*. 2020. Disponível em: <https://sketchplanations.com/the-bullwhip-effect>.

SUCKY, E. The bullwhip effect in supply chains—an overestimated problem? *International Journal of Production Economics*, v. 118, n. 1, p. 311–322, 2009. ISSN 0925-5273. Special Section on Problems and models of inventories selected papers of the fourteenth International symposium on inventories. Disponível em: <https://www.sciencedirect.com/science/article/pii/S092552730800279X>.