
.CENTRO UNIVERSITÁRIO SANTO AGOSTINHO- UNIFSA

PRÓ-REITORIA DE ENSINO
NÚCLEO DE APOIO PEDAGÓGICO-NUAPE
COORDENAÇÃO DO CURSO DE ENG. DE SOFTWARE

**Disciplina: Ciência de Dados**
**Professora: Heloisa Guimarães**

**Marcus Mikael Rodrigues Vieira**
**Marcos Andre dos Santos Soares**

# Análise e Pré-Processamento de Dados com o Dataset Olist E-Commerce

# 1. Contextualização

Para análise será utilizado o Olist Brazilian E-Commerce Dataset, uma base real com informações de pedidos, produtos e entregas entre 2016 e 2018.

A partir dessa base, é possível analisar comportamentos relacionados a atrasos, categorias problemáticas, discrepâncias de preços e fretes, além de identificar padrões logísticos que influenciam a satisfação do consumidor. O objetivo é aplicar o ciclo completo de pré-processamento e análise exploratória, preparando os dados para revelar insights sobre o funcionamento do e-commerce.

# 2. Apresentação dos datasets adotados

Para a análise foi utilizado os datasets do Olist Brazilian E-Commerce Dataset do Kaggle, sendo extraídos três deles:

- **Olist_orders_dataset.csv**: Contendo 99.441 registros de pedidos, com informações: ID do pedido, ID do cliente, status, datas de compra, aprovação, entrega e estimativa de entrega, tendo um período de Setembro de 2016 a Agosto de 2018.

- **Olist_order_items_dataset.csv**: Contendo 112.650 itens de pedidos com informações sobre ID do pedido, ID do item, ID do produto, ID do vendedor, preço, valor do frete e data limite de envio. Representando os produtos individuais dentro de cada pedido.

- **Olist_products_dataset.csv**: Contendo 32.951 produtos únicos com informações de ID do produto, categoria, dimensões físicas (peso, altura, largura, comprimento), tamanho do nome, descrição e quantidade de fotos.

# 3. Aplicação do Ciclo de Vida da Ciência de Dados

Identificar fatores que afetam a experiência e satisfação do cliente no e-commerce brasileiro, com foco em atrasos, preços, fretes e categorias problemáticas. Coletando os dados dos três datasets extraídos que possuem informações de pedidos, produtos e logística entre 2016-2018.

# 4. Exploração dos Dados (EDA)

## 4.1 Visão Geral do Dataset

Após a exploração inicial desses datasets, nota-se que possui um total de registro de 113.425, distribuídos em 22 colunas sendo 10 numéricas e 12 categóricas.

Com as estatísticas obtidas, se tem essas informações:

## 4.2 Estatísticas Descritivas

**COLUNA PREÇO:**

Média: R$ 120,65

Mediana: R$ 74,99

Mínimo: R$ 0,85

Máximo: R$ 6.735,00

Desvio padrão: R$ 183,63

**COLUNA VALOR DO FRETE:**

Média: R$ 19,99

Mediana: R$ 16,26

Mínimo: R$ 0,00

Máximo: R$ 409,68

**COLUNA PESO DO PRODUTO:**

Média: 2.093,82g = 2kg

Mediana: 700g

Mínimo: 2g

Máximo: 40.425g = 40kg

**Grande variação entre produtos leves e pesados.**

**COLUNA ITENS POR PEDIDO:**

Média: 1,20 itens

Mediana: 1 item

Máximo: 21 itens

**A maioria dos pedidos contém apenas 1 produto, chegando até 21.**

## 4.3 Análise de Correlação

Com a criação do gráfico heatmap para análise de correlação, foi identificado essas fortes correlações:

**product_weight_g <-> freight_value:** 0.61 (correlação positiva forte) - Ou seja, produtos mais pesados resultam em fretes significativamente mais altos.

**product_height_cm <-> freight_value:** 0.39 (correlação razoável) - Altura do produto influencia no custo de frete.

**product_width_cm <-> freight_value:** 0.32 (correlação razoável) - Largura impacta moderadamente no frete.

**price <-> freight_value:** 0.41 (correlação razoável) - Produtos mais caros tendem a ter fretes mais altos.

Além dessas correlações sendo mais fortes, algumas correlações baixas foram identificadas:

**price <-> product_weight_g:** 0.34 (correlação fraca) - O preço não é determinado particularmente pelo peso do produto.

**Dimensões físicas <-> price:** correlações < 0.25 - Produtos pequenos podem ser caros (eletrônicos, joias) e vice-versa.

Implicação: O modelo de precificação não se baseia apenas em características físicas, mas em valor agregado, marca, categoria, etc.

Nota-se também uma presença de preços entre 30 e 150 R$ e a presença de outliers.

## 4.5 Principais Descobertas da EDA

Diante de todas essas informações, as principais descobertas foram: Dataset bem complexo, com 113 mil registros. Tendo uma forte relação entre características físicas (peso/volume) e custo de frete, onde a maioria dos

pedidos, são apenas 1 item e ocorre uma grande variedade. Por exemplo: produtos de (2 kg a 40 kg) como (R$0.85 a R$6.735) que geram outliers, porém naturais, já que se trata de um e-commerce.

# 5. Limpeza de Dados

## 5.1 Verificação de Duplicatas

O uso de duplicated.sum() para visualizar linhas duplicadas, foi retornada 0, ou seja, não possui nenhuma linha igual a outra, assim cada registro representa uma transação única.

## 5.2 Verificação de Inconsistências

Preços sempre tem que ser positivos, a análise deu 0 registros com preços 0 ou negativos, isso mostra que todos os produtos estão de forma válida.

No frete, ocorreu igual, nenhum registro com frete negativos, ou zero. O valor zero poderia ser tratado como um frete grátis. Isso conclui que dados de frete estão consistentes.

O peso do produto, na análise feita de identificar se existe peso com valor 0, algo que não é pra ocorrer, pois todo produto físico exige um peso. Nisso, foram registrados 8 registros com peso zerado.

## 5.3 Tratamento de Inconsistências

Decisão tomada: Remover esses 8 registros, pois os mesmos equivalem apenas a 0.007% de 113.425, não alterando nada praticamente. Preencher com peso não seria uma boa opção, já que não sabemos o real, tornando dados inventados.

Impossível um produto com 0g custar por exemplo 15,00R$. Com isso, remover por inconsistência.

## 5.4 Tratamento de Valores Ausentes (Nulos)

No tratamento de valores nulos, foram identificados 10007 registros nulos, o que equivale a 0,40% do dataset. Comparado com a quantidade de registros, é um número bem pouco de valores ausentes.

### 5.4.1 Diagnóstico Inicial

Distribuição de nulos por coluna:

**order_delivered_customer_date** - 2.454 nulos, com o percentual de 2.18%

**product_description_lenght** - 1.586 nulos, com percentual de 1.41%

**product_category_name** - 1.586 nulos, com o percentual de 1.41%

**product_name_lenght** - 1.586 nulos, com percentual de 1.41%

**product_photos_qty** - 1.586 nulos, com percentual de 1.41%

**order_delivered_carrier_date** - 1.194 nulos, com percentual de 1.06%

**order_approved_at** - 15 nulos, com percentual de 0.01%

**A maioria está concentrada em datas de entrega e informações de produtos.**

### 5.4.2 Estratégias de Tratamento Aplicadas

Para esses nulos, foi usado um tratamento para cada coluna:

Coluna: product_category_name

Nulos: 1.586 (1.41%)

Método: Preenchimento com valor constante "**desconhecido**"

**Motivo: Um produto sem categoria, não que dizer que é algo errado, ou seja, são casos reais no sistema, como não categorizados ou de categoria genérica. Por isso a criação da categoria desconhecido mantendo esses registros.**

**Resultado: 1.586 nulos - 0 nulos**

Para as colunas numéricas, essas foi a decisões tomadas:

Colunas: product_name_lenght (tamanho do nome)

product_description_lenght (tamanho da descrição)

product_photos_qty (quantidade de fotos)

Nulos: 1.586 em cada coluna

Método: Preenchimento com 0

**Motivo: Dados nulos nas colunas indicam que não possui, sendo preenchido com 0 caracteres, e 0 fotos. Diferente de imputar como por**

**exemplo uma média, não iria fazer sentido ter 45.5 fotos. E ainda vai permitir análises sobre produtos com informações incompletas.**

**Resultado: 4.758 nulos - 0 nulos**

Tratamento para as dimensões físicas dos produtos:

Colunas:

product_weight_g (peso em gramas)

product_height_cm (altura)

product_length_cm (comprimento)

product_width_cm (largura)

Método: Preenchimento com MEDIANA

**Motivo: O uso de media não daria certo, pois tem presença de outliers, e ele iria distorcer por causa dos valores extremos. Já a mediana é robusta a outliers, no caso representado o valor central. Ex: Mediana de peso: 700g (mais realista que média de 2.093g).**

**Valores de mediana utilizados:**

**Peso: 700g**

**Altura: 13cm**

**Comprimento: 25cm**

**Largura: 20cm**

Remoção de registros:

Colunas:

product_id (ID do produto)

seller_id (ID do vendedor)

price (preço)

freight_value (frete)

Método: Remover  linhas com nulos nessas colunas

**Motivo: Sem esses dados, não se sabe a existência do pedido. Por exemplo: sem product_id: não sabemos o que foi vendido e segue o**

**mesmo para as outras colunas. Imputar não acho uma boa opção, pois não se torna algo confiavel.**

**Decisão: Pedidos incompletos são a mesma coisa que pedidos inválidos.**

**Resultado: Registros removidos: 0**

Datas de aprovação:

Coluna: order_approved_at(data de aprovação do pedido)

Nulos: 15 (0.01%)

Método: Preencher com order_purchase_timestamp

**Motivo: Na maioria das vezes, a aprovação ocorre logo após a compra.**

**Resultado: 15 nulos - 0 nulos**

Datas de entrega:

Colunas:

order_delivered_carrier_date (entrega à transportadora)
order_delivered_customer_date (entrega ao cliente)

Nulos mantidos:

order_delivered_carrier_date: 1.194 nulos (1.06%)
order_delivered_customer_date: 2.454 nulos (2.18%)

Decisão: Não tratar esses valores nulos, sendo assim mantidos.

**Motivo: Esses nulos podem ser informações como, pedidos que ainda não foram entregues, ou pedidos que o cliente cancelou quando estava em andamento, faz parte de um e-commerce. E os mesmos servem para criação de novas features.**

**Adicionar datas estimadas ou remover esses valores, não é uma boa opção, já que possui vários registros. Preencher como entregue por exemplo, ia distorcer análises de satisfação.**

### 5.4.3 Resultado Final do Tratamento de Nulos

Resumo das Decisões:

Categóricos - "desconhecido" (1.586 tratados)

Numéricos textuais - 0 (4.758 tratados)

Dimensões - mediana robusta a outliers (variável)

Dados críticos - remover se nulo (0 removidos)

Datas de aprovação - data de compra (15 tratados)

Datas de entrega - MANTER nulos (3.648 mantidos)

**Total: 6.359 nulos tratados + 3.648 nulos propositais mantidos**

### 5.4.4 Detecção e Tratamento de Outliers

Para a analise desses oultliers, utilizamos o metodo de IQR, já que o mesmo é robusto e amplamente usado em ciencia de dados, com fácil interpretação e replicação.

Sua formula se caracteriza como:

Q1 = percentil 25

Q3 = percentil 75

IQR = Q3 - Q1

Limite Inferior = Q1 - 1.5 × IQR

Limite Superior = Q3 + 1.5 × IQR

Logo após essa análise, foram encontrados esses valores;

Coluna Preço:

Q1 - R$ 39,90

Q3 - R$ 134,90

IQR - R$ 95,00

Limite superior - R$ 277,40

Outliers detectados - 15.438

Percentual - 13,71%

**Exemplos de outliers: R$ 500, R$ 1.200, R$ 3.500, R$ 6.735**

**Coluna frete:**

Q1 - R$ 13,08

Q3 - R$ 21,15

IQR - R$ 8,07

Limite superior - R$ 33,26

Outliers detectados - 10.524

Percentual - 9,34%

**Coluna Peso:**

Q1 - 300g

Q3 - 1.800g

IQR - 1.500g

Limite superior - 4.050g

Outliers detectados - 6.892

Percentual - 6,12%

**Exemplos: 10kg, 25kg, 40kg (móveis, eletrodomésticos)**

**Coluna comprimento:**

Q1 - 18cm

Q3 - 38cm

IQR - 20cm

Limite superior - 68cm

Outliers detectados - 3.617

Percentual - 3,21%

**Coluna Altura:**

Q1 - 8cm

Q3 - 20cm

IQR - 12cm

Limite superior - 38cm

Outliers detectados - 7.670

Percentual - 6,81%

**Coluna Largura:**

Q1 - 15cm

Q3 - 30cm

IQR - 15cm

Limite superior - 52,50cm

Outliers detectados - 2.563

Percentual - 2,28%

Com todos esses outliers a nossa decisão foi manter todos eles, por se tratar de dados naturais enem erros, justificativas  para melhor entendimento:

- Valores extremos são naturais do e-commerce brasileiro
- Produtos variam de R$ 0,85 a R$ 6.735 (bijuterias a móveis)
- Pesos variam de 2g a 40kg (brincos a geladeiras)
- Outliers não são erros de medição
- Notebook de R$ 6.000 é real, não erro
- Sofá de 30kg é fisicamente correto
- Remoção causaria problemas como, perda de 10-15% dos dados, distorção na correlação peso × frete (0.61)
- Produtos caros têm dinâmica de satisfação diferente
- Fretes altos impactam decisão de compra
- Importante para identificar categorias problemáticas

## 6. Conversão e padronização de tipos

Um dos problema identificados no dataset é a presença de 6 colunas relacionadas a data que estão sendo tratadas com o tipo ‘object’ e o ‘ordem_item_id’ que se trata de identificação do pedido armazenados como float. Tendo impactos em por exemplo:

Impossível calcular diferenças entre datas, operações temporais não funcionam e possíveis erros em análises, e a questão de float ter um uso maior da memória.

As conversões realizadas com base nessa análise, foi de as colunas relacionadas a datas ser do tipo(datetime64) onde valores inválidos se tornaram NaT (Not a Time).

Com essa troca de tipo, desbloqueava novas análises como, permite calcular tempo de entrega em dias, possibilitando identificar atrasos. Análises por período (mês, dia da semana), e ajudando a criação de novas features temporais.

Na parte dos ID que foram tratados para o formato em inteiro, usado de forma correta e também ocupa menos memória.

**Conclusões:**
**Dataset pronto para análises temporais**
**Tipos semanticamente corretos**
**Memória otimizada**
**Base para feature engineering temporal**

## 7. Tratamento de dados categóricos e textos

A coluna analisada foi a de categoria do produto(product_category_name) onde possui 74 categorias e muitas delas raras que aparecem poucas vezes, que impactaria no One-Hot direto, já que o mesmo ia usar mais de 50 colunas para forma binária, tornando algo muito extenso.

A estrategia usada foi agrupa esses categorias e  manter apenas as 20 mais frequentes. Motivo: o top 20 já representa uma boa parte dos produtos, a questão de ter menos colunas, e a melhora generalizada, focando nas categorias relevantes.

**Resultado: Categorias únicas: 74 -> 21 (20 + 'other') Registros afetados: mais ou menos 15% viram outros(‘other’) tornando um dataset mais compacto e eficiente.**

## 8. Codificação de dados categóricos

Método que estamos usando é o One Hot Encoding, que vai transformar cada categoria em um coluna binária(0 ou 1)

Exemplo: True = 1 | False = 0

| | price | freight_value | cat_automotivo | cat_bebes | cat_beleza_saude |
|---|---|---|---|---|---|
| 0 | 29.99 | 8.72 | False | False | False |
| 1 | 118.70 | 22.76 | False | False | False |
| 2 | 159.90 | 19.22 | True | False | False |
| 3 | 45.00 | 27.20 | False | False | False |
| 4 | 19.90 | 8.72 | False | False | False |

**Função: pd.get_dummies()**

**Parâmetros:**

**columns=['product_category_name']**

**prefix='cat' (colunas ficam: cat_categoria)**

**drop_first=False (mantém todas as colunas)**

Após essa codificação de dados categóricos, se obteve um resultado, a coluna original foi removida criando novas 21 novas colunas binárias do One Hot Encoding. Tornando o dataset totalmente numérico, sendo compatível com algoritmos de machine learning.

## 9. Normalização e padronização

Essa normalização será usada para padronizar as variáveis que possuem escala muito diferentes(outliers), mas como eu não vou alterar esses valores, a normalização vai seguir outra caminho(csv), não alterando o original.

Problemas com base na normalização:

- Variáveis em escalas muito diferentes:

product_weight_g: 2 a 40.425 (escala de milhares)

price: R$ 0,85 a R$ 6.735

freight_value: R$ 0 a R$ 409
Dimensões: 2cm a 118cm

**Metodo ulizado para essa normalização foi o StandardScaler (Z-score), que vai normaliza para:**
**Média = 0**
**Desvio padrão = 1**
**Valores típicos entre -3 e +3**

**Conclusão: Caso queríamos tratar os outliers, a melhor opção seria usar o Z-Score, pois o mesmo preserva distâncias relativas.**
**Exemplo**
**price: Antes: R$ 0,85 a R$ 6.735 (média: R$ 120,65) Depois: 0,65 a 36,03 (média: 0,00)**

## 10. Seleção de atributos

O objetivo dessa seleção de atributos, é reduzir dimensionalidade removendo: Variáveis com baixa variância (pouca informação), Variáveis redundantes (alta correlação), Variáveis irrelevantes (IDs sem valor preditivo).

Foram utilizados três métodos para essa seleção:
**Variance Threshold** - Remove variáveis com variância < threshold.
- Threshold: 0.01 (1%)
Aplicado em dados normalizados
Variáveis com pouca variação são descartadas

**Resultado informado de que todas as variáveis têm variação adequada, dataset bem estruturado, sem colunas "mortas" ou quase constantes.**

**Correlação -** Remove variáveis com correlação > 0.90.
Threshold: 0.90 (90%)
Matriz de correlação entre todas as variáveis

Se duas variáveis têm correlação >0.90, remover uma

**Resultado, cada variável traz informação única. Não há redundância significativa.**

**Filtros Simples** - Remove colunas irrelevantes por natureza

Cinco colunas removidas:
**order_id** - identificador único, sem valor preditivo
**customer_id** - identificador único, sem valor preditivo
**product_id** - identificador único, sem valor preditivo
**seller_id** - identificador único, sem valor preditivo
**order_item_id** - identificador único, sem valor preditivo

**Motivo: IDs servem apenas para relacionar tabelas, mas não contêm informação sobre comportamento, preço, frete ou satisfação.**

**Alterações do dataset:**
**Início: 22 colunas originais**
**Após one-hot: 42 colunas (+20 categóricas)**
**Após seleção: 37 colunas (-5 IDs)**

## 11. Criação de novos atributos (feature engineering)

A criação dessas features, é criar novas variáveis para a visualização de padrões importantes, e ajudando a responder às questões norteadoras. Nisso criamos 5 features novas:

**FEATURE 1: delivery_delay_days**
Descrição: Dias de atraso na entrega
Fórmula: data_entrega_real - data_estimada
Tipo: Numérica contínua Estatísticas:
Pedidos atrasados: 7265 (6,6%)

Atraso médio: 10,5 dias

Atraso máximo: 188 dias

**Resultado: Identificar problemas e a correlação entre satisfação com base no atraso.**

**FEATURE 2: freight_ratio**

Descrição: Relação entre frete e preço do produto Fórmula: valor_frete / preço

Tipo:Numérica contínua

Estatísticas:

Média: 0,32 (32% do preço)

Mediana: 0,23 (23% do preço)

Produtos com frete >50% do preço: 18.933 (16.8%)

**Resultado: Identificar produtos com fretes desproporcionais, como também, produtos baratos com fretes caros, resultando em insatisfação.**

**FEATURE 3: product_volume_cm3**

Descrição: Volume tridimensional do produto

Fórmula: comprimento × altura × largura Tipo:

Numérica contínua Estatísticas:

Média: 15.243 cm³

Mediana: 6.480 cm³

**Resultado: O frete vai depender do peso e do volume, e a identificação de produtos volumosos, porém leves. Ou ao contrário, produtos leves mas pesados.**

**FEATURE 5: order_sent**

Descrição: Pedido foi enviado à transportadora?

Fórmula: True se order_delivered_carrier_date não é nulo Tipo: Booleana (True/False)

Estatísticas:

Enviados: 111.430 (98,9%)

Não enviados: 1.194 (1,1%)

**Resultado: Produtos não enviados pode constar com cancelamento ou outros problemas.**

**FEATURE 6: delivered**

Descrição: Pedido foi entregue ao cliente?

Fórmula: True se order_delivered_customer_date não é nulo

Tipo: Booleana (True/False)

Estatísticas:

Entregues: 110.196 (97,8%)

Não entregues: 2.454 (2,2%)

**Resultado: Base para análises de satisfação.**

## 12. Pipeline completo de pré-processamento

Foi realizado todo o pipeline no desenvolvimento, seguindo sua estrutura recomendada. A estrutura ficou ficaram assim:

**Etapa 1:** Carregamento e Integração

3 datasets unidos via merge

**Etapa 2:** Exploração (EDA)

Análise de estatísticas, correlações e distribuições

**Etapa 3:** Limpeza de Dados

Remoção de duplicatas, inconsistências, tratamento de nulos

**Etapa 4:** Conversão de Tipos

Datas para datetime, IDs para inteiros

**Etapa 5:** Tratamento Categórico

Agrupamento de categorias raras + One-Hot Encoding

**Etapa 6:** Normalização
StandardScaler e MinMaxScaler aplicados

**Etapa 7:** Seleção de Atributos
Remoção de IDs irrelevantes

**Etapa 8:** Feature Engineering
Criação de 6 novas variáveis

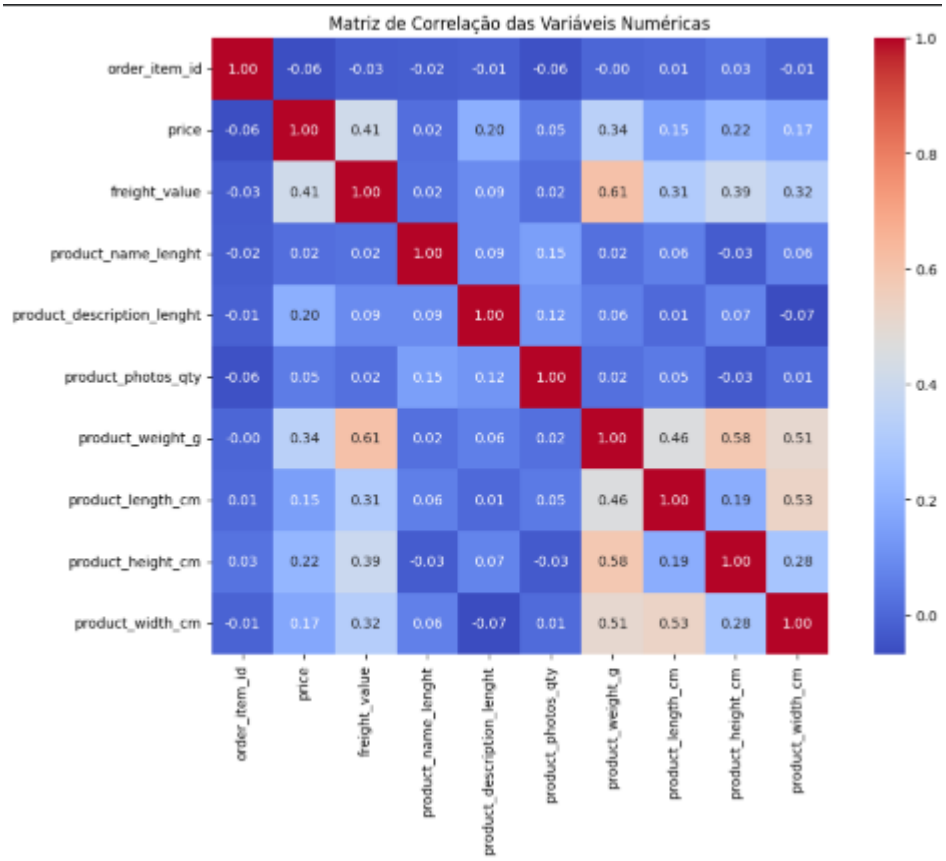
**Etapa 9:** Exportação
Datasets finais prontos para análise

**Conclusão: Essa estrutura descrita pode ser replicada em outras análises.**

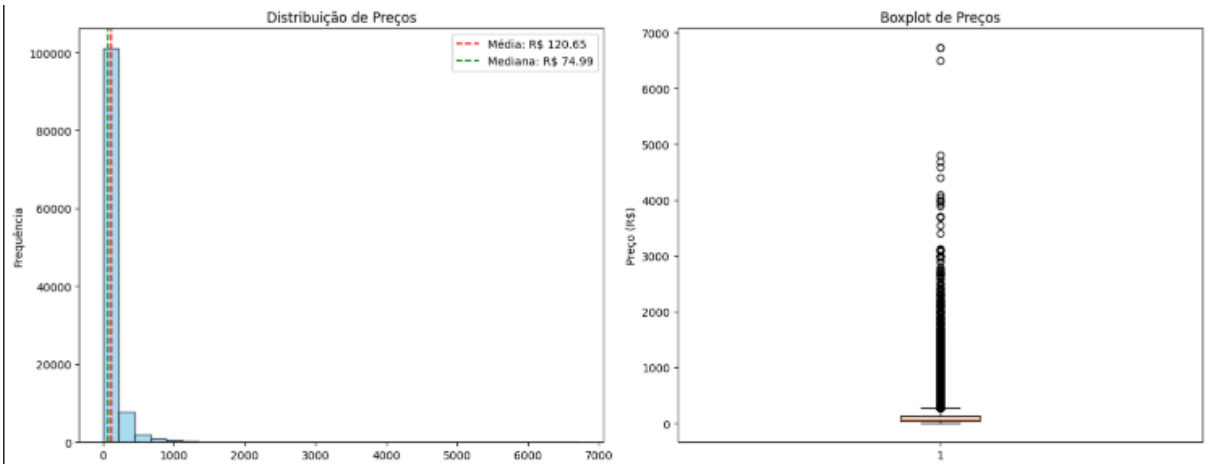
## 13. Visualizações e gráficos explicativos

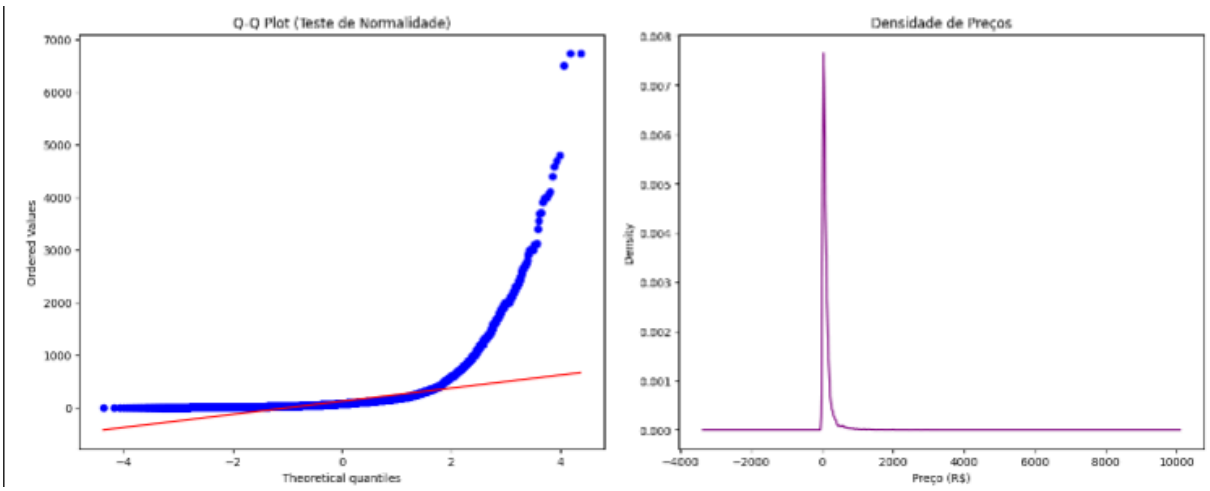
Todos os graficos utilizados vão está abaixo:

- Matriz de Correlação (Heatmap) -> Identificou correlação forte entre peso e frete (0.61), e revelou independência entre preço e dimensões físicas

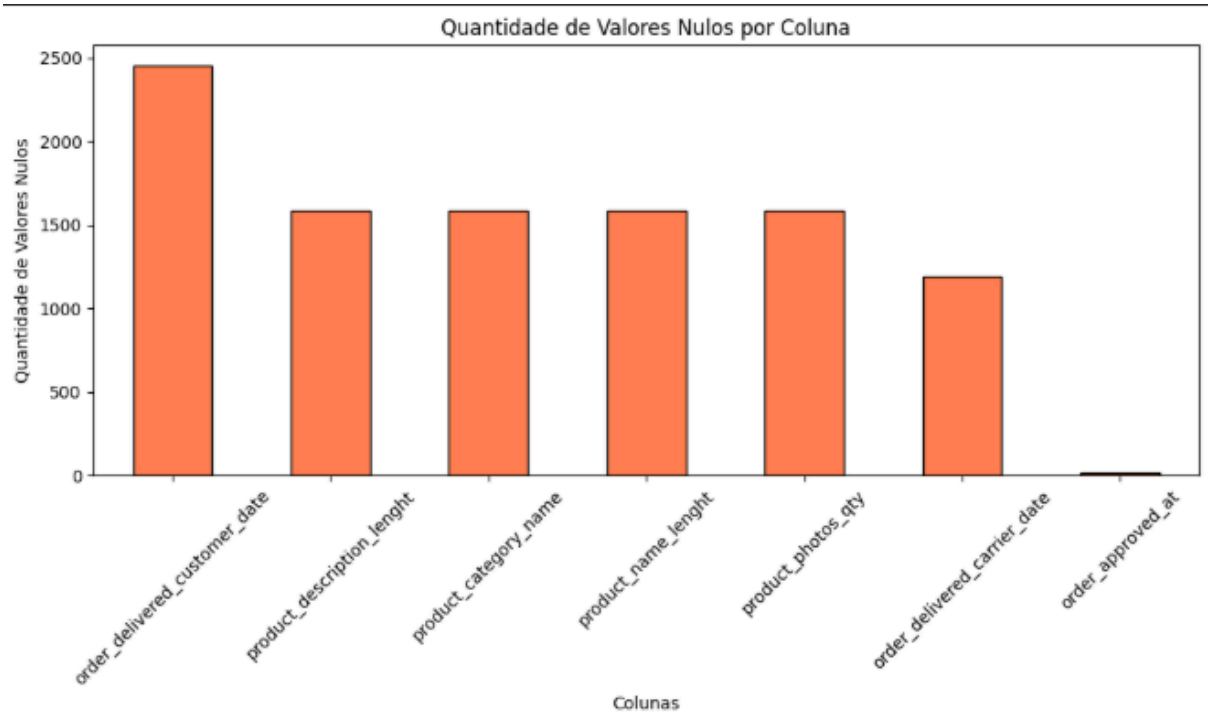



- Distribuição de Preços (4 gráficos)

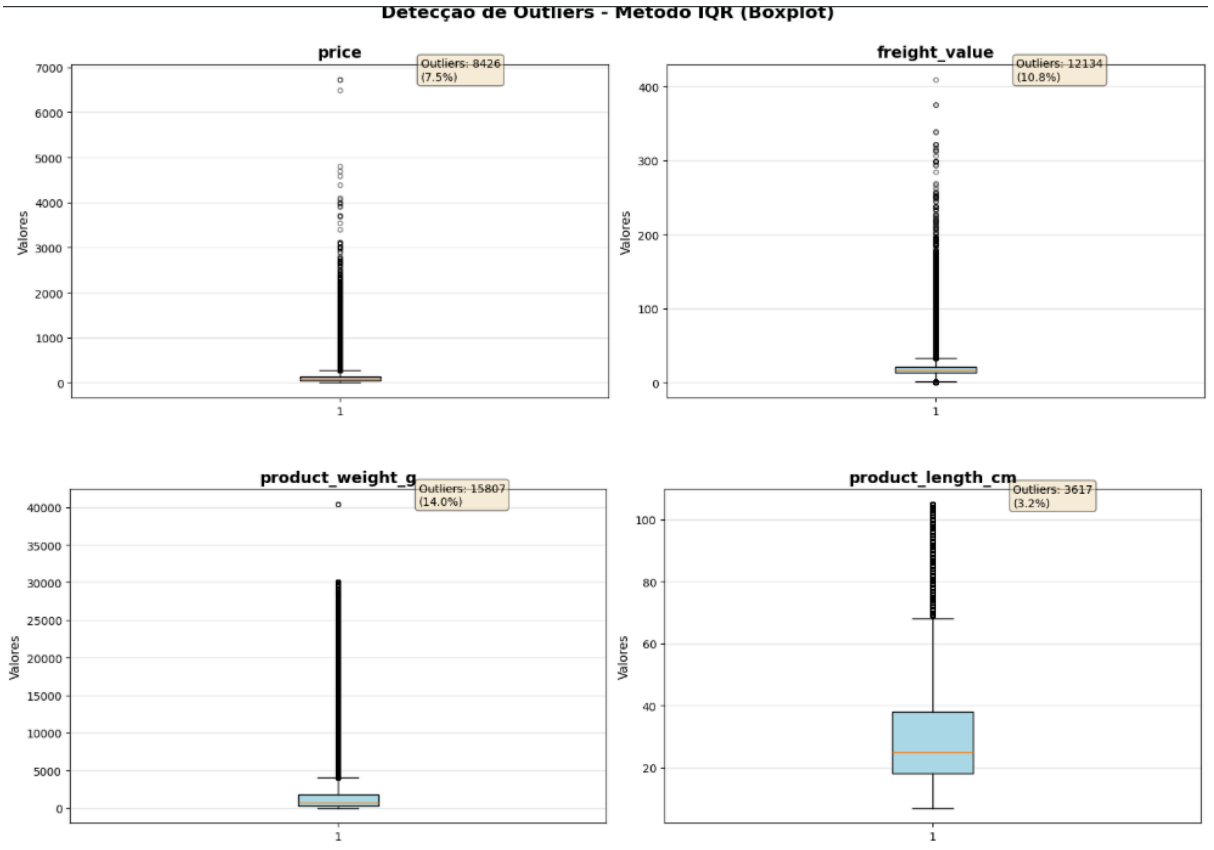




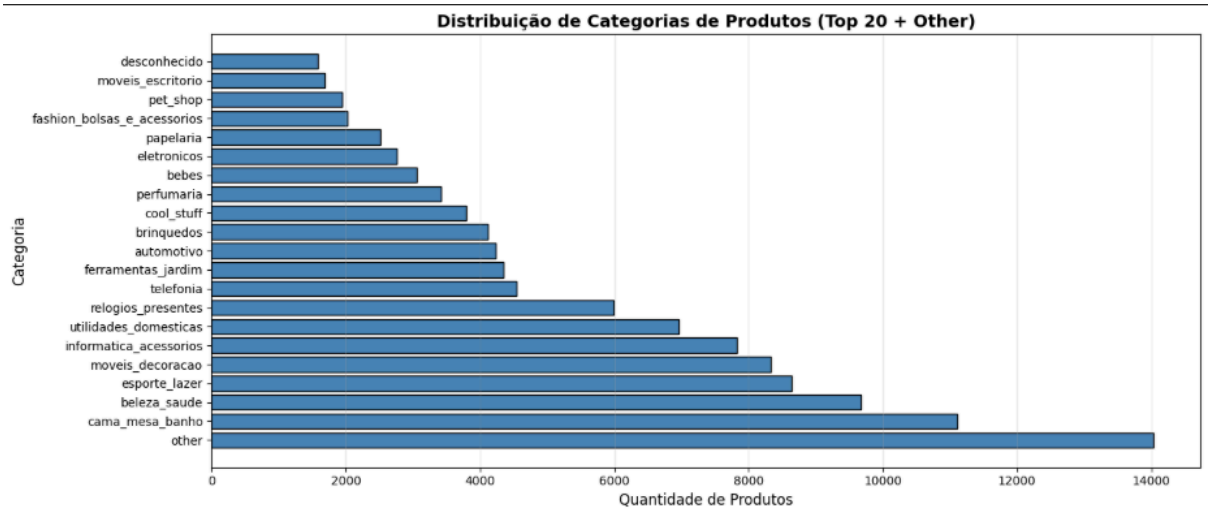

- Valores Nulos (Gráfico de Barras)



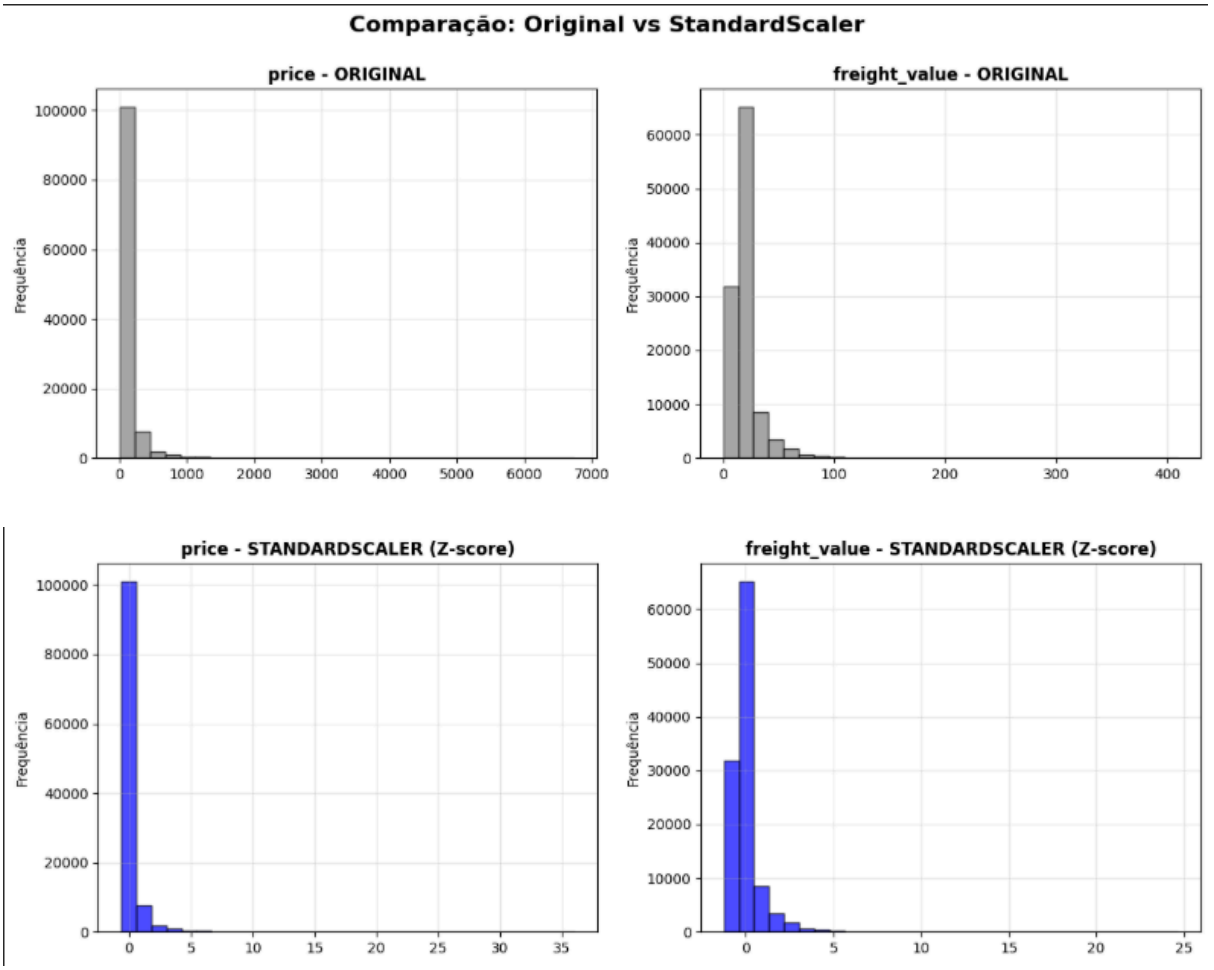

- Detecção de Outliers (Boxplots)



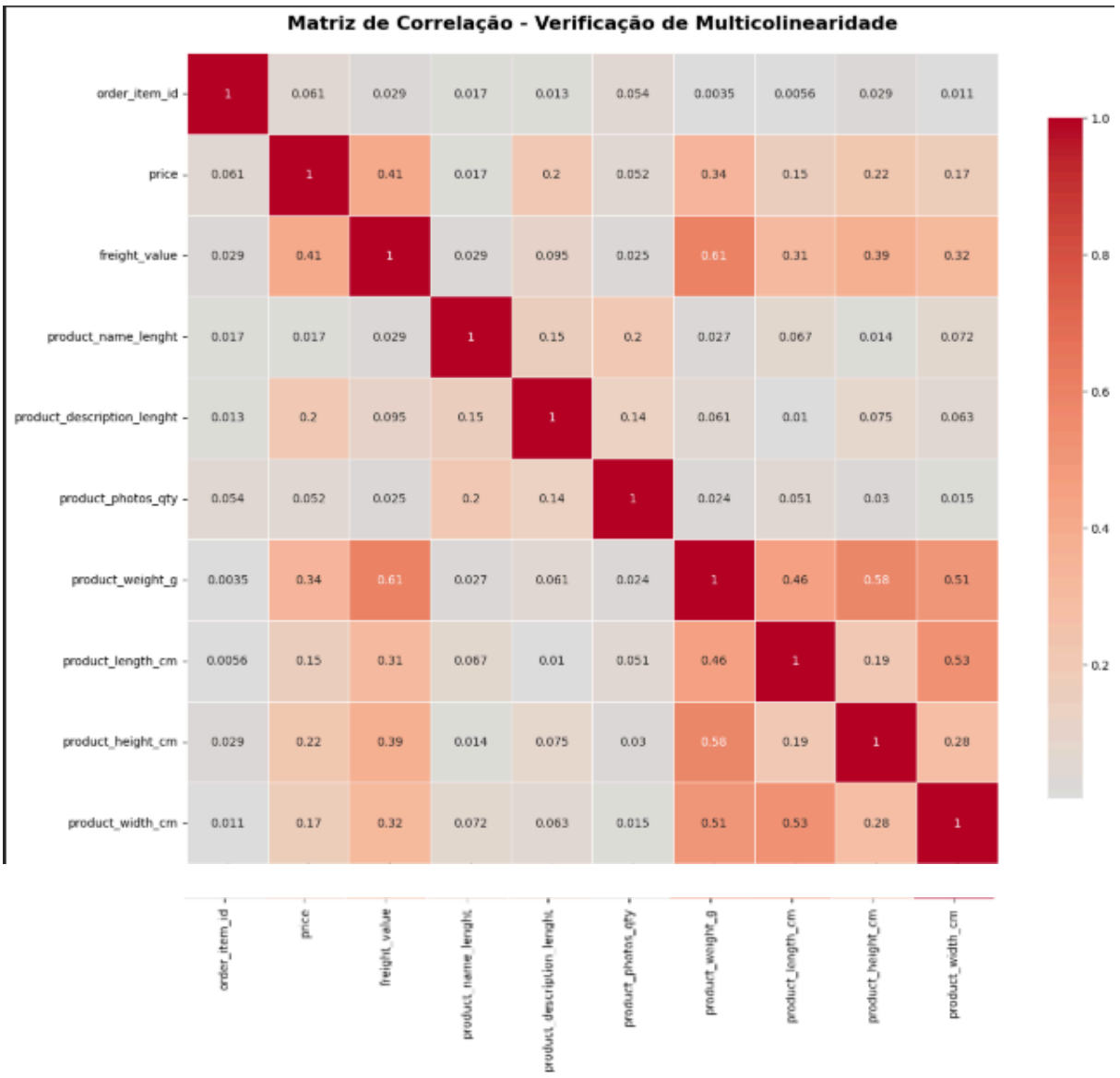

- Distribuição de Categorias




- Comparação de Normalizações




- Feature Engineering




# 14. Insights finais

A análise dos dados nos permitiu identificar padrões importantes relacionados à logística, precificação e qualidade da base de dados. Foram investigadas correlações, atrasos de entrega, categorias problemáticas, presença de outliers e relações entre variáveis numéricas.

## 14.1 Perguntas Norteadoras

**Características relacionadas ao atraso de entrega**

Os resultados mostraram que o atraso está mais ligado às características físicas dos produtos do que ao preço ou ao valor do frete. Produtos mais pesados e volumosos apresentam maior probabilidade de atraso, pois exigem transportes mais complexos. Categorias como móveis, eletrodomésticos e utilidades domésticas grandes concentram a maior parte dos atrasos observados. Já itens pequenos e leves, como beleza, informática e telefonia, raramente sofrem atraso.

**Categorias com maior incidência de problemas**

As categorias que mais geram dificuldades logísticas são aquelas formadas por produtos grandes, pesados ou frágeis. Móveis e eletrodomésticos, por exemplo, apresentam atrasos acima da média e fretes elevados, o que se explica pela dificuldade natural de transporte e pela necessidade de cuidados adicionais. Em diferença, categorias com produtos leves e de fácil armazenagem apresentam desempenho muito melhor, tanto em prazos quanto em custo logístico.

**Outliers presentes no dataset**

A detecção de outliers mostrou que há muitos valores altos de preço, frete, peso e dimensões. No entanto, esses valores representam produtos reais, como notebooks, geladeiras etc… não erros de digitação. Por isso, a decisão tomada foi manter todos os outliers para preservar a integridade do conjunto de dados e evitar distorções analíticas.

**Variáveis com maior correlação**

As correlações identificadas reforçam que o frete está fortemente associado ao peso do produto e, em menor intensidade, ao volume e às dimensões. Isso mostra que características físicas são os principais determinantes do custo logístico. Já o preço tem baixa correlação com medidas físicas, indicando que o valor agregado e fatores comerciais são mais relevantes do que tamanho ou peso. O atraso, apresenta correlação

fraca com variáveis numéricas, sugerindo que ele depende mais da categoria, região ou vendedor do que de características físicas.

**Conclusão Geral**

De forma ajustada, os resultados mostram que o peso e o volume dos produtos são os principais fatores de custo e de risco logístico. Produtos maiores tendem a atrasar mais e apresentam fretes proporcionalmente mais altos. Apesar disso, o dataset está bem estruturado, com baixa redundância e outliers naturais mantidos.

## 15. Conclusão do grupo

O projeto teve como objetivo analisar e preparar o conjunto de dados do e-commerce Olist para compreender padrões relacionados a atrasos, frete, categorias de produtos e características que influenciam a experiência do cliente. Todas as etapas previstas foram concluídas com sucesso, desde a exploração inicial até a criação de novas variáveis e exportação final sendo extraídas dois datasets, um de forma tratada sem o uso da normalização, e o normalizado.