Relatório do Projeto de Disciplina

Análise Exploratória de Dados

Eraldo N. Ferreira Pinto Júnior

10 abril, 2024

Sumário

Introdução

Para uma melhor organização deste projeto, foram criadas pastas com propósitos específicos para armazenamento dos arquivos.

A estrutura é dividida da seguinte forma:

- / - Pasta raiz do projeto:
  - Dataset - Pasta com os arquivos de extensão CSV.
    - Brent
    - Cambio
    - Combustivel
    - PPI
  - Image - Pasta com os arquivo de extensão PNG.
  - App

Preparando o Ambiente

Pacotes

Durante a análise foi verificado a necessidade de utilização de alguns pacotes. A lista de pacotes utilizados encontra-se abaixo:

```
# Lista com todos os pacotes
package <- c("tidyverse", "ggplot2", "summarytools", "data.table", "knitr", "dlookr", "ggpubr", "naniar
```

Com a lista de pacotes mapeados, a próxima etapa tem como foco verificar se todos os pacotes necessários para a análise se encontram instalados. Caso não estejam instalados, o processo de instalação será automático.

Para verificação dos pacotes instalados, a variável *is_installed* receberá o resultado *TRUE* para os pacotes instalados da lista *package* ou *FALSE* para os pacotes não instalados da lista *package*.

```
# Veritifica se o Pacote está instalado e o instala se for necessário.
is_installed <- package %in% rownames(installed.packages())

if(any(is_installed == FALSE)){
  install.packages(package[!is_installed])
}
```

Com os pacotes instalados, agora é necessário o carregamento deles, para que assim possamos iniciar as tratativas necessárias com os dados da análise.

```
# Carregando os Pacotes
invisible(lapply(package, library, character.only = TRUE))
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.2     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.0
## v ggplot2   3.4.3     v tibble    3.2.1
## v lubridate 1.9.2     v tidyr     1.3.0
## v purrr     1.0.1
```

```
## -- Conflicts ---------------------------------------- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
##
## Attaching package: 'summarytools'
##
##
## The following object is masked from 'package:tibble':
##
##     view
##
##
##
## Attaching package: 'data.table'
##
##
## The following objects are masked from 'package:lubridate':
##
##     hour, isoweek, mday, minute, month, quarter, second, wday, week,
##     yday, year
##
##
## The following objects are masked from 'package:dplyr':
##
##     between, first, last
##
##
## The following object is masked from 'package:purrr':
##
##     transpose
##
##
##
## Attaching package: 'dlookr'
##
##
## The following object is masked from 'package:tidyr':
##
##     extract
##
##
## The following object is masked from 'package:base':
##
##     transform
##
##
## corrplot 0.92 loaded
##
##
## Attaching package: 'mice'
##
##
## The following object is masked from 'package:stats':
```

```
##
##     filter
##
##
## The following objects are masked from 'package:base':
##
##     cbind, rbind
##
##
##
## Attaching package: 'simputation'
##
##
## The following object is masked from 'package:naniar':
##
##     impute_median
```

Pacotes carregados. Agora vamos remover as variáveis desnecessárias para as próximas etapas.

```
# Removendo variáveis desnecessárias
rm(list=ls())
```

Funções Criadas

Durante a elaboração da análise foi identificado a necessidade de realizar o tratamento das datas, uniformizando-as a partir das diferentes fontes de dados. Foram criadas funções para este tratamento.

A função *transform_date_one* transforma no formato AAAA-mm-dd todos os valores com o formato 12.10.2023 ou 12/10/2023.

A função *transform_date_two* transforma no formato AAAA-mm-dd todos os valores com o formato 12102023.

A função *transform_date_three* transforma no formato AAAA-mm-dd todos os valores com o formato Apr 2023.

Foi criada uma função específica que retornará a data formatada com padrão único, chamada *format_data*. Esta função será utilizada por todas as outras funções *transform_date_*.

```
transform_date_one <- function(data){
  partes_da_data <- strsplit(data, "[/.]")
  ano <- as.numeric(sapply(partes_da_data, `[`,3))
  mes <- as.numeric(sapply(partes_da_data, `[`,2))

  data_formatada <- format_data(ano, mes)

  return(data_formatada)
}

transform_date_two <- function(data){
  ano <- as.numeric(substr(data, nchar(data) - 3, nchar(data)))
  mes <- as.numeric(substr(data, nchar(data) - 5, nchar(data) - 4))

  data_formatada <- format_data(ano, mes)
```

```
  return(data_formatada)
}

transform_date_three <- function(data){
  ano <- as.numeric(substr(data, nchar(data) - 3, nchar(data)))
  mes_abreviado <- substr(data, nchar(data) - 7, nchar(data) - 5)
  mes <- as.integer(match(mes_abreviado, month.abb))

  data_formatada <- format_data(ano, mes)

  return(data_formatada)
}

format_data <- function(ano_data, mes_data){
  data_formatada <- as.Date(sprintf("%04d-%02d-01", ano_data, mes_data))
  return(data_formatada)
}
```

Dando continuidade a necessidade de funções específicas, alguns dados foram disponibilizados pelas suas fontes em arquivos distintos. Para uma carga de dados mais eficiente foram criadas funções que possibilitarão uma agilidade neste processo.

```
extractor_csv2 = function(dados){
  readr::read_csv2(dados, locale = locale(encoding = 'UTF-8'), show_col_types = FALSE)
}

extractor_csv = function(dados){
  read.csv(dados, header = FALSE, sep = ";", dec = ",")
}
```

Abaixo é descrita as funções geradoras de binwidths.

```
fd <- function(x) {
  n <-length(x)
  return((2*IQR(x))/n^(1/3))
}


sr <- function(x) {
  n <-length(x)
  return((3.49*sd(x))/n^(1/3))
}
```

Dos Dados

Contexto

O valor de venda dos derivados de petróleo aos consumidores brasileiros é sempre um assunto polêmico. Há muitas variáveis que influenciam na flutuação do valor de venda. Para o consumidor final o que importa é o quanto estas flutuações impactam no orçamento mensal da família.

Os meios de comunicação frequentemente noticiam o aumento ou a redução dos derivados do petróleo diante da flutuação de algumas variáveis, como por exemplo o brent e o câmbio.

A flutuação destas variáveis e de outras são oriundas de acontecimentos mundiais. Os grandes canais de comunicação noticiam periodicamente estes eventos.

- O "Petróleo sobe mais de 3% em meio a tensões no Oriente Médio" (CNN-Brasil, 2024).
- A "Guerra e petróleo: veja reações mais drásticas da commodity a grandes conflitos" (CNN-Brasil, 2023).
- A "Gerra no Oriente Médio pode aumentar preço do diesel, diz Petrobras" (AgênciaBrasil-EBC, 2023).

Além das variáveis, uma sigla foi introduzida na vida dos brasileiros diante a mudança da política de preço praticada pela petrolífera brasileira (Petrobras). Esta sigla é conhecida como o Preço de Paridade de Importação - PPI.

Um breve histórico da adoção do PPI pela Petrobras e seus desdobramentos políticos pode ser lido na matéria "Gasolina cara, lucro recorde: como foi o PPI, antiga política da Petrobras" (Economia UOL, 2023)

O fim da adoção do PPI pela Petrobras em 16 de maio de 2023 repercutiu nacionalmente.

- A "Petrobras anuncia fim da paridade internacional de preços do petróleo" (CNN Brasil, 2023). "Para Ineep, fim do PPI na Petrobras trouxe maior estabilidade de preço dos combustíveis" (InfoMoney, 2024).
- A "Gasolina da Petrobras está 17% mais barata que preço internacional" (Metrópoles, 2024).

Em resumo, o experimento visa compreender o preço de venda de combustíveis entre o período de adoção da política de preço da Petrobras e o fim da adoção desta política.

O espaço amostral para o experimento envolve os combustíveis Diesel, Diesel S10 e Gasolina pesquisados através de pesquisa nos postos de combustíveis selecionados pela ANP.

O período de coleta dos dados ocorrerá entre janeiro de 2020 até março de 2024.

Fonte dos dados

Um fator crucial para qualquer análise é a busca de fontes de dados abertos confiáveis. Portanto, buscou-se através de sites oficiais de governo e instituições renomadas os dados necessários para a respectiva análise.

O primeiro dado a ser obtido foi a "Série Histórica de Preços de Combustíveis e de GLP" (Dados Abertos-ANP, 2024). Esta fonte de dados possui os dados das pesquisas realizadas até março de 2024. Os dados utilizandos para esta análise foram os dados oriundos das pesquisas realizadas até março de 2024.

Ainda no site da ANP, foi utilizado os "Preços de paridade de importação" (PPI-ANP, 2024).

O Brent foi obtido através da *U.S. Energy information Administration - EIA*. Os dados obtidos fazem parte da visão histórica dos dados em *PETROLEUM & OTHER LIQUIDS* (EIA, 2024).

A série histórica da taxa cambial foi obtida através do site do Banco Central do Brasil, em sua área "Cotações e boletins" (BCB, 2024).

Realizando a carga dos dados

Neste momento será realizado o carregamento dos dados obtidos através das fontes de dados supracitadas. Os dados foram armazenados nas subpastas da pasta Dataset.

Para realização desta etapa, duas estratégias foram adotadas.

A primeira estratégia, de forma recursiva, se utilizou o *list.files* para localizar todos os arquivos a partir de um *pattern* (padrão) no nome dos arquivos. Uma variável com a lista contendo o nome do arquivo e o caminho foi criada para armazená-las. Posteriormente foi utilizada a função map_dfr para aplicar cada elemento (arquivos) na função criada para extração dos dados. Esta estratégia envolve a carga de dados histórica dos combustíveis, glp e taxa de câmbio que são constituídas de vários arquivos.

```
arquivos <- list.files(pattern = "^ca-", recursive = TRUE)
origin_combustiveis_agg <- map_dfr(arquivos, extractor_csv2)
```

```
## i Using "','" as decimal and "'.'" as grouping mark. Use `read_delim()` for more control.
## i Using "','" as decimal and "'.'" as grouping mark. Use `read_delim()` for more control.
## i Using "','" as decimal and "'.'" as grouping mark. Use `read_delim()` for more control.
## i Using "','" as decimal and "'.'" as grouping mark. Use `read_delim()` for more control.
## i Using "','" as decimal and "'.'" as grouping mark. Use `read_delim()` for more control.
## i Using "','" as decimal and "'.'" as grouping mark. Use `read_delim()` for more control.
## i Using "','" as decimal and "'.'" as grouping mark. Use `read_delim()` for more control.
## i Using "','" as decimal and "'.'" as grouping mark. Use `read_delim()` for more control.
## i Using "','" as decimal and "'.'" as grouping mark. Use `read_delim()` for more control.
## i Using "','" as decimal and "'.'" as grouping mark. Use `read_delim()` for more control.
## i Using "','" as decimal and "'.'" as grouping mark. Use `read_delim()` for more control.
## i Using "','" as decimal and "'.'" as grouping mark. Use `read_delim()` for more control.
## i Using "','" as decimal and "'.'" as grouping mark. Use `read_delim()` for more control.
## i Using "','" as decimal and "'.'" as grouping mark. Use `read_delim()` for more control.
## i Using "','" as decimal and "'.'" as grouping mark. Use `read_delim()` for more control.
```

```
message("Dados carregados dos arquivos CSV.")
```

```
## Dados carregados dos arquivos CSV.
```

```
rm(arquivos)
```

Iniciando a carga dos dados do câmbio.

```
arquivos <- list.files(pattern = "^CotacoesMoedasPeriodo", recursive = TRUE)
origin_cambio <- map_dfr(arquivos, extractor_csv)

rm(arquivos)
```

A segunda estratégia, foi mais simples, pois se refere a extração de um único arquivo com todos os dados históricos do Brent.

```
origin_brent <- read.table("Dataset/Brent/Europe_Brent_Spot_Price_FOB.csv", sep=",",
                    header = TRUE)
```

A segunda estratégia também foi adotada para a extração de um único arquivo com todos os dados históricos do PPI.

Atenção: Os dados do PPI foram disponibilizados em vários sheets em uma única planilha do Excel, com extensão XLSX. Foi necessário tratar os dados diretamente no Excel, possibilitando assim um carregamento mais célere.

```
origin_ppi_agg <- read.table("Dataset/PPI/ppi.csv", sep=";", dec = ",",
                  header = TRUE)
```

Visão Inicial dos Dados

Após a importação das bases de dados, vamos apresentar as primeiras observações para conhecimento das variáveis.

```
head(origin_combustiveis_agg)
```

```
## # A tibble: 6 x 16
##   `Regiao - Sigla` `Estado - Sigla` Municipio Revenda          `CNPJ da Revenda`
##   <chr>            <chr>            <chr>     <chr>            <chr>
## 1 SE               SP               GUARULHOS AUTO POSTO SAKA~ 49.051.667/0001-~
## 2 SE               SP               GUARULHOS AUTO POSTO SAKA~ 49.051.667/0001-~
## 3 SE               SP               GUARULHOS AUTO POSTO SAKA~ 49.051.667/0001-~
## 4 SE               SP               GUARULHOS AUTO POSTO SAKA~ 49.051.667/0001-~
## 5 NE               BA               SALVADOR  PETROBRAS DISTR~ 34.274.233/0015-~
## 6 NE               BA               SALVADOR  PETROBRAS DISTR~ 34.274.233/0015-~
## # i 11 more variables: `Nome da Rua` <chr>, `Numero Rua` <chr>,
## #   Complemento <chr>, Bairro <chr>, Cep <chr>, Produto <chr>,
## #   `Data da Coleta` <chr>, `Valor de Venda` <dbl>, `Valor de Compra` <dbl>,
## #   `Unidade de Medida` <chr>, Bandeira <chr>
```

```
head(origin_cambio)
```

```
##        V1  V2 V3  V4     V5     V6 V7 V8
## 1 2012020 220  A USD 4.0207 4.0213  1  1
## 2 3012020 220  A USD 4.0516 4.0522  1  1
## 3 6012020 220  A USD 4.0548 4.0554  1  1
## 4 7012020 220  A USD 4.0835 4.0841  1  1
## 5 8012020 220  A USD 4.0666 4.0672  1  1
## 6 9012020 220  A USD 4.0738 4.0744  1  1
```

```
head(origin_brent)
```

```
##      Month Europe.Brent.Spot.Price.FOB.Dollars.per.Barrel
## 1 Mar 2024                                          85.41
## 2 Feb 2024                                          83.48
## 3 Jan 2024                                          80.12
## 4 Dec 2023                                          77.63
## 5 Nov 2023                                          82.94
## 6 Oct 2023                                          90.60
```

```r
head(origin_ppi_agg)
```

```
##         Data    PPI  Produto Unidade.de.Medida
## 1 01/01/2020 1.9595 Gasolina          R$/litro
## 2 01/01/2020 1.9262 Gasolina          R$/litro
## 3 01/01/2020 1.9198 Gasolina          R$/litro
## 4 01/01/2020 1.8878 Gasolina          R$/litro
## 5 01/01/2020 1.7951 Gasolina          R$/litro
## 6 01/01/2020 2.4272   Diesel          R$/litro
```

```r
#knitr::kable(origin_combustiveis_agg,
#             format="latex",
#             caption = "Base de dados da pesquisa de combustíveis.",
#             align = "c",
#             booktabs = TRUE,
#             longtable = TRUE,
#             linesep = ""
#)
```

Analisando os tipos das variáveis nas bases

Após a carga dos dados, foi necessário identificar o tipo de cada variável nas bases. Utilizar-se-á a função diagnose do pacote dlookr que reportará o tipo em todas as bases.

Base de Dados Combustíveis

```r
origin_combustiveis_agg %>% dlookr::diagnose()
```

```
## # A tibble: 16 x 6
##    variables        types missing_count missing_percent unique_count unique_rate
##    <chr>            <chr>         <int>           <dbl>        <int>       <dbl>
##  1 Regiao - Sigla   char~             0           0                5 0.00000140
##  2 Estado - Sigla   char~             0           0               27 0.00000755
##  3 Municipio        char~             0           0              469 0.000131
##  4 Revenda          char~             0           0            18569 0.00519
##  5 CNPJ da Revenda  char~             0           0            19961 0.00558
##  6 Nome da Rua      char~             0           0            12042 0.00337
##  7 Numero Rua       char~          1549           0.0433        5151 0.00144
##  8 Complemento      char~       2774672          77.6           3580 0.00100
##  9 Bairro           char~          8769           0.245         7691 0.00215
## 10 Cep              char~             0           0            13929 0.00389
## 11 Produto          char~             0           0                6 0.00000168
## 12 Data da Coleta   char~             0           0             1049 0.000293
## 13 Valor de Venda   nume~             0           0             4976 0.00139
## 14 Valor de Compra  nume~       3420835          95.6          22767 0.00637
## 15 Unidade de Medi~ char~             0           0                3 0.000000839
## 16 Bandeira         char~             0           0               80 0.0000224
```

```r
origin_combustiveis_agg %>% dplyr::select(Produto) %>% base::unique()
```

```
## # A tibble: 6 x 1
##   Produto
##   <chr>
## 1 GASOLINA
## 2 ETANOL
## 3 DIESEL S10
## 4 GNV
## 5 DIESEL
## 6 GASOLINA ADITIVADA
```

```r
origin_combustiveis_agg %>%
  dplyr::group_by(Produto) %>%
  dplyr::mutate(n_linhas = 1) %>%
  dplyr::summarise(n_linhas = sum(n_linhas))
```

```
## # A tibble: 6 x 2
##   Produto            n_linhas
##   <chr>                 <dbl>
## 1 DIESEL               433411
## 2 DIESEL S10           740397
## 3 ETANOL               824394
## 4 GASOLINA             940511
## 5 GASOLINA ADITIVADA   569366
## 6 GNV                   68768
```

As variáveis deste contjunto de dados pode ser classificadas conforme a tabela abaixo.

\begin{center} Table 1: Classificação das variáveis da base de dados combustíveis. \end{center

Variável	Classificação
Regiao - Sigla	Qualitativa nominal
Estado - Sigla	Qualitativa nominal
Municipio	Qualitativa nominal
Revenda	Qualitativa nominal
CNPJ da Revenda	Qualitativa nominal
Nome da Rua	Qualitativa nominal
Numero Rua	Qualitativa nominal
Complemento	Qualitativa nominal
Bairro	Qualitativa nominal
Cep	Qualitativa nominal
Produto	Qualitativa nominal
Data da Coleta	Qualitativa nominal
Valor de Venda	Quantitativa contínua
Valor de Compra	Quantitativa contínua
Unidade de Medida	Qualitativa nominal
Bandeira	Qualitativa nominal

É possível identificar a existência na base de 16 variáveis. Um total de 14 variáveis são qualitativas, sendo estas nominais. Sobre as variáveis quantitativas, temos Valor de Venda e Valor de Compra, ambas variáveis contínuas.

Observa-se que as variáveis Numero da rua, Complemento, Bairro e Valor de Venda possuem missing.

No Brasil, a pesquisa foi realizada:

- Em 5 regiões.
- Em 27 estados.
- Em 469 municípios.
- Em 19.961 revendas por CNPJ.
- Considerando 6 produtos comercializados.
  - GASOLINA
  - ETANOL
  - DIESEL S10
  - GNV
  - DIESEL
  - GASOLINA ADITIVADA
- E o missing do Valor de Compra foi de 3.420.835, o qual representa 95,64% da base de dados.
- Em 80 bandeiras diferentes.
- As variáveis de interesse nesta base de dados são:
  - Regiao - Sigla
  - Estado - Sigla
  - Municipio
  - CNPJ da Revenda
  - Produto
  - Data da Coleta
  - Valor de Venda
  - Valor de Compra
  - Unidade de Medida
  - Bandeira

Base de Dados Câmbio

Dando continuidade, será realizada a análise da próxima base de dados, taxa de câmbio.

```
str(origin_cambio)
```

```
## 'data.frame':    1063 obs. of  8 variables:
##  $ V1: int  2012020 3012020 6012020 7012020 8012020 9012020 10012020 13012020 14012020 15012020 ...
##  $ V2: int  220 220 220 220 220 220 220 220 220 220 ...
##  $ V3: chr  "A" "A" "A" "A" ...
##  $ V4: chr  "USD" "USD" "USD" "USD" ...
##  $ V5: num  4.02 4.05 4.05 4.08 4.07 ...
##  $ V6: num  4.02 4.05 4.06 4.08 4.07 ...
##  $ V7: num  1 1 1 1 1 1 1 1 1 1 ...
##  $ V8: num  1 1 1 1 1 1 1 1 1 1 ...
```

```
origin_cambio %>% dlookr::diagnose()
```

```
## # A tibble: 8 x 6
##   variables types     missing_count missing_percent unique_count unique_rate
##   <chr>     <chr>             <int>           <dbl>        <int>       <dbl>
## 1 V1        integer               0               0         1063    1
## 2 V2        integer               0               0            1    0.000941
## 3 V3        character             0               0            1    0.000941
## 4 V4        character             0               0            1    0.000941
## 5 V5        numeric               0               0         1011    0.951
## 6 V6        numeric               0               0         1012    0.952
## 7 V7        numeric               0               0            1    0.000941
## 8 V8        numeric               0               0            1    0.000941
```

Ao inspecionar a base de dados, é possível identificar na base a existência de 8 variáveis. Neste primeiro momento não foi possível identificar claramente o propósito das variáveis. Para isso, foi utilizado a função head para leitura dos primeiros dados da base.

É possível observar que a variável V1 é do tipo integer, contudo, ela expressa a data de cotação do câmbio, portanto, é uma variável qualitativa nominal. As variáveis V5 e V6 expressão, respectivamente, cotação de compra e venda na moeda real. Portanto, são quantitativas e ambas são contínuas. A variável V2, V3 e V4, através da informação contida na descrição da base, são: V2 - Código da Moeda, V3 - Tipo da Moeda e V4 - Símbolo da Moeda. Ambas são qualitativas nominais. As variáveis V7 e V8 não possuem descrição na base.

Para o objetivo da análise, as variáveis selecionadas serão V1 e V5.

Base de Dados Brent

A análise de variável desta base de dados foi mais simples. A base é composta por duas variáveis. A data é uma variável qualitativa nominal e a variável Preco USD/Barril é quantitativa contínua.

```
str(origin_brent)
```

```
## 'data.frame':    51 obs. of  2 variables:
##  $ Month                                        : chr  "Mar 2024" "Feb 2024" "Jan 2024" "Dec 2023"
##  $ Europe.Brent.Spot.Price.FOB.Dollars.per.Barrel: num  85.4 83.5 80.1 77.6 82.9 ...
```

```
origin_brent %>% dlookr::diagnose()
```

```
## # A tibble: 2 x 6
##   variables         types missing_count missing_percent unique_count unique_rate
##   <chr>             <chr>         <int>           <dbl>        <int>       <dbl>
## 1 Month             char~             0               0           51           1
## 2 Europe.Brent.Spo~ nume~             0               0           51           1
```

Base de Dados PPI

Considerando que esta base de dados passou por um tratamento prévio e externo, as variáveis existentes são os valores do PPI por portos e pontos de entrega. A data é uma variável qualitativa nominal e as variáveis que representam os portos e os pontos de entrega são quantitativas contínuas.

```r
str(origin_ppi_agg)
```

```
## 'data.frame':    666 obs. of  4 variables:
##  $ Data             : chr  "01/01/2020" "01/01/2020" "01/01/2020" "01/01/2020" ...
##  $ PPI              : num  1.96 1.93 1.92 1.89 1.8 ...
##  $ Produto          : chr  "Gasolina" "Gasolina" "Gasolina" "Gasolina" ...
##  $ Unidade.de.Medida: chr  "R$/litro" "R$/litro" "R$/litro" "R$/litro" ...
```

```r
origin_ppi_agg %>% dlookr::diagnose()
```

```
## # A tibble: 4 x 6
##   variables         types missing_count missing_percent unique_count unique_rate
##   <chr>             <chr>         <int>           <dbl>        <int>       <dbl>
## 1 Data              char~             0               0           51     0.0766
## 2 PPI               nume~             0               0          650     0.976
## 3 Produto           char~             0               0            3     0.00450
## 4 Unidade.de.Medida char~             0               0            2     0.00300
```

Para a respectiva análise utilizar-se-á as variáveis Data e PPI por produto.

Normatizando o nome e escolhendo as variáveis

Normatizando as variáveis da base de combustíveis.

```r
colnames(origin_combustiveis_agg) <- c("Regiao",
                                       "UF",
                                       "Municipio",
                                       "Revenda",
                                       "CNPJ_Revenda",
                                       "Rua",
                                       "Numero_rua",
                                       "Complemento",
                                       "Bairro",
                                       "CEP",
                                       "Produto",
                                       "Data",
                                       "Valor_de_Venda",
                                       "Valor_de_Compra",
                                       "Unidade_de_Medida",
                                       "Bandeira"
)
```

Selecionando um conjunto de variáveis da base combustíveis e os produtos.

```r
sample_combustiveis_agg <- origin_combustiveis_agg[,c(1,2,3,5,11,12,16,13,14)]

sample_combustiveis_agg <- sample_combustiveis_agg %>%
   dplyr::filter(Produto == "GASOLINA" | Produto == "DIESEL S10" | Produto == "DIESEL")
```

Normatizando a base de dados brent.

```
colnames(origin_brent) <- c("Data", "Brent_USD_Barril")
```

Selecionando um conjunto de variáveis e normatizando a base de dados cambio.

```
sample_cambio <- origin_cambio[, c(1, 5)]
colnames(sample_cambio) <- c("Data", "Taxa_Cambio")
```

Tratamento e padronizando as Datas

As quatro bases de dados dispostas nesta análise (combustiveis, cambio, brent e ppi) possuem datas com formatos e características diferentes.

A uniformização das datas possibilitará mesclar estes dados em uma única base de dados.

Para esta uniformização, as funções de transformação das datas serão chamadas, passando como parâmetro o campo data das bases de dados.

```
origin_combustiveis_agg$Data <- transform_date_one(origin_combustiveis_agg$Data)
sample_combustiveis_agg$Data <- transform_date_one(sample_combustiveis_agg$Data)
origin_brent$Data <- transform_date_three(origin_brent$Data)
sample_cambio$Data <- transform_date_two(sample_cambio$Data)
origin_ppi_agg$Data <- transform_date_one(origin_ppi_agg$Data)

rm(origin_cambio)
```

Gerando dados calculados

Para a realização da próxima etapa da análise, a qual envolve o *merge* entre as bases de dados até aqui apresentada, será necessário realizar filtros e cálculos.

A variável taxa de câmbio da base de dados cambio é composta por valores cotados diariamente.

Para uma uniformização mensal dos dados, foi realizado cálculos estatísticos do câmbio praticado mensalmente a partir do valor diário.

```
calc_cambio <- sample_cambio %>%
   dplyr::group_by(Data) %>%
   dplyr::summarise(Cambio = mean(Taxa_Cambio),.groups = 'drop')
#   dplyr::summarise(Cambio = mean(Taxa_Cambio),
#                    Max_Cambio = max(Taxa_Cambio),
#                    Min_Cambio = min(Taxa_Cambio),
#                    SD_Cambio = sd(Taxa_Cambio),.groups = 'drop')
```

Foi gerada uma nova base de dados, tendo como descrição o termo calc de calculada. Utilizou-se a função group_by() do pacote dplyr para agrupar os dados pela data e posteriormente a função summarise() do pacote dplyr para os cálculos.

A variável PPI da base de dados ppi é fruto da análise semanal. As datas semanais foram tratadas para uniformização mensal dos dados. Foram realizados cálculos estatísticos a partir dos valores semanais para o respectivo cálculo mensal.

```
calc_ppi <- origin_ppi_agg %>%
   dplyr::group_by(Data, Produto) %>%
   dplyr::summarise(PPI = mean(PPI), .groups = 'drop')
#   dplyr::summarise(PPI = mean(PPI),
#                    Max_PPI = max(PPI),
#                    Min_PPI = min(PPI),
#                    SD_PPI = sd(PPI),.groups = 'drop')
```

Foi gerada uma nova base de dados, tendo como descrição o termo calc de calculada. Utilizou-se a função group_by() do pacote dplyr para agrupar os dados pela data e posteriormente a função summarise() do pacote dplyr para os cálculos.

Merge das bases de dados

Após a realização do filtro e os devidos cálculos, realizar-se-á nesta etapa o *merge* das bases de dados.

```
sample_combustiveis_agg <- dplyr::bind_rows(base::merge(sample_combustiveis_agg %>%
                                                        dplyr::filter(Produto == "DIESEL" | Produto =
                                                    calc_ppi %>%
                                                        dplyr::filter(Produto == "Diesel") %>%
                                                        dplyr::select(Data, PPI),
                                                    by = "Data",
                                                    all = TRUE),
                                            base::merge(sample_combustiveis_agg %>%
                                                        dplyr::filter(Produto == "GASOLINA"),
                                                    calc_ppi %>%
                                                        dplyr::filter(Produto == "Gasolina") %>%
                                                        dplyr::select(Data, PPI),
                                                    by = "Data",
                                                    all = TRUE)
                                            )

sample_combustiveis_agg <- base::merge(sample_combustiveis_agg,
                                       origin_brent,
                                       by = "Data",
                                       all = TRUE)

sample_combustiveis_agg <- base::merge(sample_combustiveis_agg,
                                       calc_cambio,
                                       by = "Data",
                                       all = TRUE)
```

```
sample_combustiveis_agg <- sample_combustiveis_agg %>%
   dplyr::mutate(Brent_Real_Barril = Brent_USD_Barril * Cambio)
```

Análise de frequências de variáveis qualitativas

Na base de dados sample_combustiveis_agg, as variáveis Produto, Região, UF, Município e Revenda são variáveis qualitativas nominais na base. São variáveis interessantes para extração das frequências.

Para esta primeira análise de frequência, analisaremos a variável produto utilizando a função freq() do pacote summarytools.

```
origin_combustiveis_agg %>%
   dplyr::select(Produto) %>%
   summarytools::freq(., style = 'rmarkdown', order = "freq", plain.ascii = FALSE)
```

```
## ### Frequencies
## #### origin_combustiveis_agg$Produto
## **Type:** Character
##
## |                    |    Freq | % Valid | % Valid Cum. | % Total | % Total Cum. |
## |------------------------:|--------:|--------:|-------------:|--------:|-------------:|
## |            **GASOLINA** |  940511 |   26.29 |        26.29 |   26.29 |        26.29 |
## |              **ETANOL** |  824394 |   23.05 |        49.34 |   23.05 |        49.34 |
## |          **DIESEL S10** |  740397 |   20.70 |        70.04 |   20.70 |        70.04 |
## | **GASOLINA ADITIVADA** |  569366 |   15.92 |        85.96 |   15.92 |        85.96 |
## |              **DIESEL** |  433411 |   12.12 |        98.08 |   12.12 |        98.08 |
## |                 **GNV** |   68768 |    1.92 |       100.00 |    1.92 |       100.00 |
## |              **\<NA\>** |       0 |         |              |    0.00 |       100.00 |
## |               **Total** | 3576847 |  100.00 |       100.00 |  100.00 |       100.00 |
```

```
sample_combustiveis_agg %>%
   dplyr::select(Produto) %>%
   summarytools::freq(., style = 'rmarkdown', order = "freq", plain.ascii = FALSE)
```

```
## ### Frequencies
## #### sample_combustiveis_agg$Produto
## **Type:** Character
##
## |            |    Freq |    % Valid | % Valid Cum. |    % Total | % Total Cum. |
## |----------------:|--------:|-----------:|-------------:|-----------:|-------------:|
## |    **GASOLINA** |  940511 |  44.482928 |    44.482928 |  44.482886 |    44.482886 |
## | **DIESEL S10** |  740397 |  35.018226 |    79.501154 |  35.018193 |    79.501079 |
## |      **DIESEL** |  433411 |  20.498846 |   100.000000 |  20.498827 |    99.999905 |
## |      **\<NA\>** |       2 |            |              |   0.000095 |   100.000000 |
## |       **Total** | 2114321 | 100.000000 |   100.000000 | 100.000000 |   100.000000 |
```

Foi possivel observar que durante o **merge** dos dados foram gerados missing. Vamos agora entender do que trata-se esse missing gerado.

```r
sample_combustiveis_agg %>%
   dplyr::select(Data) %>%
   summarytools::freq(., style = 'rmarkdown', order = "freq", plain.ascii = FALSE)
```

```
## ### Frequencies
## #### sample_combustiveis_agg$Data
## **Type:** Date
##
## |           |    Freq |    % Valid | % Valid Cum. |    % Total | % Total Cum. |
## |---------------:|--------:|-----------:|-------------:|-----------:|-------------:|
## | **2020-06-01** |   61744 |   2.920276 |     2.920276 |   2.920276 |     2.920276 |
## | **2020-01-01** |   61516 |   2.909492 |     5.829768 |   2.909492 |     5.829768 |
## | **2020-04-01** |   61307 |   2.899607 |     8.729375 |   2.899607 |     8.729375 |
## | **2020-03-01** |   61292 |   2.898898 |    11.628272 |   2.898898 |    11.628272 |
## | **2020-07-01** |   60662 |   2.869101 |    14.497373 |   2.869101 |    14.497373 |
## | **2022-08-01** |   60479 |   2.860446 |    17.357818 |   2.860446 |    17.357818 |
## | **2022-06-01** |   55724 |   2.635551 |    19.993369 |   2.635551 |    19.993369 |
## | **2022-05-01** |   55654 |   2.632240 |    22.625609 |   2.632240 |    22.625609 |
## | **2020-02-01** |   54840 |   2.593740 |    25.219349 |   2.593740 |    25.219349 |
## | **2020-05-01** |   54409 |   2.573356 |    27.792705 |   2.573356 |    27.792705 |
## | **2022-03-01** |   52633 |   2.489357 |    30.282062 |   2.489357 |    30.282062 |
## | **2022-07-01** |   50089 |   2.369035 |    32.651097 |   2.369035 |    32.651097 |
## | **2024-01-01** |   49738 |   2.352434 |    35.003531 |   2.352434 |    35.003531 |
## | **2021-11-01** |   48730 |   2.304759 |    37.308290 |   2.304759 |    37.308290 |
## | **2023-05-01** |   48608 |   2.298989 |    39.607278 |   2.298989 |    39.607278 |
## | **2023-10-01** |   47983 |   2.269428 |    41.876707 |   2.269428 |    41.876707 |
## | **2022-04-01** |   47915 |   2.266212 |    44.142919 |   2.266212 |    44.142919 |
## | **2021-12-01** |   47257 |   2.235091 |    46.378010 |   2.235091 |    46.378010 |
## | **2023-08-01** |   46609 |   2.204443 |    48.582453 |   2.204443 |    48.582453 |
## | **2021-08-01** |   46072 |   2.179045 |    50.761497 |   2.179045 |    50.761497 |
## | **2022-02-01** |   45787 |   2.165565 |    52.927063 |   2.165565 |    52.927063 |
## | **2022-01-01** |   45587 |   2.156106 |    55.083169 |   2.156106 |    55.083169 |
## | **2023-07-01** |   45551 |   2.154403 |    57.237572 |   2.154403 |    57.237572 |
## | **2021-09-01** |   44213 |   2.091121 |    59.328692 |   2.091121 |    59.328692 |
## | **2023-11-01** |   43742 |   2.068844 |    61.397536 |   2.068844 |    61.397536 |
## | **2024-02-01** |   42635 |   2.016487 |    63.414023 |   2.016487 |    63.414023 |
## | **2021-10-01** |   42228 |   1.997237 |    65.411260 |   1.997237 |    65.411260 |
## | **2021-06-01** |   41897 |   1.981582 |    67.392841 |   1.981582 |    67.392841 |
## | **2024-03-01** |   41652 |   1.969994 |    69.362836 |   1.969994 |    69.362836 |
## | **2023-12-01** |   41068 |   1.942373 |    71.305209 |   1.942373 |    71.305209 |
## | **2021-07-01** |   40923 |   1.935515 |    73.240724 |   1.935515 |    73.240724 |
## | **2023-06-01** |   40908 |   1.934806 |    75.175529 |   1.934806 |    75.175529 |
## | **2023-09-01** |   40606 |   1.920522 |    77.096051 |   1.920522 |    77.096051 |
## | **2020-08-01** |   40396 |   1.910590 |    79.006641 |   1.910590 |    79.006641 |
## | **2023-03-01** |   40321 |   1.907042 |    80.913683 |   1.907042 |    80.913683 |
## | **2023-04-01** |   38395 |   1.815949 |    82.729633 |   1.815949 |    82.729633 |
## | **2023-01-01** |   38117 |   1.802801 |    84.532434 |   1.802801 |    84.532434 |
## | **2023-02-01** |   34774 |   1.644689 |    86.177123 |   1.644689 |    86.177123 |
## | **2021-05-01** |   33873 |   1.602075 |    87.779197 |   1.602075 |    87.779197 |
## | **2021-03-01** |   33698 |   1.593798 |    89.372995 |   1.593798 |    89.372995 |
## | **2021-04-01** |   29660 |   1.402814 |    90.775809 |   1.402814 |    90.775809 |
## | **2022-12-01** |   28621 |   1.353673 |    92.129483 |   1.353673 |    92.129483 |
## | **2022-11-01** |   28161 |   1.331917 |    93.461400 |   1.331917 |    93.461400 |
```

```
## | **2022-09-01** |   27511 |   1.301174 |    94.762574 |   1.301174 |    94.762574 |
## | **2021-02-01** |   24825 |   1.174136 |    95.936710 |   1.174136 |    95.936710 |
## | **2021-01-01** |   23451 |   1.109150 |    97.045860 |   1.109150 |    97.045860 |
## | **2020-12-01** |   23275 |   1.100826 |    98.146686 |   1.100826 |    98.146686 |
## | **2022-10-01** |   17886 |   0.845945 |    98.992632 |   0.845945 |    98.992632 |
## | **2020-11-01** |   16451 |   0.778075 |    99.770707 |   0.778075 |    99.770707 |
## | **2020-10-01** |    4846 |   0.229199 |    99.999905 |   0.229199 |    99.999905 |
## | **2020-09-01** |       2 |   0.000095 |   100.000000 |   0.000095 |   100.000000 |
## |     **\<NA\>** |       0 |            |              |   0.000000 |   100.000000 |
## |      **Total** | 2114321 | 100.000000 |   100.000000 | 100.000000 |   100.000000 |
```

```
sample_combustiveis_agg %>%
   dplyr::filter(Data == "2020-09-01")
```

```
##         Data Regiao   UF Municipio CNPJ_Revenda Produto Bandeira Valor_de_Venda
## 1 2020-09-01   <NA> <NA>      <NA>         <NA>    <NA>     <NA>             NA
## 2 2020-09-01   <NA> <NA>      <NA>         <NA>    <NA>     <NA>             NA
##   Valor_de_Compra     PPI Brent_USD_Barril   Cambio Brent_Real_Barril
## 1              NA 1.92770            40.91 5.398881          220.8682
## 2              NA 1.90792            40.91 5.398881          220.8682
```

Foi constatado durante as análises que no mês 09 de 2020 não ocorreu nenhuma pesquisa nos postos de combustíveis. Por este motivo, foi necessário remover o respectivo mês da base sample_combustiveis_agg.

```
sample_combustiveis_agg <- sample_combustiveis_agg %>% dplyr::filter(!Data == "2020-09-01")
```

Durando esta etapa de análise de frequências de variáveis qualitativas, vamos trabalhar para identificar os indivíduos de interesse no experimento.

Para este experimento os indivíduos possíveis mapeados são os postos de combustíveis identificados pela variável Revenda e CNPJ_Revenda, Município, UF e Regiao.

O primeiro indivíduo analisado foram os postos de combustíveis a partir do CNPJ_Revenda. Considerando ao volume de dados, a análise partiu para uma amostra menor que possibilita-se identificar que um determinado posto de combustível de uma cidade aleatório no tempo não foi pesquisado no tempo.

```
filter_municipio = "CAMPOS DOS GOYTACAZES"
filter_ano = "2020"

sample_combustiveis_agg %>%
   dplyr::mutate(ano = format(Data, "%Y")) %>%
   dplyr::filter(ano == filter_ano & Municipio == filter_municipio) %>%
   dplyr::select(CNPJ_Revenda) %>%
   summarytools::freq(., style = 'rmarkdown', order = "freq", plain.ascii = FALSE)
```

```
## ### Frequencies
## #### sample_combustiveis_agg$CNPJ_Revenda
## **Type:** Character
##
## |                  | Freq | % Valid | % Valid Cum. | % Total | % Total Cum. |
## |----------------------:|-----:|--------:|-------------:|--------:|-------------:|
## | **08.993.516/0001-96** |   60 |    3.90 |         3.90 |    3.90 |         3.90 |
## | **05.750.198/0001-44** |   57 |    3.70 |         7.60 |    3.70 |         7.60 |
```

03.771.829/0001-86	54	3.51	11.10	3.51	11.10
39.229.745/0002-43	51	3.31	14.42	3.31	14.42
03.860.043/0001-35	48	3.12	17.53	3.12	17.53
28.954.287/0001-08	46	2.99	20.52	2.99	20.52
28.934.362/0001-79	45	2.92	23.44	2.92	23.44
07.961.986/0001-05	44	2.86	26.30	2.86	26.30
12.561.113/0003-07	42	2.73	29.03	2.73	29.03
13.412.918/0001-90	39	2.53	31.56	2.53	31.56
02.793.152/0001-14	37	2.40	33.96	2.40	33.96
09.133.553/0001-97	36	2.34	36.30	2.34	36.30
28.794.279/0001-41	34	2.21	38.51	2.21	38.51
28.890.267/0001-10	30	1.95	40.45	1.95	40.45
12.561.113/0001-45	28	1.82	42.27	1.82	42.27
11.804.672/0001-76	27	1.75	44.03	1.75	44.03
36.181.766/0001-67	26	1.69	45.71	1.69	45.71
03.784.646/0001-03	25	1.62	47.34	1.62	47.34
28.871.135/0001-41	25	1.62	48.96	1.62	48.96
00.647.473/0001-85	24	1.56	50.52	1.56	50.52
03.360.754/0001-40	24	1.56	52.08	1.56	52.08
07.631.165/0001-00	24	1.56	53.64	1.56	53.64
17.305.026/0001-40	24	1.56	55.19	1.56	55.19
19.475.429/0001-63	24	1.56	56.75	1.56	56.75
20.437.137/0001-15	24	1.56	58.31	1.56	58.31
04.718.173/0001-09	21	1.36	59.68	1.36	59.68
32.360.034/0001-83	21	1.36	61.04	1.36	61.04
05.064.564/0001-01	20	1.30	62.34	1.30	62.34
05.809.583/0001-10	20	1.30	63.64	1.30	63.64
02.503.556/0001-26	18	1.17	64.81	1.17	64.81
05.241.281/0001-98	18	1.17	65.97	1.17	65.97
07.614.346/0004-71	18	1.17	67.14	1.17	67.14
39.237.292/0001-16	18	1.17	68.31	1.17	68.31
03.455.258/0001-70	16	1.04	69.35	1.04	69.35
28.931.491/0001-03	16	1.04	70.39	1.04	70.39
08.652.445/0001-68	15	0.97	71.36	0.97	71.36
05.064.564/0002-92	14	0.91	72.27	0.91	72.27
07.614.346/0002-00	14	0.91	73.18	0.91	73.18
07.614.346/0003-90	14	0.91	74.09	0.91	74.09
05.308.115/0001-61	13	0.84	74.94	0.84	74.94
03.433.342/0001-93	12	0.78	75.71	0.78	75.71
03.497.607/0001-17	12	0.78	76.49	0.78	76.49
03.750.110/0002-40	12	0.78	77.27	0.78	77.27
07.756.124/0001-40	12	0.78	78.05	0.78	78.05
09.517.520/0001-40	12	0.78	78.83	0.78	78.83
09.667.118/0001-42	12	0.78	79.61	0.78	79.61
10.730.629/0001-40	12	0.78	80.39	0.78	80.39
20.638.205/0001-04	12	0.78	81.17	0.78	81.17
28.875.086/0002-04	12	0.78	81.95	0.78	81.95
28.888.915/0001-02	12	0.78	82.73	0.78	82.73
31.212.889/0001-02	12	0.78	83.51	0.78	83.51
39.713.615/0001-09	12	0.78	84.29	0.78	84.29
03.312.137/0001-70	11	0.71	85.00	0.71	85.00
03.581.079/0002-60	11	0.71	85.71	0.71	85.71
04.968.522/0001-32	11	0.71	86.43	0.71	86.43
26.179.832/0001-00	11	0.71	87.14	0.71	87.14

```
## | **07.105.628/0001-09** |   10 |    0.65 |        87.79 |    0.65 |        87.79 |
## | **10.820.316/0001-83** |   10 |    0.65 |        88.44 |    0.65 |        88.44 |
## | **13.639.940/0001-77** |    9 |    0.58 |        89.03 |    0.58 |        89.03 |
## | **28.875.086/0001-15** |    9 |    0.58 |        89.61 |    0.58 |        89.61 |
## | **39.238.340/0001-90** |    9 |    0.58 |        90.19 |    0.58 |        90.19 |
## | **02.846.659/0001-99** |    8 |    0.52 |        90.71 |    0.52 |        90.71 |
## | **03.428.937/0001-50** |    8 |    0.52 |        91.23 |    0.52 |        91.23 |
## | **03.484.679/0001-20** |    8 |    0.52 |        91.75 |    0.52 |        91.75 |
## | **03.854.073/0001-39** |    8 |    0.52 |        92.27 |    0.52 |        92.27 |
## | **04.762.533/0001-61** |    8 |    0.52 |        92.79 |    0.52 |        92.79 |
## | **14.610.456/0001-88** |    8 |    0.52 |        93.31 |    0.52 |        93.31 |
## | **27.593.110/0001-60** |    8 |    0.52 |        93.83 |    0.52 |        93.83 |
## | **28.346.559/0001-97** |    8 |    0.52 |        94.35 |    0.52 |        94.35 |
## | **28.935.153/0001-40** |    8 |    0.52 |        94.87 |    0.52 |        94.87 |
## | **31.212.889/0002-85** |    8 |    0.52 |        95.39 |    0.52 |        95.39 |
## | **39.235.973/0001-45** |    8 |    0.52 |        95.91 |    0.52 |        95.91 |
## | **39.686.407/0001-50** |    8 |    0.52 |        96.43 |    0.52 |        96.43 |
## | **39.695.887/0001-15** |    8 |    0.52 |        96.95 |    0.52 |        96.95 |
## | **40.218.026/0001-25** |    8 |    0.52 |        97.47 |    0.52 |        97.47 |
## | **05.318.203/0001-44** |    7 |    0.45 |        97.92 |    0.45 |        97.92 |
## | **06.057.223/0117-00** |    7 |    0.45 |        98.38 |    0.45 |        98.38 |
## | **07.937.096/0001-68** |    6 |    0.39 |        98.77 |    0.39 |        98.77 |
## | **06.211.735/0001-40** |    5 |    0.32 |        99.09 |    0.32 |        99.09 |
## | **05.786.901/0001-74** |    4 |    0.26 |        99.35 |    0.26 |        99.35 |
## | **07.631.180/0001-59** |    4 |    0.26 |        99.61 |    0.26 |        99.61 |
## | **29.952.132/0001-03** |    4 |    0.26 |        99.87 |    0.26 |        99.87 |
## | **31.273.089/0001-93** |    2 |    0.13 |       100.00 |    0.13 |       100.00 |
## |              **\<NA\>** |    0 |         |              |    0.00 |       100.00 |
## |               **Total** | 1540 |  100.00 |       100.00 |  100.00 |       100.00 |
```

É possível observar que determinados indivíduos denominados Revendas (postos de combustíveis) não contribuem mensalmente para a pesquisa, ratificando a metodologia de que a pesquisa de preços abrange a seleção das revendas em operação cadastradas na ANP, as quais farão parte da amostra.

Diante deste fato, o próximo passo foi avaliar o indivíduo Município

A mesma estratégia foi utilizada, a partir de uma amostra menor que ratifica-se a hipóteses de que nem todos os municípios são pesquisados todos os meses.

```
filter_uf = "RJ"
filter_ano = "2020"

sample_combustiveis_agg %>%
   dplyr::mutate(ano = format(Data, "%Y")) %>%
   dplyr::filter(ano == filter_ano & UF == filter_uf) %>%
   dplyr::select(Municipio) %>%
   summarytools::freq(., style = 'rmarkdown', order = "freq", plain.ascii = FALSE)
```

```
## ### Frequencies
## #### sample_combustiveis_agg$Municipio
## **Type:** Character
##
## |                              |  Freq | % Valid | % Valid Cum. | % Total | % Total Cum. |
## |----------------------------------:|------:|--------:|-------------:|--------:|-------------:|
```

RIO DE JANEIRO	6983	19.98	19.98	19.98	19.98	19.98
DUQUE DE CAXIAS	2164	6.19	26.18	6.19	26.18	26.18
SAO GONCALO	1691	4.84	31.02	4.84	31.02	31.02
NOVA IGUACU	1616	4.62	35.64	4.62	35.64	35.64
CAMPOS DOS GOYTACAZES	1540	4.41	40.05	4.41	40.05	40.05
NITEROI	1477	4.23	44.28	4.23	44.28	44.28
PETROPOLIS	1327	3.80	48.07	3.80	48.07	48.07
BELFORD ROXO	1019	2.92	50.99	2.92	50.99	50.99
BARRA MANSA	984	2.82	53.81	2.82	53.81	53.81
SAO JOAO DE MERITI	953	2.73	56.53	2.73	56.53	56.53
TERESOPOLIS	926	2.65	59.18	2.65	59.18	59.18
ITABORAI	898	2.57	61.75	2.57	61.75	61.75
SAPUCAIA	871	2.49	64.25	2.49	64.25	64.25
NOVA FRIBURGO	869	2.49	66.73	2.49	66.73	66.73
ARARUAMA	825	2.36	69.09	2.36	69.09	69.09
VOLTA REDONDA	822	2.35	71.45	2.35	71.45	71.45
VALENCA	818	2.34	73.79	2.34	73.79	73.79
RESENDE	781	2.24	76.02	2.24	76.02	76.02
TRES RIOS	769	2.20	78.22	2.20	78.22	78.22
MARICA	755	2.16	80.38	2.16	80.38	80.38
MACAE	729	2.09	82.47	2.09	82.47	82.47
BARRA DO PIRAI	672	1.92	84.39	1.92	84.39	84.39
ITAPERUNA	665	1.90	86.30	1.90	86.30	86.30
ITAGUAI	654	1.87	88.17	1.87	88.17	88.17
CABO FRIO	640	1.83	90.00	1.83	90.00	90.00
SAQUAREMA	584	1.67	91.67	1.67	91.67	91.67
RIO BONITO	565	1.62	93.29	1.62	93.29	93.29
SAO FRANCISCO DE ITABAPOANA	528	1.51	94.80	1.51	94.80	94.80
SANTO ANTONIO DE PADUA	512	1.47	96.27	1.47	96.27	96.27
MAGE	490	1.40	97.67	1.40	97.67	97.67
ANGRA DOS REIS	456	1.31	98.97	1.31	98.97	98.97
NILOPOLIS	359	1.03	100.00	1.03	100.00	100.00
\<NA\>	0			0.00	100.00	
Total	34942	100.00	100.00	100.00	100.00	

Para apurar os municípios que ratificam a hipótese supra, optou-se por analisar os municípios com menor frequência apresentada.

Considerando o município de Nilópolis que possui a menor amostra, foi possível constatar que ele foi pesquisado todos os meses.

O próximo a ser analisado foi o município de Angra dos Reis. Foi constatado que ele não foi pesquisado no mês 11 de 2020.

Portanto, esta análise ratifica a hipótese apresentada, portanto, desconsiderando os municípios como indivíduos para este experimento.

```
filter_uf = "RJ"
filter_municipio = "ANGRA DOS REIS"
filter_ano = "2020"

sample_combustiveis_agg %>%
   dplyr::mutate(mes = format(Data, "%m"), ano = format(Data, "%Y")) %>%
   dplyr::filter(UF == filter_uf, Municipio == filter_municipio, ano == filter_ano) %>%
   dplyr::group_by(mes) %>%
```

```
   dplyr::mutate(n_linhas = 1) %>%
   dplyr::summarise(n_linhas = sum(n_linhas))
```

```
## # A tibble: 9 x 2
##   mes   n_linhas
##   <chr>    <dbl>
## 1 01          63
## 2 02          52
## 3 03          52
## 4 04          65
## 5 05          52
## 6 06          65
## 7 07          52
## 8 08          39
## 9 12          16
```

É possível observar que determinados indivíduos denominados Municípios não contribuem mensalmente para a pesquisa.

Diante deste fato, o próximo passo foi avaliar o indivíduo UF. Na etapa de **Analisando os tipos das variáveis nas bases** é possível constatar que são 27 unidades federativas (UF) pesquisadas.

A primeira pergunta a ser respondida é: A base de amostra de nome sample_combustiveis_agg continua com a mesma quantidade da base original de nome origin_combustiveis_agg?

```
sample_combustiveis_agg %>% dlookr::diagnose()
```

```
## # A tibble: 13 x 6
##    variables        types missing_count missing_percent unique_count unique_rate
##    <chr>            <chr>         <int>           <dbl>        <int>       <dbl>
##  1 Data             Date              0             0             50  0.0000236
##  2 Regiao           char~             0             0              5  0.00000236
##  3 UF               char~             0             0             27  0.0000128
##  4 Municipio        char~             0             0            469  0.000222
##  5 CNPJ_Revenda     char~             0             0          19927  0.00942
##  6 Produto          char~             0             0              3  0.00000142
##  7 Bandeira         char~             0             0             79  0.0000374
##  8 Valor_de_Venda   nume~             0             0           4203  0.00199
##  9 Valor_de_Compra  nume~       2002006          94.7          18046  0.00854
## 10 PPI              nume~             0             0            100  0.0000473
## 11 Brent_USD_Barril nume~             0             0             50  0.0000236
## 12 Cambio           nume~             0             0             50  0.0000236
## 13 Brent_Real_Barr~ nume~             0             0             50  0.0000236
```

É possível verificar que sim. O próximo passo é verificar se todos os 27 estados são pesquisados mensalmente.

```
filter_ano = "2020"

sample_combustiveis_agg %>%
   dplyr::mutate(mes = format(Data, "%m"), ano = format(Data, "%Y")) %>%
#   dplyr::filter(ano == filter_ano) %>%
   dplyr::group_by(mes, ano) %>%
   dplyr::distinct(UF) %>%
```

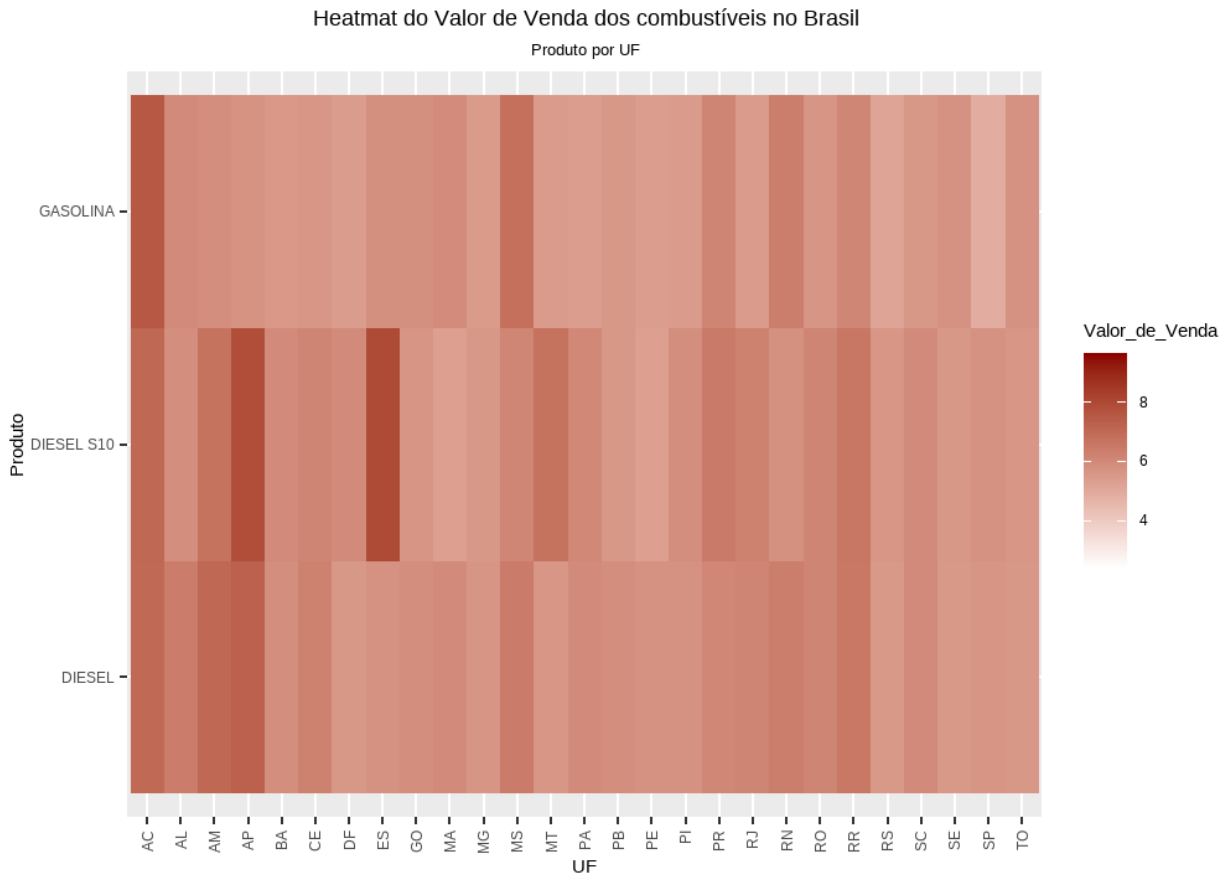
```
   dplyr::count(UF) %>%
   dplyr::summarise(n_uf = sum(n), .groups = 'drop')
```

```
## # A tibble: 50 x 3
##    mes   ano    n_uf
##    <chr> <chr> <int>
##  1 01    2020     27
##  2 01    2021     27
##  3 01    2022     27
##  4 01    2023     27
##  5 01    2024     27
##  6 02    2020     27
##  7 02    2021     27
##  8 02    2022     27
##  9 02    2023     27
## 10 02    2024     27
## # i 40 more rows
```

É possível observar que todos os indivíduos denominados UF contribuíram mensalmente para a pesquisa. Tal afirmação só é valida porque o Brasil é consituído por 26 estados e o Distrito Federal, contabilizando assim 27 unidades federativas.

A análise a seguir visa apresentar um mapa de calor a partir do valor de venda dos combustíveis no Brasil e posteriormente por município da UF selecionada para análise.

```
filter_produto = c("GASOLINA","DIESEL","DIESEL S10")
graph_file <- "Heatmat do Valor de Venda dos combustíveis no Brasil"
extensao_file <- "jpg"

sample_combustiveis_agg %>%
   dplyr::filter(Produto %in% filter_produto) %>%
   ggplot(aes(x = UF,
              y = Produto,
              fill = Valor_de_Venda)) +
   geom_tile() +
   labs(x = "UF",
        y = "Produto",
        title = graph_file,
        subtitle = "Produto por UF") +
   theme(axis.text.x = element_text(angle = 90,
                                    hjust = 1,
                                    vjust = 0.5),
         plot.title = element_text(hjust = 0.5),
         plot.subtitle = element_text(hjust = 0.5,
                                      size = 10)) +
   scale_fill_gradient(low = "white", high = "darkred")
```

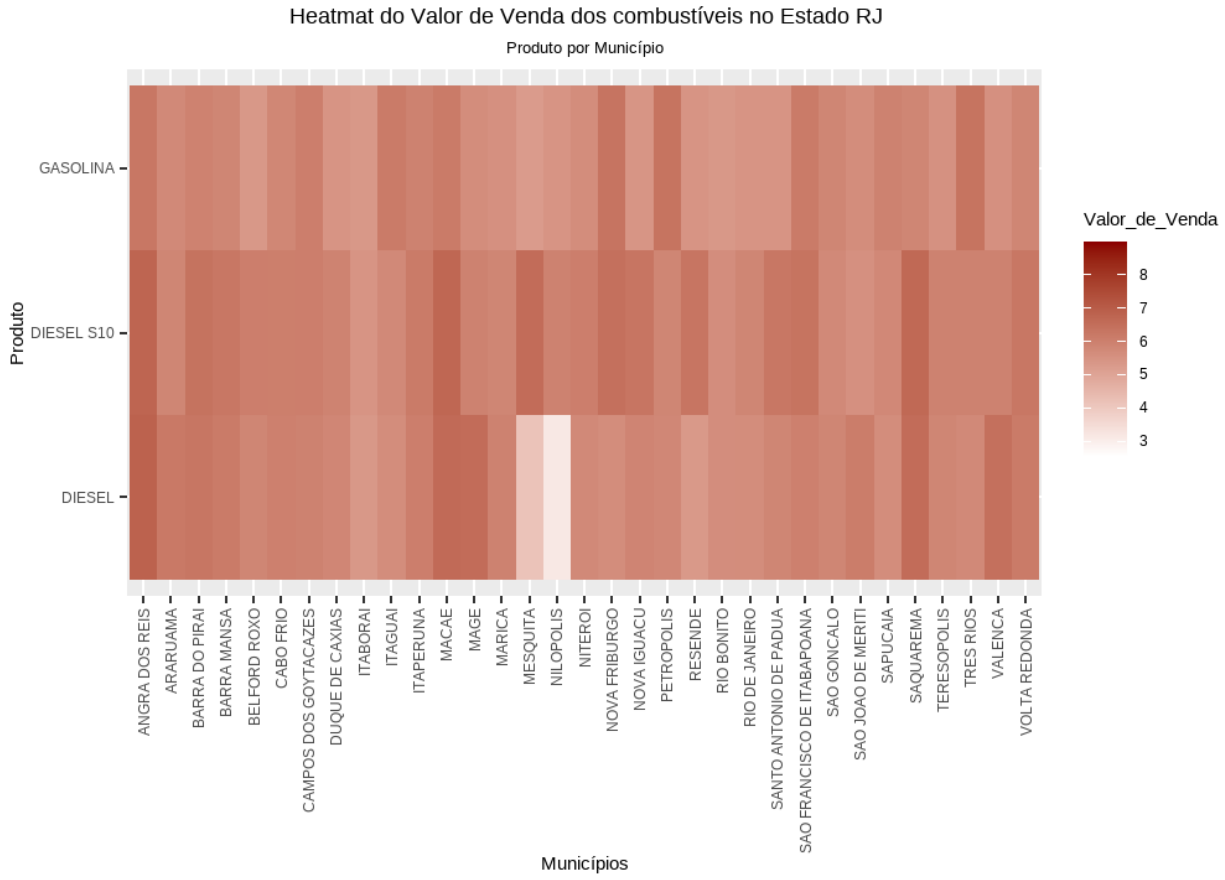



```
ggsave(paste(graph_file, extensao_file, sep = "."), path = "./Images")
```

```
## Saving 7 x 5 in image
```

```
filter_uf = "RJ"
filter_produto = c("GASOLINA","DIESEL","DIESEL S10")
graph_file <- paste("Heatmat do Valor de Venda dos combustíveis no Estado",filter_uf, sep = " ")
extensao_file <- "jpg"

sample_combustiveis_agg %>%
   dplyr::filter(Produto %in% filter_produto, UF == filter_uf) %>%
   ggplot(aes(x = Municipio,
              y = Produto,
              fill = Valor_de_Venda)) +
   geom_tile() +
   labs(x = "Municípios",
        y = "Produto",
        title = graph_file,
        subtitle = "Produto por Município") +
   theme(axis.text.x = element_text(angle = 90,
                                    hjust = 1,
                                    vjust = 0.5),
         plot.title = element_text(hjust = 0.5),
         plot.subtitle = element_text(hjust = 0.5,
```

```
                                    size = 10)) +
  scale_fill_gradient(low = "white", high = "darkred")
```




```
ggsave(paste(graph_file, extensao_file, sep = "."), path = "./Images")
```

```
## Saving 7 x 5 in image
```

Análise descritiva e de histogramas das variáveis contínuas

A próximas etapas da análise foi elaborada considerando a possibilidade de análise a partir da variável produto em relação ao indivíduo UF.

Em relação a binarização, é possível considerar regras de binarização levando em consideração regras disponíveis na literatura, como a regra de Freedman-Diaconis, bem como a regra de Sturge. Para esta análise será utilziada a regra de Freedman-Diaconis.

Como amostra para a respectiva análise, considerou-se o estado do RJ para o produto gasolina. Vale ressaltar que tal análise poderá ser feita para qualquer estado e produto constante na pesquisa realizada pela ANP.

Para as próximas análises, recomanda-se definir os parâmetros das variáveis seguintes.

```r
filter_uf = "RJ"
filter_produto = "GASOLINA"
```

Análise descritiva das variáveis contínuas

Para a variável Valor_de_Venda, podemos analisar a centralidade dos dados, dipersão, assimetria, bem como suas estatísticas de ordem, a fim de checar se há presença de outliers.

Para realizar essa análise, utilizar-se-á a função descr do pacote summarytools, e posteriormente realizar a leitura desses dados.

Analise descritiva da variável Valor de Venda por Produto no Brasil.

```r
sample_combustiveis_agg %>%
   dplyr::filter(Produto == filter_produto) %>%
   dplyr::select(Valor_de_Venda) %>%
   summarytools::descr()
```

```
## Descriptive Statistics
## sample_combustiveis_agg$Valor_de_Venda
## N: 940511
##
##                     Valor_de_Venda
## ----------------- ----------------
##              Mean             5.53
##           Std.Dev             0.98
##               Min             2.87
##                Q1             4.84
##            Median             5.49
##                Q3             6.13
##               Max             8.99
##               MAD             0.96
##               IQR             1.29
##                CV             0.18
##          Skewness             0.22
##       SE.Skewness             0.00
##          Kurtosis            -0.39
##           N.Valid        940511.00
##         Pct.Valid           100.00
```

É possível ver pelo critério de skewness, que o valor está entre 0 e 0.5 para assimetria, nos permitindo interpretar que esta distribução possui assimetria leve, com cauda à direita.

Em decorrência desta assimetria, observamos que média e mediana apresentam valores distintos, com a média tendo valor levemente superior, o que aponta que os valores mais distantes do centro da distribuição puxam o valor da média pra cima.

Já a mediana por ser uma estatística de ordem, não é sensível a dados que apresentam alto valor na distribuição, o que é reforçado por seu valor levemente mais baixo que a média.

Se houve-se outliers nesta distribuição a média se descolaria ainda mais da mediana, pois estaria totalmente suscetível à contaminação.

```r
sample_combustiveis_agg %>%
   dplyr::filter(Produto == filter_produto, UF == filter_uf) %>%
   dplyr::select(Valor_de_Venda) %>%
   summarytools::descr()
```

```
## Descriptive Statistics
## sample_combustiveis_agg$Valor_de_Venda
## N: 66160
##
##                     Valor_de_Venda
## ----------------- ----------------
##              Mean             5.82
##           Std.Dev             1.00
##               Min             3.86
##                Q1             5.00
##            Median             5.59
##                Q3             6.39
##               Max             8.99
##               MAD             0.90
##               IQR             1.39
##                CV             0.17
##          Skewness             0.65
##       SE.Skewness             0.01
##          Kurtosis            -0.41
##           N.Valid         66160.00
##         Pct.Valid           100.00
```

É possível ver pelo critério de skewness, que o valor está entre 0.5 e 1 para assimetria, nos permitindo interpretar que esta distribução possui assimetria moderada, com cauda à direita.

É possível observar a diferença entre o Brasil e o RJ, considerando que possívelmente o RJ possui mais outliers representativos.

Em decorrência desta assimetria, observamos que média e mediana apresentam valores distintos, com a média tendo valor levemente superior, o que aponta que os valores mais distantes do centro da distribuição puxam o valor da média pra cima.

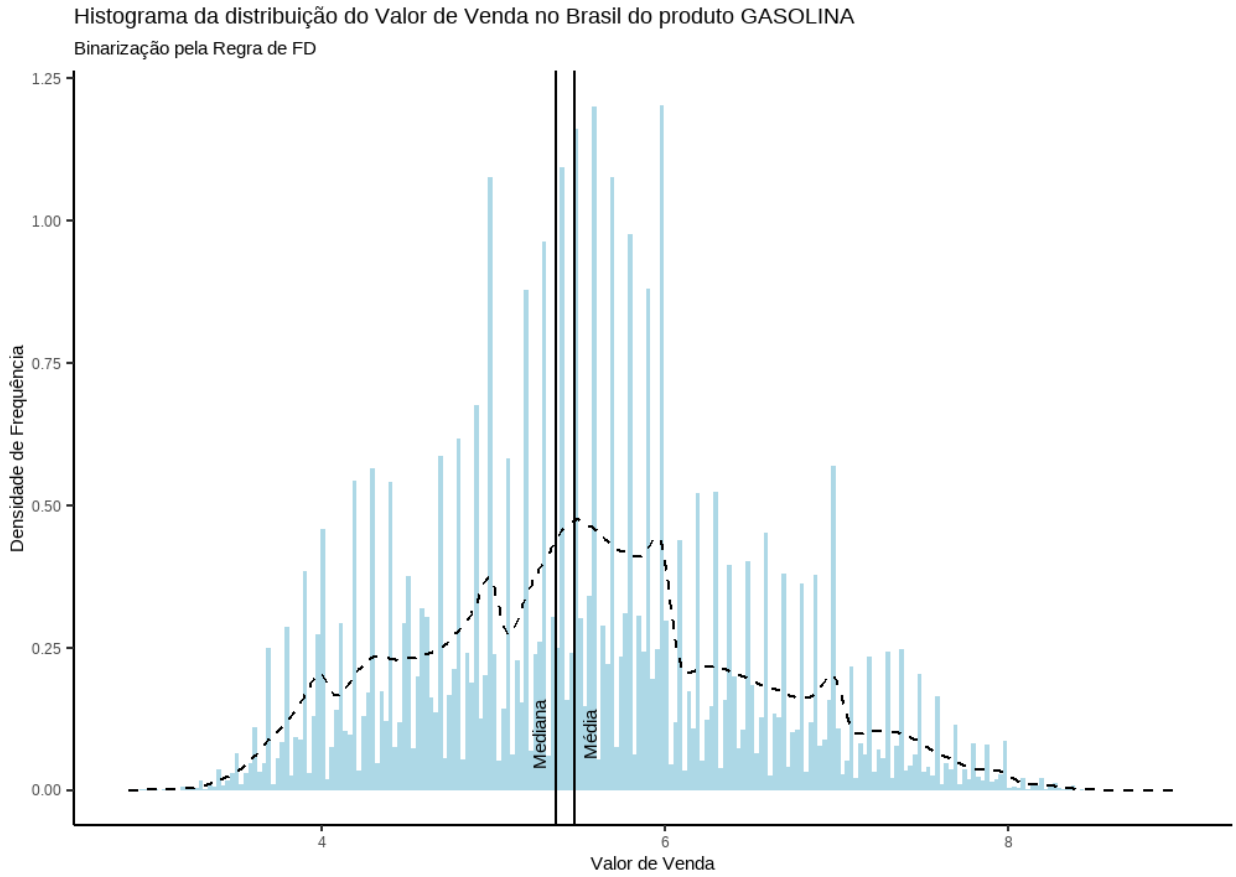
Já a mediana por ser uma estatística de ordem, não é sensível a dados que apresentam alto valor na distribuição, o que é reforçado por seu valor levemente mais baixo que a média.

Histogramas das variáveis contínuas

Histograma da variável Valor de Venda por Produto no Brasil.

```r
graph_file <- paste("Histograma da distribuição do Valor de Venda no Brasil do produto", filter_produto
extensao_file <- "jpg"

sample_combustiveis_agg %>%
   dplyr::filter(Produto == filter_produto) %>%
   dplyr::select(Valor_de_Venda) %>%
   ggplot(aes(x=Valor_de_Venda)) +
   geom_histogram(aes(y = after_stat(density)) , binwidth=fd, fill = 'lightblue') +
   xlab('Valor de Venda') +
```

```
  ylab('Densidade de Frequência') +
  labs(title = graph_file,
       subtitle = "Binarização pela Regra de FD") +
  geom_vline(xintercept=c(median(sample_combustiveis_agg$Valor_de_Venda),
                          mean(sample_combustiveis_agg$Valor_de_Venda))) +
  annotate("text", x=median(sample_combustiveis_agg$Valor_de_Venda) +
             -0.2, y=0.1, label="Mediana", angle=90) +
  annotate("text", x=mean(sample_combustiveis_agg$Valor_de_Venda) +
             0.2, y=0.1, label="Média", angle=90) +
  geom_density(linetype = 2) +
  theme_classic()
```



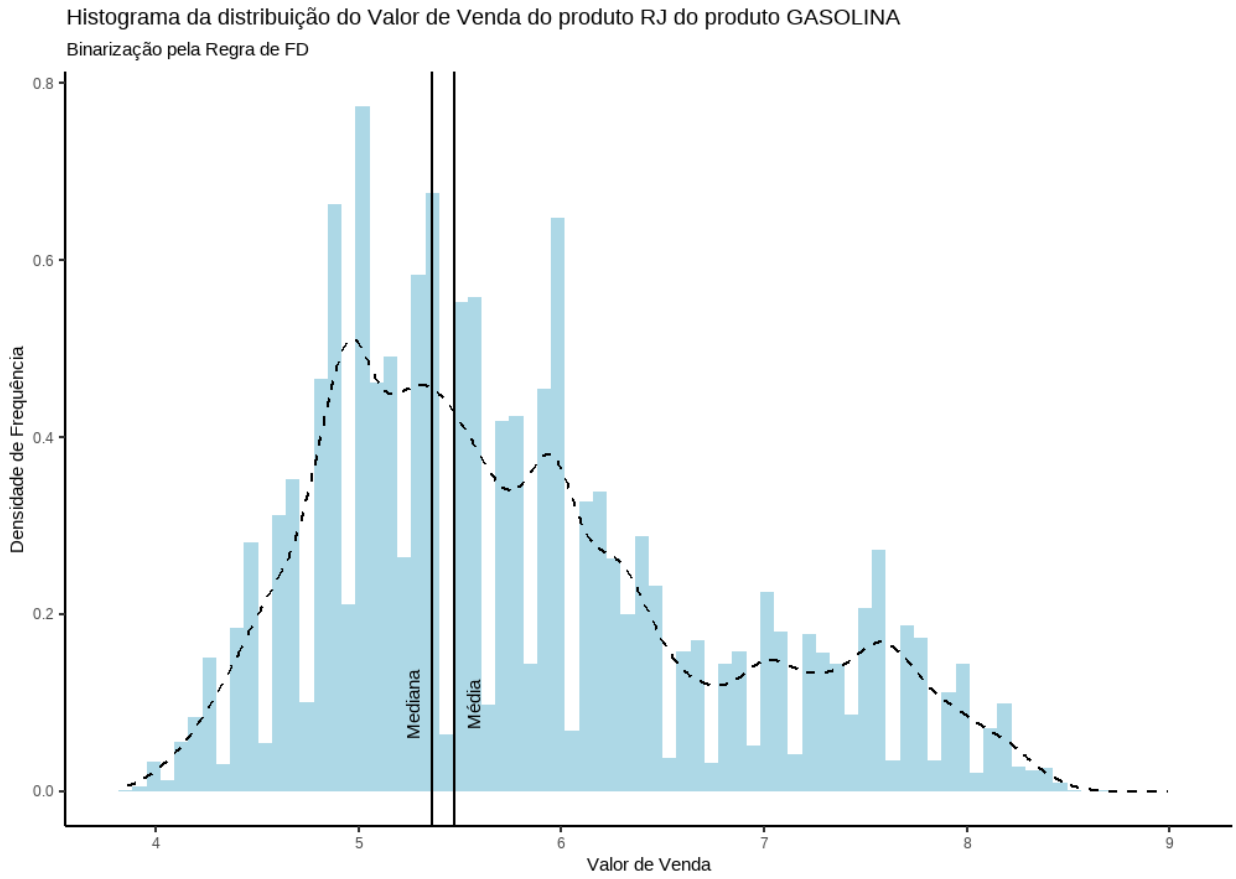

```
ggsave(paste(graph_file, extensao_file, sep = "."), path = "./Images")
```

```
## Saving 7 x 5 in image
```

Histograma da variável Valor de Venda por Produto na UF selecionada.

```
graph_file <- paste("Histograma da distribuição do Valor de Venda do produto",filter_uf,"do produto",fi
extensao_file <- "jpg"

sample_combustiveis_agg %>%
```

```
  dplyr::filter(Produto == filter_produto, UF == filter_uf) %>%
  dplyr::select(Valor_de_Venda) %>%
  ggplot(aes(x=Valor_de_Venda)) +
  geom_histogram(aes(y = after_stat(density)) , binwidth=fd, fill = 'lightblue') +
  xlab('Valor de Venda') +
  ylab('Densidade de Frequência') +
  labs(title = graph_file,
       subtitle = "Binarização pela Regra de FD") +
  geom_vline(xintercept=c(median(sample_combustiveis_agg$Valor_de_Venda),
                          mean(sample_combustiveis_agg$Valor_de_Venda))) +
  annotate("text", x=median(sample_combustiveis_agg$Valor_de_Venda) +
             -0.2, y=0.1, label="Mediana", angle=90) +
  annotate("text", x=mean(sample_combustiveis_agg$Valor_de_Venda) +
             0.2, y=0.1, label="Média", angle=90) +
  geom_density(linetype = 2) +
  theme_classic()
```




```
ggsave(paste(graph_file, extensao_file, sep = "."), path = "./Images")
```

```
## Saving 7 x 5 in image
```

Um conceito para esta etapa da análise é fundamental compreendermos.

O que é uma distribuição normal?

Podemos conceituar como sendo uma distribuição estatística no formato de um sino e simétrica em relação a média.

O que simboliza o formato de um sino?

A maioria dos dados estão concentrados no centro, diminuindo a quantidade destes dados em ambas as direções.

O que é a simetria em relação a média?

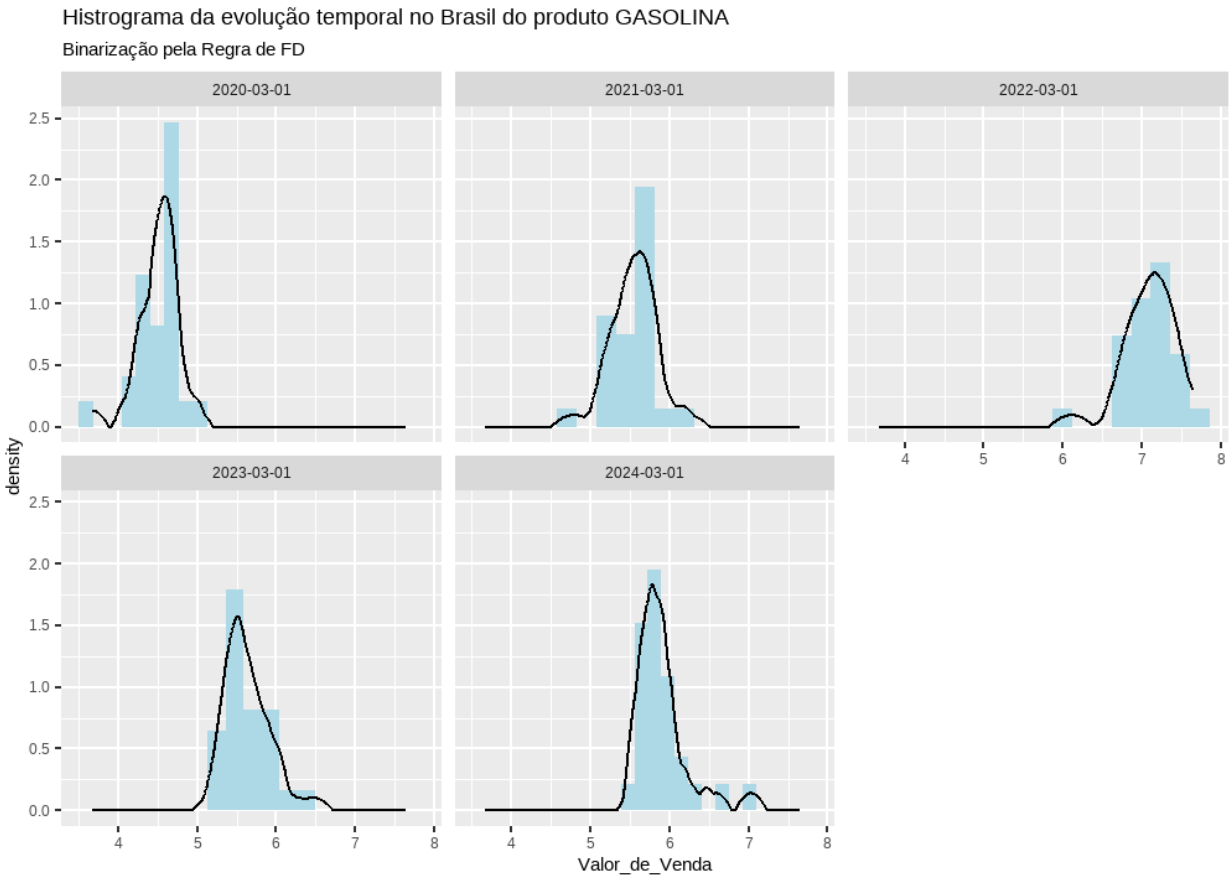
O termo simetria em relação a média é nada mais do que os valores da mediana e moda concidirem com o valor da média.

Portanto, pode-se dizer que para que haja simetria e o formato de sino, é necessário que a média, mediana e moda possuam o mesmo valor e o quantitativo de valores do lado esquerdo e direito da média são iguais.

```
sample_combustiveis_agg_periodo <- sample_combustiveis_agg %>%
   dplyr::select(UF, Produto, Data, Valor_de_Venda, Cambio, Brent_USD_Barril, Brent_Real_Barril, PPI) %
   dplyr::group_by(UF, Produto, Data) %>%
   dplyr::summarise(
      Valor_de_Venda = mean(Valor_de_Venda),
      Cambio = mean(Cambio),
      Brent_USD_Barril = mean(Brent_USD_Barril),
      Brent_Real_Barril = mean(Brent_Real_Barril),
      PPI = mean(PPI),
      .groups = 'drop') %>%
   dplyr::filter(Data %in% c(as.IDate("01-03-2020",
                                      format = "%d-%m-%Y"),
                             as.IDate("01-03-2021",
                                      format = "%d-%m-%Y"),
                             as.IDate("01-03-2022",
                                      format = "%d-%m-%Y"),
                             as.IDate("01-03-2023",
                                      format = "%d-%m-%Y"),
                             as.IDate("01-03-2024",
                                      format = "%d-%m-%Y"))
                )
```

Calculando os histogramas para o mesmo evento em diferentes instantes de tempo.

```
graph_file <- paste("Histrograma da evolução temporal no Brasil do produto", filter_produto, sep = " ")
extensao_file <- "jpg"

sample_combustiveis_agg_periodo %>%
   dplyr::filter(Produto == filter_produto) %>%
   ggplot(aes(x = Valor_de_Venda)) +
   geom_histogram(aes(y = after_stat(density)),
                  binwidth=fd,
                  fill = 'lightblue') +
   geom_density(kernel = 'epanechnikov') +
   labs(title = graph_file,
        subtitle = "Binarização pela Regra de FD") +
   facet_wrap(~Data)
```




```
ggsave(paste(graph_file, extensao_file, sep = "."), path = "./Images")
```

```
## Saving 7 x 5 in image
```

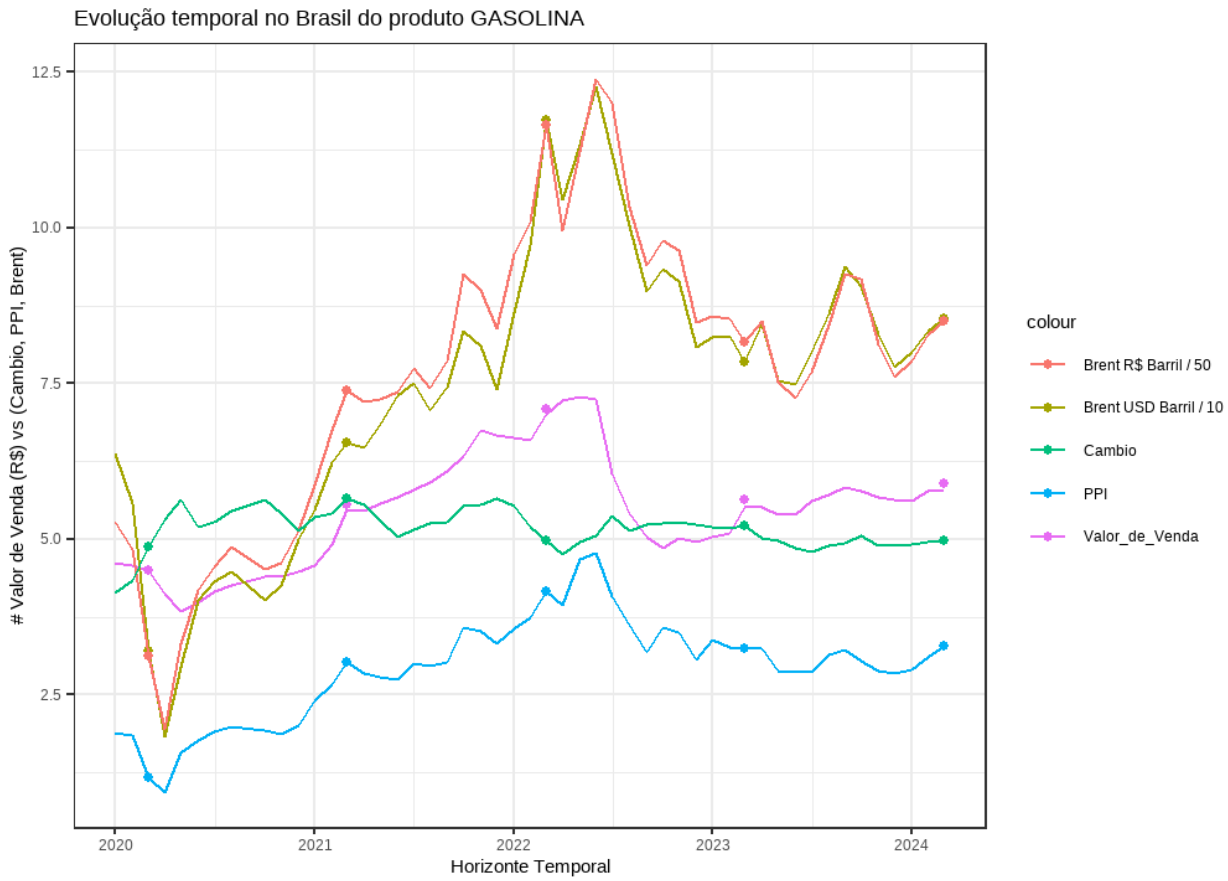
Neste histograma, considerando o custo de venda da gasolina para todo o Brasil em 03/2020 o custo da gasolina era mais barato, sofrente um aumento significativo 03/2022. É possível observar uma evolução de tendência de aumento entre 2020 e 2022. Em 03/2023 o valor de venda valta ao patamar aproximado de 03/2021. É possível observar uma tendência de curva gaussiana (distribuição normal) em alguns anos.

Analise temporal do combustivel vs Cambio vs Brent vc PPI.

```
graph_file <- paste("Evolução temporal no Brasil do produto", filter_produto, sep = " ")
extensao_file <- "jpg"

sample_combustiveis_agg %>%
   dplyr::filter(Produto == filter_produto) %>%
   dplyr::select(Data, Valor_de_Venda, Cambio, Brent_USD_Barril, Brent_Real_Barril, PPI) %>%
   dplyr::group_by(Data) %>%
   dplyr::summarise(
      Valor_de_Venda = mean(Valor_de_Venda),
      Cambio = mean(Cambio),
      Brent_USD_Barril = mean(Brent_USD_Barril),
      Brent_Real_Barril = mean(Brent_Real_Barril),
```

```
    PPI = mean(PPI)
) %>%

ggplot(aes(x = Data)) +
geom_line(aes(y = Valor_de_Venda, color = "Valor_de_Venda")) +

geom_point(data = sample_combustiveis_agg_periodo %>%
               dplyr::filter(Produto == filter_produto) %>%
               dplyr::group_by(Data) %>%
               dplyr::summarise(
                  Valor_de_Venda = mean(Valor_de_Venda),
                  Cambio = mean(Cambio),
                  Brent_USD_Barril = mean(Brent_USD_Barril),
                  Brent_Real_Barril = mean(Brent_Real_Barril),
                  PPI = mean(PPI),
                  .groups = 'drop'),
           aes(y = Valor_de_Venda, color = "Valor_de_Venda")) +

geom_line(aes(y = Brent_USD_Barril / 10, color = "Brent USD Barril / 10")) +

geom_point(data = sample_combustiveis_agg_periodo %>%
               dplyr::filter(Produto == filter_produto) %>%
               dplyr::group_by(Data) %>%
               dplyr::summarise(
                  Valor_de_Venda = mean(Valor_de_Venda),
                  Cambio = mean(Cambio),
                  Brent_USD_Barril = mean(Brent_USD_Barril),
                  Brent_Real_Barril = mean(Brent_Real_Barril),
                  PPI = mean(PPI),
                  .groups = 'drop'),
           aes(y = Brent_USD_Barril / 10, color = "Brent USD Barril / 10")) +

geom_line(aes(y = Brent_Real_Barril / 50, color = "Brent R$ Barril / 50")) +

geom_point(data = sample_combustiveis_agg_periodo %>%
               dplyr::filter(Produto == filter_produto) %>%
               dplyr::group_by(Data) %>%
               dplyr::summarise(
                  Valor_de_Venda = mean(Valor_de_Venda),
                  Cambio = mean(Cambio),
                  Brent_USD_Barril = mean(Brent_USD_Barril),
                  Brent_Real_Barril = mean(Brent_Real_Barril),
                  PPI = mean(PPI),
                  .groups = 'drop'),
           aes(y = Brent_Real_Barril / 50, color = "Brent R$ Barril / 50")) +

geom_line(aes(y = PPI, color = "PPI")) +

geom_point(data = sample_combustiveis_agg_periodo %>%
               dplyr::filter(Produto == filter_produto) %>%
               dplyr::group_by(Data) %>%
               dplyr::summarise(
                  Valor_de_Venda = mean(Valor_de_Venda),
```

```
                    Cambio = mean(Cambio),
                    Brent_USD_Barril = mean(Brent_USD_Barril),
                    Brent_Real_Barril = mean(Brent_Real_Barril),
                    PPI = mean(PPI),
                    .groups = 'drop'),
             aes(y = PPI, color = "PPI")) +

  geom_line(aes(y = Cambio, color = "Cambio")) +

  geom_point(data = sample_combustiveis_agg_periodo %>%
                dplyr::filter(Produto == filter_produto) %>%
                dplyr::group_by(Data) %>%
                dplyr::summarise(
                    Valor_de_Venda = mean(Valor_de_Venda),
                    Cambio = mean(Cambio),
                    Brent_USD_Barril = mean(Brent_USD_Barril),
                    Brent_Real_Barril = mean(Brent_Real_Barril),
                    PPI = mean(PPI),
                    .groups = 'drop'),
             aes(y = Cambio, color = "Cambio")) +

  ylab("# Valor de Venda (R$) vs (Cambio, PPI, Brent)") +
  xlab("Horizonte Temporal") +
  labs(title = graph_file) +
  theme_bw()
```



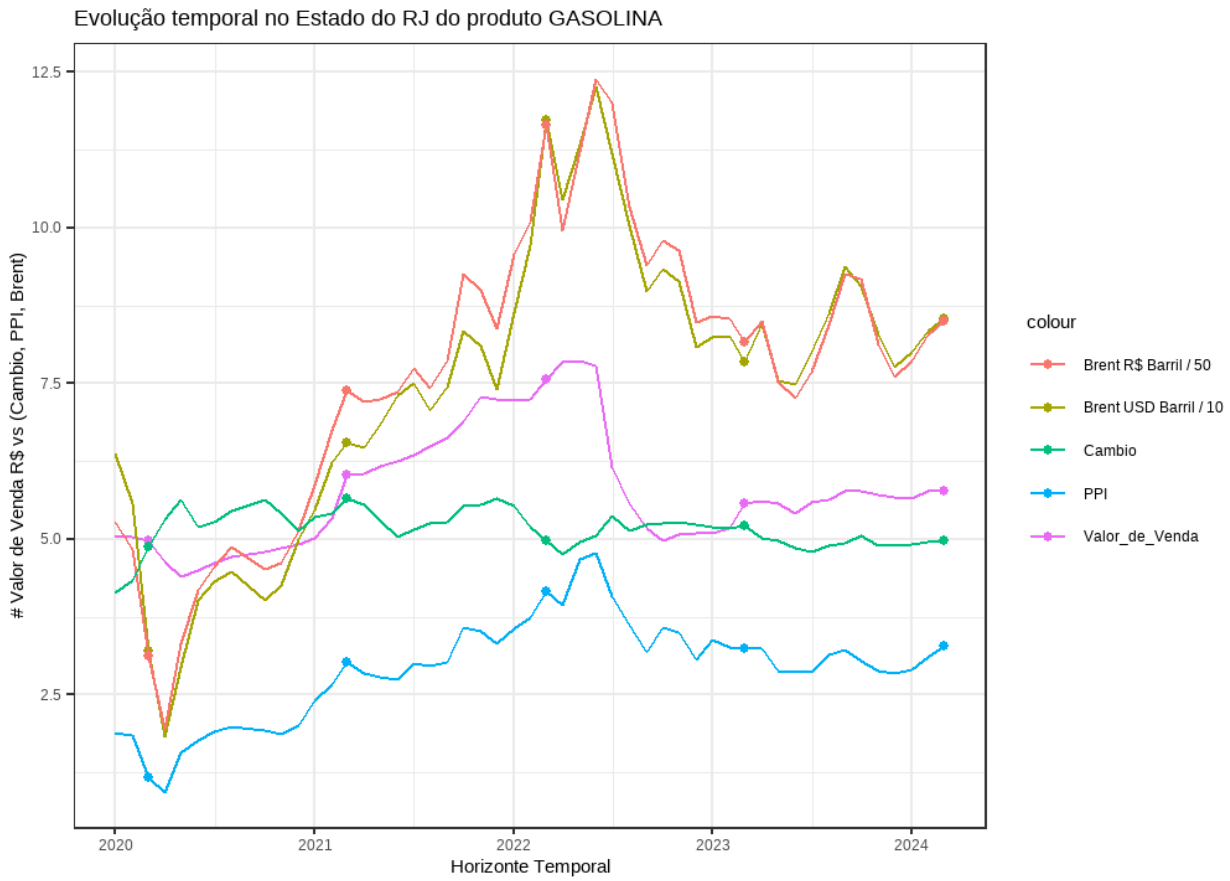

```
ggsave(paste(graph_file, extensao_file, sep = "."), path = "./Images")
```

```
## Saving 7 x 5 in image
```

Analise temporal do combustivel vs Cambio vs Brent vc PPI por um determinado UF.

```
graph_file <- paste("Evolução temporal no Estado do",filter_uf, "do produto",filter_produto, sep = " ")
extensao_file <- "jpg"

sample_combustiveis_agg %>%
   dplyr::filter(Produto == filter_produto & UF == filter_uf) %>%
   dplyr::select(UF, Data, Valor_de_Venda, Cambio, Brent_USD_Barril, Brent_Real_Barril, PPI) %>%
   dplyr::group_by(Data) %>%
   dplyr::summarise(
      Valor_de_Venda = mean(Valor_de_Venda),
      Cambio = mean(Cambio),
      Brent_USD_Barril = mean(Brent_USD_Barril),
      Brent_Real_Barril = mean(Brent_Real_Barril),
      PPI = mean(PPI)
   ) %>%

   ggplot(aes(x = Data)) +
   geom_line(aes(y = Valor_de_Venda, color = "Valor_de_Venda")) +
```

```
  geom_point(data = sample_combustiveis_agg_periodo %>%
               dplyr::filter(UF == filter_uf, Produto == filter_produto),
               aes(y = Valor_de_Venda, color = "Valor_de_Venda")) +

  geom_line(aes(y = Brent_USD_Barril / 10, color = "Brent USD Barril / 10")) +

  geom_point(data = sample_combustiveis_agg_periodo %>%
               dplyr::filter(UF == filter_uf, Produto == filter_produto),
             aes(y = Brent_USD_Barril / 10, color = "Brent USD Barril / 10")) +

  geom_line(aes(y = Brent_Real_Barril / 50, color = "Brent R$ Barril / 50")) +

  geom_point(data = sample_combustiveis_agg_periodo %>%
               dplyr::filter(UF == filter_uf, Produto == filter_produto),
             aes(y = Brent_Real_Barril / 50, color = "Brent R$ Barril / 50")) +

  geom_line(aes(y = PPI, color = "PPI")) +

  geom_point(data = sample_combustiveis_agg_periodo %>%
               dplyr::filter(UF == filter_uf, Produto == filter_produto),
             aes(y = PPI, color = "PPI")) +

  geom_line(aes(y = Cambio, color = "Cambio")) +

  geom_point(data = sample_combustiveis_agg_periodo %>%
               dplyr::filter(UF == filter_uf, Produto == filter_produto),
             aes(y = Cambio, color = "Cambio")) +

  ylab("# Valor de Venda R$ vs (Cambio, PPI, Brent)") +
  xlab("Horizonte Temporal") +
  labs(title = graph_file) +
  theme_bw()
```

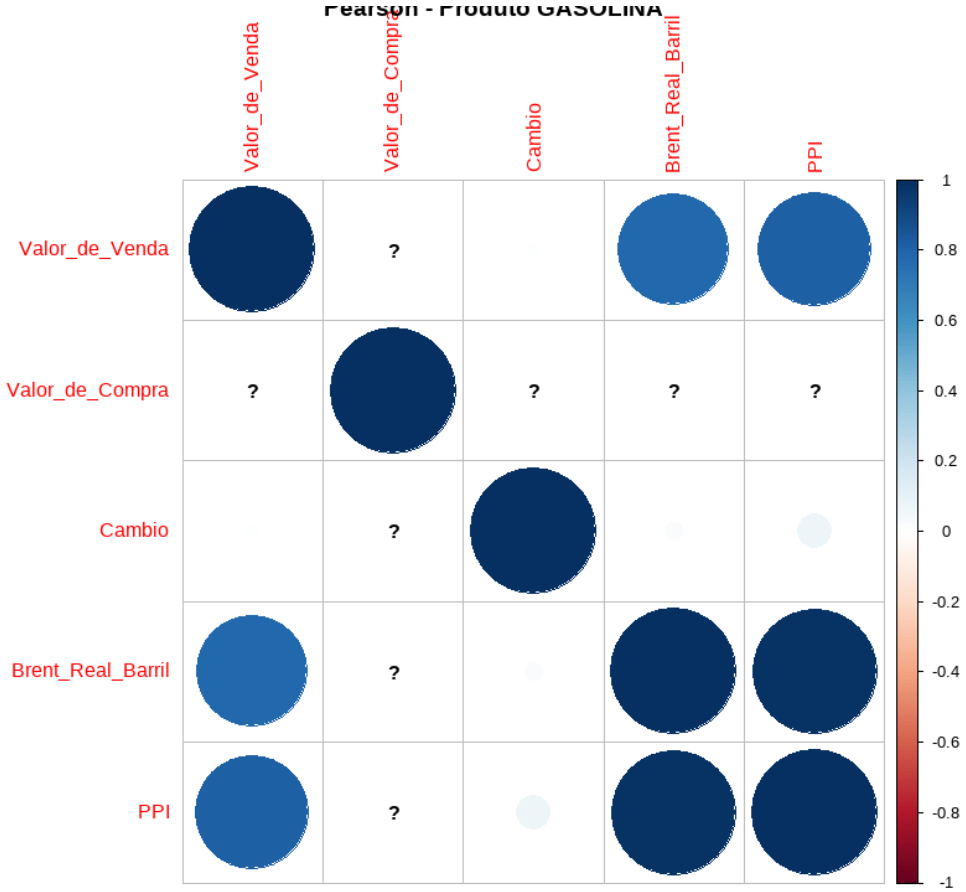



```
ggsave(paste(graph_file, extensao_file, sep = "."), path = "./Images")
```

```
## Saving 7 x 5 in image
```

A partir dos gráficos acima é possível observar alguma relação entre as variáveis Valor de Venda com o Brent e PPI. Esta relação não é significativa em relação a variável Câmbio.

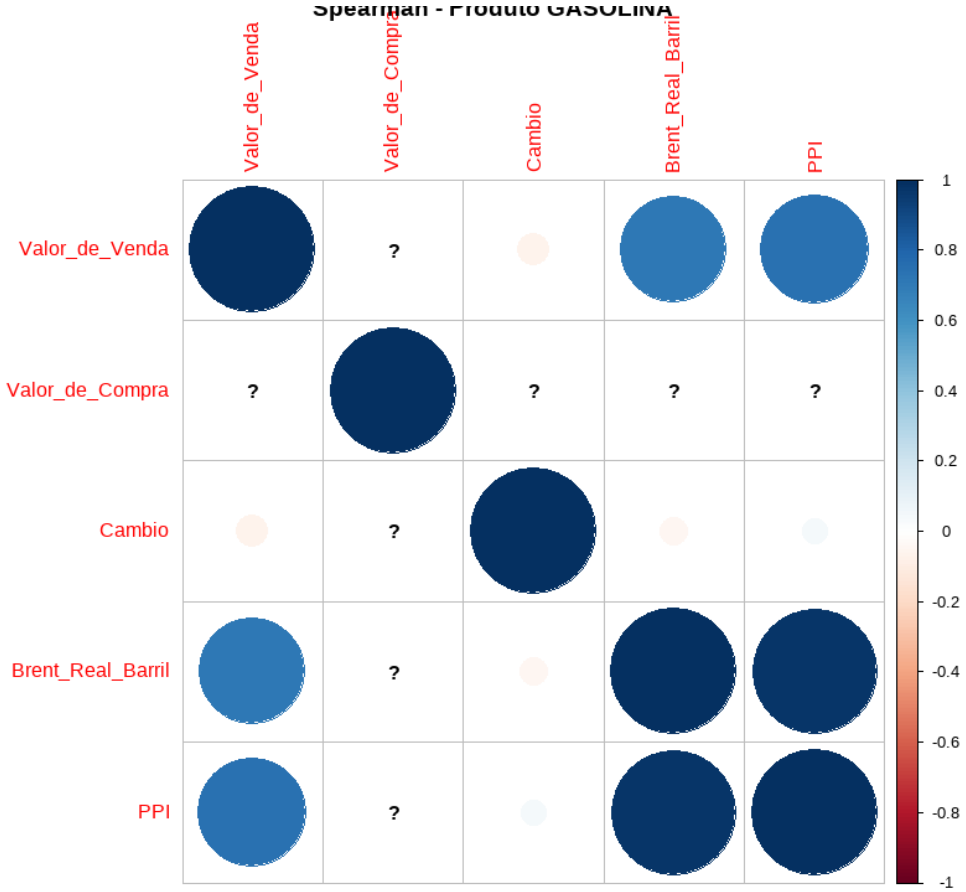
```
graph_file <- paste("Pearson - Produto",filter_produto, sep = " ")
extensao_file <- "jpg"

sample_combustiveis_agg %>%
   dplyr::filter(Produto == filter_produto) %>%
   dplyr::select(Valor_de_Venda, Valor_de_Compra, Cambio, Brent_Real_Barril, PPI) %>%
   cor(., method = "pearson") %>%
   corrplot(., title = graph_file)
```




```
ggsave(paste(graph_file, extensao_file, sep = "."), path = "./Images")
```

```
## Saving 7 x 5 in image
```

```
graph_file <- paste("Spearman - Produto",filter_produto, sep = " ")
extensao_file <- "jpg"

sample_combustiveis_agg %>%
   dplyr::filter(Produto == filter_produto) %>%
   dplyr::select(Valor_de_Venda, Valor_de_Compra, Cambio, Brent_Real_Barril, PPI) %>%
   cor(., method = "spearman") %>%
   corrplot(., title = graph_file)
```




```
ggsave(paste(graph_file, extensao_file, sep = "."), path = "./Images")
```

```
## Saving 7 x 5 in image
```

A partir do corrplot é possível ratificar a relação entre as variáveis Valor de Venda com o Brent e PPI. Esta relação não tão evidente com a variável Câmbio.

É possível observar que as variáveis PPI brent são altamente relacionadas.

Realizando o teste de Hipíotese

Considerando os resultados apresentados anteriormente pelos histogramas, o próximo passo visa realizar o teste da hipótese apresentada no início desta análise.

Hipótese

Há evidência estatística na queda do valor de venda dos combustíveis após o término do período de adoção da política de preço da Petrobras?

Teste de Shapiro Wilk

Será realizado o teste para checar normalidade nas amostras durante e após a adoção da política de preço da Petrobras.

Após Política de Preço

A hipótese nula aqui é: A distribuição do valor de venda por UF segue distribuição normal no mês selecionado para representar após política.

A hipótese alternativa: A distribuição do valor de venda por UF não segue distribuição normal no mês selecionado para representar após política.

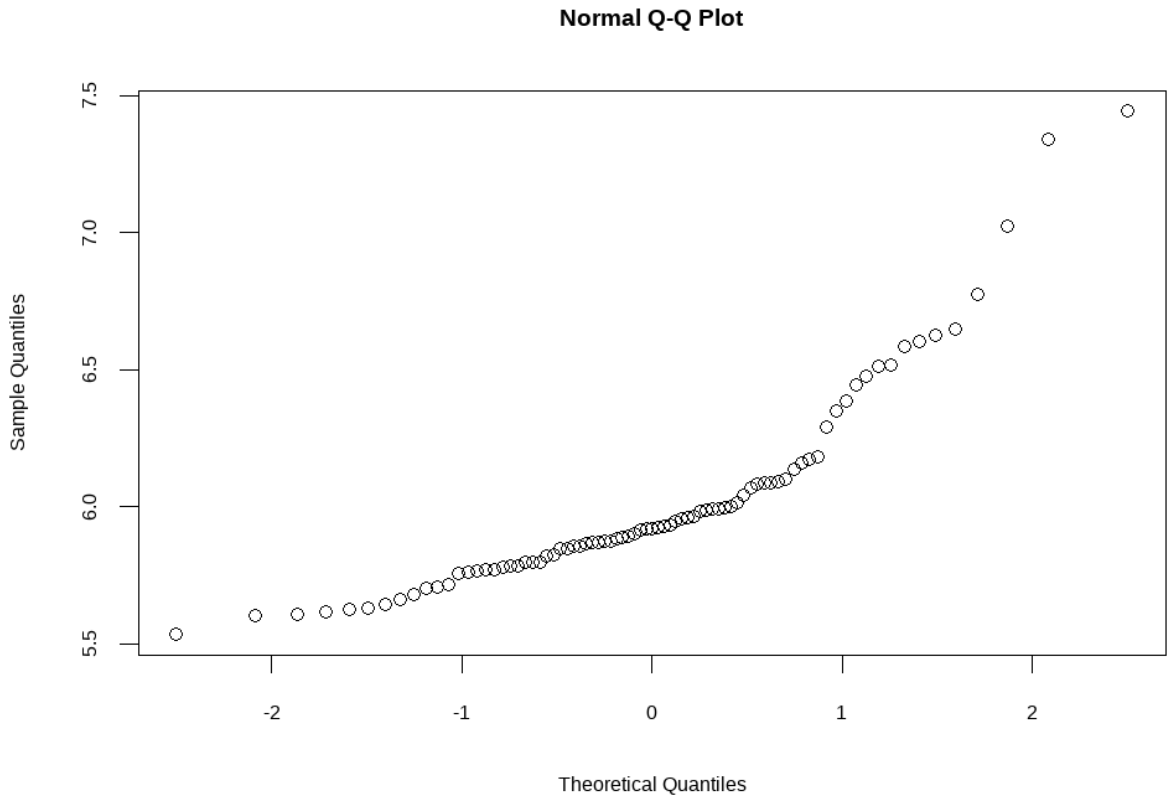
```
apos.politica <- sample_combustiveis_agg_periodo %>%
   dplyr::filter(Data %in% as.IDate("01-03-2024",format = "%d-%m-%Y")) %>%
   dplyr::select(Valor_de_Venda)

shapiro.test(apos.politica$Valor_de_Venda)
```

Aplicando o teste de Shapiro Wilk

```
##
##  Shapiro-Wilk normality test
##
## data:  apos.politica$Valor_de_Venda
## W = 0.83158, p-value = 3.543e-08
```

```
qqnorm(apos.politica$Valor_de_Venda)
```

Aplicando o QQ-Plot



Durante a Política de Preço

A política de preço adotado pela Petrobra foi até 16/05/2023.

A hipótese nula aqui é: A distribuição do valor de venda por UF segue distribuição normal no mês selecionado para representar durante a política.

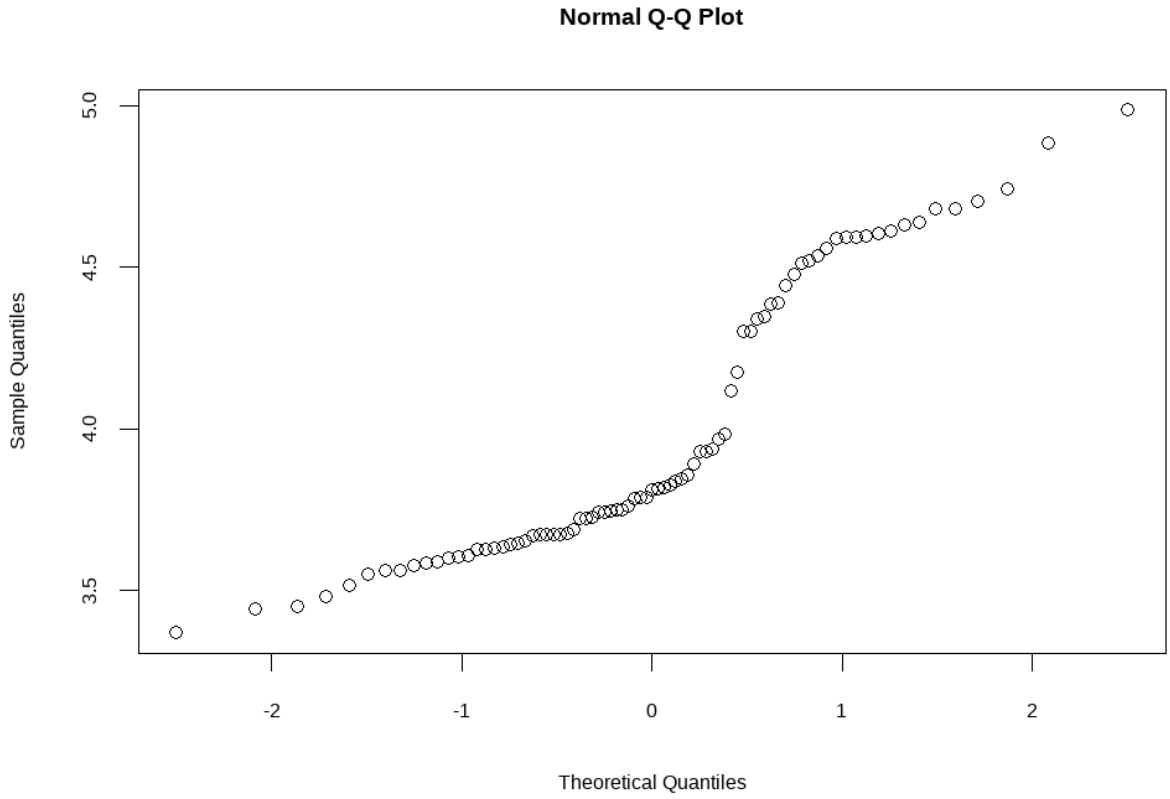
A hipótese alternativa: A distribuição do valor de venda por UF não segue distribuição normal no mês selecionado para representar durante a política.

Primeiro ano da análise - 01/03/2020

Aplicando o teste de Shapiro Wilk

```
##
##  Shapiro-Wilk normality test
##
## data:  durante.politica$Valor_de_Venda
## W = 0.87581, p-value = 1.174e-06
```

Aplicando o QQ-Plot



Teste de Wilcoxon para checar o pareamento Após política vs. Primeiro ano da Pesquisa durante a política - 01/03/2020).

Realizado o teste de Wilcoxon para checar o pareamento entre as distribuições do valor de venda após e durante a política de preço da Petrobras.

A hipótese nula aqui é: A mediana das diferenças (após - durante) entre as distribuições do valor de venda após e durante é igual a zero.

A hipótese alternativa: A mediana das diferenças (após - durante) entre as distribuições do valor de venda após e durante não é igual a zero.

```
##
##  Wilcoxon signed rank test with continuity correction
##
## data:  apos.politica$Valor_de_Venda and durante.politica$Valor_de_Venda
## V = 3321, p-value = 5.464e-15
## alternative hypothesis: true location shift is not equal to 0
```

Teste de Wilcoxon para checar o pareamento (Maior que) Após política vs. Primeiro ano da Pesquisa durante a política - 01/03/2020).

A hipótese nula aqui é: A mediana das diferenças (após - durante) entre as distribuições do valor de venda após e durante é igual a zero.

A hipótese alternativa: A mediana das diferenças (após - durante) entre as distribuições do valor de venda após e durante é maior do que zero.

```
##
##  Wilcoxon signed rank test with continuity correction
##
## data:  apos.politica$Valor_de_Venda and durante.politica$Valor_de_Venda
## V = 3321, p-value = 2.732e-15
## alternative hypothesis: true location shift is greater than 0
```

Teste de Wilcoxon para checar o pareamento (Menor que) Após política vs. Primeiro ano da Pesquisa durante a política - 01/03/2020).

A hipótese nula aqui é: A mediana das diferenças (após - durante) entre as distribuições do valor de venda após e durante é igual a zero.

A hipótese alternativa: A mediana das diferenças (após - durante) entre as distribuições do valor de venda após e durante é menor do que zero.

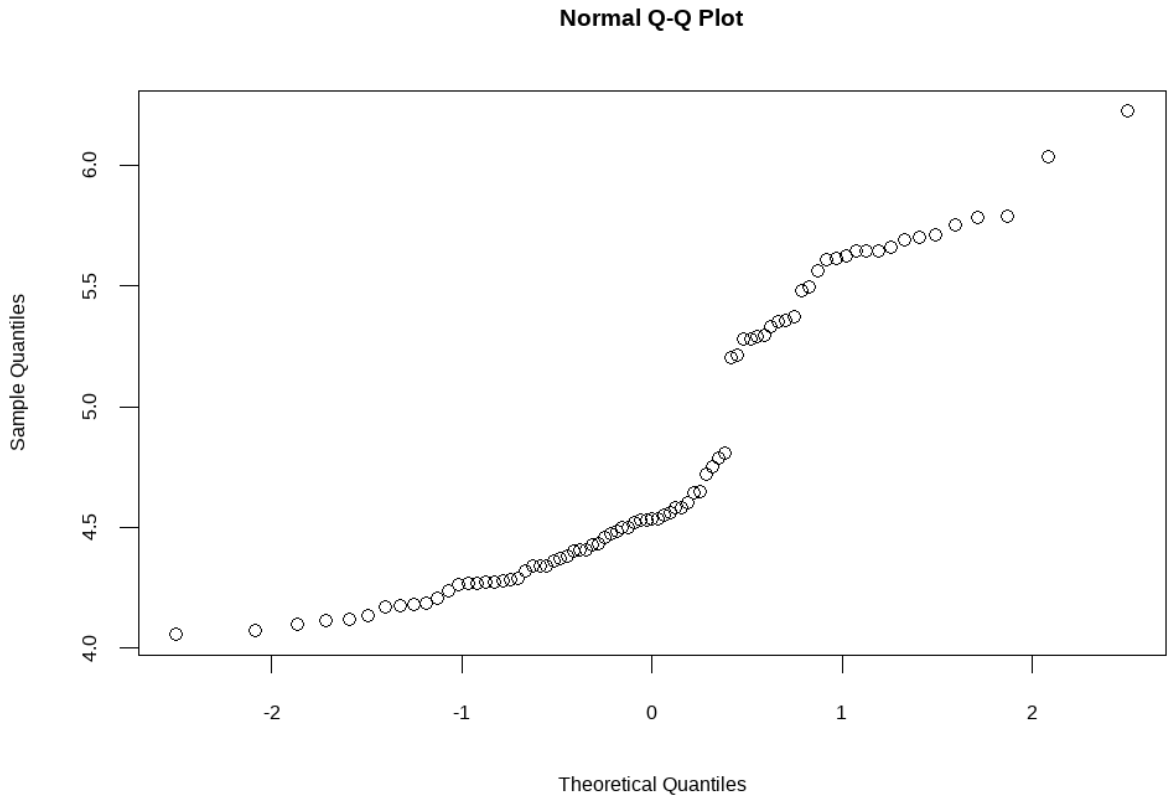
```
##
##  Wilcoxon signed rank test with continuity correction
##
## data:  apos.politica$Valor_de_Venda and durante.politica$Valor_de_Venda
## V = 3321, p-value = 1
## alternative hypothesis: true location shift is less than 0
```

Segundo ano da análise - 01/03/2021

Aplicando o teste de Shapiro Wilk

```
##
##  Shapiro-Wilk normality test
##
## data:  durante.politica$Valor_de_Venda
## W = 0.87044, p-value = 7.404e-07
```

Aplicando o QQ-Plot



Teste de Wilcoxon para checar o pareamento Após política vs. Segundo ano da Pesquisa durante a política - 01/03/2021).

Realizado o teste de Wilcoxon para checar o pareamento entre as distribuições do valor de venda após e durante a política de preço da Petrobras.

A hipótese nula aqui é: A mediana das diferenças (após - durante) entre as distribuições do valor de venda após e durante é igual a zero.

A hipótese alternativa: A mediana das diferenças (após - durante) entre as distribuições do valor de venda após e durante não é igual a zero.

```
##
##  Wilcoxon signed rank test with continuity correction
##
## data:  apos.politica$Valor_de_Venda and durante.politica$Valor_de_Venda
## V = 3296, p-value = 1.382e-14
## alternative hypothesis: true location shift is not equal to 0
```

Teste de Wilcoxon para checar o pareamento (Maior que) Após política vs. Segundo ano da Pesquisa durante a política - 01/03/2021).

A hipótese nula aqui é: A mediana das diferenças (após - durante) entre as distribuições do valor de venda após e durante é igual a zero.

A hipótese alternativa: A mediana das diferenças (após - durante) entre as distribuições do valor de venda após e durante é maior do que zero.

```
##
##  Wilcoxon signed rank test with continuity correction
##
## data:  apos.politica$Valor_de_Venda and durante.politica$Valor_de_Venda
## V = 3296, p-value = 6.908e-15
## alternative hypothesis: true location shift is greater than 0
```

Teste de Wilcoxon para checar o pareamento (Menor que) Após política vs. Segundo ano da Pesquisa durante a política - 01/03/2021).

A hipótese nula aqui é: A mediana das diferenças (após - durante) entre as distribuições do valor de venda após e durante é igual a zero.

A hipótese alternativa: A mediana das diferenças (após - durante) entre as distribuições do valor de venda após e durante é menor do que zero.

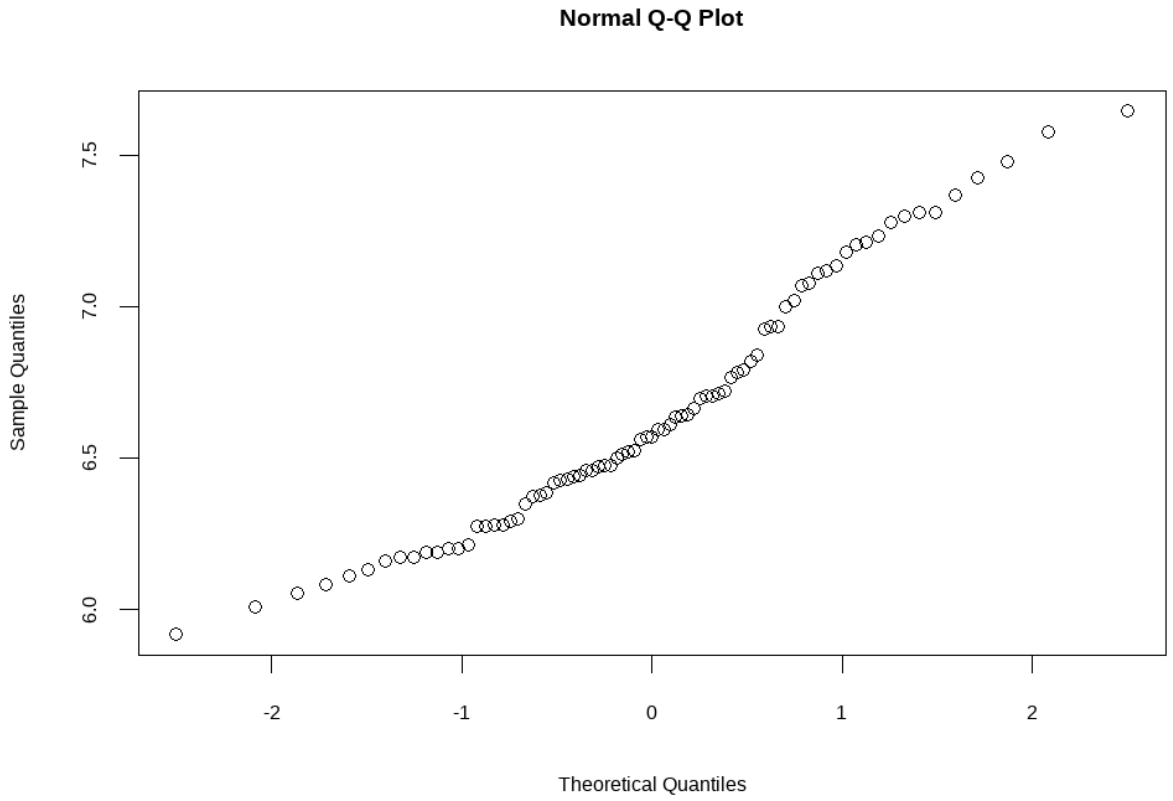
```
##
##  Wilcoxon signed rank test with continuity correction
##
## data:  apos.politica$Valor_de_Venda and durante.politica$Valor_de_Venda
## V = 3296, p-value = 1
## alternative hypothesis: true location shift is less than 0
```

Terceiro ano da análise - 01/03/2022

Aplicando o teste de Shapiro Wilk

```
##
##  Shapiro-Wilk normality test
##
## data:  durante.politica$Valor_de_Venda
## W = 0.95831, p-value = 0.009963
```

Aplicando o QQ-Plot



Teste de Wilcoxon para checar o pareamento Após política vs. Terceiro ano da Pesquisa durante a política - 01/03/2022).

Realizado o teste de Wilcoxon para checar o pareamento entre as distribuições do valor de venda após e durante a política de preço da Petrobras.

A hipótese nula aqui é: A mediana das diferenças (após - durante) entre as distribuições do valor de venda após e durante é igual a zero.

A hipótese alternativa: A mediana das diferenças (após - durante) entre as distribuições do valor de venda após e durante não é igual a zero.

```
##
##  Wilcoxon signed rank test with continuity correction
##
## data:  apos.politica$Valor_de_Venda and durante.politica$Valor_de_Venda
## V = 132, p-value = 6.28e-13
## alternative hypothesis: true location shift is not equal to 0
```

Teste de Wilcoxon para checar o pareamento (Maior que) Após política vs. Terceiro ano da Pesquisa durante a política - 01/03/2022).

A hipótese nula aqui é: A mediana das diferenças (após - durante) entre as distribuições do valor de venda após e durante é igual a zero.

A hipótese alternativa: A mediana das diferenças (após - durante) entre as distribuições do valor de venda após e durante é maior do que zero.

```
##
##  Wilcoxon signed rank test with continuity correction
##
## data:  apos.politica$Valor_de_Venda and durante.politica$Valor_de_Venda
## V = 132, p-value = 1
## alternative hypothesis: true location shift is greater than 0
```

Teste de Wilcoxon para checar o pareamento (Menor que) Após política vs. Terceiro ano da Pesquisa durante a política - 01/03/2022).

A hipótese nula aqui é: A mediana das diferenças (após - durante) entre as distribuições do valor de venda após e durante é igual a zero.

A hipótese alternativa: A mediana das diferenças (após - durante) entre as distribuições do valor de venda após e durante é menor do que zero.

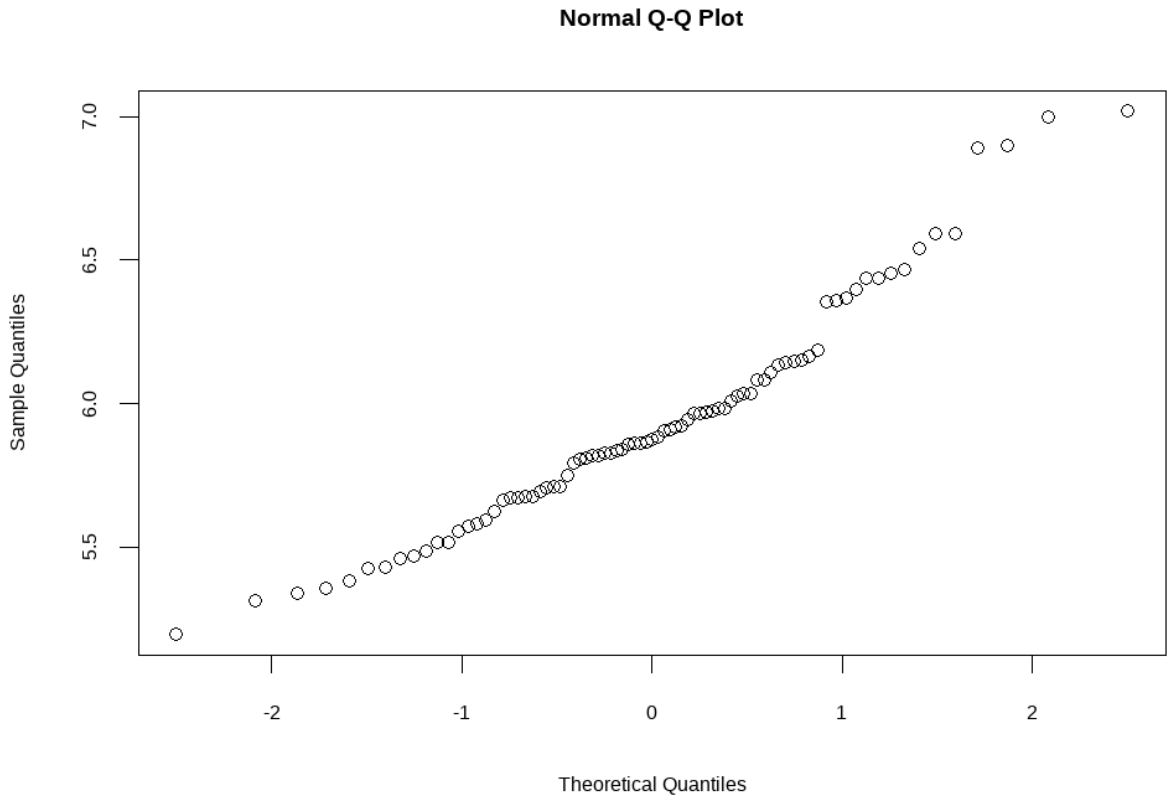
```
##
##  Wilcoxon signed rank test with continuity correction
##
## data:  apos.politica$Valor_de_Venda and durante.politica$Valor_de_Venda
## V = 132, p-value = 3.14e-13
## alternative hypothesis: true location shift is less than 0
```

Quarto ano da análise - 01/03/2023

Aplicando o teste de Shapiro Wilk

```
##
##  Shapiro-Wilk normality test
##
## data:  durante.politica$Valor_de_Venda
## W = 0.95493, p-value = 0.006233
```

Aplicando o QQ-Plot



Teste de Wilcoxon para checar o pareamento Após política vs. Quarto ano da Pesquisa durante a política - 01/03/2023).

Realizado o teste de Wilcoxon para checar o pareamento entre as distribuições do valor de venda após e durante a política de preço da Petrobras.

A hipótese nula aqui é: A mediana das diferenças (após - durante) entre as distribuições do valor de venda após e durante é igual a zero.

A hipótese alternativa: A mediana das diferenças (após - durante) entre as distribuições do valor de venda após e durante não é igual a zero.

```
##
##  Wilcoxon signed rank test with continuity correction
##
## data:  apos.politica$Valor_de_Venda and durante.politica$Valor_de_Venda
## V = 2270, p-value = 0.004139
## alternative hypothesis: true location shift is not equal to 0
```

Teste de Wilcoxon para checar o pareamento (Maior que) Após política vs. Quarto ano da Pesquisa durante a política - 01/03/2023).

A hipótese nula aqui é: A mediana das diferenças (após - durante) entre as distribuições do valor de venda após e durante é igual a zero.

A hipótese alternativa: A mediana das diferenças (após - durante) entre as distribuições do valor de venda após e durante é maior do que zero.

```
##
##  Wilcoxon signed rank test with continuity correction
##
## data:  apos.politica$Valor_de_Venda and durante.politica$Valor_de_Venda
## V = 2270, p-value = 0.00207
## alternative hypothesis: true location shift is greater than 0
```

Teste de Wilcoxon para checar o pareamento (Menor que) Após política vs. Quarto ano da Pesquisa durante a política - 01/03/2023).

A hipótese nula aqui é: A mediana das diferenças (após - durante) entre as distribuições do valor de venda após e durante é igual a zero.

A hipótese alternativa: A mediana das diferenças (após - durante) entre as distribuições do valor de venda após e durante é menor do que zero.

```
##
##  Wilcoxon signed rank test with continuity correction
##
## data:  apos.politica$Valor_de_Venda and durante.politica$Valor_de_Venda
## V = 2270, p-value = 0.998
## alternative hypothesis: true location shift is less than 0
```

Manipulando base de dados com dados faltantes e outliers

O que é completude de dados?

Completude de dados se refere a ausência de dados em um conjunto de dados. Quando os dados estão completos em um conjunto de dados e sua consistência pode ser validada, dizemos que há qualidade dos dados a serem utilizados em uma análise, possibilitando assim *insights* confiáveis.

Qual o impacto que os dados faltantes podem ter em uma análise?

A ausência de dados ou lacunas no conjunto de dados em uma análise, além de proporcionar uma baixa qualidade dos dados, poderá impossibilitar a obtenção de *insights* confiáveis e precisos. A inconfiabilidade nos dados podem gerar interpretações equivocadas e propor decisões errôneas.

Índice de completude

```
sample_combustiveis_agg %>% dlookr::diagnose()
```

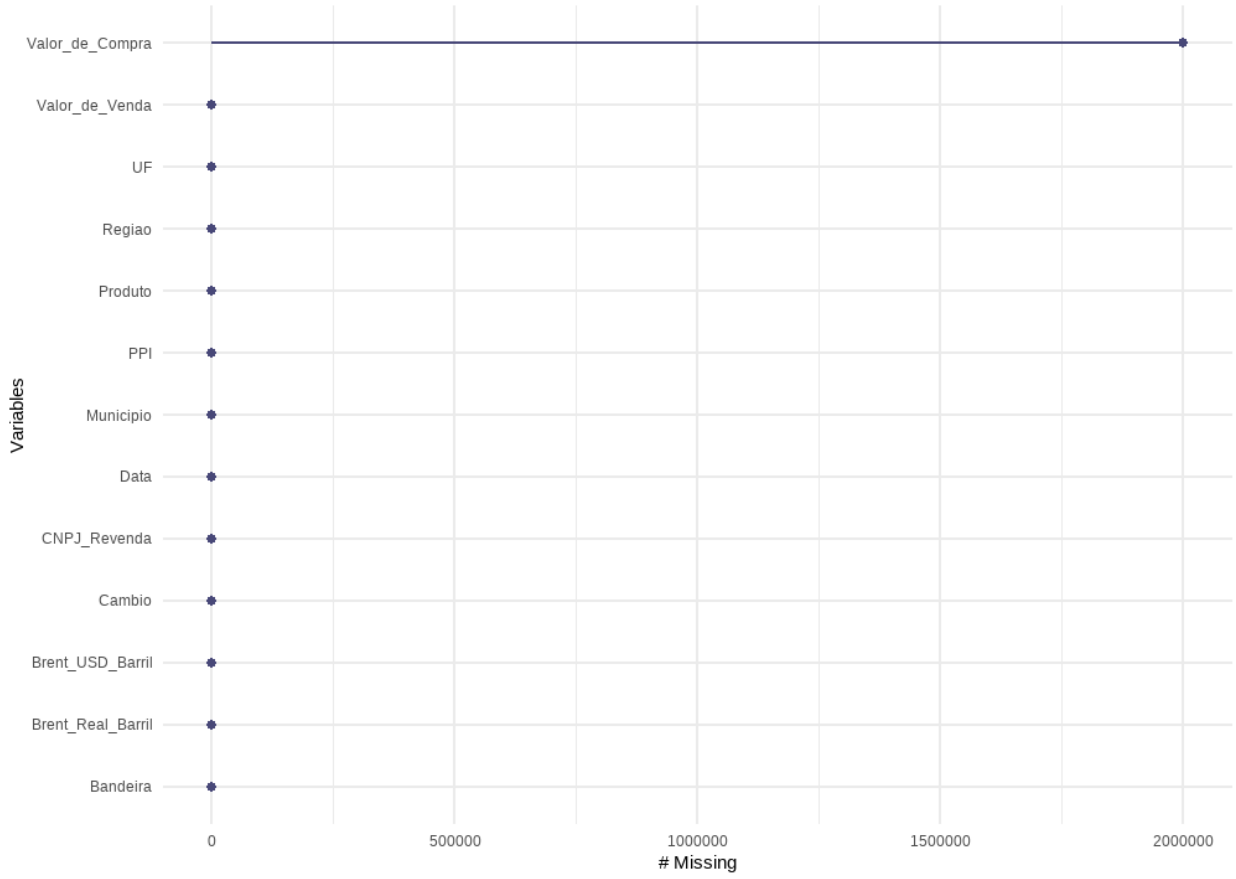
```
## # A tibble: 13 x 6
##    variables        types missing_count missing_percent unique_count unique_rate
##    <chr>            <chr>         <int>           <dbl>        <int>       <dbl>
##  1 Data             Date              0               0           50  0.0000236
##  2 Regiao           char~             0               0            5  0.00000236
##  3 UF               char~             0               0           27  0.0000128
##  4 Municipio        char~             0               0          469  0.000222
##  5 CNPJ_Revenda     char~             0               0        19927  0.00942
##  6 Produto          char~             0               0            3  0.00000142
##  7 Bandeira         char~             0               0           79  0.0000374
##  8 Valor_de_Venda   nume~             0               0         4203  0.00199
```

```
##  9 Valor_de_Compra  nume~       2002006            94.7        18046  0.00854
## 10 PPI              nume~             0             0            100  0.0000473
## 11 Brent_USD_Barril nume~             0             0             50  0.0000236
## 12 Cambio           nume~             0             0             50  0.0000236
## 13 Brent_Real_Barr~ nume~             0             0             50  0.0000236
```

```
sample_combustiveis_agg %>% naniar::miss_var_summary()
```

```
## # A tibble: 13 x 3
##    variable           n_miss pct_miss
##    <chr>               <int>    <num>
##  1 Valor_de_Compra   2002006     94.7
##  2 Data                    0      0
##  3 Regiao                  0      0
##  4 UF                      0      0
##  5 Municipio               0      0
##  6 CNPJ_Revenda            0      0
##  7 Produto                 0      0
##  8 Bandeira                0      0
##  9 Valor_de_Venda          0      0
## 10 PPI                     0      0
## 11 Brent_USD_Barril        0      0
## 12 Cambio                  0      0
## 13 Brent_Real_Barril       0      0
```

```
sample_combustiveis_agg %>% gg_miss_var()
```




```
#sample_combustiveis_agg %>% gg_miss_upset()
```

A única variável de interesse que possui missing é a Valor de Compra, intitulada Valor_de_Compra.

Base de Dados Combustíveis - Visão de missing nacional

Para uma melhor compreensão dos missing existentes nesta variável, algumas investigações foram feitas.

A primeira análise visa visualizar o missing da variável por Regiao.

```
df <- sample_combustiveis_agg %>%
  dplyr::group_by(Regiao) %>%
  dplyr::mutate(n_linhas = 1) %>%
  dplyr::summarise(n_missing = sum(is.na(Valor_de_Compra))) %>%
  dplyr::arrange(desc(n_missing))

#knitr::kable(df, format="latex")
knitr::kable(df,
             format="latex",
             caption = "Quantidade de missing da variável Valor de Compra por Região.",
             align = "c",
             booktabs = TRUE,
             longtable = TRUE,
```

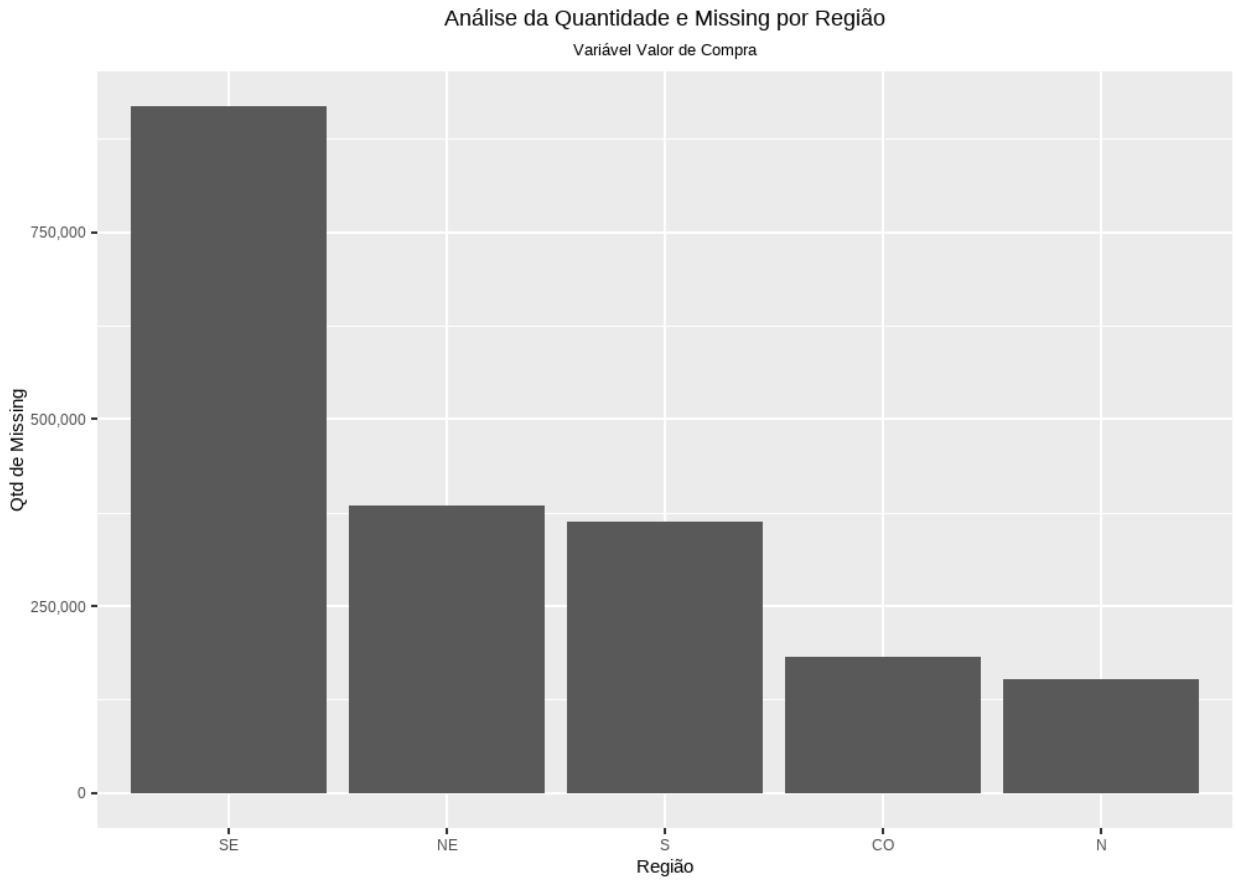
```
            linesep = ""
)
```

Table 2: Quantidade de missing da variável Valor de Compra por Região.

Regiao	n_missing
SE	918714
NE	385190
S	363446
CO	181867
N	152789

É possível observar que a região sudeste possui maior quantidade de missing. O gráfico abaixo demonstra esta análise visualmente.

```
ggplot(df, aes(x = reorder(Regiao,
                           n_missing, decreasing = TRUE),
               y = n_missing)) +
   geom_bar(stat = "identity") +
   labs(x = "Região",
        y = "Qtd de Missing",
        subtitle = "Variável Valor de Compra") +
   scale_y_continuous(labels = scales::comma_format()) +
   ggtitle("Análise da Quantidade e Missing por Região") +
   theme(plot.title = element_text(hjust = 0.5),
         plot.subtitle = element_text(hjust = 0.5,
                                      size = 10)) +
   scale_fill_brewer(palette = "Set1")
```




```
rm("df")
```

Foi feita uma segunda análise considerando a porcentagem de missing sobre a observação desta variável para cada região. É possível observar uma uniformização nas porcentagens entre as regiões.

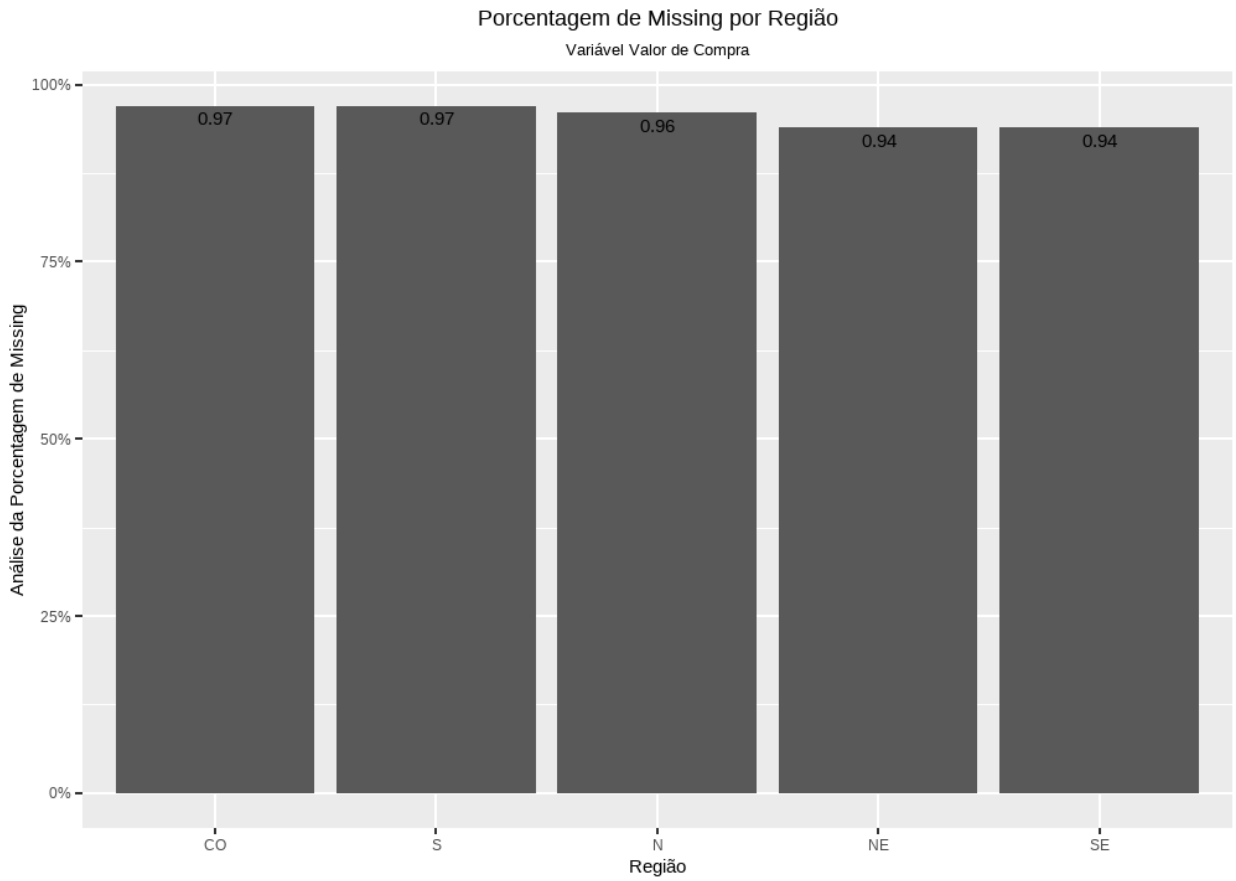
```
df <- sample_combustiveis_agg %>%
  dplyr::group_by(Regiao) %>%
  dplyr::mutate(n_linhas = 1) %>%
  dplyr::summarise(n_missing = sum(is.na(Valor_de_Compra)),
            n_linhas = sum(n_linhas),
            porcentagem_missing = round(n_missing / n_linhas, 2)) %>%
  dplyr::arrange(desc(porcentagem_missing))

#kable(df, format="latex")
knitr::kable(df,
             format="latex",
             caption = "Porcentagem de missing da variável Valor de Compra por Região.",
             align = "c",
             booktabs = TRUE,
             longtable = TRUE,
             linesep = "",)
```

Table 3: Porcentagem de missing da variável Valor de Compra por Região.

Regiao	n_missing	n_linhas	porcentagem_missing
CO	181867	186821	0.97
S	363446	376159	0.97
N	152789	159832	0.96
NE	385190	409249	0.94
SE	918714	982258	0.94

O gráfico abaixo demonstra visualmente esta análise.

```
ggplot(df, aes(x = reorder(Regiao, porcentagem_missing, decreasing = TRUE),
               y = porcentagem_missing)) +
   geom_bar(stat = "identity") +
   labs(x = "Região",
        y = "Análise da Porcentagem de Missing",
        subtitle = "Variável Valor de Compra") +
   scale_y_continuous(labels = scales::percent_format()) +
   ggtitle("Porcentagem de Missing por Região") +
   theme(plot.title = element_text(hjust = 0.5),
         plot.subtitle = element_text(hjust = 0.5,
                                      size = 10)) +
   geom_text(aes(label = porcentagem_missing),
             vjust = 1.5)
```




```
rm("df")
```

A mesma análise realizada por estado, foi feita por Estado.

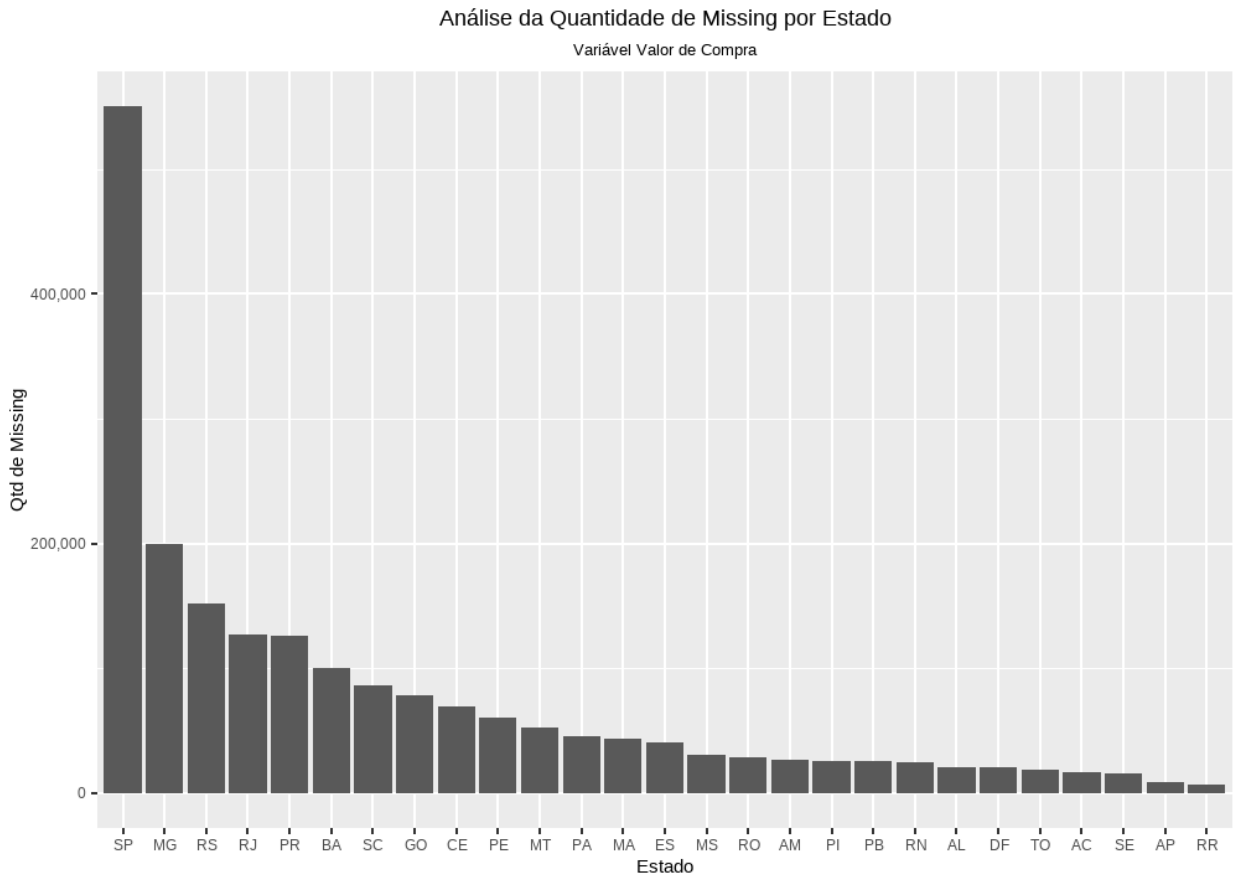
```
df <- sample_combustiveis_agg %>%
   dplyr::group_by(UF) %>%
   dplyr::mutate(n_linhas = 1) %>%
   dplyr::summarise(n_missing = sum(is.na(Valor_de_Compra))) %>%
   dplyr::arrange(desc(n_missing))

#kable(df, format="latex")

knitr::kable(df,
             format="latex",
             caption = "Quantidade de missing da variável Valor de Compra por Estado.",
             align = "c",
             booktabs = TRUE,
             longtable = TRUE,
             linesep = "",)
```

Table 4: Quantidade de missing da variável Valor de Compra por Estado.

UF	n_missing
SP	550530
MG	199947
RS	151457
RJ	127299
PR	125571
BA	100502
SC	86418
GO	77924
CE	69070
PE	60115
MT	52767
PA	45711
MA	43078
ES	40938
MS	30882
RO	29056
AM	26649
PI	26053
PB	25428
RN	24380
AL	20814
DF	20294
TO	18958
AC	17018
SE	15750
AP	8415
RR	6982

É possível observar que o missing da respectiva variável é expressivo para os postos de combustíveis de SP, MG e RJ. O gráfico abaixo demonstra esta representatividade.

```
ggplot(df, aes(x = reorder(UF, n_missing, decreasing = TRUE), y = n_missing)) +
   geom_bar(stat = "identity") +
   labs(x = "Estado", y = "Qtd de Missing", subtitle = "Variável Valor de Compra") +
   scale_y_continuous(labels = scales::comma_format()) +
   ggtitle("Análise da Quantidade de Missing por Estado") +
   theme(plot.title = element_text(hjust = 0.5),
         plot.subtitle = element_text(hjust = 0.5, size = 10)) +
   scale_fill_brewer(palette = "Set1")
```




```
rm("df")
```

Foi feita uma segunda análise considerando a porcentagem de missing sobre a observação desta variável para cada estado. É possível observar valores próximos.

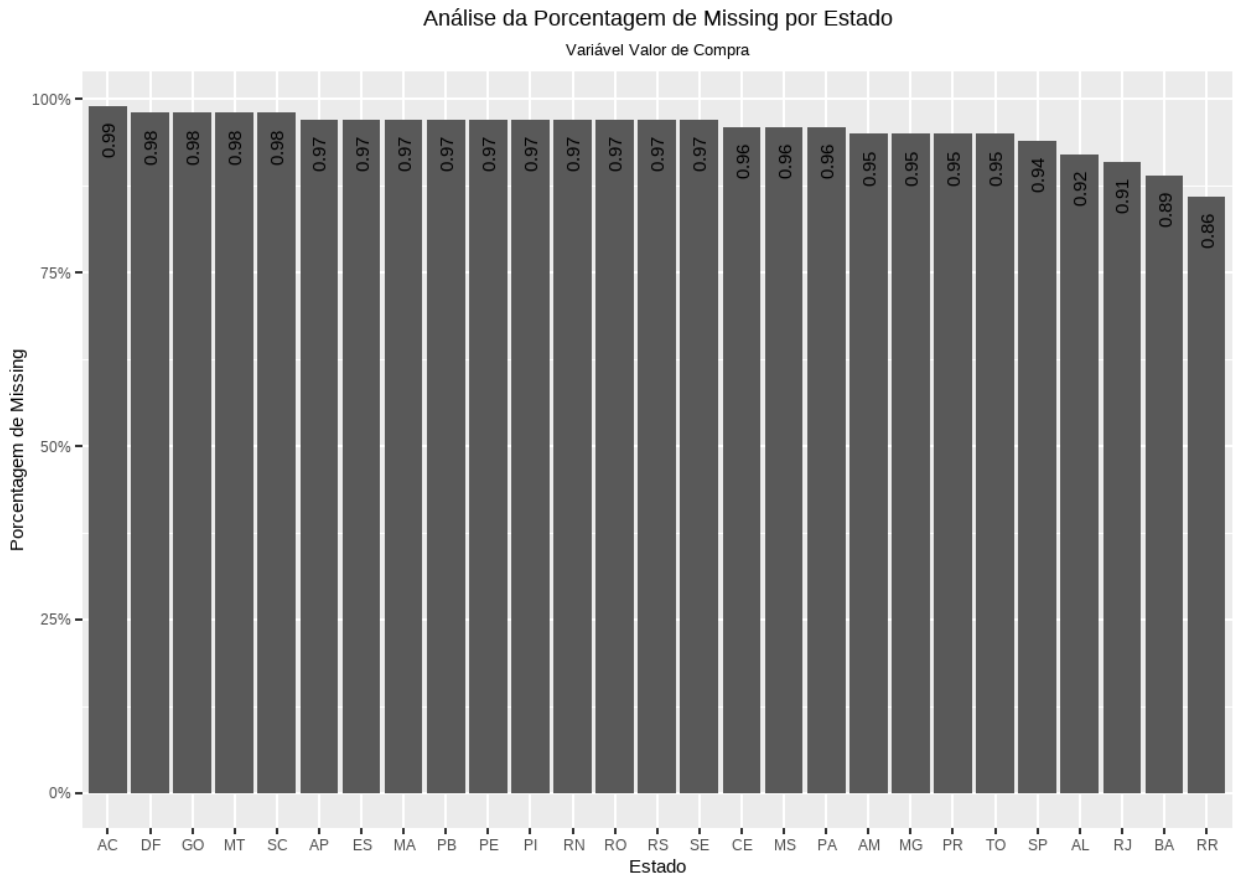
```
df <- sample_combustiveis_agg %>%
  dplyr::group_by(UF) %>%
  dplyr::mutate(n_linhas = 1) %>%
  dplyr::summarise(n_missing = sum(is.na(Valor_de_Compra)),
            n_linhas = sum(n_linhas),
            porcentagem_missing = round(n_missing / n_linhas, 2)) %>%
  dplyr::arrange(desc(porcentagem_missing))

#kable(df, format="latex")
knitr::kable(df,
             format="latex",
             caption = "Porcentagem de missing da variável Valor de Compra por Estado.",
             align = "c",
             booktabs = TRUE,
             longtable = TRUE,
             linesep = "",)
```

Table 5: Porcentagem de missing da variável Valor de Compra por Estado.

UF	n_missing	n_linhas	porcentagem_missing
AC	17018	17180	0.99
DF	20294	20798	0.98
GO	77924	79803	0.98
MT	52767	54008	0.98
SC	86418	88427	0.98
AP	8415	8680	0.97
ES	40938	42338	0.97
MA	43078	44204	0.97
PB	25428	26229	0.97
PE	60115	62119	0.97
PI	26053	26838	0.97
RN	24380	25237	0.97
RO	29056	30032	0.97
RS	151457	155573	0.97
SE	15750	16314	0.97
CE	69070	72139	0.96
MS	30882	32212	0.96
PA	45711	47845	0.96
AM	26649	28086	0.95
MG	199947	210846	0.95
PR	125571	132159	0.95
TO	18958	19882	0.95
SP	550530	588664	0.94
AL	20814	22704	0.92
RJ	127299	140410	0.91
BA	100502	113465	0.89
RR	6982	8127	0.86

O gráfico abaixo demonstra visualmente estes valores.

```
ggplot(df, aes(x = reorder(UF,
                           porcentagem_missing, decreasing = TRUE),
               y = porcentagem_missing)) +
  geom_bar(stat = "identity") +
  labs(x = "Estado",
       y = "Porcentagem de Missing",
       subtitle = "Variável Valor de Compra") +
  scale_y_continuous(labels = scales::percent_format()) +
  ggtitle("Análise da Porcentagem de Missing por Estado") +
  theme(plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5, size = 10)) +
  geom_text(aes(label = porcentagem_missing),
            angle = 90,
            vjust = 0.5,
            hjust = 1.5)
```




```
rm("df")
```

Base de Dados Combustíveis - Visão de missing estado RJ

A próxima análise visa compreender a quantidade de missing da variável Valor de Compra nos municípios do Estado do RJ.

```
df <- sample_combustiveis_agg %>% dplyr::filter(UF == "RJ") %>%
  dplyr::group_by(Municipio) %>%
  dplyr::mutate(n_linhas = 1) %>%
  dplyr::summarise(n_missing = sum(is.na(Valor_de_Compra))) %>%
  dplyr::arrange(desc(n_missing))


#kable(df, format="latex")
knitr::kable(df,
             format="latex",
             caption = "Quantidade de missing da variável Valor no Estado do RJ.",
             align = "c",
             booktabs = TRUE,
             longtable = TRUE,
             linesep = "",)
```

Table 6: Quantidade de missing da variável Valor no Estado do RJ.

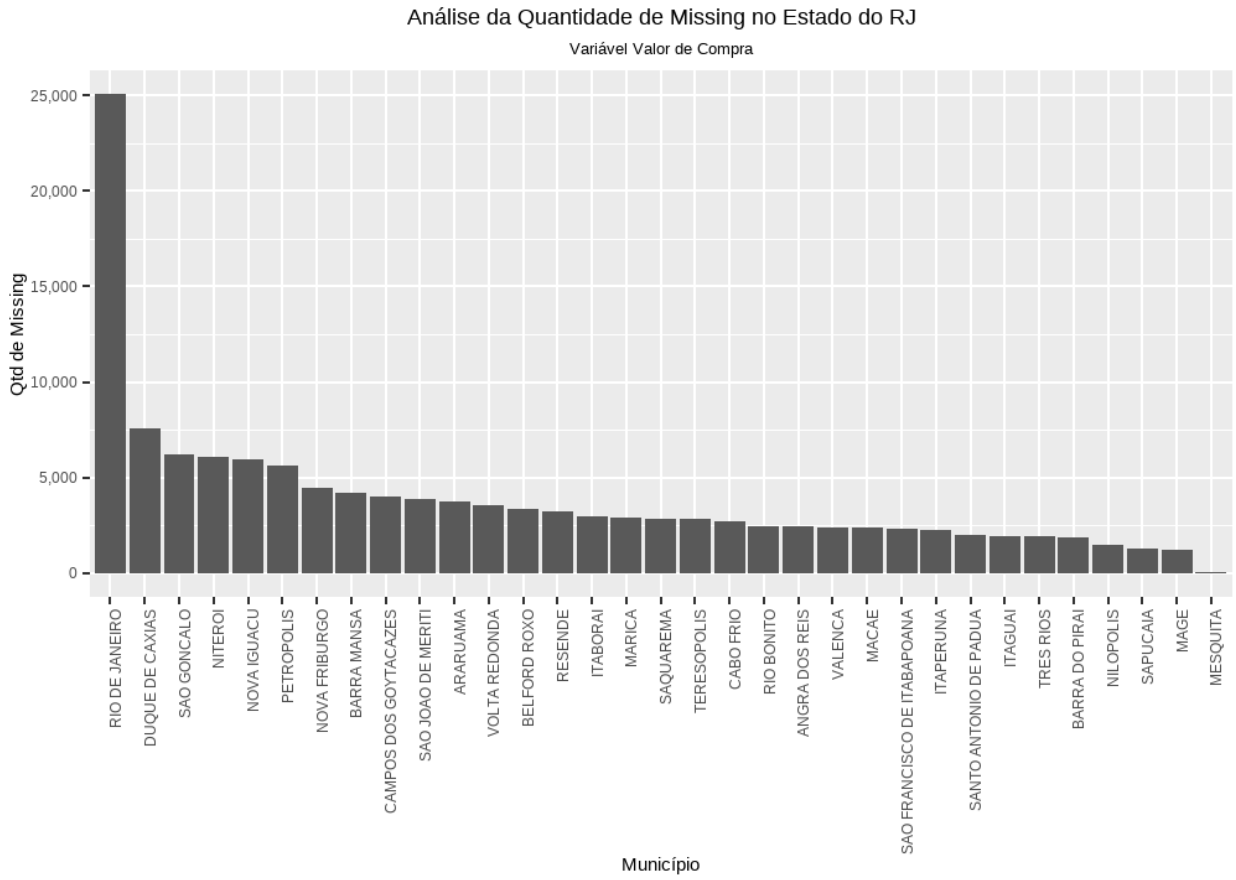
Municipio	n_missing
RIO DE JANEIRO	25056
DUQUE DE CAXIAS	7556
SAO GONCALO	6197
NITEROI	6053
NOVA IGUACU	5938
PETROPOLIS	5623
NOVA FRIBURGO	4470
BARRA MANSA	4227
CAMPOS DOS GOYTACAZES	3990
SAO JOAO DE MERITI	3848
ARARUAMA	3766
VOLTA REDONDA	3542
BELFORD ROXO	3339
RESENDE	3255
ITABORAI	2997
MARICA	2912
SAQUAREMA	2868
TERESOPOLIS	2821
CABO FRIO	2706
RIO BONITO	2472
ANGRA DOS REIS	2450
VALENCA	2399
MACAE	2365
SAO FRANCISCO DE ITABAPOANA	2334
ITAPERUNA	2257
SANTO ANTONIO DE PADUA	2026
ITAGUAI	1950
TRES RIOS	1937
BARRA DO PIRAI	1872
NILOPOLIS	1489
SAPUCAIA	1309
MAGE	1227
MESQUITA	48

É possível observar os cinco municípios com maior quantidade de missing encontram-se na região metropolitana do Rio de Janeiro.

O gráfico a seguir desmonstra esta analise visualmente.

```
ggplot(df, aes(x = reorder(Municipio, n_missing, decreasing = TRUE),
               y = n_missing)) +
   geom_bar(stat = "identity") +
   labs(x = "Município",
        y = "Qtd de Missing",
        subtitle = "Variável Valor de Compra") +
   scale_y_continuous(labels = scales::comma_format()) +
   ggtitle("Análise da Quantidade de Missing no Estado do RJ") +
   theme(plot.title = element_text(hjust = 0.5),
         plot.subtitle = element_text(hjust = 0.5,
```

```
                                        size = 10),
        axis.text.x = element_text(angle = 90,
                                   hjust = 1)
  ) +
  scale_fill_brewer(palette = "Set1")
```




```
rm("df")
```

Foi feita uma segunda análise considerando a porcentagem de missing sobre a observação desta variável para cada Município do Rio de Janeiro.

```
df <- sample_combustiveis_agg %>% dplyr::filter(UF == "RJ") %>%
  dplyr::group_by(Municipio) %>%
  dplyr::mutate(n_linhas = 1) %>%
  dplyr::summarise(n_missing = sum(is.na(Valor_de_Compra)),
            n_linhas = sum(n_linhas),
            porcentagem_missing = round(n_missing / n_linhas, 2)) %>%
  dplyr::arrange(desc(porcentagem_missing))

df
```

```
## # A tibble: 33 x 4
```

```
##    Municipio                  n_missing n_linhas porcentagem_missing
##    <chr>                          <int>    <dbl>               <dbl>
##  1 MESQUITA                          48       48                1
##  2 SAPUCAIA                        1309     1325                0.99
##  3 MACAE                           2365     2423                0.98
##  4 ANGRA DOS REIS                  2450     2514                0.97
##  5 NOVA FRIBURGO                   4470     4625                0.97
##  6 PETROPOLIS                      5623     5788                0.97
##  7 RESENDE                         3255     3371                0.97
##  8 SAO FRANCISCO DE ITABAPOANA     2334     2405                0.97
##  9 ARARUAMA                        3766     3913                0.96
## 10 BARRA MANSA                     4227     4424                0.96
## # i 23 more rows
```

```
#kable(df, format="latex")
knitr::kable(df,
             format="latex",
             caption = "Porcentagem de missing da variável
             Valor de Compra por Município RJ.",
             align = "c",
             booktabs = TRUE,
             longtable = TRUE,
             linesep = "",)
```

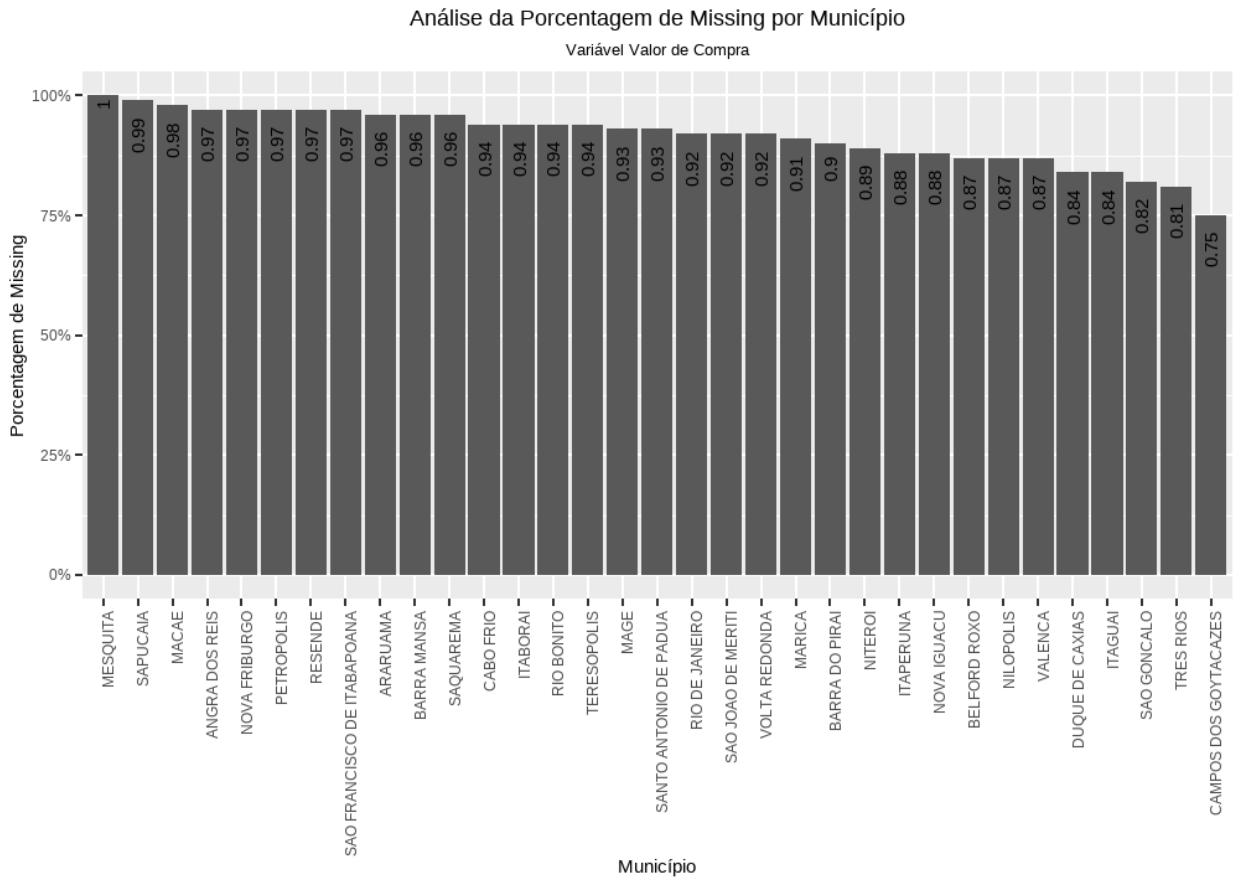
Table 7: Porcentagem de missing da variável Valor de Compra por Município RJ.

Municipio	n_missing	n_linhas	porcentagem_missing
MESQUITA	48	48	1.00
SAPUCAIA	1309	1325	0.99
MACAE	2365	2423	0.98
ANGRA DOS REIS	2450	2514	0.97
NOVA FRIBURGO	4470	4625	0.97
PETROPOLIS	5623	5788	0.97
RESENDE	3255	3371	0.97
SAO FRANCISCO DE ITABAPOANA	2334	2405	0.97
ARARUAMA	3766	3913	0.96
BARRA MANSA	4227	4424	0.96
SAQUAREMA	2868	2998	0.96
CABO FRIO	2706	2876	0.94
ITABORAI	2997	3187	0.94
RIO BONITO	2472	2628	0.94
TERESOPOLIS	2821	2993	0.94
MAGE	1227	1320	0.93
SANTO ANTONIO DE PADUA	2026	2190	0.93
RIO DE JANEIRO	25056	27203	0.92
SAO JOAO DE MERITI	3848	4162	0.92
VOLTA REDONDA	3542	3850	0.92
MARICA	2912	3186	0.91
BARRA DO PIRAI	1872	2072	0.90
NITEROI	6053	6800	0.89
ITAPERUNA	2257	2559	0.88

NOVA IGUACU	5938	6727	0.88
BELFORD ROXO	3339	3817	0.87
NILOPOLIS	1489	1707	0.87
VALENCA	2399	2761	0.87
DUQUE DE CAXIAS	7556	8955	0.84
ITAGUAI	1950	2332	0.84
SAO GONCALO	6197	7512	0.82
TRES RIOS	1937	2403	0.81
CAMPOS DOS GOYTACAZES	3990	5336	0.75

O gráfico abaixo demonstra visualmente esta análise.

```
ggplot(df, aes(x = reorder(Municipio,
                           porcentagem_missing,
                           decreasing = TRUE),
               y = porcentagem_missing)) +
  geom_bar(stat = "identity") +
  labs(x = "Município",
       y = "Porcentagem de Missing",
       subtitle = "Variável Valor de Compra") +
  scale_y_continuous(labels = scales::percent_format()) +
  ggtitle("Análise da Porcentagem de Missing por Município") +
  theme(plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5, size = 10),
        axis.text.x = element_text(angle = 90, hjust = 1)) +
  geom_text(aes(label = porcentagem_missing), angle = 90, vjust = 0.5, hjust = 1.5)
```




```
rm("df")
```

É possível observar nesta análise que o município de Mesquita que estava em último na análise de quantidade de missing agora é o primeiro. Para compreender melhor, é necessário entender a frequência relativa deste município na pesquisa. O que se pode deduzir neste primeiro momento é que 100% dos postos pesquisados não informaram o valor da variável alvo desta análise. Tal afirmação é observada pela quantidade de valores em n_linhas igual a n_missing, onde n_linhas representa a quatidade de postos pesquisados.

É possível observar que somente Mesquita, dentre os cinco primeiros municípios, faz parte dos municípios da região metropolitana do Rio de Janeiro.

Realizando teste de Little para checar se os dados faltantes são completamente aleatórios

```
sample_combustiveis_agg %>% dplyr::select(Valor_de_Compra) %>% naniar::mcar_test()
```

```
## # A tibble: 1 x 4
##   statistic    df p.value missing.patterns
##       <dbl> <dbl>   <dbl>            <int>
## 1  1.51e-22     0       0                2
```

```r
# O resultado abaixo rejeita a hipótese de que os dados faltantes são completamente aleatórios, pois o
# Portanto, neste cenário, é necessário verificar se os dados faltantes são aleatórios ou não aleatório
```

Realizando a imputação de dados

```r
sample_combustiveis_agg_input <- sample_combustiveis_agg

mice(sample_combustiveis_agg_input, method ="norm.predict", seed = 1, m = 1, print = FALSE)
```

```
## Warning: Number of logged events: 6

## Class: mids
## Number of multiple imputations:  1
## Imputation methods:
##              Data            Regiao                UF         Municipio
##                ""                ""                ""                ""
##      CNPJ_Revenda           Produto          Bandeira    Valor_de_Venda
##                ""                ""                ""                ""
##   Valor_de_Compra               PPI  Brent_USD_Barril            Cambio
##    "norm.predict"                ""                ""                ""
## Brent_Real_Barril
##                ""
## PredictorMatrix:
##              Data Regiao UF Municipio CNPJ_Revenda Produto Bandeira
## Data            0      0  0         0            0       0        0
## Regiao          1      0  0         0            0       0        0
## UF              1      0  0         0            0       0        0
## Municipio       1      0  0         0            0       0        0
## CNPJ_Revenda    1      0  0         0            0       0        0
## Produto         1      0  0         0            0       0        0
##              Valor_de_Venda Valor_de_Compra PPI Brent_USD_Barril Cambio
## Data                      1               1   1                1      1
## Regiao                    1               1   1                1      1
## UF                        1               1   1                1      1
## Municipio                 1               1   1                1      1
## CNPJ_Revenda              1               1   1                1      1
## Produto                   1               1   1                1      1
##              Brent_Real_Barril
## Data                         1
## Regiao                       1
## UF                           1
## Municipio                    1
## CNPJ_Revenda                 1
## Produto                      1
## Number of logged events:  6
##   it im dep     meth          out
## 1  0  0     constant       Regiao
## 2  0  0     constant           UF
## 3  0  0     constant    Municipio
## 4  0  0     constant CNPJ_Revenda
## 5  0  0     constant      Produto
```

```
## 6  0  0     constant     Bandeira
```