PONTIFÍCIA UNIVERSIDADE CATÓLICA DE MINAS GERAIS
NÚCLEO DE EDUCAÇÃO A DISTÂNCIA
Pós-graduação *Lato Sensu* em Ciência de Dados e Big Data

Jessica Rosa da Silva Salvador

AUMENTO DO PREÇO DOS COMBUSTÍVEIS AO LONGO DOS ANOS NO BRASIL

Belo Horizonte
2022

Jessica Rosa da Silva Salvador

AUMENTO DO PREÇO DOS COMBUSTÍVEIS AO LONGO DOS ANOS NO BRASIL

Trabalho de Conclusão de Curso apresentado ao Curso de Especialização em Ciência de Dados e Big Data como requisito parcial à obtenção do título de especialista.

Belo Horizonte
2022

SUMÁRIO

1. Introdução

1.1. Contextualização

Os combustíveis são substâncias que queimamos para produzir calor e esse calor pode ser utilizado para acionar motores de veículos (EPE, 2022). Eles são obtidos a partir de uma longa cadeia produtiva, no Brasil, por exemplo, a extração do petróleo acontece na camada do pré-sal a até 7 mil metros de profundidade. Atualmente, os combustíveis mais utilizados no mundo e que serão abordados nesse trabalho são: Gasolina Aditivada, Gasolina comum, Etanol hidratado, Óleo diesel, Óleo diesel S-10 e o GNV (Gás natural veicular).

A Petrobrás é uma empresa estatal brasileira que atua principalmente na exploração e produção do petróleo e seus derivados de gás natural. É reconhecida mundialmente e é a principal produtora de petróleo do país. Todo tipo de combustível, passa pelo processo de como o preço é formulado, a gasolina, como exemplo, após passar por todo processo de refinaria, é vendida para os distribuidores, e nesse momento, são adicionados os impostos que variam para cada estado. Após isso, os distribuidores precisam adicionar outros componentes, nesse caso, a adição do etanol anidro. Hoje, a Lei n° 8.723, de 1993, determina o percentual de 27% de etanol anidro na gasolina comum, e isso gera mais um custo em cima desse valor (PETROBRÁS, 2022). Com a mistura final feita, os distribuidores e revendedores adicionam os seus próprios custos a sua margem de lucro, formando assim o preço final.

Os combustíveis são de vital importância não só para a população brasileira, coma a população global. O petróleo por exemplo é uma das matérias-primas mais importantes da civilização moderna. É usado como fonte de energia e seus derivados são transformados em plástico, borracha, tinta, adesivos, detergentes entre outros. É um dos principais responsáveis por alimentar a economia mundial. Se o preço de um combustível aumenta, consequentemente o frete e os alimentos aumentam também. Isso afeta praticamente toda uma cadeia alimentar e principalmente o bolso da família brasileira.

Os valores dos combustíveis oscilam diariamente, mas existem dois fatores, além da composição dos preços, que contribuem para essa variação:

1) **Cotação do barril de petróleo no mercado internacional**

Essa cotação varia de acordo com o cenário de oferta e demanda de todo o mundo. O preço do petróleo, base da gasolina e do diesel, depende da flutuação do preço do barril no mercado internacional. O produto faz parte de um grupo de commodities (produto de origem primária), com classificação de qualidade e preço padronizado. Mesmo sendo um grande produtor de petróleo, o Brasil ainda precisa importar o produto refinado para atender a demanda local. (WEBPOSTO, 2022)

2) **Taxa de câmbio**

Como a Petrobrás adota o chamado PPI (preço de paridade de importação), os preços da gasolina e do Diesel são equiparados aos desses produtos importados, e tudo que é importado fica mais caro quando o dólar sobe em relação ao real. Como por exemplo, se durante alguns meses, o preço do barril se mantém estável no mercado internacional, mas o dólar dispara, isso é levado em conta na decisão de ajustar os preços dos derivados nas refinarias.

- **Por que esse problema é importante?**

Esse problema é importante, porque o aumento no preço dos combustíveis afeta diretamente o lar e finanças da família brasileira. Quando o valor cresce, tudo fica mais caro, comida, transporte, luz, dentre outros.

- **De quem são os dados analisados? De um governo? Um ministério ou secretaria? Dados de clientes?**

A série histórica de levantamento de preços são dados obtidos diretamente da ANP (Agência nacional de Petróleo), do governo do Brasil. A base de Municípios foi obtida através do serviço de dados do IBGE – Instituto Brasileiro de Geografia

e Estatística, do governo do Brasil. A base de dados do preço do barril foi obtida através da IpeaData, uma plataforma que reúne bases de dados econômicos, financeiros, entre muitos outros. Por fim, a base da taxa de câmbio, que foi obtida através do Olinda, uma plataforma de serviço de dados do Banco Central do Brasil.

- **Quais os objetivos com essa análise? O que iremos analisar?**

Iremos analisar o aumento desses preços dos combustíveis ao longo dos últimos 14 anos e os elementos que contribuem para variação.

- **Trata dos aspectos geográficos e logísticos de sua análise.**

A base de municípios foi utilizada para tratar toda essa questão do “viés geográfico”. Mas apenas estamos considerando no âmbito brasileiro.

- **Qual o período está sendo analisado? A última semana? Os últimos 6 meses? O ano passado?**

O *Dataset* da série histórica do preço dos combustíveis, trazem dados desde o ano de 2001 até 2022. O *Dataset* do preço do barril recupera dados desde o ano de 1987 até 2022. Por último, o *Dataset* da taxa de câmbio recupera os dados de 2008 até 2022. Entretanto, os períodos analisados dos dados em geral são referentes aos últimos 14 anos. Desde 2008 até o ano atual, 2022.

2. Coleta de Dados

Nesse trabalho foram utilizadas 4 fontes distintas de dados. A linguagem de programação utilizada para a análise das informações foi o Python 3.0 e a interface de desenvolvimento foi *Jupyter Notebook*. A seguir serão detalhados a origem, formato e a estrutura dos *Datasets*.

2.1 *Dataset* da série histórica do levantamento de preços dos combustíveis

Esse *Dataset* contempla toda a série histórica do levantamento de preços, a nível de granularidade mensal, segregada por abrangência geográfica e inclui os combustíveis: gasolina aditivada, gasolina comum, etanol hidratado, óleo diesel, gás natural veicular (GNV) e gás liquefeito de petróleo (GLP). Foi obtido no dia ***05/06/2022*** e a fonte foi diretamente do site do governo, ANP- Agência Nacional de Petróleo, https://www.gov.br/anp (Link exato que leva diretamente ao portal se encontra nas referências). A base de dados veio em dois arquivos separados do tipo ***.xlsx*** e foi unificado formando um, apresentando 17 variáveis (colunas) e 34.813 registros.

Listagem descritiva das colunas:

Nome da coluna/campo	Descrição	Tipo
Mês	Mês e ano em que a pesquisa foi realizada	Datetime64[ns]
Produto	Tipo de combustível (gasolina, etanol...)	object
Região	Regiões do Brasil	object
Estado	Estados do Brasil	object
Número De Postos Pesquisados	Somatório de postos pesquisados de determinado Estado	Int64
Unidade De Medida	Unidade De Medida	object
Preço Médio Revenda	Preço médio em Reais de venda do litro do combustível	float64
Desvio Padrão Revenda		float64
Preço Mínimo Revenda	Preço mínimo em Reais da venda do litro do combustível	float64
Preço Máximo Revenda	Preço máximo em Reais da vendo do litro do	float64

	combustível	
Margem Média Revenda	Margem média de lucro em Reais por litro de combustível	object
Coef De Variação Revenda		float64
Preço Médio Distribuição	Preço médio em Reais de compra do litro do combustível	object
Desvio Padrão Distribuição		object
Preço Mínimo Distribuição	Preço mínimo em Reais de compra do litro do combustível	object
Preço Máximo Distribuição	Preço máximo em Reais de compra do litro do combustível	object
Coef De Variação Distribuição		object

2.2 *Dataset* de Municípios

O dataset de Municípios contempla todo o conjunto de distritos do Brasil. Foi obtido no dia ***05/06/2022***, através de uma API de localidades no site https://servicodados.ibge.gov.br/api/docs/localidades. Com a API mencionada, um arquivo no formato json foi baixado, portando descritores com os dados das localidades, como por exemplo, Id_municípios, Municípios, Estado, Macroregião, Microregião, dentre outros. Porém, iremos considerar somente as variáveis da sigla da unidade Federativa (UF) e do Estado. Após todo o tratamento, a base apresentou 2 variáveis (colunas) e 10.649 registros.

Listagem descritiva das colunas:

Nome da coluna/campo	Descrição	Tipo
Sigla_UF	Sigla da Unidade Federativa do Brasil	object
Estado	Estado do Brasil	object

2.3 *Dataset* do Preço por barril do petróleo bruto Brent (FOB)

O terceiro *Dataset* é sobre o preço por barril do petróleo bruto tipo Brent. Produzido no Mar do Norte (Europa), Brent é uma classe de petróleo bruto que serve como benchmark para o preço internacional de diferentes tipos de petróleo.

Neste caso, é valorado no chamado preço FOB (*free on board*), que não inclui despesa de frete e seguro no preço (IPEADATA, 2022). Esta base de dados foi obtida também no dia ***05/06/2022***, através da fonte http://www.ipeadata.gov.br/ e seu arquivo é do tipo ***.csv***, que possui 3 variáveis (colunas) e 12.767 registros.

Listagem descritiva das colunas:

Nome da coluna/campo	Descrição	Tipo
Data	Data do preço em determinado momento	object
Preço - petróleo bruto - Brent (FOB)	Preço por barril do petróleo bruto Brent (FOB)	Float64
Unnamed: 2	---	Float64

2.4 *Dataset* da Taxa de câmbio – Dólar americano

O quarto *Dataset* é sobre taxa de câmbio comercial para venda. A taxa de câmbio é o preço de uma moeda estrangeira medido em unidades ou frações (centavos) da moeda nacional, refletindo o custo de uma moeda em relação à outra. O dólar comercial é a cotação do valor do dólar utilizado nas operações realizadas no mercado de câmbio, por exemplo: exportação, importação, transferências financeiras. Também conhecida como PTAX, esta cotação corresponde à média aritmética das taxas de venda das consultas realizadas diariamente (IPEADATA, 2022). Essa base foi obtida no dia ***19/06/2022***, através da fonte https://olinda.bcb.gov.br/olinda/servico/PTAX/versao/v1/documentacao (Link exato que leva diretamente ao portal se encontra nas referências). e seu arquivo é do tipo ***.csv***, que possui 3 variáveis (colunas) e 3.633 registros.

Listagem descritiva das colunas:

Nome da coluna/campo	Descrição	Tipo
cotacaoCompra	Cotação de compra do dólar contra a unidade monetária corrente: unidade monetária corrente/dólar americano.	object
cotacaoVenda	Cotação de venda do dólar contra a unidade monetária corrente: unidade monetária corrente/dólar americano.	object
dataHoraCotacao	Data, hora e minuto das cotações de compra e venda.	object

3. Processamento/Tratamento de Dados

A estrutura do código está dividida em: primeiro, na coleta do *Dataset* e em seguida no tratamento do mesmo, de forma individual, sucessivamente. Aqui trataremos da mesma forma.

3.1 Tratamento do Dataset da série histórica do levantamento de preços dos combustíveis

O dataset apresenta 17 colunas e 34.813 registros.

Correções

A primeira correção realizada foi de converter o tipo da coluna [“PREÇO MÉDIO DISTRIBUIÇÃO”] para *to_numeric*. Foi acrescentado o *coerce* pois existiam valores com **‘-‘**, e isso evitou que desse erro no momento da troca.

```
In [13]: # Acertando a tipagem das colunas para poder unir os dois dataframes
         # Como tem valores com '-' os mesmos ficam nulos quando acontece a conversão
         base_df['PREÇO MÉDIO DISTRIBUIÇÃO'] = pd.to_numeric(base_df['PREÇO MÉDIO DISTRIBUIÇÃO'], errors='coerce')
```

Também existiam alguns dados com a grafia incorreta, exemplo, a palavra ‘OLEO DIESEL’ com e sem acento. Todas as grafias foram consertadas.

```
In [15]: #Acertando os valores
         base_df = base_df.replace("OLEO DIESEL", "ÓLEO DIESEL")
         base_df = base_df.replace("OLEO DIESEL S10", "ÓLEO DIESEL S10")
         base_df = base_df.replace("AMAPA", "AMAPÁ")
         base_df = base_df.replace("CEARA", "CEARÁ")
         base_df = base_df.replace("ESPIRITO SANTO", "ESPÍRITO SANTO")
         base_df = base_df.replace("GOIAS", "GOIÁS")
         base_df = base_df.replace("MARANHAO", "MARANHÃO")
         base_df = base_df.replace("PARA", "PARÁ")
         base_df = base_df.replace("PARAIBA", "PARAÍBA")
         base_df = base_df.replace("PARANA", "PARANÁ")
         base_df = base_df.replace("PIAUI", "PIAUÍ")
         base_df = base_df.replace("RONDONIA", "RONDÔNIA")
         base_df = base_df.replace("SAO PAULO", "SÃO PAULO")
```

Foi necessário a criação de uma nova coluna [‘ANO’], pois será com ela que que utilizaremos para montar os gráficos para as análises. E na coluna [‘MÊS’], o dia foi retirado, assumindo o formato de Ano-Mês, exemplo, ‘2022-01’. Foi mantido a coluna mês, em razão que será realizado com ela a junção com os outros *Datasets*.

```
In [17]: #Criando a coluna ano
         base_df['ANO'] = base_df['MÊS'].dt.year

In [18]: # Cria a coluna DATA removendo os dias
         base_df['MÊS'] = base_df['MÊS'].dt.to_period('M')
```

As colunas a seguir foram removidas, dado que não utilizaremos para as análises.

```
In [107]: #excluindo colunas que não serão usadas do dataframe
          base_df = base_df.drop(["NÚMERO DE POSTOS PESQUISADOS","UNIDADE DE MEDIDA","DESVIO PADRÃO REVENDA","PREÇO MÍNIMO REVENDA",
                                  "PREÇO MÁXIMO REVENDA","MARGEM MÉDIA REVENDA","DESVIO PADRÃO DISTRIBUIÇÃO","PREÇO MÍNIMO DISTRIBUIÇÃO",
                                  "PREÇO MÁXIMO DISTRIBUIÇÃO","COEF DE VARIAÇÃO DISTRIBUIÇÃO","COEF DE VARIAÇÃO REVENDA"], axis=1)
```

Filtragem

Todos os dados referentes aos anos de 2007 para trás foram removidos, visto que iremos analisar os últimos 14 anos de pesquisa [2008-2022]. Foi desconsiderado também o combustível GLP (mais conhecido como o gás de cozinha), dado que só iremos considerar combustíveis usados por qualquer tipo de automóvel.

```
In [20]: #removendo 2011 para trás
         dfremove = base_df.loc[(base_df['MÊS']<= '2007-12')]
         base_df = base_df.drop(dfremove.index)

         #Removendo o combustível GLP
         dfremove_glp = base_df.loc[(base_df['PRODUTO'] == 'GLP')]
         base_df = base_df.drop(dfremove_glp.index)
         base_df['PRODUTO'].unique()

Out[20]: array(['ETANOL HIDRATADO', 'GASOLINA COMUM', 'GNV', 'ÓLEO DIESEL',
                'ÓLEO DIESEL S10', 'GASOLINA ADITIVADA'], dtype=object)
```

Tratamento de dados ausentes

Depois de ter convertido a coluna para numeric, os dados que estavam com ‘-‘ se tornaram ausentes com o valor ‘NaN’. Para resolver esse problema, foi preenchido o valor ausente com a mediana calculada sobre essa coluna. Por mais que esse campo não será utilizado nas futuras análises, foi preciso mantê-lo pois todos

os dados referentes ao ano de 2022 ficaram nulos depois do valor convertido. Mediante a isso, decidi manter com o valor da mediana ao invés de excluir a coluna por inteira.

	column type	null values	null values (%)
MÊS	period[M]	0	0.0
PRODUTO	object	0	0.0
REGIÃO	object	0	0.0
ESTADO	object	0	0.0
PREÇO MÉDIO REVENDA	float64	0	0.0
DESVIO PADRÃO REVENDA	float64	0	0.0
COEF DE VARIAÇÃO REVENDA	float64	0	0.0
PREÇO MÉDIO DISTRIBUIÇÃO	float64	1952	19.056917
DESVIO PADRÃO DISTRIBUIÇÃO	object	0	0.0
COEF DE VARIAÇÃO DISTRIBUIÇÃO	object	0	0.0
ANO	int64	0	0.0

```
In [22]: # Colocando o valor da mediana no lugar dos valores nulos
base_df['PREÇO MÉDIO DISTRIBUIÇÃO'].fillna(base_df["PREÇO MÉDIO DISTRIBUIÇÃO"].median(), inplace = True)
```

Após todos esses tratamentos, não foram encontrados dados duplicados no *Dataset* final.

```
In [23]: #Verificação de dados duplicados, resultado: 0
linhas_dup_combs = base_df[base_df.duplicated(keep=False)]
linhas_dup_combs.shape

Out[23]: (0, 11)
```

3.2 Tratamento do Dataset de Municípios

O *Dataset* apresenta 2 colunas e 10.649 registros.

Tratamento de dados ausentes

A base de municípios, após a remoção das colunas que não serão utilizadas, como por exemplo, mesorregião, microrregião, acarretou na duplicidade gerando 10.648 registros repetidos.

```
In [31]: #Verificação de dados duplicados: (ANTES): resultado: 7284
         linhas_dup_municipio = municipios[municipios.duplicated(keep=False)]
         linhas_dup_municipio.shape

Out[31]: (10648, 2)
```

Para remover toda duplicidade, foi utilizado o método *drop_duplicates()*, conforme a figura abaixo:

```
In [39]: #Removendo linhas duplicadas do dataset de municipios
         municipios_df = municipios.drop_duplicates()

         #Após a remoção das duplicatas, resultado: 0
         linhas_dup_municipio = municipios_df[municipios_df.duplicated(keep=False)]
         linhas_dup_municipio.shape

Out[39]: (0, 2)
```

Junção do *Dataframe* com a base principal

Para que fosse possível fazer o merge nas duas bases (combustíveis e municípios), a coluna ‘ESTADO’ foi elegida para realizar o link entre eles. Devido a isso, foi preciso colocar todas as letras em maiúsculo para suceder ao join. Após isso, o merge foi realizado com sucesso.

```
In [37]: #Colocando todos os estados em maiusculos para o merge com a base principal
         municipios['ESTADO'] = municipios['ESTADO'].str.upper()

         # merge dos dois dataframes em um único
         base_comb_df = pd.merge(base_df, municipios_df, how='left', on=['ESTADO'])
         display(base_comb_df)
```

Após todos esses tratamentos, não foram encontrados dados duplicados no *Dataset* final.

3.3 Tratamento do Dataset do Preço por barril do petróleo bruto Brent (FOB)

O *Dataset* apresenta 3 colunas e 12.767 registros. Primeiramente, foram realizados o tratamento de remoção de colunas desnecessárias e o *rename* da coluna principal de preços.

```
In [43]: #Renomeando a coluna
         base_prc_barril.rename(columns={'Preço - petróleo bruto - Brent (FOB) - US$ - Energy Information Administration (EIA) - EIA366_PE

         #Deletando colunas desnecessárias
         prc_barril_df = base_prc_barril.drop(["Unnamed: 2"], axis=1)
```

Após isso, foi removido os dias no campo da [‘DATA’], deixando no formato Ano-Mês.

```
In [44]: # Cria a coluna DATA removendo os dias
         prc_barril_df['Data'] = pd.to_datetime(prc_barril_df['Data'])
         prc_barril_df['Data'] = prc_barril_df['Data'].dt.to_period('M')
```

Em seguida, foi gerado um agrupamento pelo campo [‘Data’] e realizado a média sobre a coluna [‘PRC_BARRIL_BRUTO’], para que cada Ano-Mês tivesse seu devido valor, sem repetição. Logo, o campo data foi setado como índice, para realizar o merge posteriormente com o *Dataset* principal.

```
In [56]: # Agrupa também as datas pelo mês, fazendo a média do valor
         prc_barril_mes_df = prc_barril_df.groupby(['Data']).agg({'PRC_BARRIL_BRUTO': np.mean}).reset_index()

In [57]: # Define data como index da coluna para o merge posterior
         prc_barril_mes_df = prc_barril_mes_df.set_index('Data')

In [178]: #Verificando existencia de duplicatas: resultado: 0
          linhas_dup_barril = prc_barril_mes_df[prc_barril_mes_df.duplicated(keep=False)]
          linhas_dup_barril.shape

Out[178]: (0, 1)

In [58]: prc_barril_mes_df.head(5)
```

Out[58]:

Data	PRC_BARRIL_BRUTO
1987-01	18.442000
1987-02	18.551667
1987-03	18.920000
1987-04	18.698000
1987-05	18.643333

Após todos esses tratamentos, não foram encontrados dados duplicados no dataset final.

3.4 Tratamento do Dataset da Taxa de câmbio – Dólar americano

O *Dataset* apresenta 3 colunas e 3.633 registros. Primeiramente, foram removidos os dias da coluna [‘DATA’], deixando no formato Ano-Mês e a coluna [‘cotacaoCompra’] foi removida do *Dataset*, dado que somente será analisado a cotação de venda.

```
In [62]: # Cria a coluna DATA removendo os dias
         df_cambio['dataHoraCotacao'] = pd.to_datetime(df_cambio['dataHoraCotacao'])
         df_cambio['dataHoraCotacao'] = df_cambio['dataHoraCotacao'].dt.to_period('M')

In [63]: #Deletando a coluna 'cotacaoCompra' pois não a usaremos para a análise
         df_cambio = df_cambio.drop(["cotacaoCompra"], axis=1)
```

O campo [‘cotacaoVenda’] estava com o tipo object e para converte-lê foi preciso primeiro substituir a vírgula pelo ponto utilizando o lambda para tal, nos valores dessa coluna. Após a substituição, o campo foi convertido para to_numeric.

```
In [64]: #Como a coluna cotacaoVenda é do tipo object, preciso primeiramente substituir a vírgula pelo ponto
         #Pois assim não dará erro na hora de converter seu valor para numérico
         df_cambio['cotacaoVenda'] = df_cambio['cotacaoVenda'].apply(lambda x: float(x.replace(",",".")))

In [65]: #convertendo a coluna para numeric
         df_cambio['cotacaoVenda'] = pd.to_numeric(df_cambio['cotacaoVenda'], errors='coerce')
```

Em seguida, foi gerado um agrupamento pelo campo [‘dataHoraCotacao’] e realizado a média sobre a coluna [‘cotacaoVenda’], para que cada Ano-Mês tivesse seu devido valor, sem repetição. Logo, o campo data foi setado como índice, para realizar o merge posteriormente com o *Dataset* principal.

```
In [66]: # Agrupa por Ano/Mês, fazendo a média do valor da cotação da venda
         df_tx_cambio = df_cambio.groupby(['dataHoraCotacao']).agg({'cotacaoVenda': np.mean}).reset_index()

In [67]: #Setando a colunadaHoraCotacao como índice para unir com a base principal posteriormente
         df_tx_cambio = df_tx_cambio.set_index('dataHoraCotacao')
         display(df_tx_cambio)
```

	cotacaoVenda
dataHoraCotacao	
2008-01	1.774259
2008-02	1.727742
2008-03	1.707580
2008-04	1.688929
2008-05	1.660535

Após todos esses tratamentos, não foram encontrados dados duplicados no *Dataset* final. Concluindo, todos esses tratamentos nas bases individualmente, foi realizado a união com os campos necessários, no qual a data foi criada como índice.

```
In [50]: # Copia os valores do dataframe para um dataframe definitivo
         df_prc_comb = base_comb_df.copy()
         # Reseta o index para um valor sequencial e depois seta a coluna DATA como novo index
         df_prc_comb.reset_index(inplace=True)
         df_prc_comb.set_index('MÊS', inplace=True)

In [51]: # Agrega as colunas dos datasets anteriores (Barril, Câmbio, Inflação...)

In [52]: df_prc_comb['PRC_BARRIL_BRUTO'] = prc_barril_mes_df['PRC_BARRIL_BRUTO']

In [53]: df_prc_comb['TAXA CÂMBIO'] = df_tx_cambio['cotacaoVenda']

In [54]: # Deletando coluna desnecessária
         df_prc_comb = df_prc_comb.drop(["index"], axis=1)
```

O *Dataset* principal ficou dessa maneira:

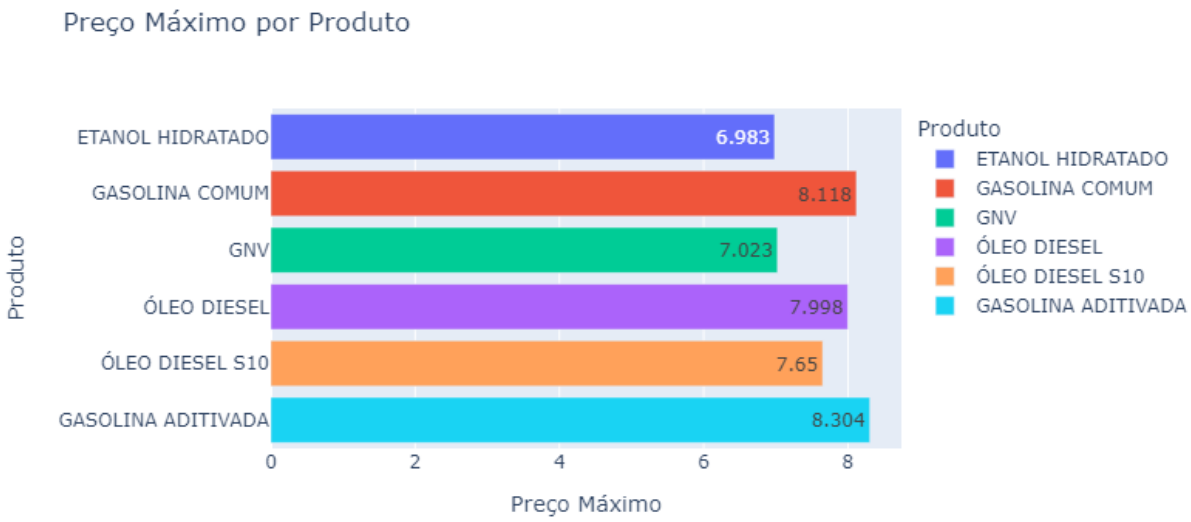
Out[55]:

MÊS	PRODUTO	REGIÃO	ESTADO	PREÇO MÉDIO REVENDA	PREÇO MÉDIO DISTRIBUIÇÃO	ANO	SIGLA_UF	PRC_BARRIL_BRUTO	TAXA CÂMBIO
2022-05	ÓLEO DIESEL S10	NORTE	RORAIMA	7.186	2.169	2022	RR	93.6	4.95505
2022-05	ÓLEO DIESEL S10	SUL	SANTA CATARINA	6.788	2.169	2022	SC	93.6	4.95505
2022-05	ÓLEO DIESEL S10	SUDESTE	SÃO PAULO	6.847	2.169	2022	SP	93.6	4.95505
2022-05	ÓLEO DIESEL S10	NORDESTE	SERGIPE	6.982	2.169	2022	SE	93.6	4.95505
2022-05	ÓLEO DIESEL S10	NORTE	TOCANTINS	6.956	2.169	2022	TO	93.6	4.95505

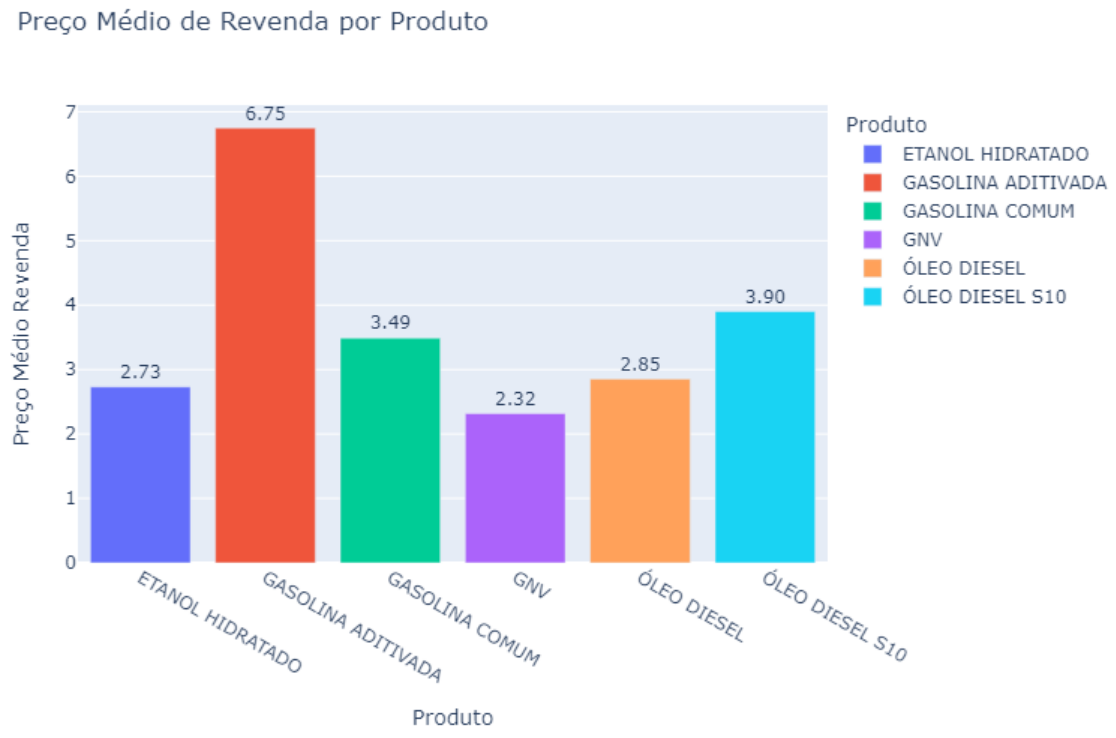
4. Análise e Exploração dos Dados

No Brasil, os veículos são movidos predominantemente com motores a combustão interna, basicamente motores do ciclo Otto e do ciclo diesel. Os veículos leves de passageiros utilizam como combustível o etanol hidratado, a gasolina e o gás natural veicular (GNV). No caso dos veículos *flex-fuel*, podem utilizar gasolina ou etanol hidratado. As Motocicletas utilizam gasolina e os modelos *flex-fuel* podem utilizar também etanol hidratado. Os Comerciais Leves podem utilizar etanol hidratado, gasolina, GNV e também o diesel. Podem também ser do tipo *flex-fuel* e utilizar gasolina ou etanol hidratado. Os Veículos Pesados de modo geral utilizam somente o diesel como combustível (CETESB, 2022).

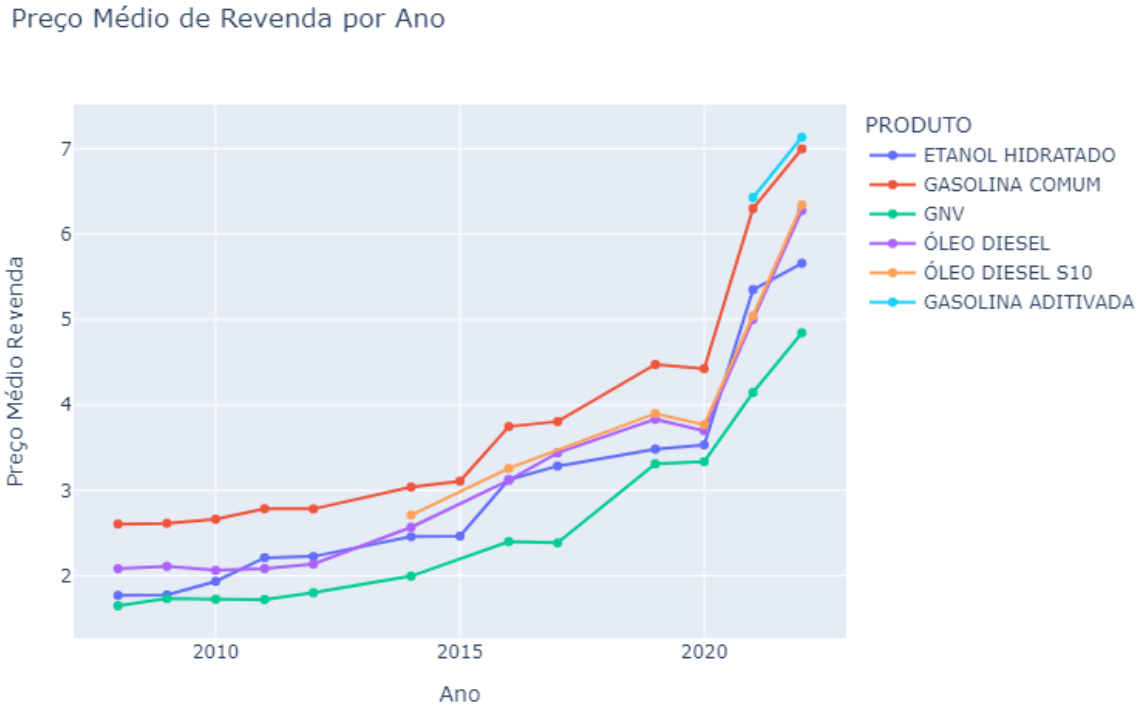
No gráfico abaixo, mostra-se os valores máximos, no geral, de cada produto. É possível notar que a Gasolina Aditivada lidera o ranking, seguida pela Gasolina comum e fechando o pódio em terceiro lugar, o óleo diesel.



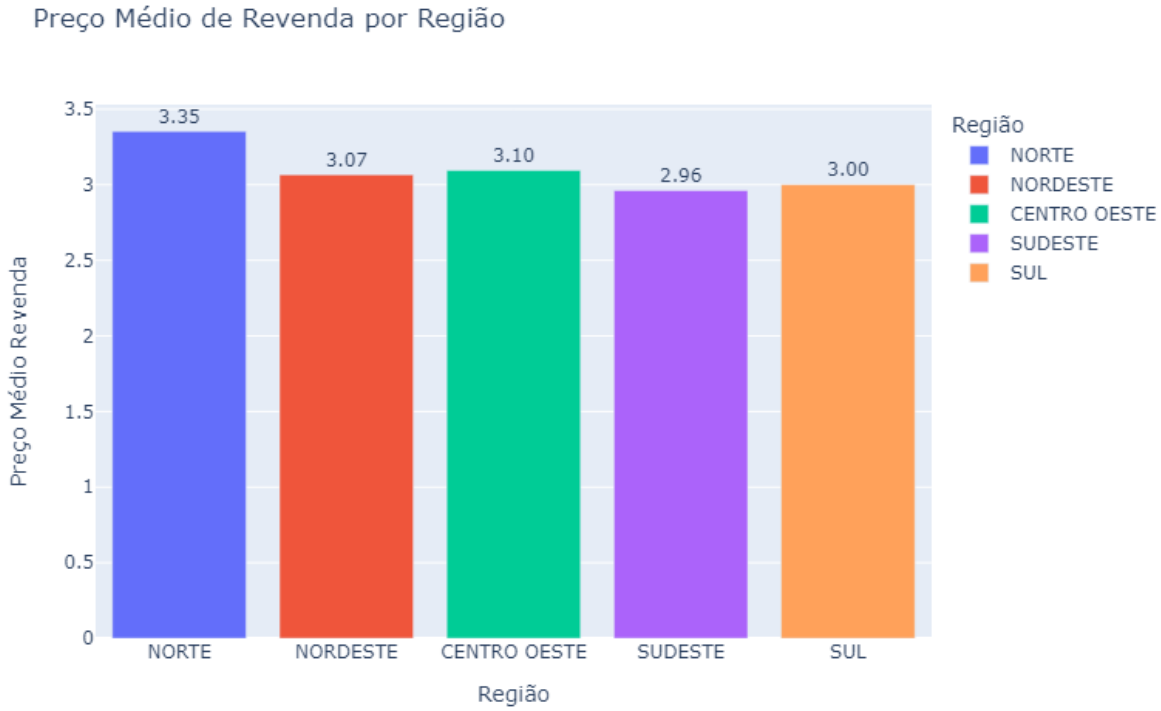
No próximo gráfico, podemos analisar o preço médio de revenda por produto. Aqui, nitidamente podemos perceber que os três combustíveis mais caros do Brasil, são: a Gasolina Aditivada, Óleo Diesel S10 e Gasolina comum.



No gráfico de preço médio de revenda por ano, podemos perceber um crescente nas linhas, ou seja, houve um aumento significativamente nos preços dos combustíveis ao longo dos anos. Desde o início da série foram poucos os períodos com redução de valores.

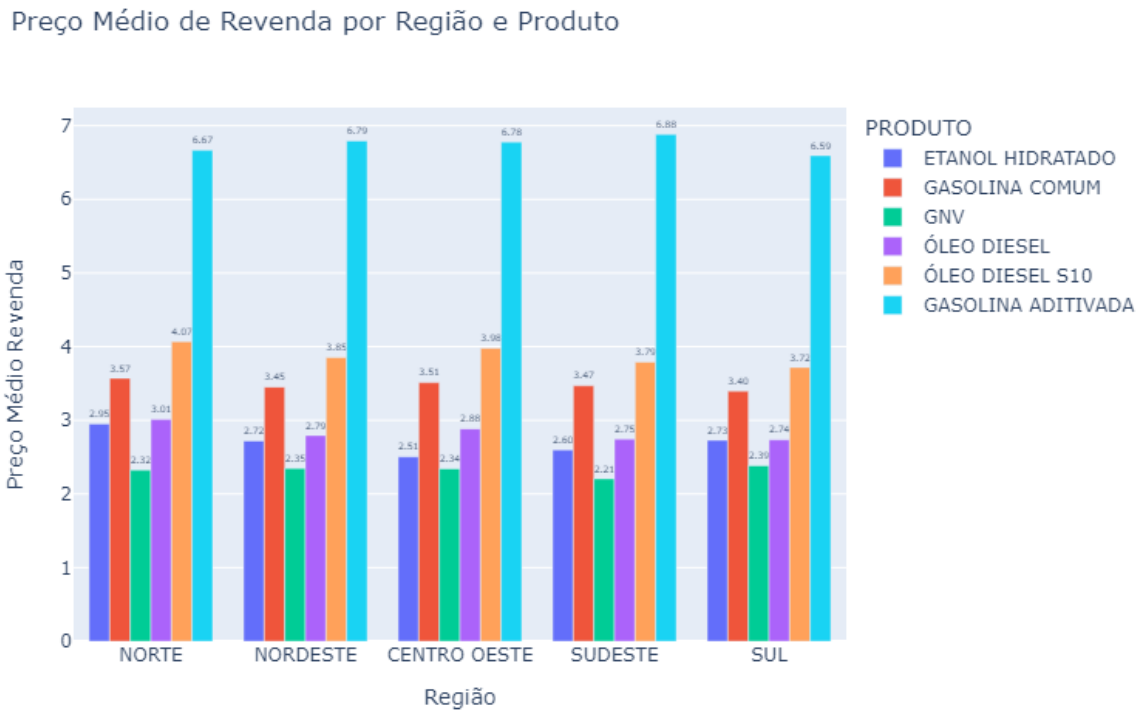


Partindo para uma análise a nível de região, a região Norte lidera ao apresentar o maior preço médio no Brasil. Segundo o gráfico abaixo, é possível notar que as regiões mais distintas, Norte, Nordeste e Centro-Oeste, possuem os valores mais caros de combustíveis. Um insight que é plausível levantar para essa situação, além dos tributos, seria o custo do frete devido a distância e condição de estradas para essa região.

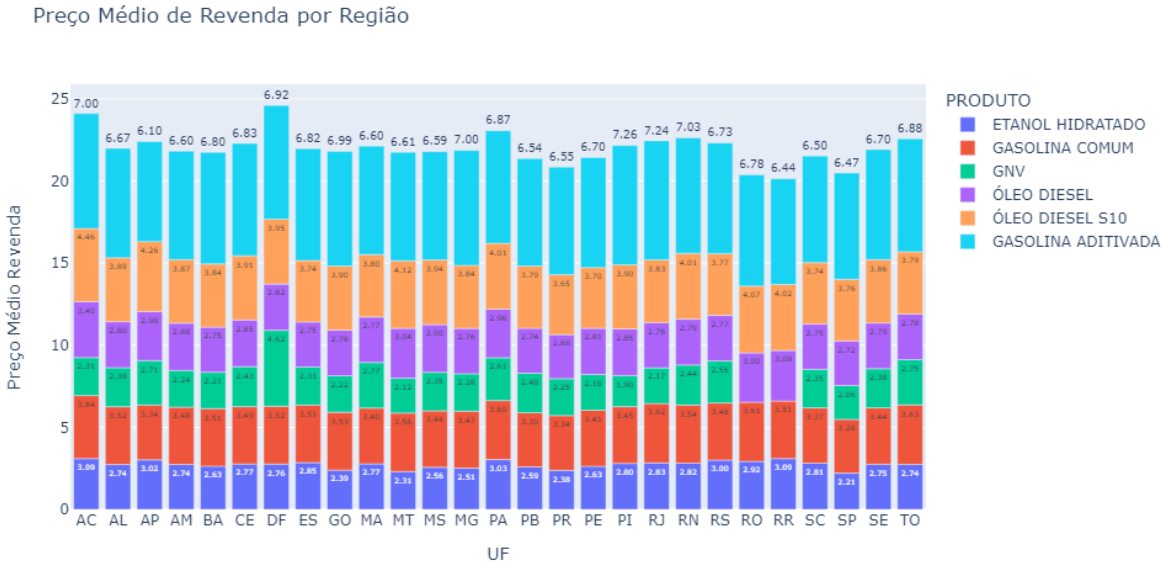


Um levantamento feito pela Ticket Log, realizado de 1 a 15 de agosto, mostra que região Norte do Brasil apresentou a média de preços mais altos para quase todos os combustíveis.

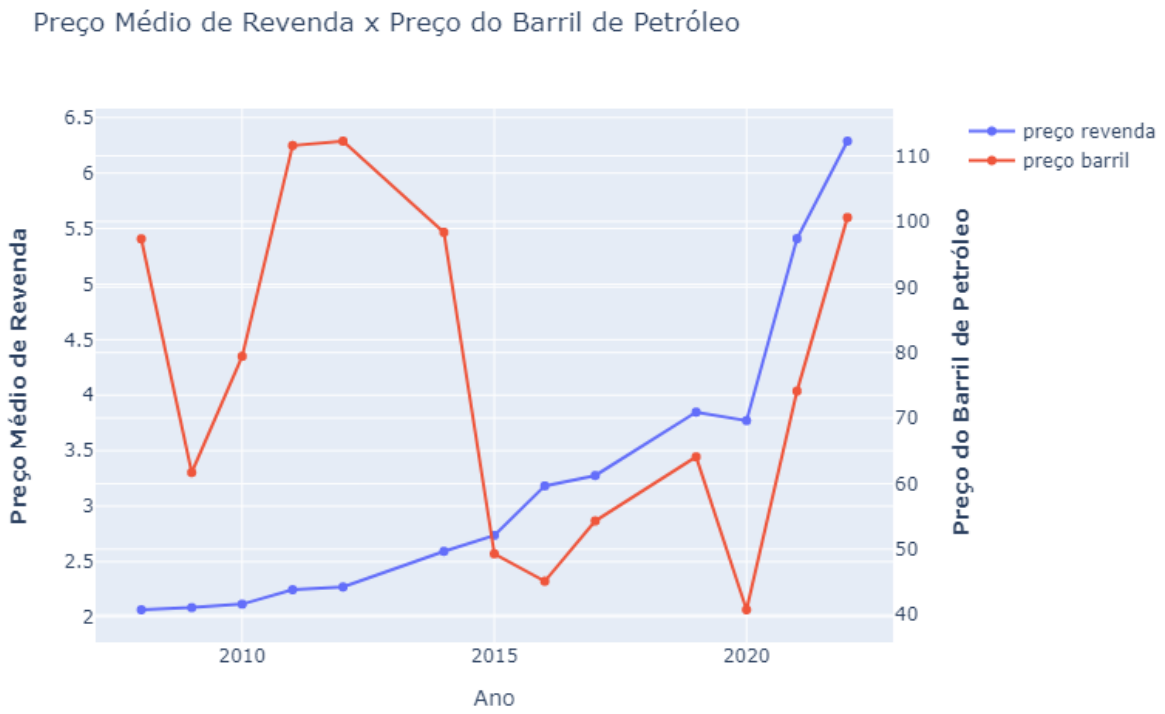
Outro ponto que podemos destacar é o preço do litro do óleo diesel estar mais caro que o litro da gasolina comum. Em todas as regiões do Brasil, ele segue em segundo no ranking. Os estoques globais estão reduzidos e abaixo das mínimas sazonais dos últimos cinco anos nas principais regiões supridoras. Esse desequilíbrio resultou na elevação dos preços do diesel no mundo inteiro, com a valorização deste combustível muito acima da valorização do petróleo. A diferença entre o preço do diesel e o preço do petróleo nunca esteve tão alta (AGÊNCIA PETROBRÁS, 2022).



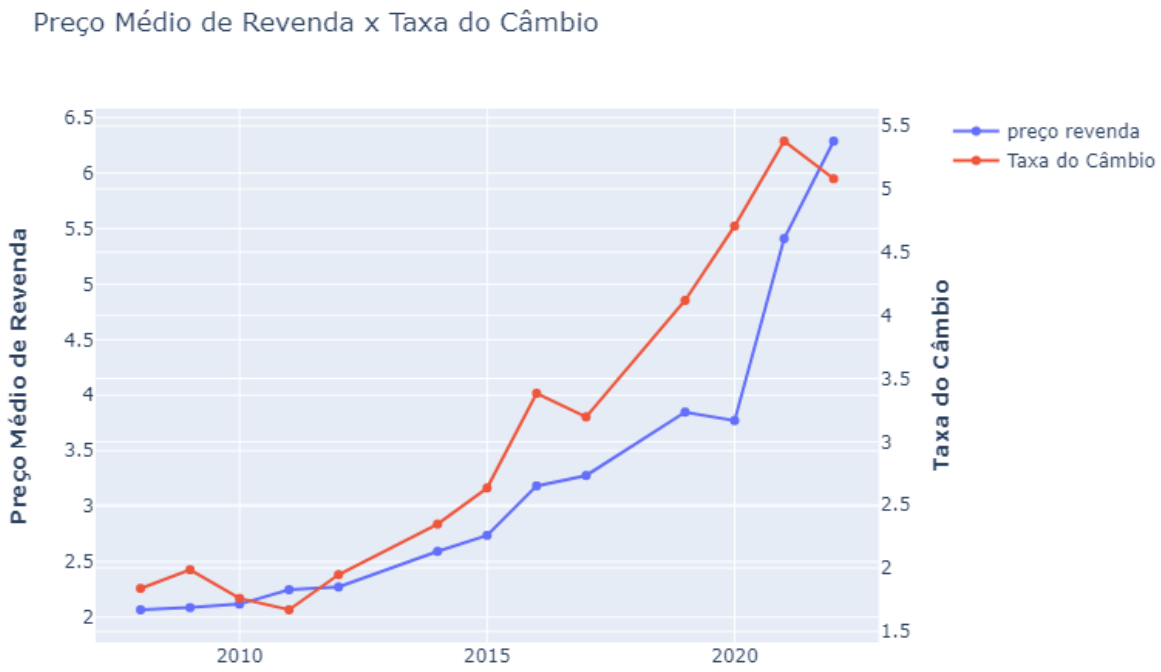
No gráfico a seguir, sobre o preço médio de revenda de cada estado do Brasil, podemos destacar o quão elevado estão os preços na região norte. É possível notar que no estado do Acre, Roraima, Tocantis, acumula-se os combustíveis mais caros vendidos. Isso ratifica ainda mais a hipótese de que nessas regiões a justificativa por um valor tão alto é devido a difícil logística da região.



Analisando o quadro da correlação, considerando o período total, podemos ver que a variável de preço médio de revenda tem uma alta correlação com as variáveis da Inflação absoluta e a taxa de cambio. Já com a variável do preço do barril a correlação é baixa negativamente e isso está ilustrado nos 3 gráficos abaixo. Entre os anos de 2008 a 2014 pode-se perceber a baixíssima correlação entre eles. A partir de 2015 até 2019, o quadro mudou, passando para uma correlação moderada. Levantando uma hipótese para esse evento, seria a implantação da política do PPI (preço de paridade de importação), que é uma referência calculada com base no preço de aquisição do combustível. A Petrobrás passou a trabalhar com o PPI em 2016 (UDOP, 2022), o que possivelmente pode ter sido a causa desta maior correlação.



Entre os anos de 2008 e 2015, referente ao gráfico acima, diferente do que ocorreu com o barril de petróleo, houve uma correlação próxima de moderada. Isso levanta a hipótese que, apesar do controle de preços, os reajustes nesse período podem ter sido motivados pelo câmbio. Sendo assim, considerando o período total, a correlação entre o câmbio e preço médio de revenda é alta, o que demonstra a boa capacidade desse elemento em explicar variações nos preços dos combustíveis.



5. Criação de Modelos de Machine Learning

Visando dar continuidade as análises do *Dataset*, vamos aplicar dois tipos de algoritmos de regressão, em busca de prever o comportamento de uma variável ao longo dos próximos anos.

O primeiro modelo escolhido foi o de regressão linear. A regressão linear se trata de uma técnica onde aproximamos a relação entre a variável independente (ou explicativa) e a variável resposta (ou dependente) por uma reta que representa a relação da melhor forma possível. Desta forma se eu não tenho o valor da variável resposta, basta aplicar o valor da variável explicativa na equação da reta que descobrimos, e conseguimos aproximar o valor da variável resposta. A equação da reta pode ser expressada por: ***y=Ax + b.***

y = variável resposta

x = variável independente

A = coeficiente angular

B = constante de interceptação

Ou seja, se eu conheço A e B que são os coeficientes da reta, é possível imputar o valor de X e consequentemente obter um resultado aproximado de Y. A Regressão Linear é uma ferramenta estatística muito poderosa que pode ser usada para:

- Prever o comportamento de uma determinada variável;
- Entender a relação entre variáveis;
- Saber quais características são significativas estatisticamente e quais não são.

No *Dataset* principal, iremos filtrar todos os dados de Preço médio de revenda e taxa de câmbio do combustível Gasolina comum, referente ao Estado do Rio de Janeiro.

```
In [191]: models = {}

In [192]: reg_df = df_prc_comb[df_prc_comb.SIGLA_UF == 'RJ'].groupby(['PRODUTO', 'MÊS']).mean()

In [193]: reg_df = reg_df.iloc[reg_df.index.get_level_values('PRODUTO') == 'GASOLINA COMUM'].groupby('MÊS').mean()

In [194]:  # Separa eixos da regressão
          X = reg_df.drop(['PREÇO MÉDIO REVENDA','PREÇO MÉDIO DISTRIBUIÇÃO','ANO','PRC_BARRIL_BRUTO'] , axis =1)
          Y = reg_df['PREÇO MÉDIO REVENDA'].values
```

Após X e Y serem definidos, instanciamos o modelo de regressão linear, treinamos a base de treino e de teste e vamos prever valores do preço médio de revenda a partir de um determinado valor da taxa de câmbio.

```
In [195]: x_train, x_test, y_train, y_test = train_test_split(X,Y,test_size=0.30)

In [196]: # Cria model de regressão e faz o treinamento
          lr=LinearRegression()
          lr.fit(x_train,y_train)

          models = lr
          predicted_values = []
          for i in range(0, len(y_test)):
              predicted_values.append(lr.predict(x_test.iloc[[i],:])[0])

In [151]: predicted_df = pd.DataFrame({'Cambio':x_test['TAXA CÂMBIO'].values,
                                       'Valor Real':y_test, 'Valor Predito':predicted_values})
```

Procurando avaliar a precisão do modelo, foi escolhido usar a técnica muito comum do R^2. O valor de R^2 ou R-squared é uma medida estatística que nos mostra o quão próximos os dados estão ajustados à linha de regressão. É um valor de 0 à 1

que, quanto mais próximo de 1, melhor o ajuste e menor o erro associado. Quando for 0% indica que o modelo não explica nada da variabilidade dos dados de resposta ao redor de sua média. Quando for 100%, indica que o modelo explica toda a variabilidade dos dados de resposta ao redor de sua média.

Nosso R^2 resultou em 0.8657, o que pode ser considerado que o modelo performou bem, em termos de precisão.

```
In [152]: r2_score(predicted_df["Valor Real"], predicted_df["Valor Predito"])
Out[152]: 0.867593640594648
```

O segundo modelo escolhido foi o *Randon Forest Regressor*. O *Random Forest* (ou Floresta Aleatória) é um método de aprendizado conjunto para classificação e regressão que opera construindo várias árvores de decisão no momento do treinamento e produzindo a classe, que é o modo das saídas geradas por árvores individuas. De igual modo ao anterior, selecionar X e Y utilizando a mesma base de dados, instanciamos o modelo do Random Forest Regressor e treinamos as bases de treino e teste, para posteriormente prevermos os valores através do *predict.*

Randon forest regressor

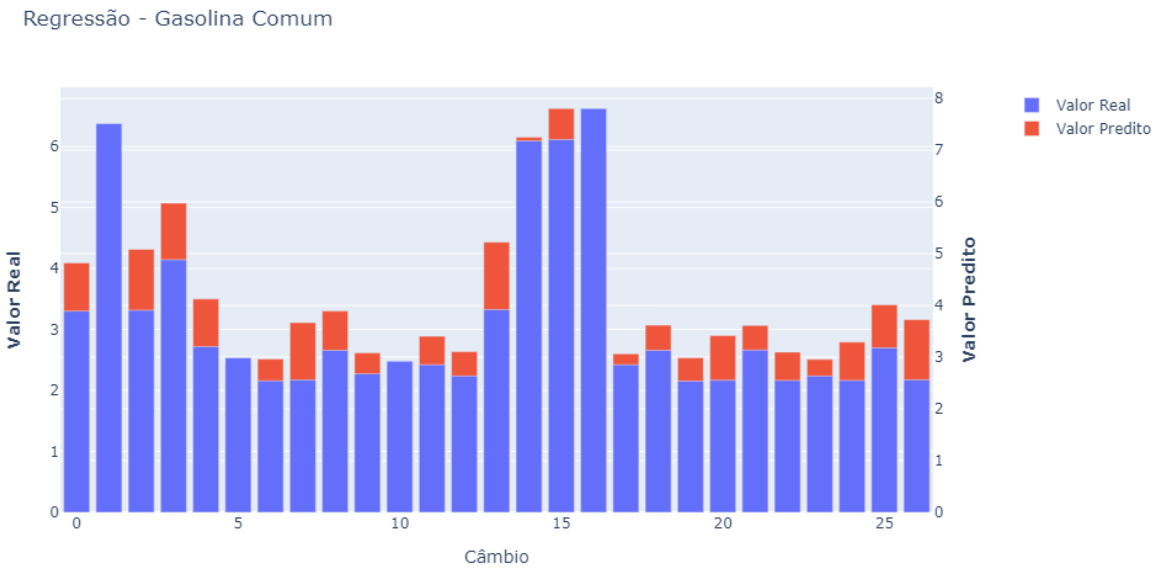
```
In [154]: models_RF = {}

In [155]: # Separa eixos da regressão
          X = reg_df.drop(['PREÇO MÉDIO REVENDA','PREÇO MÉDIO DISTRIBUIÇÃO','ANO','PRC_BARRIL_BRUTO'] , axis =1)
          Y = reg_df['PREÇO MÉDIO REVENDA'].values

In [156]: x_train, x_test, y_train, y_test = train_test_split(X,Y,test_size=0.30)

In [157]: rf = RandomForestRegressor()

In [158]: rf.fit(x_train, y_train)
Out[158]: RandomForestRegressor()

In [159]: models_RF = rf
          predicted_values_rf = []
          for i in range(0, len(y_test)):
              predicted_values_rf.append(rf.predict(x_test.iloc[[i],:])[0])

In [160]: predicted_rf_df = pd.DataFrame({'Cambio_rf':x_test['TAXA CÂMBIO'].values,
                                          'Valor_Real':y_test, 'Valor_Predito':predicted_values_rf})
```

Avaliando o resultado, nosso R2 também performou bem, conforme imagem abaixo:

```
In [161]: r2_score(predicted_rf_df["Valor_Real"], predicted_rf_df["Valor_Predito"])
Out[161]: 0.8570260903372579
```

Contudo, ao tentar prever valores futuros mais adiante (será explicado no capítulo 6), o algoritmo de regressão linear apresentou melhor performance, com mais coesão ao esperado nos resultados. Mediante a isso, ele foi o eleito como nosso modelo.

A seguir, o gráfico abaixo apresenta a relação entre o valor real e o valor predito. É possível perceber que o valor predito fica bem próximo ao real.



6. Apresentação dos Resultados

Conforme vimos no capítulo 4, sobre as análises, ao longo desses quase 15 anos, o aumento nos preços dos combustíveis foi notório. É uma curva crescente e que não sabemos onde irá parar. Visto isso, foi criado um modelo de regressão que nos possibilita prever valores desses preços para o futuro. Foi utilizado o parâmetro da taxa do câmbio, que é um dos fatores que como visto anteriormente, contribui definitivamente para o aumento desses preços.

O *Dataset* final, somente considerou o preço predito para a combustível gasolina comum referente ao Estado do Rio de Janeiro. Abaixo segue o código em *python* e o resultado da previsão.

Previsão dos valores

```
In [198]: resultado = {}

In [199]: resultado['2020-01'] = df_tx_cambio['cotacaoVenda'].iloc[-1]

In [200]: for i in range(2021, 2031):
              ano_passado = str(i - 1)+'-01'
              ano_corrente = str(i)+'-01'
              resultado[ano_corrente]  = (resultado[ano_passado] * 0.04) + resultado[ano_passado]

In [202]: resultado_df = pd.DataFrame(list(resultado.items()), columns = ['MÊS', 'TAXA CÂMBIO'])

In [203]: resultado_df['MÊS'] = pd.to_datetime(resultado_df['MÊS'])
          resultado_df.set_index('MÊS', inplace=True)

In [204]: resultado_df['PREVISÃO'] = models.predict(resultado_df)
```

O resultado obtido foi:

MÊS	TAXA CÂMBIO	PREVISÃO
2020-01-01	4.937617	5.882295
2021-01-01	5.135121	6.087069
2022-01-01	5.340526	6.300033
2023-01-01	5.554147	6.521516
2024-01-01	5.776313	6.751858
2025-01-01	6.007366	6.991413
2026-01-01	6.247660	7.240551
2027-01-01	6.497567	7.499655
2028-01-01	6.757469	7.769122
2029-01-01	7.027768	8.049369
2030-01-01	7.308879	8.340825

Para o ano de 2030, caso a taxa de câmbio for no valor de 7.308879 a previsão do preço de revenda é de 8.340825.

7. Links

- Github:
https://github.com/jrosasalvador/TCC_PUC_Minas_2022

- Link do vídeo de apresentação:
https://youtu.be/LhnIbQ81MDg

- Dataset Série histórica do levantamento de preços:
https://www.gov.br/anp/pt-br/assuntos/precos-e-defesa-da-concorrencia/precos/precos-revenda-e-de-distribuicao-combustiveis/serie-historica-do-levantamento-de-precos

- Dataset Municípios:
https://servicodados.ibge.gov.br/api/docs/localidades

- Dataset Taxa de Câmbio:
https://olinda.bcb.gov.br/olinda/servico/PTAX/versao/v1/aplicacao#!/recursos/CotacaoDolarPeriodo#eyJmb3JtdWxhcmlvIjp7IiRmb3JtYXQiOiJqc29uIiwiJHRvcCI6MTAwfX0=

- Dataset Preço por barril de Petróleo bruto brent (FOB):
http://www.ipeadata.gov.br/ (Seção Macroeconômico)

REFERÊNCIAS

AGÊNCIA PETROBRÁS. *Após 60 dias a Petrobras reajustará seus preços de diesel.* Disponível em < https://www.agenciapetrobras.com.br/Materia/ExibirMateria?p_materia=984303 >

CETESB. *Combustíveis.* Disponível em < https://cetesb.sp.gov.br/veicular/combustiveis/ > Acesso em 26 de junho de 2022.

EPE. *O que são combustíveis?* Disponível em < https://www.epe.gov.br/pt/abcdenergia/o-que-sao-combustiveis#:~:text=Combust%C3%ADveis%20s%C3%A3o%20subst%C3%A2ncias%20que%20queimamos,para%20acionar%20motores%20de%20ve%C3%ADculos. > Acesso em 26 de junho de 2022

IPEADATA. Disponível em < http://www.ipeadata.gov.br/Default.aspx > Acesso em 24 de junho de 2022.

PETROBRÁS. *Como são formados os preços.* Disponível em < https://precos.petrobras.com.br/web/precos-dos-combustiveis/w/gasolina/rj > Acesso em 26 de junho de 2022.

UDOP. *Combustíveis: O que é o PPI e por que a Petrobras segue preços internacionais?* Disponível em < https://www.udop.com.br/noticia/2021/10/13/combustiveis-o-que-e-o-ppi-e-por-que-a-petrobras-segue-precos-internacionaisy.html > Acesso em 26 de junho de 2022

WEBPOSTO. *Entenda quais fatores levam a alta dos combustíveis.* Disponível em <https://www.webposto.com.br/blog/economia/entenda-quais-fatores-levam-a-alta-dos-combustiveis/ > Acesso em 26 de junho de 2022.

APÊNDICE

Scripts

Nesse trabalho foi utilizado a biblioteca do Plotly para construir os gráficos. Plotly é uma biblioteca de visualização de dados para Python, Javascript e R. Eles tem uma série de produtos, desde para criação de dashboards até clientes SQL. Plotly permite que você utilize seus gráficos em aplicações e, claro, Jupyter Notebooks. A quantidade de visualizações disponíveis, além da gigantesca capacidade de customização me faz querer dar um abraço quentinho em todo mundo que criou essa coisa linda.

```
!pip install plotly
import plotly.offline as py
import plotly.express as px
import plotly.graph_objects as go
from plotly.subplots import make_subplots
```

OBS.: Os gráficos não aparecem no código de forma offline, é preciso rodá-lo por inteiro, para os mesmos aparecerem.