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

Monique Ellen da Silva Acacio

QUEIMADAS FLORESTAIS NO BRASIL

Belo Horizonte
2020

Monique Ellen da Silva Acacio

QUEIMADAS FLORESTAIS 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
2020

SUMÁRIO

1. Introdução

1.1. Contextualização

Todos os anos é muito comum ver notícias sobre queimadas e incêndios florestais em todo mundo, este tipo de ocorrência tem ganhado cada vez mais visibilidade na mídia nacional e internacional. Recentemente temos visto muitos relatos de animais silvestres encontrados em zona urbana, possivelmente um reflexo do impacto das queimadas em seu habitat natural.

O Brasil é formado por seis biomas de características distintas: Amazônia, Caatinga, Cerrado, Mata Atlântica, Pampa e Pantanal. Cada um desses ambientes abriga diferentes tipos de vegetação e de fauna. Como a vegetação é um dos componentes mais importantes da biota, seu estado de conservação e de continuidade definem a existência ou não de hábitats para as espécies, a manutenção de serviços ambientais e o fornecimento de bens essenciais à sobrevivência de populações humanas. Para melhor entendimento da localização destes biomas nos estados brasileiros observe o mapa abaixo.

Para a perpetuação da vida nos biomas, é necessário o estabelecimento de políticas públicas ambientais, a identificação de oportunidades para a conservação, uso sustentável e repartição de benefícios da biodiversidade.

Sabendo da importância dos biomas brasileiros, o objetivo deste projeto é a realização de análise exploratória a fim de identificar a frequência e outros padrões nas queimadas e incêndios florestais que ocorreram no Brasil nos últimos 15 anos, de 2005 a 2019. Desta forma auxiliar as autoridades no direcionamento de recursos para controle e prevenção das queimadas. Os dados utilizados são disponibilizados pelo Instituto Nacional de Pesquisas Espaciais, INEP, através do programa Queimadas.

A coleta dos dados, elaboração e resultado das análises são apresentados no decorrer deste documento.

1.2. O problema proposto

A proposta deste trabalho é a realização de análise exploratória dos focos de queimada pelo país ao longo dos anos. Para melhor visualização do problema foram respondidas as perguntas propostas nos 5-Ws:

Por que esse problema é importante?

Entender a frequência dos incêndios florestais em uma série temporal pode ajudar a tomar medidas para evitá-los, ou direcionar recursos para que os locais mais afetados possam resolver quaisquer incidentes o mais rápido possível. Ser capaz de apontar onde e quando essa frequência é mais observada pode dar alguma clareza sobre qual é a real situação.

Quais serão os dados utilizados e suas fontes?

Os dados utilizados foram disponibilizados pelo Instituto Nacional de Pesquisas Espaciais, INEP, através do Programa Queimadas disponível no portal http://queimadas.dgi.inpe.br/queimadas/portal.

O dataset inclui dados como data, estado, município, bioma e etc para cada foco de queimada identificado por satélite. Os dados foram coletados no DBQueimadas, que é um sistema de monitoramento via satélite disponibilizado no portal do Programa Queimadas.

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

O objetivo desta análise é identificar a frequência das queimadas por estado e por bioma além identificar padrões que determinem épocas do ano com maior número de ocorrências para auxiliar as autoridades no correto direcionamento de recursos para combate e prevenção das queimadas. Também será realizada predição dos focos de queimada ao longo dos anos utilizando o modelo ARIMA.

Quais são aspectos geográficos das análises?

Abrange todos os 27 estados brasileiros, identifica os municípios, latitude e longitude de cada foco de queimada.

Qual o período está sendo analisado?

O dataset possui dados dos últimos 15 anos, de 2005 até 2019. Com registros diários de cada foco de queimada.

2. Coleta de Dados

Este projeto trabalha com um único dataSet obtido através do sistema BDQueimadas, onde são reportados os focos de queimada, conforme detalhamento abaixo.

Os dados foram coletados no dia 22/09 através do Portal Queimadas, acesso através do link http://queimadas.dgi.inpe.br/queimadas/portal. Ao acessar o portal, foi utilizado o link 'Download de dados' dentro da sessão 'SISTEMAS DE MONITORAMENTO > BDQueimadas'.

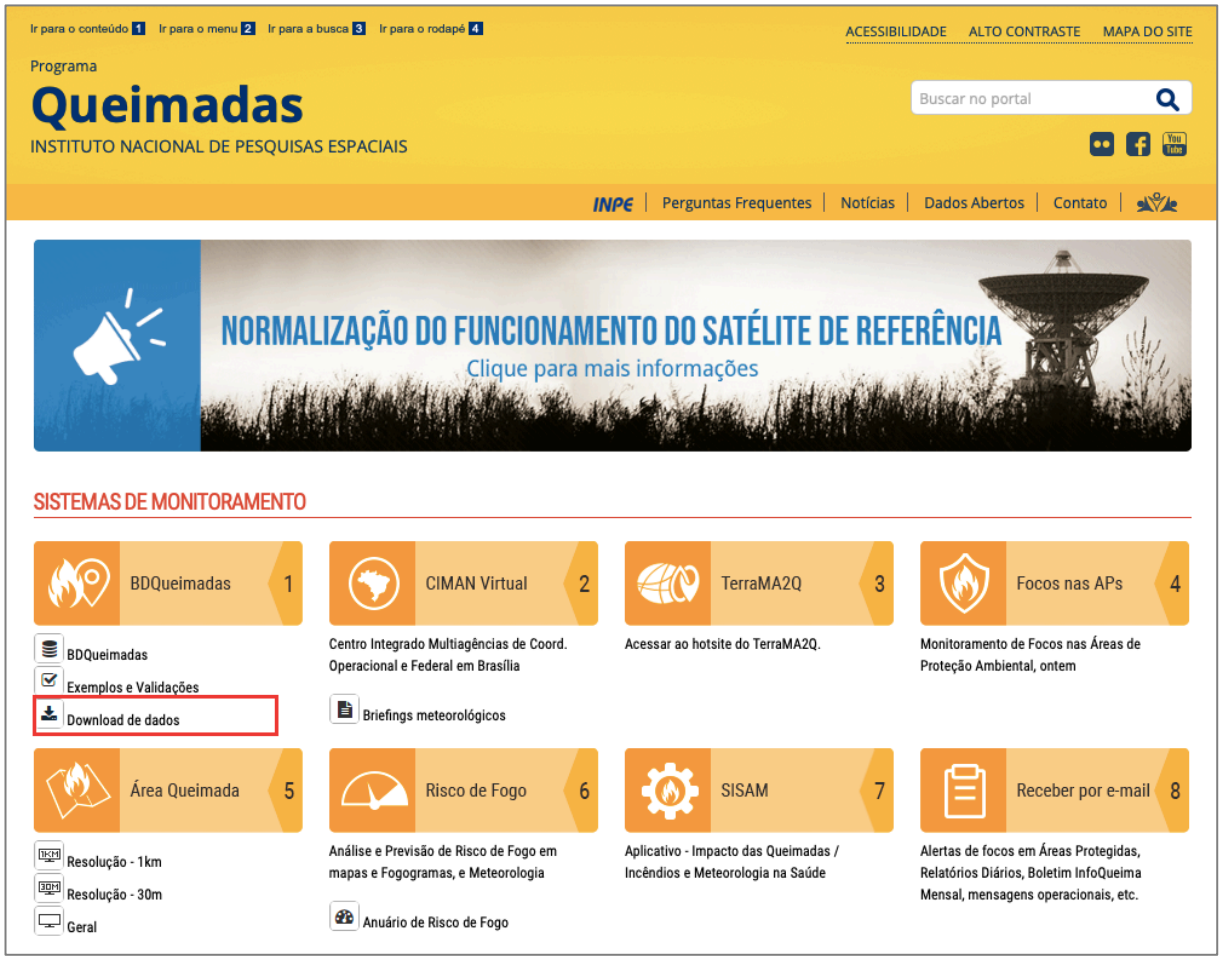


Figura 1 – Link para download do dataset no Portal Queimadas/INEP.

O link mencionado direciona para a funcionalidade de download do sistema BDQueimadas, infelizmente o sistema não permite realizar o download dos dados em período maior que 366 dias. Então, para coletar os dados utilizados neste projeto foram realizadas 15 consultas, uma para cada ano de 2005 até 2019, considerando

data e hora de início 01/01/yyyy 00:00 e data e hora de término 31/12/yyyy 23:59 para cada ano. Outras informações do preenchimento do formulário de download estão listadas abaixo:

- Deve ser informado um email para onde o link de download será enviado.
- Foi selecionado apenas o Brasil e opção ‘Todos os estados’
- Foi informado data e hora de início e término para coleta dos dados, lembrando que o período máximo é de 366 dias.
- Na definição de ‘Focos dos Satélites’ foi selecionado ‘Satélite de referência’
- Na definição de ‘Foco nos Biomas’ foi selecionado ‘TODOS’
- Selecionado o formato de exportação CSV
- Ao clicar em ‘Exportar’ o sistema envia o link de download para o email informado.

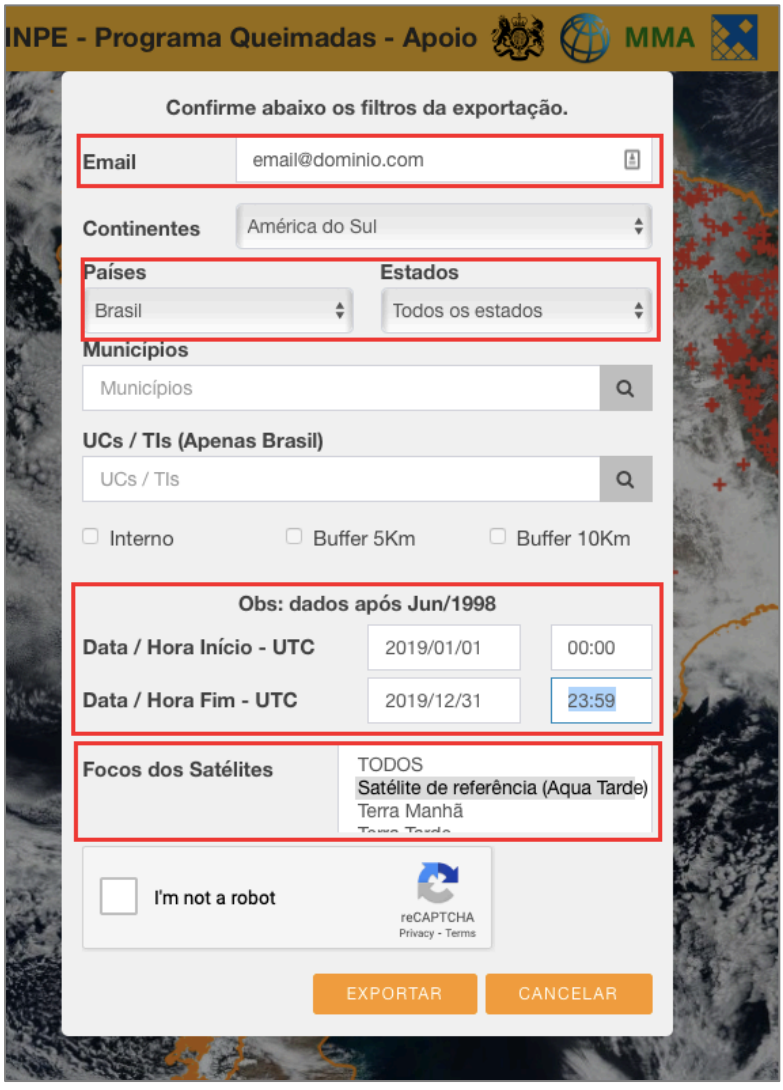


Figura 2 - Formulário de exportação de dados – Sistema DBQueimadas

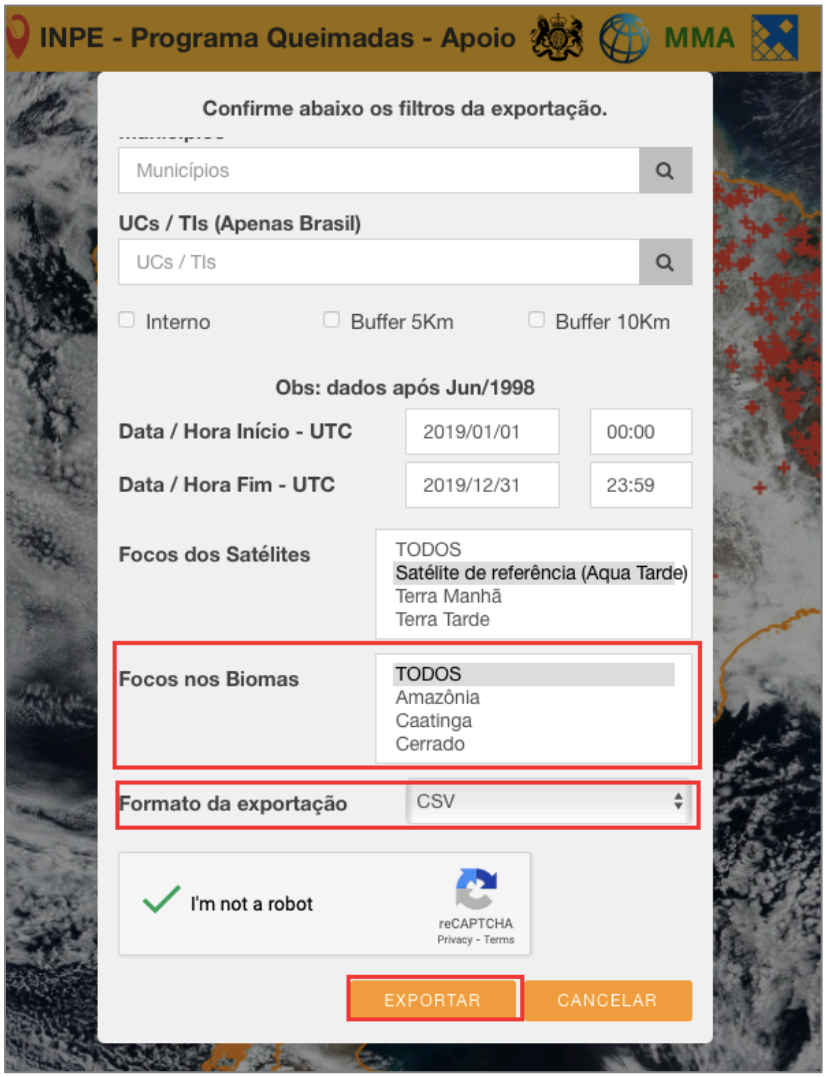


Figura 3 - Formulário de exportação de dados – Sistema BDQueimadas

A exportação dos dados resultou em 15 arquivos CSVs, uma para cada ano, todos com as mesmas estrutura de dados, onde cada linha representa o registro de um foco de queimada. Na etapa de processamento dos dados todos os CVSs são concatenados em um único DataFrame.

O dicionário de dados descrito neste documento foi disponibilizado pelo INEP através do link abaixo em resposta a questão número 40.

http://queimadas.dgi.inpe.br/queimadas/portal/informacoes/perguntas-frequentes

Nome da coluna/campo	Descrição	Tipo (identificação automática)
ID	Identificador único do registro no banco de dados. Formado pela junção dos atributos (Latgms + Longms + Data + hora) removidos os espaços	string
datahora	Horário de referência da passagem do satélite segundo o fuso horário de Greenwich (GMT); https://pt.wikipedia.org/wiki/Greenwich_Mean_Time; Representada em Hora (2 digitos) + Minutos (2 digitos) + Segundos (2 digitos)	string
satelite	Nome do algoritmo utilizado e referencia ao satélite provedor da imagem	string
país	Nome do País (nível 0 do Database of Global Administrative Areas - GADM)	string
estado	Nome do estado (nível 1 do Database of Global Administrative Areas - GADM) http://www.gadm.org	string
municipio	Nome do município. Para o Brasil foi utilizado como referência o dado do IBGE 2000 (http://mapas.ibge.gov.br/bases-e-referenciais/bases-cartograficas/malhas-digitais.html)	string
bioma	Nome do Bioma segundo referência do IBGE 2004 (http://www.ibge.gov.br/home/presidencia/noticias/21052004biomashtml.shtm). Para outros países o campo fica vazio (NULL)	string
diasemchuva	Número de dias sem chuva até a detecção do foco	integer
precipitacao	Valor da precipitação acumulada no dia até o momento da detecção do foco	double
riscofogo	Valor do Risco de Fogo previsto para o dia da detecção do foco	double
latitude	Latitude do centro do píxel de fogo ativo apresentada em unidade de graus decimais.	doble

longitude	Longitude do centro do píxel de fogo ativo apresentada em unidade de graus decimais	doble
frp	Fire Radiative Power, MW (megawatts)	doble

Na etapa de processamento os campos satélite, diasemchuva, precipitação, riscofogo e frp, pais e satelite serão removidos pois estas informações não são relevantes para o contexto das análises deste projeto.

3. Processamento/Tratamento de Dados

Para processamento, tratamento e análise dos dados foi utilizado a linguagem Python, para melhor visualização das etapas foi utilizado Jupyter Notebook. No relato abaixo estão todas as etapas realizadas para processamento e tratamento dos dados, no item 7 deste documento está o link para download do Jupyter Notebook utilizado e os dados coletados.

Importação das bibliotecas

```
#Bibliotecas básicas
import pandas as pd
import numpy as np
import glob
import matplotlib.pyplot as plt
from datetime import datetime
import seaborn as sns
import pylab as p
import math

#Bibliotecas para estatisticas
import descartes
import statsmodels.api as sm
from statsmodels.tsa.stattools import adfuller
from statsmodels.tsa.arima.model import ARIMA
from statsmodels.tsa.seasonal import seasonal_decompose
from statsmodels.graphics.tsaplots import plot_acf, plot_pacf
import pmdarima as pm
import sklearn.metrics
from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_score

#Biblioteca para tratamento de warnings
import warnings
warnings.filterwarnings("ignore")

%matplotlib inline

#Definição do padrão visual de todos os gráficos
sns.set_style('whitegrid')
```

Importação dos dados

Devido a limitação de período durante a exportação dos dados, foram gerados 10 arquivos CSVs sendo uma para cada ano de 2005 até 2019. Como todos os arquivos possuem a mesma estrutura, optei por concatená-los em um único

DataFrame durante o processo de importação. Abaixo é possível observar os 3 primeiros registros após importação.

```
#Importação dos arquivos concatenando em um único data frame

path = 'Dados/'
all_files = glob.glob(path + "/*.csv")

li = []

for filename in all_files:
    df = pd.read_csv(filename, index_col=None, header=0)
    li.append(df)

focos_df = pd.concat(li, axis=0, ignore_index=True)
```

```
#Visualização dos três primeiros registros do DataFrame criado
focos_df.head(3)
```

	datahora	satelite	pais	estado	municipio	bioma	diasemchuva	precipitacao	riscofogo	latitude	longitude	frp
0	2017/01/05 16:18:00	AQUA_M-T	Brasil	MINAS GERAIS	OURO PRETO	Mata Atlantica	0.0	0.9	0.7	-20.608	-43.510	NaN
1	2017/01/08 16:45:00	AQUA_M-T	Brasil	RIO GRANDE DO SUL	SANTA VITORIA DO PALMAR	Pampa	0.0	3.5	-999.0	-33.317	-52.848	NaN
2	2017/01/08 16:48:00	AQUA_M-T	Brasil	PARANA	CAMPO MOURAO	Mata Atlantica	0.0	2.8	-999.0	-24.049	-52.395	NaN

O dataset possui o total de 3.310.492 registros.

```
focos_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3310492 entries, 0 to 3310491
Data columns (total 12 columns):
 #   Column        Dtype
---  ------        -----
 0   datahora      object
 1   satelite      object
 2   pais          object
 3   estado        object
 4   municipio     object
 5   bioma         object
 6   diasemchuva   float64
 7   precipitacao  float64
 8   riscofogo     float64
 9   latitude      float64
 10  longitude     float64
 11  frp           float64
dtypes: float64(6), object(6)
memory usage: 303.1+ MB
```

Verificação dos valores únicos.

```
print('Pais: ', focos_df.pais.unique(), '\n')
print('Satelite: ', focos_df.satelite.unique(), '\n')
print('Estados: ', focos_df.estado.unique(), '\n')
print('Bioma: ', focos_df.bioma.unique(), '\n')
```

```
Pais:  ['Brasil']

Satelite:  ['AQUA_M-T']

Estados:  ['MINAS GERAIS' 'RIO GRANDE DO SUL' 'PARANA' 'SAO PAULO' 'PIAUI'
 'MARANHAO' 'CEARA' 'MATO GROSSO' 'SERGIPE' 'GOIAS' 'RIO GRANDE DO NORTE'
 'PARA' 'AMAZONAS' 'BAHIA' 'MATO GROSSO DO SUL' 'PERNAMBUCO' 'TOCANTINS'
 'SANTA CATARINA' 'PARAIBA' 'ALAGOAS' 'RORAIMA' 'RONDONIA'
 'ESPIRITO SANTO' 'AMAPA' 'RIO DE JANEIRO' 'DISTRITO FEDERAL' 'ACRE']

Bioma:  ['Mata Atlantica' 'Pampa' 'Caatinga' 'Cerrado' 'Amazonia' 'Pantanal' nan]
```

Remoção das colunas sem relevância para análise.

Coluna	Motivo do descarte
pais	neste dataset apenas o Brasil é representado
satelite	neste dataset todos os dados foram coletados do mesmo satélite conforme orientação duante a coletado de dados logo não é necessário manter esta coluna
diasemchuva	nas análises realizadas será considerado apenas a frequencia dos focos, neste caso os dias sem chuva não serão relevantes
precipitacao	nas análises realizadas será considerado apenas a frequencia dos focos, neste caso o valor da preciptação aculada até o momento da detecção do fogo não é relevante
riscofogo	nas análises realizadas será considerado apenas a frequencia dos focos, o valor de risco de fogo é calculado pelo INEP e não é relevante no contexto desta análise
frp	nas análises realizadas será considerado apenas a frequencia dos focos, o Fire Radioative Power não é relevante no contexto desta análise

```
#Remoção das colunas
focos_df=focos_df.drop(columns=['pais','satelite','diasemchuva', 'precipitacao', 'riscofogo','frp'])
focos_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3310492 entries, 0 to 3310491
Data columns (total 6 columns):
 #   Column     Dtype
---  ------     -----
 0   datahora   object
 1   estado     object
 2   municipio  object
 3   bioma      object
 4   latitude   float64
 5   longitude  float64
dtypes: float64(2), object(4)
memory usage: 151.5+ MB
```

Ajuste do tipo da coluna ‘datahora’ para datetime.

```
#Alterando o dtype
focos_df['datahora'] = pd.to_datetime(focos_df['datahora'], format='%Y/%m/%d %H:%M:%S')
focos_df.dtypes
```

```
datahora     datetime64[ns]
estado               object
municipio            object
bioma                object
latitude            float64
longitude           float64
dtype: object
```

Como o objetivo é analisar a frequência da ocorrência dos focos de queimada, optei por adicionar duas colunas sendo uma para registro do ano e outro do mês da ocorrência.

```
#Criando as colunas ano e mes
focos_df['ano']=pd.DatetimeIndex(focos_df['datahora']).year
focos_df['mes']=pd.DatetimeIndex(focos_df['datahora']).month_name()

#Ajustando os meses para português
mes={'January': 'Janeiro', 'February': 'Fevereiro', 'March': 'Março', 'April': 'Abril', 'May': 'Maio',
          'June': 'Junho', 'July': 'Julho', 'August': 'Agosto', 'September': 'Setembro', 'October': 'Outubro',
          'November': 'Novembro', 'December': 'Dezembro'}
focos_df['mes']=focos_df['mes'].map(mes)
#Verificando o ajuste
focos_df.mes.unique()
```

```
array(['Janeiro', 'Fevereiro', 'Abril', 'Maio', 'Junho', 'Julho',
       'Agosto', 'Setembro', 'Novembro', 'Dezembro', 'Março', 'Outubro'],
      dtype=object)
```

```
#Verificação do dataFrame
focos_df.head(5)
```

	datahora	estado	municipio	bioma	latitude	longitude	ano	mes
0	2017-01-05 16:18:00	MINAS GERAIS	OURO PRETO	Mata Atlantica	-20.608	-43.510	2017	Janeiro
1	2017-01-08 16:45:00	RIO GRANDE DO SUL	SANTA VITORIA DO PALMAR	Pampa	-33.317	-52.848	2017	Janeiro
2	2017-01-08 16:48:00	PARANA	CAMPO MOURAO	Mata Atlantica	-24.049	-52.395	2017	Janeiro
3	2017-01-08 16:48:00	SAO PAULO	SAO SEBASTIAO	Mata Atlantica	-23.764	-45.414	2017	Janeiro
4	2017-01-12 16:29:00	PIAUI	CARAUBAS DO PIAUI	Caatinga	-3.473	-41.709	2017	Janeiro

Verificação de valores nulos.

```
#Verificação do total de valores nulos em cada  coluna
focos_df.isna().sum()
```

```
datahora     0
estado       0
municipio    0
bioma        1
latitude     0
longitude    0
ano          0
mes          0
dtype: int64
```

Como o dataset apresenta apenas um registro nulo optei por deletar o registro.

```
#Remoção do registro que possui o bioma nulo
focos_df=focos_df.dropna()

#Verificação dos valores nulos
focos_df.isna().sum()
```

```
datahora     0
estado       0
municipio    0
bioma        0
latitude     0
longitude    0
ano          0
mes          0
dtype: int64
```

Após os tratamentos aplicados o tratamento final possui 3.310.491 registros e as configurações apresentadas abaixo.

```
focos_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 3310491 entries, 0 to 3310491
Data columns (total 8 columns):
 #   Column     Dtype
---  ------     -----
 0   datahora   datetime64[ns]
 1   estado     object
 2   municipio  object
 3   bioma      object
 4   latitude   float64
 5   longitude  float64
 6   ano        int64
 7   mes        object
dtypes: datetime64[ns](1), float64(2), int64(1), object(4)
memory usage: 227.3+ MB
```

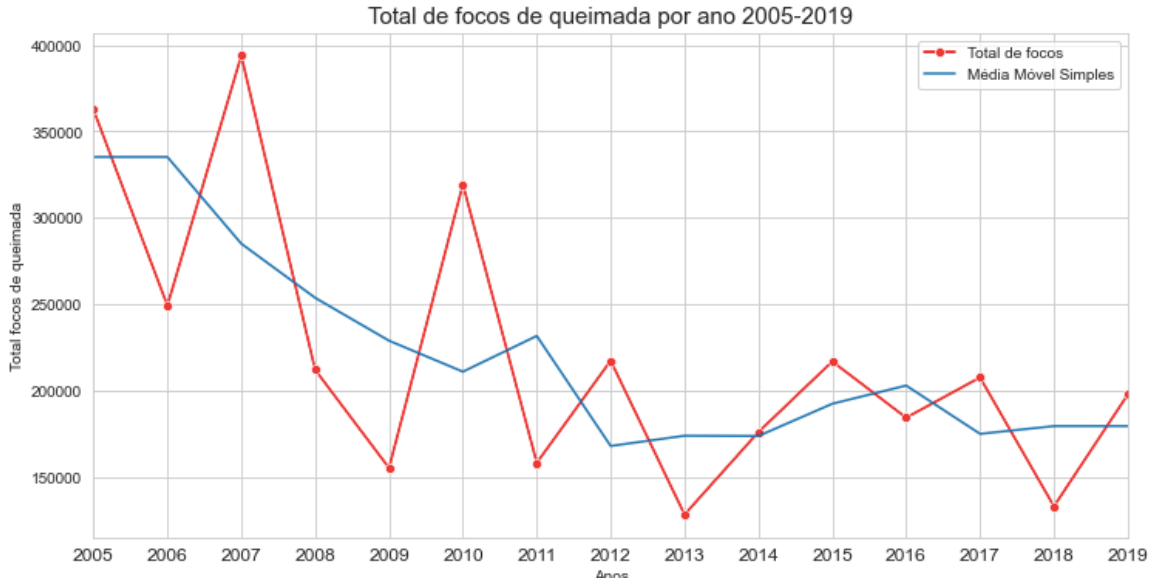
4. Análise e Exploração dos Dados

Para guiar a análise e exploração dos dados foram respondidos os questionamentos apresentados abaixo:

Qual o total de focos de queimada ao longo dos últimos 15 anos?

Para esta verificação os dados foram agrupados por ano, contabilizando o total de registros de cada ano. Como forma de verificação de tendência foi calculado a média móvel ao longo da série.

```
#Preparação dos dados
focos_ano=pd.DataFrame(focos_df.groupby(['ano'])['mes'].count()).reset_index()
focos_ano.rename(columns={'mes': 'total'}, inplace = True)
print(focos_ano)

#Calculo da média móvel simples de queimadas ao longo dos anos
ano=focos_ano.set_index('ano')
inicio = 2006
fim = 2019
mms = []

i = inicio
while i < fim:
    y=i
    w=i
    y -=1
    w +=1
    ano_anterior = ano.total.loc[y]
    ano_atual = ano.total.loc[i]
    ano_proximo = ano.total.loc[w]
    media = (ano_anterior + ano_atual + ano_proximo) / 3

    mms.append(media)
    i += 1


#Não é possível calcular a MMS para os anos extremos, 2005 e 2019. Sendo assim, foram adicionados os dados mais próx
mms.insert(0, mms[0])
mms.append(mms[-1])
mms = pd.Series(mms)

#Criando o gráfico
plt.figure(figsize=(12,6))
sns.lineplot(x='ano', y='total', data=focos_ano, marker='o', linestyle='-', color='red',label='Total de focos' )
sns.lineplot(x='ano', y=mms, data=focos_ano, label='Média Móvel Simples')
plt.ylabel('Total focos de queimada')
plt.xlabel('Anos')
plt.xlim(2005,2019)
plt.xticks(np.arange(2005, 2020, 1),fontsize=12)
plt.title('Total de focos de queimada por ano 2005-2019', fontsize=15)
plt.grid(True)

plt.legend()
plt.show()
```

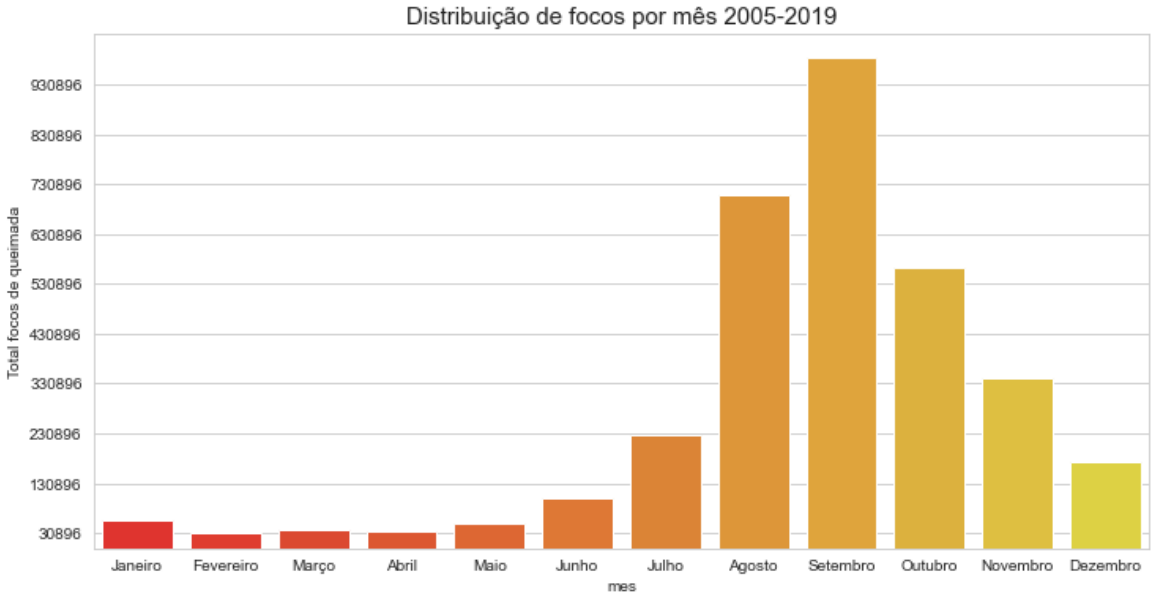


No geral, até 2019, a quantidade de incêndios florestais desde 2005 diminuiu ao longo dos anos. Houve um período crítico entre 2005 e 2010, com grande quantidade de focos de queimada, em especial em 2007 com o total de 207511.

Quais são os meses com maior incidência de focos de incêndio?

Para esta verificação os dados foram agrupados por mês, contabilizando o total de registros independente do ano.

```
#Preparação dos dados
ordem=['Janeiro','Fevereiro','Março','Abril','Maio','Junho','Julho','Agosto','Setembro','Outubro','Novembro','Dezemb
focos_mes=pd.DataFrame(focos_df.groupby('mes')['ano'].count()).reset_index()
focos_mes.rename(columns={'ano': 'total'}, inplace = True)
focos_mes['mes']=pd.Categorical(focos_mes['mes'], categories=ordem, ordered=True)
focos_mes.sort_values('mes', inplace=True)


#Criando o gráfico
plt.figure(figsize=(12,6))
plt.yticks(np.arange(focos_mes.total.min(), focos_mes.total.max(), 100000))
sns.barplot(data=focos_mes, x='mes', y='total', palette='autumn')
plt.title('Distribuição de focos por mês 2005-2019', fontsize=15)
plt.ylabel('Total focos de queimada')
plt.axis(True)
plt.show()
```



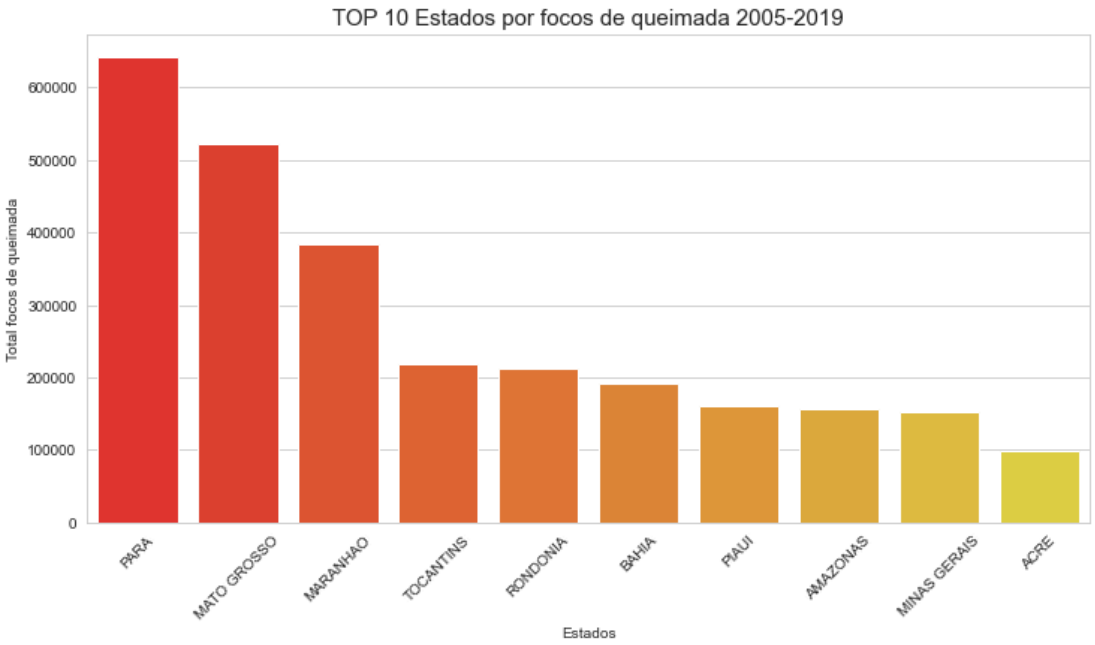
É possível observar que o segundo semestre apresenta um aumento no número de focos em relação ao primeiro semestre. Contudo, este aumento é esperado considerando o período de seca na maior parte dos biomas.

De maneira geral, os meses de agosto, setembro e outubro merecem mais atenção na prevenção de incêndios florestais.

Quais são os 10 estados mais afetados?

Para esta verificação os dados foram agrupados por estado, contabilizando o total de registros independente do ano. Ao final foram selecionados os 10 estados com maior número de focos de queimada.

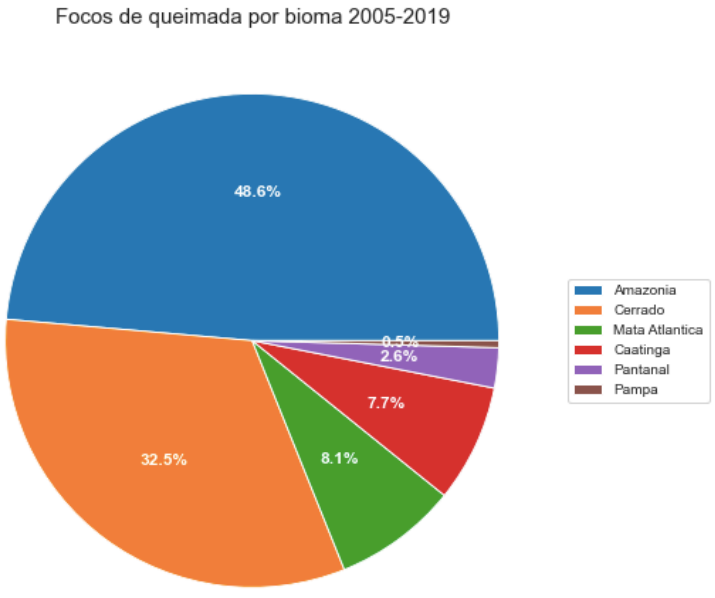
```
#Preparação dos dados
focos_estado=pd.DataFrame(focos_df.groupby(['estado'])['mes'].count()).reset_index()
focos_estado.rename(columns={'mes': 'total'}, inplace = True)
focos_estado=focos_estado.nlargest(10, columns='total')
print(focos_estado)

#Criando o gráfico
plt.figure(figsize=(12,6))

sns.barplot(data=focos_estado, x='estado', y='total', palette='autumn')
plt.xticks(rotation=45)
plt.ylabel('Total focos de queimada')
plt.xlabel('Estados')
plt.title('TOP 10 Estados por focos de queimada 2005-2019', fontsize=15)
plt.show()
```



O Pará, Mato Grosso e Maranhã lideram como os Estados com maior número de queimadas.

Focos de queima por bioma ao longo dos anos.

```
#Preparação dos dados
focos_bioma=pd.DataFrame(focos_df.groupby(['bioma'])['mes'].count()).reset_index()
focos_bioma.rename(columns={'mes': 'total'}, inplace = True)
focos_bioma=focos_bioma.sort_values('total', ascending=False)

#Criando o gráfico
plt.figure(figsize=(12,8))
plt.pie(focos_bioma['total'], autopct='%1.1f%%', textprops=dict(color="w", weight="bold", size=12))
plt.xticks(rotation=45)
plt.title('Focos de queimada por bioma 2005-2019', fontsize=15)
plt.legend(focos_bioma['bioma'],
          loc="center left",
          bbox_to_anchor=(1, 0, 0.5, 1))
plt.show()
```

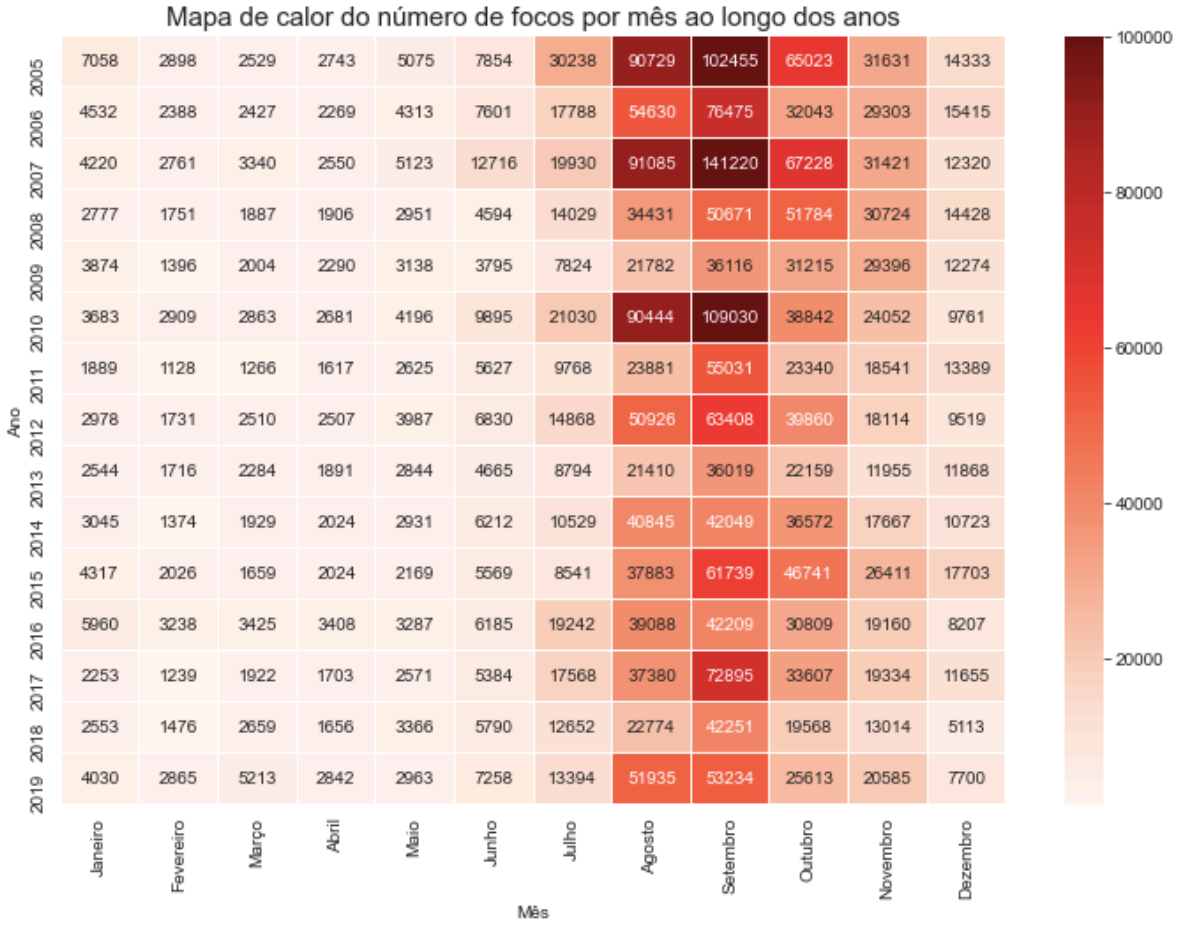


Praticamente metade dos focos de queimada que ocorreram de 2005 até 20019 foram na Amazônia, o que corresponde com o alto número de queimadas nos estados do Pará e do Mato Grosso.

Mapas de calor (Heatmap)

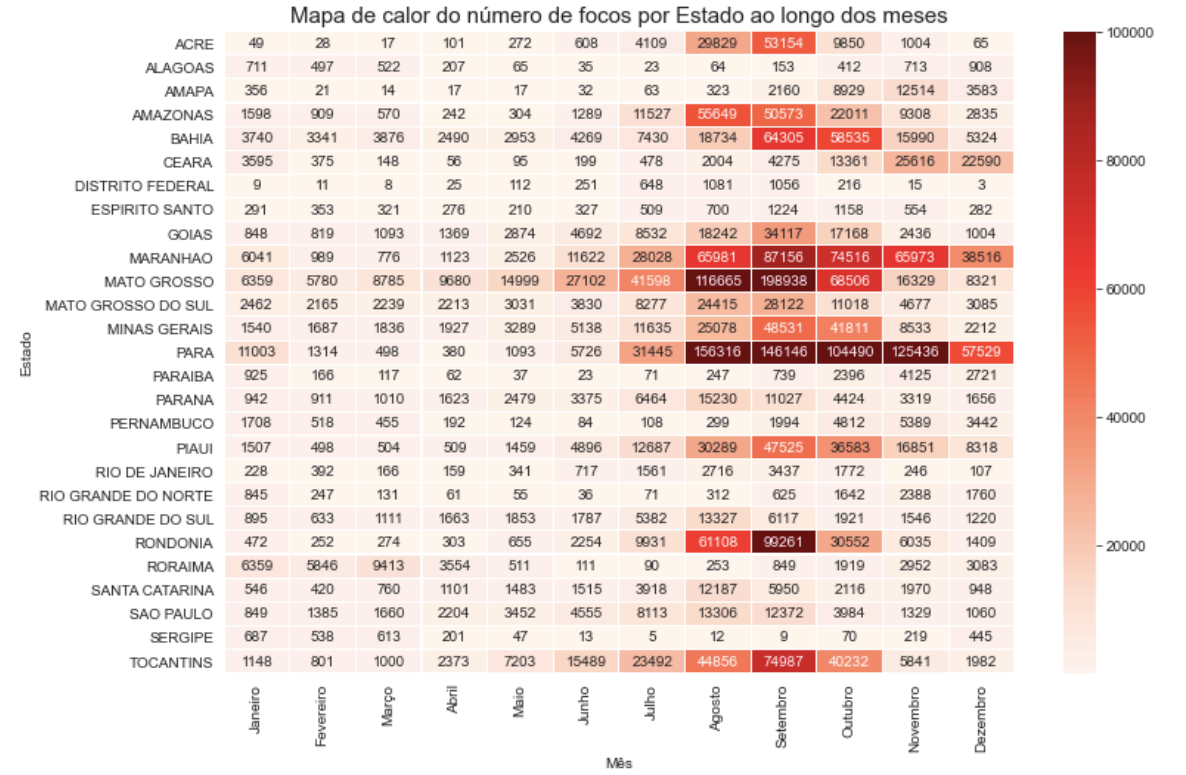
- Total de focos por mês ao longo dos anos

Observando os meses ao longo dos anos, os meses de Agosto e Setembro como os mais críticos com um total de focos de queimada bem elevado em comparação com os demais meses.



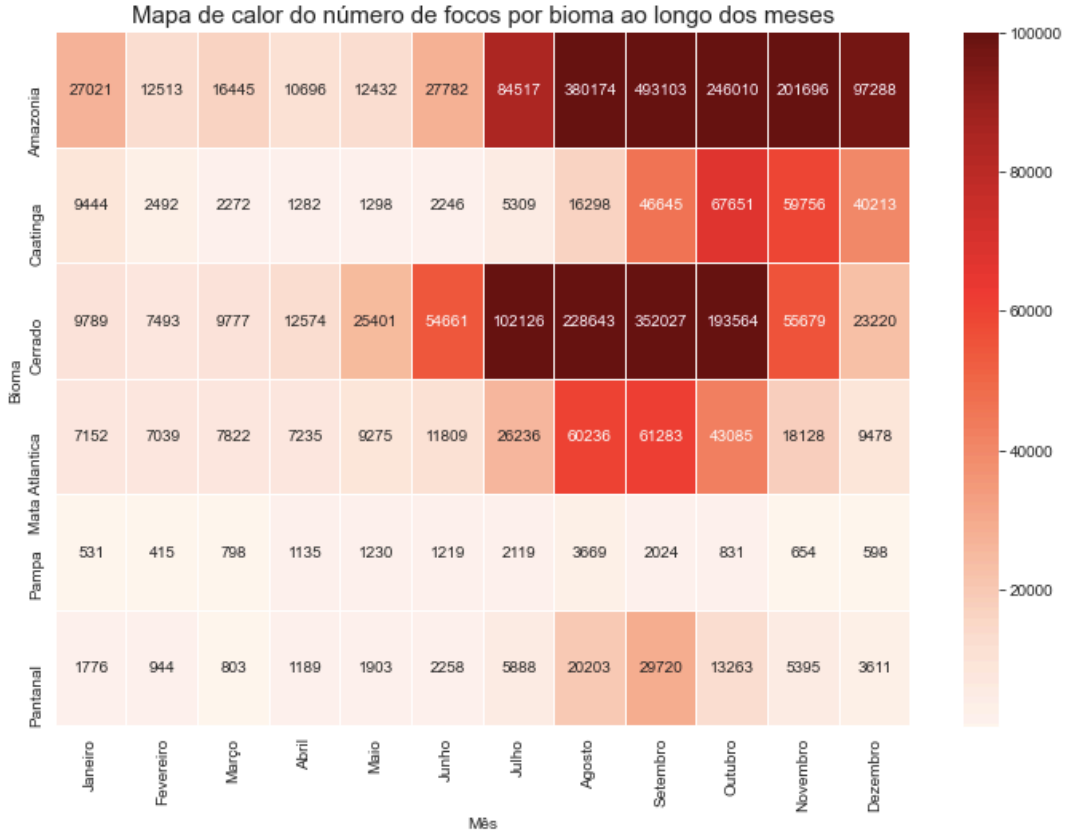
- Total de focos por estado ao longo dos meses

Observando o total de queimadas por estado ao longo dos meses é possível verificar que o estado do Pará e Maranhão apresenta um período crítico de queimadas maior que os outros estados, estendendo o alto número de queimadas até o mês de dezembro.



- Total de focos por bioma ao longo dos meses

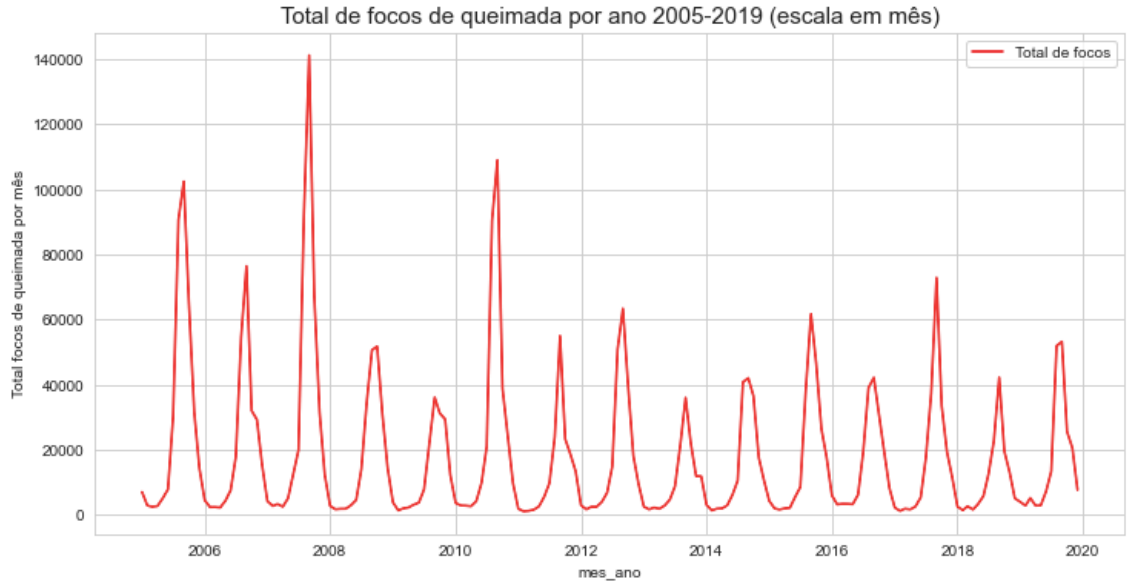
Na visualização do total de queimadas por bioma ao longo dos meses fica ainda mais evidente o alto número de queimadas na Amazônia e no Cerrado.



Predição de focos de queima utilizando modelo ARIMA

Para a serie apresentada optei por aplicar o modelo de predição ARIMA por ser o modelo apresentado no curso para séries temporais.

Para criação do modelo foi considerada a série com o somatório mensal de focos de queimadas ao longo dos anos, conforme representação gráfica abaixo.



Divisão do dataset

Optei por dividir o conjunto de dados em duas partes, uma com dados referentes a 13 anos (2005-2017) e outra com dados referentes a 2 anos (2018-2019). A primeira parte é o conjunto de dados de treinamento que usaremos para preparar um modelo ARIMA. A segunda parte é o conjunto de dados de teste que fingiremos não estar disponível, são essas etapas de tempo que trataremos como fora da amostra.

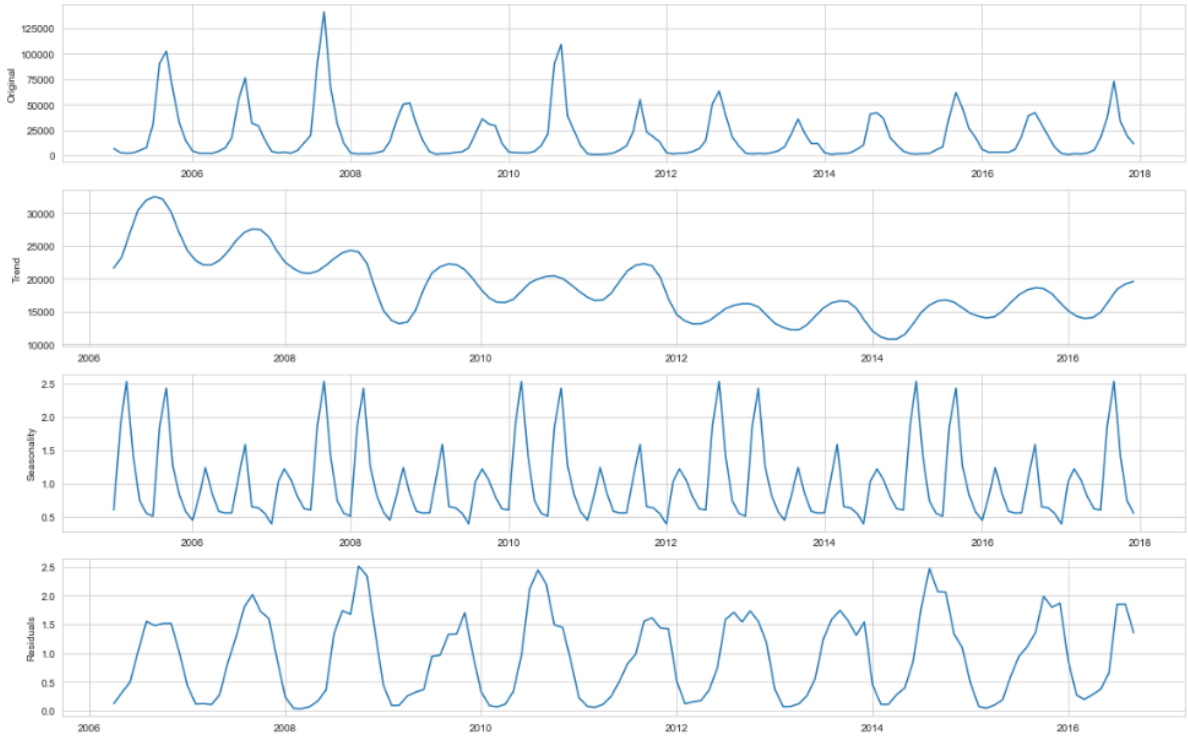
```
# dividindo o dataset
div=len(serie_ori) - 24 #24 meses = 2 anos
serie_treino, serie_teste = serie_ori[0:div], serie_ori[div:]
print('Serie Treino %d, Serie Teste %d' % (len(serie_treino), len(serie_teste)))
serie_treino.set_index('mes_ano', inplace=True)
serie_teste.set_index('mes_ano', inplace=True)
serie_treino.head()
```

```
Serie Treino 156, Serie Teste 24
```

mes_ano	total
2005-01-01	7058
2005-02-01	2898
2005-03-01	2529
2005-04-01	2743
2005-05-01	5075

Decomposição da serie

A decomposição é usada principalmente para análise de séries temporais e, como uma ferramenta de análise, pode ser usada para informar os modelos de previsão sobre o seu problema.



Podemos ver que as informações de tendência e sazonalidade extraídas da série parecem razoáveis, confirmando as observações realizadas durante a análise dos dados. Para dar continuidade a análise será verificado se a série é estacionária.

Estacionariedade

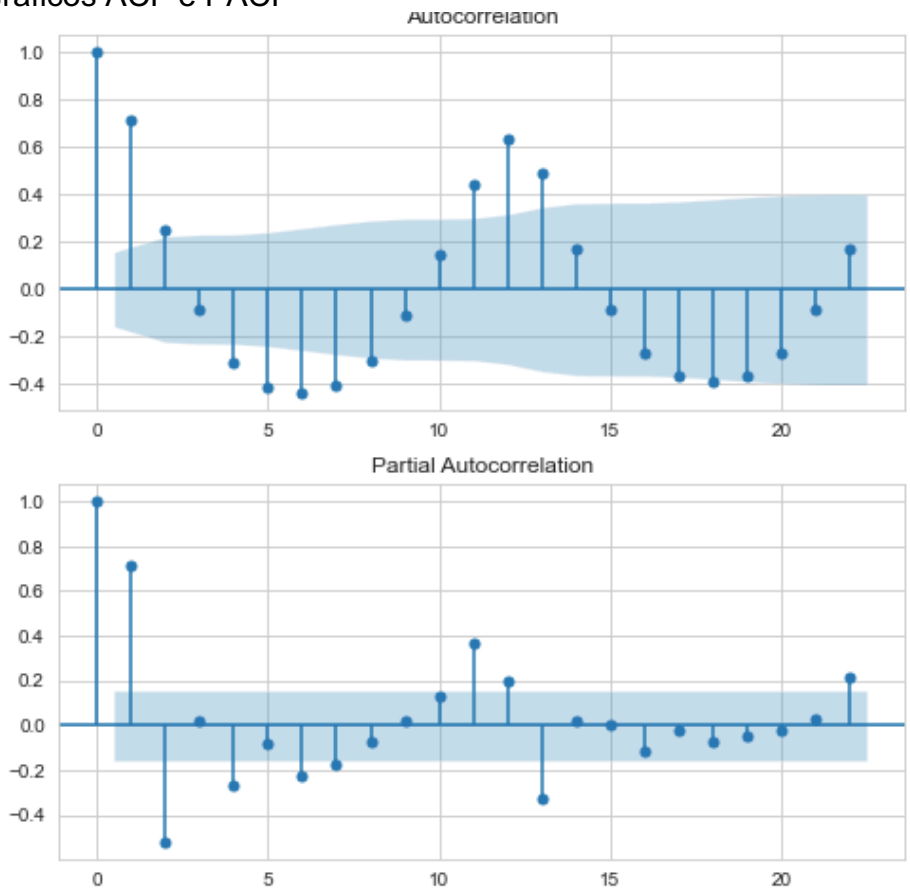
O teste ADF é um tipo de teste estatístico denominado teste de raiz unitária. A intuição por trás de um teste de raiz unitária é que ele determina a intensidade com que uma série temporal é definida por uma tendência.

Interpretamos esse resultado usando o valor p do teste. Um valor de p abaixo de um limite (como 5% ou 1%) sugere que rejeitamos a hipótese nula (estacionário), caso contrário, um valor de p acima do limite sugere que falhamos em rejeitar a hipótese nula (não estacionário).

```
Resultado do Teste Dickey-Fuller:
Teste                      -2.978947
Valor p                     0.036908
Nº de lags                 12.000000
Nº de observações         143.000000
Valor Crítico (1%)         -3.476927
Valor Crítico (5%)         -2.881973
Valor Crítico (10%)        -2.577665
dtype: float64
```

Com o valor de p abaixo de 5% no resultado do teste Dickey_Fuler, conclui-se que a séria é estacionária.

Gráficos ACF e PACF



Treinando o modelo

Para a serie estacionária sem necessidade de diferenciação (d=0), será utilizada a função auto_arima para determinar o melhor modelo que se adequa à série.

p = número de time lags do modelo auto-regressivo (AR)
q = ordem do modelo de média-móvel (MA)
d = grau de diferenciação
P = refere-se ao termo auto-regressivo para a parte sazonal
Q = refere-se ao termo de diferenciação para a parte sazonal
D = refere-se ao termo da média-móvel para a parte sazonal

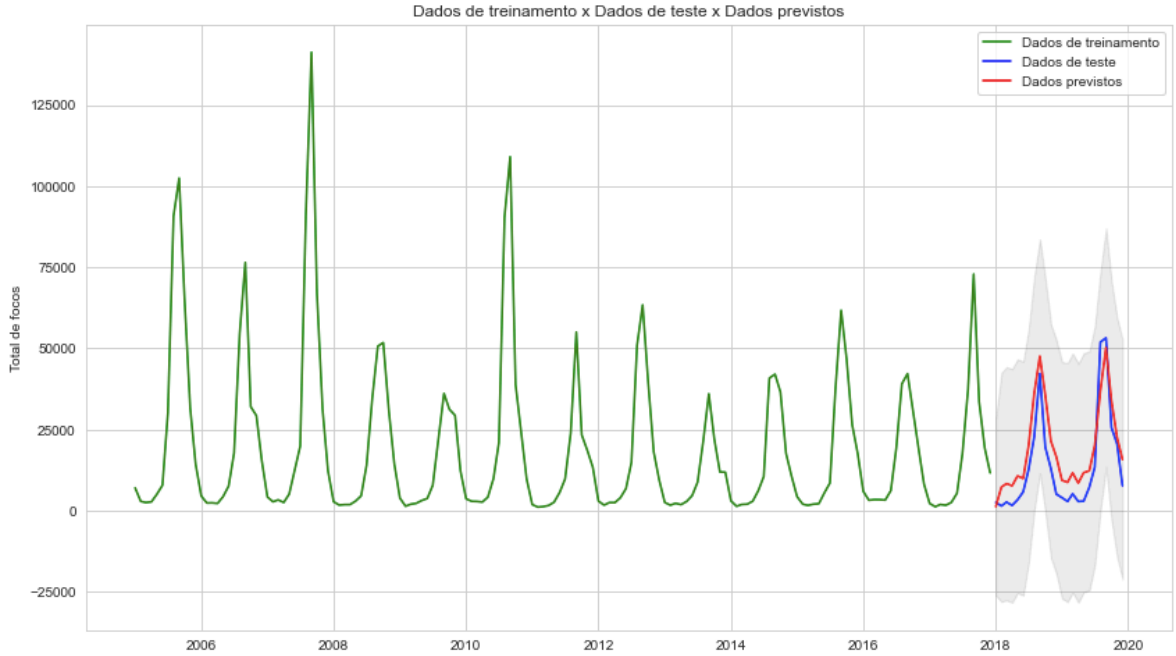
```
#Modelo
modelo = pm.auto_arima(serie_treino['total'], start_p=1, start_q=1,
                       max_p=3, max_q=3,
                       m=12,              #frequencia 12 meses
                       d=0,
                       stationary = True,
                       seasonal=True, #Sazonalidade
                       start_P=0,
                       D=1,
                       trace=True, #reporta a lista de modelos ARIMA considerados.
                       error_action='ignore',
                       suppress_warnings=True,
                       stepwise=True)

#Treinando o modelo
modelo.fit(serie_treino['total'])
```

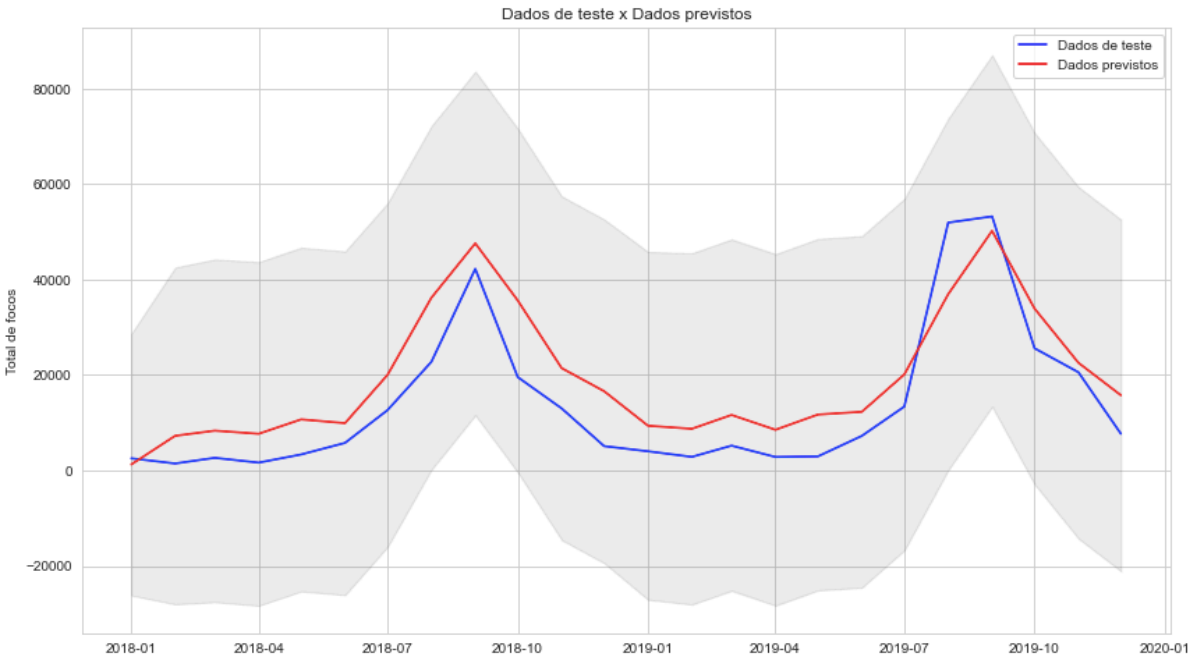
```
print(modelo.summary())
```

```
                                     SARIMAX Results
==========================================================================================
Dep. Variable:                                  y   No. Observations:                  156
Model:             SARIMAX(3, 0, 3)x(1, 0, [1, 2], 12)   Log Likelihood               -1683.281
Date:                                 Wed, 30 Sep 2020   AIC                            3388.562
Time:                                         21:34:09   BIC                            3422.111
Sample:                                              0   HQIC                           3402.188
                                                   - 156
Covariance Type:                                   opg
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
intercept   9753.6786   7786.898      1.253      0.210   -5508.361     2.5e+04
ar.L1         -1.2906      0.209     -6.165      0.000      -1.701      -0.880
ar.L2         -0.4693      0.317     -1.479      0.139      -1.091       0.153
ar.L3          0.3163      0.172      1.834      0.067      -0.022       0.654
ma.L1          2.1002      0.254      8.270      0.000       1.602       2.598
ma.L2          1.7577      0.445      3.954      0.000       0.886       2.629
ma.L3          0.4743      0.246      1.928      0.054      -0.008       0.957
ar.S.L12       0.8619      0.071     12.171      0.000       0.723       1.001
ma.S.L12      -0.6466      0.113     -5.708      0.000      -0.869      -0.425
ma.S.L24       0.2099      0.133      1.573      0.116      -0.052       0.472
sigma2      1.956e+08      0.229   8.56e+08      0.000    1.96e+08    1.96e+08
===================================================================================
Ljung-Box (Q):                       32.75   Jarque-Bera (JB):               291.55
Prob(Q):                              0.78   Prob(JB):                         0.00
Heteroskedasticity (H):               0.34   Skew:                             1.60
Prob(H) (two-sided):                  0.00   Kurtosis:                         8.88
===================================================================================

Warnings:
[1] Covariance matrix calculated using the outer product of gradients (complex-step).
[2] Covariance matrix is singular or near-singular, with condition number 1.82e+25. Standard errors may be unstable
.
```

Predição e apresentação

Os valores preditos ficam bem próximos dos valores de teste.





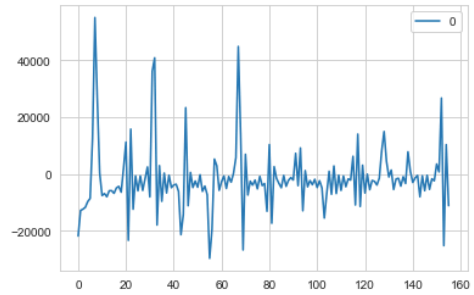
Outros dados

```
#Cálculo do erro
mse = mean_squared_error(serie_teste['total'], prev_arima['Prediction'])
print('MSE: '+str(mse))
mae = mean_absolute_error(serie_teste['total'], prev_arima['Prediction'])
print('MAE: '+str(mae))
rmse = math.sqrt(mean_squared_error(serie_teste['total'], prev_arima['Prediction']))
print('RMSE: '+str(rmse))
```

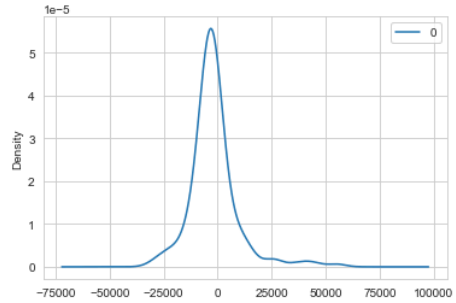
```
MSE: 64866680.45986098
MAE: 7197.89906904739
RMSE: 8053.985377430293
```

```
#Plotagem dos resíduos
residuals = pd.DataFrame(modelo.resid())
residuals.plot()
```

```
<AxesSubplot:>
```




```
#Plotagem da densidade dos resíduos
residuals.plot(kind='kde')
plt.show()
print(residuals.describe())
```



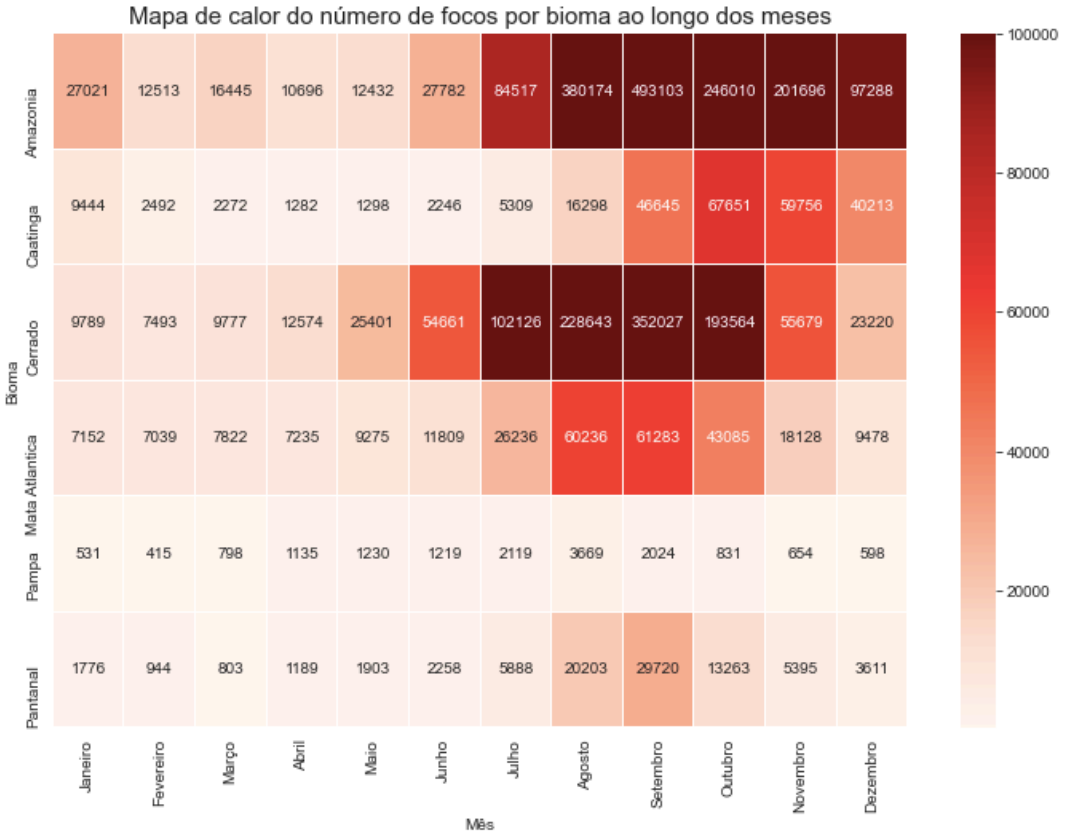

```
                0
count     156.000000
mean    -1746.896731
std     11349.595775
min    -29721.846720
25%     -6107.206813
50%     -2646.277979
75%       324.876313
max     55106.683694
```

5. Apresentação dos Resultados

Após realizadas as análises foram adquiridos alguns entendimentos sobre a frequência das queimadas florestais no Brasil, vejamos abaixo:

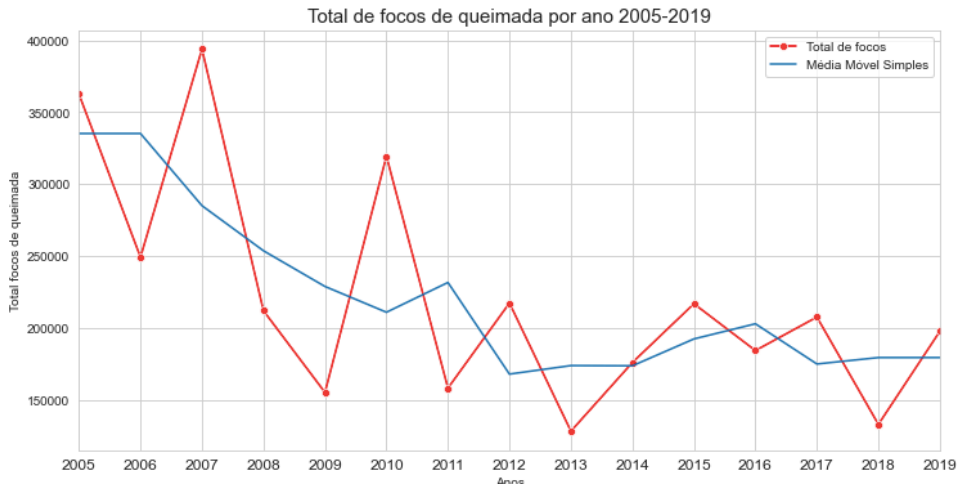
- Historicamente a Amazônia é o bioma brasileiro mais afetado pelas queimadas. Em especial nos meses de seca, o segundo semestre do ano, mas também é possível

notar mesmo nos meses fora do período de seca o número de queimas ainda é mais elevado em comparação com outros biomas.

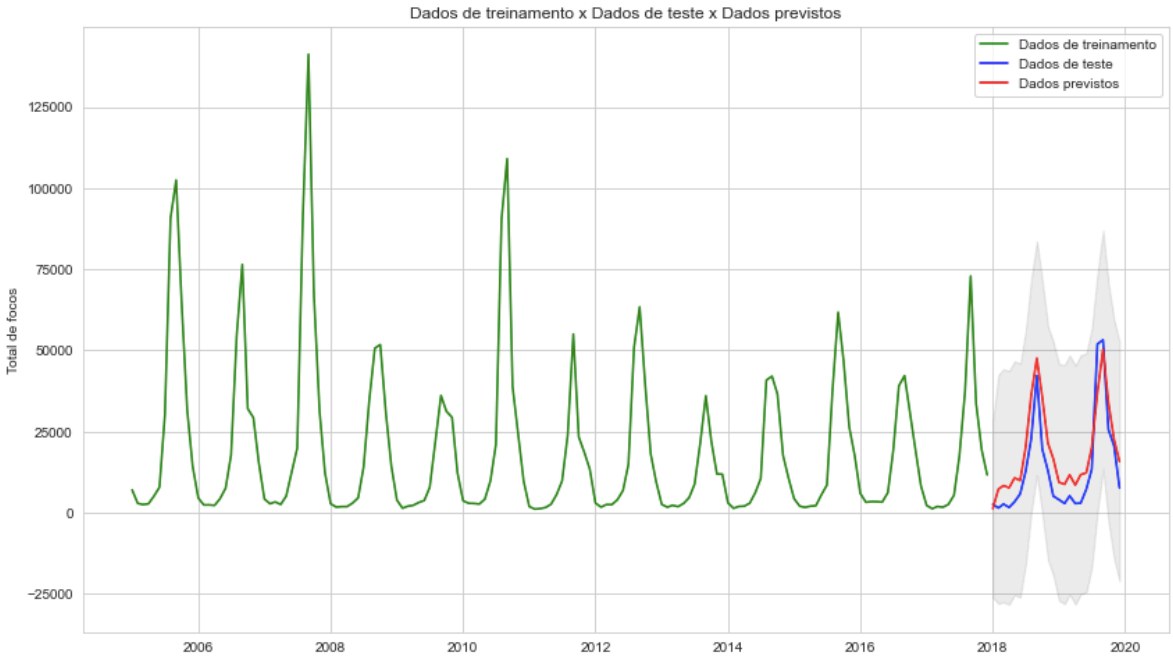


- O Pará e o Mato Grosso lideram a lista dos estados com maior número de focos de queimada, esses dois estados são os maiores causadores nos altos números de queimada na Amazônia.

- Apesar dos números preocupantes o número de focos de queima ao longo dos anos vem caindo consideravelmente. Na série analise o maior pico ocorreu em 2007, ou seja, há 12 anos atrás.



- O modelo preditivo apresentou predição muito próxima da amostra, e com maior detalhamento poderia ser utilizado para prever as queimadas ao longo dos anos.



6. Links

Todos os dados, o Jupyter Notebook utilizado e também este documento estão disponíveis no GitHub através do link https://github.com/Msacacio21/TCC.

O vídeo de apresentação foi disponibilizado no Youtube através do link https://youtu.be/G3MG2HfNa8M .