

Análise perfil abstenção e comparecimento 1º turno - Eleições 2022

Objetivos

O projeto tem como objetivo identificar quais são os tipos de perfis dos eleitores que abstiveram nas eleições de 2022, período dessa análise ocorre somente com dados referente ao primeiro turno.

Os perfis avaliados serão por gênero, idade e/ou faixa etária, nível de escolaridade e quais regiões esses eleitores pertencem.

Especificação Técnica

A base de dados utilizados estão disponíveis no site <https://dadosabertos.tse.jus.br>.

Link do arquivo:

https://cdn.tse.jus.br/estatistica/sead/odsele/perfil_comparecimento_abstencao/perfil_comparecimento_abstencao_2022.zip

Essa base de dados está em formato CSV

Preparação dos dados

Importação das principais para tratamento de dados

```
1 import pandas as pd
2 import numpy as np
```

Import da base de dados

```
1 tb_eleicoes = pd.read_csv("dataset\perfil_comparecimento_abstencao_2022
```

Filtro de colunas para trazer somente os dados necessários

```
1 tb_eleicoes = tb_eleicoes[['SG_UF', 'NM_MUNICIPIO', 'DS_GENERO', 'CD_GENE
2 'DS_GRAU_ESCOLARIDADE', 'QT_APTOS', 'QT_COMPARECIMENTO', 'QT_AB
```

▼ Criando nova coluna, a partir de dados existentes

Aqui foi utilizado a função Where, para criar uma condição em que selecionasse o estado e identificasse qual região pertence.

```

1 tb_eleicoes['Regiao'] = np.where(tb_eleicoes['SG_UF']=='ZZ','Exterior',
2     np.where(tb_eleicoes['SG_UF']=='SC','Sul',
3     np.where(tb_eleicoes['SG_UF']=='PR','Sul',
4     np.where(tb_eleicoes['SG_UF']=='RS','Sul',
5     np.where(tb_eleicoes['SG_UF']=='ES','Sudeste',
6     np.where(tb_eleicoes['SG_UF']=='MG','Sudeste',
7     np.where(tb_eleicoes['SG_UF']=='RJ','Sudeste',
8     np.where(tb_eleicoes['SG_UF']=='SP','Sudeste',
9     np.where(tb_eleicoes['SG_UF']=='DF','Centro_oeste',
10    np.where(tb_eleicoes['SG_UF']=='GO','Centro_oeste',
11    np.where(tb_eleicoes['SG_UF']=='MT','Centro_oeste',
12    np.where(tb_eleicoes['SG_UF']=='MS','Centro_oeste',
13    np.where(tb_eleicoes['SG_UF']=='AC','Norte',
14    np.where(tb_eleicoes['SG_UF']=='AP','Norte',
15    np.where(tb_eleicoes['SG_UF']=='AM','Norte',
16    np.where(tb_eleicoes['SG_UF']=='PA','Norte',
17    np.where(tb_eleicoes['SG_UF']=='RO','Norte',
18    np.where(tb_eleicoes['SG_UF']=='RR','Norte',
19    np.where(tb_eleicoes['SG_UF']=='TO','Norte', 'Nordeste'
20    )))))))

1 tb_eleicoes['Codigo_regiao'] = np.where(tb_eleicoes['Regiao']=='Exterior',1,
2     np.where(tb_eleicoes['Regiao']=='Sul',2,
3     np.where(tb_eleicoes['Regiao']=='Sudeste',3,
4     np.where(tb_eleicoes['Regiao']=='Centro_oeste',4,
5     np.where(tb_eleicoes['Regiao']=='Norte',5,6
6     ))))

1 tb_eleicoes['Cod_idade'] = np.where(tb_eleicoes['DS_FAIXA_ETARIA']=='17 anos',1,
2     np.where(tb_eleicoes['DS_FAIXA_ETARIA']=='18 anos',2,
3     np.where(tb_eleicoes['DS_FAIXA_ETARIA']=='19 anos',3,
4     np.where(tb_eleicoes['DS_FAIXA_ETARIA']=='20 anos',4,
5     np.where(tb_eleicoes['DS_FAIXA_ETARIA']=='21 a 24 anos',5,
6     np.where(tb_eleicoes['DS_FAIXA_ETARIA']=='25 a 29 anos',6,
7     np.where(tb_eleicoes['DS_FAIXA_ETARIA']=='30 a 34 anos',7,
8     np.where(tb_eleicoes['DS_FAIXA_ETARIA']=='35 a 39 anos',8,
9     np.where(tb_eleicoes['DS_FAIXA_ETARIA']=='40 a 44 anos',9,
10    np.where(tb_eleicoes['DS_FAIXA_ETARIA']=='45 a 49 anos',10,
11    np.where(tb_eleicoes['DS_FAIXA_ETARIA']=='50 a 54 anos',11,
12    np.where(tb_eleicoes['DS_FAIXA_ETARIA']=='55 a 59 anos',12,

```

```

13 np.where(tb_eleicoes['DS_FAIXA_ETARIA']== '55 a 59 anos',13,
14 np.where(tb_eleicoes['DS_FAIXA_ETARIA']== '60 a 64 anos',14,
15 np.where(tb_eleicoes['DS_FAIXA_ETARIA']== '65 a 69 anos',15,
16 np.where(tb_eleicoes['DS_FAIXA_ETARIA']== '70 a 74 anos',16,
17 np.where(tb_eleicoes['DS_FAIXA_ETARIA']== '75 a 79 anos',17,
18 np.where(tb_eleicoes['DS_FAIXA_ETARIA']== '80 a 84 anos',18,
19 np.where(tb_eleicoes['DS_FAIXA_ETARIA']== '85 a 89 anos',19,
20 np.where(tb_eleicoes['DS_FAIXA_ETARIA']== '90 a 94 anos',20,
21 np.where(tb_eleicoes['DS_FAIXA_ETARIA']== '95 a 99 anos',21,
22 np.where(tb_eleicoes['DS_FAIXA_ETARIA']== '100 anos ou mais',22,0
23 )))))))))))
24 tb_eleicoes.head()

```

	SG_UF	NM_MUNICIPIO	DS_GENERO	CD_GENERO	DS_FAIXA_ETARIA	DS_GRAU_ESCOLARIDADE
0	AP	PRACUÚBA	MASCULINO	2	40 a 44 anos	ENSINO FUNDAMENTAL COMPLETO
1	AP	PRACUÚBA	MASCULINO	2	65 a 69 anos	ENSINO FUNDAMENTAL COMPLETO
2	AP	PRACUÚBA	MASCULINO	2	75 a 79 anos	ANALFABETO
3	AP	PRACUÚBA	MASCULINO	2	75 a 79 anos	ENSINO FUNDAMENTAL INCOMPLETO
4	AP	PRACUÚBA	MASCULINO	2	25 a 29 anos	ENSINO FUNDAMENTAL INCOMPLETO

▼ Renomeando os atributos

```

1 tb_eleicoes = tb_eleicoes.rename(
2     columns={
3         'SG_UF': 'Estado',
4         'NM_MUNICIPIO': 'Municipio',
5         'DS_GENERO': 'Genero',
6         'CD_GENERO': 'Codigo_genero',
7         'DS_FAIXA_ETARIA': 'Faixa_etaria',
8         'DS_GRAU_ESCOLARIDADE': 'Escolaridade',
9         'CD_GRAU_ESCOLARIDADE': 'Codigo_escolaridade',
10        'QT_APTOS': 'Aptos',
11        'QT_COMPARECIMENTO': 'Comparecimento',
12        'QT_ABSTENCAO': 'Abstencao',
13    }
14 )
15 tb_eleicoes.head()

```

	Estado	Municipio	Genero	Codigo_genero	Faixa_etaria	Escolaridade	Aptos	Cor
0	AP	PRACUÚBA	MASCULINO	2	40 a 44 anos	ENSINO FUNDAMENTAL COMPLETO	10	
1	AP	PRACUÚBA	MASCULINO	2	65 a 69 anos	ENSINO FUNDAMENTAL COMPLETO	2	
2	AP	PRACUÚBA	MASCULINO	2	75 a 79 anos	ANALFABETO	2	

► Validando as quantidades e somas campos dataset

Trouxe a quantidade dos 26 Estados, mais DF e eleitores do Exterior *identificado pela abreviatura ZZ*

[] ↳ 7 células ocultas

▼ Iniciando a análise estatística

Visão estatística dos dados

```
1 tb_eleicoes.describe().round()
```

	Codigo_genero	Aptos	Comparecimento	Abstencao	Codigo_regiao	Cod_idade
count	4392869.0	4392869.0	4392869.0	4392869.0	4392869.0	4392869.0
mean	3.0	36.0	28.0	7.0	4.0	12.0
std	1.0	126.0	105.0	25.0	2.0	5.0
min	0.0	1.0	0.0	0.0	1.0	0.0
25%	2.0	2.0	1.0	0.0	3.0	9.0
50%	4.0	5.0	3.0	1.0	3.0	13.0
75%	4.0	20.0	15.0	5.0	6.0	16.0
max	4.0	7187.0	5443.0	1744.0	6.0	22.0

Podemos ver que há 4.392.869 instâncias. Média de comparecimento é de 28 e 7 para ausências

▼ Calculando taxas

Essas taxas serão calculadas por:

- Região,
- Genero,

- Faixa etária,
- Escolaridade.

```
1 turnover_rate = tb_eleicoes.Regiao.value_counts() / len(tb_eleicoes)
2
3 turnover_rate.round(3)
```

```
Sudeste      0.353
Nordeste     0.296
Sul          0.187
Centro_oeste 0.082
Norte        0.075
Exterior      0.008
Name: Regiao, dtype: float64
```

```
1 turnover_summary = tb_eleicoes.groupby('Regiao')
2 turnover_summary.sum()
```

```
D:\usuario\Suzy\AppData\Local\Temp\ipykernel_11008\4103618220.py:2: FutureWarning: The
turnover_summary.sum()
```

	Codigo_genero	Aptos	Comparecimento	Abstencao	Codigo_regiao	Cod_idade
Regiao						
Centro_oeste	1091480	11539323	9076598	2462725	1433288	4360525
Exterior	114430	697078	359120	337958	36919	421870
Nordeste	3971632	42390976	34115359	8275617	7788948	15639378
Norte	1002174	12560410	9930944	2629466	1649805	3942502
Sudeste	4680348	66707465	52060473	14646992	4648752	19745308
Sul	2509344	22558759	18194777	4363982	1639850	10061322

```
1 data = [
2     ('Sudeste',66707465,14646992,52060473),
3     ('Nordeste',42390976,8275617,34115359),
4     ('Sul',22558759,4363982,18194777),
5     ('Norte',12560410,2629466,9930944),
6     ('Centro_oeste',11539323,2462725,9076598),
7     ('Exterior',697078,337958,359120)
8 ]
9
10 tb_taxaregiao = pd.DataFrame(data, columns = ['Regiao','Aptos','Abstenc
11
12 tb_taxaregiao['%Abstecao'] = (tb_taxaregiao['Abstencao']/tb_taxaregiao[
13 tb_taxaregiao['%Comparecimento'] = (tb_taxaregiao['Compareceram']/tb_ta
```

14

15 `tb_taxaregiao.head(10)`

	Regiao	Aptos	Abstencao	Compareceram	%Abstecao	%Comparecimento
0	Sudeste	66707465	14646992	52060473	21.957051	78.042949
1	Nordeste	42390976	8275617	34115359	19.522120	80.477880
2	Sul	22558759	4363982	18194777	19.344956	80.655044
3	Norte	12560410	2629466	9930944	20.934555	79.065445
4	Centro_oeste	11539323	2462725	9076598	21.342023	78.657977

Aqui podemos ver que as regiões Sudeste e Nordeste são os que mais se destacam, onde:

- Sudeste detem de 35,3% do eleitorado, sendo a quantidade de 66.707.465 eleitores,
- Nordeste detem de 29,6% do eleitorado, sendo a quantidade de 42.390.96 eleitores,
- O eleitorado brasileiro que vota no exterior, foram os que mais se abstiveram tendo uma taxa de 48,5%,
- Sudeste teve a maior alta de abstenção com uma taxa de 21.9%,
- Sul possui a menor taxa de abstenção sendo 19.3%

Podemos concluir aqui, que as eleições do primeiro turno tiveram grande impacto nas Regiões Sul, Sudeste e Nordeste.

▼ Calculando taxa por genero

```
1 turnover_rate = tb_eleicoes.Genero.value_counts() / len(tb_eleicoes)
2
3 turnover_rate.round(3)
```

```
FEMININO      0.526
MASCULINO     0.469
NÃO INFORMADO 0.005
Name: Genero, dtype: float64
```

```
1 turnover_summary = tb_eleicoes.groupby('Genero')
2 turnover_summary.sum()
```

```
D:\usuario\Suzy\AppData\Local\Temp\ipykernel_11008\3819511299.py:2: FutureWarning: The
turnover summary sum()
```

```
1 data = [
2     ('Feminino',82373164,16428063,65945101),
3     ('Masculino',78044065,16288252,57775813),
4     ('Não_declarado',36782,20425,16357)
5 ]
6
7 tb_taxagenero = pd.DataFrame(data, columns = ['Genero','Aptos','Abstenc
8
9 tb_taxagenero['%Abstecao'] = (tb_taxagenero['Abstencao']/tb_taxagenero[
10 tb_taxagenero['%Comparecimento'] = (tb_taxagenero['Compareceram']/tb_ta
11
12 tb_taxagenero.head()
```

	Genero	Aptos	Abstencao	Compareceram	%Abstecao	%Comparecimento
0	Feminino	82373164	16428063	65945101	19.943465	80.056535
1	Masculino	78044065	16288252	57775813	20.870584	79.129416
2	Não_declarado	36782	20425	16357	55.529879	44.470121

Entre os eleitores, a maior parte é composta pelas mulheres.

- São 52,6% do total, sendo 82,3 milhões aptas e tiveram a menor taxa de abstenção 19,9%.
- Já os homens são 78 milhões, 46,9% do total, tendo uma taxa de abstenção e 20,9%
- Outros 36,7 mil eleitores não decalaram gênero, e esses foram os que mais tiveram uma taxa de abstenção 55,5%.

Conclui-se aqui por esses dados, que as mulheres possuem hoje o grande poder de decidir sobre quem governara nosso país, em contra partida, as pessoas que se declaram transgênero, transexuais e travestis ainda não tem confiança em nossos candidatos, que eles de fato farão diferença em suas vidas.

► Calculando taxa por faixa etária**

[] ↳ 3 células ocultas

O eleitorado acima de 70 anos não possui obrigatoriedade de votação, assim como jovens entre 16 a 18 anos. Porém idosos entre 70 e mais de 100 anos, ainda são bastante participativos.

- A menor taxa de abstenção dentro da faixa etária que tem obrigatoriedade no voto, foi de 11,9% e está dentro da faixa etária dos 50 a 59 anos,
- Já a maior taxa entre eleitores que possuem a obrigatoriedade, está entre os jovens de 21 a 24 anos, 23,4% e a segunda maior entre 25 a 29 anos, 22,4%.

- A maior quantidade de eleitores está entre a faixa etária de 35 a 39 anos, sendo 16,1 milhões.
- Os eleitores jovens que não possuem obrigatoriedade, se mantiveram em 80% de comparecimento.

Isso nos mostra que os eleitores estão cada vez mais cedo buscando fazer a diferença com suas escolhas, assim como os mais velhos acreditam ainda poder decidir sobre a escola dos governantes.

O artigo 14 capítulo 1 da constituição, dita que o voto é facultativo para analfabetos, maiores de 60 anos e menores de 18 anos.

► *Calculando taxa por escolaridade*

[] ↳ 4 células ocultas

▼ Análises

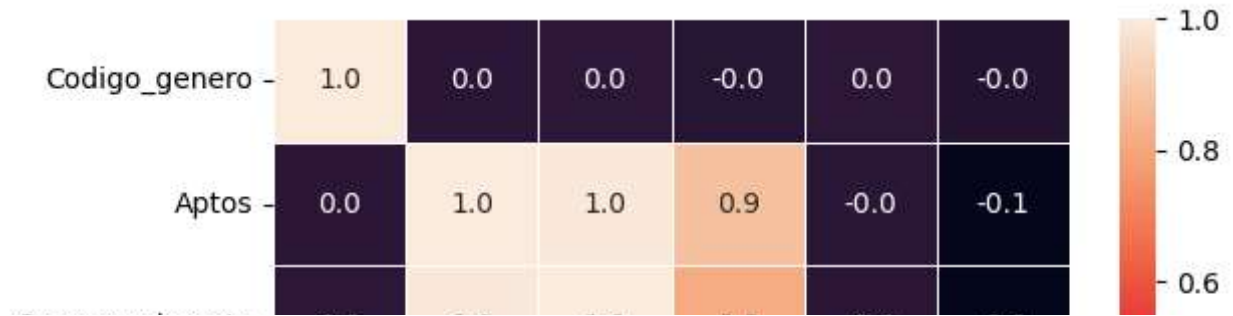
Analisando correlações

```
1 import seaborn as sn
2 import matplotlib.pyplot as plt
```

```
1 correlation = tb_eleicoes.corr()
2 plot = sn.heatmap(correlation, annot = True, fmt=".1f", linewidths=.6)
3 plot
```



```
D:\usuario\Suzy\AppData\Local\Temp\ipykernel_11008\4204524103.py:1: FutureWarning: The
correlation = tb_eleicoes.corr()
<AxesSubplot: >
```



Atributos com uma correlação positiva:

- numero de aptos vs abstenção: 0,9;
- numero de aptos vs comparecimento: 0,8.

Atributos com uma correlação negativa:

- idade vs aptos, comparecimento e abstenção: -0,1.

Aqui notamos que a idade foi o que mais contribuiu com a ida ou não as urnas, porém temos que considerar entre elas o fator onde parte dos eleitores não possuem obrigatoriedade de votação.

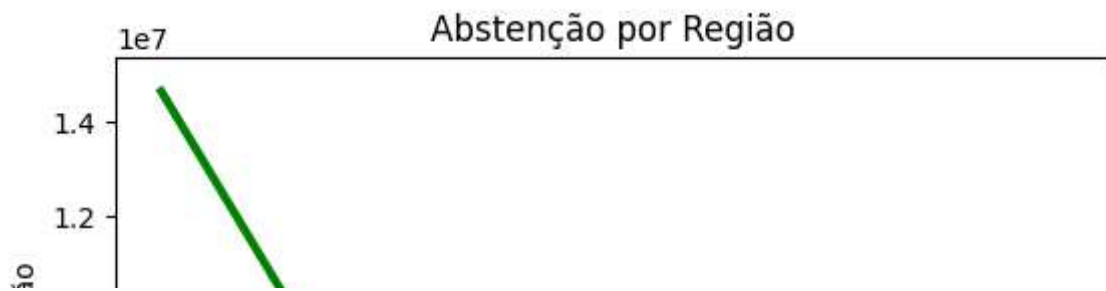
91 100 100 100 100

▼ Analisando quantidade de abstenção por Região

U

```
1 plt.plot(tb_taxaregiao['Regiao'], tb_taxaregiao['Abstencao'], color = 'r')
2 plt.ylabel('Quandtidade abstenção')
3 plt.xlabel('Regiões')
4 plt.title('Abstenção por Região')
5 plt.show
```

```
<function matplotlib.pyplot.show(close=None, block=None)>
```



Avaliando as abstenções por quantidade de eleitores, vemos aqui que o Sudeste teve a maior quantidade de abstenções seguidas do Nordeste, visto que ambas as regiões tem a maior quantidade de eleitores, esse gráfico faz sentido.

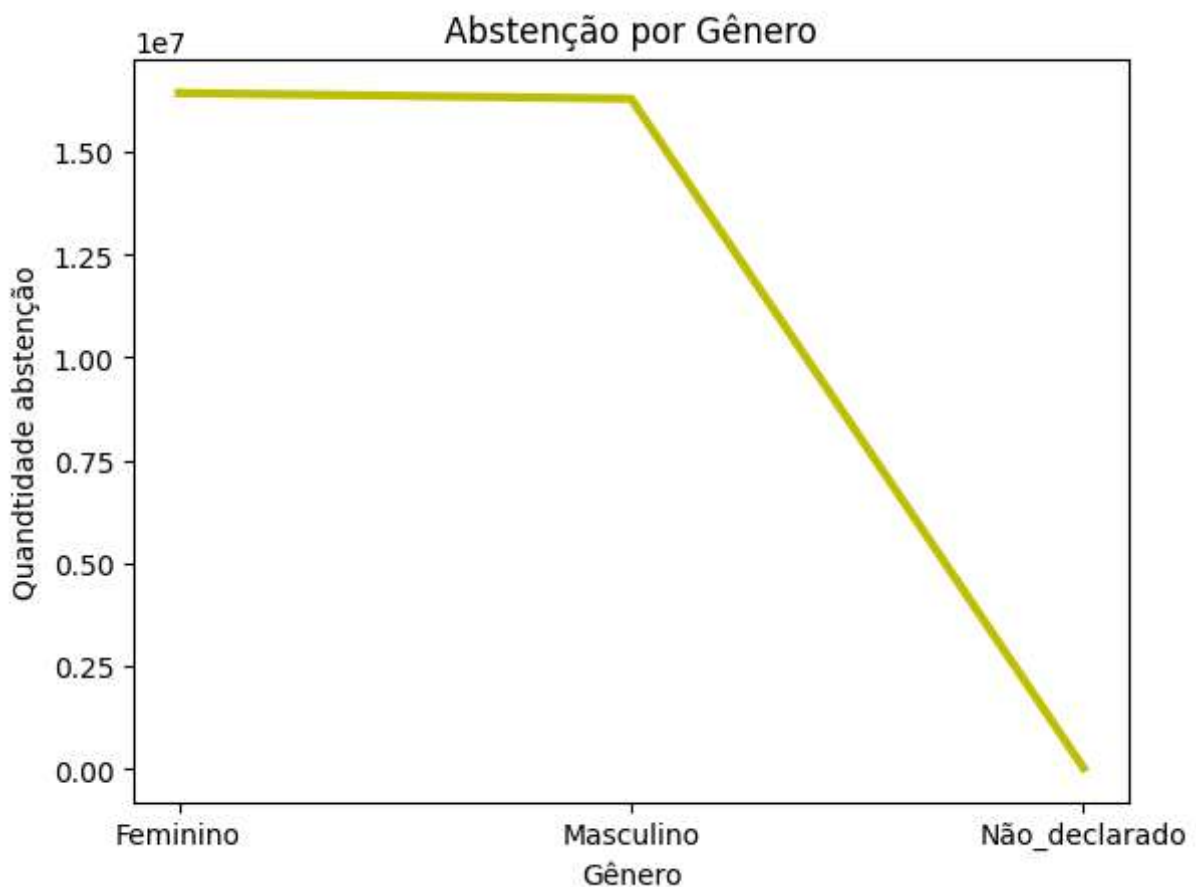
```
<function matplotlib.pyplot.show(close=None, block=None)>
```

▼ Analisando quantidade de abstenção por Gênero

```
<function matplotlib.pyplot.show(close=None, block=None)>
```

```
1 plt.plot(tb_taxagenero['Genero'], tb_taxagenero['Abstencao'], color='y')
2 plt.ylabel('Quandtidade abstenção')
3 plt.xlabel('Gênero')
4 plt.title('Abstenção por Gênero')
5 plt.show
```

```
<function matplotlib.pyplot.show(close=None, block=None)>
```

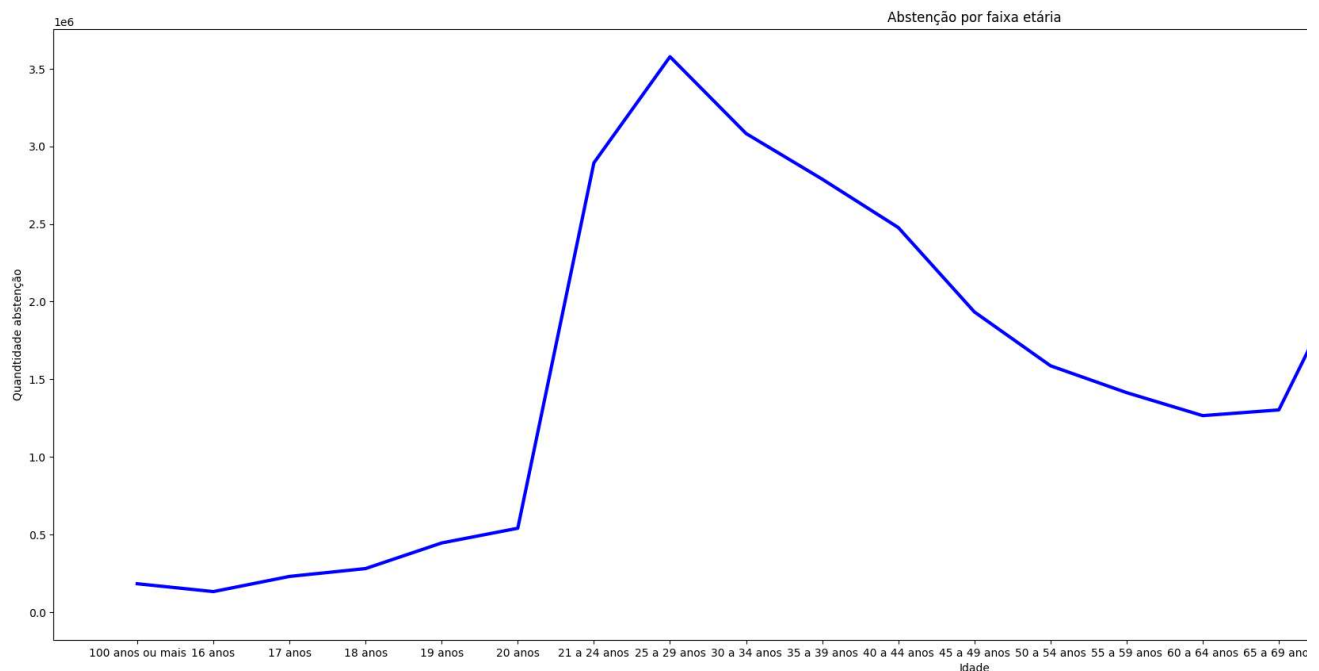


Vimos que as mulheres são em maior quantidade de eleitores, porém em abstenções elas se igualam aos homens.

▼ Analisando quantidade de abstenção por Faixa Etária

```
1 figura = plt.figure(figsize=(30,10))
2 plt.plot(tb_taxaidade['Faixa_etaria'], tb_taxaidade['Abstencao'], color='blue')
3 plt.ylabel('Quandtidade abstenção')
4 plt.xlabel('Idade')
5 plt.title('Abstenção por faixa etária')
6 plt.show
```

<function matplotlib.pyplot.show(close=None, block=None)>

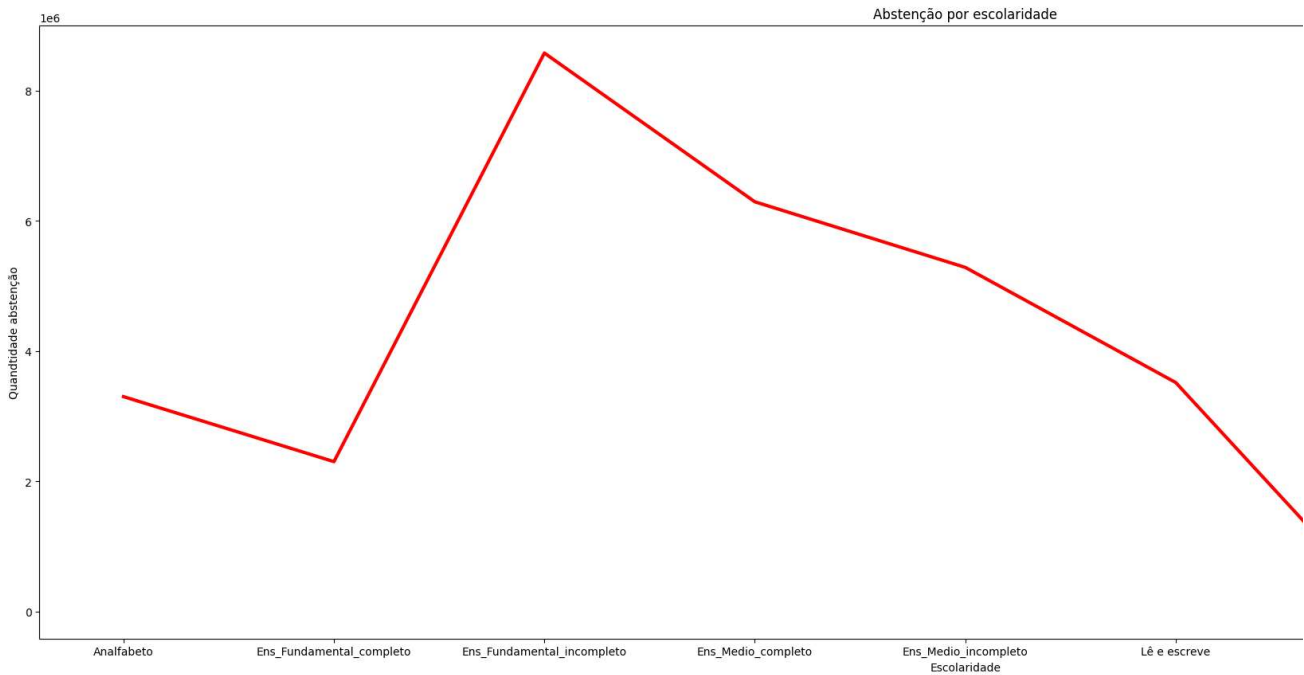


Aqui temos a maior quantidade de abstenções em pessoas que possuem de 25 a 29 anos.

▼ Analisando quantidade de abstenção por Escolaridade

```
1 figura = plt.figure(figsize=(30,10))
2 plt.plot(tb_taxaescolaridade['Escolaridade'], tb_taxaescolaridade['Abstencao'], color='blue')
3 plt.ylabel('Quandtidade abstenção')
4 plt.xlabel('Escolaridade')
5 plt.title('Abstenção por escolaridade')
6 plt.show
```

```
<function matplotlib.pyplot.show(close=None, block=None)>
```



Quem tem ensino fundamental incompleto foram os que mais abstiveram, visto aqui por quantidade.

FIM