



Learn Git and GitHub without any code!

Using the Hello World guide, you'll start a branch, write comments, and open a pull request.

[Read the guide](#)

 [mariannamassi](#) / [Data-Visualization](#)

Code

Issues

Pull requests

Actions

Projects

Wiki

Security

Insights

 main ▾



[Data-Visualization](#) / [EstudoSocial_ENEM2019 \(1\).ipynb](#)



mariannamassi Add files via upload



 1 contributor



Raw

Blame



1061 lines (1061 sloc) 229 KB

Estudo Social - ENEM 2019

Análise das condições sociais dos candidatos do ENEM em 2019.

Marianna Massi

```
In [119]: import pandas as pd

fonte = "https://github.com/alura-cursos/imersao-dados-2-2020/
blob/master/MICRODADOS_ENEM_2019_SAMPLE_43278.csv?raw=true"

dados = pd.read_csv(fonte)
provas = ['NU_NOTA_CN', 'NU_NOTA_CH', 'NU_NOTA_MT', 'NU_NOTA_L
C', 'NU_NOTA_REDACAO']
dados['NU_NOTA_TOTAL'] = dados[provas].sum(axis=1)
dados.head()
```

```
Out[119]:
```

	NU_INSCRICAO	NU_ANO	CO_MUNICIPIO_RESIDENCIA	NO_MUNICIPIO_R
0	190001004661	2019	1506138	Redenção
1	190001004674	2019	1504208	Marabá
2	190001004722	2019	1501402	Belém
3	190001004735	2019	1507300	São Félix do Xingu
4	190001004776	2019	1500800	Ananindeua

5 rows × 137 columns

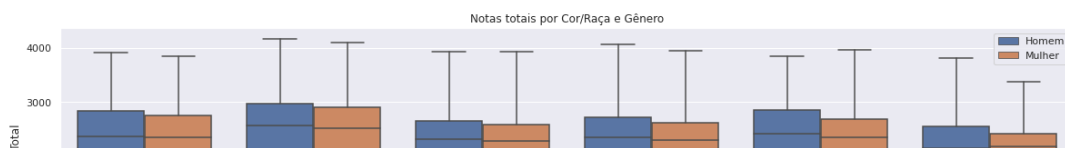
```
In [ ]: sns.boxplot(x="TXT_COR_RACA", y="NU_NOTA_TOTAL", data = dados,
hue="TP_SEXO")
```

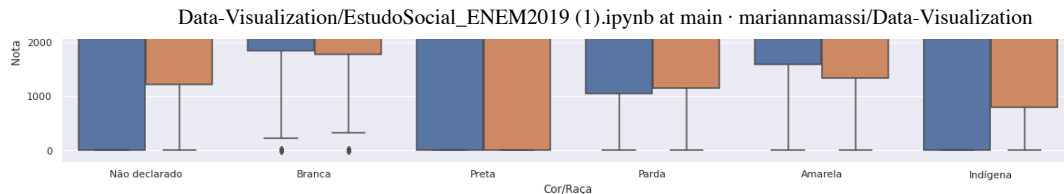
```
In [166]: import seaborn as sns
import matplotlib.pyplot as plt

dados['TXT_COR_RACA']=pd.cut(
    dados['TP_COR_RACA'],
    bins=[-1, 0, 1, 2, 3, 4, 5],
    labels=['Não declarado', 'Branca', 'Preta', 'Parda', 'Amare
la', 'Indígena']
)

plt.figure(figsize=(20, 5))
sns.boxplot(x="TXT_COR_RACA", y="NU_NOTA_TOTAL", data = dados,
hue="TP_SEXO")
plt.xlabel("Cor/Raça")
plt.ylabel("Nota Total")
L=plt.legend()
L.get_texts()[0].set_text('Homem')
L.get_texts()[1].set_text('Mulher')
plt.title("Notas totais por Cor/Raça e Gênero")
```

```
Out[166]: Text(0.5, 1.0, 'Notas totais por Cor/Raça e Gênero')
```





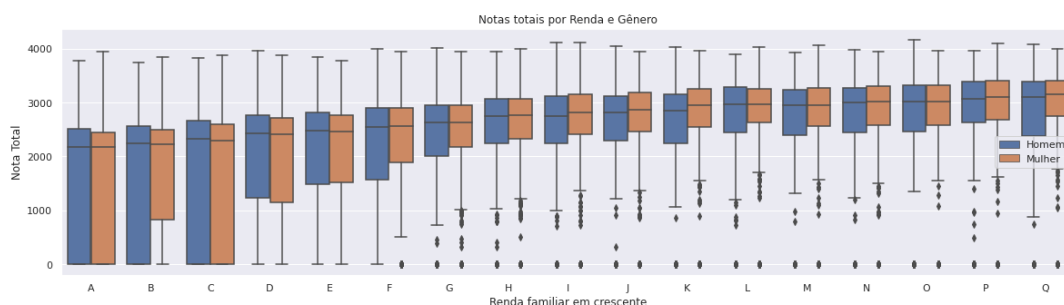
```
In [158]: import seaborn as sns
import matplotlib.pyplot as plt

plt.figure(figsize=(20, 5))

renda = dados["Q006"].unique()
renda.sort()

sns.boxplot(x="Q006", y="NU_NOTA_TOTAL", data = dados, hue="TP_SEXO", order = renda)
plt.xlabel("Renda familiar em crescente")
plt.ylabel("Nota Total")
L=plt.legend()
L.get_texts()[0].set_text('Homem')
L.get_texts()[1].set_text('Mulher')
plt.title("Notas totais por Renda e Gênero")
```

Out[158]: Text(0.5, 1.0, 'Notas totais por Renda e Gênero')



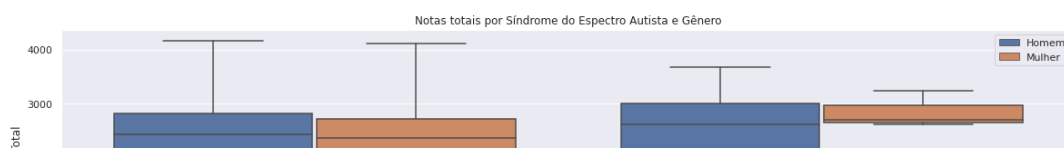
```
In [167]: import seaborn as sns
import matplotlib.pyplot as plt

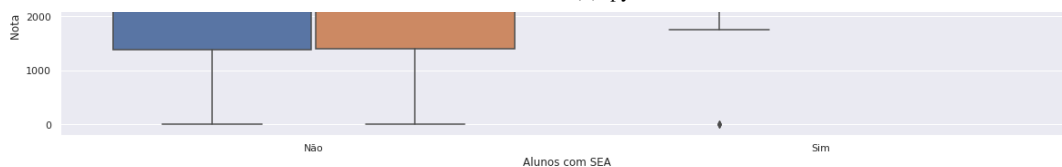
dados['TXT_AUTISMO']=pd.cut(
    dados['IN_AUTISMO'],
    bins=[-1, 0, 1],
    labels=['Não', 'Sim']
)

plt.figure(figsize=(20, 5))

sns.boxplot(x="TXT_AUTISMO", y="NU_NOTA_TOTAL", data = dados,
hue="TP_SEXO")
plt.xlabel("Alunos com SEA")
plt.ylabel("Nota Total")
L=plt.legend()
L.get_texts()[0].set_text('Homem')
L.get_texts()[1].set_text('Mulher')
plt.title("Notas totais por Síndrome do Espectro Autista e Gênero")
```

Out[167]: Text(0.5, 1.0, 'Notas totais por Síndrome do Espectro Autista e Gênero')





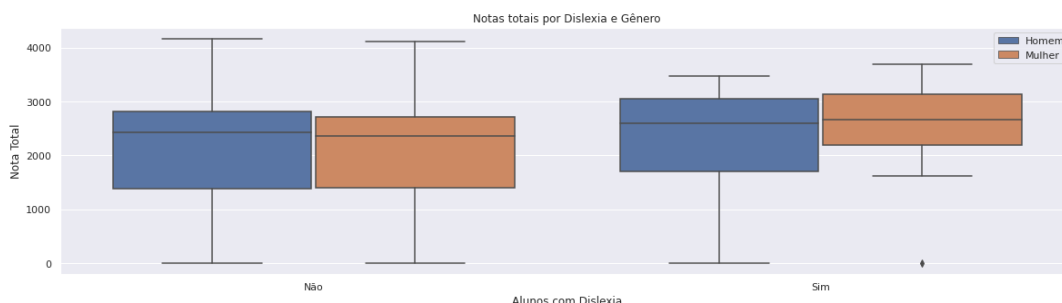
```
In [160]: import seaborn as sns
import matplotlib.pyplot as plt

dados['TXT_DISLEXIA']=pd.cut(
    dados['IN_DISLEXIA'],
    bins=[-1, 0, 1],
    labels=['Não', 'Sim']
)

plt.figure(figsize=(20, 5))

sns.boxplot(x="TXT_DISLEXIA", y="NU_NOTA_TOTAL", data = dados,
hue="TP_SEXO")
plt.xlabel("Alunos com Dislexia")
plt.ylabel("Nota Total")
L=plt.legend()
L.get_texts()[0].set_text('Homem')
L.get_texts()[1].set_text('Mulher')
plt.title("Notas totais por Dislexia e Gênero")
```

Out[160]: Text(0.5, 1.0, 'Notas totais por Dislexia e Gênero')

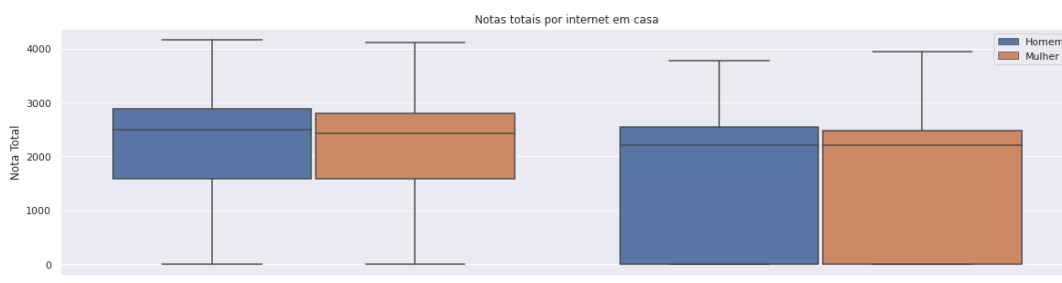


```
In [168]: import seaborn as sns
import matplotlib.pyplot as plt

plt.figure(figsize=(20, 5))

sns.boxplot(x="Q025", y="NU_NOTA_TOTAL", data = dados, hue="TP_SEXO")
plt.xlabel("Alunos com e sem internet")
plt.ylabel("Nota Total")
L=plt.legend()
L.get_texts()[0].set_text('Homem')
L.get_texts()[1].set_text('Mulher')
plt.title("Notas totais por internet em casa")
print("OBS: A é igual a Não e B igual a Sim")
```

OBS: A é igual a Não e B igual a Sim



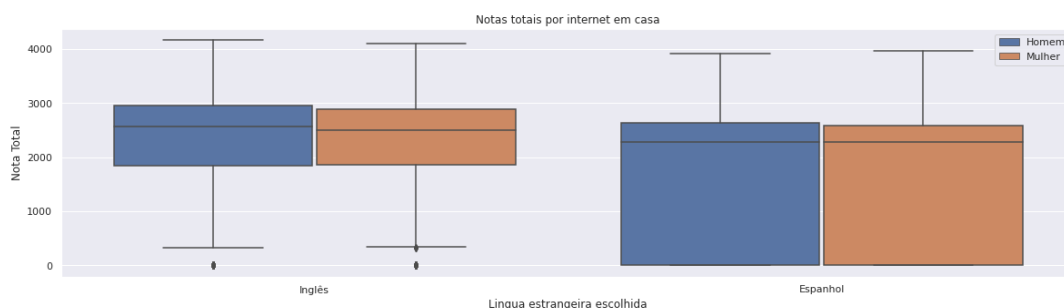
```
In [164]: import seaborn as sns
import matplotlib.pyplot as plt

dados['TXT_LINGUA']=pd.cut(
    dados['TP_LINGUA'],
    bins=[-1, 0, 1],
    labels=['Inglês', 'Espanhol']
)

plt.figure(figsize=(20, 5))

sns.boxplot(x="TXT_LINGUA", y="NU_NOTA_TOTAL", data = dados, hue="TP_SEXO")
plt.xlabel("Lingua estrangeira escolhida")
plt.ylabel("Nota Total")
L=plt.legend()
L.get_texts()[0].set_text('Homem')
L.get_texts()[1].set_text('Mulher')
plt.title("Notas totais por internet em casa")
```

Out[164]: Text(0.5, 1.0, 'Notas totais por internet em casa')



```
In [165]: import seaborn as sns
import matplotlib.pyplot as plt

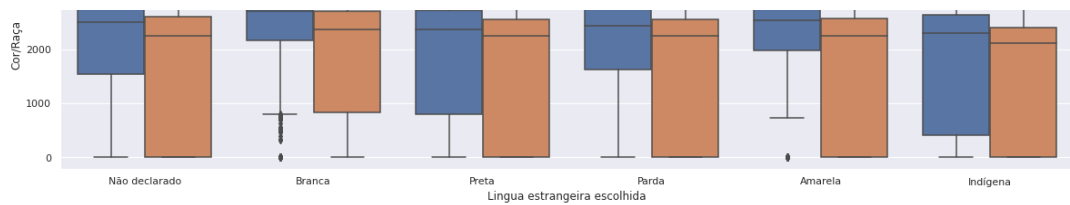
dados['TXT_COR_RACA']=pd.cut(
    dados['TP_COR_RACA'],
    bins=[-1, 0, 1, 2, 3, 4, 5],
    labels=['Não declarado', 'Branca', 'Preta', 'Parda', 'Amarela', 'Indígena']
)

plt.figure(figsize=(20, 5))

sns.boxplot(x="TXT_COR_RACA", y="NU_NOTA_TOTAL", data = dados, hue="TP_LINGUA")
plt.xlabel("Lingua estrangeira escolhida")
plt.ylabel("Cor/Raça")
L=plt.legend()
L.get_texts()[0].set_text('Inglês')
L.get_texts()[1].set_text('Espanhol')
plt.title("Notas totais por linguas estrangeiras escolhidas e cor/raça")
```

Out[165]: Text(0.5, 1.0, 'Notas totais por linguas estrangeiras escolhidas e cor/raça')





```
In [156]: import seaborn as sns
import matplotlib.pyplot as plt

dados['TXT_COR_RACA']=pd.cut(
    dados['TP_COR_RACA'],
    bins=[-1, 0, 1, 2, 3, 4, 5],
    labels=['Não declarado', 'Branca', 'Preta', 'Parda', 'Amarela', 'Indígena']
)

plt.figure(figsize=(20, 5))
sns.boxplot(x="TXT_COR_RACA", y="NU_NOTA_TOTAL", data = dados,
            hue="IN_GESTANTE")
plt.xlabel("Nota Total")
plt.ylabel("Cor/Raça")
L=plt.legend()
L.get_texts()[0].set_text('Não Gestante')
L.get_texts()[1].set_text('Gestante')
plt.title("Notas totais por Cor/Raça e Gênero")
```

Out[156]: Text(0.5, 1.0, 'Notas totais por Cor/Raça e Gênero')

