



PROJETO DE AVALIAÇÃO LPAA

ALUNO: PEDRO LIMA



Missão

Basicamente irei tratar os dados conforme solicitado, realizar pequenas análises baseadas no conteúdo exposto durante a apresentação da disciplina.

Metodologia

Será utilizado programação em Python para otimizar e clarear os dados envolvidos no dataframe indicado.

Conteúdo abordado: Classes, elaboração de gráficos, exploração de bibliotecas, limpeza de dados.

Exibindo dados do DataFrame

```
###lendo arquivo csv que irá ser tratado
df = pd.read_csv('/content/Health_AnimalBites.csv')
print(df.head(10))
###exibindo 10 primeiras linhas do arquivo sem nenhum tratamento de dados
```

	bite_date	SpeciesIDDesc	BreedIDDesc	GenderIDDesc	color	\
0	1985-05-05 00:00:00	DOG	NaN	FEMALE	LIG. BROWN	
1	1986-02-12 00:00:00	DOG	NaN	UNKNOWN	BRO & BLA	
2	1987-05-07 00:00:00	DOG	NaN	UNKNOWN	NaN	
3	1988-10-02 00:00:00	DOG	NaN	MALE	BLA & BRO	
4	1989-08-29 00:00:00	DOG	NaN	FEMALE	BLK-WHT	
5	1989-11-24 00:00:00	DOG	NaN	UNKNOWN	NaN	
6	1990-02-08 00:00:00	DOG	NaN	FEMALE	BLACK/WHIT	
7	1990-02-22 00:00:00	DOG	NaN	MALE	BLK-WHT	
8	1990-08-02 00:00:00	DOG	NaN	MALE	BROWN	
9	1990-08-19 00:00:00	DOG	NaN	UNKNOWN	BRN-TAN	

	vaccination_yrs	vaccination_date	victim_zip	AdvIssuedYNDesc	\
0	1.0	1985-06-20 00:00:00	40229	NO	
1	NaN	NaN	40218	NO	
2	NaN	NaN	40219	NO	
3	NaN	NaN	NaN	NO	
4	NaN	NaN	NaN	NO	
5	NaN	NaN	40211	NO	
6	1.0	1990-02-13 00:00:00	40203	NO	
7	NaN	NaN	NaN	NO	

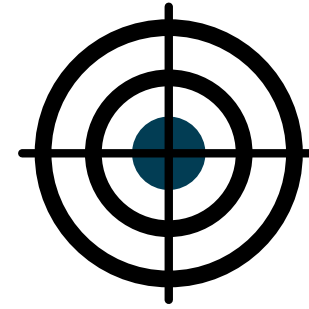
Removendo duplicatas e dados NaN da coluna “data”.

```
###sabendo do que se trata o dataframe, realizaremos
um estudo de casos
###definindo quem é a coluna de data
coluna_de_data = 'bite_date'
###primeiramente limparemos as linhas que nao
tiverem datas registros, pois para o estudo so
consideramos com data
###além disso, removeremos linhas duplicatas
df_limpo = df.drop_duplicates().dropna(subset=
[coluna_de_data])
###exibindo dataframe após remoção das linhas sem
data
print(df_limpo)
```

	bite_date	SpeciesIDDesc	BreedIDDesc	GenderIDDesc	\
0	1985-05-05 00:00:00	DOG	NaN	FEMALE	
1	1986-02-12 00:00:00	DOG	NaN	UNKNOWN	
2	1987-05-07 00:00:00	DOG	NaN	UNKNOWN	
3	1988-10-02 00:00:00	DOG	NaN	MALE	
4	1989-08-29 00:00:00	DOG	NaN	FEMALE	
...	
8998	2017-09-05 00:00:00	DOG	NaN	NaN	
8999	2017-09-07 00:00:00	DOG	POMERANIAN	MALE	
9000	2017-09-07 00:00:00	DOG	LABRADOR RETRIV	MALE	
9001	2017-09-07 00:00:00	DOG	LABRADOR RETRIV	FEMALE	
9002	2017-09-07 00:00:00	DOG	BOXER	NaN	

	color	vaccination_yrs	vaccination_date	victim_zip	\
0	LIG. BROWN	1.0	1985-06-20 00:00:00	40229	
1	BRO & BLA	NaN	NaN	40218	
2	NaN	NaN	NaN	40219	
3	BLA & BRO	NaN	NaN	NaN	
4	BLK-WHT	NaN	NaN	NaN	
...	
8998	NaN	NaN	NaN	40243	
8999	RED	NaN	NaN	40204	
9000	BROWN	NaN	NaN	47130	
9001	BLK WHT	NaN	NaN	40229	
9002	BRN BLK	NaN	NaN	40229	





Análise quantitativa

```
###agora temos a curiosidade de descobrir qual animal que mais mordeu durante todo o período do estudo
###definindo a coluna de animais
coluna_de_animal = 'SpeciesIDDesc'
animal_que_mais_mordeu = df_limpo[coluna_de_animal].mode().iloc[0]

###resultado do animal que mais mordeu
print(f"O animal que mais mordeu durante o período do estudo foi {animal_que_mais_mordeu}")
```

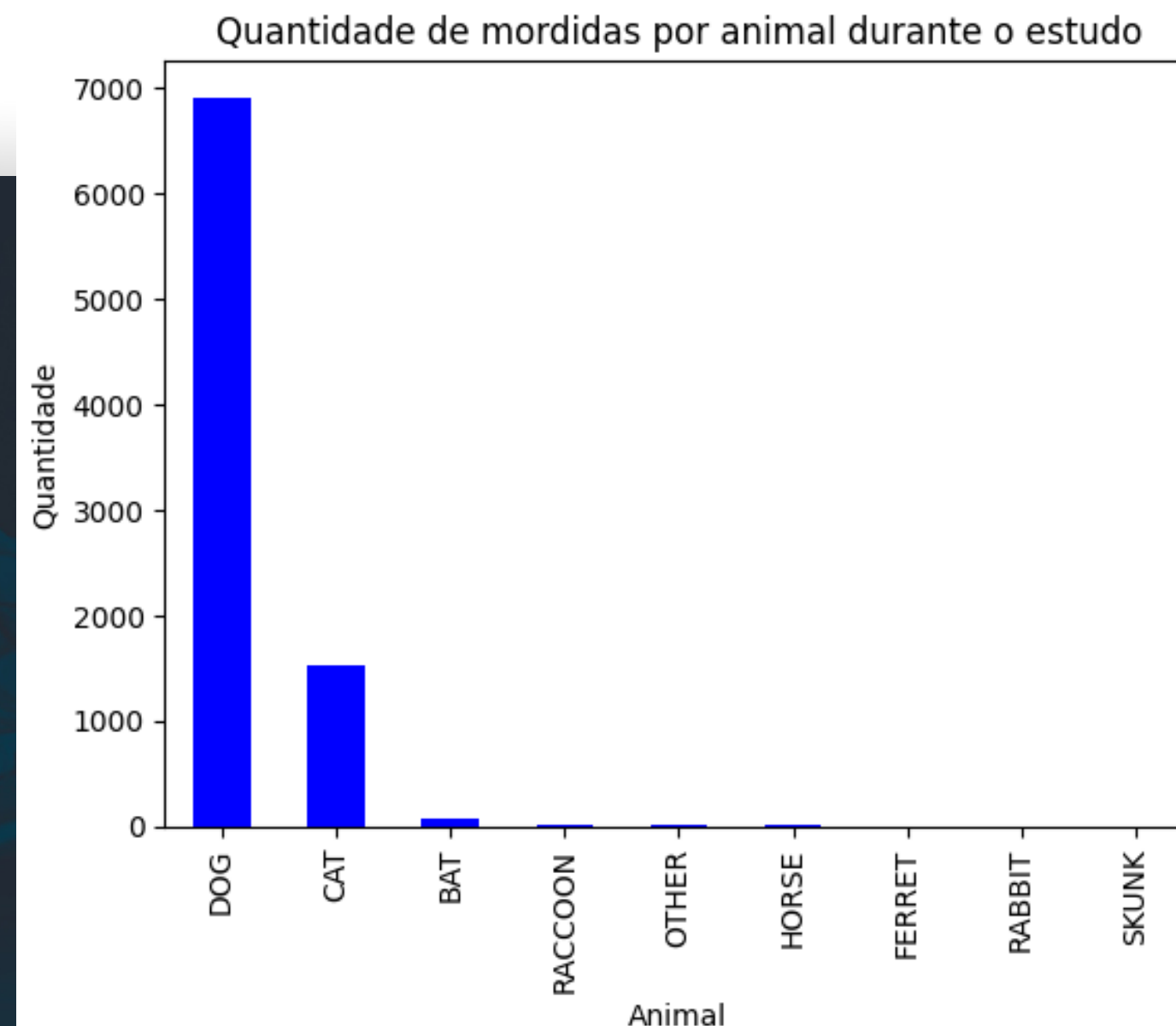
```
O animal que mais mordeu durante o período do estudo foi DOG
```

Descobrimos qual o animal que mais mordeu durante o período do estudo

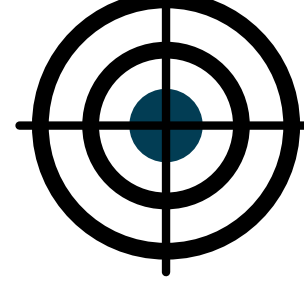


Análise gráfica 1.0

```
###que tal jogarmos isso num grafico de barras azul?  
  
###contando a quantidade de vezes que certo animal mordeu durante o periodo do estudo  
  
contagem_animal = df_limpo[coluna_de_animal].value_counts()  
  
###plotando o grafico requerido  
contagem_animal.plot(kind='bar',color='blue')  
plt.title(f'Quantidade de mordidas por animal durante o estudo')  
plt.xlabel('Animal')  
plt.ylabel('Quantidade')  
plt.show()
```



Agora realizando uma análise visual dos dados expostos (Animal x Quantidade de mordidas)

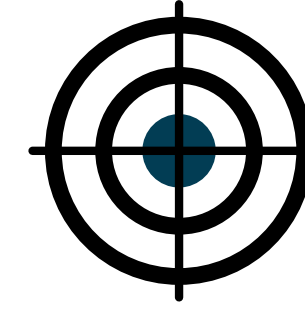


Análise gráfica 2.0

```
###fazendo outro grafico, agora de quantas vezes um certo animal mordeu tal gênero
#definindo qual coluna de genero
coluna_de_genero = 'GenderIDDesc'
contagem_ocorrencias = df.groupby([coluna_de_animal,coluna_de_genero]).size().reset_index(name='Contage')

##criando grafico
plt.figure(figsize=(25,28))
plt.bar(contagem_ocorrencias[coluna_de_animal] + ' - ' + contagem_ocorrencias[coluna_de_genero], contagem_ocorrencias['Contage'])
plt.title('Quantidade de vezes que cada animal mordeu por genero')
plt.xlabel('Animal - Sexo')
plt.ylabel('Contage')
plt.show()
```

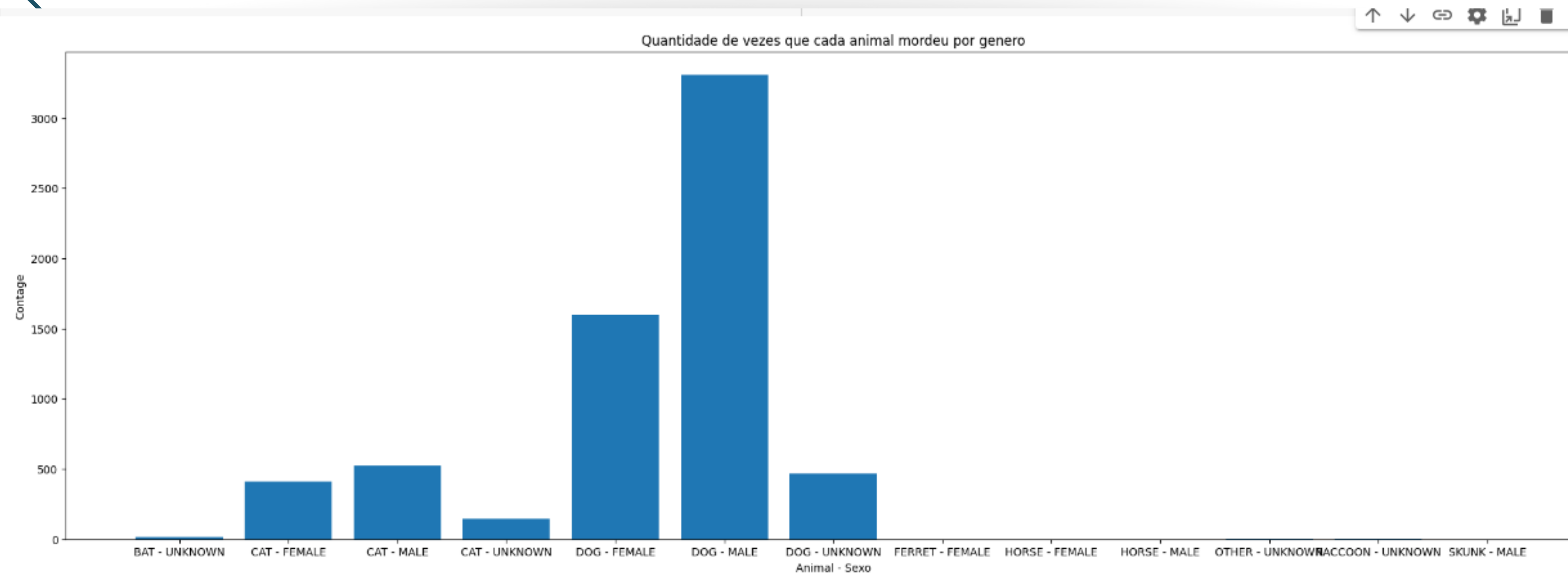
Agora realizando uma análise visual dos dados expostos (Mordida de animais x Gênero)



Análise gráfica 2.0

```
###fazendo outro grafico, agora de quantas vezes um certo animal mordeu tal gênero
#definindo qual coluna de genero
coluna_de_genero = 'GenderIDDesc'
contagem_ocorrencias = df.groupby([coluna_de_animal,coluna_de_genero]).size().reset_index(name='Contage')

##criando grafico
plt.figure(figsize=(25,28))
plt.bar(contagem_ocorrencias[coluna_de_animal] + ' - ' + contagem_ocorrencias[coluna_de_genero], contagem_ocorrencias['Contage'])
plt.title('Quantidade de vezes que cada animal mordeu por genero')
plt.xlabel('Animal - Sexo')
plt.ylabel('Contage')
plt.show()
```



Agora realizando uma análise visual dos dados expostos (Mordida de animais x Gênero)

Referências



Youtube - PROFESSOR
NETO



Blog - Cubos academy



Livro em pdf - Python para
leigos

