INSTITUTO FEDERAL
DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
Brasília
Campus Brasília

TECNOLOGIA EM SISTEMAS PARA INTERNET

Matheus Nícollas de Souza Mota
Thiago Marinho da Silva Campos

RELATÓRIO DE PRÁTICA INTEGRADA
DE
CIÊNCIA DE DADOS E INTELIGÊNCIA ARTIFICIAL

Brasília - DF
18/09/2020

Sumário

1. Objetivos

O objetivo é descobrir fatos interessantes a partir da coleta de dados de observações de OVINIs, a partir da fonte: http://www.nuforc.org/ e desenvolver um script Python que irá executar a coleta dentro de um período de vinte anos, entre setembro 1997 e agosto de 2017, armazenando tudo em um DataFrame e depois salvando em um arquivo .CSV com o nome OVNIS.csv. no GITHub, após esta coleta será feita a etapa do pipeline dos dados, na qual ocorrerá a exploração dos dados coletados respondendo diversas questões descritas no decorrer deste relatório.

2. Descrição do problema

O projeto consiste em reunir fatos interessantes relacionados a OVINIs, a partir de relatos realizados dentro de um período de vinte anos usando o site Nuforc. O desafio consiste em fazer a extração de dados de forma tabular, afinal, para que os dados possam ser analisados eles acabam se tornando tabelas.

O WebScraping consiste em extrair os dados formatados com tag's da linguagem HTML. Como iremos extrair vinte anos de dados, consultaremos 240 páginas web, uma por cada mês, por vinte anos, entre setembro 1997 e agosto de 2017.

Para extrair conhecimento destes dados é necessário uma exploração dos mesmos, e esta exploração visa responder as seguintes questões:

1. Saber a quantidade de linhas, observações ou variáveis que foram coletadas.
2. Quantos relatos ocorreram por estado em ordem decrescente?
3. Remover possíveis campos vazios (sem estado).
4. Limitar a análise aos estados dos Estados Unidos.
5. Consulta por cidades, com o objetivo de saber quais contêm o maior número de relatos (cidades que apresentem ao menos 10 relatos).
6. Com o dado anterior, responder a seguinte pergunta: por que será que essa é a cidade que possui mais relatos?
7. Fazer uma *query* exclusiva para o estado com maior número de relatos, buscando cidades que possuam um número superior a 10 relatórios. Enfatizar a cidade, a quantidade de relatos e formato do objeto não identificado.

.

3. Desenvolvimento

O desenvolvimento do algoritmo foi feito na plataforma Google Collab, esta plataforma foi escolhida pois ao iniciar um notebook na mesma, as bibliotecas e dependências do Python são todas da nuvem.

O desenvolvimento foi dividido em 7 etapas de acordo com as perguntas elaboradas no exercício. A primeira questão pede a quantidade de variáveis que foram coletadas, no óvnis.csv, foi feito um script que usa a função count para trazer a quantidade exata. A segunda questão pede os relatos que ocorreram por estado em ordem decrescente, foi necessário usar a função group by do pandas recebendo o uma coluna como parâmetro, que no caso é o estado, também é usado um count para saber a quantidade e por ultimo a função sort_values para fazer o ordenamento decrescente. A terceira questão solicita a remoção de campos vazios, então usamos a função dropna e o parâmetro é a coluna estado. A quarta questão solicita a limitação da análise somente apara estados dos EUA, sendo assim foi necessário criar um vetor com os estados para fazer a comparação com os dados da página, caso o dado da página seja igual ao contido no vetor, este dado é atribuido ao dataframe. A questão 5 solicita a consulta por cidades com o objetivo de saber quais tem o maior relato, foi necessário usar o group by para agrupar os dados por cidade, a função count para contar e a função sort para ordenar do maior para o menor além disso foi preciso uma restrição de query para filtrar as cidades que tem mais de dez relatos. A questão 6 pergunta porque será que a cidade encontrada na questão cinco é a que tem mais relatos? Achamos que por se tratar de local afastado há maior movimentação de quem não quer ser visto. A questão 7 pede para fazer uma query com maior numero s de relatos com cidades que possuem número maior que dez relatos, além disso é necessário trazer a cidade, quantidade de relatos e o formato do objeto, para resolver este problema criamos um novo dataframe com as 3 colunas solicitadas, passamos o dataframe com estado da Califórnia por se tratar do estado com maior numero de casos, criamos um vetor com as colunas do novo dataframe que será iniciado recebendo as colunas que são atribuídas para cada elemento extraído acima.

3.1 Código implementado

```
# -*-
coding:
utf-8 -
*-
        """[DEF] 5.4 - Exploração dos dados com SQL.ipynb
        Automatically generated by Colaboratory.
        Original file is located at

        https://colab.research.google.com/drive/1k_F2lBxZ48FPGLvCGTkRpKXeCONP9qv7
        """

        import pandas as pd

        df_ovnis = pd.read_csv("ovnis.csv", index_col=[0])

        print("1. Quantidade de variáveis que foram coletadas:")
        #Conta a quantidade de vdados para cada variavel com a funcao count()
        df_ovnis.count()

        print("2. Relatos que ocorreram por estado em ordem decrescente:")
        #Usa a funcao groupby() pra agrupar por estado, a funcao count() para contar
        e a funcao sort_values() para ordenar de forma decrescente
        df_relatos = df_ovnis
        df_relatos = df_relatos.groupby('State').count()
        df_relatos = df_relatos.sort_values(ascending=False, by="Posted")['Posted']
        df_relatos

        print("3. Remover possíveis campos vazios (sem estado):")
        #Exclui toda linha cujo Estado seja nulo com a funcao dropna()
        df_ovnis = df_ovnis.dropna(subset=["State"])
        df_ovnis

        print("4. Limitar a análise aos estados dos Estados Unidos.")

        #como a planilha nao oferece dados dos paises de origem, o vetor abaixo
        representa a sigla dos estados dos EUA
        states = ["AL", "AK", "AZ", "AR", "CA", "CO", "CT", "DC", "DE", "FL", "GA",
                  "HI", "ID", "IL", "IN", "IA", "KS", "KY", "LA", "ME", "MD",
                  "MA", "MI", "MN", "MS", "MO", "MT", "NE", "NV", "NH", "NJ",
                  "NM", "NY", "NC", "ND", "OH", "OK", "OR", "PA", "RI", "SC",
                  "SD", "TN", "TX", "UT", "VT", "VA", "WA", "WV", "WI", "WY"]

        selection = df_ovnis['State'].isin(states)
        df_ovnis = df_ovnis[selection]
```

```
df_ovnis

print("5. Consulta por cidades, com o objetivo de saber quais contêm o maior número de relatos (cidades que apresentem ao menos 10 relatos).")

#Usa a funcao groupby() pra agrupar por estado, a funcao count() para contar e a funcao sort_values() para ordenar de forma decrescente
cidades = df_ovnis
cidades = cidades.groupby('City').count()
cidades = cidades.sort_values(ascending=False, by="Posted")
cidades = cidades.query('Posted >= 10')['Posted']
cidades

print("6. Com o dado anterior, responder a seguinte pergunta: por que será que essa é a cidade que possui mais relatos?")
print("Porque é a cidade onde fica mais afastada de grandes metrópoles, construída no local de antigos canais indigenas")

print("7. Fazer uma query exclusiva para o estado com maior número de relatos, buscando cidades que possuam um número superior a 10 relatórios. Enfatizar a cidade, a quantidade de relatos e formato do objeto não identificado.")

california = df_ovnis[df_ovnis['State']=='CA']

COLUNAS = [
    'City',
    'Shape',
    'Quantidade'
]

df_final = pd.DataFrame(columns=COLUNAS)
df_final['City'] = california['City']
df_final['Shape'] = california['Shape']
df_final['Quantidade'] =
california.groupby('City')['City'].transform('count')

#gera arquivo .csv
df_final.to_csv("ovnis_maiores_relatos.csv")

df_final
```

Github: https://github.com/Prof-Fabio-Henrique/pratica-integrada-icd-e-iia-2020-1-g13-mmt

4. Considerações Finais

Com este projeto foi possível responder a uma série de questões com embasamento no webscrapping feito no projeto anterior, abaixo seguem as respostas e dados obtidos.

In [3]:

```
import pandas as pd

df_ovnis = pd.read_csv("ovnis.csv", index_col=[0])

print("1. Quantidade de variáveis que foram coletadas:")
#Conta a quantidade de vdados para cada variavel com a funcao count()
df_ovnis.count()
```

```
1. Quantidade de variáveis que foram coletadas:
```

Out[3]:

```
Date / Time    71901
City           71759
State          67565
Shape          70508
Duration       69939
Summary        71894
Posted         71901
dtype: int64
```

In [4]:

```
print("2. Relatos que ocorreram por estado em ordem decrescente:")
#Usa a funcao groupby() pra agrupar por estado, a funcao count() para con
tar e a funcao sort_values() para ordenar de forma decrescente
df_relatos = df_ovnis
df_relatos = df_relatos.groupby('State').count()
df_relatos = df_relatos.sort_values(ascending=False, by="Posted")['Posted
']
df_relatos
```

```
2. Relatos que ocorreram por estado em ordem decrescente:
```

Out[4]:

```
State
CA    7911
FL    4352
WA    3225
TX    2882
NY    2824
      ...
NF      21
YT      14
PE       9
NT       7
SA       4
Name: Posted, Length: 64, dtype: int64
```

In [5]:

```
print("3. Remover possíveis campos vazios (sem estado):")
```

```
#Exclui toda linha cujo Estado seja nulo com a funcao dropna()
df_ovnis = df_ovnis.dropna(subset=["State"])
df_ovnis
```

3. Remover possíveis campos vazios (sem estado):

Out[5]:

	Date / Time	City	State	Shape	Duration	Summary	Posted
0	9/30/97 20:00	Nova Scotia (Canada)	NS	Light	8-10 seconds.	Single light resembling a star, but moving spu...	10/30/06
1	9/22/97 20:00	Solomons Island	MD	Disk	10 minutes	Close up at twilight, Stationary UFO.	8/5/09
2	9/19/97	Garden Grove	CA	Rectangle	4 mins.	Around 6:30 PM I was walking through a Vons Pa...	12/1/19
3	9/18/97 20:15	Panama City	FL	Unknown	30 seconds	Looked like stars in the sky so far up/moveing...	3/13/12
4	9/15/97 00:00	Houston	TX	Disk	5 minutes	Beautiful silver-colored flying saucer about t...	7/19/10
...	...	...	...	...	...	...	...
71896	8/1/17 02:45	Corcoran	MN	Light	Still going	Small light south west of Minneapolis maneuver...	8/4/17
71897	8/1/17 02:00	Moreno Valley	CA	Other	10 seconds	I was looking out the front windshield and loo...	8/4/17
71898	8/1/17 01:00	Bradenton	FL	Other	<20 seconds	I was walking my dog about 1am on August 1, 20...	5/9/19
71899	8/1/17 00:00	Springdale	AR	NaN	1 hour	Glowing flying people . seven of them flying l...	2/13/20
71900	8/1/17	Laurel	MD	Other	NaN	It was an alien project level 1 federal ran on...	6/25/20

67565 rows × 7 columns

In [7]:

```
print("4. Limitar a análise aos estados dos Estados Unidos.")

#como a planilha nao oferece dados dos paises de origem, o vetor abaixo r
epresenta a sigla dos estados dos EUA
states = ["AL", "AK", "AZ", "AR", "CA", "CO", "CT", "DC", "DE", "FL", "GA
",
          "HI", "ID", "IL", "IN", "IA", "KS", "KY", "LA", "ME", "MD",
          "MA", "MI", "MN", "MS", "MO", "MT", "NE", "NV", "NH", "NJ",
          "NM", "NY", "NC", "ND", "OH", "OK", "OR", "PA", "RI", "SC",
          "SD", "TN", "TX", "UT", "VT", "VA", "WA", "WV", "WI", "WY"]

selection = df_ovnis['State'].isin(states)
df_ovnis = df_ovnis[selection]

df_ovnis
```

4. Limitar a análise aos estados dos Estados Unidos.

Out[7]:

	Date / Time	City	State	Shape	Duration	Summary	Posted
1	9/22/97 20:00	Solomons Island	MD	Disk	10 minutes	Close up at twilight, Stationary UFO.	8/5/09
2	9/19/97	Garden Grove	CA	Rectangle	4 mins.	Around 6:30 PM I was walking through a Vons Pa...	12/1/19
3	9/18/97 20:15	Panama City	FL	Unknown	30 seconds	Looked like stars in the sky so far up/moveing...	3/13/12
4	9/15/97 00:00	Houston	TX	Disk	5 minutes	Beautiful silver-colored flying saucer about t...	7/19/10
5	9/15/97 20:00	Santa Fe	NM	Light	2-3 minutes	Saw white dot of light moving in zig-zag motio...	11/9/17
...	...	...	...	...	...	...	...
71896	8/1/17 02:45	Corcoran	MN	Light	Still going	Small light south west of Minneapolis maneuver...	8/4/17
71897	8/1/17 02:00	Moreno Valley	CA	Other	10 seconds	I was looking out the front windshield and loo...	8/4/17
71898	8/1/17 01:00	Bradenton	FL	Other	<20 seconds	I was walking my dog about 1am on August 1, 20...	5/9/19
71899	8/1/17 00:00	Springdale	AR	NaN	1 hour	Glowing flying people . seven of them flying l...	2/13/20
71900	8/1/17	Laurel	MD	Other	NaN	It was an alien project level 1 federal ran on...	6/25/20

64972 rows × 7 columns

In [ ]:

```
print("5. Consulta por cidades, com o objetivo de saber quais contêm o ma
ior número de relatos (cidades que apresentem ao menos 10 relatos).")

#Usa a funcao groupby() pra agrupar por estado, a funcao count() para con
tar e a funcao sort_values() para ordenar de forma decrescente
cidades = df_ovnis
cidades = cidades.groupby('City').count()
cidades = cidades.sort_values(ascending=False, by="Posted")
cidades = cidades.query('Posted >= 10')['Posted']
cidades
```

```
5. Consulta por cidades, com o objetivo de saber quais contêm o maior núm
ero de relatos (cidades que apresentem ao menos 10 relatos).
```

Out[ ]:

```
City
Phoenix        366
Las Vegas      339
Seattle        324
Portland       318
San Diego      273
               ...
Centreville     10
Hamden          10
```

```
San Dimas       10
La Grande       10
Riverhead       10
Name: Posted, Length: 1490, dtype: int64
```

In [ ]:

```
print("6. Com o dado anterior, responder a seguinte pergunta: por que ser
á que essa é a cidade que possui mais relatos?")
print("Porque é a cidade onde fica mais afastada de grandes metrópoles, c
onstruída no local de antigos canais indigenas")
```

```
6. Com o dado anterior, responder a seguinte pergunta: por que será que e
ssa é a cidade que possui mais relatos?
Porque é a cidade onde fica mais afastada de grandes metrópoles, construí
da no local de antigos canais indigenas
```

In [8]:

```
print("7. Fazer uma query exclusiva para o estado com maior número de rel
atos, buscando cidades que possuam um número superior a 10 relatórios. En
fatizar a cidade, a quantidade de relatos e formato do objeto não identif
icado.")

california = df_ovnis[df_ovnis['State']=='CA']

COLUNAS = [
    'City',
    'Shape',
    'Quantidade'
]

df_final = pd.DataFrame(columns=COLUNAS)
df_final['City'] = california['City']
df_final['Shape'] = california['Shape']
df_final['Quantidade'] = california.groupby('City')['City'].transform('co
unt')

#gera arquivo .csv
df_final.to_csv("ovnis_maiores_relatos.csv")

df_final
```

```
7. Fazer uma query exclusiva para o estado com maior número de relatos, b
uscando cidades que possuam um número superior a 10 relatórios. Enfatizar
a cidade, a quantidade de relatos e formato do objeto não identificado.
```

Out[8]:

	City	Shape	Quantidade
2	Garden Grove	Rectangle	29.0
11	Carlsbad	Light	39.0
13	Milford	Disk	2.0
19	Daggett	Rectangle	1.0
35	El Centro	Oval	14.0
...	...	...	...
71852	Indio	Sphere	24.0
71873	San Bernardino	Flash	39.0
71879	Sun Valley	Sphere	6.0
71892	Newhall	Teardrop	5.0
71897	Moreno Valley	Other	25.0

7911 rows × 3 columns

Ao desenvolver esta aplicação foi possível adquirir novos conhecimentos em ciência de dados, novas bibliotecas na linguagem python e ainda fazer o uso de uma ferramenta de versionamento para completo controle do avanço do projeto entre os desenvolvedores.

Referências

PANDAS. **Https://pandas.pydata.org/docs/**. Disponível em: https://pandas.pydata.org/docs/. Acesso em: 19 set. 2020.

SANTANA, Felipe. **Mapeando Instruções SQL em Comandos Pandas – Operações Fundamentais para todo Cientista de Dados.** Disponível em: https://minerandodados.com.br/mapeando-instrucoes-sql-em-comandos-pandas-operacoes-fundamentais-para-todo-cientista-de-dados/. Acesso em: 19 set. 2020.

SANTANA, Rodrigo. **Dominando o Pandas: A Biblioteca para Análise de Dados preferida entre os Cientistas de Dados (Parte 1)**. Disponível em: https://minerandodados.com.br/python-para-analise-de-dados/. Acesso em: 19 set. 2020.

SANTANA, Rodrigo. **Análise de Dados com Python usando Pandas.** Disponível em: https://minerandodados.com.br/analise-de-dados-com-python-usando-pandas/. Acesso em: 19 set. 2020.