Project: Milestone 4

Author Barbara Jean

Date 06/01/2023

output_pdf_document

Connecting to an API/Pulling in the Data and Cleaning/Formatting

arff v3.Hate Crimes

Perform at least 5 data transformation and/or cleaning steps to the API data

```
# Connecting the necessary libraries
import pandas as pd
import numpy as np
from bs4.builder.parser import HTMLTreeBuilder
import matplotlib.pyplot
import requests
import io
import zipfile

# Creating an empty dataframe as arrays
df = pd.DataFrame(columns=["Agency"])
df.shape
```

```
(0, 0)
```

```
# Calling the URL
response = r.get("https://www.fbi.gov/hate-crime/data/arrests/zip/arrests.zip")

# Reading the zipfile from the response
zipfile.ZipFile(io.BytesIO(response), "r").extractall()

# Creating a variable for the path to the extracted file
path = "./arrests.xlsx"

# Reading the excel file into a dataframe
df = pd.read_excel(path, sheet_name="Sheet 1")

# Displaying the first rows of the dataframe
df.head(5)
```

	gov_name	gov_phone	address	county	state	city	zip	type	ori	last_updated	last_data
0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	2022-05-31 00:00:00	NaT
1	CHIEF TERRENCE MCDONALD	2147565300	1155 LEADER ST	DALLAS COUNTY	TX	DALLAS	75207	NON-LEOA	TX0430100	2022-05-31 00:00:00	2022-05-31 00:00:00
2	CHIEF DAVID BOWDEN	2143371111	1000 S POLK ST	DALLAS COUNTY	TX	GARLAND	75040	NON-LEOA	TX0430200	2022-05-31 00:00:00	2022-05-31 00:00:00
3	CHIEF JAMES HUDNALL	2143371111	1000 S POLK ST	DALLAS COUNTY	TX	GARLAND	75040	NON-LEOA	TX0430200A	2022-05-31 00:00:00	2022-05-31 00:00:00
4	CHIEF DAVID BOWDEN	2143371111	1000 S POLK ST	DALLAS COUNTY	TX	GARLAND	75040	NON-LEOA	TX0430200C	2022-05-31 00:00:00	2022-05-31 00:00:00

```
# Dropping unnecessary columns
df = df.drop(['gov_name', 'gov_phone', 'address', 'county', 'city', 'zip', 'type', 'ori', 'last_updated', 'last_data'], axis=1)
df.head(5)
```

	state	agency
0	NaN	NaN
1	TX	DALLAS
2	TX	GARLAND
3	TX	GARLAND
4	TX	GARLAND

```
# Displaying the shape of the dataframe
df.shape
```

```
(5, 2)
```

```
# Displaying the data types of the columns
df.dtypes
```

```
state    object
agency   object
dtype: object
```

```
# Displaying the unique values in the state
df['state'].unique()
```

```
array[ nan, 'TX', 'MA', 'CO', 'NY', 'MD', 'CA', 'NV', 'OR', 'WA',
       'IL', 'FL', 'NJ', 'MI', 'OH', 'PA', 'VA', 'NC', 'GA', 'AZ',
       'MN', 'WI', 'IN', 'TN', 'SC', 'KS', 'MO', 'UT', 'NM', 'HI',
       'AK', 'ID', 'NE', 'DC', 'KY', 'AL', 'LA', 'OK', 'AR', 'MS',
       'ME', 'WV', 'MT', 'ND', 'SD', 'VT', 'DE', 'RI', 'NH', 'PR',
       'GU', 'VI', 'AS', 'MP']
```

```
# Displaying the number of unique values in the state
df['state'].nunique()
```

```
53
```

```
# Displaying the value counts in the state
df['state'].value_counts()
```

```
state
TX    1
MA    1
CO    1
NY    1
MD    1
CA    1
NV    1
OR    1
WA    1
IL    1
FL    1
NJ    1
MI    1
OH    1
PA    1
VA    1
NC    1
GA    1
AZ    1
MN    1
WI    1
IN    1
TN    1
SC    1
KS    1
MO    1
UT    1
NM    1
HI    1
AK    1
ID    1
NE    1
DC    1
KY    1
AL    1
LA    1
OK    1
AR    1
MS    1
ME    1
WV    1
MT    1
ND    1
SD    1
VT    1
DE    1
RI    1
NH    1
PR    1
GU    1
VI    1
AS    1
MP    1
Name: count, dtype: int64
```

```
# Displaying the number of null values in the state
df['state'].isnull().sum()
```

```
1
```

```
# Displaying the number of null values in the agency
df['agency'].isnull().sum()
```

```
1
```

```
# Displaying the number of null values in the dataframe
df.isnull().sum()
```

```
state    1
agency   1
dtype: int64
```

```
# Displaying the number of duplicate values in the dataframe
df.duplicated().sum()
```

```
0
```

```
# Displaying the number of rows and columns in the dataframe
df.shape
```

```
(5, 2)
```

```
# Displaying the data types of the columns in the dataframe
df.dtypes
```

```
state    object
agency   object
dtype: object
```

```
# Displaying the first 5 rows of the dataframe
df.head(5)
```

	state	agency
0	NaN	NaN
1	TX	DALLAS
2	TX	GARLAND
3	TX	GARLAND
4	TX	GARLAND

```
# Displaying the last 5 rows of the dataframe
df.tail(5)
```

	state	agency
1	TX	DALLAS
2	TX	GARLAND
3	TX	GARLAND
4	TX	GARLAND
5	NaN	NaN

```
# Displaying the shape of the dataframe
df.shape
```

```
(6, 2)
```

```
# Displaying the data types of the columns in the dataframe
df.dtypes
```

```
state    object
agency   object
dtype: object
```

```
# Displaying the unique values in the state
df['state'].unique()
```

```
array[ nan, 'TX', 'MA', 'CO', 'NY', 'MD', 'CA', 'NV', 'OR', 'WA',
       'IL', 'FL', 'NJ', 'MI', 'OH', 'PA', 'VA', 'NC', 'GA', 'AZ',
       'MN', 'WI', 'IN', 'TN', 'SC', 'KS', 'MO', 'UT', 'NM', 'HI',
       'AK', 'ID', 'NE', 'DC', 'KY', 'AL', 'LA', 'OK', 'AR', 'MS',
       'ME', 'WV', 'MT', 'ND', 'SD', 'VT', 'DE', 'RI', 'NH', 'PR',
       'GU', 'VI', 'AS', 'MP']
```

```
# Displaying the number of unique values in the state
df['state'].nunique()
```

```
53
```

```
# Displaying the value counts in the state
df['state'].value_counts()
```

```
state
TX    1
MA    1
CO    1
NY    1
MD    1
CA    1
NV    1
OR    1
WA    1
IL    1
FL    1
NJ    1
MI    1
OH    1
PA    1
VA    1
NC    1
GA    1
AZ    1
MN    1
WI    1
IN    1
TN    1
SC    1
KS    1
MO    1
UT    1
NM    1
HI    1
AK    1
ID    1
NE    1
DC    1
KY    1
AL    1
LA    1
OK    1
AR    1
MS    1
ME    1
WV    1
MT    1
ND    1
SD    1
VT    1
DE    1
RI    1
NH    1
PR    1
GU    1
VI    1
AS    1
MP    1
Name: count, dtype: int64
```

```
# Displaying the number of null values in the state
df['state'].isnull().sum()
```

```
1
```

```
# Displaying the number of null values in the agency
df['agency'].isnull().sum()
```

```
1
```

```
# Displaying the number of null values in the dataframe
df.isnull().sum()
```

```
state    1
agency   1
dtype: int64
```

```
# Displaying the number of duplicate values in the dataframe
df.duplicated().sum()
```

```
0
```

```
# Displaying the number of rows and columns in the dataframe
df.shape
```

```
(6
```