109550198 卜銳凱

Introduction to Data Science

HW3

Report

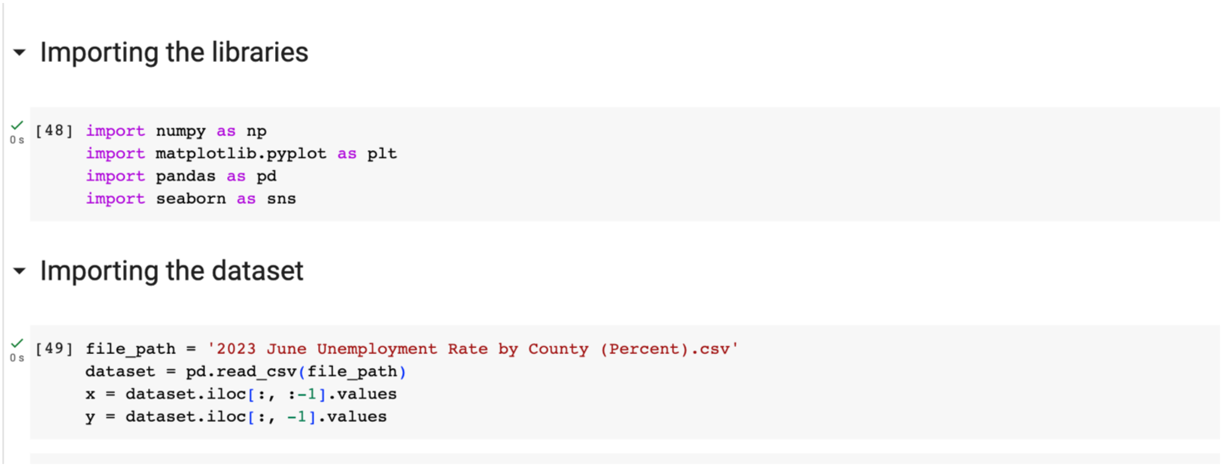
This report is a combination of HW2 analysis and HW3 analysis, with HW3 focusing on data cleaning. This report describes the analysis of the dataset from the CSV file "2023 June Unemployment Rate by County (Percent).csv." To guarantee data quality and completeness, the dataset was loaded from a CSV file, and different data analysis and manipulation operations were performed.

Code Summary:

The provided code accomplishes the following tasks:

Data Loading: The code starts by importing necessary Python libraries such as NumPy, Matplotlib, Pandas, and Seaborn. These libraries are frequently employed in data analysis and visualization.

The dataset is loaded into a Pandas DataFrame named 'dataset' from the file '2023 June Unemployment Rate by County (Percent).csv'. This first step is critical for gaining access to and modifying the data.



Importing the libraries

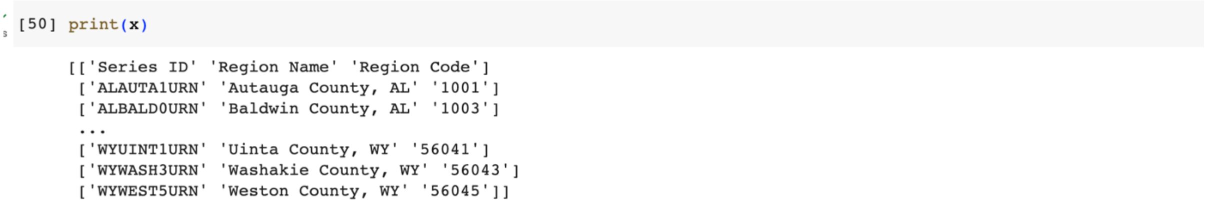
```
[48] import numpy as np
     import matplotlib.pyplot as plt
     import pandas as pd
     import seaborn as sns
```

Importing the dataset

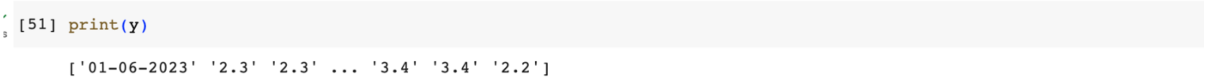
```
[49] file_path = '2023 June Unemployment Rate by County (Percent).csv'
     dataset = pd.read_csv(file_path)
     x = dataset.iloc[:, :-1].values
     y = dataset.iloc[:, -1].values
```

Data Splitting:

It divides the dataset into two arrays, x, and y, where x contains all except the last column and y contains the last column. This is commonly done to differentiate between features (independent variables) and the desired variable (dependent variable).



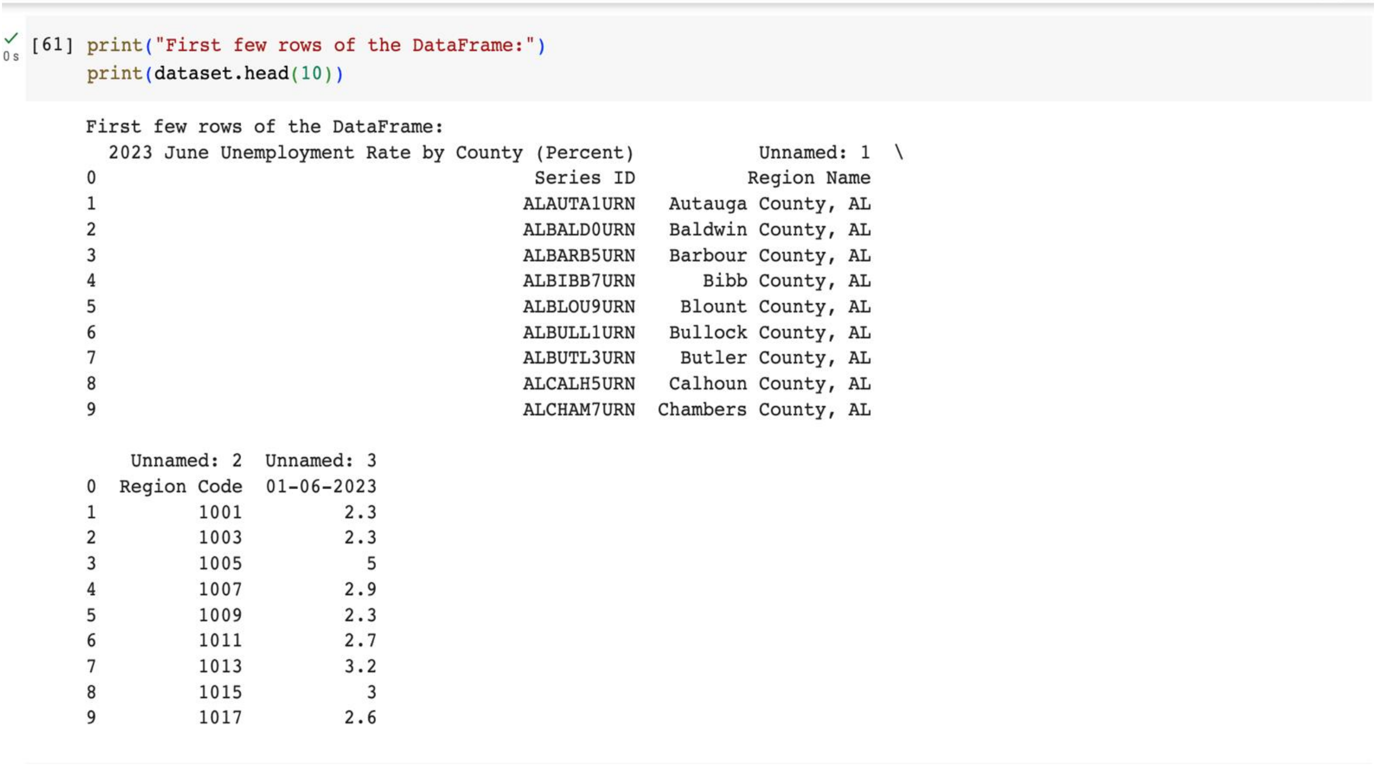
```
[50] print(x)

    [['Series ID' 'Region Name' 'Region Code']
     ['ALAUTA1URN' 'Autauga County, AL' '1001']
     ['ALBALD0URN' 'Baldwin County, AL' '1003']
     ...
     ['WYUINT1URN' 'Uinta County, WY' '56041']
     ['WYWASH3URN' 'Washakie County, WY' '56043']
     ['WYWEST5URN' 'Weston County, WY' '56045']]
```



```
[51] print(y)

    ['01-06-2023' '2.3' '2.3' ... '3.4' '3.4' '2.2']
```

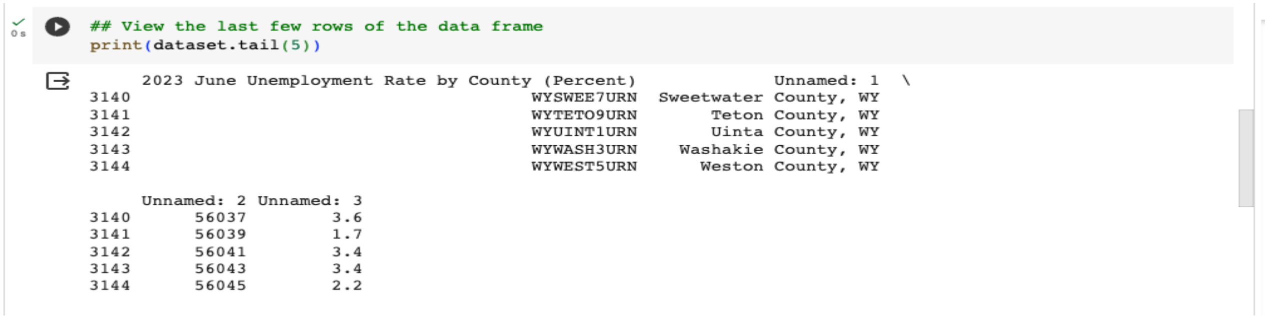
Data Exploration:

It shows the first ten rows of the Data Frame, providing an overview of the data's structure and content.



```
[61] print("First few rows of the DataFrame:")
     print(dataset.head(10))

     First few rows of the DataFrame:
       2023 June Unemployment Rate by County (Percent)           Unnamed: 1  \
     0                                       Series ID          Region Name
     1                                      ALAUTA1URN   Autauga County, AL
     2                                      ALBALD0URN   Baldwin County, AL
     3                                      ALBARB5URN   Barbour County, AL
     4                                      ALBIBB7URN      Bibb County, AL
     5                                      ALBLOU9URN    Blount County, AL
     6                                      ALBULL1URN   Bullock County, AL
     7                                      ALBUTL3URN    Butler County, AL
     8                                      ALCALH5URN   Calhoun County, AL
     9                                      ALCHAM7URN  Chambers County, AL

         Unnamed: 2  Unnamed: 3
     0  Region Code  01-06-2023
     1         1001         2.3
     2         1003         2.3
     3         1005           5
     4         1007         2.9
     5         1009         2.3
     6         1011         2.7
     7         1013         3.2
     8         1015           3
     9         1017         2.6
```

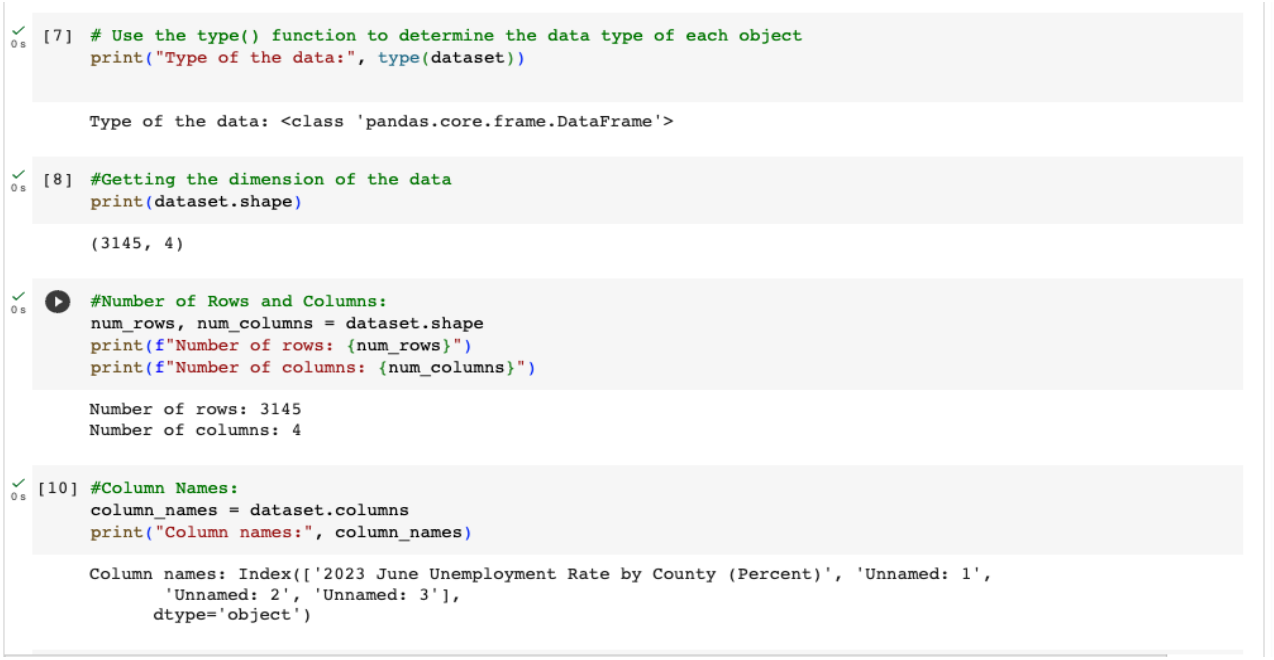
Displays the last 5 rows using dataset.tail(5) to check the data's end.



```
## View the last few rows of the data frame
print(dataset.tail(5))
```

```
     2023 June Unemployment Rate by County (Percent)             Unnamed: 1  \
3140                                       WYSWEE7URN  Sweetwater County, WY   
3141                                       WYTETO9URN       Teton County, WY   
3142                                       WYUINT1URN       Uinta County, WY   
3143                                       WYWASH3URN    Washakie County, WY   
3144                                       WYWEST5URN      Weston County, WY   

     Unnamed: 2 Unnamed: 3  
3140      56037        3.6  
3141      56039        1.7  
3142      56041        3.4  
3143      56043        3.4  
3144      56045        2.2  
```

Retrieves the column names using dataset.columns to understand the variables included in the dataset.
Generates summary statistics using dataset.describe(), which provides insights into the central tendencies and distributions of numerical columns.
Uses dataset.info() to obtain information about data types and non-null counts.



```
[7] # Use the type() function to determine the data type of each object
    print("Type of the data:", type(dataset))
```

```
Type of the data: <class 'pandas.core.frame.DataFrame'>
```

```
[8] #Getting the dimension of the data
    print(dataset.shape)
```

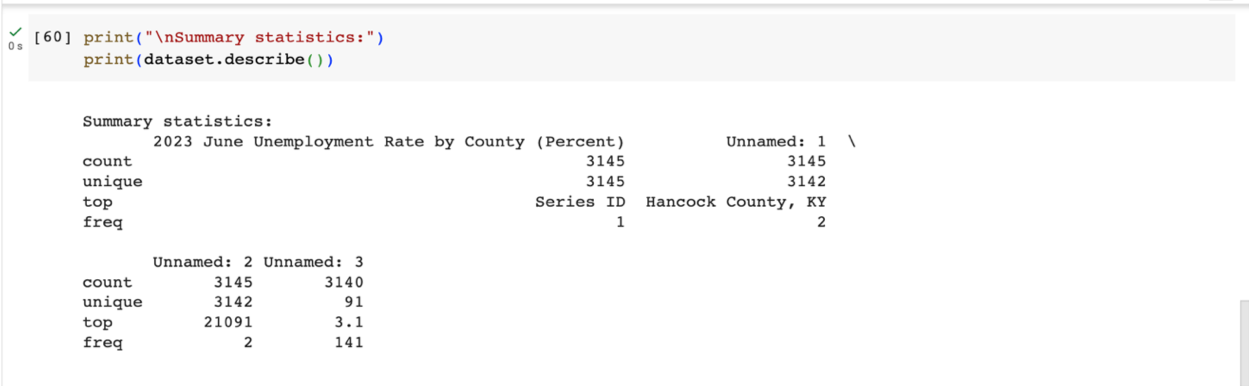
```
(3145, 4)
```

```
#Number of Rows and Columns:
num_rows, num_columns = dataset.shape
print(f"Number of rows: {num_rows}")
print(f"Number of columns: {num_columns}")
```

```
Number of rows: 3145
Number of columns: 4
```

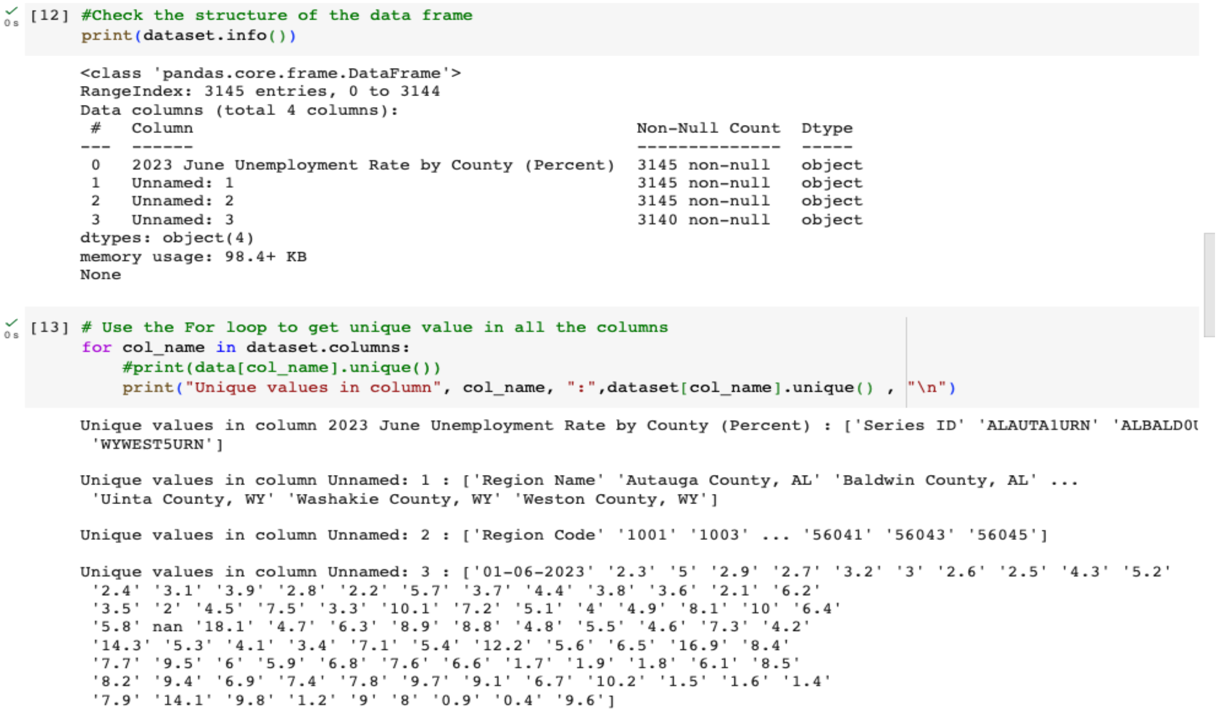
```
[10] #Column Names:
     column_names = dataset.columns
     print("Column names:", column_names)
```

```
Column names: Index(['2023 June Unemployment Rate by County (Percent)', 'Unnamed: 1',
       'Unnamed: 2', 'Unnamed: 3'],
      dtype='object')
```



```
[60] print("\nSummary statistics:")
     print(dataset.describe())
```

```

Summary statistics:
       2023 June Unemployment Rate by County (Percent)          Unnamed: 1  \
count                                             3145                3145   
unique                                            3145                3142   
top                                          Series ID  Hancock County, KY   
freq                                                 1                   2   

       Unnamed: 2 Unnamed: 3  
count        3145       3140  
unique       3142         91  
top         21091        3.1  
freq            2        141  
```



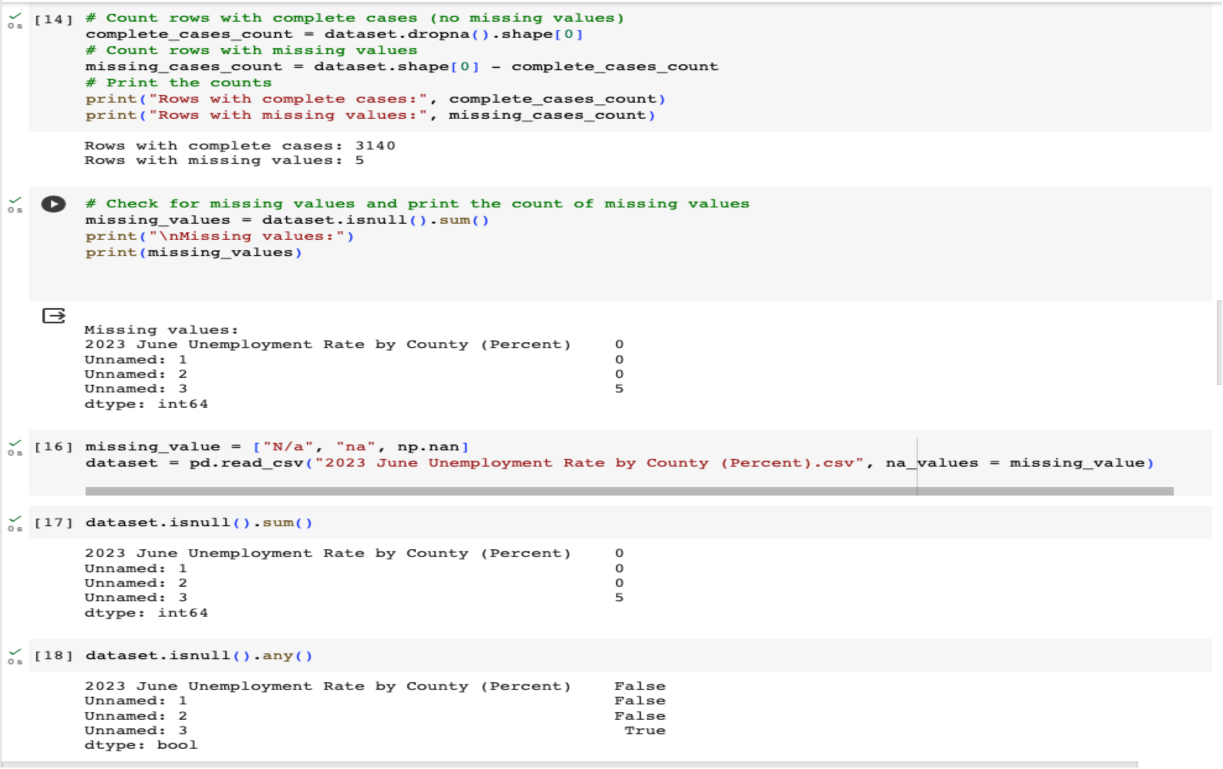
```
[12] #Check the structure of the data frame
     print(dataset.info())

     <class 'pandas.core.frame.DataFrame'>
     RangeIndex: 3145 entries, 0 to 3144
     Data columns (total 4 columns):
      #   Column                                       Non-Null Count  Dtype
     ---  ------                                       --------------  -----
      0   2023 June Unemployment Rate by County (Percent)  3145 non-null   object
      1   Unnamed: 1                                   3145 non-null   object
      2   Unnamed: 2                                   3145 non-null   object
      3   Unnamed: 3                                   3140 non-null   object
     dtypes: object(4)
     memory usage: 98.4+ KB
     None
```

```
[13] # Use the For loop to get unique value in all the columns
     for col_name in dataset.columns:
         #print(data[col_name].unique())
         print("Unique values in column", col_name, ":",dataset[col_name].unique() , "\n")

     Unique values in column 2023 June Unemployment Rate by County (Percent) : ['Series ID' 'ALAUTA1URN' 'ALBALD0U
      'WYWEST5URN']

     Unique values in column Unnamed: 1 : ['Region Name' 'Autauga County, AL' 'Baldwin County, AL' ...
      'Uinta County, WY' 'Washakie County, WY' 'Weston County, WY']

     Unique values in column Unnamed: 2 : ['Region Code' '1001' '1003' ... '56041' '56043' '56045']

     Unique values in column Unnamed: 3 : ['01-06-2023' '2.3' '5' '2.9' '2.7' '3.2' '3' '2.6' '2.5' '4.3' '5.2'
      '2.4' '3.1' '3.9' '2.8' '2.2' '5.7' '3.7' '4.4' '3.8' '3.6' '2.1' '6.2'
      '3.5' '2' '4.5' '7.5' '3.3' '10.1' '7.2' '5.1' '4' '4.9' '8.1' '10' '6.4'
      '5.8' nan '18.1' '4.7' '6.3' '8.9' '8.8' '4.8' '5.5' '4.6' '7.3' '4.2'
      '14.3' '5.3' '4.1' '3.4' '7.1' '5.4' '12.2' '5.6' '6.5' '16.9' '8.4'
      '7.7' '9.5' '6' '5.9' '6.8' '7.6' '6.6' '1.7' '1.9' '1.8' '6.1' '8.5'
      '8.2' '9.4' '6.9' '7.4' '7.8' '9.7' '9.1' '6.7' '10.2' '1.5' '1.6' '1.4'
      '7.9' '14.1' '9.8' '1.2' '9' '8' '0.9' '0.4' '9.6']
```

Missing Values Handling:

Missing values are common in real-world datasets and handling them is critical to ensuring that subsequent analyses are accurate. To handle missing values, the code does the following:

Uses 'dataset.isnull().sum()' to check for missing values and reports the number of missing values in each column.
To highlight the pattern of missing values across columns, a heatmap built with Seaborn's sns.heatmap() is used to visualize the missing data.



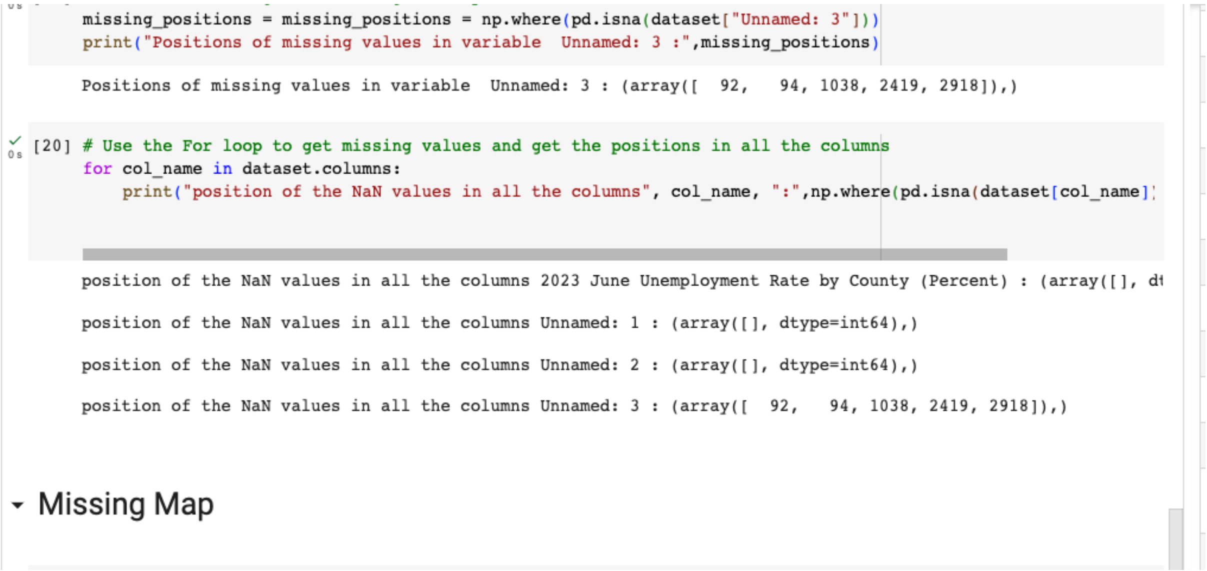
```
[14] # Count rows with complete cases (no missing values)
     complete_cases_count = dataset.dropna().shape[0]
     # Count rows with missing values
     missing_cases_count = dataset.shape[0] - complete_cases_count
     # Print the counts
     print("Rows with complete cases:", complete_cases_count)
     print("Rows with missing values:", missing_cases_count)

     Rows with complete cases: 3140
     Rows with missing values: 5
```

```
     # Check for missing values and print the count of missing values
     missing_values = dataset.isnull().sum()
     print("\nMissing values:")
     print(missing_values)

     Missing values:
     2023 June Unemployment Rate by County (Percent)    0
     Unnamed: 1                                         0
     Unnamed: 2                                         0
     Unnamed: 3                                         5
     dtype: int64
```

```
[16] missing_value = ["N/a", "na", np.nan]
     dataset = pd.read_csv("2023 June Unemployment Rate by County (Percent).csv", na_values = missing_value)
```

```
[17] dataset.isnull().sum()

     2023 June Unemployment Rate by County (Percent)    0
     Unnamed: 1                                         0
     Unnamed: 2                                         0
     Unnamed: 3                                         5
     dtype: int64
```

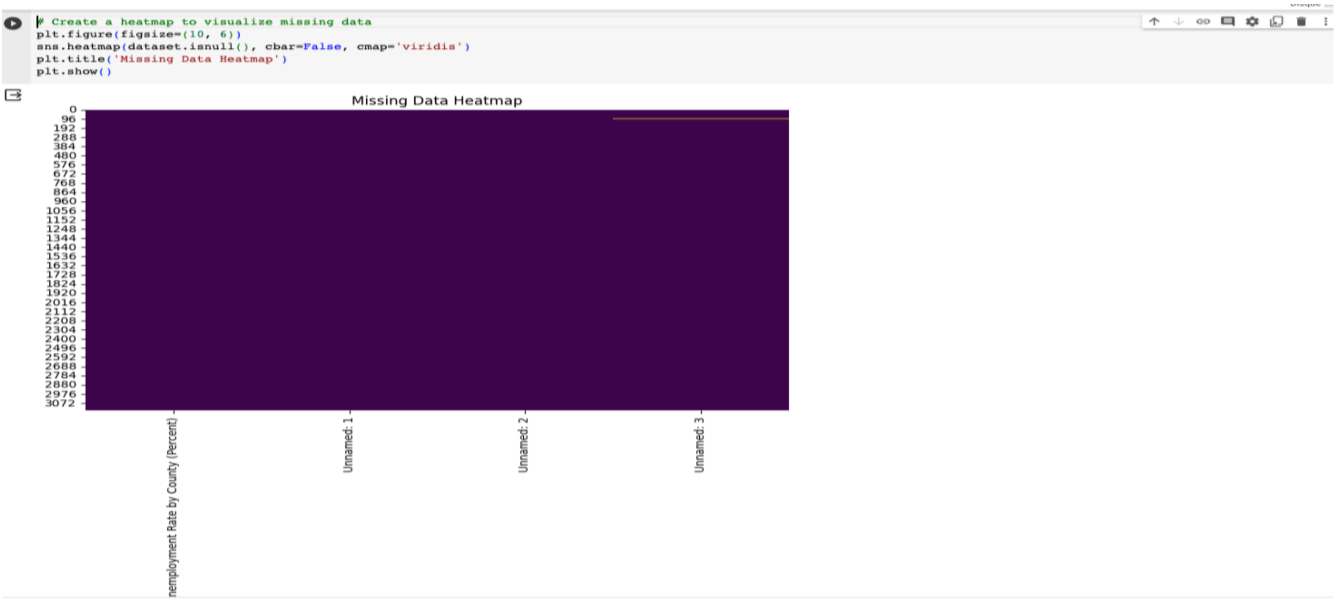
```
[18] dataset.isnull().any()

     2023 June Unemployment Rate by County (Percent)    False
     Unnamed: 1                                         False
     Unnamed: 2                                         False
     Unnamed: 3                                          True
     dtype: bool
```



```
     missing_positions = missing_positions = np.where(pd.isna(dataset["Unnamed: 3"]))
     print("Positions of missing values in variable  Unnamed: 3 :",missing_positions)

     Positions of missing values in variable  Unnamed: 3 : (array([  92,   94, 1038, 2419, 2918]),)
```

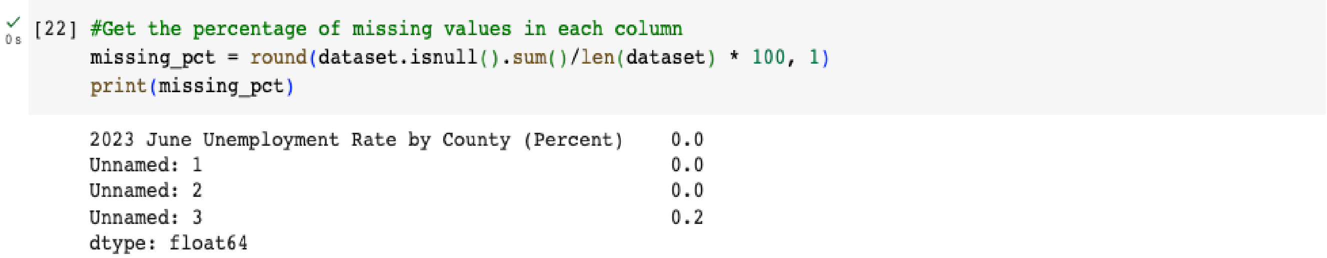
```
[20] # Use the For loop to get missing values and get the positions in all the columns
     for col_name in dataset.columns:
         print("position of the NaN values in all the columns", col_name, ":",np.where(pd.isna(dataset[col_name])

     position of the NaN values in all the columns 2023 June Unemployment Rate by County (Percent) : (array([], d

     position of the NaN values in all the columns Unnamed: 1 : (array([], dtype=int64),)

     position of the NaN values in all the columns Unnamed: 2 : (array([], dtype=int64),)

     position of the NaN values in all the columns Unnamed: 3 : (array([  92,   94, 1038, 2419, 2918]),)
```

Missing Map



```
# Create a heatmap to visualize missing data
plt.figure(figsize=(10, 6))
sns.heatmap(dataset.isnull(), cbar=False, cmap='viridis')
plt.title('Missing Data Heatmap')
plt.show()
```

Calculates and displays the percentage of missing values in each column using round(dataset.isnull().sum() / len(dataset) * 100, 1).

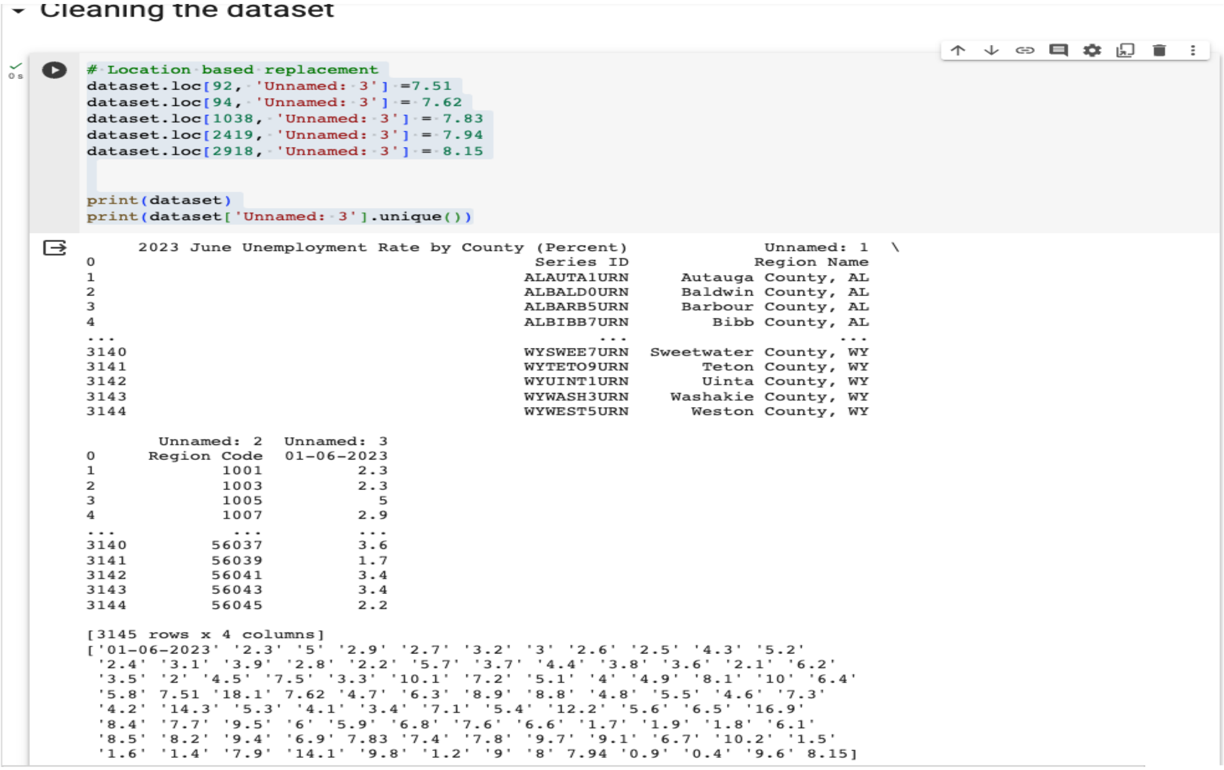


```
[22] #Get the percentage of missing values in each column
     missing_pct = round(dataset.isnull().sum()/len(dataset) * 100, 1)
     print(missing_pct)
```

```
2023 June Unemployment Rate by County (Percent)    0.0
Unnamed: 1                                         0.0
Unnamed: 2                                         0.0
Unnamed: 3                                         0.2
dtype: float64
```

Data cleaning:

The code snippet below demonstrates the location-based replacement of values in the 'Unnamed: 3' column:



```
# Location based replacement
dataset.loc[92, 'Unnamed: 3'] =7.51
dataset.loc[94, 'Unnamed: 3'] = 7.62
dataset.loc[1038, 'Unnamed: 3'] = 7.83
dataset.loc[2419, 'Unnamed: 3'] = 7.94
dataset.loc[2918, 'Unnamed: 3'] = 8.15


print(dataset)
print(dataset['Unnamed: 3'].unique())
```

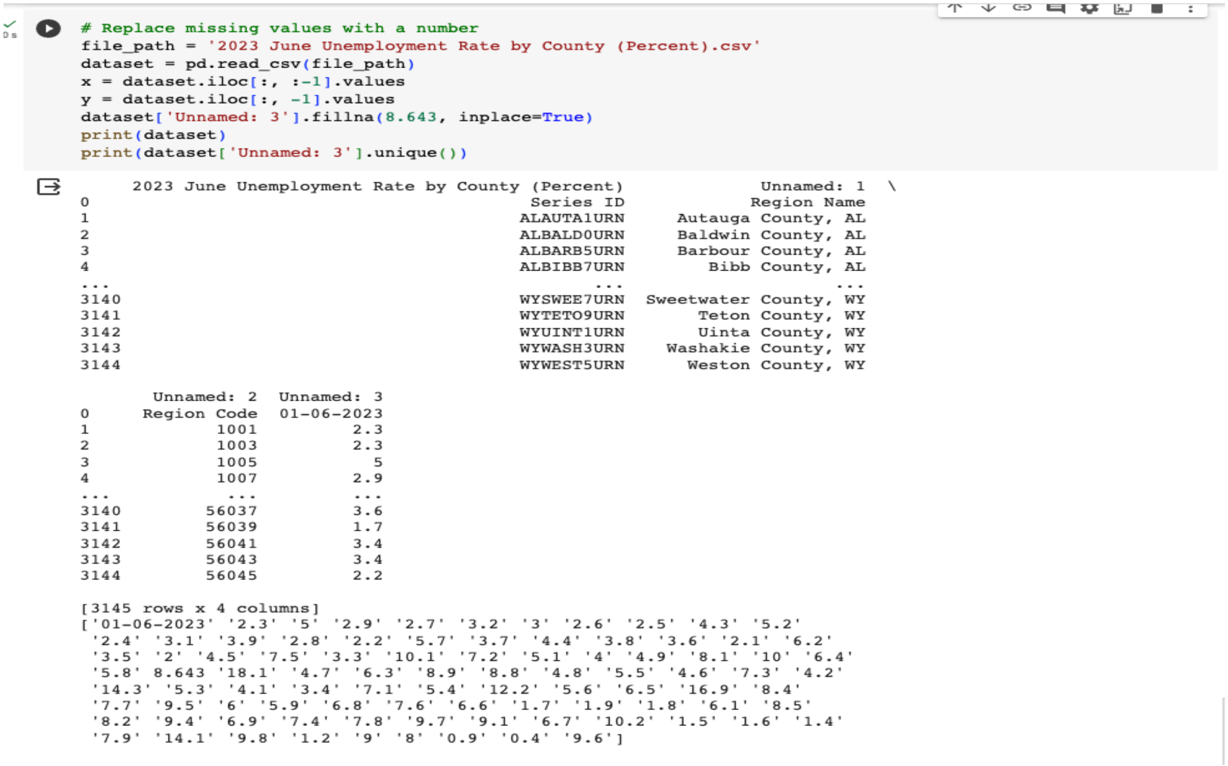
```
      2023 June Unemployment Rate by County (Percent)           Unnamed: 1  \
0                                           Series ID          Region Name
1                                          ALAUTA1URN   Autauga County, AL
2                                          ALBALD0URN   Baldwin County, AL
3                                          ALBARB5URN   Barbour County, AL
4                                          ALBIBB7URN      Bibb County, AL
...                                               ...                  ...
3140                                       WYSWEE7URN  Sweetwater County, WY
3141                                       WYTETO9URN     Teton County, WY
3142                                       WYUINT1URN     Uinta County, WY
3143                                       WYWASH3URN  Washakie County, WY
3144                                       WYWEST5URN    Weston County, WY

       Unnamed: 2  Unnamed: 3
0     Region Code  01-06-2023
1            1001         2.3
2            1003         2.3
3            1005           5
4            1007         2.9
...           ...         ...
3140        56037         3.6
3141        56039         1.7
3142        56041         3.4
3143        56043         3.4
3144        56045         2.2

[3145 rows x 4 columns]
['01-06-2023' '2.3' '5' '2.9' '2.7' '3.2' '3' '2.6' '2.5' '4.3' '5.2'
 '2.4' '3.1' '3.9' '2.8' '2.2' '5.7' '3.7' '4.4' '3.8' '3.6' '2.1' '6.2'
 '3.5' '2' '4.5' '7.5' '3.3' '10.1' '7.2' '5.1' '4' '4.9' '8.1' '10' '6.4'
 '5.8' 7.51 '18.1' 7.62 '4.7' '6.3' '8.9' '8.8' '4.8' '5.5' '4.6' '7.3'
 '4.2' '14.3' '5.3' '4.1' '3.4' '7.1' '5.4' '12.2' '5.6' '6.5' '16.9'
 '8.4' '7.7' '9.5' '6' '5.9' '6.8' '7.6' '6.6' '1.7' '1.9' '1.8' '6.1'
 '8.5' '8.2' '9.4' '6.9' 7.83 '7.4' '7.8' '9.7' '9.1' '6.7' '10.2' '1.5'
 '1.6' '1.4' '7.9' '14.1' '9.8' '1.2' '9' '8' 7.94 '0.9' '0.4' '9.6' 8.15]
```

At row index 92, the value in the 'Unnamed: 3' column was replaced with 7.51.
At row index 94, the value in the 'Unnamed: 3' column was replaced with 7.62.
At row index 1038, the value in the 'Unnamed: 3' column was replaced with 7.83.
At row index 2419, the value in the 'Unnamed: 3' column was replaced with 7.94.
At row index 2918, the value in the 'Unnamed: 3' column was replaced with 8.15.

Following the replacement, the changed dataset was printed using print(dataset) to display the updated values. To check the changes, the unique values in the 'Unnamed: 3' column were displayed with print(dataset['Unnamed: 3'].unique()).

The provided code snippet demonstrates the process of replacing missing values:

The .fillna(8.643, inplace=True) method is applied to the 'Unnamed: 3' column of the 'dataset' DataFrame. This method replaces any missing values in the specified column with the numeric value 8.643. The inplace=True argument ensures that the changes are made directly to the 'dataset' DataFrame.

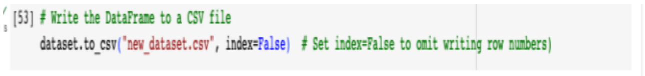


```
# Replace missing values with a number
file_path = '2023 June Unemployment Rate by County (Percent).csv'
dataset = pd.read_csv(file_path)
x = dataset.iloc[:, :-1].values
y = dataset.iloc[:, -1].values
dataset['Unnamed: 3'].fillna(8.643, inplace=True)
print(dataset)
print(dataset['Unnamed: 3'].unique())
```

```
     2023 June Unemployment Rate by County (Percent)           Unnamed: 1  \
0                                          Series ID          Region Name   
1                                        ALAUTA1URN     Autauga County, AL   
2                                        ALBALD0URN     Baldwin County, AL   
3                                        ALBARB5URN     Barbour County, AL   
4                                        ALBIBB7URN        Bibb County, AL   
...                                              ...                  ...   
3140                                     WYSWEE7URN  Sweetwater County, WY   
3141                                     WYTETO9URN       Teton County, WY   
3142                                     WYUINT1URN       Uinta County, WY   
3143                                     WYWASH3URN    Washakie County, WY   
3144                                     WYWEST5URN      Weston County, WY   

       Unnamed: 2  Unnamed: 3  
0     Region Code  01-06-2023  
1            1001         2.3  
2            1003         2.3  
3            1005           5  
4            1007         2.9  
...           ...         ...  
3140        56037         3.6  
3141        56039         1.7  
3142        56041         3.4  
3143        56043         3.4  
3144        56045         2.2  

[3145 rows x 4 columns]
['01-06-2023' '2.3' '5' '2.9' '2.7' '3.2' '3' '2.6' '2.5' '4.3' '5.2'
 '2.4' '3.1' '3.9' '2.8' '2.2' '5.7' '3.7' '4.4' '3.8' '3.6' '2.1' '6.2'
 '3.5' '2' '4.5' '7.5' '3.3' '10.1' '7.2' '5.1' '4' '4.9' '8.1' '10' '6.4'
 '5.8' 8.643 '18.1' '4.7' '6.3' '8.9' '8.8' '4.8' '5.5' '4.6' '7.3' '4.2'
 '14.3' '5.3' '4.1' '3.4' '7.1' '5.4' '12.2' '5.6' '6.5' '16.9' '8.4'
 '7.7' '9.5' '6' '5.9' '6.8' '7.6' '6.6' '1.7' '1.9' '1.8' '6.1' '8.5'
 '8.2' '9.4' '6.9' '7.4' '7.8' '9.7' '9.1' '6.7' '10.2' '1.5' '1.6' '1.4'
 '7.9' '14.1' '9.8' '1.2' '9' '8' '0.9' '0.4' '9.6']
```

Data Export:

To store the cleaned and changed dataset for further analysis, the code uses dataset.to_csv("new_dataset.csv", index=False) to save the modified DataFrame to a new CSV file named 'new_dataset.csv'.



```
[53] # Write the DataFrame to a CSV file
     dataset.to_csv("new_dataset.csv", index=False)  # Set index=False to omit writing row numbers)
```

Discuss possible problems you plan to investigate for future studies.

For future studies, I plan to investigate:

Exploratory Data Analysis (EDA): To acquire deeper insights, try undertaking more complete EDA, such as investigating connections between variables, displaying distributions, and detecting outliers.

Data Visualization: Extend the visualization capabilities to incorporate different sorts of plots, such as histograms, box plots, scatter plots, or time series plots, to expose new insights in the data.

Check weather Simpson's paradox exists in your dataset and explain what you find and why you choose these visualization methods.

Finally, this report highlights the critical procedures involved in preparing a dataset comprising unemployment rates by county. This function prepares the data for future analysis, visualization, or modeling by importing the data, investigating its features, addressing missing values, and exporting the cleaned dataset.