Code Documentation

Overview

This Python script analyzes and visualizes the top 4 scoring minutes in a dataset of football goalscorers. It uses matplotlib for visualization and pandas for data manipulation, with a focus on generating a pie chart and a horizontal bar chart for comparison.

Code Explanation

1. Importing Necessary Libraries

```
import matplotlib.pyplot as plt
import pandas as pd
from collections import Counter
```

- matplotlib.pyplot: For creating visualizations such as pie and bar charts.
- pandas: For data manipulation and analysis.
- collections.Counter: For counting occurrences of scoring minutes.

2. Setting Plot Style

```
plt.style.use('fivethirtyeight')
```

- This applies the 'fivethirtyeight' style to the plots, giving them a clean and professional appearance.

3. Loading and Cleaning Data

```
data = pd.read_csv("Data(file_format_CSV)\Goalscorers.csv")
Scoring_Minutes_Data = list((data['minute'].dropna().fillna("No Goals")))
```

- Reads the CSV file containing the data.
- dropna(): Removes any rows with missing values in the 'minute' column.
- fillna("No Goals"): Fills any remaining missing values with the string 'No Goals'.
- The 'minute' column is then converted into a list for further processing.

4. Counting and Extracting Top Scoring Minutes

```
Count_Scoring_Minutes = Counter(Scoring_Minutes_Data)
Top_Scoring_minutes = Count_Scoring_Minutes.most_common(4)
```

- Counter: Counts the occurrences of each scoring minute.
- most_common(4): Retrieves the 4 most common scoring minutes.

5. Preparing Data for Visualization

```
Scoring_Minutes = [str(items[0]) for items in Top_Scoring_minutes]
No_of_Goals = [items[1] for items in Top_Scoring_minutes]
```

- Scoring_Minutes: List of the top 4 scoring minutes as strings.
- No_of_Goals: Corresponding list of the number of goals for each top scoring minute.

6. Creating Subplots for Visualizations

```
fig, axs = plt.subplots(1, 2, figsize=(18, 12))
explode = [0.04, 0, 0, 0]
```

- Creates a figure with 2 subplots arranged side by side.
- figsize=(18, 12): Sets the dimensions of the entire figure.
- explode: Highlights the first segment in the pie chart for better visibility.

7. Pie Chart

```
axs[0].pie(No_of_Goals, labels=Scoring_Minutes, explode=explode,
autopct="%1.1f%%")
axs[0].legend(loc='lower left')
axs[0].set_title('Top 4 Scoring Minutes of All Time(1872 - 2024)\nPie Chart')
```

- axs[0].pie: Creates a pie chart on the first subplot.
- explode: Highlights the first segment.
- autopct="%1.1f%%": Displays the percentage for each segment.
- legend: Adds a legend for labels.
- set_title: Sets the title for the pie chart.

8. Horizontal Bar Chart

```
Scoring_Minutes.reverse()
No_of_Goals.reverse()
axs[1].barh(Scoring_Minutes, No_of_Goals, color='skyblue')
axs[1].set_title('Top 4 Scoring Minutes of All Time (1872 - 2024)\nBar Chart')
axs[1].set_xlabel("Number of Goals")
axs[1].set_ylabel("Time of Goal")
```

- Reversing Data: The data is reversed to display the largest values at the top in the bar chart.
- axs[1].barh: Creates a horizontal bar chart on the second subplot.
- set_xlabel: Adds a label for the x-axis representing the number of goals.
- set_ylabel: Adds a label for the y-axis representing the scoring minutes.

9. Displaying the Plots

```
plt.show()
```

- Renders and displays the created visualizations.

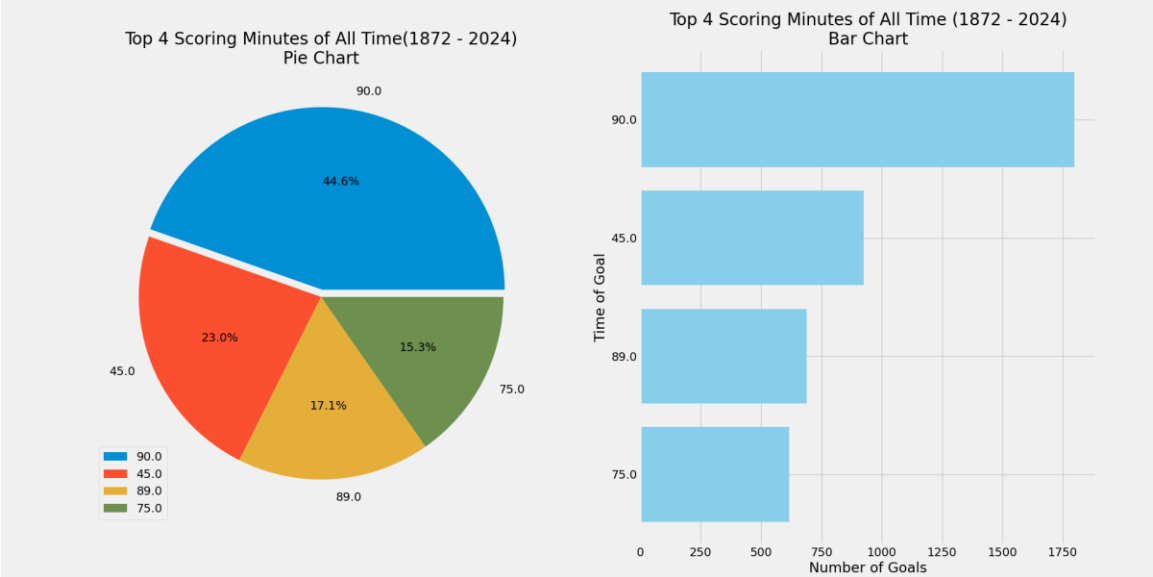
Output Description

1. Pie Chart:
- Displays the percentage distribution of goals scored during the top 4 scoring minutes.
- Highlights the most common scoring minute using an exploded segment.

2. Horizontal Bar Chart:
- Shows the absolute number of goals scored during the top 4 scoring minutes.

- Provides a clear comparison in terms of actual counts.

Visualization:



Use Case

This script is useful for analyzing football goal-scoring trends over time and visually comparing the most common scoring moments in matches.

Customization

1. Modify the most_common(4) method to analyze more or fewer scoring minutes.
2. Change the file path and column name (minute) to work with a different dataset or analysis metric.