Api link for NBA data: https://nba-apidocumentation.knowledgeowl.com/help

Installation Process as Follows

In [91]:
```
pip install nba-api
```

```
Requirement already satisfied: nba-api in c:\users\dsu\desktop\nba data\.venv\lib\si
te-packages (1.9.0)
Requirement already satisfied: numpy>=1.26.0 in c:\users\dsu\desktop\nba data\.venv
\lib\site-packages (from nba-api) (2.2.5)
Requirement already satisfied: pandas>=2.2.0 in c:\users\dsu\desktop\nba data\.venv
\lib\site-packages (from nba-api) (2.2.3)
Requirement already satisfied: requests<3.0.0,>=2.32.3 in c:\users\dsu\desktop\nba d
ata\.venv\lib\site-packages (from nba-api) (2.32.3)
Requirement already satisfied: charset-normalizer<4,>=2 in c:\users\dsu\desktop\nba
data\.venv\lib\site-packages (from requests<3.0.0,>=2.32.3->nba-api) (3.4.2)
Requirement already satisfied: idna<4,>=2.5 in c:\users\dsu\desktop\nba data\.venv\l
ib\site-packages (from requests<3.0.0,>=2.32.3->nba-api) (3.10)
Requirement already satisfied: urllib3<3,>=1.21.1 in c:\users\dsu\desktop\nba data\.
venv\lib\site-packages (from requests<3.0.0,>=2.32.3->nba-api) (2.4.0)
Requirement already satisfied: certifi>=2017.4.17 in c:\users\dsu\desktop\nba data\.
venv\lib\site-packages (from requests<3.0.0,>=2.32.3->nba-api) (2025.4.26)
Requirement already satisfied: python-dateutil>=2.8.2 in c:\users\dsu\desktop\nba da
ta\.venv\lib\site-packages (from pandas>=2.2.0->nba-api) (2.9.0.post0)
Requirement already satisfied: pytz>=2020.1 in c:\users\dsu\desktop\nba data\.venv\l
ib\site-packages (from pandas>=2.2.0->nba-api) (2025.2)
Requirement already satisfied: tzdata>=2022.7 in c:\users\dsu\desktop\nba data\.venv
\lib\site-packages (from pandas>=2.2.0->nba-api) (2025.2)
Requirement already satisfied: six>=1.5 in c:\users\dsu\desktop\nba data\.venv\lib\s
ite-packages (from python-dateutil>=2.8.2->pandas>=2.2.0->nba-api) (1.17.0)
Note: you may need to restart the kernel to use updated packages.
```

In [92]:
```
%pip install matplotlib
```

```
Requirement already satisfied: matplotlib in c:\users\dsu\desktop\nba data\.venv\lib
\site-packages (3.10.1)
Requirement already satisfied: contourpy>=1.0.1 in c:\users\dsu\desktop\nba data\.ve
nv\lib\site-packages (from matplotlib) (1.3.2)
Requirement already satisfied: cycler>=0.10 in c:\users\dsu\desktop\nba data\.venv\l
ib\site-packages (from matplotlib) (0.12.1)
Requirement already satisfied: fonttools>=4.22.0 in c:\users\dsu\desktop\nba data\.v
env\lib\site-packages (from matplotlib) (4.57.0)
Requirement already satisfied: kiwisolver>=1.3.1 in c:\users\dsu\desktop\nba data\.v
env\lib\site-packages (from matplotlib) (1.4.8)
Requirement already satisfied: numpy>=1.23 in c:\users\dsu\desktop\nba data\.venv\li
b\site-packages (from matplotlib) (2.2.5)
Requirement already satisfied: packaging>=20.0 in c:\users\dsu\desktop\nba data\.ven
v\lib\site-packages (from matplotlib) (25.0)
Requirement already satisfied: pillow>=8 in c:\users\dsu\desktop\nba data\.venv\lib
\site-packages (from matplotlib) (11.2.1)
Requirement already satisfied: pyparsing>=2.3.1 in c:\users\dsu\desktop\nba data\.ve
nv\lib\site-packages (from matplotlib) (3.2.3)
Requirement already satisfied: python-dateutil>=2.7 in c:\users\dsu\desktop\nba data
\.venv\lib\site-packages (from matplotlib) (2.9.0.post0)
Requirement already satisfied: six>=1.5 in c:\users\dsu\desktop\nba data\.venv\lib\s
ite-packages (from python-dateutil>=2.7->matplotlib) (1.17.0)
Note: you may need to restart the kernel to use updated packages.
```

In [93]:
```python
import matplotlib.pyplot as plt
import pandas as pd
import numpy as np
```

Here is the a sample on how to get the data for one player

Documentation Link: https://github.com/swar/nba_api/blob/master/docs/examples/Ba

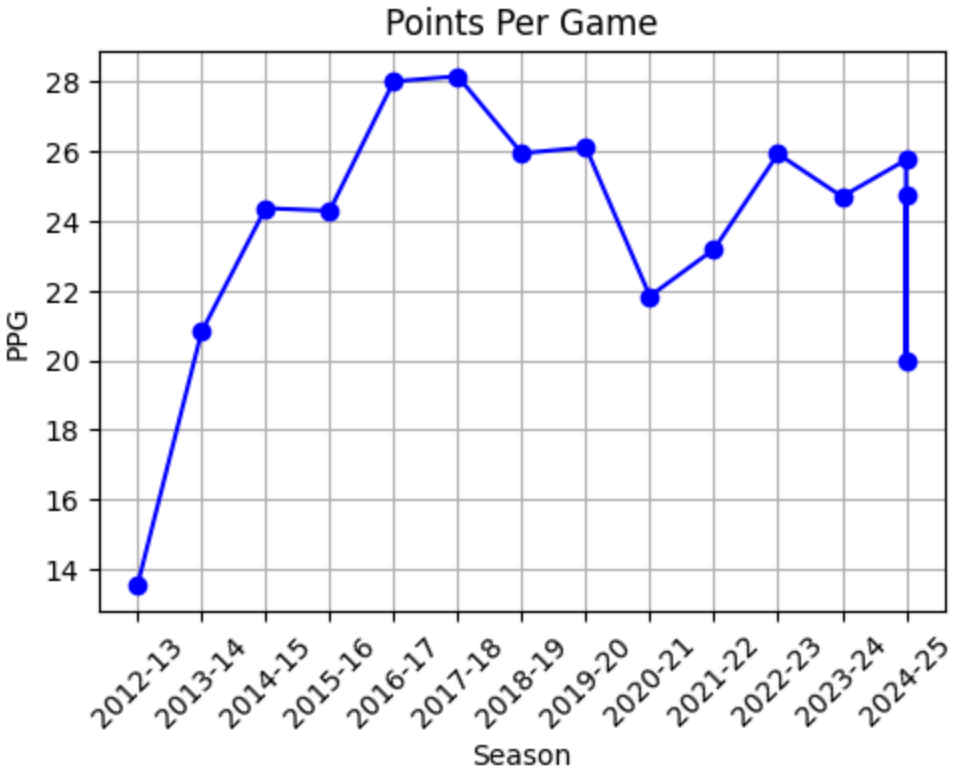
In [94]:
```python
from nba_api.stats.endpoints import playercareerstats

# Anthony Davis
career = playercareerstats.PlayerCareerStats(player_id="203076")
career.get_data_frames()[0].head()
```

Out[94]:

	PLAYER_ID	SEASON_ID	LEAGUE_ID	TEAM_ID	TEAM_ABBREVIATION	PLAYER_AGE	G
0	203076	2012-13	00	1610612740	NOH	20.0	6
1	203076	2013-14	00	1610612740	NOP	21.0	6
2	203076	2014-15	00	1610612740	NOP	22.0	6
3	203076	2015-16	00	1610612740	NOP	23.0	6
4	203076	2016-17	00	1610612740	NOP	24.0	7

To graph my data, I will be using matplotlib alongside pandas and numpy, tools I learnt in my CIS 372 course. Here is a sample graph to show its output.

In [95]:
```
df = career.get_data_frames()[0] # putting the data we retrieve into a dataframe la
season_labels = df['SEASON_ID'].values # this puts all the seasons into this list f

plt.figure(figsize=(12, 8)) # the framework for the graph

# plot points per game
plt.subplot(2, 2, 1) # this indicates that it is a 2 x 2 grid and we are starting a
plt.plot(season_labels, df['PTS']/df['GP'], 'b-o') # the parameters are x (seasons)
plt.title('Points Per Game')
plt.xlabel('Season')
plt.ylabel('PPG')
plt.xticks(rotation=45)
plt.grid(True)
```



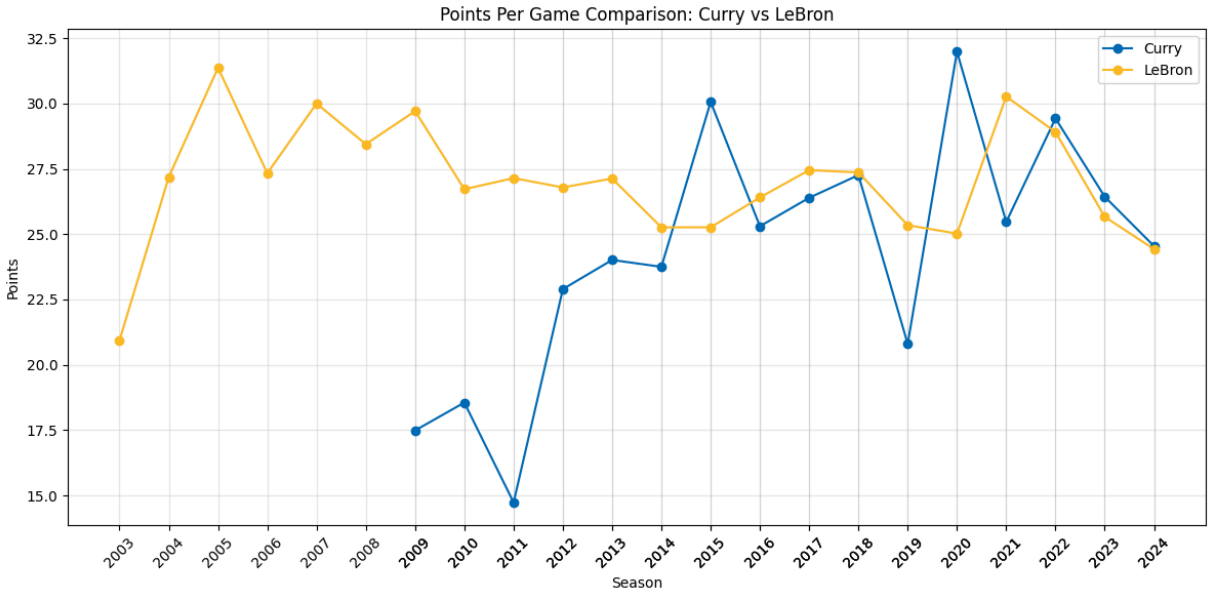
I would like to look at difference in season statistics throughout the careers of Stephen Curry and Lebron James. They are considered the two greatest players of this generation, so I will look at how much they vary individually.

The approach we will be taking is to look at the trends overtime for average points, assists, offensive rebounds, defensive rebounds, steals, blocks, turnovers, players fouled, field goal percentage, 3 point field goal percentage, and free

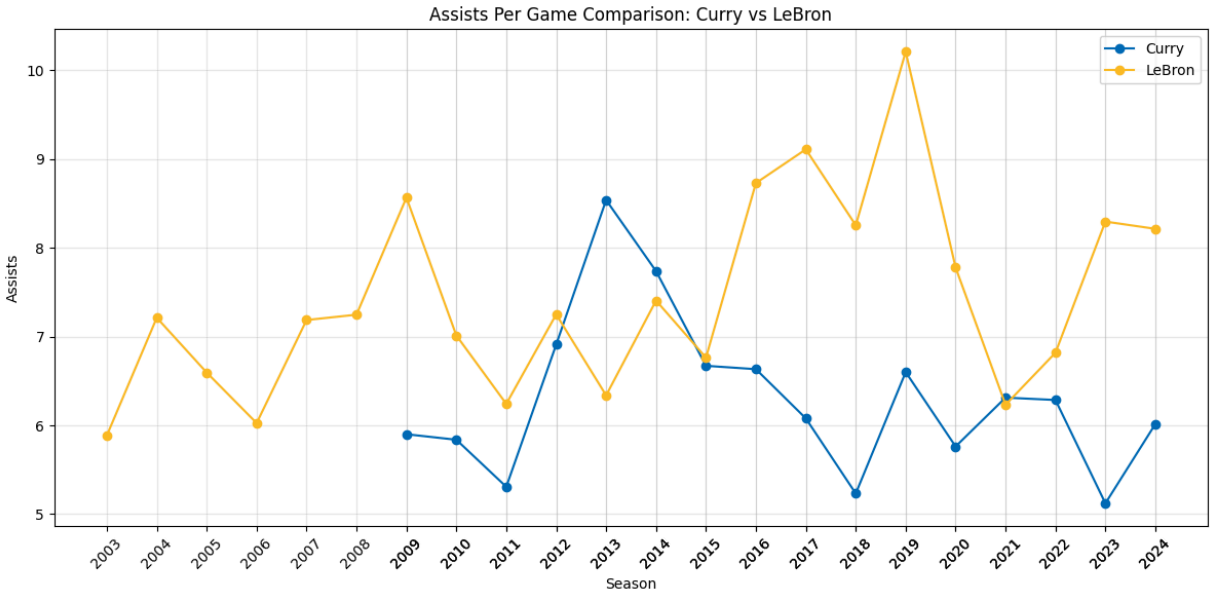
throw percentage. This will be visualized with one overarching bar graph, and indiviudal line graphs.

In [ ]:
```
from nba_api.stats.endpoints import playercareerstats
import pandas as pd
import matplotlib.pyplot as plt

# Get player data
curry = playercareerstats.PlayerCareerStats(player_id='201939').get_data_frames()[0
lebron = playercareerstats.PlayerCareerStats(player_id='2544').get_data_frames()[0]

def process_data(df): # calculates the average for each stat
    df['YEAR'] = df['SEASON_ID'].str[:4].astype(int)  # converting the string to an
    df['PPG'] = df['PTS'] / df['GP']
    df['APG'] = df['AST'] / df['GP']
    df['ORPG'] = df['OREB'] / df['GP']
    df['DRPG'] = df['DREB'] / df['GP']
    df['SPG'] = df['STL'] / df['GP']
    df['BPG'] = df['BLK'] / df['GP']
    df['TOPG'] = df['TOV'] / df['GP']
    df['PFPG'] = df['PF'] / df['GP']
    df['FG%'] = df['FG_PCT'] * 100
    df['3P%'] = df['FG3_PCT'] * 100
    df['FT%'] = df['FT_PCT'] * 100
    return df

curry_avg = process_data(curry)
lebron_avg = process_data(lebron)
curry_avg = curry_avg[curry_avg['YEAR'] <= 2024] # error check to make sure we dont
lebron_avg = lebron_avg[lebron_avg['YEAR'] <= 2024]
```

In [ ]:
```
plt.figure(figsize=(12, 6)) # creating the size of the diagram
plt.plot(curry_avg['YEAR'], curry_avg['PPG'], marker='o', label='Curry', color='#00
plt.plot(lebron_avg['YEAR'], lebron_avg['PPG'], marker='o', label='LeBron', color='
plt.title('Points Per Game Comparison: Curry vs LeBron')
plt.xlabel('Season')
plt.ylabel('Points')
plt.xticks(curry_avg['YEAR'].tolist() + lebron_avg['YEAR'].tolist(), rotation=45)
plt.grid(True, alpha=0.3) # adding the grid design
plt.legend()
plt.tight_layout()
plt.show()
```

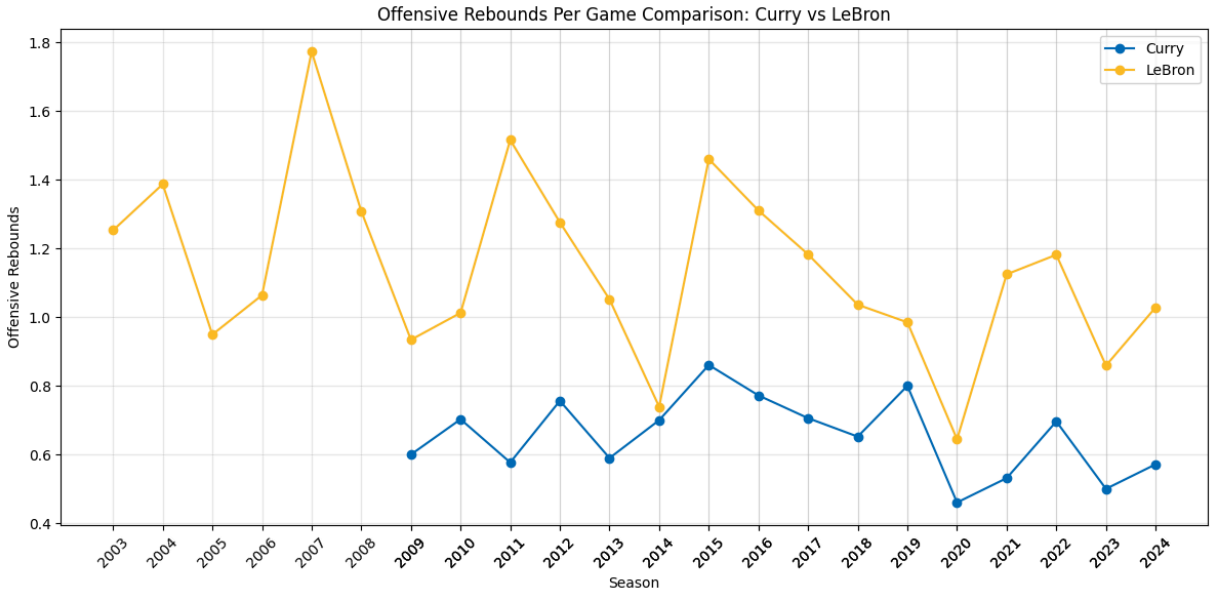


In [98]:
```
plt.figure(figsize=(12, 6))
plt.plot(curry_avg['YEAR'], curry_avg['APG'], marker='o', label='Curry', color='#00
plt.plot(lebron_avg['YEAR'], lebron_avg['APG'], marker='o', label='LeBron', color='
plt.title('Assists Per Game Comparison: Curry vs LeBron')
plt.xlabel('Season')
plt.ylabel('Assists')
plt.xticks(curry_avg['YEAR'].tolist() + lebron_avg['YEAR'].tolist(), rotation=45)
plt.grid(True, alpha=0.3)
plt.legend()
plt.tight_layout()
plt.show()
```



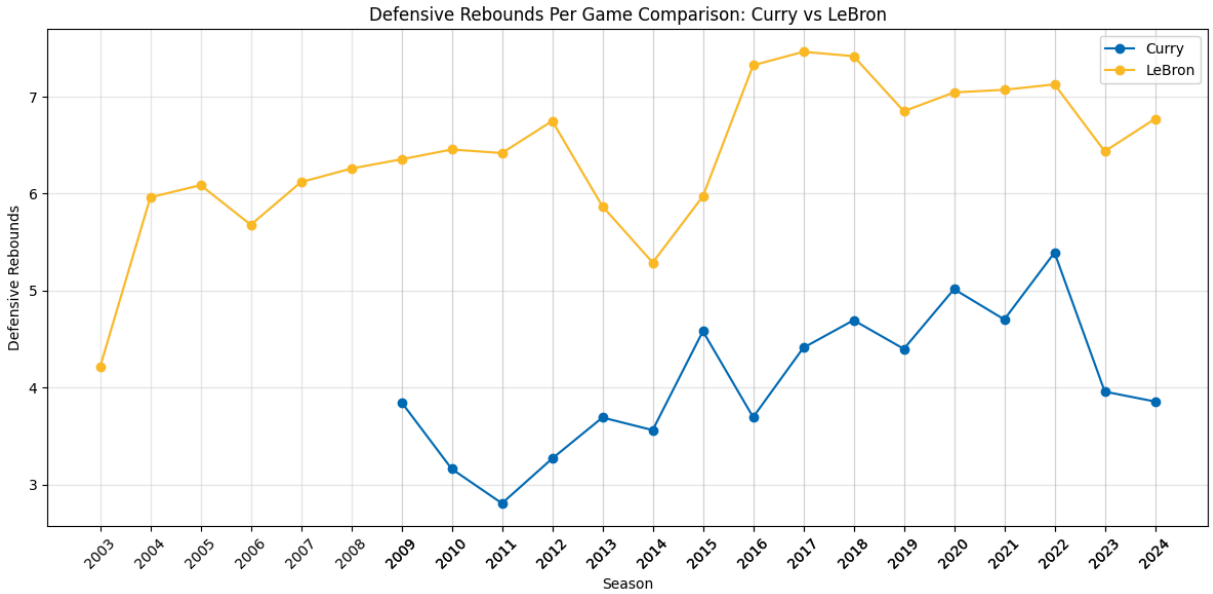
In [99]:
```
plt.figure(figsize=(12, 6))
plt.plot(curry_avg['YEAR'], curry_avg['ORPG'], marker='o', label='Curry', color='#0
plt.plot(lebron_avg['YEAR'], lebron_avg['ORPG'], marker='o', label='LeBron', color=
plt.title('Offensive Rebounds Per Game Comparison: Curry vs LeBron')
plt.xlabel('Season')
plt.ylabel('Offensive Rebounds')
```

```
plt.xticks(curry_avg['YEAR'].tolist() + lebron_avg['YEAR'].tolist(), rotation=45)
plt.grid(True, alpha=0.3)
plt.legend()
plt.tight_layout()
plt.show()
```



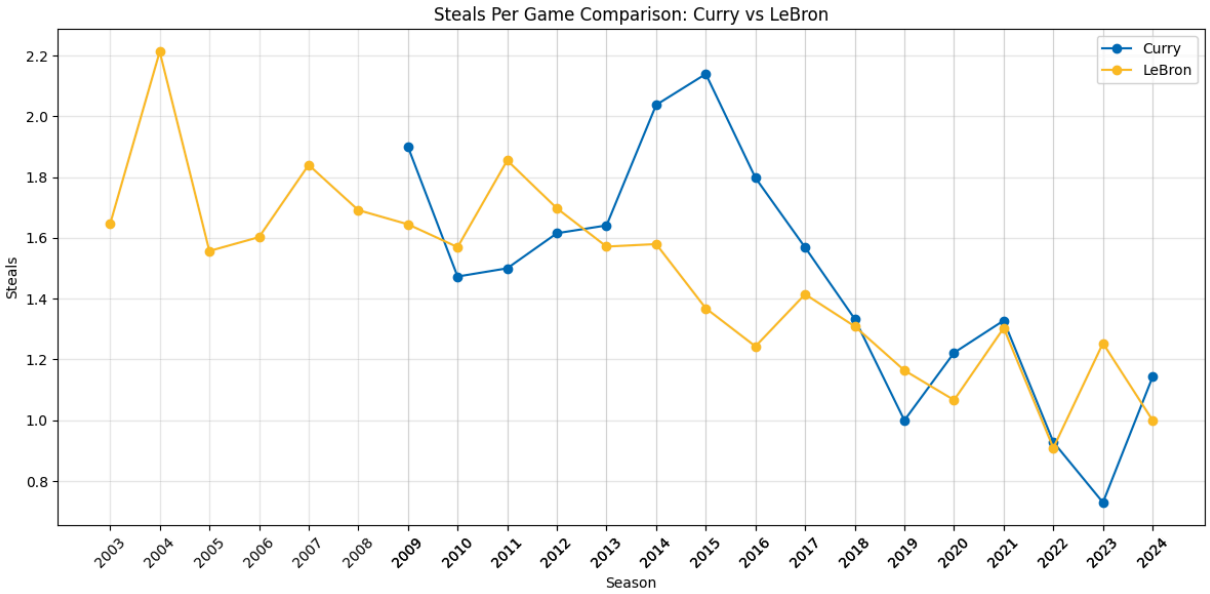
In [100…

```
plt.figure(figsize=(12, 6))
plt.plot(curry_avg['YEAR'], curry_avg['DRPG'], marker='o', label='Curry', color='#0
plt.plot(lebron_avg['YEAR'], lebron_avg['DRPG'], marker='o', label='LeBron', color=
plt.title('Defensive Rebounds Per Game Comparison: Curry vs LeBron')
plt.xlabel('Season')
plt.ylabel('Defensive Rebounds')
plt.xticks(curry_avg['YEAR'].tolist() + lebron_avg['YEAR'].tolist(), rotation=45)
plt.grid(True, alpha=0.3)
plt.legend()
plt.tight_layout()
plt.show()
```



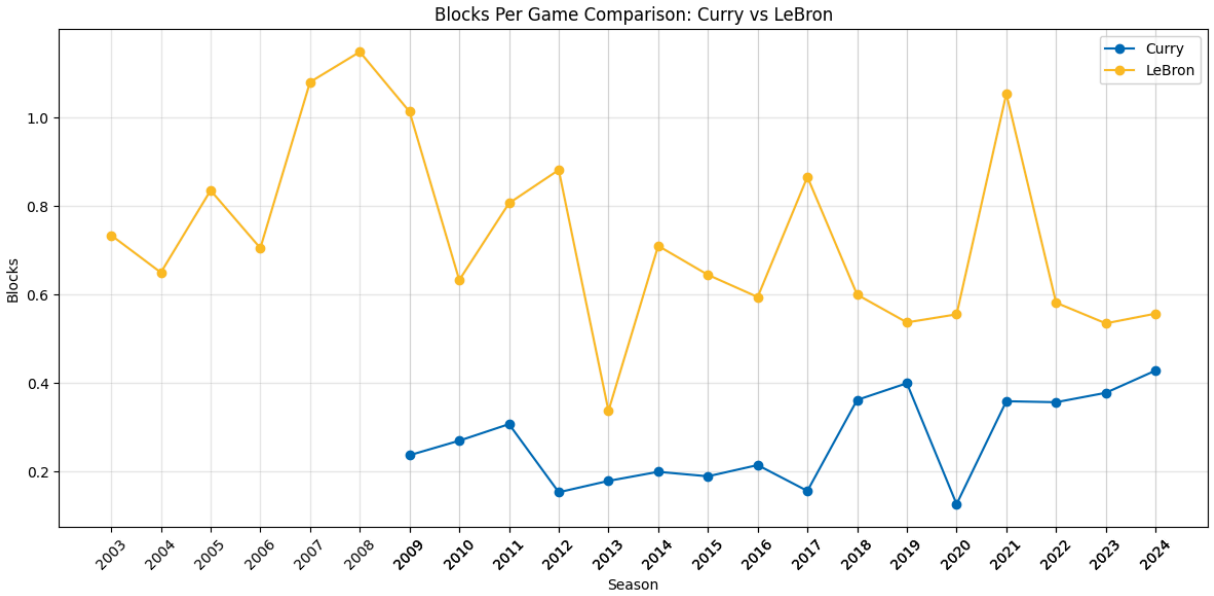
In [101…

```
plt.figure(figsize=(12, 6))
plt.plot(curry_avg['YEAR'], curry_avg['SPG'], marker='o', label='Curry', color='#00
plt.plot(lebron_avg['YEAR'], lebron_avg['SPG'], marker='o', label='LeBron', color='
plt.title('Steals Per Game Comparison: Curry vs LeBron')
plt.xlabel('Season')
plt.ylabel('Steals')
plt.xticks(curry_avg['YEAR'].tolist() + lebron_avg['YEAR'].tolist(), rotation=45)
plt.grid(True, alpha=0.3)
plt.legend()
plt.tight_layout()
plt.show()
```



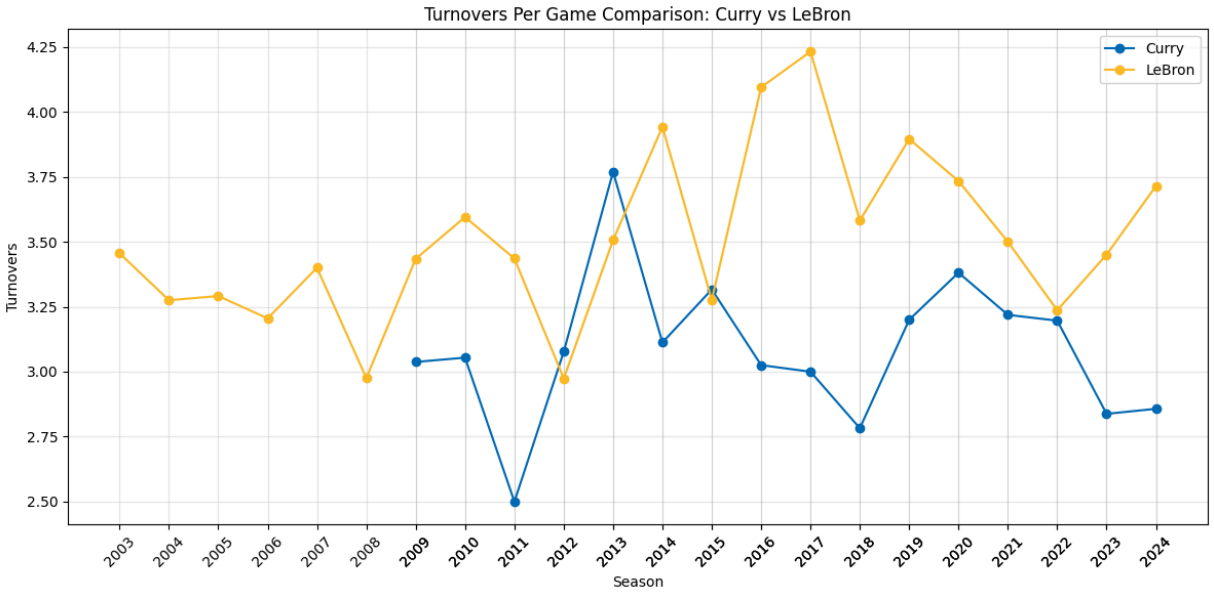
In [102…

```
plt.figure(figsize=(12, 6))
plt.plot(curry_avg['YEAR'], curry_avg['BPG'], marker='o', label='Curry', color='#00
plt.plot(lebron_avg['YEAR'], lebron_avg['BPG'], marker='o', label='LeBron', color='
plt.title('Blocks Per Game Comparison: Curry vs LeBron')
plt.xlabel('Season')
plt.ylabel('Blocks')
plt.xticks(curry_avg['YEAR'].tolist() + lebron_avg['YEAR'].tolist(), rotation=45)
plt.grid(True, alpha=0.3)
plt.legend()
plt.tight_layout()
plt.show()
```



In [103…

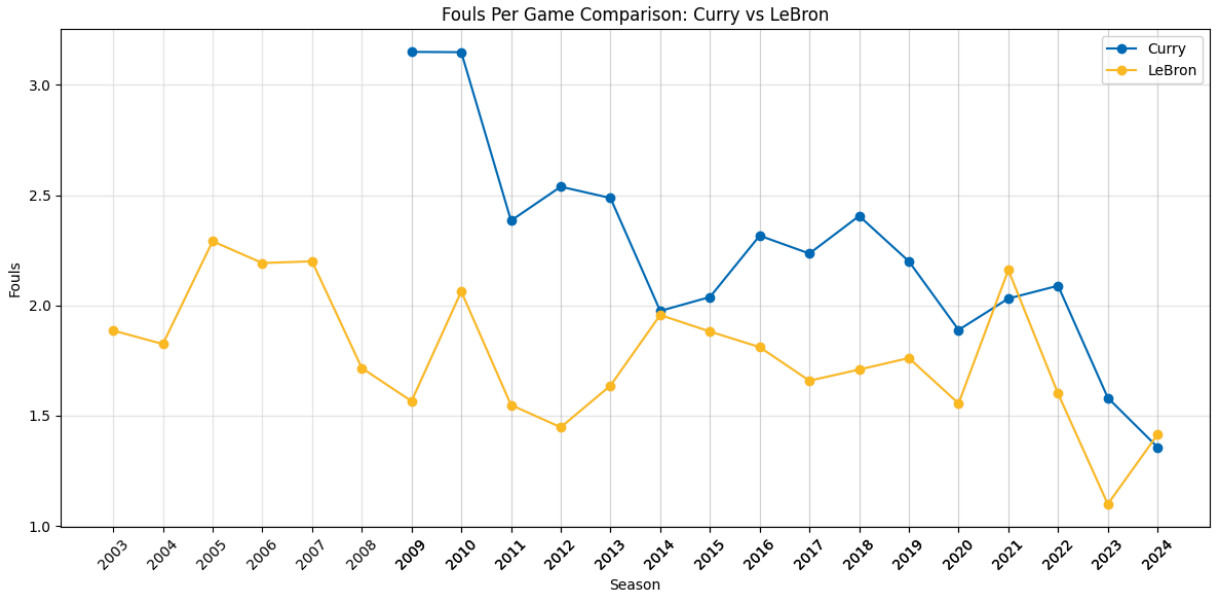
```
plt.figure(figsize=(12, 6))
plt.plot(curry_avg['YEAR'], curry_avg['TOPG'], marker='o', label='Curry', color='#0
plt.plot(lebron_avg['YEAR'], lebron_avg['TOPG'], marker='o', label='LeBron', color=
plt.title('Turnovers Per Game Comparison: Curry vs LeBron')
plt.xlabel('Season')
plt.ylabel('Turnovers')
plt.xticks(curry_avg['YEAR'].tolist() + lebron_avg['YEAR'].tolist(), rotation=45)
plt.grid(True, alpha=0.3)
plt.legend()
plt.tight_layout()
plt.show()
```



In [104…

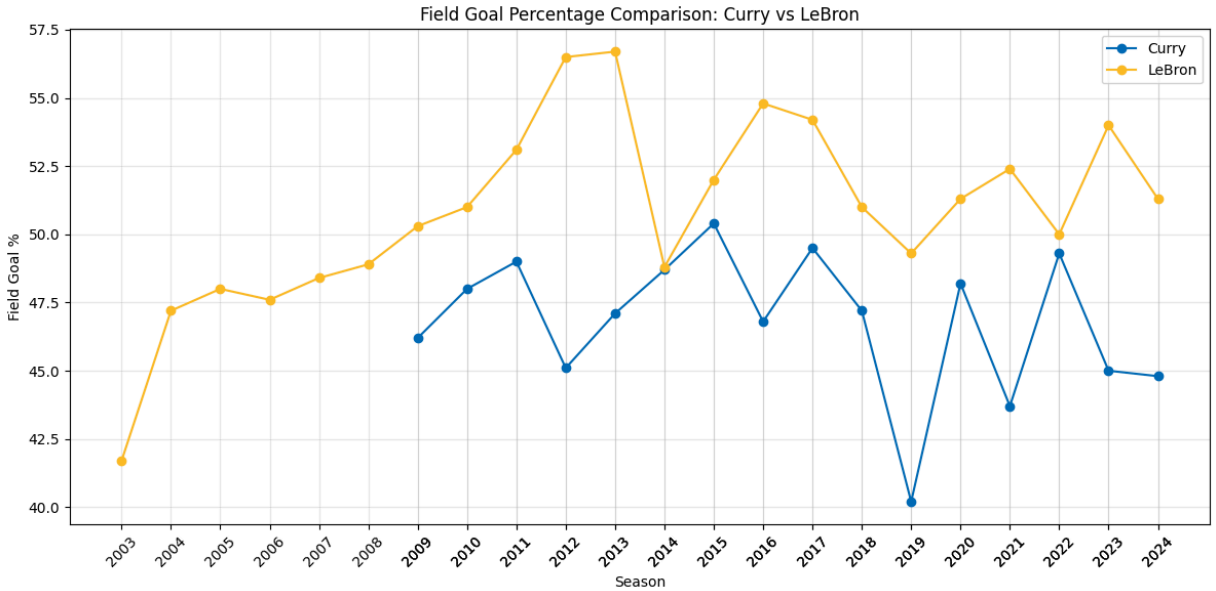
```
plt.figure(figsize=(12, 6))
plt.plot(curry_avg['YEAR'], curry_avg['PFPG'], marker='o', label='Curry', color='#0
plt.plot(lebron_avg['YEAR'], lebron_avg['PFPG'], marker='o', label='LeBron', color=
plt.title('Fouls Per Game Comparison: Curry vs LeBron')
plt.xlabel('Season')
plt.ylabel('Fouls')
```

```
plt.xticks(curry_avg['YEAR'].tolist() + lebron_avg['YEAR'].tolist(), rotation=45)
plt.grid(True, alpha=0.3)
plt.legend()
plt.tight_layout()
plt.show()
```



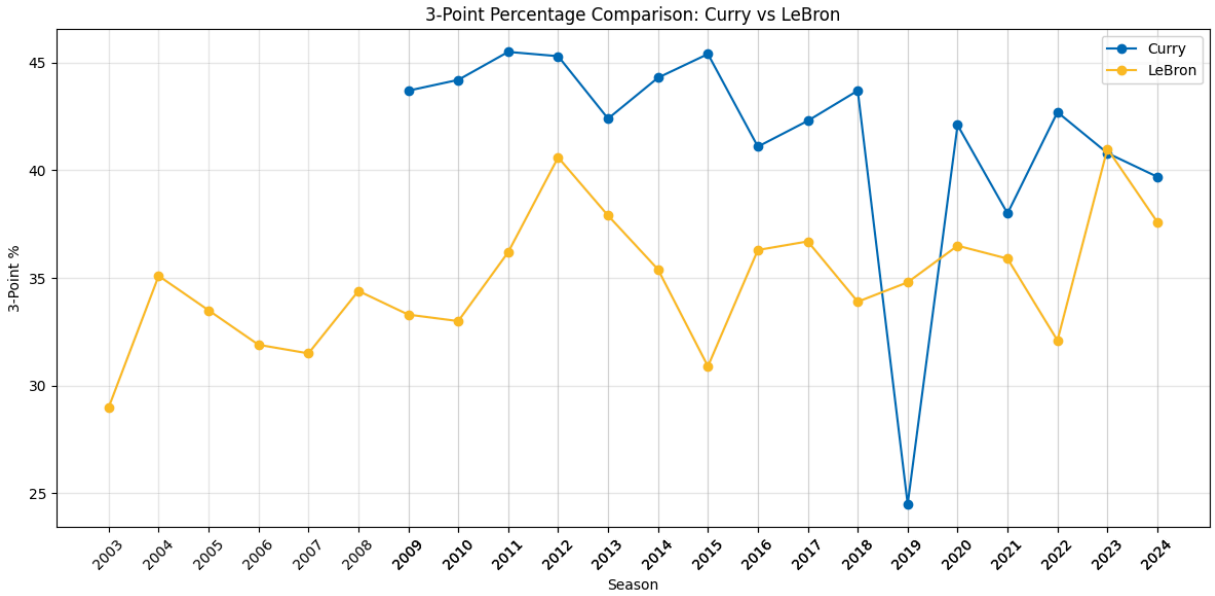
In [105…

```
plt.figure(figsize=(12, 6))
plt.plot(curry_avg['YEAR'], curry_avg['FG%'], marker='o', label='Curry', color='#00
plt.plot(lebron_avg['YEAR'], lebron_avg['FG%'], marker='o', label='LeBron', color='
plt.title('Field Goal Percentage Comparison: Curry vs LeBron')
plt.xlabel('Season')
plt.ylabel('Field Goal %')
plt.xticks(curry_avg['YEAR'].tolist() + lebron_avg['YEAR'].tolist(), rotation=45)
plt.grid(True, alpha=0.3)
plt.legend()
plt.tight_layout()
plt.show()
```



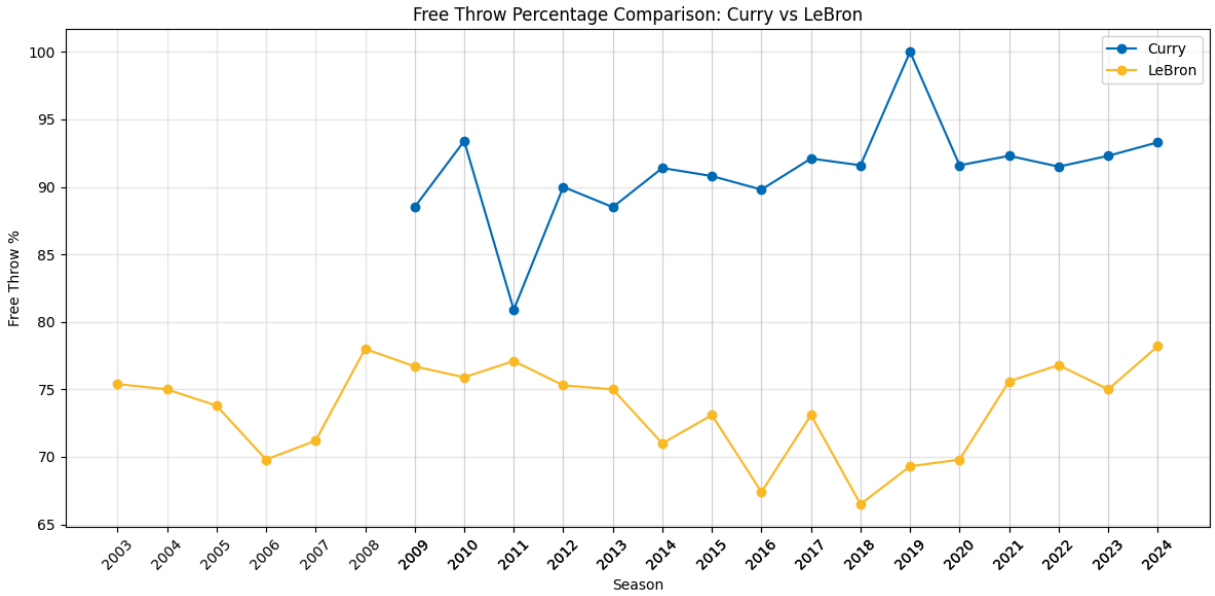
In [106…

```
plt.figure(figsize=(12, 6))
plt.plot(curry_avg['YEAR'], curry_avg['3P%'], marker='o', label='Curry', color='#00
plt.plot(lebron_avg['YEAR'], lebron_avg['3P%'], marker='o', label='LeBron', color='
plt.title('3-Point Percentage Comparison: Curry vs LeBron')
plt.xlabel('Season')
plt.ylabel('3-Point %')
plt.xticks(curry_avg['YEAR'].tolist() + lebron_avg['YEAR'].tolist(), rotation=45)
plt.grid(True, alpha=0.3)
plt.legend()
plt.tight_layout()
plt.show()
```

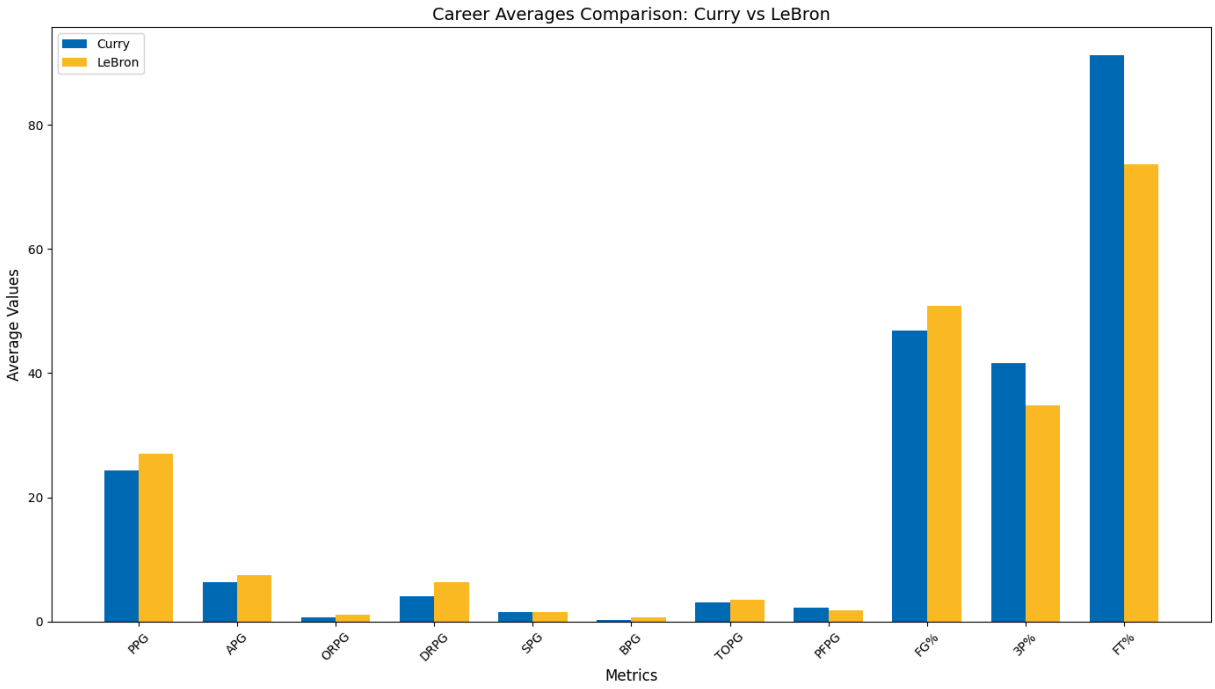


In [107…

```
plt.figure(figsize=(12, 6))
plt.plot(curry_avg['YEAR'], curry_avg['FT%'], marker='o', label='Curry', color='#00
plt.plot(lebron_avg['YEAR'], lebron_avg['FT%'], marker='o', label='LeBron', color='
plt.title('Free Throw Percentage Comparison: Curry vs LeBron')
plt.xlabel('Season')
plt.ylabel('Free Throw %')
plt.xticks(curry_avg['YEAR'].tolist() + lebron_avg['YEAR'].tolist(), rotation=45)
plt.grid(True, alpha=0.3)
plt.legend()
plt.tight_layout()
plt.show()
```



In [ ]:
```
metrics = ['PPG', 'APG', 'ORPG', 'DRPG', 'SPG', 'BPG', 'TOPG', 'PFPG', 'FG%', '3P%'

curry_averages = [curry_avg[metric].mean() for metric in metrics]
lebron_averages = [lebron_avg[metric].mean() for metric in metrics]

x = np.arange(len(metrics))
width = 0.35

plt.figure(figsize=(14, 8))
plt.bar(x - width/2, curry_averages, width, label='Curry', color='#006BB6')
plt.bar(x + width/2, lebron_averages, width, label='LeBron', color='#FDB927')

plt.xlabel('Metrics', fontsize=12)
plt.ylabel('Average Values', fontsize=12)
plt.title('Career Averages Comparison: Curry vs LeBron', fontsize=14)
plt.xticks(x, metrics, rotation=45)
plt.legend()

plt.tight_layout()
plt.show()
```



Looking at the data presented to us, we can see that Lebron was more dominant in the more physical aspects of the game, whereas Curry was more dominant at shooting. Lebron has higher average offensive and defensive rebounds and Curry has a better 3 point field goal percentage and free throw percentage. In a level playing field, they on a more equal scale. The points per game and assist per game are similar, indicated close skill levels between the two.