tutorial

May 12, 2023

1 What Stat really makes an NBA Championship Team

Group Members: Fiyin Oluseye, Amara Ihediohanma, and Roger Thibaudeau

1.1 Part 1: Data Scraping and Preparation

[1]:
```
import pandas as pd
df = pd.read_csv("2020_Playoff_Stats.csv")
df.head(17)
```

[1]:
```
      Rk                    Team   G    MP   FG   FGA    FG%   3P  3PA    3P%  \
0    1.0          Boston Celtics  24  5760  881  1962  0.449  328  879  0.373
1    2.0   Golden State Warriors  22  5280  910  1895  0.480  308  821  0.375
2    3.0        Dallas Mavericks  18  4320  653  1455  0.449  284  747  0.380
3    4.0              Miami Heat  18  4320  684  1536  0.445  196  626  0.313
4    5.0            Phoenix Suns  13  3120  535  1076  0.497  128  353  0.363
5    6.0       Memphis Grizzlies  12  2880  477  1096  0.435  157  430  0.365
6    7.0      Philadelphia 76ers  12  2905  437   939  0.465  149  400  0.373
7    8.0         Milwaukee Bucks  12  2880  462  1056  0.438  127  388  0.327
8    9.0    New Orleans Pelicans   6  1440  234   506  0.462   56  158  0.354
9   10.0  Minnesota Timberwolves   6  1440  218   492  0.443   83  214  0.388
10  11.0         Toronto Raptors   6  1465  230   516  0.446   59  197  0.299
11  12.0               Utah Jazz   6  1440  210   474  0.443   49  179  0.274
12  13.0          Denver Nuggets   5  1200  197   414  0.476   56  157  0.357
13  14.0           Atlanta Hawks   5  1200  172   391  0.440   57  175  0.326
14  15.0           Chicago Bulls   5  1200  182   451  0.404   52  184  0.283
15  16.0           Brooklyn Nets   4   960  157   312  0.503   46  109  0.422
16   NaN          League Average  11  2613  415   911  0.456  133  376  0.355

    …    FT%  ORB  DRB   TRB  AST  STL  BLK  TOV   PF   PTS
0   …  0.797  216  814  1030  588  154  150  353  497  2533
1   …  0.766  216  750   966  594  170  109  320  474  2461
2   …  0.771  117  540   657  345  129   50  184  380  1914
3   …  0.804  177  562   739  394  150   66  233  386  1876
4   …  0.817  123  399   522  334   86   49  173  292  1399
5   …  0.735  149  401   550  302  110   73  168  249  1350
6   …  0.849   92  376   468  261   73   52  177  247  1254
7   …  0.731  117  488   605  250   76   54  165  230  1233
```

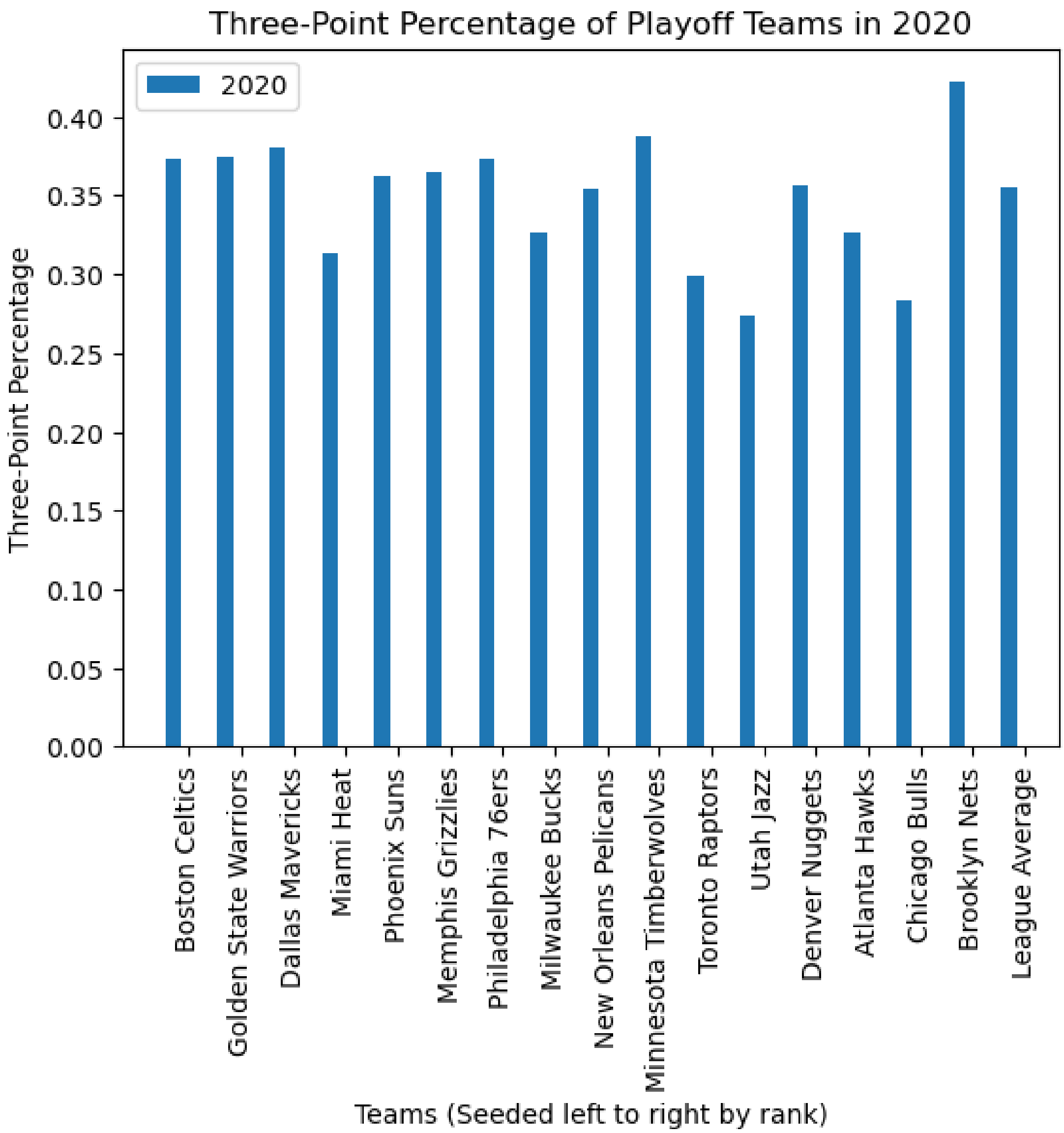
```
8   …  0.780   91  183   274  128   38   21   87  127   659
9   …  0.810   42  198   240  137   49   47  106  161   655
10  …  0.794   60  165   225  124   40   29   60  133   619
11  …  0.786   56  213   269  103   24   22   71  132   594
12  …  0.794   56  153   209  125   35   17   82  125   550
13  …  0.782   43  154   197   93   29   12   82  108   487
14  …  0.833   41  179   220  115   39   16   65   93   476
15  …  0.738   34  102   136   89   32   26   61   99   436
16  …  0.785  102  355   457  249   77   50  149  233  1156

[17 rows x 25 columns]
```

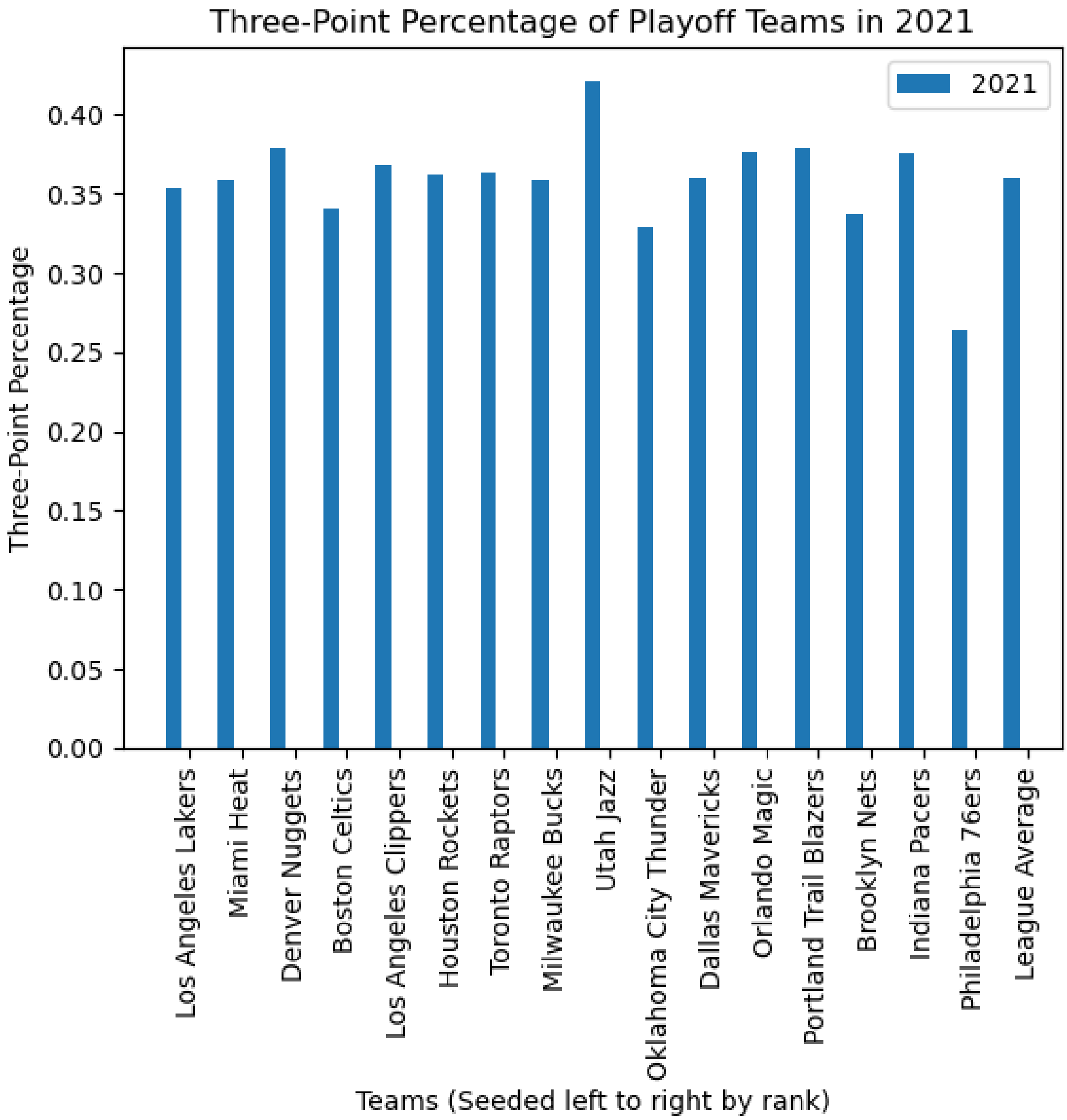
[2]:
```python
import matplotlib.pyplot as plt

#read data from csv file from 2020
data_2020 = pd.read_csv("2020_Playoff_Stats.csv")
data_2021 = pd.read_csv("2021_Playoff_Stats.csv")
data_2022 = pd.read_csv("2022_Playoff_Stats.csv")

#setting up data
playoff_list = [data_2020, data_2021, data_2022]
```

1.1.1 Statistic: Three Point Percentage (3P%)

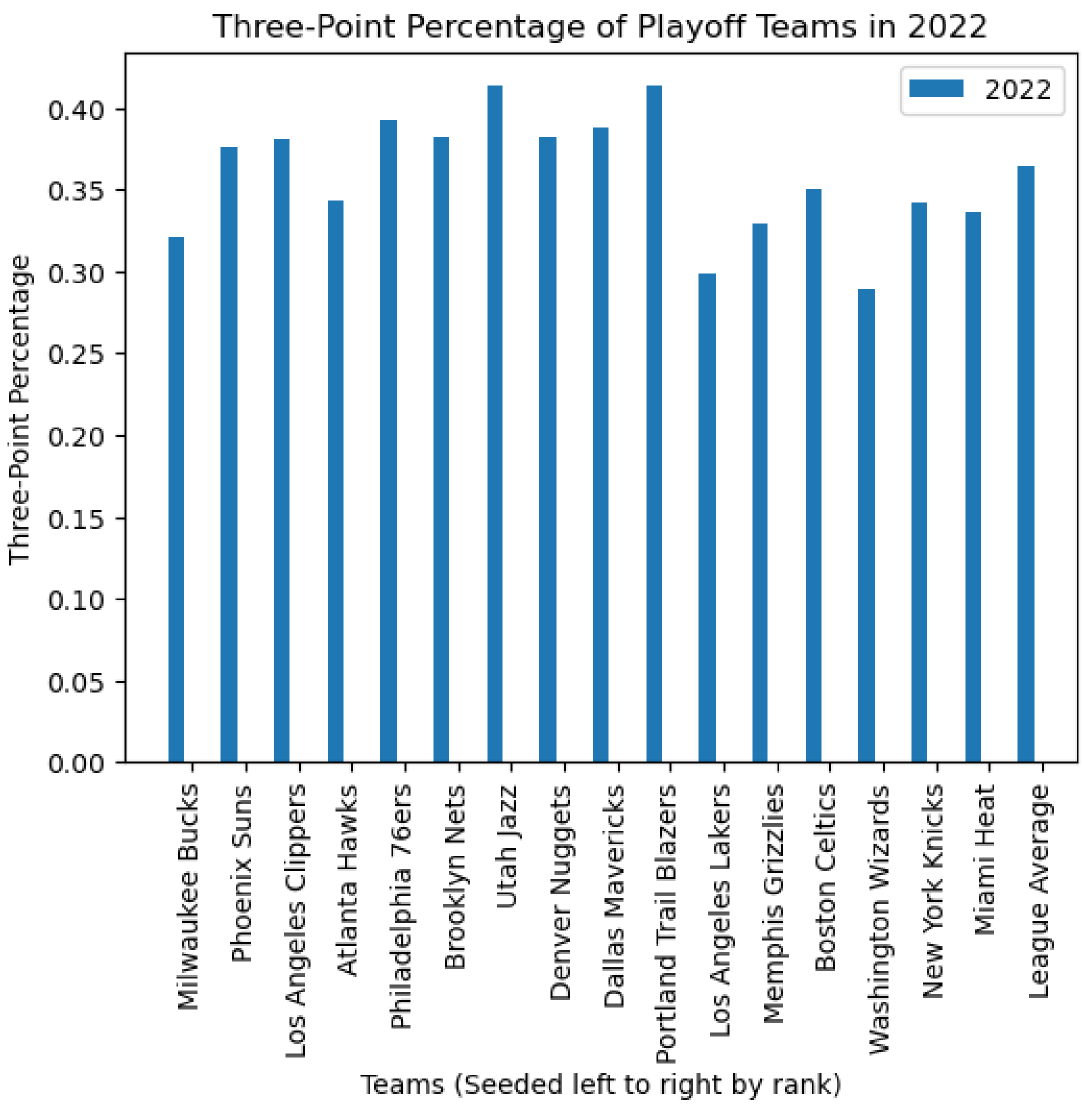
[3]:
```python
#Since the playoff teams are different per year, we cannot add them on the same␣
↪plot, so there will be multiple plots per year.
year = 2020

for i in playoff_list:
    #gathering all relevant data columns for the playoff year
    teams = i["Team"]
    three_point_pct = i["3P%"]

    # Set the width of the bars
    bar_width = 0.3
    # Set the positions of the bars on the x-axis
    bar_positions = range(len(teams))

    #3 pt percentage plots ////////////////////////////////////////////////////////
↪///////////////////
    # Plot the bars
    plt.bar(bar_positions, three_point_pct, width=bar_width, label=str(year))

    # Add labels, title, and legend
    plt.xlabel("Teams (Seeded left to right by rank)")
    plt.ylabel("Three-Point Percentage")
```

```python
plt.title("Three-Point Percentage of Playoff Teams in " + str(year))
plt.xticks([r + bar_width for r in range(len(teams))], teams, rotation=90)
plt.legend()

# Display the graph
plt.show()

year += 1
```





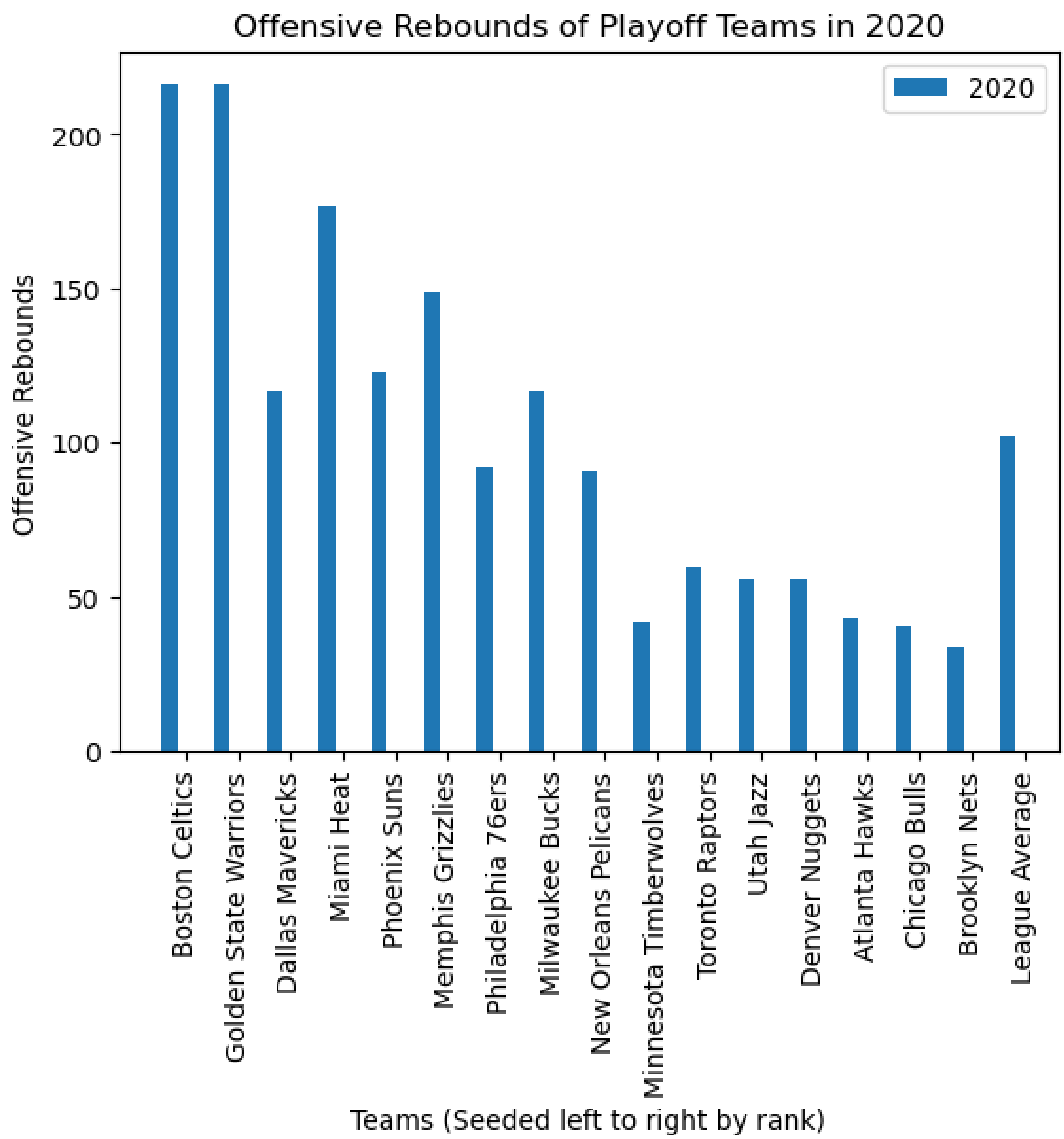
Three-Point Percentage of Playoff Teams in 2021
2021
Three-Point Percentage
0.00
0.05
0.10
0.15
0.20
0.25
0.30
0.35
0.40
Los Angeles Lakers
Miami Heat
Denver Nuggets
Boston Celtics
Los Angeles Clippers
Houston Rockets
Toronto Raptors
Milwaukee Bucks
Utah Jazz
Oklahoma City Thunder
Dallas Mavericks
Orlando Magic
Portland Trail Blazers
Brooklyn Nets
Indiana Pacers
Philadelphia 76ers
League Average
Teams (Seeded left to right by rank)

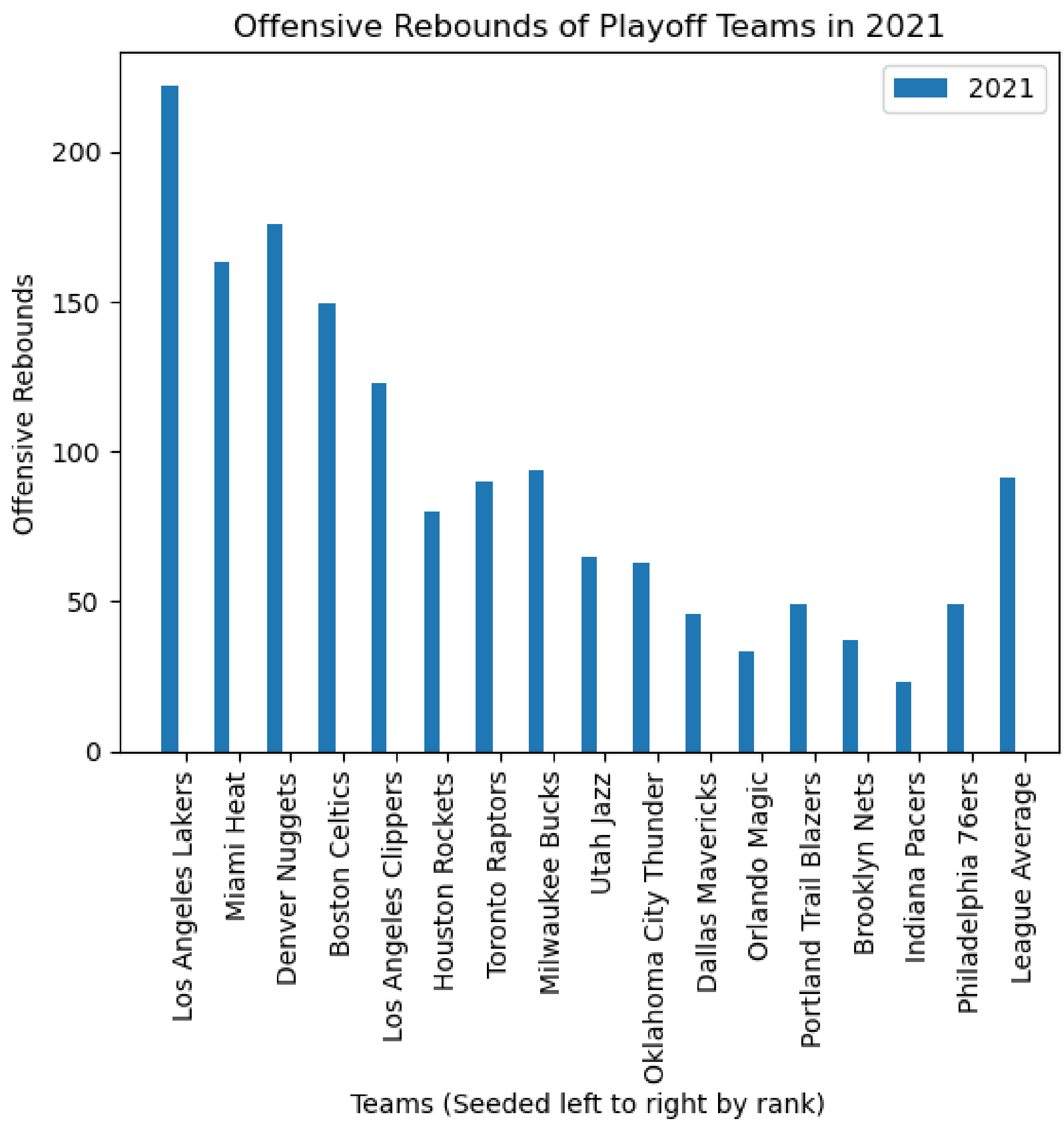


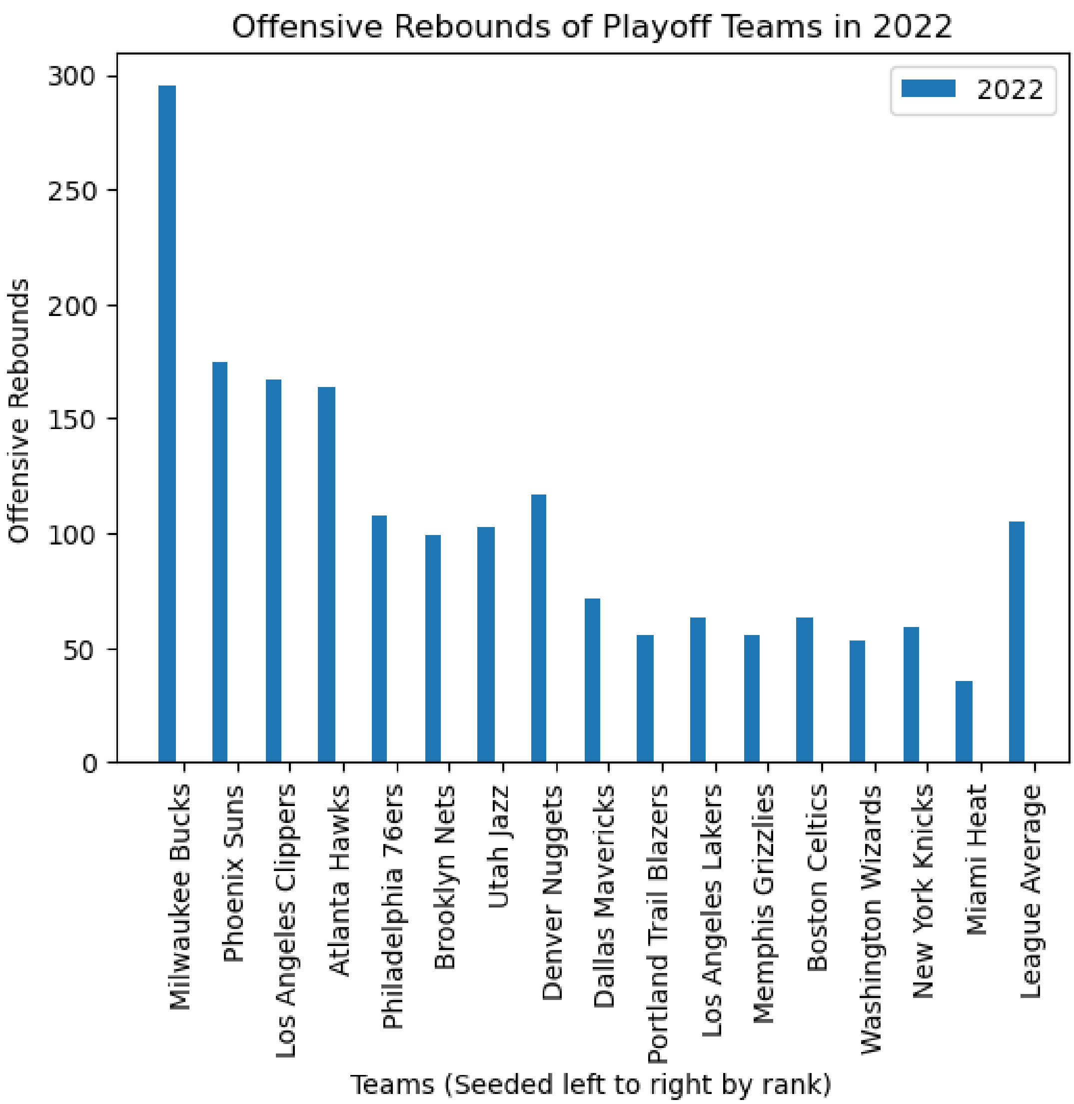
1.1.2 Statistic: Offensive Rebounds (ORB)

[4]:
```
#Since the playoff teams are different per year, we cannot add them on the same␣
↪plot, so there will be multiple plots per year.
year = 2020

for i in playoff_list:
    #gathering all relevant data columns for the playoff year
    teams = i["Team"]
    offensive_rebounds = i["ORB"]

    # Set the width of the bars
```

```
    bar_width = 0.3
    # Set the positions of the bars on the x-axis
    bar_positions = range(len(teams))


    #ORB plots /////////////////////////////////////////////////////////////////
↪///////
    # Plot the bars
    plt.bar(bar_positions, offensive_rebounds, width=bar_width, label=str(year))

    # Add labels, title, and legend
    plt.xlabel("Teams (Seeded left to right by rank)")
    plt.ylabel("Offensive Rebounds")
    plt.title("Offensive Rebounds of Playoff Teams in " + str(year))
    plt.xticks([r + bar_width for r in range(len(teams))], teams, rotation=90)
    plt.legend()

    # Display the graph
    plt.show()

    year += 1
```


Offensive Rebounds of Playoff Teams in 2020
2020
Offensive Rebounds
0
50
100
150
200
Boston Celtics
Golden State Warriors
Dallas Mavericks
Miami Heat
Phoenix Suns
Memphis Grizzlies
Philadelphia 76ers
Milwaukee Bucks
New Orleans Pelicans
Minnesota Timberwolves
Toronto Raptors
Utah Jazz
Denver Nuggets
Atlanta Hawks
Chicago Bulls
Brooklyn Nets
League Average
Teams (Seeded left to right by rank)

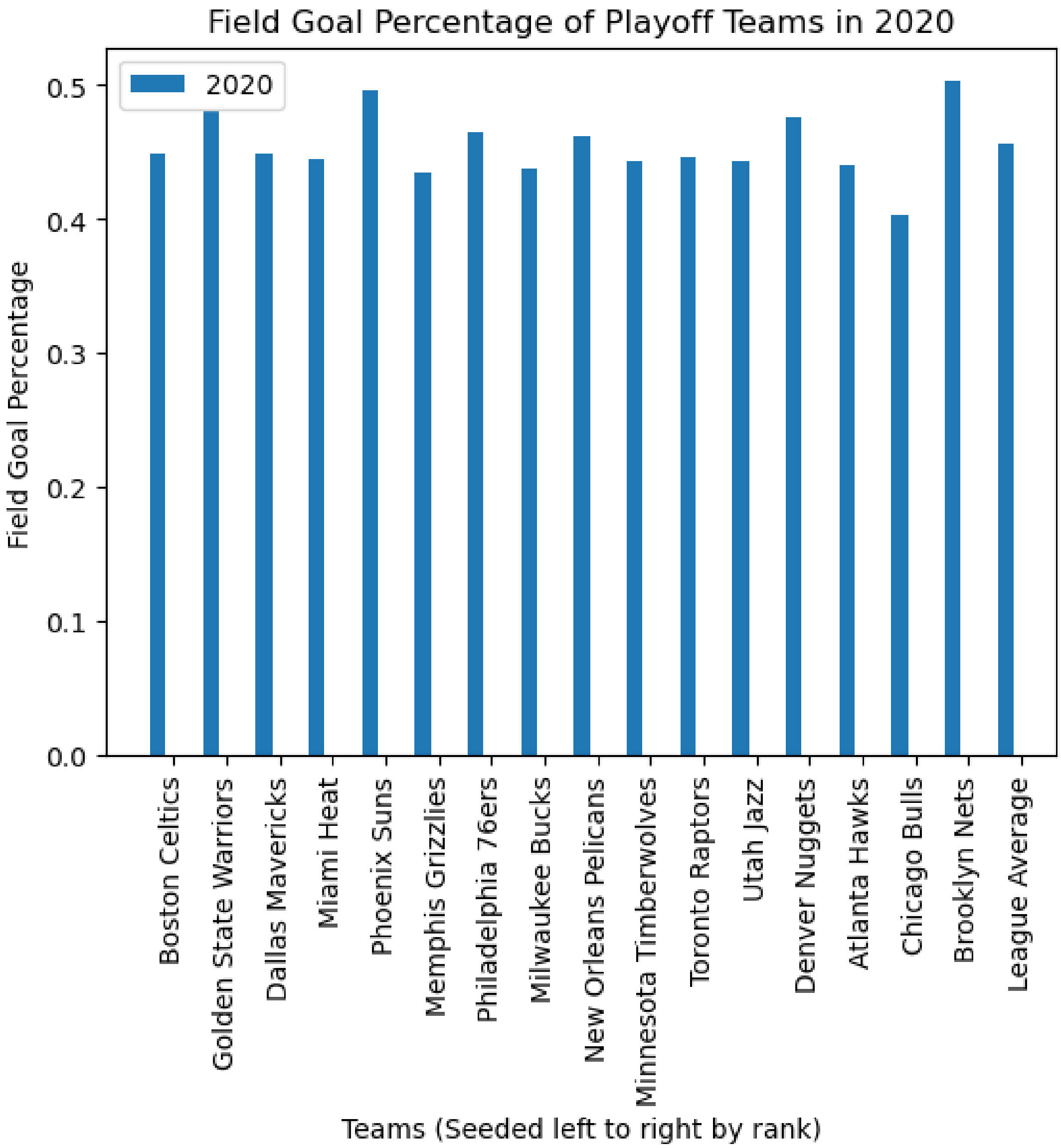

Offensive Rebounds of Playoff Teams in 2021
2021
Offensive Rebounds
0
50
100
150
200
Los Angeles Lakers
Miami Heat
Denver Nuggets
Boston Celtics
Los Angeles Clippers
Houston Rockets
Toronto Raptors
Milwaukee Bucks
Utah Jazz
Oklahoma City Thunder
Dallas Mavericks
Orlando Magic
Portland Trail Blazers
Brooklyn Nets
Indiana Pacers
Philadelphia 76ers
League Average
Teams (Seeded left to right by rank)



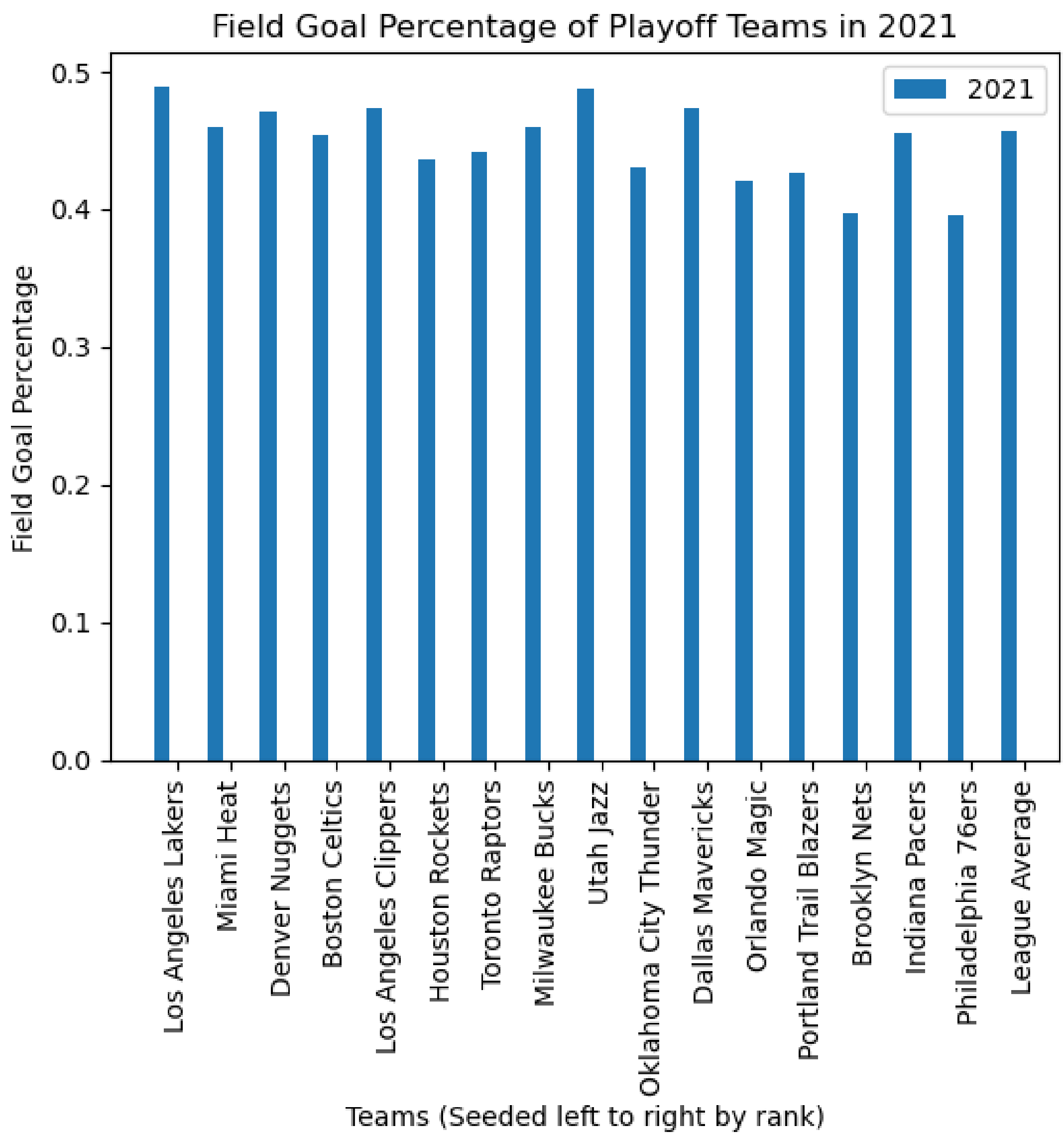
1.1.3 Statistic: Field Goal Percentage (FG%)

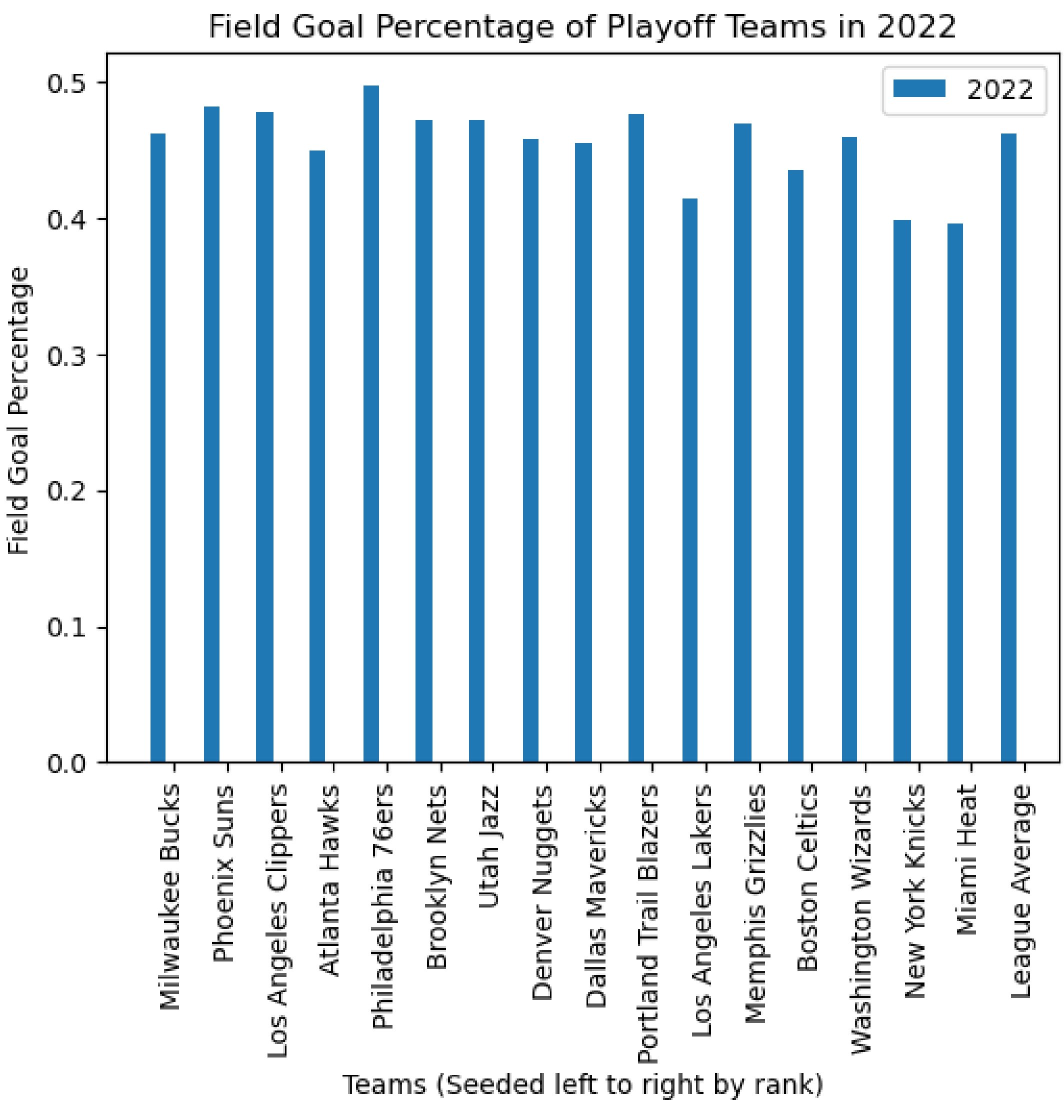
[5]:
```
#Since the playoff teams are different per year, we cannot add them on the same␣
↪plot, so there will be multiple plots per year.
year = 2020

for i in playoff_list:
    #gathering all relevant data columns for the playoff year
    teams = i["Team"]
    field_goal_pct = i["FG%"]

    # Set the width of the bars
```

```
    bar_width = 0.3
    # Set the positions of the bars on the x-axis
    bar_positions = range(len(teams))

    #FG plots ///////////////////////////////////////////////////////////////
↪//////
    # Plot the bars
    plt.bar(bar_positions, field_goal_pct, width=bar_width, label=str(year))

    # Add labels, title, and legend
    plt.xlabel("Teams (Seeded left to right by rank)")
    plt.ylabel("Field Goal Percentage")
    plt.title("Field Goal Percentage of Playoff Teams in " + str(year))
    plt.xticks([r + bar_width for r in range(len(teams))], teams, rotation=90)
    plt.legend()

    # Display the graph
    plt.show()

    year += 1
```



Field Goal Percentage of Playoff Teams in 2020
2020
Field Goal Percentage
0.0
0.1
0.2
0.3
0.4
0.5
Boston Celtics
Golden State Warriors
Dallas Mavericks
Miami Heat
Phoenix Suns
Memphis Grizzlies
Philadelphia 76ers
Milwaukee Bucks
New Orleans Pelicans
Minnesota Timberwolves
Toronto Raptors
Utah Jazz
Denver Nuggets
Atlanta Hawks
Chicago Bulls
Brooklyn Nets
League Average
Teams (Seeded left to right by rank)

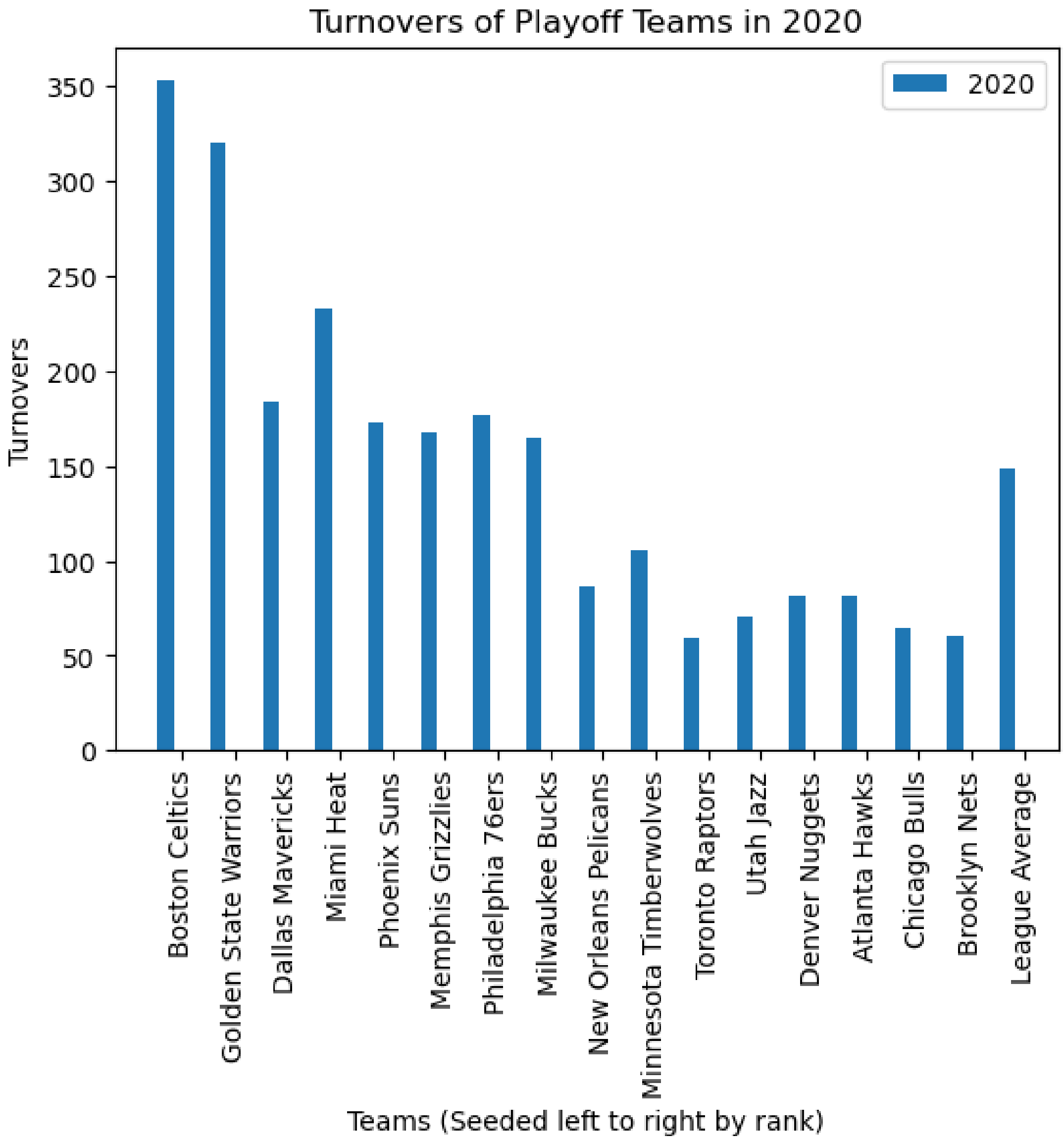

Field Goal Percentage of Playoff Teams in 2021
2021
Field Goal Percentage
0.0
0.1
0.2
0.3
0.4
0.5
Los Angeles Lakers
Miami Heat
Denver Nuggets
Boston Celtics
Los Angeles Clippers
Houston Rockets
Toronto Raptors
Milwaukee Bucks
Utah Jazz
Oklahoma City Thunder
Dallas Mavericks
Orlando Magic
Portland Trail Blazers
Brooklyn Nets
Indiana Pacers
Philadelphia 76ers
League Average
Teams (Seeded left to right by rank)



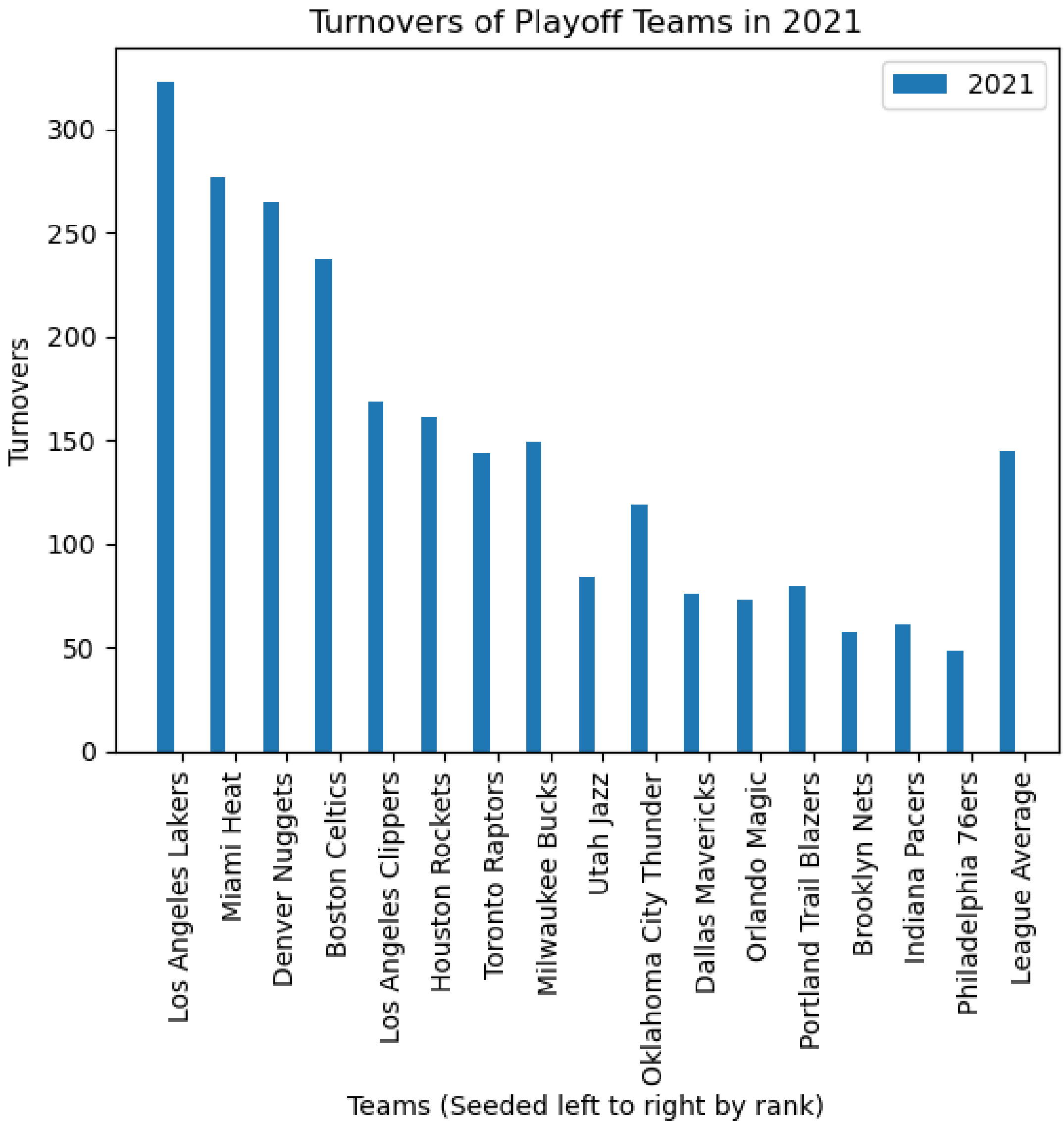
1.1.4 Statistic: Turnovers (TOV)

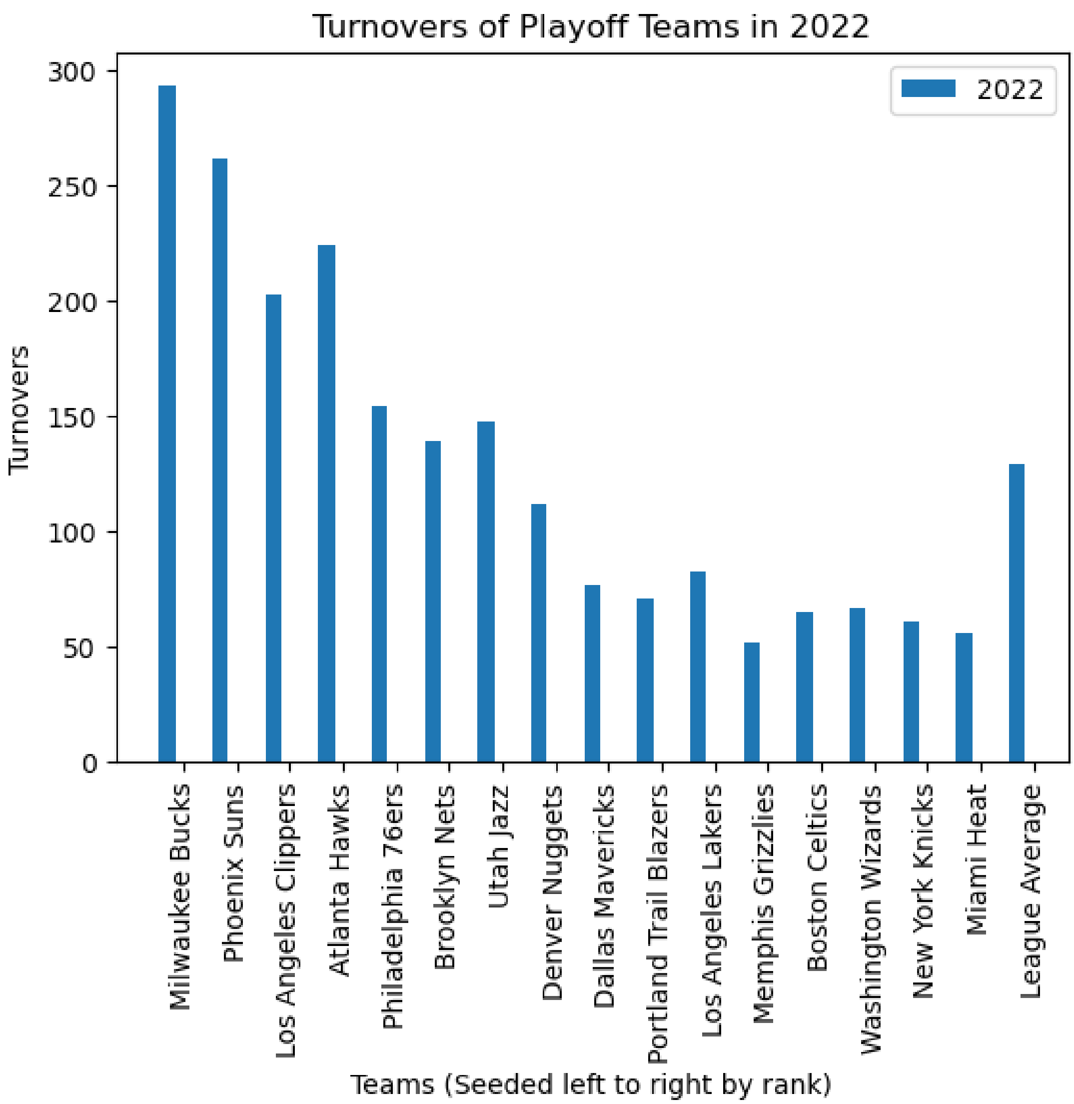
[6]:
```
#Since the playoff teams are different per year, we cannot add them on the same␣
↪plot, so there will be multiple plots per year.
year = 2020

for i in playoff_list:
    #gathering all relevant data columns for the playoff year
    teams = i["Team"]
    turnovers = i["TOV"]

    # Set the width of the bars
```

```
    bar_width = 0.3
    # Set the positions of the bars on the x-axis
    bar_positions = range(len(teams))


    #Turnovers plots ////////////////////////////////////////////////////////////////////////
    # Plot the bars
    plt.bar(bar_positions, turnovers, width=bar_width, label=str(year))

    # Add labels, title, and legend
    plt.xlabel("Teams (Seeded left to right by rank)")
    plt.ylabel("Turnovers")
    plt.title("Turnovers of Playoff Teams in " + str(year))
    plt.xticks([r + bar_width for r in range(len(teams))], teams, rotation=90)
    plt.legend()

    # Display the graph
    plt.show()

    year += 1
```



Turnovers of Playoff Teams in 2020
2020
Turnovers
0
50
100
150
200
250
300
350
Boston Celtics
Golden State Warriors
Dallas Mavericks
Miami Heat
Phoenix Suns
Memphis Grizzlies
Philadelphia 76ers
Milwaukee Bucks
New Orleans Pelicans
Minnesota Timberwolves
Toronto Raptors
Utah Jazz
Denver Nuggets
Atlanta Hawks
Chicago Bulls
Brooklyn Nets
League Average
Teams (Seeded left to right by rank)

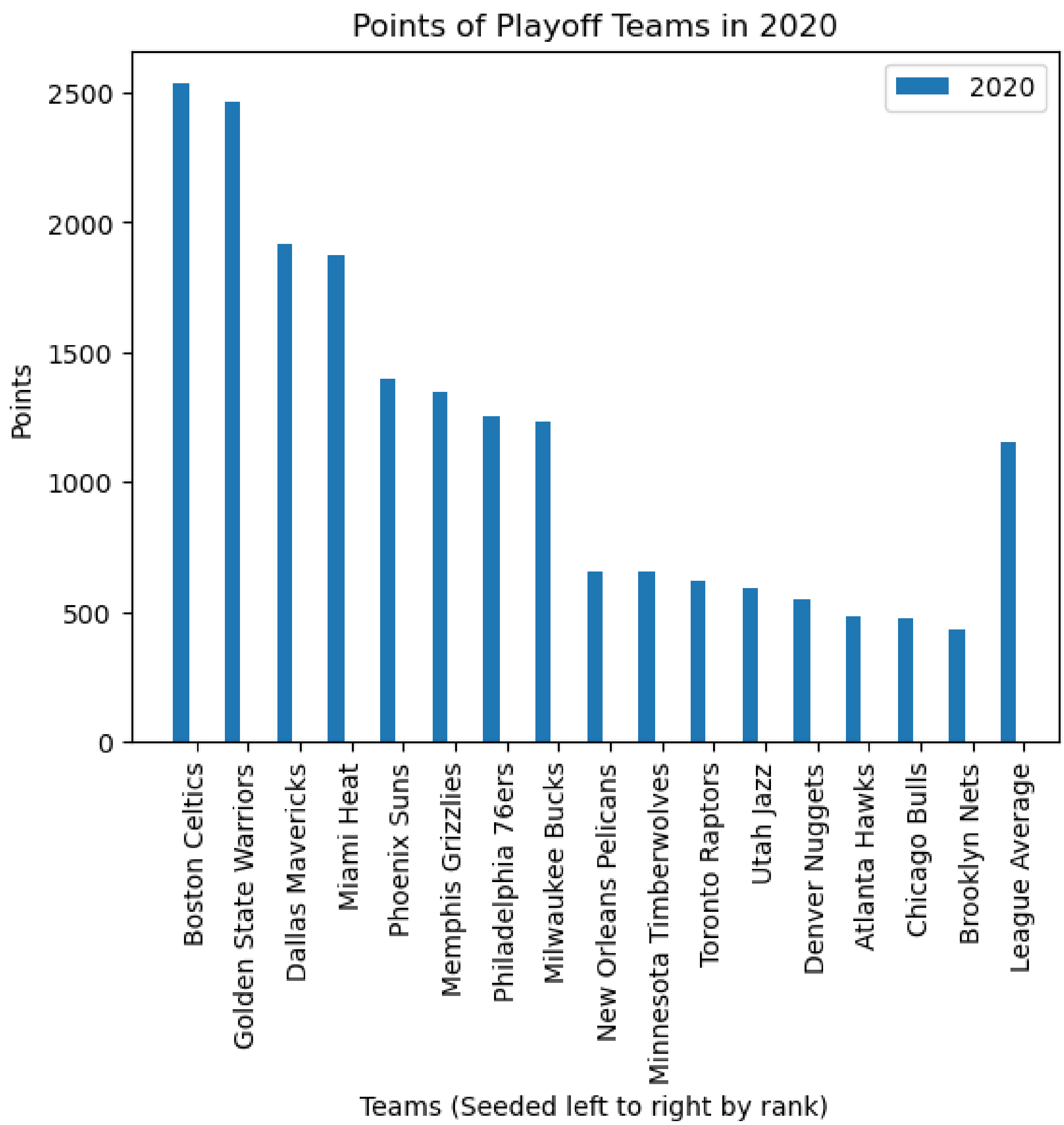

Turnovers of Playoff Teams in 2021
2021
Turnovers
0
50
100
150
200
250
300
Los Angeles Lakers
Miami Heat
Denver Nuggets
Boston Celtics
Los Angeles Clippers
Houston Rockets
Toronto Raptors
Milwaukee Bucks
Utah Jazz
Oklahoma City Thunder
Dallas Mavericks
Orlando Magic
Portland Trail Blazers
Brooklyn Nets
Indiana Pacers
Philadelphia 76ers
League Average
Teams (Seeded left to right by rank)



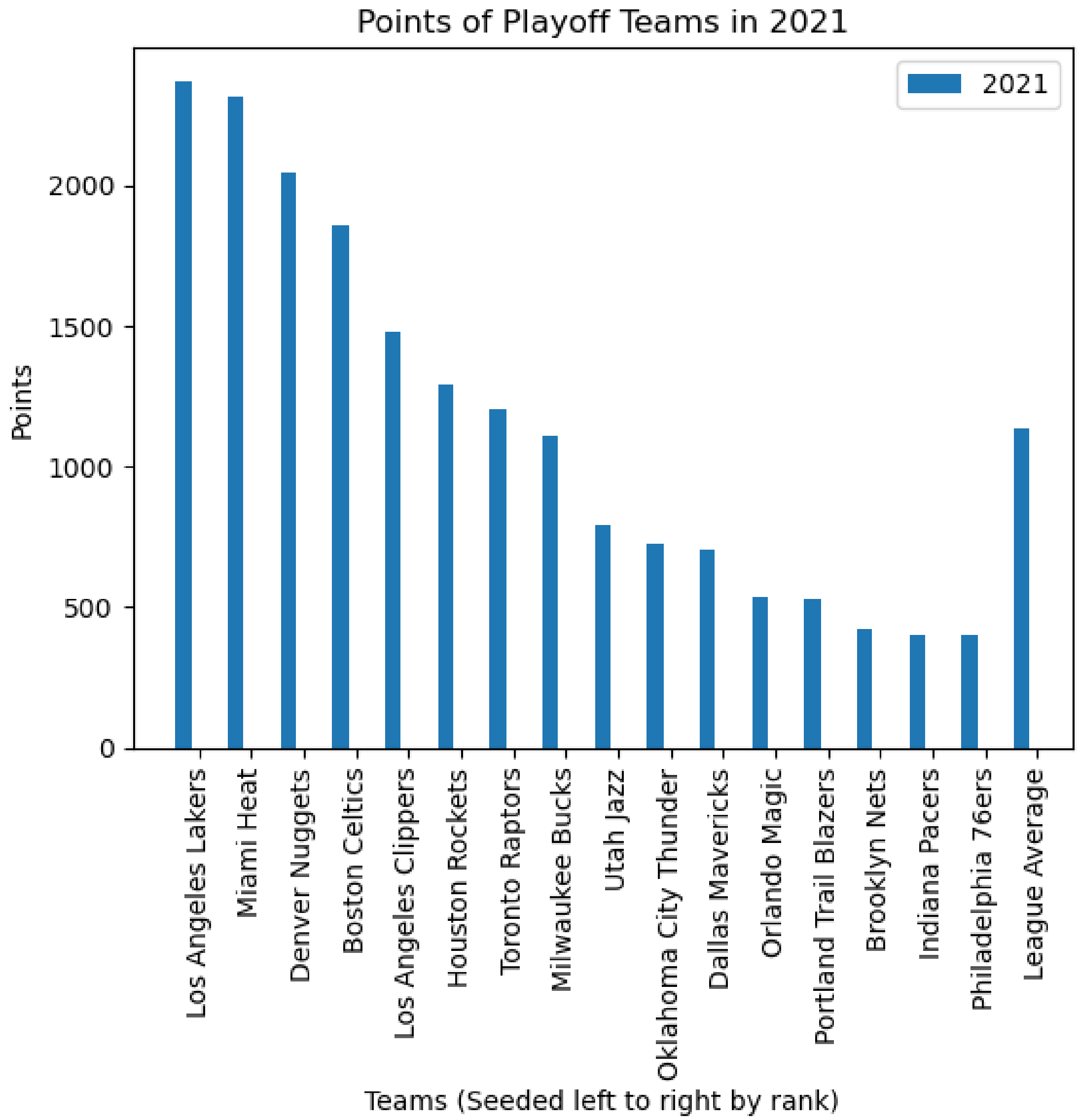
1.1.5 Statistic: Points (PTS)

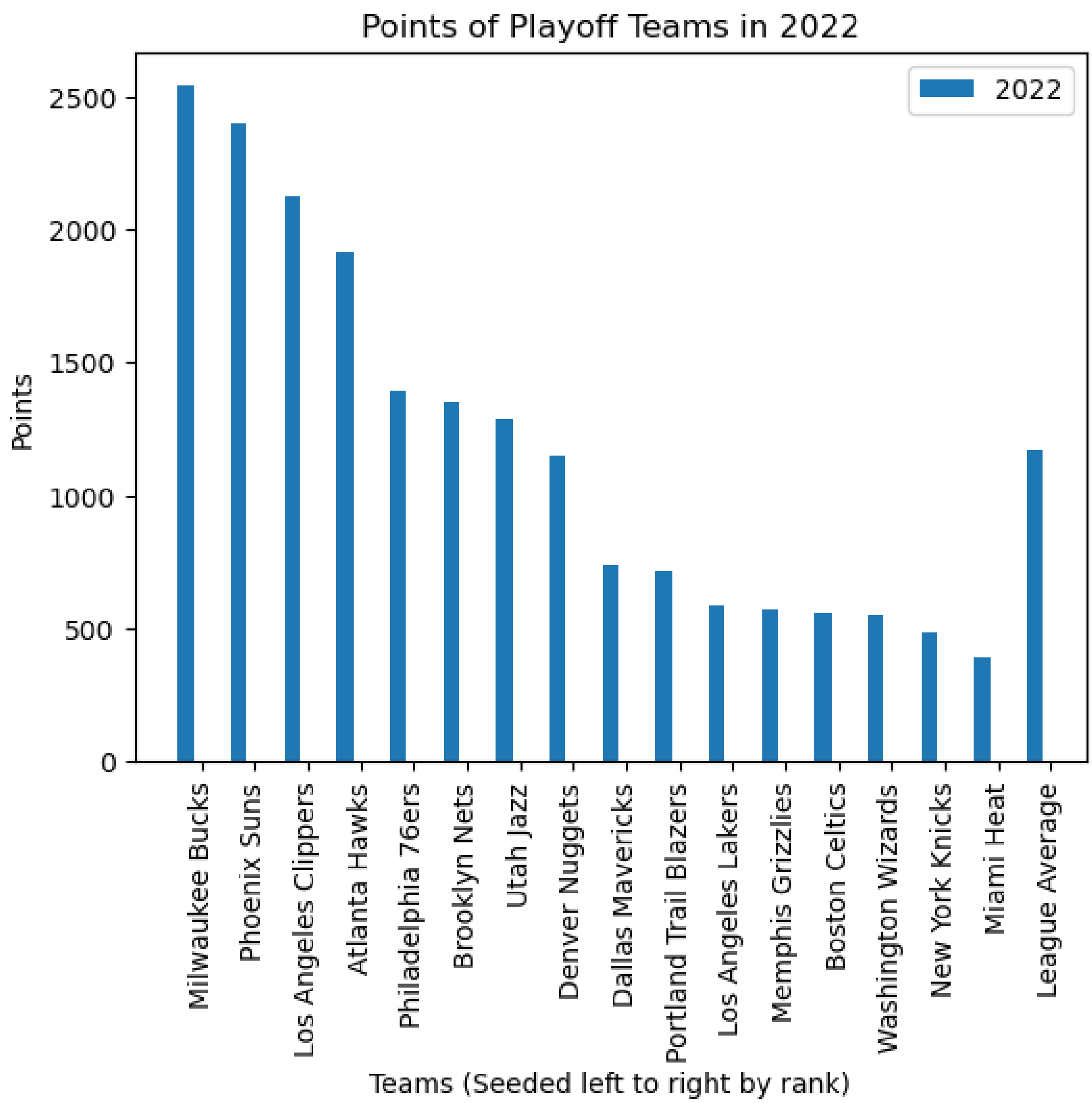
[7]:
```
#Since the playoff teams are different per year, we cannot add them on the same␣
↪plot, so there will be multiple plots per year.
year = 2020

for i in playoff_list:
    #gathering all relevant data columns for the playoff year
    teams = i["Team"]
    points = i["PTS"]

    # Set the width of the bars
```

```
    bar_width = 0.3
    # Set the positions of the bars on the x-axis
    bar_positions = range(len(teams))

    #Total Points plots ////////////////////////////////////////////////////////
↪////////////////
    # Plot the bars
    plt.bar(bar_positions, points, width=bar_width, label=str(year))

    # Add labels, title, and legend
    plt.xlabel("Teams (Seeded left to right by rank)")
    plt.ylabel("Points")
    plt.title("Points of Playoff Teams in " + str(year))
    plt.xticks([r + bar_width for r in range(len(teams))], teams, rotation=90)
    plt.legend()

    # Display the graph
    plt.show()

    year += 1
```



Points of Playoff Teams in 2020
2020
Points
2500
2000
1500
1000
500
0
Boston Celtics
Golden State Warriors
Dallas Mavericks
Miami Heat
Phoenix Suns
Memphis Grizzlies
Philadelphia 76ers
Milwaukee Bucks
New Orleans Pelicans
Minnesota Timberwolves
Toronto Raptors
Utah Jazz
Denver Nuggets
Atlanta Hawks
Chicago Bulls
Brooklyn Nets
League Average
Teams (Seeded left to right by rank)

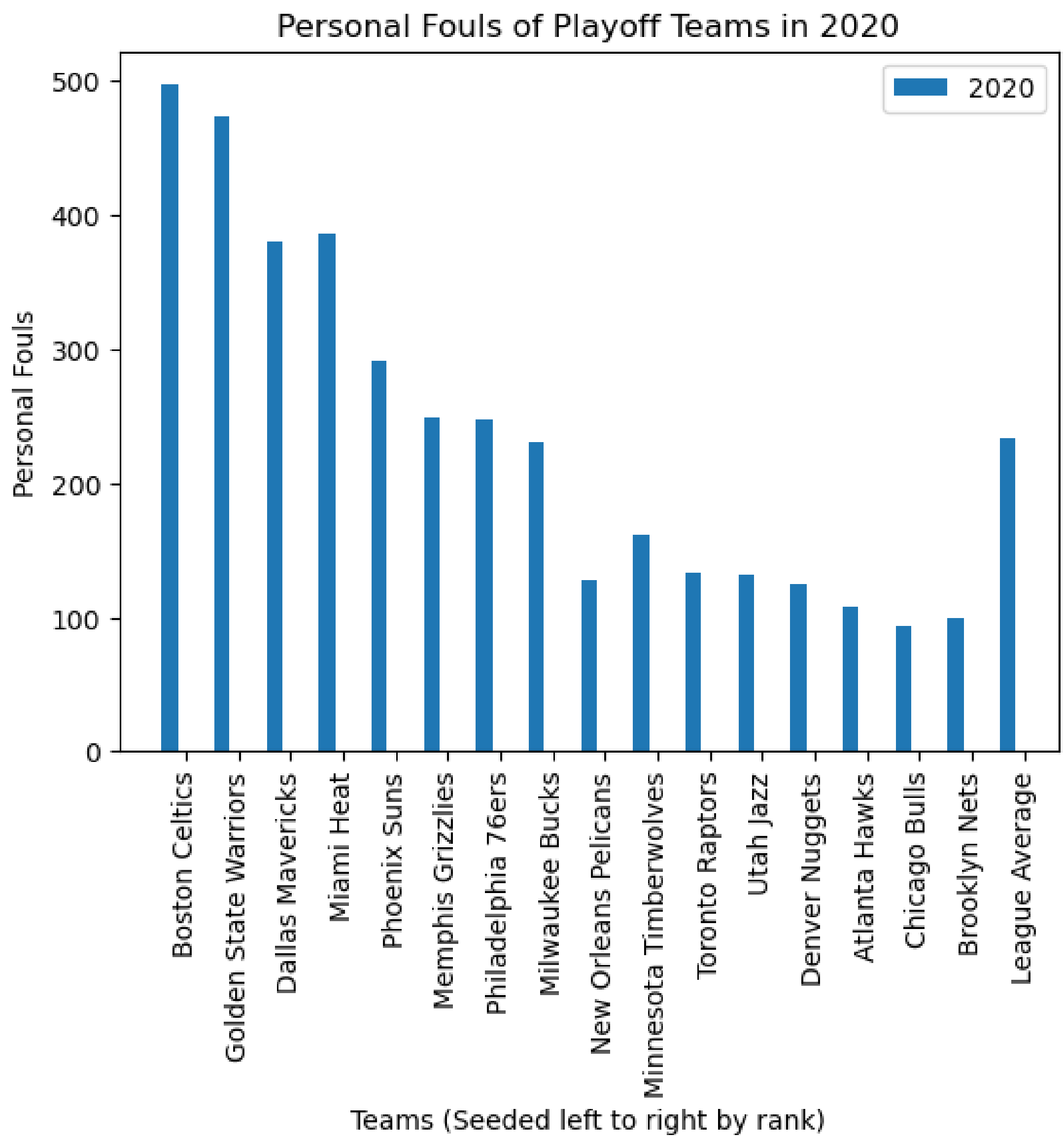

Points of Playoff Teams in 2021
2021
Points
2000
1500
1000
500
0
Los Angeles Lakers
Miami Heat
Denver Nuggets
Boston Celtics
Los Angeles Clippers
Houston Rockets
Toronto Raptors
Milwaukee Bucks
Utah Jazz
Oklahoma City Thunder
Dallas Mavericks
Orlando Magic
Portland Trail Blazers
Brooklyn Nets
Indiana Pacers
Philadelphia 76ers
League Average
Teams (Seeded left to right by rank)

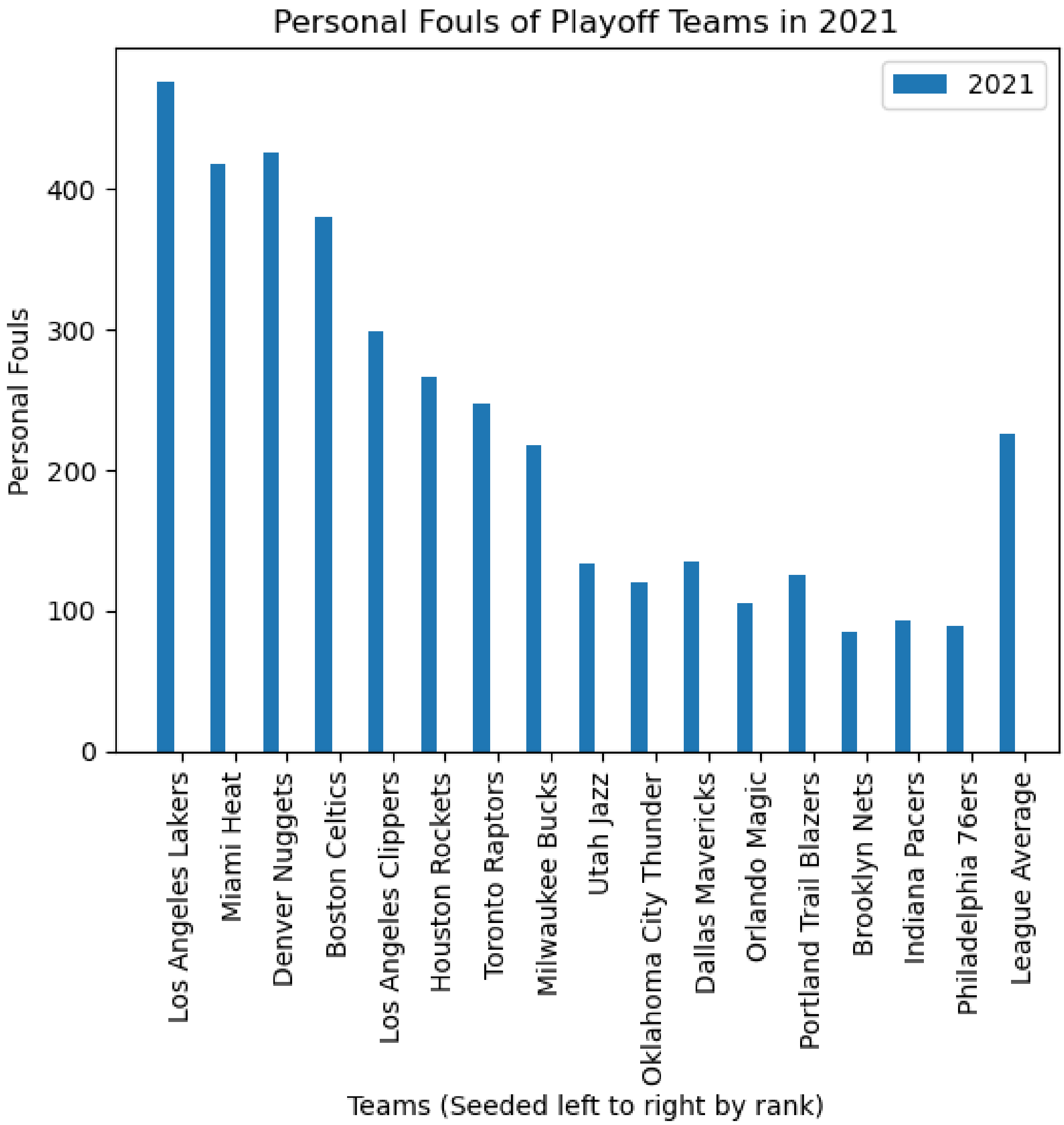


1.1.6 Statistic: Personal Fouls (PF)

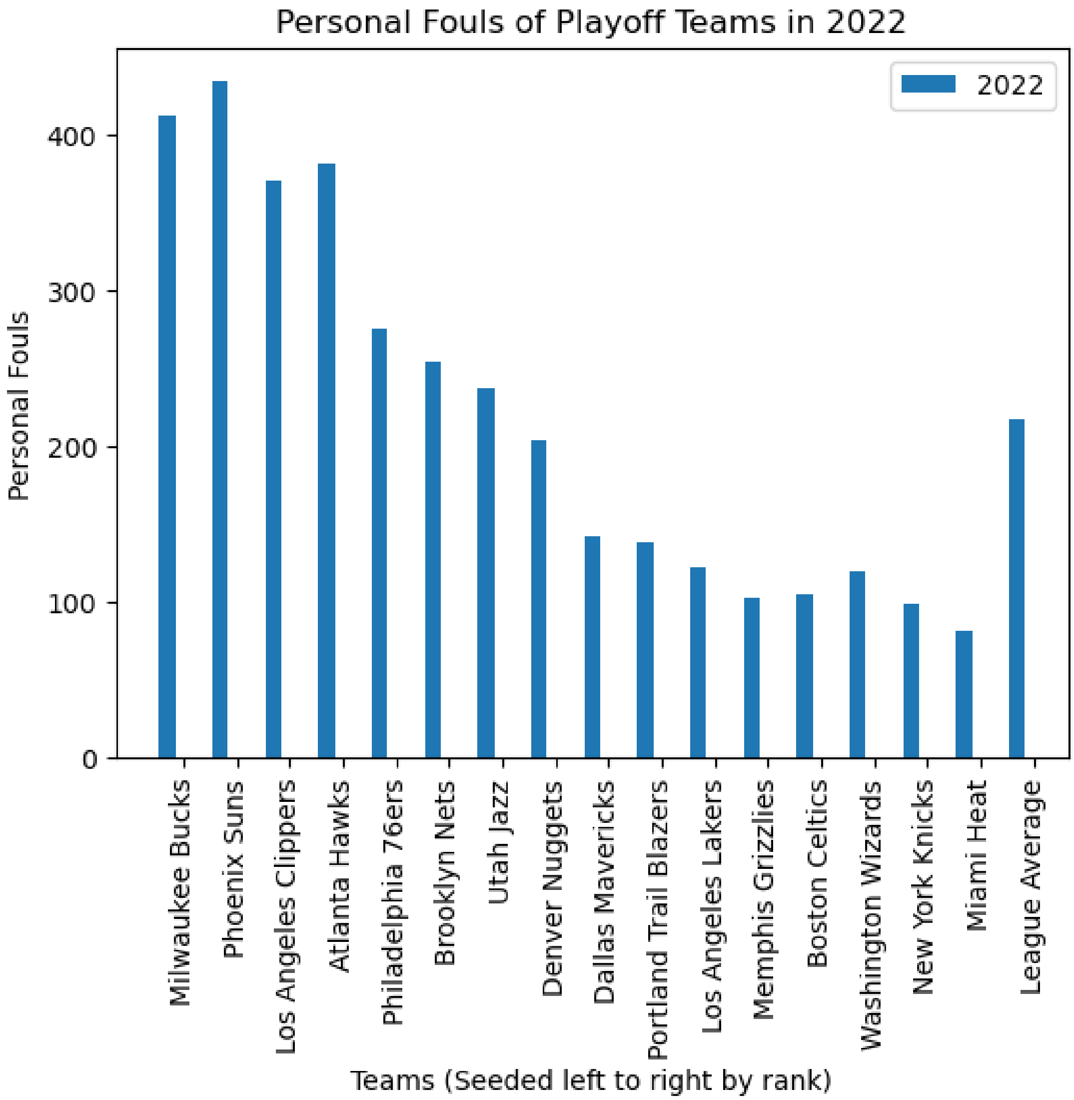
[8]:
```
#Since the playoff teams are different per year, we cannot add them on the same␣
↪plot, so there will be multiple plots per year.
year = 2020

for i in playoff_list:
    #gathering all relevant data columns for the playoff year
    teams = i["Team"]
    personal_fouls = i["PF"]

    # Set the width of the bars
```

```
    bar_width = 0.3
    # Set the positions of the bars on the x-axis
    bar_positions = range(len(teams))

    #Personal Fouls plots ///////////////////////////////////////////////////////
↪//////////////////
    # Plot the bars
    plt.bar(bar_positions, personal_fouls, width=bar_width, label=str(year))

    # Add labels, title, and legend
    plt.xlabel("Teams (Seeded left to right by rank)")
    plt.ylabel("Personal Fouls")
    plt.title("Personal Fouls of Playoff Teams in " + str(year))
    plt.xticks([r + bar_width for r in range(len(teams))], teams, rotation=90)
    plt.legend()

    # Display the graph
    plt.show()

    year += 1
```


Personal Fouls of Playoff Teams in 2020
2020
Personal Fouls
0
100
200
300
400
500
Boston Celtics
Golden State Warriors
Dallas Mavericks
Miami Heat
Phoenix Suns
Memphis Grizzlies
Philadelphia 76ers
Milwaukee Bucks
New Orleans Pelicans
Minnesota Timberwolves
Toronto Raptors
Utah Jazz
Denver Nuggets
Atlanta Hawks
Chicago Bulls
Brooklyn Nets
League Average
Teams (Seeded left to right by rank)



Personal Fouls of Playoff Teams in 2021
2021
Personal Fouls
0
100
200
300
400
Los Angeles Lakers
Miami Heat
Denver Nuggets
Boston Celtics
Los Angeles Clippers
Houston Rockets
Toronto Raptors
Milwaukee Bucks
Utah Jazz
Oklahoma City Thunder
Dallas Mavericks
Orlando Magic
Portland Trail Blazers
Brooklyn Nets
Indiana Pacers
Philadelphia 76ers
League Average
Teams (Seeded left to right by rank)



Q: What do you notice about the data?

`A: Almost all of the data shows a decreasing trend as we go lower and lower in the ranks. Those`

Q: Looking at the data, why do you think the plot in for example, turnovers in each year decrease as we supposedly go down in ranking (meaning the supposed "best team" has the most turnovers)?

`A: This could be caused by a number of factors but since the higher ranked teams end up playing`

Lets verify this with a linear regression.

Now let's look at the data with a fitted regression line for each scatter plot to see trends in various statistics for each of the last three playoff years.

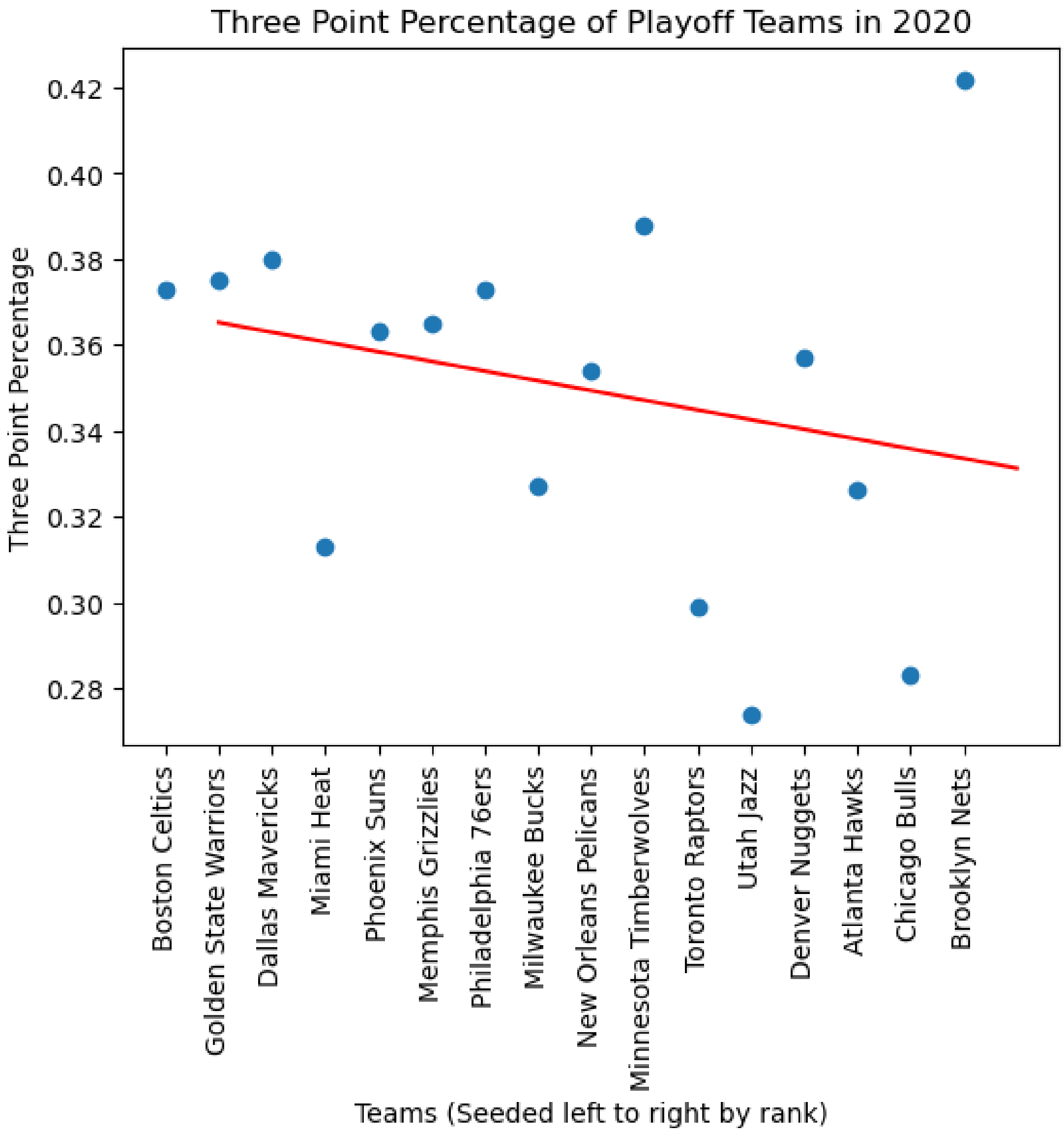
1.2 Part 2: Regression and Analysis

[9]:
```
import numpy as np
#same code as last part, but adding regression lines to each plot

#since league average is in the last row of the df, and does not have a team␣
↪rank, we must remove it
data_2020_n = data_2020.drop(16)
data_2021_n = data_2021.drop(16)
data_2022_n = data_2022.drop(16)

playoff_list_2 = [data_2020_n, data_2021_n, data_2022_n]
```

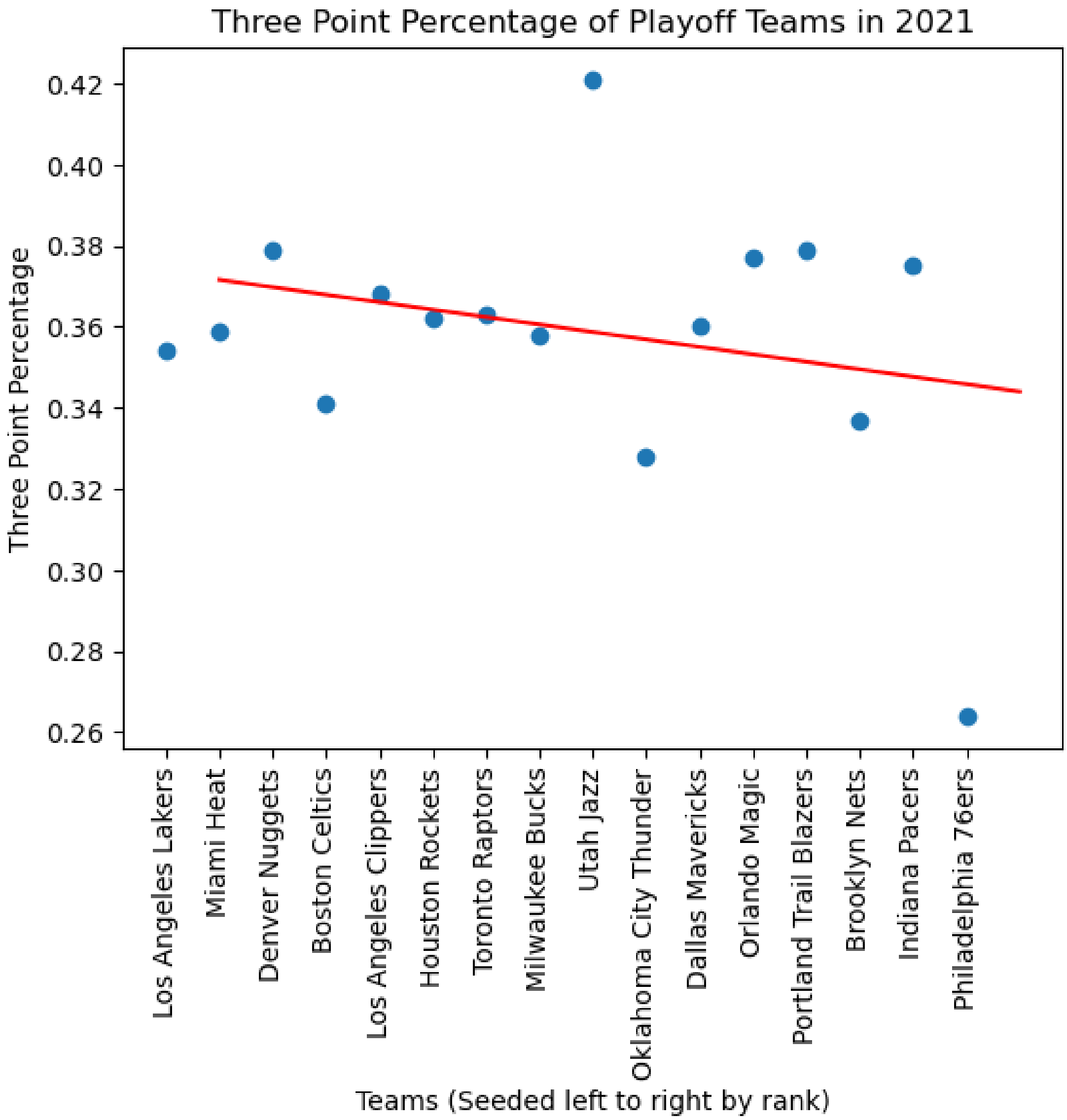
1.2.1 Regression: Three Point Percentage (3P%)

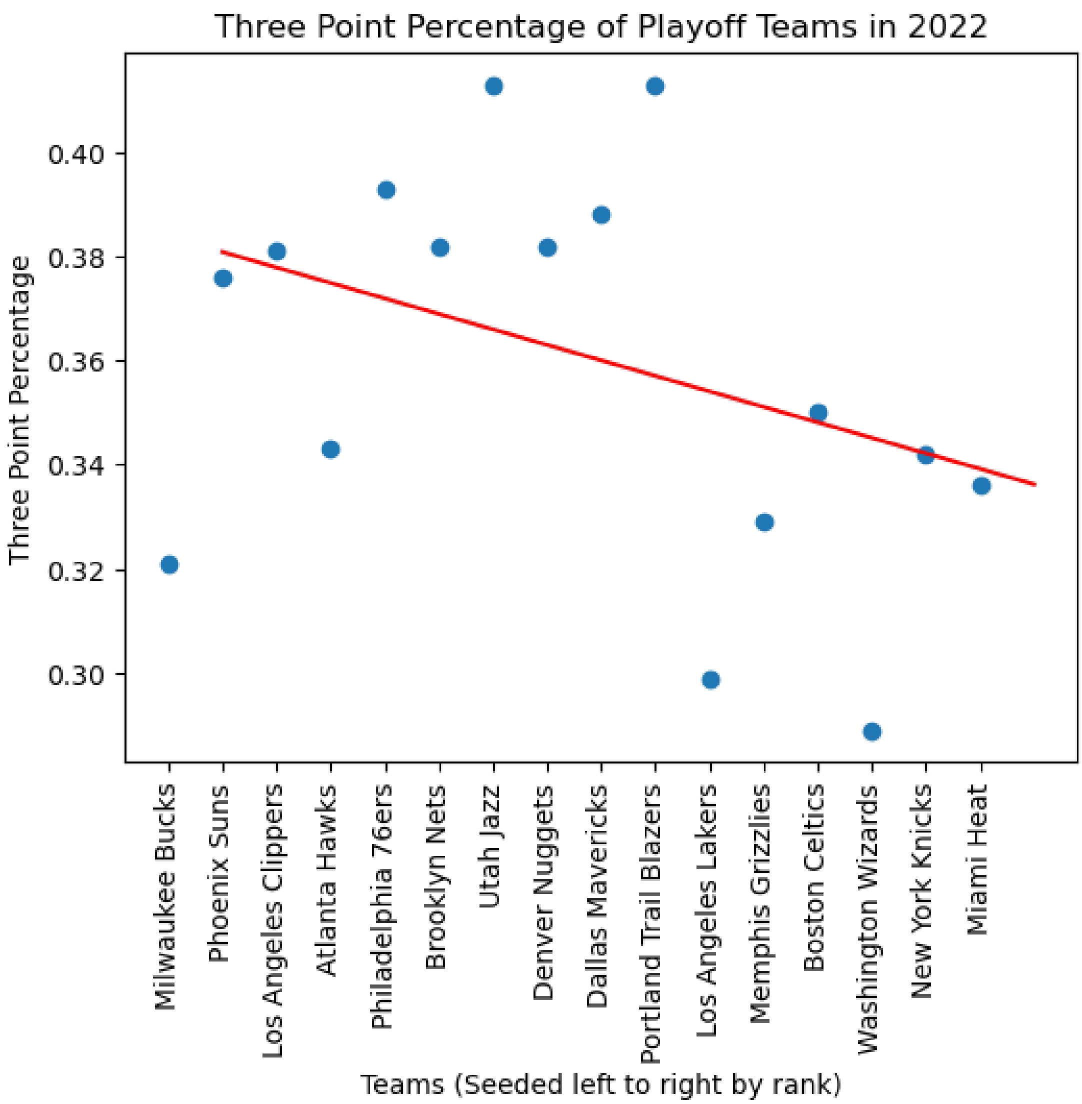
[10]:
```
year = 2020

for i in playoff_list_2:
    #gathering all relevant data columns for the playoff year
    teams = i["Team"]
    three_point_pct = i["3P%"]
    offensive_rebounds = i["ORB"]
    field_goal_pct = i["FG%"]
    turnovers = i["TOV"]
    points = i["PTS"]
    personal_fouls = i["PF"]

    #3 pt percentage plots ////////////////////////////////////////////////////
↪//////////////////
    # Plot the scatter plot
    plt.scatter(teams, three_point_pct)

    # Add labels, title, and legend
    plt.xlabel("Teams (Seeded left to right by rank)")
    plt.ylabel("Three Point Percentage")
    plt.title("Three Point Percentage of Playoff Teams in " + str(year))
    plt.xticks(teams, rotation=90)

    #make regression plot
    #need to use rank because team names are not numerical for regression␣
↪purposes, achieves same goal since teams are in order by rank
    a, b = np.polyfit(i["Rk"], three_point_pct, 1)
    plt.plot(i["Rk"], a * (i["Rk"]) + b, color='red')

    # Display the graph
    plt.show()
```

```
year += 1
```



Three Point Percentage of Playoff Teams in 2020
Three Point Percentage
0.42
0.40
0.38
0.36
0.34
0.32
0.30
0.28
Boston Celtics
Golden State Warriors
Dallas Mavericks
Miami Heat
Phoenix Suns
Memphis Grizzlies
Philadelphia 76ers
Milwaukee Bucks
New Orleans Pelicans
Minnesota Timberwolves
Toronto Raptors
Utah Jazz
Denver Nuggets
Atlanta Hawks
Chicago Bulls
Brooklyn Nets
Teams (Seeded left to right by rank)

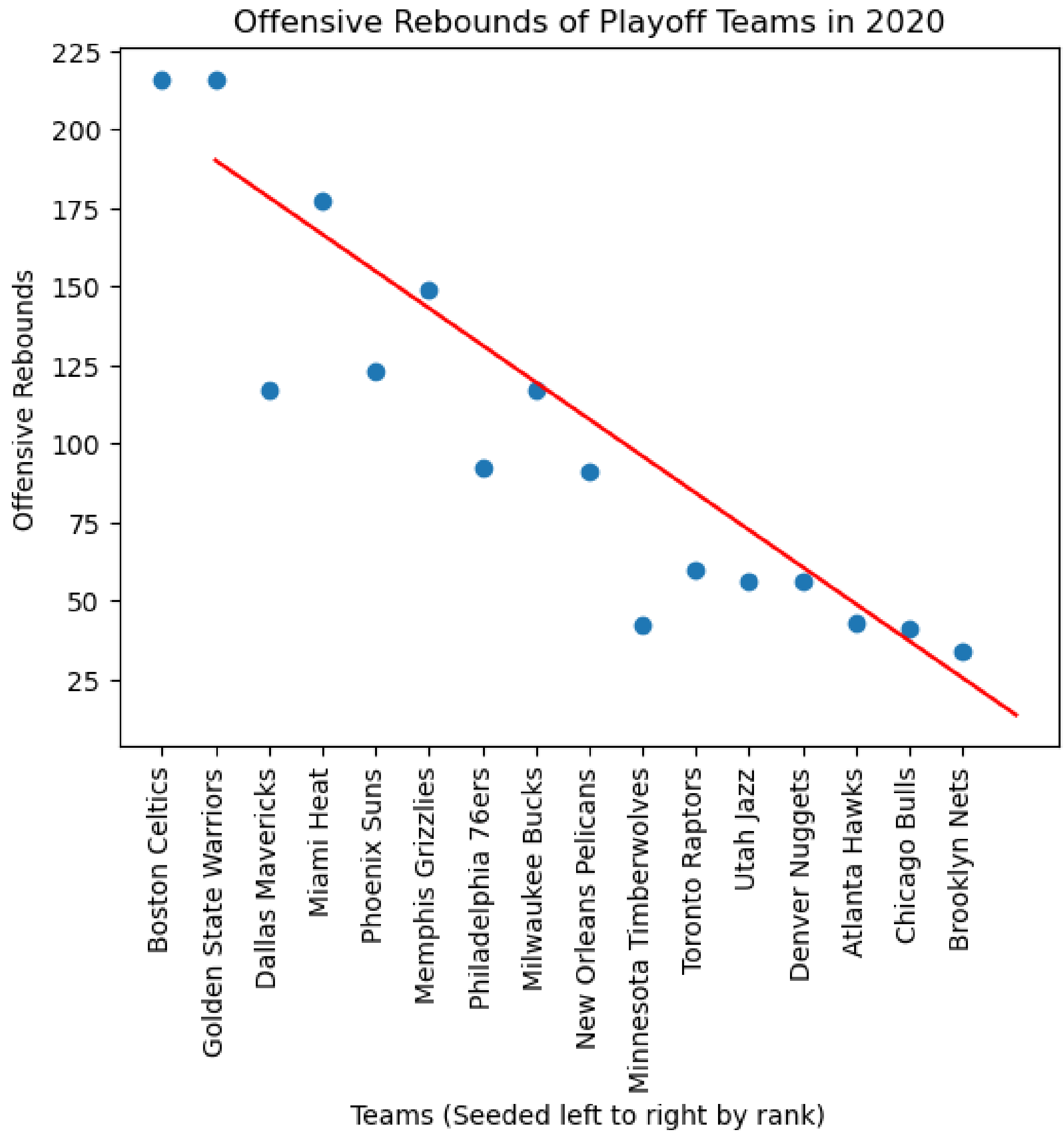

Three Point Percentage of Playoff Teams in 2021
Three Point Percentage
0.42
0.40
0.38
0.36
0.34
0.32
0.30
0.28
0.26
Los Angeles Lakers
Miami Heat
Denver Nuggets
Boston Celtics
Los Angeles Clippers
Houston Rockets
Toronto Raptors
Milwaukee Bucks
Utah Jazz
Oklahoma City Thunder
Dallas Mavericks
Orlando Magic
Portland Trail Blazers
Brooklyn Nets
Indiana Pacers
Philadelphia 76ers
Teams (Seeded left to right by rank)

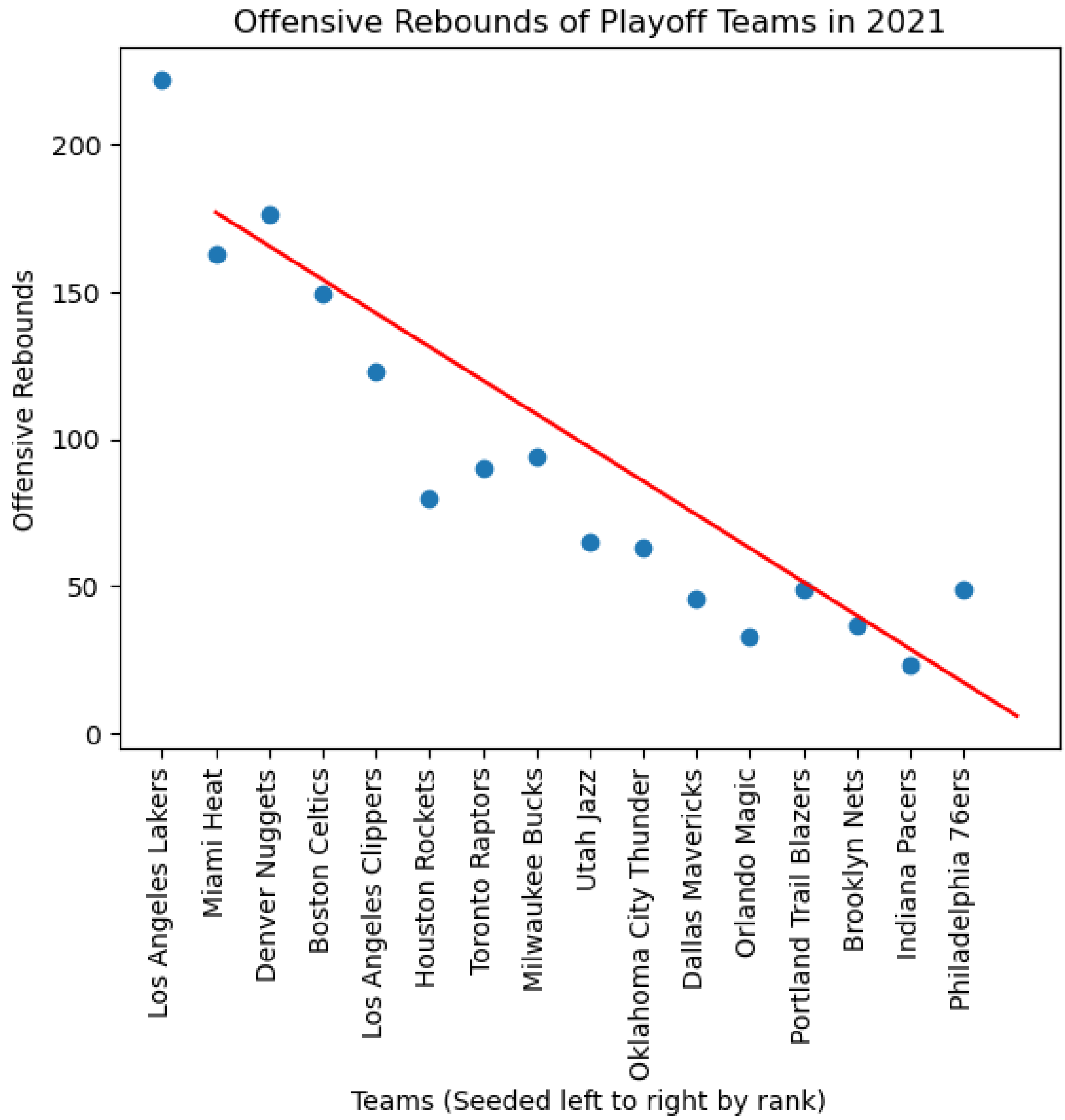


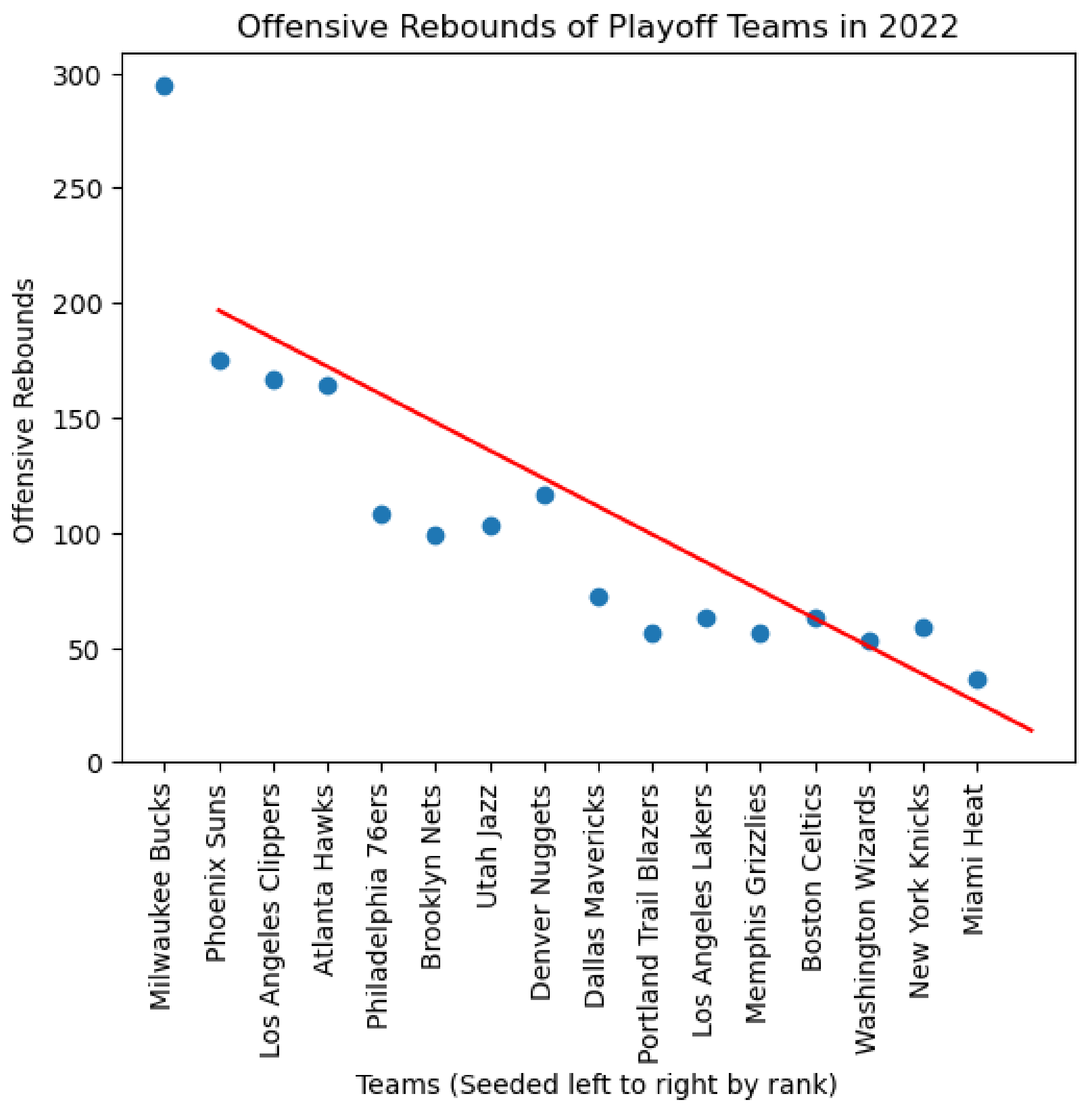
1.2.2 Regression: Offensive Rebounds (ORB)

[11]:
```
year = 2020

for i in playoff_list_2:
    #gathering all relevant data columns for the playoff year
    teams = i["Team"]
    three_point_pct = i["3P%"]
    offensive_rebounds = i["ORB"]
    field_goal_pct = i["FG%"]
    turnovers = i["TOV"]
    points = i["PTS"]
    personal_fouls = i["PF"]
```

```
    #ORB plots //////////////////////////////////////////////////////////////////
↪///////
    # Plot the scatter plot
    plt.scatter(teams, offensive_rebounds)

    # Add labels, title, and legend
    plt.xlabel("Teams (Seeded left to right by rank)")
    plt.ylabel("Offensive Rebounds")
    plt.title("Offensive Rebounds of Playoff Teams in " + str(year))
    plt.xticks(teams, rotation=90)

    #make regression plot
    #need to use rank because team names are not numerical for regression␣
↪purposes, achieves same goal since teams are in order by rank
    a, b = np.polyfit(i["Rk"], offensive_rebounds, 1)
    plt.plot(i["Rk"], a * (i["Rk"]) + b, color='red')

    # Display the graph
    plt.show()

    year += 1
```


Offensive Rebounds of Playoff Teams in 2020
Offensive Rebounds
225
200
175
150
125
100
75
50
25
Boston Celtics
Golden State Warriors
Dallas Mavericks
Miami Heat
Phoenix Suns
Memphis Grizzlies
Philadelphia 76ers
Milwaukee Bucks
New Orleans Pelicans
Minnesota Timberwolves
Toronto Raptors
Utah Jazz
Denver Nuggets
Atlanta Hawks
Chicago Bulls
Brooklyn Nets
Teams (Seeded left to right by rank)

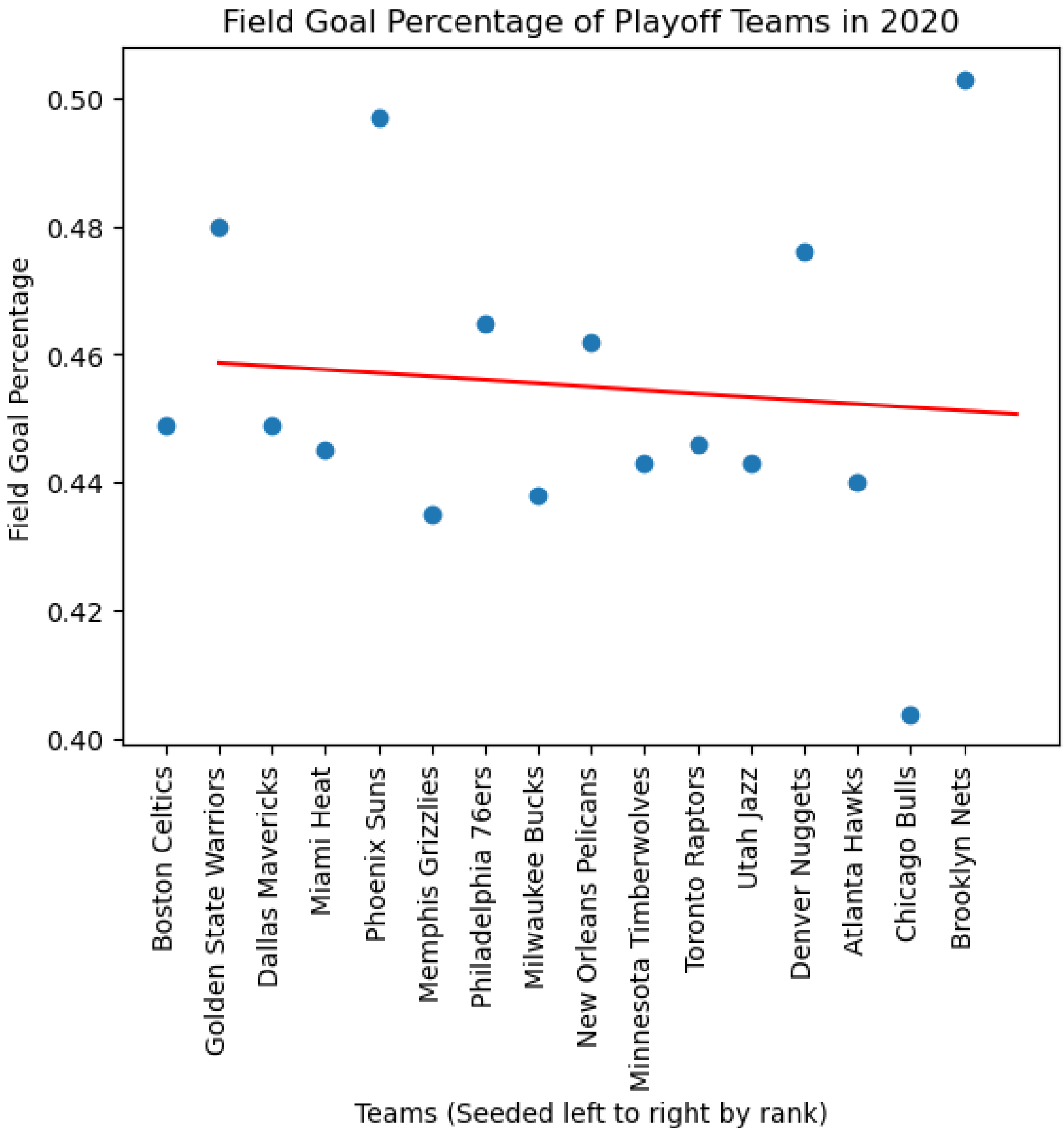

Offensive Rebounds of Playoff Teams in 2021
Offensive Rebounds
200
150
100
50
0
Los Angeles Lakers
Miami Heat
Denver Nuggets
Boston Celtics
Los Angeles Clippers
Houston Rockets
Toronto Raptors
Milwaukee Bucks
Utah Jazz
Oklahoma City Thunder
Dallas Mavericks
Orlando Magic
Portland Trail Blazers
Brooklyn Nets
Indiana Pacers
Philadelphia 76ers
Teams (Seeded left to right by rank)

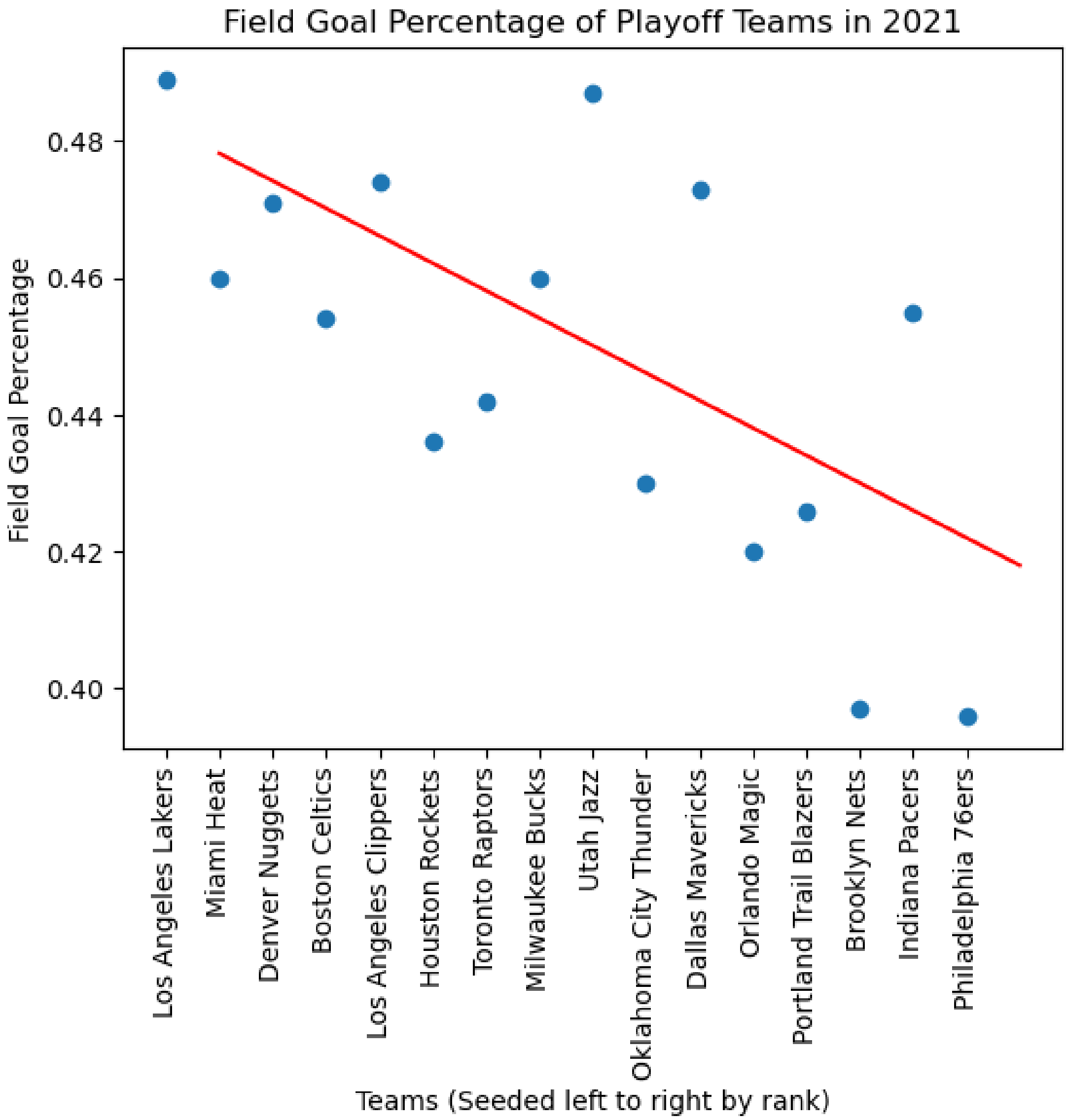


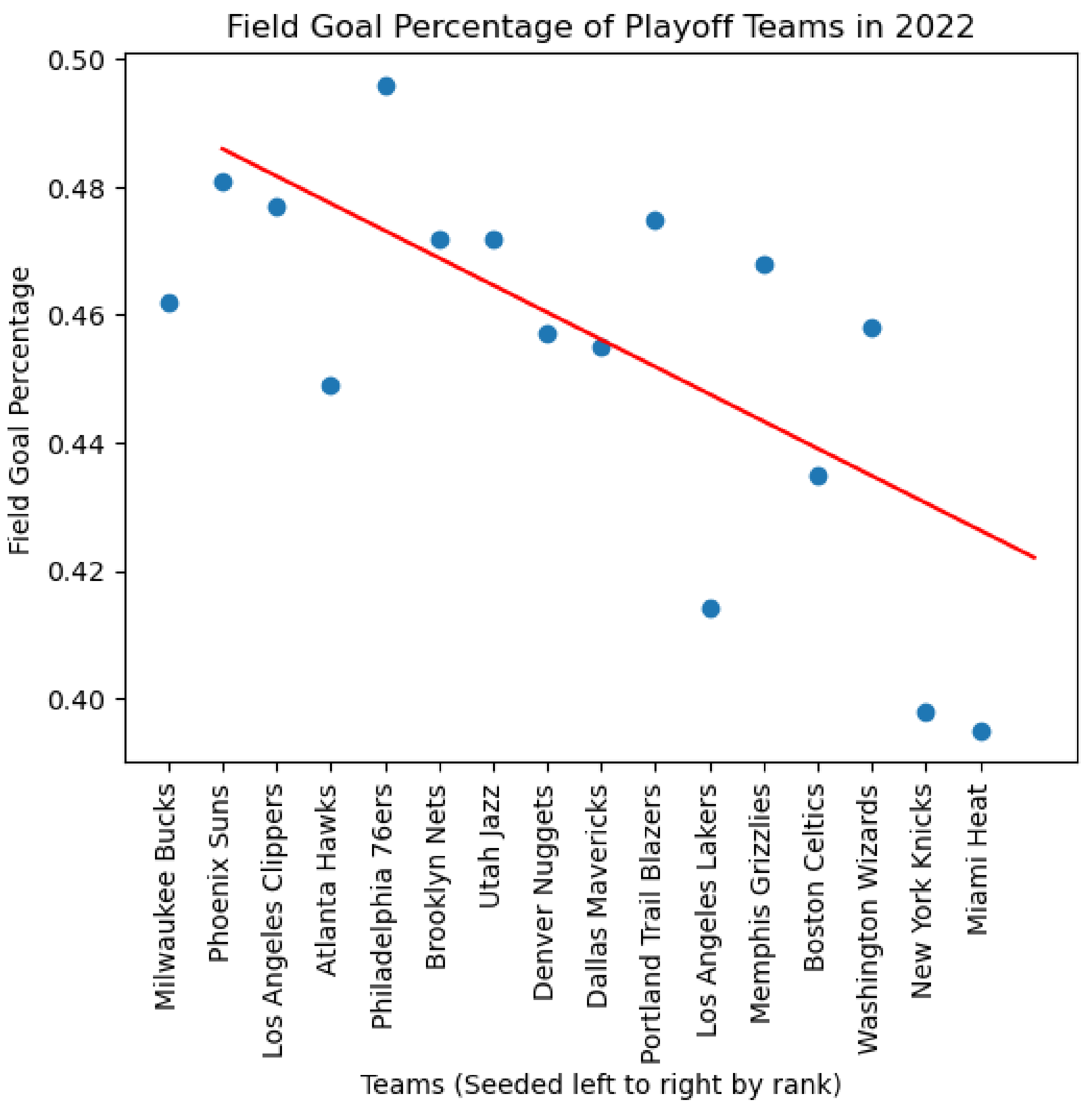
1.2.3 Regression: Field Goal Percentage (FG%)

[12]:
```
year = 2020

for i in playoff_list_2:
    #gathering all relevant data columns for the playoff year
    teams = i["Team"]
    three_point_pct = i["3P%"]
    offensive_rebounds = i["ORB"]
    field_goal_pct = i["FG%"]
    turnovers = i["TOV"]
    points = i["PTS"]
    personal_fouls = i["PF"]
```

```
    #FG plots ///////////////////////////////////////////////////////////////////
↪//////
    # Plot the scatter plot
    plt.scatter(teams, field_goal_pct)

    # Add labels, title, and legend
    plt.xlabel("Teams (Seeded left to right by rank)")
    plt.ylabel("Field Goal Percentage")
    plt.title("Field Goal Percentage of Playoff Teams in " + str(year))
    plt.xticks(teams, rotation=90)

    #make regression plot
    #need to use rank because team names are not numerical for regression␣
↪purposes, achieves same goal since teams are in order by rank
    a, b = np.polyfit(i["Rk"], field_goal_pct, 1)
    plt.plot(i["Rk"], a * (i["Rk"]) + b, color='red')

    # Display the graph
    plt.show()

    year += 1
```


Field Goal Percentage of Playoff Teams in 2020
0.50
0.48
0.46
0.44
0.42
0.40
Field Goal Percentage
Boston Celtics
Golden State Warriors
Dallas Mavericks
Miami Heat
Phoenix Suns
Memphis Grizzlies
Philadelphia 76ers
Milwaukee Bucks
New Orleans Pelicans
Minnesota Timberwolves
Toronto Raptors
Utah Jazz
Denver Nuggets
Atlanta Hawks
Chicago Bulls
Brooklyn Nets
Teams (Seeded left to right by rank)

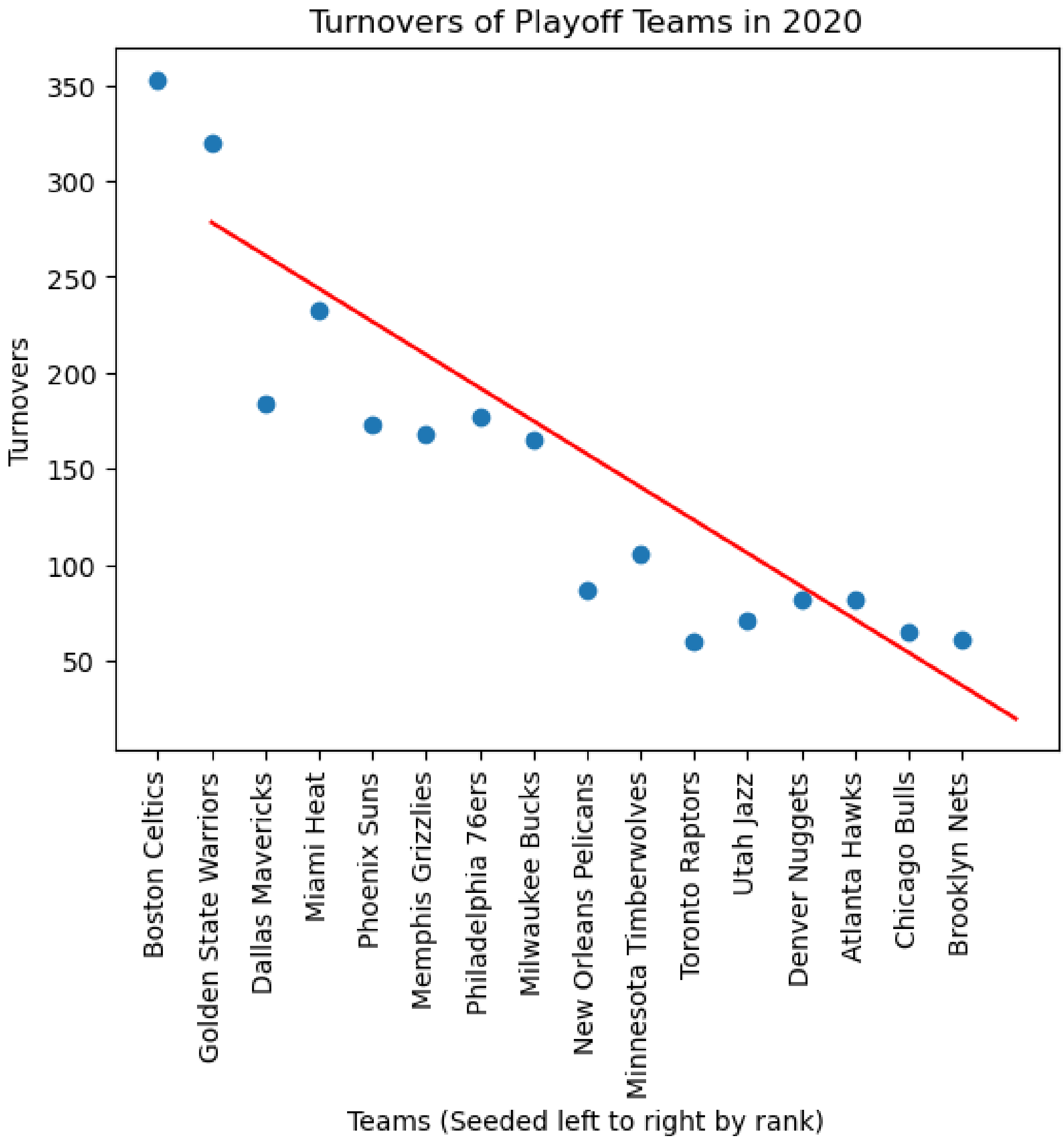

Field Goal Percentage of Playoff Teams in 2021
Field Goal Percentage
0.48
0.46
0.44
0.42
0.40
Los Angeles Lakers
Miami Heat
Denver Nuggets
Boston Celtics
Los Angeles Clippers
Houston Rockets
Toronto Raptors
Milwaukee Bucks
Utah Jazz
Oklahoma City Thunder
Dallas Mavericks
Orlando Magic
Portland Trail Blazers
Brooklyn Nets
Indiana Pacers
Philadelphia 76ers
Teams (Seeded left to right by rank)

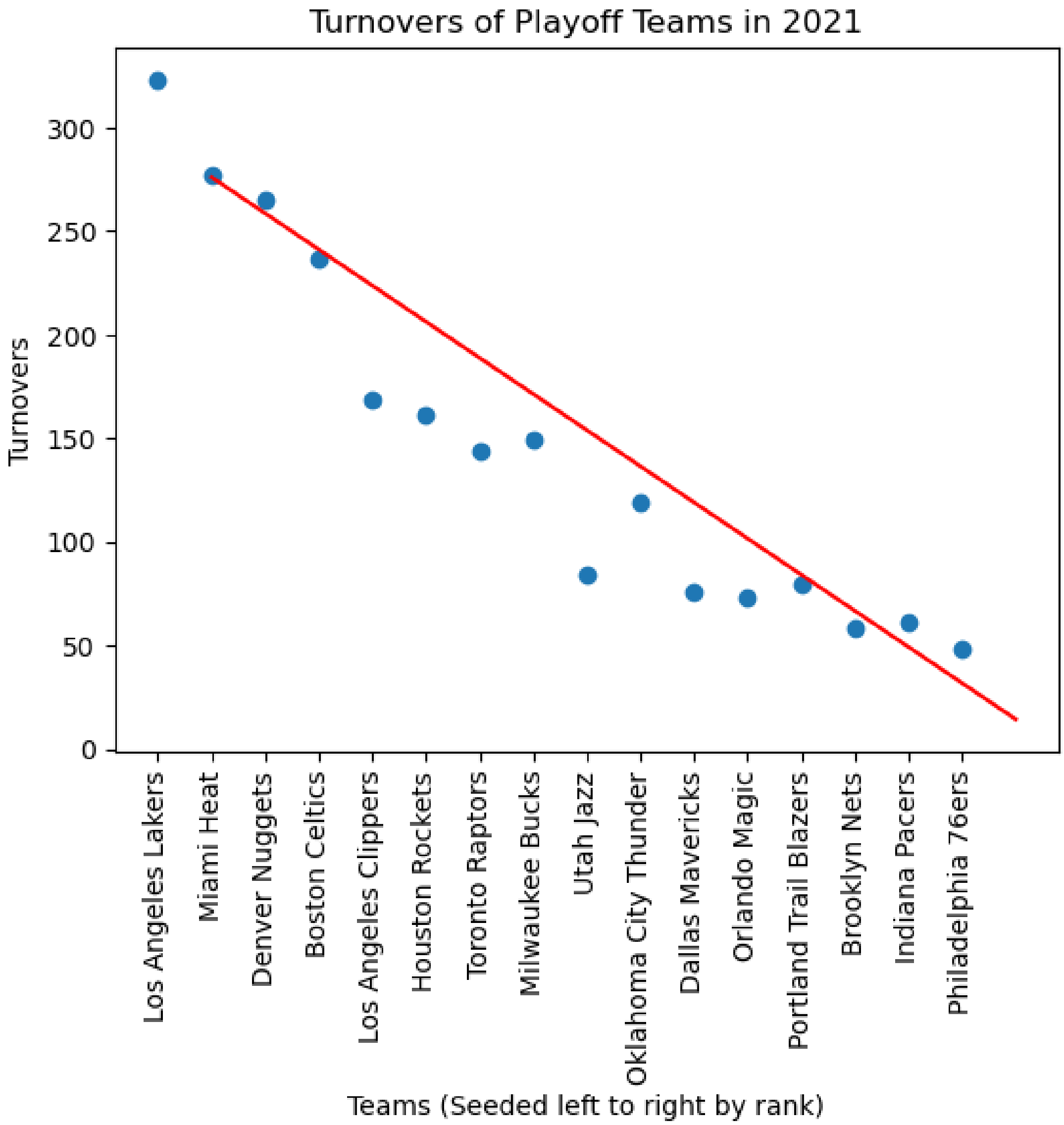


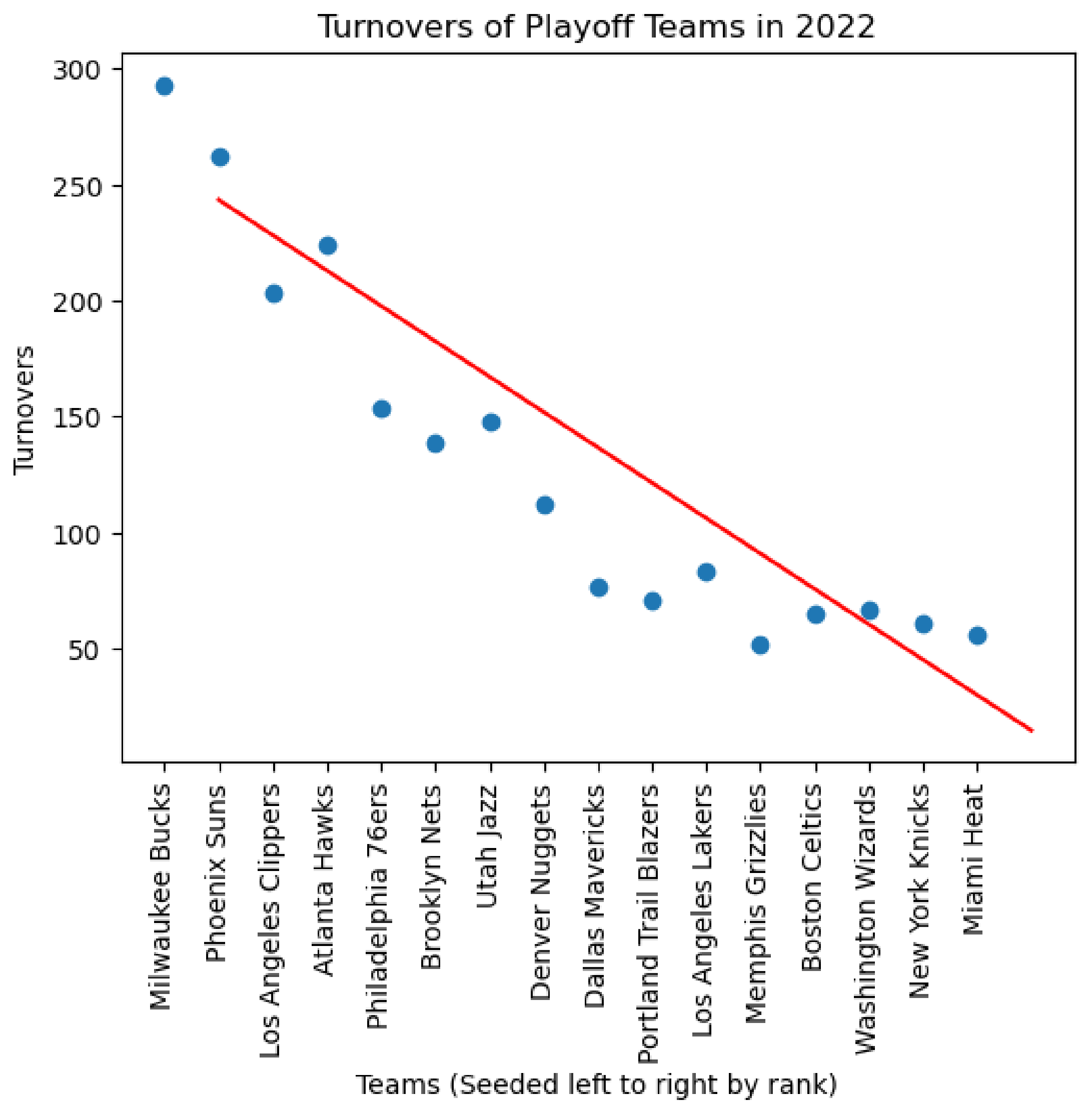
1.2.4 Regression: Turnovers (TOV)

[13]:
```
year = 2020

for i in playoff_list_2:
    #gathering all relevant data columns for the playoff year
    teams = i["Team"]
    three_point_pct = i["3P%"]
    offensive_rebounds = i["ORB"]
    field_goal_pct = i["FG%"]
    turnovers = i["TOV"]
    points = i["PTS"]
    personal_fouls = i["PF"]
```

```
#Turnovers plots /////////////////////////////////////////////////////////////////////////
# Plot the scatter plot
plt.scatter(teams, turnovers)

# Add labels, title, and legend
plt.xlabel("Teams (Seeded left to right by rank)")
plt.ylabel("Turnovers")
plt.title("Turnovers of Playoff Teams in " + str(year))
plt.xticks(teams, rotation=90)

#make regression plot
#need to use rank because team names are not numerical for regression purposes, achieves same goal since teams are in order by rank
a, b = np.polyfit(i["Rk"], turnovers, 1)
plt.plot(i["Rk"], a * (i["Rk"]) + b, color='red')

# Display the graph
plt.show()

year += 1
```


Turnovers of Playoff Teams in 2020
Turnovers
350
300
250
200
150
100
50
Boston Celtics
Golden State Warriors
Dallas Mavericks
Miami Heat
Phoenix Suns
Memphis Grizzlies
Philadelphia 76ers
Milwaukee Bucks
New Orleans Pelicans
Minnesota Timberwolves
Toronto Raptors
Utah Jazz
Denver Nuggets
Atlanta Hawks
Chicago Bulls
Brooklyn Nets
Teams (Seeded left to right by rank)

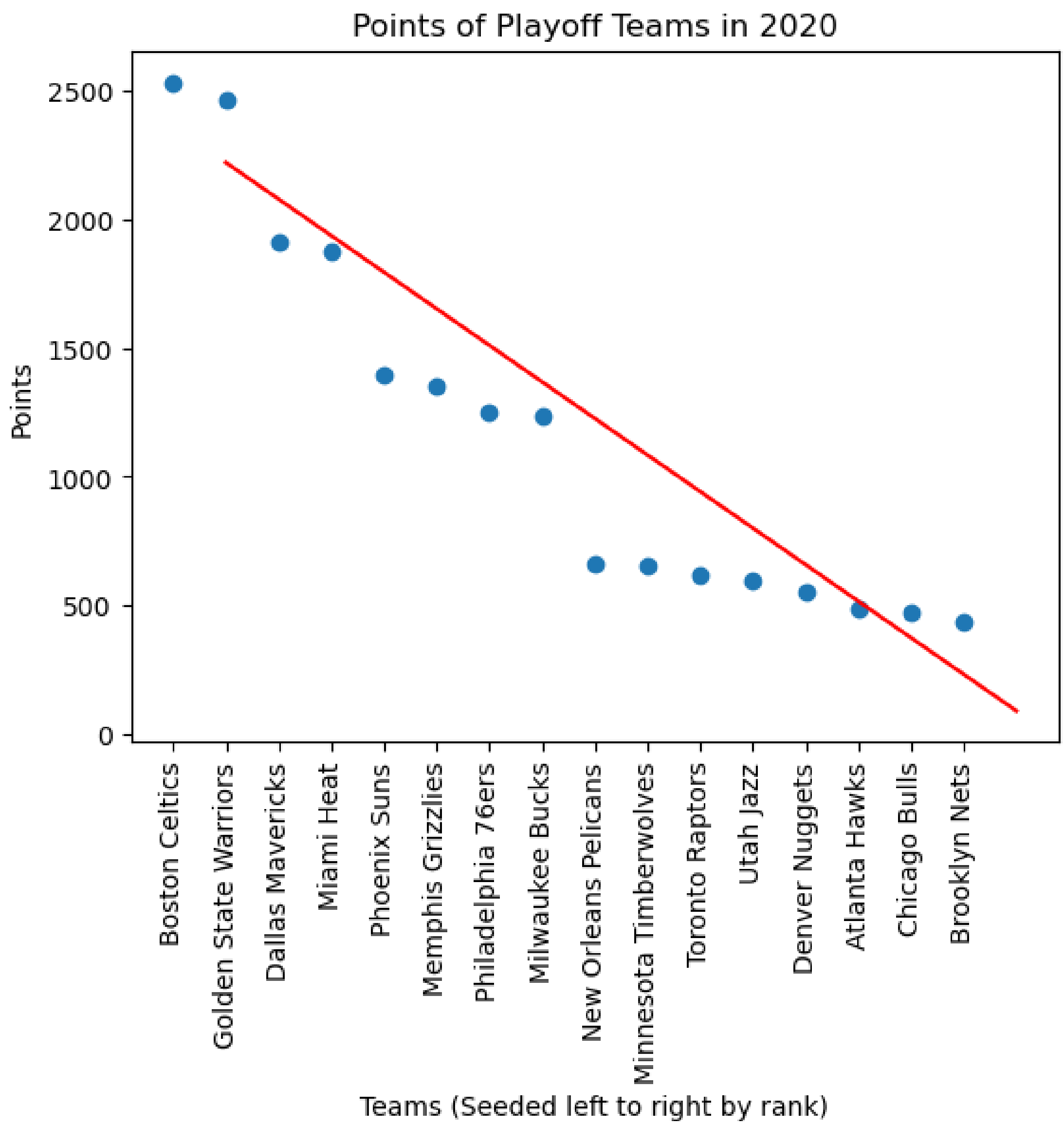

Turnovers of Playoff Teams in 2021
Turnovers
0
50
100
150
200
250
300
Los Angeles Lakers
Miami Heat
Denver Nuggets
Boston Celtics
Los Angeles Clippers
Houston Rockets
Toronto Raptors
Milwaukee Bucks
Utah Jazz
Oklahoma City Thunder
Dallas Mavericks
Orlando Magic
Portland Trail Blazers
Brooklyn Nets
Indiana Pacers
Philadelphia 76ers
Teams (Seeded left to right by rank)

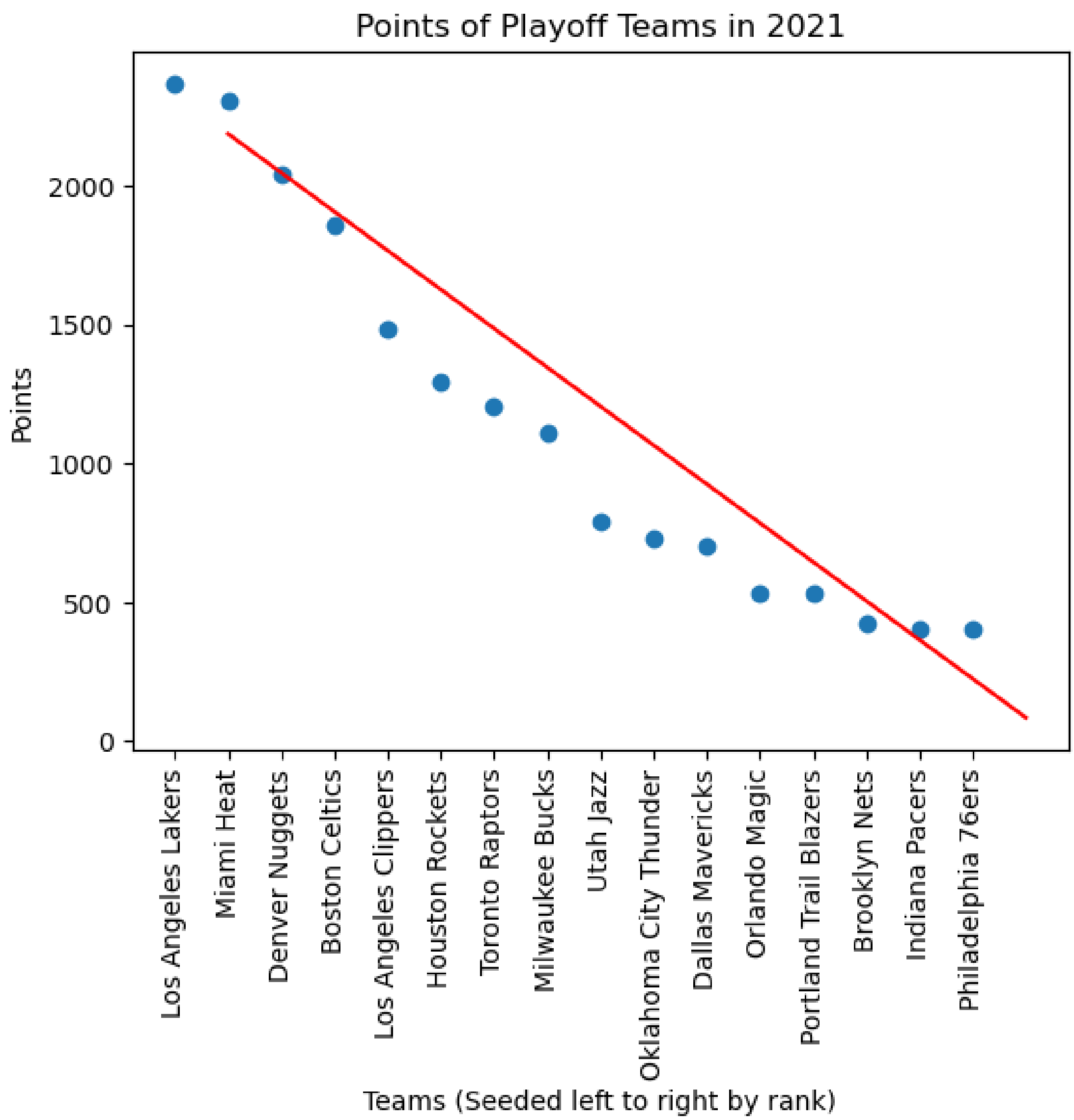


1.2.5 Regression: Points (PTS)

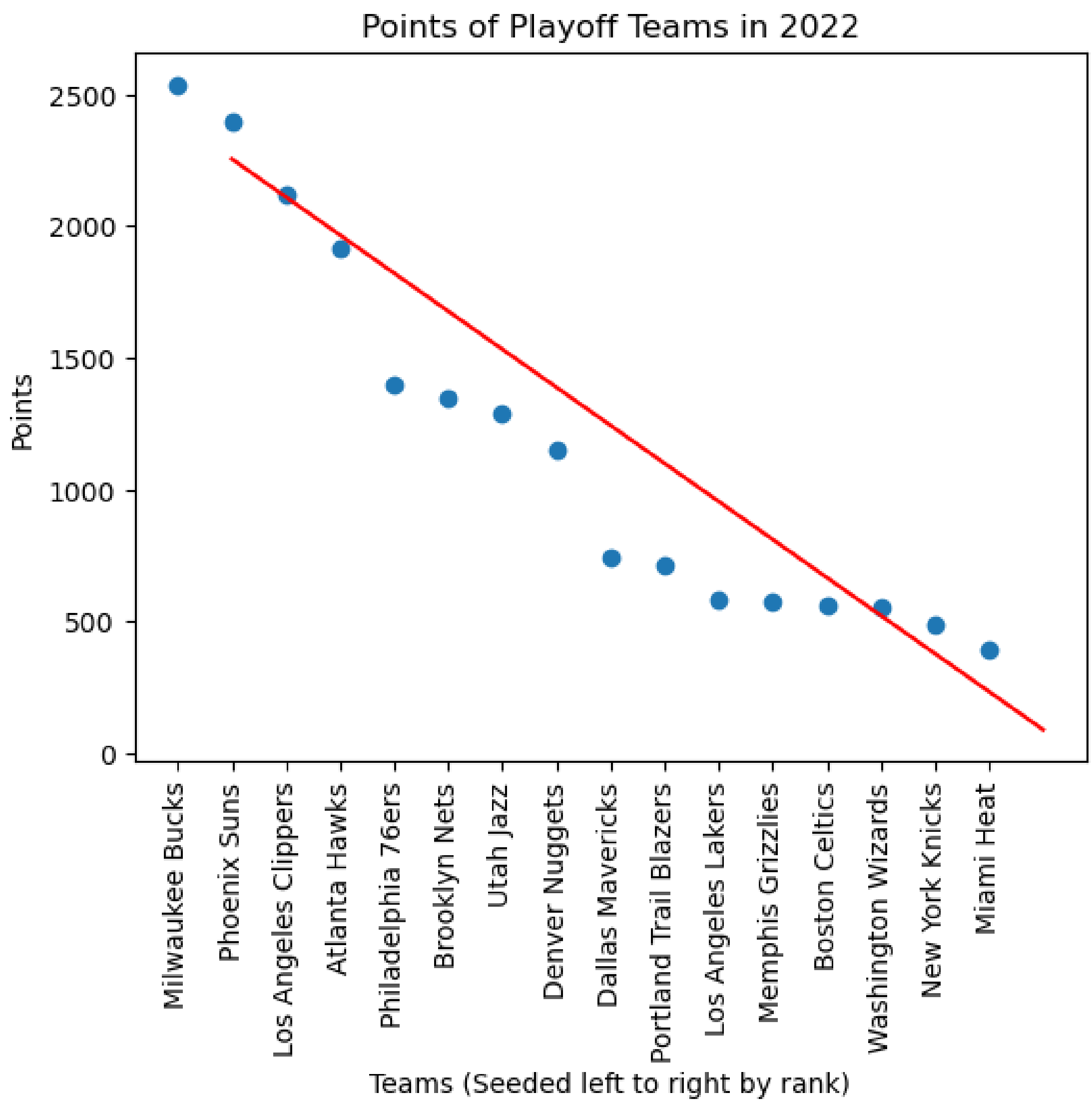
[14]:
```
year = 2020

for i in playoff_list_2:
    #gathering all relevant data columns for the playoff year
    teams = i["Team"]
    three_point_pct = i["3P%"]
    offensive_rebounds = i["ORB"]
    field_goal_pct = i["FG%"]
    turnovers = i["TOV"]
    points = i["PTS"]
    personal_fouls = i["PF"]
```

```
    #Total Points plots ///////////////////////////////////////////////////////////
↪////////////////
    # Plot the scatter plot
    plt.scatter(teams, points)

    # Add labels, title, and legend
    plt.xlabel("Teams (Seeded left to right by rank)")
    plt.ylabel("Points")
    plt.title("Points of Playoff Teams in " + str(year))
    plt.xticks(teams, rotation=90)

    #make regression plot
    #need to use rank because team names are not numerical for regression␣
↪purposes, achieves same goal since teams are in order by rank
    a, b = np.polyfit(i["Rk"], points, 1)
    plt.plot(i["Rk"], a * (i["Rk"]) + b, color='red')

    # Display the graph
    plt.show()

    year += 1
```


Points of Playoff Teams in 2020
Points
2500
2000
1500
1000
500
0
Boston Celtics
Golden State Warriors
Dallas Mavericks
Miami Heat
Phoenix Suns
Memphis Grizzlies
Philadelphia 76ers
Milwaukee Bucks
New Orleans Pelicans
Minnesota Timberwolves
Toronto Raptors
Utah Jazz
Denver Nuggets
Atlanta Hawks
Chicago Bulls
Brooklyn Nets
Teams (Seeded left to right by rank)



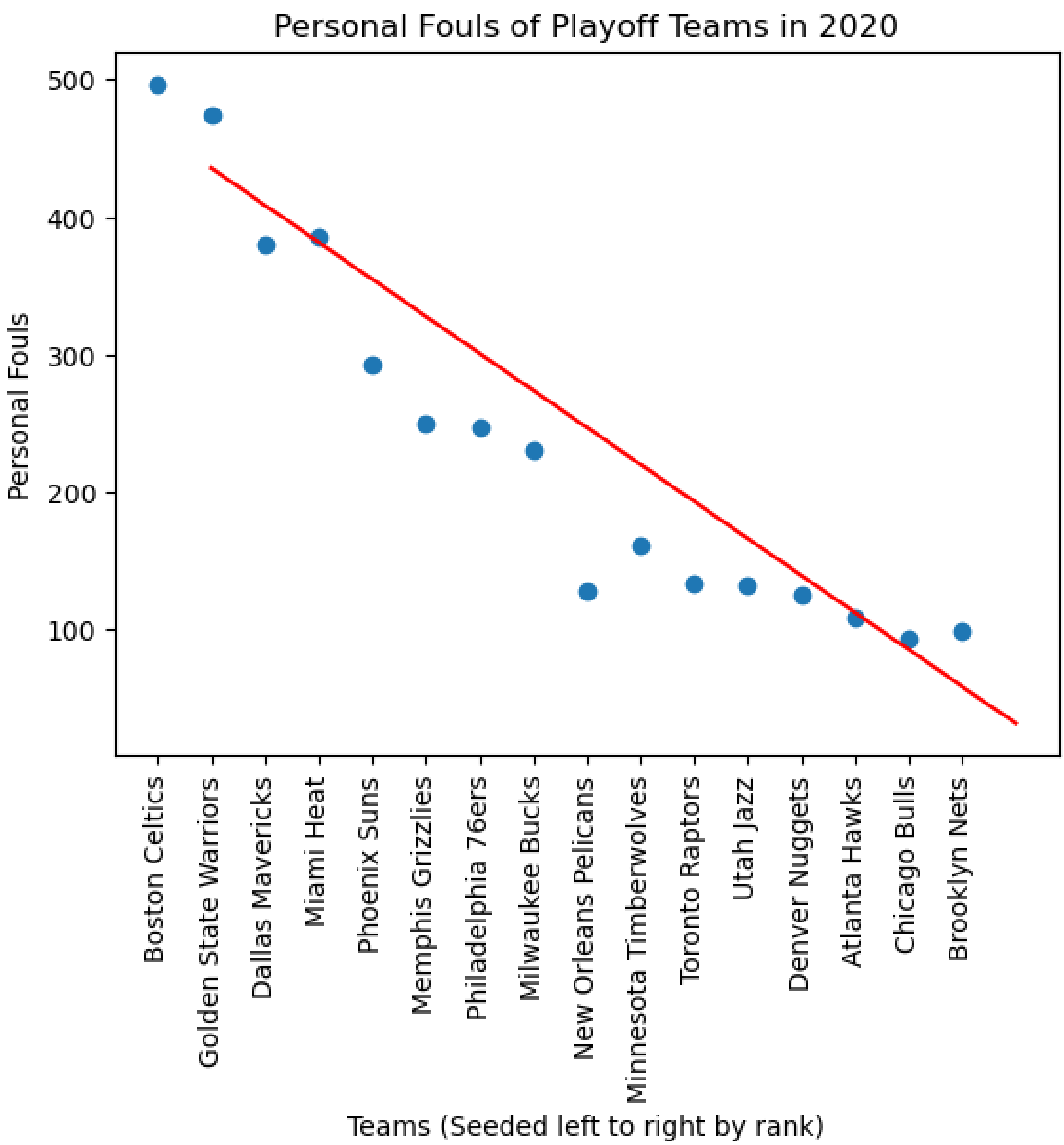
Points of Playoff Teams in 2021
Points
2000
1500
1000
500
0
Los Angeles Lakers
Miami Heat
Denver Nuggets
Boston Celtics
Los Angeles Clippers
Houston Rockets
Toronto Raptors
Milwaukee Bucks
Utah Jazz
Oklahoma City Thunder
Dallas Mavericks
Orlando Magic
Portland Trail Blazers
Brooklyn Nets
Indiana Pacers
Philadelphia 76ers
Teams (Seeded left to right by rank)



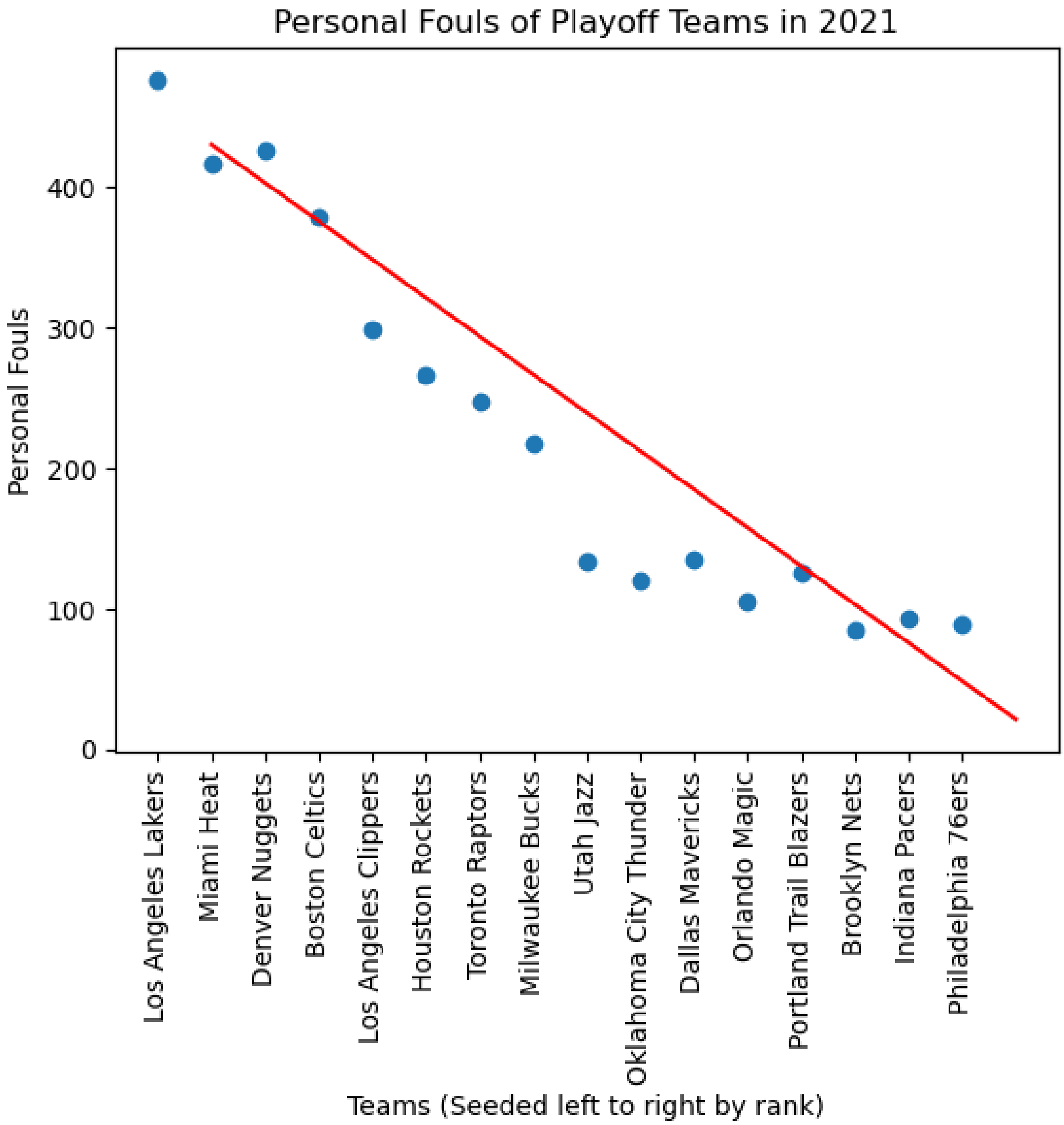
1.2.6 Regression: Personal Fouls (PF)

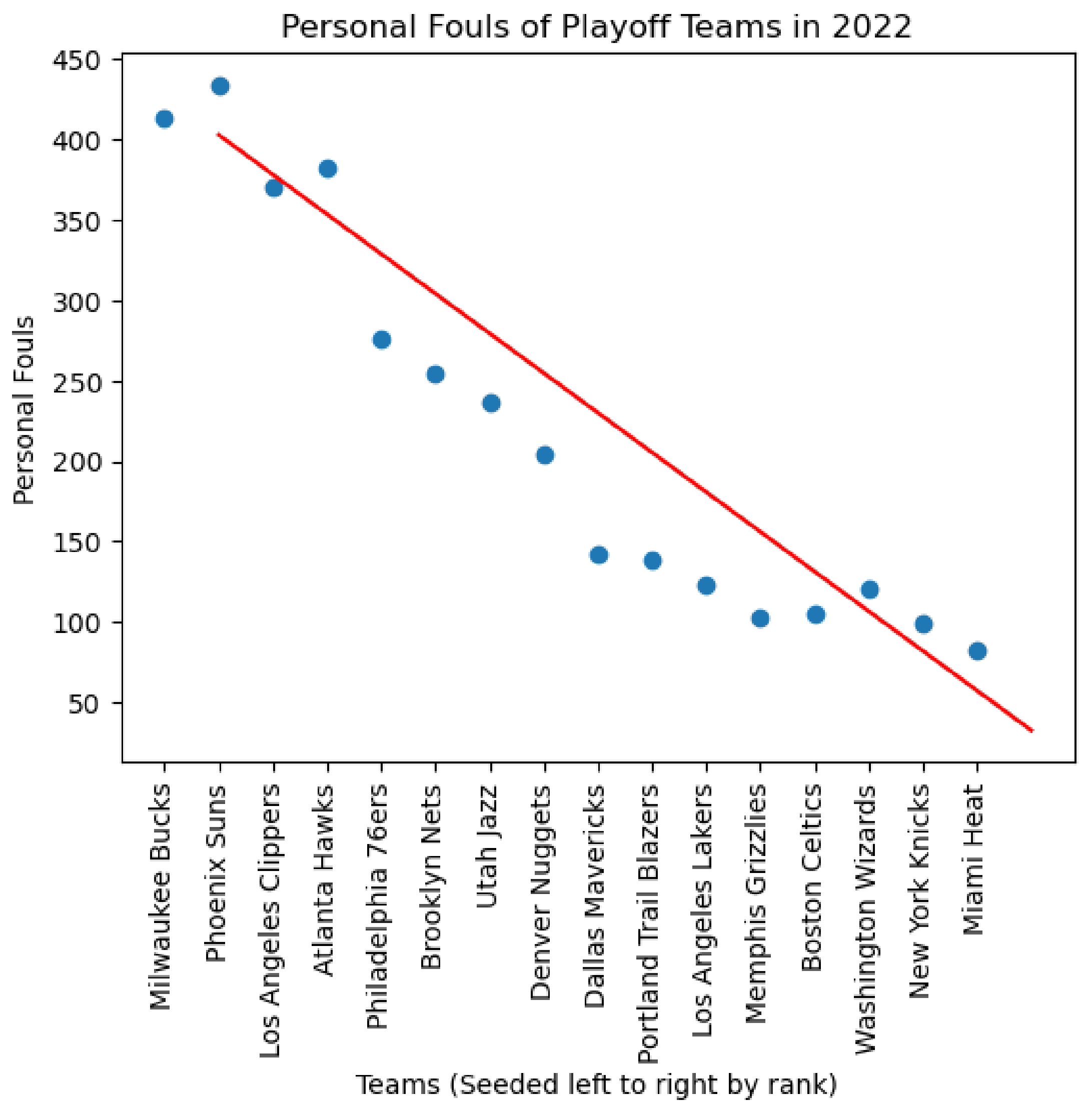
[15]:
```
year = 2020

for i in playoff_list_2:
    #gathering all relevant data columns for the playoff year
    teams = i["Team"]
    three_point_pct = i["3P%"]
    offensive_rebounds = i["ORB"]
    field_goal_pct = i["FG%"]
    turnovers = i["TOV"]
    points = i["PTS"]
    personal_fouls = i["PF"]
```

```
    #Personal Fouls plots ///////////////////////////////////////////////////////
↪/////////////////
    # Plot the scatter plot
    plt.scatter(teams, personal_fouls)

    # Add labels, title, and legend
    plt.xlabel("Teams (Seeded left to right by rank)")
    plt.ylabel("Personal Fouls")
    plt.title("Personal Fouls of Playoff Teams in " + str(year))
    plt.xticks(teams, rotation=90)

    #make regression plot
    #need to use rank because team names are not numerical for regression␣
↪purposes, achieves same goal since teams are in order by rank
    a, b = np.polyfit(i["Rk"], personal_fouls, 1)
    plt.plot(i["Rk"], a * (i["Rk"]) + b, color='red')

    # Display the graph
    plt.show()

    year += 1
```



Personal Fouls of Playoff Teams in 2020
500
400
300
200
100
Personal Fouls
Boston Celtics
Golden State Warriors
Dallas Mavericks
Miami Heat
Phoenix Suns
Memphis Grizzlies
Philadelphia 76ers
Milwaukee Bucks
New Orleans Pelicans
Minnesota Timberwolves
Toronto Raptors
Utah Jazz
Denver Nuggets
Atlanta Hawks
Chicago Bulls
Brooklyn Nets
Teams (Seeded left to right by rank)


Personal Fouls of Playoff Teams in 2021
Personal Fouls
0
100
200
300
400
Los Angeles Lakers
Miami Heat
Denver Nuggets
Boston Celtics
Los Angeles Clippers
Houston Rockets
Toronto Raptors
Milwaukee Bucks
Utah Jazz
Oklahoma City Thunder
Dallas Mavericks
Orlando Magic
Portland Trail Blazers
Brooklyn Nets
Indiana Pacers
Philadelphia 76ers
Teams (Seeded left to right by rank)



Confirm the comments we made about turnovers.

Q: Now looking at the regression lines, what can we verify about the comments we made in the previous part?

`A: The trend still stands, all of the comments made about the depricating nature of the plot a`

Q: How do you think we can we determine the most valuable statistic (guess)?

`A: We can look closer at the specific values of the linear regression and see the most valuabl`

Let's take a closer look at the linear regression values for each statistic

1.2.7 Values of Regression: 2020 Playoffs

In order to determine the most valubale statistic, lets look at the statistic with the minimum p-value for each year.

Write them all down for reference.

[16]:
```python
import statsmodels.api as s
from sklearn.linear_model import LinearRegression as lr

#we should include all stats for each year in a cell, will be easier to look at␣
↪in determining the most meaningful statistic per year

#Three point Percentage
print("3P% Regression Results for 2020")
x = data_2020_n["Rk"]
y = data_2020_n["3P%"]
r = s.OLS(y,x).fit()
print(r.summary())
print("\n\n")

#Offensive Rebound
print("Offensive Rebound Regression Results for 2020")
x = data_2020_n["Rk"]
y = data_2020_n["ORB"]
r = s.OLS(y,x).fit()
print(r.summary())
print("\n\n")

#Field Goal Percentage
print("Field Goal Percentage Regression Results for 2020")
x = data_2020_n["Rk"]
y = data_2020_n["FG%"]
r = s.OLS(y,x).fit()
print(r.summary())
print("\n\n")

#Turnovers
print("Turnover Results for 2020")
x = data_2020_n["Rk"]
y = data_2020_n["TOV"]
r = s.OLS(y,x).fit()
print(r.summary())
print("\n\n")

#Points
print("Total Points Results for 2020")
x = data_2020_n["Rk"]
```

```python
y = data_2020_n["PTS"]
r = s.OLS(y,x).fit()
print(r.summary())
print("\n\n")

#Personal Fouls
print("Personal Fouls Results for 2020")
x = data_2020_n["Rk"]
y = data_2020_n["PF"]
r = s.OLS(y,x).fit()
print(r.summary())
print("\n\n")
```

```
3P% Regression Results for 2020
                                 OLS Regression Results
================================================================================
=======
Dep. Variable:                    3P%   R-squared (uncentered):
0.738
Model:                            OLS   Adj. R-squared (uncentered):
0.721
Method:                 Least Squares   F-statistic:
42.32
Date:                Fri, 12 May 2023   Prob (F-statistic):
9.94e-06
Time:                        02:36:45   Log-Likelihood:
4.7968
No. Observations:                  16   AIC:
-7.594
Df Residuals:                      15   BIC:
-6.821
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Rk             0.0311      0.005      6.505      0.000       0.021       0.041
==============================================================================
Omnibus:                        1.982   Durbin-Watson:                   0.117
Prob(Omnibus):                  0.371   Jarque-Bera (JB):                0.959
Skew:                           0.014   Prob(JB):                        0.619
Kurtosis:                       1.801   Cond. No.                         1.00
==============================================================================

Notes:
[1] R² is computed without centering (uncentered) since the model does not
contain a constant.
```

[2] Standard Errors assume that the covariance matrix of the errors is correctly specified.

```
Offensive Rebound Regression Results for 2020
                                OLS Regression Results
================================================================================
=======
Dep. Variable:                    ORB   R-squared (uncentered):
0.292
Model:                            OLS   Adj. R-squared (uncentered):
0.245
Method:                 Least Squares   F-statistic:
6.185
Date:                Fri, 12 May 2023   Prob (F-statistic):
0.0251
Time:                        02:36:45   Log-Likelihood:
-96.259
No. Observations:                  16   AIC:
194.5
Df Residuals:                      15   BIC:
195.3
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Rk             6.5882      2.649      2.487      0.025       0.942      12.235
==============================================================================
Omnibus:                        1.584   Durbin-Watson:                   0.175
Prob(Omnibus):                  0.453   Jarque-Bera (JB):                1.184
Skew:                           0.458   Prob(JB):                        0.553
Kurtosis:                       2.032   Cond. No.                         1.00
==============================================================================
```

Notes:
[1] R^2 is computed without centering (uncentered) since the model does not contain a constant.
[2] Standard Errors assume that the covariance matrix of the errors is correctly specified.

```
Field Goal Percentage Regression Results for 2020
                                OLS Regression Results
================================================================================
=======
```

```
Dep. Variable:                    FG%   R-squared (uncentered):
0.766
Model:                            OLS   Adj. R-squared (uncentered):
0.750
Method:                 Least Squares   F-statistic:
49.11
Date:                Fri, 12 May 2023   Prob (F-statistic):
4.22e-06
Time:                        02:36:45   Log-Likelihood:
1.5054
No. Observations:                  16   AIC:
-1.011
Df Residuals:                      15   BIC:
-0.2383
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Rk             0.0412      0.006      7.008      0.000       0.029       0.054
==============================================================================
Omnibus:                        2.140   Durbin-Watson:                   0.059
Prob(Omnibus):                  0.343   Jarque-Bera (JB):                0.992
Skew:                           0.023   Prob(JB):                        0.609
Kurtosis:                       1.781   Cond. No.                         1.00
==============================================================================

Notes:
[1] R² is computed without centering (uncentered) since the model does not
contain a constant.
[2] Standard Errors assume that the covariance matrix of the errors is correctly
specified.



Turnover Results for 2020
                                 OLS Regression Results
================================================================================
=======
Dep. Variable:                    TOV   R-squared (uncentered):
0.290
Model:                            OLS   Adj. R-squared (uncentered):
0.242
Method:                 Least Squares   F-statistic:
6.112
Date:                Fri, 12 May 2023   Prob (F-statistic):
0.0259
Time:                        02:36:45   Log-Likelihood:
```

```
-102.45
No. Observations:                  16   AIC:
206.9
Df Residuals:                      15   BIC:
207.7
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Rk             9.6451      3.901      2.472      0.026       1.330      17.960
==============================================================================
Omnibus:                        1.784   Durbin-Watson:                   0.123
Prob(Omnibus):                  0.410   Jarque-Bera (JB):                1.394
Skew:                           0.662   Prob(JB):                        0.498
Kurtosis:                       2.420   Cond. No.                         1.00
==============================================================================

Notes:
[1] R² is computed without centering (uncentered) since the model does not
contain a constant.
[2] Standard Errors assume that the covariance matrix of the errors is correctly
specified.



Total Points Results for 2020
                                 OLS Regression Results
================================================================================
=======
Dep. Variable:                    PTS   R-squared (uncentered):
0.273
Model:                            OLS   Adj. R-squared (uncentered):
0.224
Method:                 Least Squares   F-statistic:
5.626
Date:                Fri, 12 May 2023   Prob (F-statistic):
0.0315
Time:                        02:36:45   Log-Likelihood:
-135.46
No. Observations:                  16   AIC:
272.9
Df Residuals:                      15   BIC:
273.7
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
```

```
------------------------------------------------------------------------------
Rk            72.8209     30.701      2.372      0.032       7.384     138.258
==============================================================================
Omnibus:                        1.967   Durbin-Watson:                   0.060
Prob(Omnibus):                  0.374   Jarque-Bera (JB):                1.392
Skew:                           0.519   Prob(JB):                        0.499
Kurtosis:                       1.996   Cond. No.                         1.00
==============================================================================

Notes:
[1] R² is computed without centering (uncentered) since the model does not
contain a constant.
[2] Standard Errors assume that the covariance matrix of the errors is correctly
specified.


Personal Fouls Results for 2020
                                 OLS Regression Results
================================================================================
=======
Dep. Variable:                     PF   R-squared (uncentered):
0.296
Model:                            OLS   Adj. R-squared (uncentered):
0.249
Method:                 Least Squares   F-statistic:
6.315
Date:                Fri, 12 May 2023   Prob (F-statistic):
0.0239
Time:                        02:36:45   Log-Likelihood:
-109.35
No. Observations:                  16   AIC:
220.7
Df Residuals:                      15   BIC:
221.5
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Rk            15.0856      6.003      2.513      0.024       2.290      27.881
==============================================================================
Omnibus:                        1.990   Durbin-Watson:                   0.061
Prob(Omnibus):                  0.370   Jarque-Bera (JB):                1.396
Skew:                           0.517   Prob(JB):                        0.498
Kurtosis:                       1.989   Cond. No.                         1.00
==============================================================================
```

```
Notes:
[1] R² is computed without centering (uncentered) since the model does not
contain a constant.
[2] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
```

```
/opt/conda/lib/python3.10/site-packages/scipy/stats/_stats_py.py:1736:
UserWarning: kurtosistest only valid for n>=20 … continuing anyway, n=16
  warnings.warn("kurtosistest only valid for n>=20 … continuing "
/opt/conda/lib/python3.10/site-packages/scipy/stats/_stats_py.py:1736:
UserWarning: kurtosistest only valid for n>=20 … continuing anyway, n=16
  warnings.warn("kurtosistest only valid for n>=20 … continuing "
/opt/conda/lib/python3.10/site-packages/scipy/stats/_stats_py.py:1736:
UserWarning: kurtosistest only valid for n>=20 … continuing anyway, n=16
  warnings.warn("kurtosistest only valid for n>=20 … continuing "
/opt/conda/lib/python3.10/site-packages/scipy/stats/_stats_py.py:1736:
UserWarning: kurtosistest only valid for n>=20 … continuing anyway, n=16
  warnings.warn("kurtosistest only valid for n>=20 … continuing "
/opt/conda/lib/python3.10/site-packages/scipy/stats/_stats_py.py:1736:
UserWarning: kurtosistest only valid for n>=20 … continuing anyway, n=16
  warnings.warn("kurtosistest only valid for n>=20 … continuing "
/opt/conda/lib/python3.10/site-packages/scipy/stats/_stats_py.py:1736:
UserWarning: kurtosistest only valid for n>=20 … continuing anyway, n=16
  warnings.warn("kurtosistest only valid for n>=20 … continuing "
```

This yields a min p-value for 2020 of 4.22e-06 for the statistic: Field Goal Percentage

1.2.8 Values of Regression: 2021 Playoffs

[17]:
```python
#we should include all stats for each year in a cell, will be easier to look at␣
↪in determining the most meaningful statistic per year

#Three point Percentage
print("3P% Regression Results for 2021")
x = data_2021_n["Rk"]
y = data_2021_n["3P%"]
r = s.OLS(y,x).fit()
print(r.summary())
print("\n\n")

#Offensive Rebound
print("Offensive Rebound Regression Results for 2021")
x = data_2021_n["Rk"]
y = data_2021_n["ORB"]
r = s.OLS(y,x).fit()
```

```
print(r.summary())
print("\n\n")

#Field Goal Percentage
print("Field Goal Percentage Regression Results for 2021")
x = data_2021_n["Rk"]
y = data_2021_n["FG%"]
r = s.OLS(y,x).fit()
print(r.summary())
print("\n\n")

#Turnovers
print("Turnover Results for 2021")
x = data_2021_n["Rk"]
y = data_2021_n["TOV"]
r = s.OLS(y,x).fit()
print(r.summary())
print("\n\n")

#Points
print("Total Points Results for 2021")
x = data_2021_n["Rk"]
y = data_2021_n["PTS"]
r = s.OLS(y,x).fit()
print(r.summary())
print("\n\n")

#Personal Fouls
print("Personal Fouls Results for 2021")
x = data_2021_n["Rk"]
y = data_2021_n["PF"]
r = s.OLS(y,x).fit()
print(r.summary())
print("\n\n")
```

```
3P% Regression Results for 2021
                                 OLS Regression Results
================================================================================
=======
Dep. Variable:                    3P%   R-squared (uncentered):
0.747
Model:                            OLS   Adj. R-squared (uncentered):
0.730
Method:                 Least Squares   F-statistic:
44.31
Date:                Fri, 12 May 2023   Prob (F-statistic):
7.65e-06
```

```
Time:                        02:36:45   Log-Likelihood:
4.6765
No. Observations:                  16   AIC:
-7.353
Df Residuals:                      15   BIC:
-6.580
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Rk             0.0321      0.005      6.657      0.000       0.022       0.042
==============================================================================
Omnibus:                        0.646   Durbin-Watson:                   0.102
Prob(Omnibus):                  0.724   Jarque-Bera (JB):                0.678
Skew:                          -0.320   Prob(JB):                        0.712
Kurtosis:                       2.220   Cond. No.                         1.00
==============================================================================

Notes:
[1] R² is computed without centering (uncentered) since the model does not
contain a constant.
[2] Standard Errors assume that the covariance matrix of the errors is correctly
specified.


Offensive Rebound Regression Results for 2021
                                 OLS Regression Results
================================================================================
=======
Dep. Variable:                    ORB   R-squared (uncentered):
0.263
Model:                            OLS   Adj. R-squared (uncentered):
0.214
Method:                 Least Squares   F-statistic:
5.361
Date:                Fri, 12 May 2023   Prob (F-statistic):
0.0352
Time:                        02:36:45   Log-Likelihood:
-95.128
No. Observations:                  16   AIC:
192.3
Df Residuals:                      15   BIC:
193.0
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
```

```
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Rk             5.7152      2.468      2.315      0.035       0.454      10.977
==============================================================================
Omnibus:                        1.713   Durbin-Watson:                   0.088
Prob(Omnibus):                  0.425   Jarque-Bera (JB):                1.383
Skew:                           0.600   Prob(JB):                        0.501
Kurtosis:                       2.202   Cond. No.                         1.00
==============================================================================

Notes:
[1] R² is computed without centering (uncentered) since the model does not
contain a constant.
[2] Standard Errors assume that the covariance matrix of the errors is correctly
specified.



Field Goal Percentage Regression Results for 2021
                                 OLS Regression Results
================================================================================
=======
Dep. Variable:                    FG%   R-squared (uncentered):
0.736
Model:                            OLS   Adj. R-squared (uncentered):
0.718
Method:                 Least Squares   F-statistic:
41.75
Date:                Fri, 12 May 2023   Prob (F-statistic):
1.07e-05
Time:                        02:36:45   Log-Likelihood:
0.75329
No. Observations:                  16   AIC:
0.4934
Df Residuals:                      15   BIC:
1.266
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Rk             0.0398      0.006      6.461      0.000       0.027       0.053
==============================================================================
Omnibus:                        1.488   Durbin-Watson:                   0.060
Prob(Omnibus):                  0.475   Jarque-Bera (JB):                0.854
Skew:                          -0.058   Prob(JB):                        0.652
Kurtosis:                       1.874   Cond. No.                         1.00
==============================================================================
```

Notes:
[1] R^2 is computed without centering (uncentered) since the model does not contain a constant.
[2] Standard Errors assume that the covariance matrix of the errors is correctly specified.

Turnover Results for 2021

```
                                 OLS Regression Results
================================================================================
=======
Dep. Variable:                    TOV   R-squared (uncentered):
0.282
Model:                            OLS   Adj. R-squared (uncentered):
0.234
Method:                 Least Squares   F-statistic:
5.891
Date:                Fri, 12 May 2023   Prob (F-statistic):
0.0283
Time:                        02:36:45   Log-Likelihood:
-102.06
No. Observations:                  16   AIC:
206.1
Df Residuals:                      15   BIC:
206.9
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Rk             9.2406      3.807      2.427      0.028       1.126      17.355
==============================================================================
Omnibus:                        1.769   Durbin-Watson:                   0.069
Prob(Omnibus):                  0.413   Jarque-Bera (JB):                1.297
Skew:                           0.498   Prob(JB):                        0.523
Kurtosis:                       2.024   Cond. No.                         1.00
==============================================================================
```

Notes:
[1] R^2 is computed without centering (uncentered) since the model does not contain a constant.
[2] Standard Errors assume that the covariance matrix of the errors is correctly specified.

Total Points Results for 2021

```
                                 OLS Regression Results
================================================================================
=======
Dep. Variable:                    PTS   R-squared (uncentered):
0.274
Model:                            OLS   Adj. R-squared (uncentered):
0.226
Method:                 Least Squares   F-statistic:
5.673
Date:                Fri, 12 May 2023   Prob (F-statistic):
0.0309
Time:                        02:36:45   Log-Likelihood:
-135.09
No. Observations:                  16   AIC:
272.2
Df Residuals:                      15   BIC:
273.0
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Rk            71.4672     30.005      2.382      0.031       7.513     135.421
==============================================================================
Omnibus:                        2.276   Durbin-Watson:                   0.040
Prob(Omnibus):                  0.320   Jarque-Bera (JB):                1.330
Skew:                           0.420   Prob(JB):                        0.514
Kurtosis:                       1.864   Cond. No.                         1.00
==============================================================================

Notes:
[1] R² is computed without centering (uncentered) since the model does not
contain a constant.
[2] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
```

Personal Fouls Results for 2021

```
                                 OLS Regression Results
================================================================================
=======
Dep. Variable:                     PF   R-squared (uncentered):
0.281
Model:                            OLS   Adj. R-squared (uncentered):
0.233
Method:                 Least Squares   F-statistic:
```

```
5.852
Date:                Fri, 12 May 2023   Prob (F-statistic):
0.0287
Time:                        02:36:45   Log-Likelihood:
-109.13
No. Observations:                  16   AIC:
220.3
Df Residuals:                      15   BIC:
221.0
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Rk            14.3255      5.922      2.419      0.029       1.703      26.948
==============================================================================
Omnibus:                        2.853   Durbin-Watson:                   0.050
Prob(Omnibus):                  0.240   Jarque-Bera (JB):                1.475
Skew:                           0.432   Prob(JB):                        0.478
Kurtosis:                       1.789   Cond. No.                         1.00
==============================================================================

Notes:
[1] R² is computed without centering (uncentered) since the model does not
contain a constant.
[2] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
```

```
/opt/conda/lib/python3.10/site-packages/scipy/stats/_stats_py.py:1736:
UserWarning: kurtosistest only valid for n>=20 … continuing anyway, n=16
  warnings.warn("kurtosistest only valid for n>=20 … continuing "
/opt/conda/lib/python3.10/site-packages/scipy/stats/_stats_py.py:1736:
UserWarning: kurtosistest only valid for n>=20 … continuing anyway, n=16
  warnings.warn("kurtosistest only valid for n>=20 … continuing "
/opt/conda/lib/python3.10/site-packages/scipy/stats/_stats_py.py:1736:
UserWarning: kurtosistest only valid for n>=20 … continuing anyway, n=16
  warnings.warn("kurtosistest only valid for n>=20 … continuing "
/opt/conda/lib/python3.10/site-packages/scipy/stats/_stats_py.py:1736:
UserWarning: kurtosistest only valid for n>=20 … continuing anyway, n=16
  warnings.warn("kurtosistest only valid for n>=20 … continuing "
/opt/conda/lib/python3.10/site-packages/scipy/stats/_stats_py.py:1736:
UserWarning: kurtosistest only valid for n>=20 … continuing anyway, n=16
  warnings.warn("kurtosistest only valid for n>=20 … continuing "
/opt/conda/lib/python3.10/site-packages/scipy/stats/_stats_py.py:1736:
UserWarning: kurtosistest only valid for n>=20 … continuing anyway, n=16
```

```
warnings.warn("kurtosistest only valid for n>=20 … continuing "
```

This yields a min p-value for 2021 of 7.65e-06 for the statistic: Three Point Percentage

1.2.9 Values of Regression: 2022 Playoffs

[20]:
```
#we should include all stats for each year in a cell, will be easier to look at␣
↪in determining the most meaningful statistic per year

#Three point Percentage
print("3P% Regression Results for 2022")
x = data_2022_n["Rk"]
y = data_2022_n["3P%"]
r = s.OLS(y,x).fit()
print(r.summary())
print("\n\n")

#Offensive Rebound
print("Offensive Rebound Regression Results for 2022")
x = data_2022_n["Rk"]
y = data_2022_n["ORB"]
r = s.OLS(y,x).fit()
print(r.summary())
print("\n\n")

#Field Goal Percentage
print("Field Goal Percentage Regression Results for 2022")
x = data_2022_n["Rk"]
y = data_2022_n["FG%"]
r = s.OLS(y,x).fit()
print(r.summary())
print("\n\n")

#Turnovers
print("Turnover Results for 2022")
x = data_2022_n["Rk"]
y = data_2022_n["TOV"]
r = s.OLS(y,x).fit()
print(r.summary())
print("\n\n")

#Points
print("Total Points Results for 2022")
x = data_2022_n["Rk"]
y = data_2022_n["PTS"]
r = s.OLS(y,x).fit()
print(r.summary())
print("\n\n")
```

```python
#Personal Fouls
print("Personal Fouls Results for 2022")
x = data_2022_n["Rk"]
y = data_2022_n["PF"]
r = s.OLS(y,x).fit()
print(r.summary())
print("\n\n")
```

```
3P% Regression Results for 2022
                                 OLS Regression Results
================================================================================
=======
Dep. Variable:                    3P%   R-squared (uncentered):
0.733
Model:                            OLS   Adj. R-squared (uncentered):
0.716
Method:                 Least Squares   F-statistic:
41.24
Date:                Fri, 12 May 2023   Prob (F-statistic):
1.15e-05
Time:                        02:37:57   Log-Likelihood:
4.1962
No. Observations:                  16   AIC:
-6.392
Df Residuals:                      15   BIC:
-5.620
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Rk             0.0319      0.005      6.422      0.000       0.021       0.043
==============================================================================
Omnibus:                        4.201   Durbin-Watson:                   0.081
Prob(Omnibus):                  0.122   Jarque-Bera (JB):                1.459
Skew:                          -0.256   Prob(JB):                        0.482
Kurtosis:                       1.612   Cond. No.                         1.00
==============================================================================

Notes:
[1] R² is computed without centering (uncentered) since the model does not
contain a constant.
[2] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
```

Offensive Rebound Regression Results for 2022

```
                                 OLS Regression Results                                
=======================================================================================
Dep. Variable:                    ORB   R-squared (uncentered):                   0.283
Model:                            OLS   Adj. R-squared (uncentered):              0.235
Method:                 Least Squares   F-statistic:                              5.920
Date:                Fri, 12 May 2023   Prob (F-statistic):                      0.0280
Time:                        02:37:57   Log-Likelihood:                         -97.139
No. Observations:                  16   AIC:                                      196.3
Df Residuals:                      15   BIC:                                      197.1
Df Model:                           1                                                  
Covariance Type:            nonrobust                                                  
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Rk             6.8102      2.799      2.433      0.028       0.844      12.776
==============================================================================
Omnibus:                        4.669   Durbin-Watson:                   0.143
Prob(Omnibus):                  0.097   Jarque-Bera (JB):                2.600
Skew:                           0.970   Prob(JB):                        0.273
Kurtosis:                       3.371   Cond. No.                         1.00
==============================================================================

Notes:
[1] R² is computed without centering (uncentered) since the model does not contain a constant.
[2] Standard Errors assume that the covariance matrix of the errors is correctly specified.
```

Field Goal Percentage Regression Results for 2022

```
                                 OLS Regression Results                                
=======================================================================================
Dep. Variable:                    FG%   R-squared (uncentered):                   0.734
Model:                            OLS   Adj. R-squared (uncentered):              0.716
```

```
Method:                 Least Squares   F-statistic:
41.39
Date:                Fri, 12 May 2023   Prob (F-statistic):
1.13e-05
Time:                        02:37:57   Log-Likelihood:
0.49384
No. Observations:                  16   AIC:
1.012
Df Residuals:                      15   BIC:
1.785
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Rk             0.0403      0.006      6.433      0.000       0.027       0.054
==============================================================================
Omnibus:                        1.687   Durbin-Watson:                   0.052
Prob(Omnibus):                  0.430   Jarque-Bera (JB):                0.929
Skew:                          -0.143   Prob(JB):                        0.628
Kurtosis:                       1.854   Cond. No.                         1.00
==============================================================================

Notes:
[1] R² is computed without centering (uncentered) since the model does not
contain a constant.
[2] Standard Errors assume that the covariance matrix of the errors is correctly
specified.


Turnover Results for 2022
                                 OLS Regression Results
================================================================================
=======
Dep. Variable:                    TOV   R-squared (uncentered):
0.286
Model:                            OLS   Adj. R-squared (uncentered):
0.238
Method:                 Least Squares   F-statistic:
5.995
Date:                Fri, 12 May 2023   Prob (F-statistic):
0.0271
Time:                        02:37:57   Log-Likelihood:
-100.17
No. Observations:                  16   AIC:
202.3
Df Residuals:                      15   BIC:
```

```
203.1
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Rk             8.2834      3.383      2.448      0.027       1.072      15.495
==============================================================================
Omnibus:                        1.987   Durbin-Watson:                   0.074
Prob(Omnibus):                  0.370   Jarque-Bera (JB):                1.516
Skew:                           0.603   Prob(JB):                        0.469
Kurtosis:                       2.094   Cond. No.                         1.00
==============================================================================

Notes:
[1] R² is computed without centering (uncentered) since the model does not
contain a constant.
[2] Standard Errors assume that the covariance matrix of the errors is correctly
specified.



Total Points Results for 2022
                                 OLS Regression Results
================================================================================
=======
Dep. Variable:                    PTS   R-squared (uncentered):
0.273
Model:                            OLS   Adj. R-squared (uncentered):
0.224
Method:                 Least Squares   F-statistic:
5.628
Date:                Fri, 12 May 2023   Prob (F-statistic):
0.0315
Time:                        02:37:57   Log-Likelihood:
-135.66
No. Observations:                  16   AIC:
273.3
Df Residuals:                      15   BIC:
274.1
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Rk            73.7493     31.088      2.372      0.031       7.486     140.013
==============================================================================
Omnibus:                        2.148   Durbin-Watson:                   0.047
```

```
Prob(Omnibus):                  0.342   Jarque-Bera (JB):               1.411
Skew:                           0.497   Prob(JB):                       0.494
Kurtosis:                       1.937   Cond. No.                        1.00
==============================================================================

Notes:
[1] R² is computed without centering (uncentered) since the model does not
contain a constant.
[2] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
```

Personal Fouls Results for 2022

```
                                 OLS Regression Results
================================================================================
=======
Dep. Variable:                     PF   R-squared (uncentered):
0.305
Model:                            OLS   Adj. R-squared (uncentered):
0.258
Method:                 Least Squares   F-statistic:
6.571
Date:                Fri, 12 May 2023   Prob (F-statistic):
0.0216
Time:                        02:37:58   Log-Likelihood:
-108.04
No. Observations:                  16   AIC:
218.1
Df Residuals:                      15   BIC:
218.8
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Rk            14.1798      5.531      2.563      0.022       2.390      25.970
==============================================================================
Omnibus:                        3.028   Durbin-Watson:                  0.052
Prob(Omnibus):                  0.220   Jarque-Bera (JB):               1.471
Skew:                           0.407   Prob(JB):                       0.479
Kurtosis:                       1.758   Cond. No.                        1.00
==============================================================================

Notes:
[1] R² is computed without centering (uncentered) since the model does not
contain a constant.
[2] Standard Errors assume that the covariance matrix of the errors is correctly
```

```
specified.
```

```
/opt/conda/lib/python3.10/site-packages/scipy/stats/_stats_py.py:1736:
UserWarning: kurtosistest only valid for n>=20 … continuing anyway, n=16
  warnings.warn("kurtosistest only valid for n>=20 … continuing "
/opt/conda/lib/python3.10/site-packages/scipy/stats/_stats_py.py:1736:
UserWarning: kurtosistest only valid for n>=20 … continuing anyway, n=16
  warnings.warn("kurtosistest only valid for n>=20 … continuing "
/opt/conda/lib/python3.10/site-packages/scipy/stats/_stats_py.py:1736:
UserWarning: kurtosistest only valid for n>=20 … continuing anyway, n=16
  warnings.warn("kurtosistest only valid for n>=20 … continuing "
/opt/conda/lib/python3.10/site-packages/scipy/stats/_stats_py.py:1736:
UserWarning: kurtosistest only valid for n>=20 … continuing anyway, n=16
  warnings.warn("kurtosistest only valid for n>=20 … continuing "
/opt/conda/lib/python3.10/site-packages/scipy/stats/_stats_py.py:1736:
UserWarning: kurtosistest only valid for n>=20 … continuing anyway, n=16
  warnings.warn("kurtosistest only valid for n>=20 … continuing "
/opt/conda/lib/python3.10/site-packages/scipy/stats/_stats_py.py:1736:
UserWarning: kurtosistest only valid for n>=20 … continuing anyway, n=16
  warnings.warn("kurtosistest only valid for n>=20 … continuing "
```

This yields a min p-value for 2022 of 1.13e-05 for the statistic: Field Goal Percentage

but, has a close second of another min p-value for 2022 of 1.15e-05 for the statistic: Three Point Percentage

Q: What do you notice? Make clear observations about the data.

```
A: The 2022 year has two statistics with very close p-values that are the minumum for 2022. Thi
```

Q: What statistic seems to be the most valuable?

```
A: According to the data, Field Goal Percentage seems to be the most valuable statistic in what
```

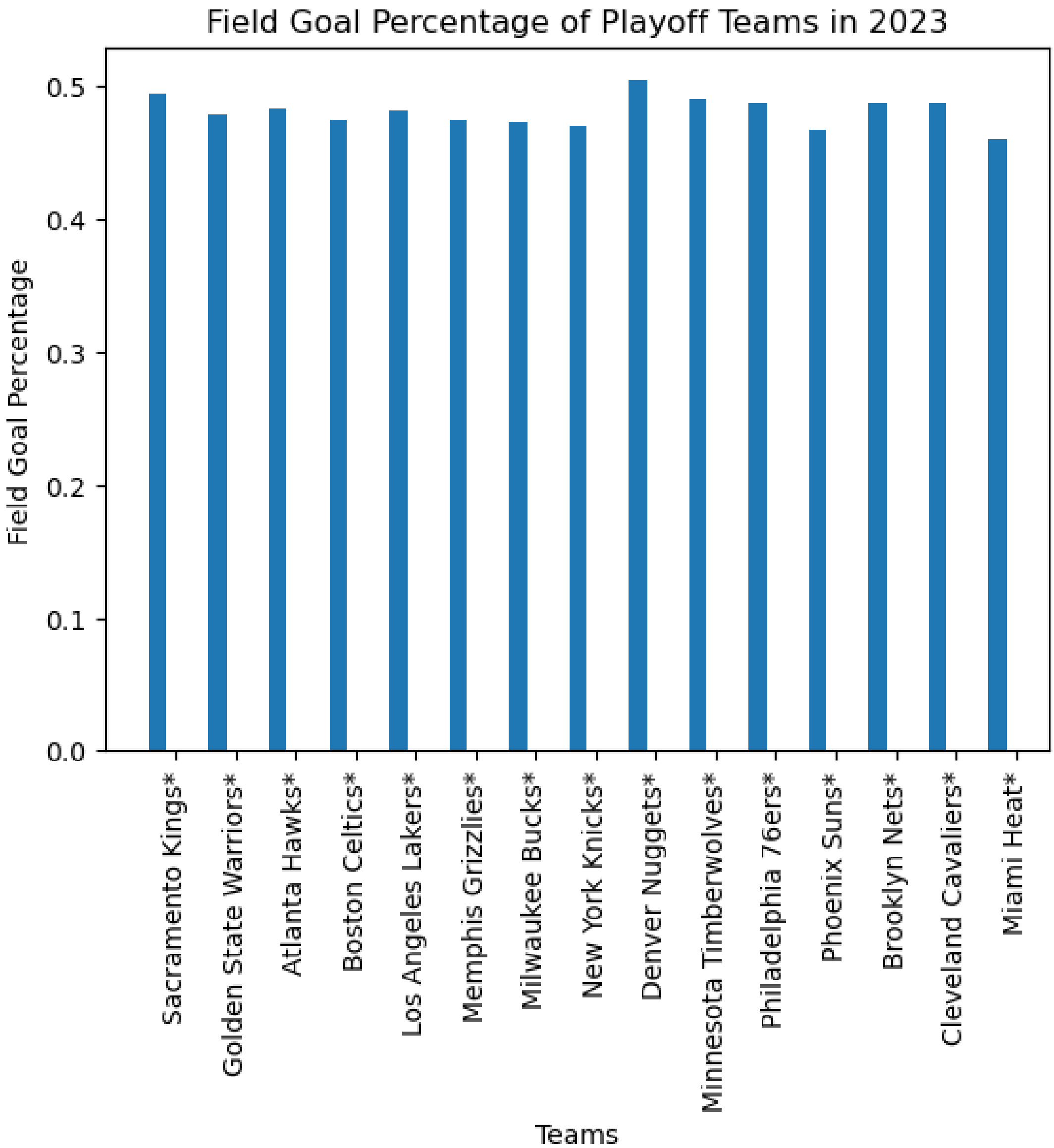
1.3 Part 3: Application to Current Season

Now, let's choose the statistic and apply them to the new data set for the current season.

Create a graph based on this statistic for the teams in the playoffs for 2023.

[19]:
```
data_2023 = pd.read_csv("2023_Playoff_Stats.csv")



#drop all NaN rows to avoid errors
year = 2023

for i in range(15, 31, 1):
    data_2023 = data_2023.drop(i)
```

```python
#gathering all relevant data columns for the playoff year
teams = data_2023["Team"]
field_goal_pct_new = data_2023["FG%"]

# Set the width of the bars
bar_width = 0.3
# Set the positions of the bars on the x-axis
bar_positions = range(len(teams))
#FG plots /////////////////////////////////////////////////////////////////////
 ↪//
# Plot the bars
plt.bar(bar_positions, field_goal_pct_new, width=bar_width, label=str(year))

# Add labels, title, and legend
plt.xlabel("Teams")
plt.ylabel("Field Goal Percentage")
plt.title("Field Goal Percentage of Playoff Teams in 2023")
plt.xticks([r + bar_width for r in range(len(teams))], teams, rotation=90)


# Display the graph
plt.show()
```



Q: Which team based on this statistic is most likely to win the NBA Championship in 2023?

`A: The Denver Nuggets`

Q: How well did this team do in the 2023 playoffs? (At the time of submission: Q: Are they still in contention for the 2023 NBA Championship?)

`A: They are currently still in the playoffs and winning their current series of games at this t`