Introduction: Anyone who follows the NBA is aware of the increase in 3 point attempts in recent years. As the 3-point shot has exponentially increased, it's 2-point counterpart (the midrange shot) is quickly becoming extinct. Many people attribute it to Stephen Curry whose rise to stardom follows a similar path to the 3-point shot , but others blame it on Data Analysts on NBA teams who argue that the 3-point shot is a lot more efficient than the 2-point shot, and more importantly, they agree that the 2-point shot is inefficient in and of itself.

Goal: My goal for this project to explore if NBA teams and players were correct in their stance that players should opt for 3-point shots if they want to succed in the league. I'm going to be creating vizualizations which I hope can answer some of my questions such as "Are NBA players who follow this playstyle more succesful" by comparing their three-point attempts and their efficiency/team success"

In [1]:
```
import pandas as pd

NBA_season_stats = pd.read_csv('Seasons_Stats.csv')

#Since the three-point line wasn't implemented until Fall-1979, I'm going to onl

NBA_Modern_stats = NBA_season_stats.loc[NBA_season_stats['Year'] > 1979]
NBA_Modern_stats.head()
```

Out[1]:

	Unnamed: 0	Year	Player	Pos	Age	Tm	G	GS	MP	PER	...	FT%	ORB
5727	5727	1980.0	Kareem Abdul-Jabbar*	C	32.0	LAL	82.0	NaN	3143.0	25.3	...	0.765	190.0
5728	5728	1980.0	Tom Abernethy	PF	25.0	GSW	67.0	NaN	1222.0	11.0	...	0.683	62.0
5729	5729	1980.0	Alvan Adams	C	25.0	PHO	75.0	NaN	2168.0	19.2	...	0.797	158.0
5730	5730	1980.0	Tiny Archibald*	PG	31.0	BOS	80.0	80.0	2864.0	15.3	...	0.830	59.0
5731	5731	1980.0	Dennis Awtrey	C	31.0	CHI	26.0	NaN	560.0	7.4	...	0.640	29.0

5 rows × 53 columns

In [2]:
```
for col in NBA_Modern_stats.columns:
    print(col)
```

```
Unnamed: 0
Year
Player
Pos
Age
Tm
G
GS
MP
PER
```

```
TS%
3PAr
FTr
ORB%
DRB%
TRB%
AST%
STL%
BLK%
TOV%
USG%
blanl
OWS
DWS
WS
WS/48
blank2
OBPM
DBPM
BPM
VORP
FG
FGA
FG%
3P
3PA
3P%
2P
2PA
2P%
eFG%
FT
FTA
FT%
ORB
DRB
TRB
AST
STL
BLK
TOV
PF
PTS
```

Data Description: As you can see above, this data shows statisitics for every player in the NBA since 1955. Since the three point line wasn't invented until 1979 (first- three pointer was in october 1979), I decided to filter the data, so it only shows data from 1980 to present. Also, as you can see above there are dozens of features including 2 point attempts and three-point attempts which is very important for my analysis. It also has features such total points, FG%, and eFG%, which are all solid indicators on the success of a player in the NBA.

Three Point Progression OVER the Years:

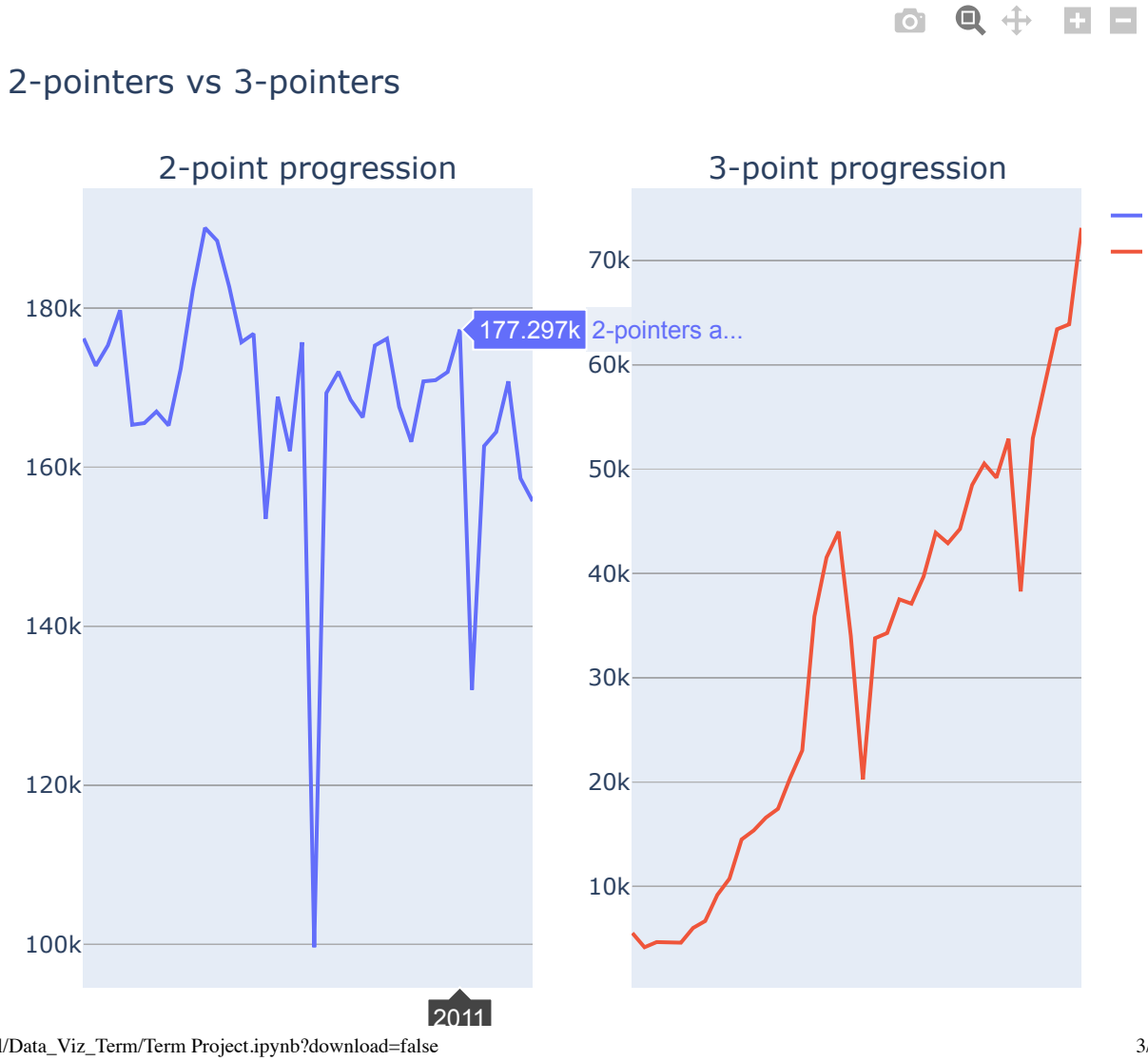
First and foremost, I want to begin by exploring, the progression of three-point and two-point attempts in the last four decades. My goal here is here to vizualize this trend before I proceed to my other questions.

In [3]:
```
from plotly.subplots import make_subplots
import plotly.graph_objects as go
```

In [4]:
```
Three_Point = NBA_Modern_stats[['Year', 'Player', 'G', '2PA', '3PA', 'eFG%']]

Attempts_Holders = Three_Point.groupby('Year').sum()
Three_Point_years = Attempts_Holders.index
Three_Point_years
```

Out[4]:
```
Float64Index([1980.0, 1981.0, 1982.0, 1983.0, 1984.0, 1985.0, 1986.0, 1987.0,
              1988.0, 1989.0, 1990.0, 1991.0, 1992.0, 1993.0, 1994.0, 1995.0,
              1996.0, 1997.0, 1998.0, 1999.0, 2000.0, 2001.0, 2002.0, 2003.0,
              2004.0, 2005.0, 2006.0, 2007.0, 2008.0, 2009.0, 2010.0, 2011.0,
              2012.0, 2013.0, 2014.0, 2015.0, 2016.0, 2017.0],
             dtype='float64', name='Year')
```

In [5]:
```
fig = make_subplots(rows=1, cols=2, subplot_titles=("2-point progression", "3-po
fig.add_trace(
    go.Scatter(x=Three_Point_years, y=Attempts_Holders['2PA'], name = "2-pointer
    row=1, col=1
)
fig.add_trace(
    go.Scatter(x=Three_Point_years, y=Attempts_Holders['3PA'], name = "3-pointer
    row=1, col=2
)
fig.update_layout(height=600, width=800, title_text="2-pointers vs 3-pointers",
xaxis={'visible': False, 'showticklabels': False},xaxis2={'visible': False, 'sho
fig.show()
```



In [ ]:

As you can see above, I was able to vizualize the progression of three-point and two-point attempts using a bar chart. After storing only the necessary features in dataframe, I grouped the data by year and summed it using pandas/groupby, so that I could store the total amount of attempts for each year since 1980. I also stored the year in a dataframe, to make it easier to peform the vizualization.

In the two bar charts, the first thing you notice is that three-pointer progression chart has had a steady increase almost every season since 1980. There is an outlier or two, which is due to lockout NBA seasons which had less games than usual. On the 2-point progression chart, you can see a slight decrease on average as time passes. You can notice that the first-two decades in the chart have similar totals, but the second half of the chart sees a dip which has stayed consistent for the past couple of years.

In [ ]:

EFG% across the league as three pointers increased:

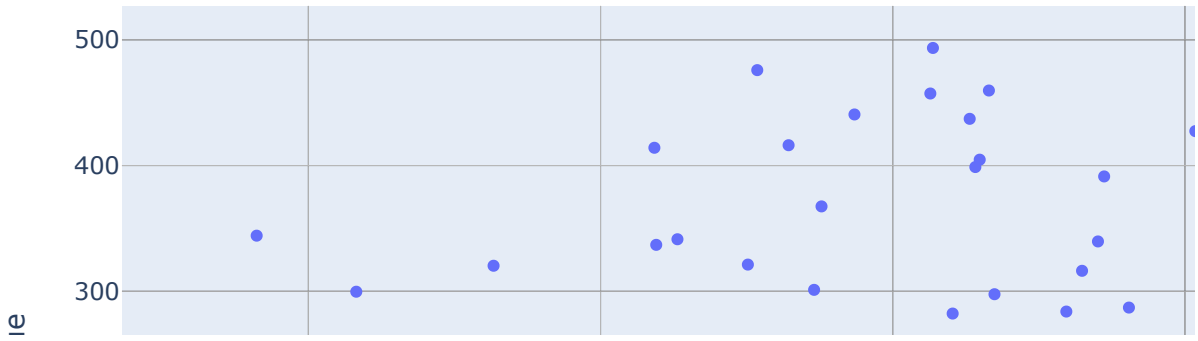
For this next vizualization, I want to figure out if players became more efficient as they increased three-pointers. In basketball, efficiency is generally described as getting as much points you can per possesion. In terms of jump-shooting, this is computed using FG% which divides your made shots and your total attempted shots. As three-pointers became a popular staple in NBA games, statisticians began using EFG% which accounts for three-point shots and adds a slight multiplier for them. This is due to the obvious fact that three-pointers are worth more than two-pointers. Since we're trying to figure out if which players are getting the most-points per possesion, it makes sense since shooting 20- three pointers at 40% FG versus 20-two pointers at 50%, the three-point shots will generate more points.

In [6]:
```
efg_holder = Three_Point.groupby('Year').mean()
```

In [7]:
```
import plotly.express as px
```

In [15]:
```
fig = px.scatter(efg_holder, x="eFG%", y=['2PA','3PA'],
title = 'Are players more efficient as they take more threes?')

def customLegend(fig, nameSwap):
    for i, dat in enumerate(fig.data):
        for elem in dat:
```

```
            if elem == 'name':
                fig.data[i].name = nameSwap[fig.data[i].name]
    return(fig)

fig = customLegend(fig=fig, nameSwap = {'2PA': '2-pointers attempted', '3PA':'3-

fig.show()
```



In [ ]:

First, I grouped my data by year and averaged the shot attempts per year. Next, I vizualized a scatter plot which showed an interesting conclusion. In the data vizualization, the orange points are 3-point attempts and the 2-point attempts are the blue points. As you can see in the vizualization, as three-pointers increased, players eFG% were rising, which means players were becoming more efficient and succesful as they increased their three-point output. If you focus on the 2PA attempts side of the scatter plot, you notice that players in the league had average efficiency during the peak of 2PA attempts, and as they decreased their 2-point attempts and increased three-point attempts, their efficiency had a positive boost.

Do Three Pointers have a correlation to team

success?

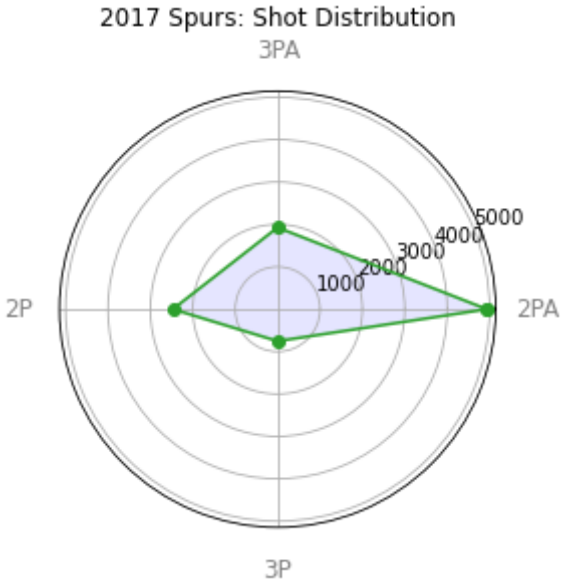
Now, I want to explore if shooting more three-pointers has any correlation to team success. I did some research and found out the most succesful team during the past decade has been the spurs, and the least successful team during that same period has been the Minnesota Timberwolves. For this part of the analysis, I want to compare their shot distribution to see if there's any significant differences in their shot attempts.

In [62]:
```
New_df = NBA_Modern_stats.groupby('Tm')
New_df
```

Out[62]:
```
<pandas.core.groupby.generic.DataFrameGroupBy object at 0x11fbadc70>
```

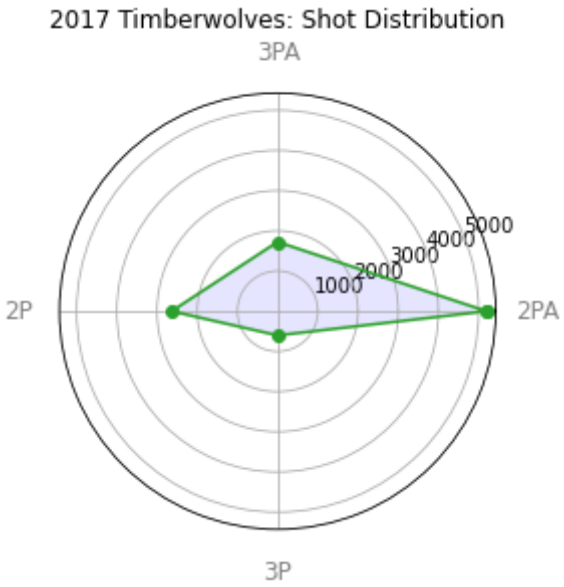
In [157…
```
df_1 = NBA_Modern_stats.loc[NBA_Modern_stats['Tm']== 'SAS']

df_Modern_Spurs = df_1.loc[df_1['Year'].isin([2011,2012,2013,2014,2015,2016,2017
Spurs_1 = df_Modern_Spurs[['Year','2PA','3PA','PTS','FT','2P','3P','Tm','3P%','F

Modern_Spurs = Spurs_1.groupby('Year').sum()
```

In [158…
```
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np

df = pd.DataFrame({'Col A': ['2PA', '3PA', '2P', '3P'],
                   'Col B': [4937.0, 1927.0, 2469.0, 753.0]})
fig = plt.figure()
ax = fig.add_subplot(111, projection="polar")

theta = np.arange(len(df) + 1) / float(len(df)) * 2 * np.pi
values = df['Col B'].values
values = np.append(values, values[0])


l1, = ax.plot(theta, values, color="C2", marker="o", label="Name of Col B")
plt.xticks(theta[:-1], df['Col A'], color='grey', size=12)
ax.tick_params(pad=10)
ax.fill(theta, values, 'blue', alpha=0.1)

plt.title("2017 Spurs: Shot Distribution")
plt.show()
```



In [165…

```
df_2 = NBA_Modern_stats.loc[NBA_Modern_stats['Tm']== 'MIN']

df_Modern_Timbs = df_2.loc[df_2['Year'].isin([2011,2012,2013,2014,2015,2016,2017
Timbs_1 = df_Modern_Timbs[['Year','2PA','3PA','PTS','FT','2P','3P','Tm','3P%','F

Modern_Timbs = Timbs_1.groupby('Year').sum()
#Modern_Spurs = Spurs_1.groupby(['Year']).sum().reset_index().groupby('Year').su

import pandas as pd
import matplotlib.pyplot as plt
import numpy as np

df = pd.DataFrame({'Col A': ['2PA', '3PA', '2P', '3P'],
                   'Col B': [5198, 1724, 2634.0, 601.0]})
fig = plt.figure()
ax = fig.add_subplot(111, projection="polar")

theta = np.arange(len(df) + 1) / float(len(df)) * 2 * np.pi
values = df['Col B'].values
values = np.append(values, values[0])


l1, = ax.plot(theta, values, color="C2", marker="o", label="Name of Col B")
plt.xticks(theta[:-1], df['Col A'], color='grey', size=12)
ax.tick_params(pad=10)
ax.fill(theta, values, 'blue', alpha=0.1)

plt.title("2017 Timberwolves: Shot Distribution")
plt.show()
```

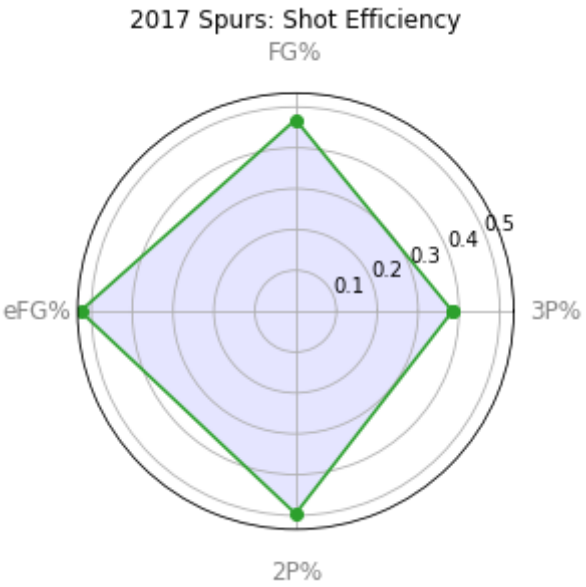


Using a radar chart, I compared both of those team's shot distribution during the 2017 season. While it wasn't significant, I noticed there was a notable difference in three-point attempts, with the spurs edging the timberwolves with almost 200 extra three-point attempts. Another thing to note, is that the timberwolves also had close to 300 more 2-point attempts than the spurs. Since these these two teams are on the opposite sides of the spectrum in terms of winnnig, I feel this indicates that three-point attempts does have a correlation with winning in today's NBA.

Next, I wanted to compare the percentage of the shots that these two teams were actually making. I feel that there is more to the story than just shooting more three-pointers.

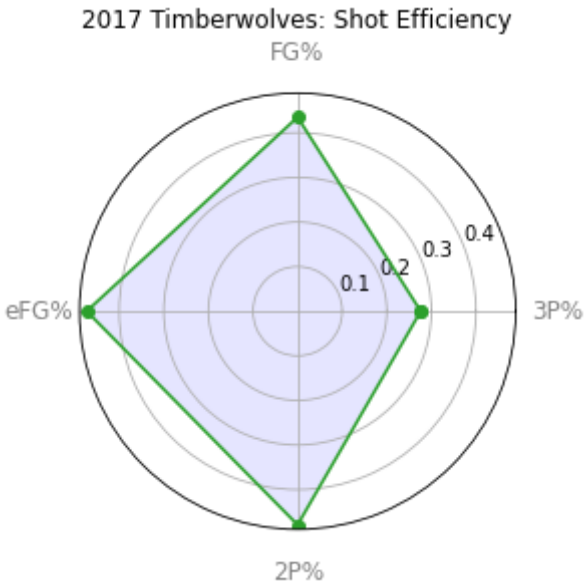
In [159…

```
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np


Modern_Spurs_1 = Spurs_1.groupby('Year').mean()

df = pd.DataFrame({'Col A': ['3P%', 'FG%', 'eFG%', '2P%'],
                   'Col B': [0.384, .469, .527, .496]})
fig = plt.figure()
ax = fig.add_subplot(111, projection="polar")

theta = np.arange(len(df) + 1) / float(len(df)) * 2 * np.pi
values = df['Col B'].values
values = np.append(values, values[0])


l1, = ax.plot(theta, values, color="C2", marker="o", label="Name of Col B")
plt.xticks(theta[:-1], df['Col A'], color='grey', size=12)
ax.tick_params(pad=10)
ax.fill(theta, values, 'blue', alpha=0.1)

plt.title("2017 Spurs: Shot Efficiency")
plt.show()
```



In [167…

```
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np

Modern_Timbs_1 = Timbs_1.groupby('Year').mean()

df = pd.DataFrame({'Col A': ['3P%', 'FG%', 'eFG%', '2P%'],
                   'Col B': [0.276, .435, .473, .479]})
fig = plt.figure()
ax = fig.add_subplot(111, projection="polar")

theta = np.arange(len(df) + 1) / float(len(df)) * 2 * np.pi
values = df['Col B'].values
values = np.append(values, values[0])


l1, = ax.plot(theta, values, color="C2", marker="o", label="Name of Col B")
plt.xticks(theta[:-1], df['Col A'], color='grey', size=12)
ax.tick_params(pad=10)
ax.fill(theta, values, 'blue', alpha=0.1)

plt.title("2017 Timberwolves: Shot Efficiency")
plt.show()
```



As you can see above in the two radar charts, there is a significant difference in these two teams' shot making ability. In term of 3-point percentage, the Spurs blew the Timberwolves out the water. They are also more efficient in 2-point percentage, Field Goal percetnage, and effective Field goal percentage. This means that the Spurs are shooting a slightly more three-pointers than the timberwolves, but making a percentage of those three-pointers.

Overall, I think this means that in the Modern NBA, teams should have a focus on increasing three-point attempts, but realistically, their goal should be to sign NBA players who are great at shooting three-pointers. As you saw above, Minnessota was shooting a lot of three-pointers and with a low percentage of makes, it didn't result in wins for the team.

DOES three-point attempts correlate to individual player success??

Finally, I want to explore if the players who have had the most- success in the NBA opt for more three-pointers than two-pointers. In my project, I'm only measuring success based on their scoring ability amongst their peers. That's why I'm going to be examing James Harden, who has been deemed the best scorer of this era. James Harden's prime also matches up with the peak of three-point shooting around the league, so he should be a good indication on what shots you should take if you want to be successful in the league.

In [20]:
```
import pandas as pd
import numpy as np

from scipy.stats import norm, gaussian_kde, percentileofscore

pd.options.display.max_columns = None
from nba_api.stats.static import players
from nba_api.stats.endpoints import shotchartdetail
from nba_api.stats.endpoints import playercareerstats


import matplotlib.pyplot as plt
```

```
import seaborn as sns

from matplotlib import cm
from matplotlib.patches import Circle, Rectangle, Arc, ConnectionPatch
from matplotlib.patches import Polygon
from matplotlib.collections import PatchCollection
from matplotlib.colors import LinearSegmentedColormap, ListedColormap, BoundaryN
from matplotlib.path import Path
from matplotlib.patches import PathPatch

sns.set_style('white')
sns.set_color_codes()
```

In [21]:

```
def get_player_shotchartdetail(player_name, season_id):
    nba_players = players.get_players()
    player_dict = [player for player in nba_players if player['full_name'] == pl
    career = playercareerstats.PlayerCareerStats(player_id=player_dict['id'])
    career_df = career.get_data_frames()[0]
    team_id = career_df[career_df['SEASON_ID'] == season_id]['TEAM_ID']
    shotchartlist = shotchartdetail.ShotChartDetail(team_id=int(team_id),
                                                    player_id=int(player_dict['id
                                                    season_type_all_star='Regular
                                                    season_nullable=season_id,
                                                    context_measure_simple="FGA")

    return shotchartlist[0], shotchartlist[1]
```

In [22]:

```
Harden_shots, league_avg = get_player_shotchartdetail('James Harden', '2018-19')
```

In [23]:

```
def draw_court(ax=None, color="blue", lw=1, shotzone=False, outer_lines=False):

    if ax is None:
        ax = plt.gca()

    hoop = Circle((0, 0), radius=7.5, linewidth=lw, color=color, fill=False)


    backboard = Rectangle((-30, -12.5), 60, 0, linewidth=lw, color=color)

    outer_box = Rectangle((-80, -47.5), 160, 190, linewidth=lw, color=color,
                          fill=False)

    inner_box = Rectangle((-60, -47.5), 120, 190, linewidth=lw, color=color,
                          fill=False)

    top_free_throw = Arc((0, 142.5), 120, 120, theta1=0, theta2=180,
                         linewidth=lw, color=color, fill=False)

    bottom_free_throw = Arc((0, 142.5), 120, 120, theta1=180, theta2=0,
                            linewidth=lw, color=color, linestyle='dashed')

    restricted = Arc((0, 0), 80, 80, theta1=0, theta2=180, linewidth=lw,
                     color=color)


    corner_three_a = Rectangle((-220, -47.5), 0, 140, linewidth=lw,
                               color=color)
```

```
    corner_three_b = Rectangle((220, -47.5), 0, 140, linewidth=lw, color=color)

    three_arc = Arc((0, 0), 475, 475, theta1=22, theta2=158, linewidth=lw,
                    color=color)


    center_outer_arc = Arc((0, 422.5), 120, 120, theta1=180, theta2=0,
                           linewidth=lw, color=color)
    center_inner_arc = Arc((0, 422.5), 40, 40, theta1=180, theta2=0,
                           linewidth=lw, color=color)

    if (shotzone == True):
        inner_circle = Circle((0, 0), radius=80, linewidth=lw, color='black', fi
        outer_circle = Circle((0, 0), radius=160, linewidth=lw, color='black', f
        corner_three_a_x =  Rectangle((-250, 92.5), 30, 0, linewidth=lw, color=c
        corner_three_b_x = Rectangle((220, 92.5), 30, 0, linewidth=lw, color=col


        inner_line_1 = Rectangle((40, 69.28), 80, 0, 60, linewidth=lw, color=col

        inner_line_2 = Rectangle((-40, 69.28), 80, 0, 120, linewidth=lw, color=c


        inner_line_3 = Rectangle((53.20, 150.89), 290, 0, 70.53, linewidth=lw, c
        inner_line_4 = Rectangle((-53.20, 150.89), 290, 0, 109.47, linewidth=lw,

        inner_line_5 = Rectangle((130.54, 92.5), 80, 0, 35.32, linewidth=lw, col
        inner_line_6 = Rectangle((-130.54, 92.5), 80, 0, 144.68, linewidth=lw, c


        court_elements = [hoop, backboard, outer_box, inner_box, top_free_throw,
                          bottom_free_throw, restricted, corner_three_a,
                          corner_three_b, three_arc, center_outer_arc,
                          center_inner_arc, inner_circle, outer_circle,
                          corner_three_a_x, corner_three_b_x,
                          inner_line_1, inner_line_2, inner_line_3, inner_line_4
    else:
        court_elements = [hoop, backboard, outer_box, inner_box, top_free_throw,
                          bottom_free_throw, restricted, corner_three_a,
                          corner_three_b, three_arc, center_outer_arc,
                          center_inner_arc]

    if outer_lines:
        outer_lines = Rectangle((-250, -47.5), 500, 470, linewidth=lw,
                                color=color, fill=False)
        court_elements.append(outer_lines)

    for element in court_elements:
        ax.add_patch(element)


    return ax
```

In [24]:

```
def shot_chart(data, title="", color="b",
               xlim=(-250, 250), ylim=(422.5, -47.5), line_color="blue",
               court_color="white", court_lw=2, outer_lines=False,
               flip_court=False, gridsize=None,
               ax=None, despine=False, **kwargs):
```

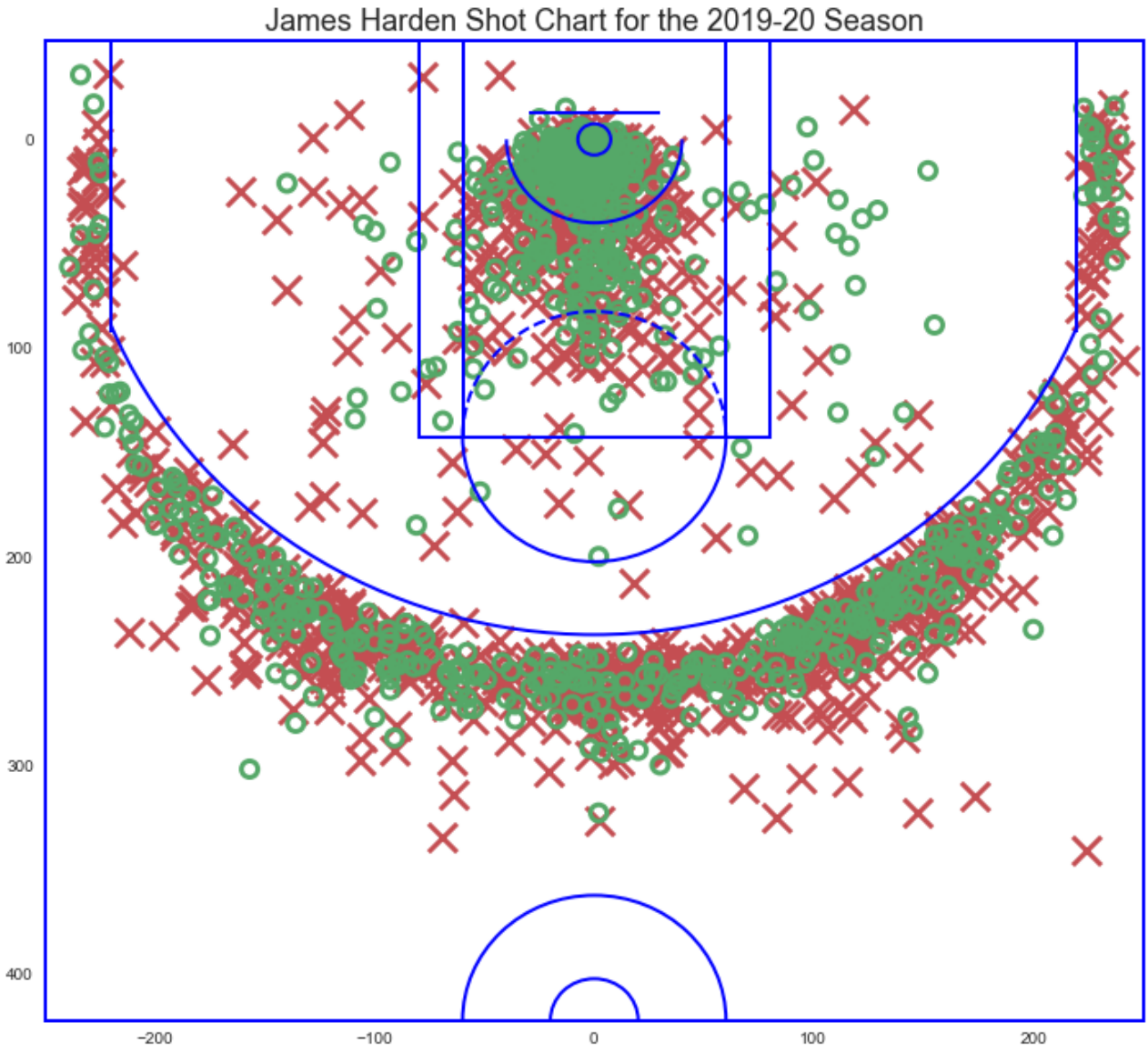
```
    if ax is None:
        ax = plt.gca()

    if not flip_court:
        ax.set_xlim(xlim)
        ax.set_ylim(ylim)
    else:
        ax.set_xlim(xlim[::-1])
        ax.set_ylim(ylim[::-1])

    ax.tick_params(labelbottom="off", labelleft="off")
    ax.set_title(title, fontsize=18)


    draw_court(ax, color=line_color, lw=court_lw, outer_lines=outer_lines)

    x_missed = data[data['EVENT_TYPE'] == 'Missed Shot']['LOC_X']
    y_missed = data[data['EVENT_TYPE'] == 'Missed Shot']['LOC_Y']

    x_made = data[data['EVENT_TYPE'] == 'Made Shot']['LOC_X']
    y_made = data[data['EVENT_TYPE'] == 'Made Shot']['LOC_Y']

    ax.scatter(x_missed, y_missed, c='r', marker="x", s=300, linewidths=3, **kwa
    ax.scatter(x_made, y_made, facecolors='none', edgecolors='g', marker="o", s=


    for spine in ax.spines:
        ax.spines[spine].set_lw(court_lw)
        ax.spines[spine].set_color(line_color)

    if despine:
        ax.spines["top"].set_visible(False)
        ax.spines["bottom"].set_visible(False)
        ax.spines["right"].set_visible(False)
        ax.spines["left"].set_visible(False)

    return ax
```

In [25]:

```
plt.rcParams['figure.figsize'] = (12, 11)
```

In [35]:

```
shot_chart(Harden_shots, title="James Harden Shot Chart for the 2019-20 Season")
plt.show()

print("FG%: ", "{0:.3f}".format(Harden_shots['SHOT_MADE_FLAG'].sum()/len(Harden_

TGREEN =  '\033[32m'
print (TGREEN + "O: Made Shots")
TRED =  '\033[31m'
print (TRED + "X: Missed Shots")
```




```
FG%:  0.442 (843-1909)
O: Made Shots
X: Missed Shots
```

The

shotchart above shows all the attempted shots in James Harden 2019-2020 season. the red x's represent missed shots and the green circles represent made shots. As you can see the three point is completely filled with shot attempts. In the mid-range area, there is barely any attempts, and you can see the court clearly. The only two-point shots that were taken are layups/shots under the rim. This indicates that one of top players in the league took almost no mid-range shots.

In [27]:

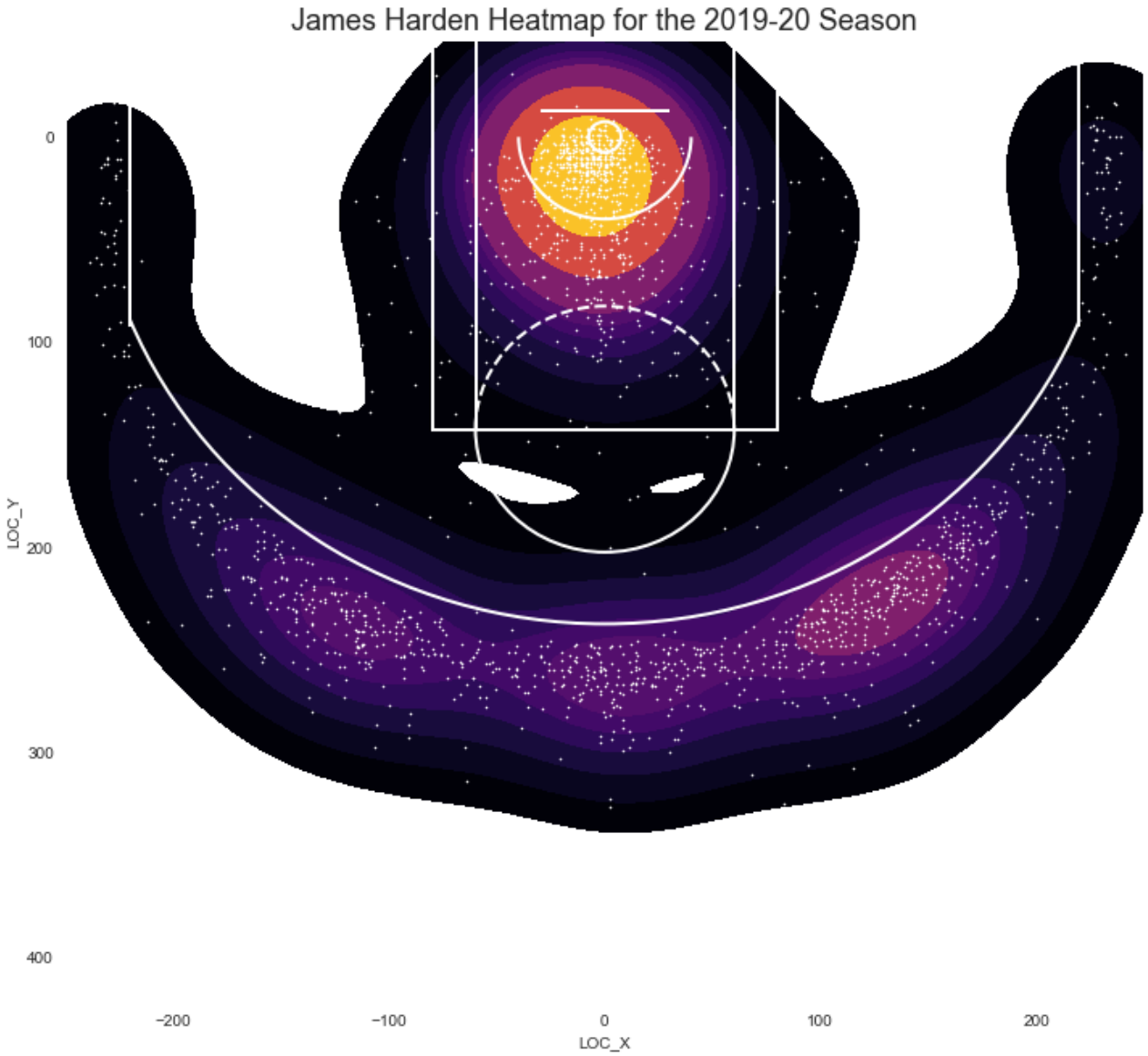
```
def heatmap(data, title="", color="b",
                xlim=(-250, 250), ylim=(422.5, -47.5), line_color="white",
                court_color="white", court_lw=2, outer_lines=False,
                flip_court=False, gridsize=None,
                ax=None, despine=False, **kwargs):

    if ax is None:
```

```
        ax = plt.gca()

    if not flip_court:
        ax.set_xlim(xlim)
        ax.set_ylim(ylim)
    else:
        ax.set_xlim(xlim[::-1])
        ax.set_ylim(ylim[::-1])

    ax.tick_params(labelbottom="off", labelleft="off")
    ax.set_title(title, fontsize=18)

    # draws the court
    draw_court(ax, color=line_color, lw=court_lw, outer_lines=outer_lines)

    x = data['LOC_X']
    y = data['LOC_Y']

    sns.kdeplot(x, y, shade=True, cmap='inferno', ax=ax, **kwargs)

    ax.scatter(x, y, facecolors='w', s=2, linewidths=0.1, **kwargs)

    # Set the spines to match the rest of court lines, makes outer_lines
    # somewhate unnecessary
    for spine in ax.spines:
        ax.spines[spine].set_lw(court_lw)
        ax.spines[spine].set_color(line_color)

    if despine:
        ax.spines["top"].set_visible(False)
        ax.spines["bottom"].set_visible(False)
        ax.spines["right"].set_visible(False)
        ax.spines["left"].set_visible(False)

    return ax
```

In [37]:
```
heatmap(Harden_shots,title="James Harden Heatmap for the 2019-20 Season" )
```

```
/opt/anaconda3/lib/python3.8/site-packages/seaborn/_decorators.py:36: FutureWarn
ing:

Pass the following variable as a keyword arg: y. From version 0.12, the only val
id positional argument will be `data`, and passing other arguments without an ex
plicit keyword will result in an error or misinterpretation.
```

Out[37]:
```
<AxesSubplot:title={'center':'James Harden Heatmap for the 2019-20 Season'}, xla
bel='LOC_X', ylabel='LOC_Y'>
```



Heatmap:

I also designed a heatmap which is also another representation of James Harden's 2019-2020 season. The heatmap has black for areas of the court where James Harden wasn't really attempting shots. The darker colors like purple and orange/red represent the areas of the court where most of James Harden's shots were coming from. It reiterates what we saw on his shotchart.James Harden completely ignores the mid-range shot.

I thought it would be a great idea to represent James Harden's shot chart in the form of a heatmap because I feel it's easier to understand than a shotchart. Even a casual NBA fan can quickly understand a heat map, but sometimes shot charts are a little trick.

CONCLUSION:

After analyzing this dataset, and producing several vizualizations, it's obvious that three-point shooting is at an all-time high, but my original question was "should players and team opt for

more threes (rather than 2's) if they want to succeed in this league". The vizualizations point to a resounding yes, but I believe that there's more to the story. But first, let's reveiw what we discovered. With the line charts, we seen the steady increase in three-pointers in the league almost every year, while 2-pointers were slowly decreasing as time passed. We also analyzed how more three-pointers affect player efficiency. We concluded that as the league has increased their three-point output and decreased their 2-point attempts, as a whole players have become more efficient. We also chose one of the best scorers in the league (MVP candidate for the past 5 years) and his shotchart and heatmap showed us he only shoots three-pointers and layups. Finally, going back to the radar charts, which compares the Spurs and Timberwolves three-point shooting. While the Spurs did have more three-point attempts than the Timberwolves, the Timberwolves weren't that far off in attempts. On the other hand, the percentage of three-point attempts that were actually converting were substantial. This causes me to believe that shooting more three-pointers isn't the only goal, having players on your team that can convert those attempts at a high rate would be closer to the goal that teams should chase if they want to be successful in the NBA.

In [ ]: