NBA Player Performance Analysis in Different Season---Brandon Ingram

In [9]:

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

In [10]:

```
from sklearn.model_selection import train_test_split, cross_val_score
from sklearn.linear_model import LinearRegression
from sklearn import metrics
from sklearn.pipeline import make_pipeline
```

In [11]:

```
games = pd.read_csv("games.csv")
details = pd.read_csv("games_details.csv")
teams = pd.read_csv("teams.csv")
players = pd.read_csv("players.csv")
ranking = pd.read_csv("ranking.csv")
```

In [12]:

```
players.head()
```

Out[12]:

	PLAYER_NAME	TEAM_ID	PLAYER_ID	SEASON
0	Royce O'Neale	1610612762	1626220	2019
1	Bojan Bogdanovic	1610612762	202711	2019
2	Rudy Gobert	1610612762	203497	2019
3	Donovan Mitchell	1610612762	1628378	2019
4	Mike Conley	1610612762	201144	2019

In [13]:

```
details.head()
```

Out[13]:

	GAME_ID	TEAM_ID	TEAM_ABBREVIATION	TEAM_CITY	PLAYER_ID	PLAYER_NAME	NIC
0	22100213	1610612764	WAS	Washington	203484	Kentavious Caldwell-Pope	Ke
1	22100213	1610612764	WAS	Washington	1628398	Kyle Kuzma	
2	22100213	1610612764	WAS	Washington	1629655	Daniel Gafford	
3	22100213	1610612764	WAS	Washington	203078	Bradley Beal	
4	22100213	1610612764	WAS	Washington	203915	Spencer Dinwiddie	

5 rows × 29 columns

In [14]:

```
teams.head()
```

Out[14]:

	LEAGUE_ID	TEAM_ID	MIN_YEAR	MAX_YEAR	ABBREVIATION	NICKNAME	YEARFOUNDI
0	0	1610612737	1949	2019	ATL	Hawks	19
1	0	1610612738	1946	2019	BOS	Celtics	19
2	0	1610612740	2002	2019	NOP	Pelicans	20
3	0	1610612741	1966	2019	CHI	Bulls	19
4	0	1610612742	1980	2019	DAL	Mavericks	19

In [15]:

```
ranking.head()
```

Out[15]:

	TEAM_ID	LEAGUE_ID	SEASON_ID	STANDINGSDATE	CONFERENCE	TEAM	G	W	L
0	1610612744	0	22021	2021-11-17	West	Golden State	14	12	2
1	1610612756	0	22021	2021-11-17	West	Phoenix	14	11	3
2	1610612742	0	22021	2021-11-17	West	Dallas	14	9	5
3	1610612743	0	22021	2021-11-17	West	Denver	14	9	5
4	1610612746	0	22021	2021-11-17	West	LA Clippers	14	9	5

In [16]:

```
games.head()
```

Out[16]:

	GAME_DATE_EST	GAME_ID	GAME_STATUS_TEXT	HOME_TEAM_ID	VISITOR_TEAM_ID	SEA
0	2021-11-17	22100213	Final	1610612766	1610612764	
1	2021-11-17	22100214	Final	1610612765	1610612754	
2	2021-11-17	22100215	Final	1610612737	1610612738	
3	2021-11-17	22100216	Final	1610612751	1610612739	
4	2021-11-17	22100217	Final	1610612748	1610612740	

5 rows × 21 columns

In [17]:

```
details = details.drop_duplicates(subset=["GAME_ID","PLAYER_NAME"])
details.head
```

Out[17]:

```
<bound method NDFrame.head of          GAME_ID     TEAM_ID TEAM_ABBREVIATION   TEAM_
CITY  PLAYER_ID  \
0       22100213  1610612764              WAS  Washington     203484
1       22100213  1610612764              WAS  Washington    1628398
2       22100213  1610612764              WAS  Washington    1629655
3       22100213  1610612764              WAS  Washington     203078
4       22100213  1610612764              WAS  Washington     203915
...          ...         ...              ...         ...        ...
626106  11200005  1610612743              DEN      Denver     202706
626107  11200005  1610612743              DEN      Denver     202702
626108  11200005  1610612743              DEN      Denver     201585
626109  11200005  1610612743              DEN      Denver     202389
626110  11200005  1610612743              DEN      Denver     201951

                    PLAYER_NAME    NICKNAME START_POSITION COMMENT    MIN  \
0       Kentavious Caldwell-Pope  Kentavious              F     NaN  27:41
1                   Kyle Kuzma        Kyle              F     NaN  30:28
2                Daniel Gafford      Daniel              C     NaN  24:21
3                  Bradley Beal     Bradley              G     NaN  35:07
4              Spencer Dinwiddie     Spencer              G     NaN  28:34
...                          ...         ...            ...     ...    ...
626106           Jordan Hamilton         NaN            NaN     NaN     19
626107            Kenneth Faried         NaN            NaN     NaN     23
626108              Kosta Koufos         NaN            NaN     NaN     15
626109            Timofey Mozgov         NaN            NaN     NaN     19
626110                 Ty Lawson         NaN            NaN     NaN     27

        ...  OREB  DREB  REB  AST  STL  BLK   TO   PF   PTS  PLUS_MINUS
0       ...   1.0   5.0  6.0  2.0  1.0  0.0  1.0  0.0   3.0         2.0
1       ...   1.0   4.0  5.0  3.0  1.0  2.0  1.0  1.0   5.0       -14.0
2       ...   2.0   7.0  9.0  1.0  2.0  1.0  1.0  4.0  20.0        -2.0
3       ...   0.0   3.0  3.0  7.0  2.0  0.0  2.0  3.0  24.0        -9.0
4       ...   0.0   3.0  3.0  2.0  0.0  0.0  2.0  1.0   0.0        -5.0
...     ...   ...   ...  ...  ...  ...  ...  ...  ...   ...         ...
626106  ...   0.0   2.0  2.0  0.0  2.0  0.0  1.0  3.0  17.0         NaN
626107  ...   1.0   0.0  1.0  1.0  1.0  0.0  3.0  3.0  18.0         NaN
626108  ...   3.0   5.0  8.0  0.0  1.0  0.0  0.0  3.0   6.0         NaN
626109  ...   1.0   2.0  3.0  1.0  0.0  0.0  4.0  2.0   2.0         NaN
626110  ...   0.0   2.0  2.0  6.0  2.0  0.0  6.0  1.0   8.0         NaN

[625783 rows x 29 columns]>
```

In [19]:

```
ingram = details[details["PLAYER_NAME"]== "Brandon Ingram"]
ingram.drop(["TEAM_ID","TEAM_CITY","PLAYER_ID","PLAYER_NAME","COMMENT"],axis= "columns",inplace=Tru
```

In [20]:

```
ingram.head
```

Out[20]:

```
<bound method NDFrame.head of          GAME_ID TEAM_ABBREVIATION NICKNAME START_POSI
TION    MIN  FGM   FGA  \
105     22100217               NOP  Brandon              F  37:28  6.0  14.0
415     22100201               NOP  Brandon              F  32:07  9.0  23.0
926     22100188               NOP  Brandon              F  29:27  7.0  17.0
1884    22100150               NOP  Brandon            NaN    NaN  NaN   NaN
2544    22100130               NOP  Brandon            NaN    NaN  NaN   NaN
...          ...               ...      ...            ...    ...  ...   ...
477238  11600058               LAL      NaN            NaN  27:50  3.0   7.0
477661  11600047               LAL      NaN            NaN  24:12  2.0   6.0
477986  11600036               LAL      NaN            NaN  19:04  1.0   2.0
478310  11600028               LAL      NaN            NaN  17:28  2.0   8.0
478756  11600015               LAL      NaN            NaN  26:41  0.0   5.0

        FG_PCT  FG3M  FG3A  ...  OREB  DREB  REB  AST  STL  BLK   TO   PF  \
105      0.429   1.0   6.0  ...   0.0   4.0  4.0  5.0  0.0  0.0  5.0  2.0
415      0.391   2.0   4.0  ...   0.0   5.0  5.0  4.0  0.0  0.0  1.0  1.0
926      0.412   1.0   4.0  ...   0.0   3.0  3.0  4.0  0.0  1.0  4.0  3.0
1884       NaN   NaN   NaN  ...   NaN   NaN  NaN  NaN  NaN  NaN  NaN  NaN
2544       NaN   NaN   NaN  ...   NaN   NaN  NaN  NaN  NaN  NaN  NaN  NaN
...        ...   ...   ...  ...   ...   ...  ...  ...  ...  ...  ...  ...
477238   0.429   1.0   4.0  ...   1.0   1.0  2.0  2.0  0.0  0.0  0.0  2.0
477661   0.333   1.0   3.0  ...   1.0   1.0  2.0  0.0  0.0  1.0  1.0  1.0
477986   0.500   0.0   0.0  ...   0.0   2.0  2.0  0.0  1.0  3.0  0.0  1.0
478310   0.250   1.0   2.0  ...   1.0   1.0  2.0  2.0  1.0  1.0  0.0  1.0
478756   0.000   0.0   4.0  ...   0.0   2.0  2.0  1.0  1.0  2.0  1.0  2.0

         PTS  PLUS_MINUS
105     19.0       -13.0
415     31.0        -2.0
926     19.0        10.0
1884     NaN         NaN
2544     NaN         NaN
...      ...         ...
477238   7.0       -27.0
477661   5.0         1.0
477986   4.0         0.0
478310   6.0        18.0
478756   2.0        20.0

[378 rows x 24 columns]>
```

In [21]:

```
game_date = games[["GAME_DATE_EST","GAME_ID","SEASON"]]
game_date.head()
```

Out[21]:

	GAME_DATE_EST	GAME_ID	SEASON
0	2021-11-17	22100213	2021
1	2021-11-17	22100214	2021
2	2021-11-17	22100215	2021
3	2021-11-17	22100216	2021
4	2021-11-17	22100217	2021

In [22]:

```
stats=ingram.merge(game_date,on="GAME_ID", how="left")
```

In [23]:

```
stats.head
```

Out[23]:

```
<bound method NDFrame.head of       GAME_ID TEAM_ABBREVIATION NICKNAME START_POSITIO
N    MIN  FGM   FGA  \
0    22100217              NOP  Brandon              F  37:28  6.0  14.0
1    22100201              NOP  Brandon              F  32:07  9.0  23.0
2    22100188              NOP  Brandon              F  29:27  7.0  17.0
3    22100150              NOP  Brandon            NaN    NaN  NaN   NaN
4    22100130              NOP  Brandon            NaN    NaN  NaN   NaN
..        ...              ...      ...            ...    ...  ...   ...
376  11600058              LAL      NaN            NaN  27:50  3.0   7.0
377  11600047              LAL      NaN            NaN  24:12  2.0   6.0
378  11600036              LAL      NaN            NaN  19:04  1.0   2.0
379  11600028              LAL      NaN            NaN  17:28  2.0   8.0
380  11600015              LAL      NaN            NaN  26:41  0.0   5.0

     FG_PCT  FG3M  FG3A  ...  REB  AST  STL  BLK   TO   PF   PTS  PLUS_MINUS  \
0     0.429   1.0   6.0  ...  4.0  5.0  0.0  0.0  5.0  2.0  19.0       -13.0
1     0.391   2.0   4.0  ...  5.0  4.0  0.0  0.0  1.0  1.0  31.0        -2.0
2     0.412   1.0   4.0  ...  3.0  4.0  0.0  1.0  4.0  3.0  19.0        10.0
3       NaN   NaN   NaN  ...  NaN  NaN  NaN  NaN  NaN  NaN   NaN         NaN
4       NaN   NaN   NaN  ...  NaN  NaN  NaN  NaN  NaN  NaN   NaN         NaN
..      ...   ...   ...  ...  ...  ...  ...  ...  ...  ...   ...         ...
376   0.429   1.0   4.0  ...  2.0  2.0  0.0  0.0  0.0  2.0   7.0       -27.0
377   0.333   1.0   3.0  ...  2.0  0.0  0.0  1.0  1.0  1.0   5.0         1.0
378   0.500   0.0   0.0  ...  2.0  0.0  1.0  3.0  0.0  1.0   4.0         0.0
379   0.250   1.0   2.0  ...  2.0  2.0  1.0  1.0  0.0  1.0   6.0        18.0
380   0.000   0.0   4.0  ...  2.0  1.0  1.0  2.0  1.0  2.0   2.0        20.0

     GAME_DATE_EST  SEASON
0       2021-11-17    2021
1       2021-11-15    2021
2       2021-11-13    2021
3       2021-11-08    2021
4       2021-11-05    2021
..             ...     ...
376     2016-10-13    2016
377     2016-10-11    2016
378     2016-10-09    2016
379     2016-10-07    2016
380     2016-10-04    2016

[381 rows x 26 columns]>
```

In [24]:

```
seasonal_stats = stats.groupby("SEASON").sum()/stats.groupby("SEASON").count()
```

In [25]:

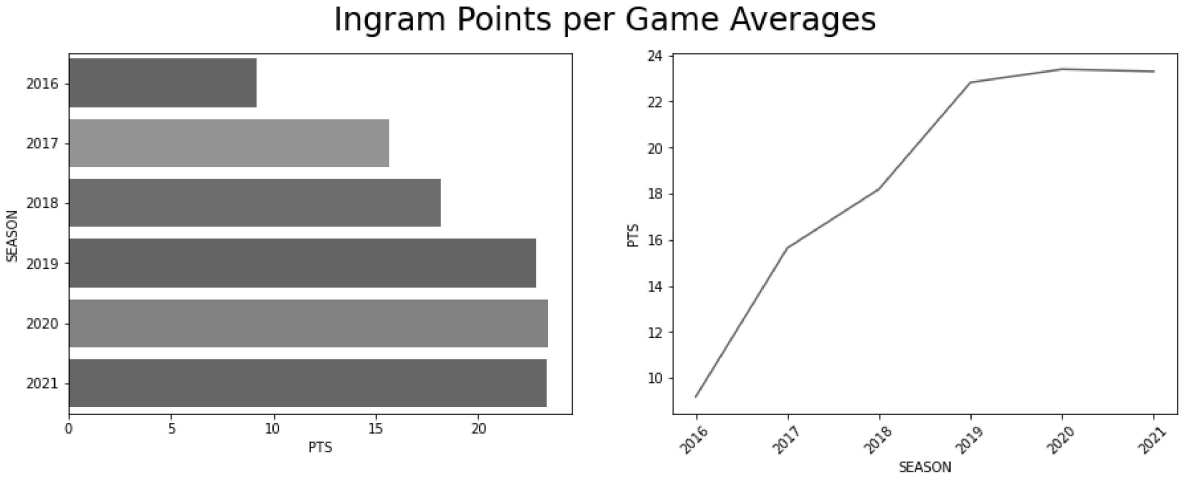
```
seasonal_stats.head
```

Out[25]:

```
<bound method NDFrame.head of              AST       BLK      DREB      FG3A      FG
3M   FG3_PCT        FGA  \
SEASON
2016    2.057471  0.505747  3.103448  2.356322  0.712644  0.255161   8.413793
2017    3.750000  0.703125  4.296875  1.828125  0.687500  0.264047  12.703125
2018    2.842105  0.596491  4.350877  1.771930  0.561404  0.259053  13.789474
2019    3.985507  0.565217  5.086957  5.913043  2.318841  0.394217  16.927536
2020    4.833333  0.590909  4.454545  6.060606  2.287879  0.351303  17.787879
2021    4.500000  0.400000  5.100000  4.800000  1.900000  0.367900  19.600000

             FGM    FG_PCT       FTA  ...  NICKNAME      OREB        PF  \
SEASON                                ...
2016    3.379310  0.387161  2.781609  ...       NaN  0.758621  1.942529
2017    5.890625  0.469609  4.656250  ...       NaN  0.937500  2.750000
2018    6.824561  0.499158  5.771930  ...       NaN  0.771930  2.842105
2019    7.840580  0.470072  5.681159  ...       NaN  0.811594  2.797101
2020    8.257576  0.466939  5.242424  ...       NaN  0.575758  2.015152
2021    8.900000  0.458600  4.200000  ...       NaN  0.600000  2.500000

        PLUS_MINUS        PTS       REB  START_POSITION       STL  \
SEASON
2016     -4.195402   9.183908  3.862069             NaN  0.643678
2017     -2.468750  15.640625  5.234375             NaN  0.765625
2018     -0.771930  18.192982  5.122807             NaN  0.614035
2019     -0.507246  22.826087  5.898551             NaN  0.942029
2020      0.712121  23.393939  5.030303             NaN  0.696970
2021     -5.700000  23.300000  5.700000             NaN  0.400000

        TEAM_ABBREVIATION        TO
SEASON
2016                  NaN  1.390805
2017                  NaN  2.531250
2018                  NaN  2.456140
2019                  NaN  2.927536
2020                  NaN  2.515152
2021                  NaN  3.500000

[6 rows x 25 columns]>
```

In [26]:

```
fig,axes = plt.subplots(1,2,figsize= (15,5))
fig.suptitle("Ingram Points per Game Averages",fontsize = 24)
sns.barplot(x=seasonal_stats["PTS"], y=seasonal_stats.index.map(str), ax=axes[0])
sns.lineplot(x=seasonal_stats.index.map(str), y=seasonal_stats["PTS"], ax=axes[1])
axes[1].tick_params(axis='x', labelrotation=45)
```



In [ ]: