DSBDAL Assignment 3 -Descriptive Statistics - Measures of Central Tendency and variability

Part 1

Data Preprocessing

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

In [2]:
```
df=pd.read_csv('nba.csv')
```

In [3]:
```
df.head()
```

Out[3]:

	Name	Team	Number	Position	Age	Height	Weight	College	Salary
0	Avery Bradley	Boston Celtics	0.0	PG	25.0	6-2	180.0	Texas	7730337.0
1	Jae Crowder	Boston Celtics	99.0	SF	25.0	6-6	235.0	Marquette	6796117.0
2	John Holland	Boston Celtics	30.0	SG	27.0	6-5	205.0	Boston University	NaN
3	R.J. Hunter	Boston Celtics	28.0	SG	22.0	6-5	185.0	Georgia State	1148640.0
4	Jonas Jerebko	Boston Celtics	8.0	PF	29.0	6-10	231.0	NaN	5000000.0

In [4]:
```
df.isnull().sum().sort_values(ascending=False)
```

Out[4]:
```
College     85
Salary      12
Name         1
Team         1
Number       1
Position     1
Age          1
Height       1
Weight       1
dtype: int64
```

In [5]:
```
print('Our data set contains {} rows and {} columns'.format(df.shape[0],df.sh
```

```
Our data set contains 458 rows and 9 columns
```

In [6]:
```
df=df[df['Name'].notnull()]
df["College"].fillna("No College",inplace=True)
```

In [7]:
```
df['Salary'] = df['Salary'].fillna(df.groupby('Team')['Salary'].transform('me
df['Salary'] = df['Salary'].fillna(df['Salary'].mean())
```

```
df
```

Out[7]:

	Name	Team	Number	Position	Age	Height	Weight	College	Salary
0	Avery Bradley	Boston Celtics	0.0	PG	25.0	6-2	180.0	Texas	7.730337e+06
1	Jae Crowder	Boston Celtics	99.0	SF	25.0	6-6	235.0	Marquette	6.796117e+06
2	John Holland	Boston Celtics	30.0	SG	27.0	6-5	205.0	Boston University	4.181505e+06
3	R.J. Hunter	Boston Celtics	28.0	SG	22.0	6-5	185.0	Georgia State	1.148640e+06
4	Jonas Jerebko	Boston Celtics	8.0	PF	29.0	6-10	231.0	No College	5.000000e+06
...	...	...	...	...	...	...	...	...	...
452	Trey Lyles	Utah Jazz	41.0	PF	20.0	6-10	234.0	Kentucky	2.239800e+06
453	Shelvin Mack	Utah Jazz	8.0	PG	26.0	6-3	203.0	Butler	2.433333e+06
454	Raul Neto	Utah Jazz	25.0	PG	24.0	6-1	179.0	No College	9.000000e+05
455	Tibor Pleiss	Utah Jazz	21.0	C	26.0	7-3	256.0	No College	2.900000e+06
456	Jeff Withey	Utah Jazz	24.0	C	26.0	7-0	231.0	Kansas	9.472760e+05

457 rows × 9 columns

In [8]:
```
df.isnull().sum().sort_values(ascending=False)
```

Out[8]:
```
Name        0
Team        0
Number      0
Position    0
Age         0
Height      0
Weight      0
College     0
Salary      0
dtype: int64
```

In [9]:
```
print('Our data set contains {} rows and {} columns'.format(df.shape[0],df.sh
```

```
Our data set contains 457 rows and 9 columns
```

In [10]:
```
# split the strings
df.Height = [s.split('-') for s in df.Height]
# convert to inches
df.Height = [float(value[0])*12 + float(value[1]) for value in df.Height]
```

In [11]:
```
df.Height
```

Out[11]:
```
0      74.0
1      78.0
```

```
2      77.0
3      77.0
4      82.0
       ...
452    82.0
453    75.0
454    73.0
455    87.0
456    84.0
Name: Height, Length: 457, dtype: float64
```

In [12]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 457 entries, 0 to 456
Data columns (total 9 columns):
 #   Column    Non-Null Count  Dtype  
---  ------    --------------  -----  
 0   Name      457 non-null    object 
 1   Team      457 non-null    object 
 2   Number    457 non-null    float64
 3   Position  457 non-null    object 
 4   Age       457 non-null    float64
 5   Height    457 non-null    float64
 6   Weight    457 non-null    float64
 7   College   457 non-null    object 
 8   Salary    457 non-null    float64
dtypes: float64(5), object(4)
memory usage: 35.7+ KB
```

In [13]:
```
df.dtypes
```

Out[13]:
```
Name          object
Team          object
Number       float64
Position      object
Age          float64
Height       float64
Weight       float64
College       object
Salary       float64
dtype: object
```

In [14]:
```
df = df.astype({'Number': 'int', 'Age': 'int','Weight':'int','Height':'int','
```

In [15]:
```
df[['Number','Age','Weight','Salary','Height']].dtypes
```

Out[15]:
```
Number    int64
Age       int64
Weight    int64
Salary    int64
Height    int64
dtype: object
```

In [16]:
```
df['Position'].value_counts()
```

Out[16]:
```
SG    102
PF    100
PG     92
SF     85
C      78
Name: Position, dtype: int64
```

In [17]:
```
df.describe()
```

Out[17]:

	Number	Age	Height	Weight	Salary
count	457.000000	457.000000	457.000000	457.000000	4.570000e+02
mean	17.678337	26.938731	79.190372	221.522976	4.851922e+06
std	15.966090	4.404016	3.432442	26.368343	5.170364e+06
min	0.000000	19.000000	69.000000	161.000000	3.088800e+04
25%	5.000000	24.000000	77.000000	200.000000	1.100602e+06
50%	13.000000	26.000000	80.000000	220.000000	2.854940e+06
75%	25.000000	30.000000	82.000000	240.000000	6.486486e+06
max	99.000000	40.000000	87.000000	307.000000	2.500000e+07

Salary Stats by grouping according to teams

In [18]:
```
df.groupby(["Team"])["Salary"].describe()
```

Out[18]:

Team	count	mean	std	min	25%	50%	
Atlanta Hawks	15.0	4.860197e+06	5.194508e+06	525093.0	1152260.00	2854940.0	68732
Boston Celtics	15.0	4.181505e+06	3.031593e+06	1148640.0	1994760.00	3425510.0	58980
Brooklyn Nets	15.0	3.501898e+06	5.317817e+06	134215.0	947276.00	1335480.0	25126
Charlotte Hornets	15.0	5.222728e+06	4.538601e+06	189455.0	1543138.00	4204200.0	66657
Chicago Bulls	15.0	5.785559e+06	6.251088e+06	525093.0	1203290.50	2380440.0	79743
Cleveland Cavaliers	15.0	7.642049e+06	7.449131e+06	111196.0	1211638.00	5000000.0	11624
Dallas Mavericks	15.0	4.746582e+06	5.030279e+06	525093.0	1185783.00	3950313.0	52894
Denver Nuggets	15.0	4.294424e+06	4.163062e+06	258489.0	1647099.50	3000000.0	4319
Detroit Pistons	15.0	4.477884e+06	4.668478e+06	111444.0	1711452.50	2891760.0	56350
Golden State Warriors	15.0	5.924600e+06	5.664282e+06	289755.0	1201462.00	3815000.0	11540
Houston Rockets	15.0	5.018868e+06	6.414749e+06	200600.0	973638.00	2288205.0	73397
Indiana Pacers	15.0	4.450122e+06	4.584514e+06	211744.0	1053513.00	4000000.0	5697
Los Angeles Clippers	15.0	6.323643e+06	7.600225e+06	111444.0	1024164.00	3110796.0	83675
Los Angeles Lakers	15.0	4.784695e+06	6.835688e+06	525093.0	896167.50	1724250.0	51611
Memphis Grizzlies	18.0	5.467920e+06	4.548734e+06	700902.0	1939075.00	5311269.5	54679

Team	count	mean	std	min	25%	50%	
Miami Heat	15.0	6.347359e+06	7.266418e+06	261894.0	947276.00	2854940.0	82494
Milwaukee Bucks	16.0	4.350220e+06	4.875071e+06	295327.0	1483589.00	2254167.0	55143
Minnesota Timberwolves	14.0	4.593054e+06	3.977223e+06	947276.0	1590540.00	3049180.5	57449
New Orleans Pelicans	19.0	4.355304e+06	4.537874e+06	55722.0	981348.50	2850000.0	77853
New York Knicks	16.0	4.581494e+06	5.952487e+06	30888.0	921721.75	2225421.0	49494
Oklahoma City Thunder	15.0	6.251020e+06	6.632400e+06	222888.0	1742280.00	3344000.0	86942
Orlando Magic	14.0	4.297248e+06	3.068412e+06	845059.0	2311302.00	3956580.0	51443
Philadelphia 76ers	15.0	2.213778e+06	1.831273e+06	525093.0	947276.00	1074169.0	31636
Phoenix Suns	15.0	4.229676e+06	5.022561e+06	55722.0	964312.00	2041080.0	55000
Portland Trail Blazers	15.0	3.220121e+06	2.392741e+06	525093.0	1181398.00	2854940.0	46261
Sacramento Kings	15.0	4.778911e+06	4.701792e+06	525093.0	998384.50	3156600.0	68803
San Antonio Spurs	15.0	5.629516e+06	6.396804e+06	200600.0	1045078.00	2814000.0	87500
Toronto Raptors	15.0	4.741174e+06	4.195943e+06	245177.0	1683000.00	2900000.0	66343
Utah Jazz	15.0	4.204006e+06	4.467878e+06	900000.0	1262160.00	2433333.0	42763
Washington Wizards	15.0	5.088576e+06	4.869388e+06	200600.0	1510421.00	4000000.0	68473

In [19]:
```
bins=[19,26,33,40]
labels=['19-26','26-33','33-40']
df['Age_Group']=pd.cut(df['Age'],bins=bins,labels=labels)
```

In [20]:
```
df.Age_Group.value_counts()
```

Out[20]:
```
19-26    233
26-33    180
33-40     42
Name: Age_Group, dtype: int64
```

Salary Stats by grouping according to age group

In [21]:
```
df.groupby(['Age_Group'])['Salary'].describe()
```

Out[21]:

Age_Group	count	mean	std	min	25%	50%	75%
19-26	233.0	3.641185e+06	4.269774e+06	30888.0	981348.00	1842000.0	4171680.0(

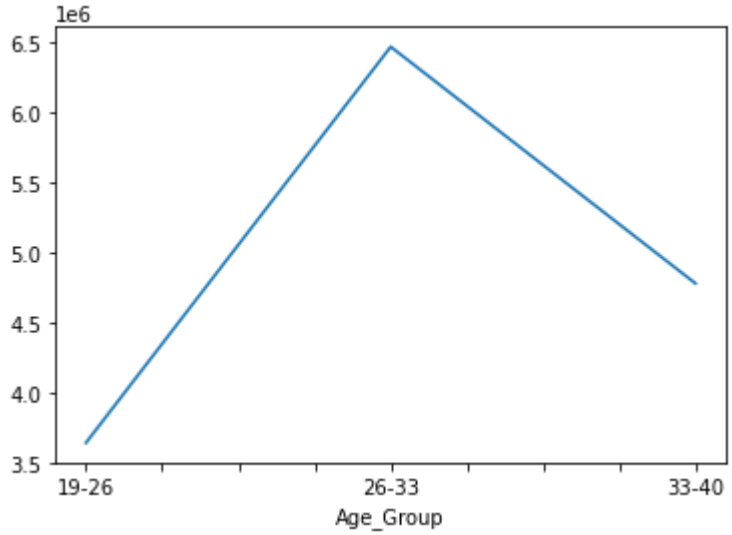
Age_Group	count	mean	std	min	25%	50%	75%
26-33	180.0	6.468612e+06	5.836236e+06	55722.0	1636500.50	4975000.0	9522106.5
33-40	42.0	4.779077e+06	5.022117e+06	222888.0	1073091.25	3459250.0	5728609.2

In [22]:
```
df.groupby(['Age_Group'])['Salary'].median()
```

Out[22]:
```
Age_Group
19-26    1842000
26-33    4975000
33-40    3459250
Name: Salary, dtype: int64
```

In [23]:
```
df.groupby(['Age_Group'])['Salary'].mean().plot()
```

Out[23]:
```
<AxesSubplot:xlabel='Age_Group'>
```



In [ ]:
```

```