Güzide Nur Demir
215150432
CS-240

Exploratory Data Analysis

Project Report

Section 1

I have found some questions which are related to dataet of NBA players and coaches. Questions are stated below:

***1*.** Is there a negative relationship between the number of losing match scores of coaches and the average of the assists produced by the players?

2. Does the time has positive effect on the games which are won by the coaches?

3. What is the characteristics of players that have points in the awards?

Between those three question, I will answer the first one. Reason of my choice is that I want to figure out the relation between number of games which are lost by coaches and assists by players affect negatively in a same time.

My hypothesis for this experiment is whether the average assists scores by players affect negatively to number of lost games by coaches or it depends on the chance.

The test statistic for this hypothesis testing, firstly I calculated the means of two dataets. One of them is mean of the lost games by coaches and another one is mean of the assists by the players. Lastly, I found the difference between these two means of dataets to get test statistic of the hypothesis.

Section 2

I have used two dataet which are “basketball_coaches.csv” and “basketball_players.csv”. Firstly, I defined variables of these two dataet and showed first 10 rows which are combined with some columns which are include coaches ID, years, team ID, assists, lost number etc.

```
In [1]: import pandas as pd
coaches_data=pd.read_csv("basketball_coaches.csv") #reading NBA coaches which is my first data set
coaches_data.head(10)
```

Out[1]:

	coachID	year	tmID	lgID	stint	won	lost	post_wins	post_losses
0	johnsne01	1961	PGR	ABL1	1	41.0	40.0	0.0	1.0
1	auerbre01	1946	WSC	NBA	1	49.0	11.0	2.0	4.0
2	birchpa01	1946	PIT	NBA	1	15.0	45.0	0.0	0.0
3	cliffro01	1946	CLR	NBA	2	13.0	10.0	1.0	2.0
4	cohalne01	1946	NYK	NBA	1	33.0	27.0	2.0	3.0
5	curtigl01	1946	DTF	NBA	1	12.0	22.0	0.0	0.0
6	dehnere01	1946	CLR	NBA	1	17.0	20.0	0.0	0.0
7	fitzgdi01	1946	TRH	NBA	1	2.0	1.0	0.0	0.0
8	gottled01	1946	PHW	NBA	1	35.0	25.0	8.0	2.0
9	haymale01	1946	TRH	NBA	3	0.0	1.0	0.0	0.0

```
In [40]: players_data=pd.read_csv("basketball_players.csv") #reading NBA players which is my second data set
players_data.head(10)
```

```
C:\ProgramData\Anaconda3\lib\site-packages\IPython\core\interactiveshell.py:2728: DtypeWarning: Columns (41) have mixed types.
Specify dtype option on import or set low_memory=False.
  interactivity=interactivity, compiler=compiler, result=result)
```

Out[40]:

	playerID	year	stint	tmID	lgID	GP	GS	minutes	points	oRebounds	...	PostBlocks	PostTurnovers	PostPF	PostfgAttempted	PostfgMade	PostftAtt
0	abramjo01	1946	1	PIT	NBA	47	0	0	527	0	...	0	0	0	0	0	
1	aubucch01	1946	1	DTF	NBA	30	0	0	65	0	...	0	0	0	0	0	
2	bakerno01	1946	1	CHS	NBA	4	0	0	0	0	...	0	0	0	0	0	
3	baltihe01	1946	1	STB	NBA	58	0	0	138	0	...	0	0	3	10	2	
4	barrjo01	1946	1	STB	NBA	58	0	0	295	0	...	0	0	0	0	0	
5	baumhfr01	1946	1	CLR	NBA	45	0	0	631	0	...	0	0	0	0	0	
6	beckemo01	1946	1	PIT	NBA	17	0	0	108	0	...	0	0	0	0	0	
7	beckemo01	1946	2	BOS	NBA	6	0	0	13	0	...	0	0	0	0	0	
8	beckemo01	1946	3	DTF	NBA	20	0	0	41	0	...	0	0	0	0	0	
9	beendha01	1946	1	PRO	NBA	58	0	0	713	0	...	0	0	0	0	0	

10 rows × 42 columns

I have planned to use specific columns to interpret my hypothesis testing. Therefore, I choose the coaches ID, years, team ID, assists, lost number columns from two dataets. Figure is illustrated below:

```
In [4]: players_data=players_data[["playerID", "year", "tmID",  "assists"]]
        players_data.head(10)
```

Out[4]:

	playerID	year	tmID	assists
0	abramjo01	1946	PIT	35
1	aubucch01	1946	DTF	20
2	bakerno01	1946	CHS	0
3	baltihe01	1946	STB	16
4	barrjo01	1946	STB	54
5	baumhfr01	1946	CLR	54
6	beckemo01	1946	PIT	14
7	beckemo01	1946	BOS	1
8	beckemo01	1946	DTF	15
9	beendha01	1946	PRO	37

Figure-1

```
In [3]: coaches_data=coaches_data[["coachID", "year", "tmID", "lost"]]
        coaches_data.head(10)
```

Out[3]:

	coachID	year	tmID	lost
0	johnsne01	1961	PGR	40.0
1	auerbre01	1946	WSC	11.0
2	birchpa01	1946	PIT	45.0
3	cliffro01	1946	CLR	10.0
4	cohalne01	1946	NYK	27.0
5	curtigl01	1946	DTF	22.0
6	dehnere01	1946	CLR	20.0
7	fitzgdi01	1946	TRH	1.0
8	gottled01	1946	PHW	25.0
9	haymale01	1946	TRH	1.0

Figure-2

In figure 1, I showed player and team ID, years and assists number from the player dataet. In figure 2, I eliminated and figured out that coaches and team ID, year and lost numbers. After these steps, I calculated the mean of assists by grouping tmID and year data to organize size of data.

```
In [41]: #finding avarage of assists by groupping year and team id
assists_data = players_data.groupby(["year", "tmID"]).agg({"tmID": "min", "year": "min", "assists": "mean"})
assists_data.head(15)
```

Out[41]:

year	tmID	tmID	year	assists
1937	AFS	AFS	1937	0.0
	AGW	AGW	1937	0.0
	BFB	BFB	1937	0.0
	CNC	CNC	1937	0.0
	COL	COL	1937	0.0
	DYM	DYM	1937	0.0
	FWE	FWE	1937	0.0
	INK	INK	1937	0.0
	KNK	KNK	1937	0.0
	OSH	OSH	1937	0.0
	PGP	PGP	1937	0.0
	WAR	WAR	1937	0.0
	WHT	WHT	1937	0.0
1938	AFS	AFS	1938	0.0
	AGW	AGW	1938	0.0

Finally, I showed the information about assists and lost data. These information include means, counts, min value etc of the these two data.

Section 3

In that section, before the I plot visualization, I merged dataframes with multiple columns with inner method that considers intersections.

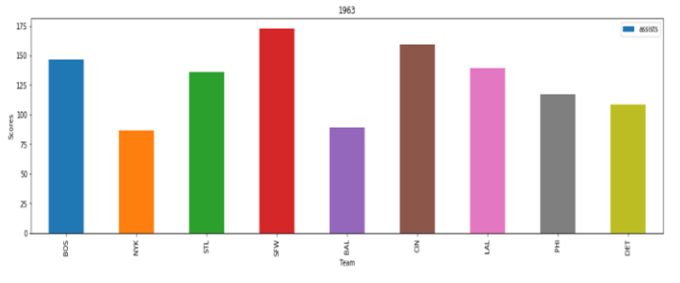
```
In [10]: #merging dataframes with multiple columns by inner method that considers intersections
merged_data = pd.merge(coaches_data, assists_data, how="inner", left_on=["year", "tmID"], right_on=["year", "tmID"])
merged_data.head(10)
```

Out[10]:

	coachID	year	tmID	lost	assists
0	johnsne01	1961	PGR	40.0	59.700000
1	auerbre01	1946	WSC	11.0	27.000000
2	birchpa01	1946	PIT	45.0	16.000000
3	cliffro01	1946	CLR	10.0	27.444444
4	dehnere01	1946	CLR	20.0	27.444444
5	cohalne01	1946	NYK	27.0	22.850000
6	curtigl01	1946	DTF	22.0	32.133333
7	sachsph01	1946	DTF	18.0	32.133333
8	fitzgdi01	1946	TRH	1.0	23.100000
9	haymale01	1946	TRH	1.0	23.100000

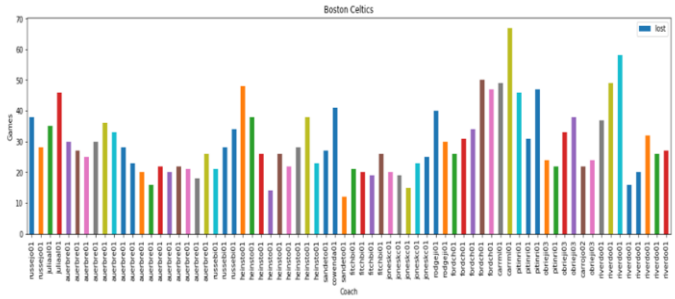
After the merging data, I plotted specific data and their histogram, pdf and cdf.

```
In [13]: #drawing bar figure for teams and assists which are scored in 1963
pylab.rcParams["figure.figsize"] = (20, 5)
merged_data[merged_data.year == 1963].plot(kind="bar", x="tmID", y="assists")
plt.title("1963")
plt.xlabel("Team")
plt.ylabel("Scores")
plt.show()
```



This figure shows the teams and players' assists which are played in 1963.

```
In [38]: #drawing bar that shows coaches and games which are lost for Boston Celtics
pylab.rcParams["figure.figsize"] = (20, 5)
merged_data[merged_data.tmID == "BOS"].plot(kind="bar", x="coachID", y="lost")
plt.title("Boston Celtics")
plt.xlabel("Coach")
plt.ylabel("Games")
plt.show()
```



This figure illustrates that coaches and games which are lost for Boston Celtics.

```
In [39]: #drawing bar that shows assists for New York Knick scored per year
pylab.rcParams["figure.figsize"] = (20, 5)
merged_data[merged_data.tmID == "NYK"].plot(kind="scatter", x="year", y="assists")
plt.title("New York Knick")
plt.xlabel("Years")
plt.ylabel("Scores")
plt.show()
```

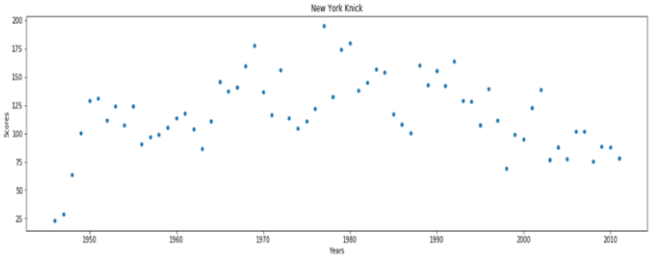
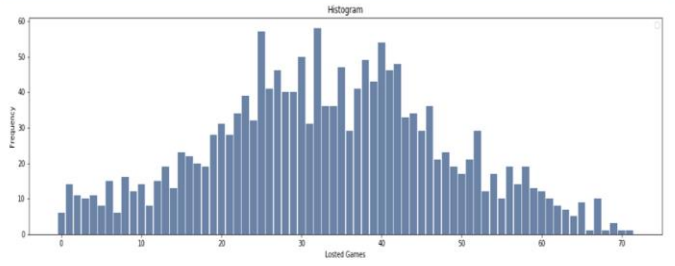


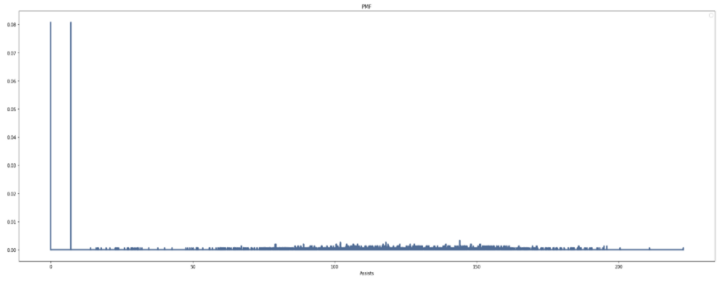
Figure plotted by “scatter” method and it shows that assists for New York Knicks scored per year.

Histogram, CDF and PDF of these datasets given below:

```
[22]: import thinkstats2, thinkplot
#visualizing frequencies of number of losted games
coaches_hist = thinkstats2.Hist(coaches_data.lost)
thinkplot.Hist(coaches_hist)
thinkplot.Config(title="Histogram", xlabel="Losted Games", ylabel="Frequency")
No handles with labels found to put in legend.
```




```
: #visualizing PMF of means of assists which are scored by players
pmf_assists = thinkstats2.Pmf(assists_data.assists)
pylab.rcParams["figure.figsize"] = (30, 10)
thinkplot.Pmf(pmf_assists)
thinkplot.Config(title="PMF", xlabel="Assists")
No handles with labels found to put in legend.
```



Histogram

I visualize the frequency of number of games which are lost by coaches. By using thinstats2 module and

PDF

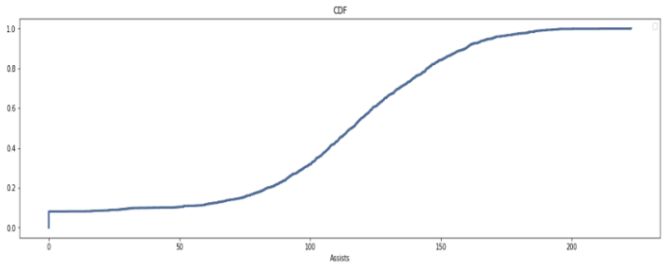
I visualize the Probability Mass Function of average of assists number scored by

Hist() funcion, I plotted the histogram.

using Pmf() function and thinkstats2 module.

```
n [43]: #visualizing CDF of means of assists which are scored by players
cdf_assists = thinkstats2.Cdf(assists_data.assists)
pylab.rcParams["figure.figsize"] = (20, 5)
thinkplot.Cdf(cdf_assists)
thinkplot.Config(title="CDF", xlabel="Assists")
```

```
No handles with labels found to put in legend.
```



I visualize CDF of means of assists which are scored by players.

CDF

Section 4

In this part, I applied the least squares method regression to calculate the goodness of fit. I found the summary of the dataset then found a table. From that table I got the R squared result because to understand whether line fits data or not, we need to know R squared.

Out[44]:

OLS Regression Results

Dep. Variable:	lost	**R-squared:**	0.013
Model:	OLS	**Adj. R-squared:**	0.013
Method:	Least Squares	**F-statistic:**	22.29
Date:	Sun, 03 Jun 2018	**Prob (F-statistic):**	2.54e-06
Time:	22:27:39	**Log-Likelihood:**	-6889.6
No. Observations:	1680	**AIC:**	1.378e+04
Df Residuals:	1678	**BIC:**	1.379e+04
Df Model:	1		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	27.7144	1.233	22.476	0.000	25.296	30.133
assists	0.0483	0.010	4.721	0.000	0.028	0.068

Omnibus:	18.677	**Durbin-Watson:**	1.819
Prob(Omnibus):	0.000	**Jarque-Bera (JB):**	13.253
Skew:	0.097	**Prob(JB):**	0.00132
Kurtosis:	2.611	**Cond. No.**	417.

As it can be seen in the figure above, R squared is 1,3%. Due to the fact that it is greater than 1% threshold. Thus, it means that line fits data.

Section 5

```
7]: #calculating correlation of losted games and assi
    cols = merged_data[["lost", "assists"]]
    cols.corr()
```

7]:

	lost	assists
lost	1.000000	0.114497
assists	0.114497	1.000000

In the fifth part of this project, I calculated the correlation between two data sets which are we interested in.

Section 6

In the last part, I calculated difference of means of assists and merged data to find Test Statistic. After the calculating Test Statistic, I considered the summary table which is showed above and clarified that the p value of this statistic is closed to 0 and smaller than threshold. Thus, it is statistically significant. Meaning of this result is that it does not occur by a chance.

```
In [18]: merged_data_mean=merged_data["assists"].mean()
         merged_data_mean
Out[18]: 114.98199125993246

In [19]: assists_data_mean=assists_data["assists"].mean()
         assists_data_mean
Out[19]: 109.3757147892138

In [20]: #calculating the difference of means
         abs(assists_data_mean - merged_data_mean)
Out[20]: 5.6062764707186545
```

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	27.7144	1.233	22.476	0.000	25.296	30.133
assists	0.0483	0.010	4.721	0.000	0.028	0.068

Omnibus:	18.677	Durbin-Watson:	1.819
Prob(Omnibus):	0.000	Jarque-Bera (JB):	13.253
Skew:	0.097	Prob(JB):	0.00132
Kurtosis:	2.611	Cond. No.	417.

Section 7

To conclude, I want to clarify whether number of game which are lost by coaches has negative relationship with the average assists by the players or not. In this project, after some calculation to reach the result of the question, my hypothesis illustrates that the average of the assists by the players are not negatively related to number of games which are lost by coaches in same years.