.ipynb

September 4, 2025

1 English Premier League (EPL) Pythagorean Predictor

1.1 Step 1

In [2]:
```
# Load the packages

import pandas as pd
import numpy as np
import statsmodels.formula.api as smf
import matplotlib.pyplot as plt
import seaborn as sns
```

In [3]:
```
# Load the data.
# EPL results for the 2017/18 season

EPL18 = pd.read_excel('Assignment Data/Week 1/EPL2017-18.xlsx')
print(EPL18.columns.tolist())
```

```
['Date', 'HomeTeam', 'AwayTeam', 'FTHG', 'FTAG', 'FTR']
```

1.2 Step 2

In [10]:
```
#Create a value for a home wins (win= 1, draw=0.5, loss= 0) and away wins and a count

EPL18['hwinvalue']=np.where(EPL18['FTR']=='H',1,np.where(EPL18['FTR']=='D',.5,0))
EPL18['awinvalue']=np.where(EPL18['FTR']=='A',1,np.where(EPL18['FTR']=='D',.5,0))
EPL18['count']=1

EPL18
```

Out[10]:

	Date	HomeTeam	AwayTeam	FTHG	FTAG	FTR	hwinvalue	\
0	20170811	Arsenal	Leicester	4	3	H	1.0	
1	20170812	Brighton	Man City	0	2	A	0.0	
2	20170812	Chelsea	Burnley	2	3	A	0.0	
3	20170812	Crystal Palace	Huddersfield	0	3	A	0.0	
4	20170812	Everton	Stoke	1	0	H	1.0	
5	20170812	Southampton	Swansea	0	0	D	0.5	

6	20170812	Watford	Liverpool	3	3	D	0.5
7	20170812	West Brom	Bournemouth	1	0	H	1.0
8	20170813	Man United	West Ham	4	0	H	1.0
9	20170813	Newcastle	Tottenham	0	2	A	0.0
10	20170819	Bournemouth	Watford	0	2	A	0.0
11	20170819	Burnley	West Brom	0	1	A	0.0
12	20170819	Leicester	Brighton	2	0	H	1.0
13	20170819	Liverpool	Crystal Palace	1	0	H	1.0
14	20170819	Southampton	West Ham	3	2	H	1.0
15	20170819	Stoke	Arsenal	1	0	H	1.0
16	20170819	Swansea	Man United	0	4	A	0.0
17	20170820	Huddersfield	Newcastle	1	0	H	1.0
18	20170820	Tottenham	Chelsea	1	2	A	0.0
19	20170821	Man City	Everton	1	1	D	0.5
20	20170826	Bournemouth	Man City	1	2	A	0.0
21	20170826	Crystal Palace	Swansea	0	2	A	0.0
22	20170826	Huddersfield	Southampton	0	0	D	0.5
23	20170826	Man United	Leicester	2	0	H	1.0
24	20170826	Newcastle	West Ham	3	0	H	1.0
25	20170826	Watford	Brighton	0	0	D	0.5
26	20170827	Chelsea	Everton	2	0	H	1.0
27	20170827	Liverpool	Arsenal	4	0	H	1.0
28	20170827	Tottenham	Burnley	1	1	D	0.5
29	20170827	West Brom	Stoke	1	1	D	0.5
..	...	...	...	...	...	..	...
350	20180428	Swansea	Chelsea	0	1	A	0.0
351	20180429	Man United	Arsenal	2	1	H	1.0
352	20180429	West Ham	Man City	1	4	A	0.0
353	20180430	Tottenham	Watford	2	0	H	1.0
354	20180504	Brighton	Man United	1	0	H	1.0
355	20180505	Bournemouth	Swansea	1	0	H	1.0
356	20180505	Everton	Southampton	1	1	D	0.5
357	20180505	Leicester	West Ham	0	2	A	0.0
358	20180505	Stoke	Crystal Palace	1	2	A	0.0
359	20180505	Watford	Newcastle	2	1	H	1.0
360	20180505	West Brom	Tottenham	1	0	H	1.0
361	20180506	Arsenal	Burnley	5	0	H	1.0
362	20180506	Chelsea	Liverpool	1	0	H	1.0
363	20180506	Man City	Huddersfield	0	0	D	0.5
364	20180508	Swansea	Southampton	0	1	A	0.0
365	20180509	Chelsea	Huddersfield	1	1	D	0.5
366	20180509	Leicester	Arsenal	3	1	H	1.0
367	20180509	Man City	Brighton	3	1	H	1.0
368	20180509	Tottenham	Newcastle	1	0	H	1.0
369	20180510	West Ham	Man United	0	0	D	0.5
370	20180513	Burnley	Bournemouth	1	2	A	0.0
371	20180513	Crystal Palace	West Brom	2	0	H	1.0
372	20180513	Huddersfield	Arsenal	0	1	A	0.0

373	20180513	Liverpool	Brighton	4	0	H	1.0
374	20180513	Man United	Watford	1	0	H	1.0
375	20180513	Newcastle	Chelsea	3	0	H	1.0
376	20180513	Southampton	Man City	0	1	A	0.0
377	20180513	Swansea	Stoke	1	2	A	0.0
378	20180513	Tottenham	Leicester	5	4	H	1.0
379	20180513	West Ham	Everton	3	1	H	1.0

	awinvalue	count
0	0.0	1
1	1.0	1
2	1.0	1
3	1.0	1
4	0.0	1
5	0.5	1
6	0.5	1
7	0.0	1
8	0.0	1
9	1.0	1
10	1.0	1
11	1.0	1
12	0.0	1
13	0.0	1
14	0.0	1
15	0.0	1
16	1.0	1
17	0.0	1
18	1.0	1
19	0.5	1
20	1.0	1
21	1.0	1
22	0.5	1
23	0.0	1
24	0.0	1
25	0.5	1
26	0.0	1
27	0.0	1
28	0.5	1
29	0.5	1
..	...	...
350	1.0	1
351	0.0	1
352	1.0	1
353	0.0	1
354	0.0	1
355	0.0	1
356	0.5	1
357	1.0	1

```
358        1.0      1
359        0.0      1
360        0.0      1
361        0.0      1
362        0.0      1
363        0.5      1
364        1.0      1
365        0.5      1
366        0.0      1
367        0.0      1
368        0.0      1
369        0.5      1
370        1.0      1
371        0.0      1
372        1.0      1
373        0.0      1
374        0.0      1
375        0.0      1
376        1.0      1
377        1.0      1
378        0.0      1
379        0.0      1

[380 rows x 9 columns]
```

1.3 Step 3

```
In [13]: #Create a file for games played in 2017 (before date 20180000) and another one for ga

         Gin2017 =EPL18[EPL18.Date <20180000]
         Gin2017.describe()
```

Out[13]:

	Date	FTHG	FTAG	hwinvalue	awinvalue	count
count	2.090000e+02	209.000000	209.000000	209.000000	209.000000	209.0
mean	2.017106e+07	1.473684	1.181818	0.574163	0.425837	1.0
std	1.451426e+02	1.362452	1.273039	0.422336	0.422336	0.0
min	2.017081e+07	0.000000	0.000000	0.000000	0.000000	1.0
25%	2.017092e+07	0.000000	0.000000	0.000000	0.000000	1.0
50%	2.017110e+07	1.000000	1.000000	0.500000	0.500000	1.0
75%	2.017121e+07	2.000000	2.000000	1.000000	1.000000	1.0
max	2.017123e+07	7.000000	6.000000	1.000000	1.000000	1.0

1.4 Step 4 (home team)

```
In [15]: print(EPL18.columns)
```

```
Index(['Date', 'HomeTeam', 'AwayTeam', 'FTHG', 'FTAG', 'FTR', 'hwinvalue',
       'awinvalue', 'count'],
```

```
      dtype='object')
```

```
In [19]: #For the 2017 games, use .groupby to create a dataframe aggregating by home team the

         EPLHome = EPL18.groupby('HomeTeam')['count','hwinvalue', 'FTHG','FTAG'].sum().reset_in
         EPLHome
```

Out[19]:

	HomeTeam	count	hwinvalue	FTHG	FTAG
0	Arsenal	19	16.0	54	20
1	Bournemouth	19	9.5	26	30
2	Brighton	19	11.0	24	25
3	Burnley	19	9.5	16	17
4	Chelsea	19	13.0	30	16
5	Crystal Palace	19	9.5	29	27
6	Everton	19	12.0	28	22
7	Huddersfield	19	8.5	16	25
8	Leicester	19	10.0	25	22
9	Liverpool	19	15.5	45	10
10	Man City	19	17.0	61	14
11	Man United	19	16.0	38	9
12	Newcastle	19	10.0	21	17
13	Southampton	19	7.5	20	26
14	Stoke	19	7.5	20	30
15	Swansea	19	7.5	17	24
16	Tottenham	19	15.0	40	16
17	Watford	19	10.0	27	31
18	West Brom	19	7.5	21	29
19	West Ham	19	10.0	24	26

1.5 Step 5 (home team)

```
In [22]: #Rename the variables to denote whether they are aggregates for home team or away tea

         EPLHome = EPLHome.rename(columns={'HomeTeam':'team','count':'Ph','FTHG':'FTHGh','FTAG
         EPLHome
```

Out[22]:

	team	Ph	hwinvalue	FTHGh	FTAGh
0	Arsenal	19	16.0	54	20
1	Bournemouth	19	9.5	26	30
2	Brighton	19	11.0	24	25
3	Burnley	19	9.5	16	17
4	Chelsea	19	13.0	30	16
5	Crystal Palace	19	9.5	29	27
6	Everton	19	12.0	28	22
7	Huddersfield	19	8.5	16	25
8	Leicester	19	10.0	25	22
9	Liverpool	19	15.5	45	10
10	Man City	19	17.0	61	14

11	Man United	19	16.0	38	9
12	Newcastle	19	10.0	21	17
13	Southampton	19	7.5	20	26
14	Stoke	19	7.5	20	30
15	Swansea	19	7.5	17	24
16	Tottenham	19	15.0	40	16
17	Watford	19	10.0	27	31
18	West Brom	19	7.5	21	29
19	West Ham	19	10.0	24	26

1.6 Step 6 (home team)

In [ ]:

1.7 Optional steps, not required for Assessment

1.7.1 (Uncomment to run)

In [4]:
```
# Run the regression

pyth_lm = smf.ols(formula = 'wpc17 ~ pyth17', data=EPL17).fit()
pyth_lm.summary()
```

```
---------------------------------------------------------------------------

NameError                                 Traceback (most recent call last)

<ipython-input-4-c917f04aabbe> in <module>
  1 # Run the regression
  2
----> 3 pyth_lm = smf.ols(formula = 'wpc17 ~ pyth17', data=EPL17).fit()
  4 pyth_lm.summary()


NameError: name 'EPL17' is not defined
```

1.8 Step 7 (=Step 4 (away team))

In [21]:
```
#For the 2017 games, use .groupby to create a dataframe aggregating by away team the

EPLAway = EPL18.groupby('AwayTeam')['count','awinvalue','FTAG','FTHG'].sum().reset_in
EPLAway
```

Out[21]:

	AwayTeam	count	awinvalue	FTAG	FTHG
0	Arsenal	19	6.0	20	31
1	Bournemouth	19	7.0	19	31
2	Brighton	19	4.5	10	29

3	Burnley	19	10.5	20	22
4	Chelsea	19	11.5	32	22
5	Crystal Palace	19	7.0	16	28
6	Everton	19	6.0	16	36
7	Huddersfield	19	5.5	12	33
8	Leicester	19	7.5	31	38
9	Liverpool	19	11.5	39	28
10	Man City	19	17.0	45	13
11	Man United	19	12.0	30	19
12	Newcastle	19	6.0	18	30
13	Southampton	19	7.0	17	30
14	Stoke	19	5.5	15	38
15	Swansea	19	5.0	11	32
16	Tottenham	19	12.0	34	20
17	Watford	19	5.0	17	33
18	West Brom	19	5.0	10	27
19	West Ham	19	6.0	24	42

1.9 Step 7 (=Step 5 (away team))

In [23]:
```
#Rename the variables to denote whether they are aggregates for home team or away tea

EPLAway = EPLAway.rename(columns={'AwayTeam':'team','count':'Ph','FTAG':'FTAGh','FTHG
EPLAway
```

Out[23]:

	team	Ph	awinvalue	FTAGh	FTHGh
0	Arsenal	19	6.0	20	31
1	Bournemouth	19	7.0	19	31
2	Brighton	19	4.5	10	29
3	Burnley	19	10.5	20	22
4	Chelsea	19	11.5	32	22
5	Crystal Palace	19	7.0	16	28
6	Everton	19	6.0	16	36
7	Huddersfield	19	5.5	12	33
8	Leicester	19	7.5	31	38
9	Liverpool	19	11.5	39	28
10	Man City	19	17.0	45	13
11	Man United	19	12.0	30	19
12	Newcastle	19	6.0	18	30
13	Southampton	19	7.0	17	30
14	Stoke	19	5.5	15	38
15	Swansea	19	5.0	11	32
16	Tottenham	19	12.0	34	20
17	Watford	19	5.0	17	33
18	West Brom	19	5.0	10	27
19	West Ham	19	6.0	24	42

1.10 Step 7 (=Step 6 (away team))

In [ ]:

1.11 Optional steps, not required for Assessment

1.11.1 (Uncomment to run)

In [ ]:
```
# Plot the data

#sns.relplot(x="pyth18", y="wpc18", data =EPL2018)
```

In [ ]:
```
# Run the regression

#pyth_lm = smf.ols(formula = 'wpc18 ~ pyth18', data=EPL2018).fit()
#pyth_lm.summary()
```

1.12 Step 8

In [ ]:

1.13 Step 9

In [ ]:

1.14 Optional steps, not required for Assessment

1.14.1 (Uncomment to run)

In [ ]:
```
#sns.relplot(x="pyth17", y="wpc18", data =Half2predictor)
```

In [ ]:
```
#sns.relplot(x="wpc17", y="wpc18", data =Half2predictor)
```

Now you have completed the assignment, are these results consistent with those we found for Major League Baseball?