# 1 Pythagorean expectation and MLB

September 4, 2025

## 1 Getting Started

In this series of MOOCs we aim to introduce participants to methods for analyzing sports data using Python. In this first MOOC we introduce some basic concepts. These can be broken down into three areas:

1. How to code sports data so that you can apply statistical methods
2. The use of statistical methods
3. The interpretation of results

As we go along we will introduce you to the concepts by analyzing data from different sports and generating results. Once you get the hang of how this works, you'll Pythagorean be able to do it for yourself.

In this first week, we're going to go through simple but powerful examples that introduce you to all three elements.

## 2 The Pythagorean Expectation

The Pythagorean expectation is an idea devised by the famous baseball analyst, Bill James, but it can in fact be applied to any sport.

In any sports league, teams win games by accumulating a higher total than opponent. In baseball and cricket the relevant totals are runs, in basketball it is points, and in soccer and hockey it is goals (by "hockey" we mean here what the world outside of the US and Canada usually calls ice hockey, but in fact the same is true in field hockey).

The Pythagorean expectation can be described thus: in any season, the percentage of games won will be proportional to the square of total runs/points/goals scored by the team *squared* divided by the sum of total runs/points/goals scored by the team *squared* plus total runs/points/goals conceded by the team *squared*.

or *wpc = TF2 / ( TF2 + TA2 )*

Where TF is runs/points/goals scored and TA is runs/points/goals conceded.

This is a concept which can help to explain not only why teams are successful, but also can be used as the basis for predicting results in the future.

In this first week we are going to derive the Pythagorean expectation for five leagues in five different sports:

Major League Baseball The English Premier League (soccer) The Indian Premier League (cricket) The National Basketball Association (NBA) The National Hockey League (NHL)

### 2.1 Coding the data

To derive the Pythagorean Expectation we will need to manipulate the data, which is a core skill that we expect you to obtain from these MOOCs. However, for this first week, we move quite quickly through the code, since our main objective is to show you the kinds of analysis you will be able to produce once you master Python.

### 2.2 The Pythagorean Expectation for baseball

We begin, naturally enough, with baseball. Running code in Python typically involves the following steps:

1. Importing "packages" - these enable to run certain types of commands. The same ones come up over and over again - pandas, numpy, matplotlib.pyplot and so on.
2. Import the raw data - from a csv or excel file - in these MOOCs we will provide the data for you
3. Running commands to shape the data in preparation for running the statistical model
4. Running the statistical model
5. Reviewing the results

With each line of code below, there is a brief explanation of the code. When you are ready, read each line, then place the cursor on the relevant line and press "run" in the toolbar.

```
In [7]: # Here are the packages we need

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

```
In [8]: # This command imports our data, which is a log of games played in 2018 doenloaded fro
        #(you can find the data here: https://www.retrosheet.org/)
        # the second line of the command prints a list of variable names - there are many more

        MLB = pd.read_excel('../../Data/Week 1/Retrosheet MLB game log 2018.xlsx')
        print(MLB.columns.tolist())
```

```
['Date', 'DoubleHeader', 'DayOfWeek', 'VisitingTeam', 'VisitingTeamLeague', 'VisitingTeamGameNu
```

```
In [9]: # We can see what our dataframe looks like simply by typing its name

        MLB
```

Out[9]:

| | Date | DoubleHeader | DayOfWeek | VisitingTeam | VisitingTeamLeague | \ |
|---|---|---|---|---|---|---|
| 0 | 20180329 | 0 | Thu | COL | NL | |
| 1 | 20180329 | 0 | Thu | PHI | NL | |
| 2 | 20180329 | 0 | Thu | SFN | NL | |
| 3 | 20180329 | 0 | Thu | CHN | NL | |
| 4 | 20180329 | 0 | Thu | SLN | NL | |
| 5 | 20180329 | 0 | Thu | MIL | NL | |
| 6 | 20180329 | 0 | Thu | MIN | AL | |
| 7 | 20180329 | 0 | Thu | CHA | AL | |
| 8 | 20180329 | 0 | Thu | ANA | AL | |
| 9 | 20180329 | 0 | Thu | CLE | AL | |
| 10 | 20180329 | 0 | Thu | BOS | AL | |
| 11 | 20180329 | 0 | Thu | HOU | AL | |
| 12 | 20180329 | 0 | Thu | NYA | AL | |
| 13 | 20180330 | 0 | Fri | COL | NL | |
| 14 | 20180330 | 0 | Fri | PHI | NL | |
| 15 | 20180330 | 0 | Fri | WAS | NL | |
| 16 | 20180330 | 0 | Fri | SFN | NL | |
| 17 | 20180330 | 0 | Fri | CHN | NL | |
| 18 | 20180330 | 0 | Fri | MIL | NL | |
| 19 | 20180330 | 0 | Fri | PIT | NL | |
| 20 | 20180330 | 0 | Fri | ANA | AL | |
| 21 | 20180330 | 0 | Fri | BOS | AL | |
| 22 | 20180330 | 0 | Fri | HOU | AL | |
| 23 | 20180330 | 0 | Fri | NYA | AL | |
| 24 | 20180331 | 0 | Sat | COL | NL | |
| 25 | 20180331 | 0 | Sat | PHI | NL | |
| 26 | 20180331 | 0 | Sat | WAS | NL | |
| 27 | 20180331 | 0 | Sat | SFN | NL | |
| 28 | 20180331 | 0 | Sat | CHN | NL | |
| 29 | 20180331 | 0 | Sat | SLN | NL | |
| ... | ... | ... | ... | ... | ... | |
| 2401 | 20180929 | 0 | Sat | DET | AL | |
| 2402 | 20180929 | 0 | Sat | MIA | NL | |
| 2403 | 20180929 | 0 | Sat | ATL | NL | |
| 2404 | 20180929 | 0 | Sat | ARI | NL | |
| 2405 | 20180929 | 0 | Sat | LAN | NL | |
| 2406 | 20180929 | 0 | Sat | OAK | AL | |
| 2407 | 20180929 | 1 | Sat | HOU | AL | |
| 2408 | 20180929 | 2 | Sat | HOU | AL | |
| 2409 | 20180929 | 0 | Sat | NYA | AL | |
| 2410 | 20180929 | 0 | Sat | CLE | AL | |
| 2411 | 20180929 | 0 | Sat | CHA | AL | |
| 2412 | 20180929 | 0 | Sat | TEX | AL | |
| 2413 | 20180929 | 0 | Sat | TOR | AL | |
| 2414 | 20180930 | 0 | Sun | SLN | NL | |
| 2415 | 20180930 | 0 | Sun | PIT | NL | |
| 2416 | 20180930 | 0 | Sun | WAS | NL | |

| | | | | | |
|---|---|---|---|---|---|
| 2417 | 20180930 | 0 | Sun | DET | AL |
| 2418 | 20180930 | 0 | Sun | MIA | NL |
| 2419 | 20180930 | 0 | Sun | ATL | NL |
| 2420 | 20180930 | 0 | Sun | ARI | NL |
| 2421 | 20180930 | 0 | Sun | LAN | NL |
| 2422 | 20180930 | 0 | Sun | OAK | AL |
| 2423 | 20180930 | 0 | Sun | HOU | AL |
| 2424 | 20180930 | 0 | Sun | NYA | AL |
| 2425 | 20180930 | 0 | Sun | CLE | AL |
| 2426 | 20180930 | 0 | Sun | CHA | AL |
| 2427 | 20180930 | 0 | Sun | TEX | AL |
| 2428 | 20180930 | 0 | Sun | TOR | AL |
| 2429 | 20181001 | 0 | Mon | MIL | NL |
| 2430 | 20181001 | 0 | Mon | COL | NL |

| | VisitingTeamGameNumber | HomeTeam | HomeTeamLeague | HomeTeamGameNumber \ |
|---|---|---|---|---|
| 0 | 1 | ARI | NL | 1 |
| 1 | 1 | ATL | NL | 1 |
| 2 | 1 | LAN | NL | 1 |
| 3 | 1 | MIA | NL | 1 |
| 4 | 1 | NYN | NL | 1 |
| 5 | 1 | SDN | NL | 1 |
| 6 | 1 | BAL | AL | 1 |
| 7 | 1 | KCA | AL | 1 |
| 8 | 1 | OAK | AL | 1 |
| 9 | 1 | SEA | AL | 1 |
| 10 | 1 | TBA | AL | 1 |
| 11 | 1 | TEX | AL | 1 |
| 12 | 1 | TOR | AL | 1 |
| 13 | 2 | ARI | NL | 2 |
| 14 | 2 | ATL | NL | 2 |
| 15 | 1 | CIN | NL | 1 |
| 16 | 2 | LAN | NL | 2 |
| 17 | 2 | MIA | NL | 2 |
| 18 | 2 | SDN | NL | 2 |
| 19 | 1 | DET | AL | 1 |
| 20 | 2 | OAK | AL | 2 |
| 21 | 2 | TBA | AL | 2 |
| 22 | 2 | TEX | AL | 2 |
| 23 | 2 | TOR | AL | 2 |
| 24 | 3 | ARI | NL | 3 |
| 25 | 3 | ATL | NL | 3 |
| 26 | 2 | CIN | NL | 2 |
| 27 | 3 | LAN | NL | 3 |
| 28 | 3 | MIA | NL | 3 |
| 29 | 2 | NYN | NL | 2 |
| ... | ... | ... | ... | ... |
| 2401 | 161 | MIL | NL | 161 |

```
2402                  160     NYN            NL                  161
2403                  161     PHI            NL                  161
2404                  161     SDN            NL                  161
2405                  161     SFN            NL                  161
2406                  161     ANA            AL                  161
2407                  160     BAL            AL                  160
2408                  161     BAL            AL                  161
2409                  161     BOS            AL                  161
2410                  161     KCA            AL                  161
2411                  161     MIN            AL                  161
2412                  161     SEA            AL                  161
2413                  161     TBA            AL                  161
2414                  162     CHN            NL                  162
2415                  161     CIN            NL                  162
2416                  162     COL            NL                  162
2417                  162     MIL            NL                  162
2418                  161     NYN            NL                  162
2419                  162     PHI            NL                  162
2420                  162     SDN            NL                  162
2421                  162     SFN            NL                  162
2422                  162     ANA            AL                  162
2423                  162     BAL            AL                  162
2424                  162     BOS            AL                  162
2425                  162     KCA            AL                  162
2426                  162     MIN            AL                  162
2427                  162     SEA            AL                  162
2428                  162     TBA            AL                  162
2429                  163     CHN            NL                  163
2430                  163     LAN            NL                  163

      VisitorRunsScored  ...    HomeBatting7Name  HomeBatting7Position  \
0                     2  ...          Nick Ahmed                     6
1                     5  ...      Dansby Swanson                     6
2                     1  ...     Yasmani Grandal                     2
3                     8  ...        Miguel Rojas                     6
4                     4  ...      Kevin Plawecki                     2
5                     2  ...       Freddy Galvis                     6
6                     2  ...       Pedro Alvarez                    10
7                    14  ...         Alex Gordon                     8
8                     5  ...        Matt Chapman                     5
9                     1  ...          Ryon Healy                     3
10                    4  ...  Adeiny Hechavarria                     6
11                    4  ...   Robinson Chirinos                     2
12                    6  ...      Russell Martin                     2
13                    8  ...          Nick Ahmed                     6
14                    5  ...      Dansby Swanson                     6
15                    2  ...         Jose Peraza                     6
16                    1  ...         Chase Utley                     4
```

| | | | | |
|---|---|---|---|---|
| 17 | 1 | ... | Miguel Rojas | 6 |
| 18 | 8 | ... | Cory Spangenberg | 5 |
| 19 | 13 | ... | Mikie Mahtook | 7 |
| 20 | 2 | ... | Matt Chapman | 5 |
| 21 | 1 | ... | Adeiny Hechavarria | 6 |
| 22 | 1 | ... | Robinson Chirinos | 2 |
| 23 | 4 | ... | Russell Martin | 2 |
| 24 | 2 | ... | Nick Ahmed | 6 |
| 25 | 2 | ... | Ryan Flaherty | 5 |
| 26 | 13 | ... | Jose Peraza | 6 |
| 27 | 0 | ... | Austin Barnes | 2 |
| 28 | 10 | ... | Miguel Rojas | 6 |
| 29 | 2 | ... | Juan Lagares | 8 |
| ... | ... | ... | ... | ... |
| 2401 | 5 | ... | Jonathan Schoop | 4 |
| 2402 | 0 | ... | Austin Jackson | 8 |
| 2403 | 0 | ... | Scott Kingery | 6 |
| 2404 | 5 | ... | Manuel Margot | 8 |
| 2405 | 10 | ... | Gorkys Hernandez | 8 |
| 2406 | 5 | ... | Taylor Ward | 5 |
| 2407 | 4 | ... | DJ Stewart | 9 |
| 2408 | 5 | ... | John Andreoli | 7 |
| 2409 | 8 | ... | Jackie Bradley | 8 |
| 2410 | 4 | ... | Brian Goodwin | 8 |
| 2411 | 3 | ... | Logan Forsythe | 4 |
| 2412 | 1 | ... | Daniel Vogelbach | 3 |
| 2413 | 3 | ... | Austin Meadows | 10 |
| 2414 | 5 | ... | Kyle Schwarber | 7 |
| 2415 | 6 | ... | Dilson Herrera | 4 |
| 2416 | 0 | ... | Ian Desmond | 3 |
| 2417 | 0 | ... | Manny Pina | 2 |
| 2418 | 0 | ... | Austin Jackson | 8 |
| 2419 | 1 | ... | Scott Kingery | 6 |
| 2420 | 3 | ... | Jose Pirela | 4 |
| 2421 | 15 | ... | Gorkys Hernandez | 8 |
| 2422 | 4 | ... | Kaleb Cowart | 6 |
| 2423 | 0 | ... | DJ Stewart | 9 |
| 2424 | 2 | ... | Ian Kinsler | 4 |
| 2425 | 2 | ... | Alcides Escobar | 5 |
| 2426 | 4 | ... | Logan Forsythe | 4 |
| 2427 | 1 | ... | Kristopher Negron | 4 |
| 2428 | 4 | ... | Austin Meadows | 9 |
| 2429 | 3 | ... | Jason Heyward | 8 |
| 2430 | 2 | ... | Yasiel Puig | 9 |

| | HomeBatting8PlayerID | HomeBatting8Name | HomeBatting8Position | \ |
|---|---|---|---|---|
| 0 | dysoj001 | Jarrod Dyson | 9 | |
| 1 | flahr001 | Ryan Flaherty | 5 | |

| | | | |
|---|---|---|---|
| 2 | forsl001 | Logan Forsythe | 5 |
| 3 | wallc001 | Chad Wallach | 2 |
| 4 | syndn001 | Noah Syndergaard | 1 |
| 5 | hedga001 | Austin Hedges | 2 |
| 6 | gentc001 | Craig Gentry | 9 |
| 7 | escoa003 | Alcides Escobar | 6 |
| 8 | lucrj001 | Jonathan Lucroy | 2 |
| 9 | marjm001 | Mike Marjama | 2 |
| 10 | robed004 | Daniel Robertson | 4 |
| 11 | odorr001 | Rougned Odor | 4 |
| 12 | pillk001 | Kevin Pillar | 8 |
| 13 | murpj001 | John Ryan Murphy | 2 |
| 14 | flahr001 | Ryan Flaherty | 5 |
| 15 | bailh001 | Homer Bailey | 1 |
| 16 | forsl001 | Logan Forsythe | 5 |
| 17 | wallc001 | Chad Wallach | 2 |
| 18 | hedga001 | Austin Hedges | 2 |
| 19 | iglej001 | Jose Iglesias | 6 |
| 20 | lucrj001 | Jonathan Lucroy | 2 |
| 21 | robed004 | Daniel Robertson | 4 |
| 22 | odorr001 | Rougned Odor | 4 |
| 23 | pillk001 | Kevin Pillar | 8 |
| 24 | mathj001 | Jeff Mathis | 2 |
| 25 | stewc001 | Chris Stewart | 2 |
| 26 | barnt001 | Tucker Barnhart | 2 |
| 27 | farmk001 | Kyle Farmer | 5 |
| 28 | holab001 | Bryan Holaday | 2 |
| 29 | degrj001 | Jacob deGrom | 1 |
| ... | ... | ... | ... |
| 2401 | krate001 | Erik Kratz | 2 |
| 2402 | plawk001 | Kevin Plawecki | 2 |
| 2403 | alfaj002 | Jorge Alfaro | 2 |
| 2404 | guerj004 | Javy Guerra | 6 |
| 2405 | blang001 | Gregor Blanco | 7 |
| 2406 | cowak001 | Kaleb Cowart | 4 |
| 2407 | rickj001 | Joey Rickard | 7 |
| 2408 | wynna001 | Austin Wynns | 2 |
| 2409 | swihb001 | Blake Swihart | 9 |
| 2410 | escoa003 | Alcides Escobar | 5 |
| 2411 | fielj003 | Johnny Field | 7 |
| 2412 | zunim001 | Mike Zunino | 2 |
| 2413 | sucrj001 | Jesus Sucre | 2 |
| 2414 | contw001 | Willson Contreras | 2 |
| 2415 | fedet001 | Tim Federowicz | 2 |
| 2416 | iannc001 | Chris Iannetta | 2 |
| 2417 | arcio002 | Orlando Arcia | 6 |
| 2418 | nidot001 | Tomas Nido | 2 |
| 2419 | knapa001 | Andrew Knapp | 2 |

| | | | |
|---|---|---|---|
| 2420 | margm001 | Manuel Margot | 8 |
| 2421 | blang001 | Gregor Blanco | 7 |
| 2422 | hudsj002 | Joe Hudson | 2 |
| 2423 | wilks001 | Steve Wilkerson | 4 |
| 2424 | leons001 | Sandy Leon | 2 |
| 2425 | philb002 | Brett Phillips | 7 |
| 2426 | astuw001 | Willians Astudillo | 5 |
| 2427 | freid001 | David Freitas | 2 |
| 2428 | bauej001 | Jake Bauers | 3 |
| 2429 | quinj001 | Jose Quintana | 1 |
| 2430 | herne001 | Enrique Hernandez | 4 |

| | HomeBatting9PlayerID | HomeBatting9Name | HomeBatting9Position | \ |
|---|---|---|---|---|
| 0 | corbp001 | Patrick Corbin | 1 | |
| 1 | tehej001 | Julio Teheran | 1 | |
| 2 | kersc001 | Clayton Kershaw | 1 | |
| 3 | urenj001 | Jose Urena | 1 | |
| 4 | rosaa003 | Amed Rosario | 6 | |
| 5 | richc002 | Clayton Richard | 1 | |
| 6 | josec002 | Caleb Joseph | 2 | |
| 7 | buted001 | Drew Butera | 2 | |
| 8 | poweb002 | Boog Powell | 8 | |
| 9 | suzui001 | Ichiro Suzuki | 7 | |
| 10 | refsr001 | Rob Refsnyder | 10 | |
| 11 | rua-r001 | Ryan Rua | 7 | |
| 12 | diaza003 | Aledmys Diaz | 6 | |
| 13 | ray-r002 | Robbie Ray | 1 | |
| 14 | foltm001 | Mike Foltynewicz | 1 | |
| 15 | hamib001 | Billy Hamilton | 8 | |
| 16 | wooda002 | Alex Wood | 1 | |
| 17 | smitc006 | Caleb Smith | 1 | |
| 18 | luccj001 | Joey Lucchesi | 1 | |
| 19 | machd001 | Dixon Machado | 4 | |
| 20 | pindc001 | Chad Pinder | 7 | |
| 21 | refsr001 | Rob Refsnyder | 7 | |
| 22 | rua-r001 | Ryan Rua | 7 | |
| 23 | diaza003 | Aledmys Diaz | 6 | |
| 24 | greiz001 | Zack Greinke | 1 | |
| 25 | mccab001 | Brandon McCarthy | 1 | |
| 26 | castl003 | Luis Castillo | 1 | |
| 27 | maedk001 | Kenta Maeda | 1 | |
| 28 | despo001 | Odrisamer Despaigne | 1 | |
| 29 | rosaa003 | Amed Rosario | 6 | |
| ... | ... | ... | ... | |
| 2401 | milew001 | Wade Miley | 1 | |
| 2402 | matzs001 | Steven Matz | 1 | |
| 2403 | nolaa001 | Aaron Nola | 1 | |
| 2404 | nix-j002 | Jacob Nix | 1 | |

| | | | |
|---|---|---|---|
| 2405 | rodrd001 | Dereck Rodriguez | 1 |
| 2406 | bricj001 | Jose Briceno | 2 |
| 2407 | josec002 | Caleb Joseph | 2 |
| 2408 | wilks001 | Steve Wilkerson | 4 |
| 2409 | vazqc001 | Christian Vazquez | 2 |
| 2410 | vilom001 | Meibrys Viloria | 2 |
| 2411 | gimec001 | Chris Gimenez | 2 |
| 2412 | gordd002 | Dee Gordon | 8 |
| 2413 | velaa001 | Andrew Velazquez | 5 |
| 2414 | montm002 | Mike Montgomery | 1 |
| 2415 | romas001 | Sal Romano | 1 |
| 2416 | andet002 | Tyler Anderson | 1 |
| 2417 | gonzg003 | Gio Gonzalez | 1 |
| 2418 | syndn001 | Noah Syndergaard | 1 |
| 2419 | suarr001 | Ranger Suarez | 1 |
| 2420 | luccj001 | Joey Lucchesi | 1 |
| 2421 | suara002 | Andrew Suarez | 1 |
| 2422 | johns002 | Sherman Johnson | 4 |
| 2423 | josec002 | Caleb Joseph | 2 |
| 2424 | bradj001 | Jackie Bradley | 8 |
| 2425 | vilom001 | Meibrys Viloria | 2 |
| 2426 | gratj001 | Juan Graterol | 2 |
| 2427 | romia001 | Andrew Romine | 6 |
| 2428 | ciufn001 | Nick Ciuffo | 2 |
| 2429 | contw001 | Willson Contreras | 2 |
| 2430 | buehw001 | Walker Buehler | 1 |

| | AdditionalInfo | AcquisitionInfo |
|---|---|---|
| 0 | NaN | Y |
| 1 | NaN | Y |
| 2 | NaN | Y |
| 3 | NaN | Y |
| 4 | NaN | Y |
| 5 | NaN | Y |
| 6 | NaN | Y |
| 7 | NaN | Y |
| 8 | NaN | Y |
| 9 | NaN | Y |
| 10 | NaN | Y |
| 11 | NaN | Y |
| 12 | NaN | Y |
| 13 | NaN | Y |
| 14 | NaN | Y |
| 15 | NaN | Y |
| 16 | NaN | Y |
| 17 | NaN | Y |
| 18 | NaN | Y |
| 19 | umpchange,8,umphome,randt901,8,ump2b,(None) | Y |

```
20                                                          NaN              Y
21                                                          NaN              Y
22                                                          NaN              Y
23                                                          NaN              Y
24                                                          NaN              Y
25                                                          NaN              Y
26                                                          NaN              Y
27                                                          NaN              Y
28                                                          NaN              Y
29                                                          NaN              Y
...                                                         ...            ...
2401                                                        NaN              Y
2402                                                        NaN              Y
2403                                                        NaN              Y
2404                                                        NaN              Y
2405                                                        NaN              Y
2406                                                        NaN              Y
2407                                                        NaN              Y
2408                                                        NaN              Y
2409                                                        NaN              Y
2410                                                        NaN              Y
2411                                                        NaN              Y
2412                                                        NaN              Y
2413                                                        NaN              Y
2414                                                        NaN              Y
2415                                                        NaN              Y
2416                                                        NaN              Y
2417                                                        NaN              Y
2418                                                        NaN              Y
2419                                                        NaN              Y
2420                                                        NaN              Y
2421                                                        NaN              Y
2422                                                        NaN              Y
2423                                                        NaN              Y
2424                                                        NaN              Y
2425                                                        NaN              Y
2426                                                        NaN              Y
2427                                                        NaN              Y
2428                                                        NaN              Y
2429                                                        NaN              Y
2430                                                        NaN              Y

[2431 rows x 161 columns]
```

In [10]:
```
# For the Pythagorean Expectation we need only runs scored and conceded. Of course, w
# and the date will also be useful. We put these into a new dataframe (df) which we c
# The variable names are rather lengthy, so to make life easier we can rename columns
# If we want to see what the data looks like, we can just type the name of the df.
```

```python
MLB18 = MLB[['VisitingTeam','HomeTeam','VisitorRunsScored','HomeRunsScore','Date']]
MLB18 = MLB18.rename(columns={'VisitorRunsScored':'VisR','HomeRunsScore':'HomR'})
MLB18
```

Out[10]:

| | VisitingTeam | HomeTeam | VisR | HomR | Date |
|---|---|---|---|---|---|
| 0 | COL | ARI | 2 | 8 | 20180329 |
| 1 | PHI | ATL | 5 | 8 | 20180329 |
| 2 | SFN | LAN | 1 | 0 | 20180329 |
| 3 | CHN | MIA | 8 | 4 | 20180329 |
| 4 | SLN | NYN | 4 | 9 | 20180329 |
| 5 | MIL | SDN | 2 | 1 | 20180329 |
| 6 | MIN | BAL | 2 | 3 | 20180329 |
| 7 | CHA | KCA | 14 | 7 | 20180329 |
| 8 | ANA | OAK | 5 | 6 | 20180329 |
| 9 | CLE | SEA | 1 | 2 | 20180329 |
| 10 | BOS | TBA | 4 | 6 | 20180329 |
| 11 | HOU | TEX | 4 | 1 | 20180329 |
| 12 | NYA | TOR | 6 | 1 | 20180329 |
| 13 | COL | ARI | 8 | 9 | 20180330 |
| 14 | PHI | ATL | 5 | 4 | 20180330 |
| 15 | WAS | CIN | 2 | 0 | 20180330 |
| 16 | SFN | LAN | 1 | 0 | 20180330 |
| 17 | CHN | MIA | 1 | 2 | 20180330 |
| 18 | MIL | SDN | 8 | 6 | 20180330 |
| 19 | PIT | DET | 13 | 10 | 20180330 |
| 20 | ANA | OAK | 2 | 1 | 20180330 |
| 21 | BOS | TBA | 1 | 0 | 20180330 |
| 22 | HOU | TEX | 1 | 5 | 20180330 |
| 23 | NYA | TOR | 4 | 2 | 20180330 |
| 24 | COL | ARI | 2 | 1 | 20180331 |
| 25 | PHI | ATL | 2 | 15 | 20180331 |
| 26 | WAS | CIN | 13 | 7 | 20180331 |
| 27 | SFN | LAN | 0 | 5 | 20180331 |
| 28 | CHN | MIA | 10 | 6 | 20180331 |
| 29 | SLN | NYN | 2 | 6 | 20180331 |
| ... | ... | ... | ... | ... | ... |
| 2401 | DET | MIL | 5 | 6 | 20180929 |
| 2402 | MIA | NYN | 0 | 1 | 20180929 |
| 2403 | ATL | PHI | 0 | 3 | 20180929 |
| 2404 | ARI | SDN | 5 | 4 | 20180929 |
| 2405 | LAN | SFN | 10 | 6 | 20180929 |
| 2406 | OAK | ANA | 5 | 2 | 20180929 |
| 2407 | HOU | BAL | 4 | 3 | 20180929 |
| 2408 | HOU | BAL | 5 | 2 | 20180929 |
| 2409 | NYA | BOS | 8 | 5 | 20180929 |
| 2410 | CLE | KCA | 4 | 9 | 20180929 |
| 2411 | CHA | MIN | 3 | 8 | 20180929 |

```
2412           TEX       SEA      1     4  20180929
2413           TOR       TBA      3     4  20180929
2414           SLN       CHN      5    10  20180930
2415           PIT       CIN      6     5  20180930
2416           WAS       COL      0    12  20180930
2417           DET       MIL      0    11  20180930
2418           MIA       NYN      0     1  20180930
2419           ATL       PHI      1     3  20180930
2420           ARI       SDN      3     4  20180930
2421           LAN       SFN     15     0  20180930
2422           OAK       ANA      4     5  20180930
2423           HOU       BAL      0     4  20180930
2424           NYA       BOS      2    10  20180930
2425           CLE       KCA      2     1  20180930
2426           CHA       MIN      4     5  20180930
2427           TEX       SEA      1     3  20180930
2428           TOR       TBA      4     9  20180930
2429           MIL       CHN      3     1  20181001
2430           COL       LAN      2     5  20181001

[2431 rows x 5 columns]
```

In [11]:
```python
# We will need to know who won the game - which we can tell by who scored the more ru
#(there are no ties in baseball)
# The variable 'hwin' is defined here as equaling 1 if the home team scored more runs
# The variable 'awin' is defined in a similar way for the away team.
# we also create a 'counter' variable = 1 for each row.

MLB18['hwin']= np.where(MLB18['HomR']>MLB18['VisR'],1,0)
MLB18['awin']= np.where(MLB18['HomR']<MLB18['VisR'],1,0)
MLB18['count']=1
MLB18
```

Out[11]:

| | VisitingTeam | HomeTeam | VisR | HomR | Date | hwin | awin | count |
|---|---|---|---|---|---|---|---|---|
| 0 | COL | ARI | 2 | 8 | 20180329 | 1 | 0 | 1 |
| 1 | PHI | ATL | 5 | 8 | 20180329 | 1 | 0 | 1 |
| 2 | SFN | LAN | 1 | 0 | 20180329 | 0 | 1 | 1 |
| 3 | CHN | MIA | 8 | 4 | 20180329 | 0 | 1 | 1 |
| 4 | SLN | NYN | 4 | 9 | 20180329 | 1 | 0 | 1 |
| 5 | MIL | SDN | 2 | 1 | 20180329 | 0 | 1 | 1 |
| 6 | MIN | BAL | 2 | 3 | 20180329 | 1 | 0 | 1 |
| 7 | CHA | KCA | 14 | 7 | 20180329 | 0 | 1 | 1 |
| 8 | ANA | OAK | 5 | 6 | 20180329 | 1 | 0 | 1 |
| 9 | CLE | SEA | 1 | 2 | 20180329 | 1 | 0 | 1 |
| 10 | BOS | TBA | 4 | 6 | 20180329 | 1 | 0 | 1 |
| 11 | HOU | TEX | 4 | 1 | 20180329 | 0 | 1 | 1 |
| 12 | NYA | TOR | 6 | 1 | 20180329 | 0 | 1 | 1 |
| 13 | COL | ARI | 8 | 9 | 20180330 | 1 | 0 | 1 |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 14 | PHI | ATL | 5 | 4 | 20180330 | 0 | 1 | 1 |
| 15 | WAS | CIN | 2 | 0 | 20180330 | 0 | 1 | 1 |
| 16 | SFN | LAN | 1 | 0 | 20180330 | 0 | 1 | 1 |
| 17 | CHN | MIA | 1 | 2 | 20180330 | 1 | 0 | 1 |
| 18 | MIL | SDN | 8 | 6 | 20180330 | 0 | 1 | 1 |
| 19 | PIT | DET | 13 | 10 | 20180330 | 0 | 1 | 1 |
| 20 | ANA | OAK | 2 | 1 | 20180330 | 0 | 1 | 1 |
| 21 | BOS | TBA | 1 | 0 | 20180330 | 0 | 1 | 1 |
| 22 | HOU | TEX | 1 | 5 | 20180330 | 1 | 0 | 1 |
| 23 | NYA | TOR | 4 | 2 | 20180330 | 0 | 1 | 1 |
| 24 | COL | ARI | 2 | 1 | 20180331 | 0 | 1 | 1 |
| 25 | PHI | ATL | 2 | 15 | 20180331 | 1 | 0 | 1 |
| 26 | WAS | CIN | 13 | 7 | 20180331 | 0 | 1 | 1 |
| 27 | SFN | LAN | 0 | 5 | 20180331 | 1 | 0 | 1 |
| 28 | CHN | MIA | 10 | 6 | 20180331 | 0 | 1 | 1 |
| 29 | SLN | NYN | 2 | 6 | 20180331 | 1 | 0 | 1 |
| ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 2401 | DET | MIL | 5 | 6 | 20180929 | 1 | 0 | 1 |
| 2402 | MIA | NYN | 0 | 1 | 20180929 | 1 | 0 | 1 |
| 2403 | ATL | PHI | 0 | 3 | 20180929 | 1 | 0 | 1 |
| 2404 | ARI | SDN | 5 | 4 | 20180929 | 0 | 1 | 1 |
| 2405 | LAN | SFN | 10 | 6 | 20180929 | 0 | 1 | 1 |
| 2406 | OAK | ANA | 5 | 2 | 20180929 | 0 | 1 | 1 |
| 2407 | HOU | BAL | 4 | 3 | 20180929 | 0 | 1 | 1 |
| 2408 | HOU | BAL | 5 | 2 | 20180929 | 0 | 1 | 1 |
| 2409 | NYA | BOS | 8 | 5 | 20180929 | 0 | 1 | 1 |
| 2410 | CLE | KCA | 4 | 9 | 20180929 | 1 | 0 | 1 |
| 2411 | CHA | MIN | 3 | 8 | 20180929 | 1 | 0 | 1 |
| 2412 | TEX | SEA | 1 | 4 | 20180929 | 1 | 0 | 1 |
| 2413 | TOR | TBA | 3 | 4 | 20180929 | 1 | 0 | 1 |
| 2414 | SLN | CHN | 5 | 10 | 20180930 | 1 | 0 | 1 |
| 2415 | PIT | CIN | 6 | 5 | 20180930 | 0 | 1 | 1 |
| 2416 | WAS | COL | 0 | 12 | 20180930 | 1 | 0 | 1 |
| 2417 | DET | MIL | 0 | 11 | 20180930 | 1 | 0 | 1 |
| 2418 | MIA | NYN | 0 | 1 | 20180930 | 1 | 0 | 1 |
| 2419 | ATL | PHI | 1 | 3 | 20180930 | 1 | 0 | 1 |
| 2420 | ARI | SDN | 3 | 4 | 20180930 | 1 | 0 | 1 |
| 2421 | LAN | SFN | 15 | 0 | 20180930 | 0 | 1 | 1 |
| 2422 | OAK | ANA | 4 | 5 | 20180930 | 1 | 0 | 1 |
| 2423 | HOU | BAL | 0 | 4 | 20180930 | 1 | 0 | 1 |
| 2424 | NYA | BOS | 2 | 10 | 20180930 | 1 | 0 | 1 |
| 2425 | CLE | KCA | 2 | 1 | 20180930 | 0 | 1 | 1 |
| 2426 | CHA | MIN | 4 | 5 | 20180930 | 1 | 0 | 1 |
| 2427 | TEX | SEA | 1 | 3 | 20180930 | 1 | 0 | 1 |
| 2428 | TOR | TBA | 4 | 9 | 20180930 | 1 | 0 | 1 |
| 2429 | MIL | CHN | 3 | 1 | 20181001 | 0 | 1 | 1 |
| 2430 | COL | LAN | 2 | 5 | 20181001 | 1 | 0 | 1 |

```
[2431 rows x 8 columns]
```

In [13]:
```
# Since our data refers to games, for each game there are two teams, but what we want
# by each team and its win percentage.
# To create this we are going to define two dfs, one for home teams and one for away
# the stats for the entire season.
# Here we define a df for home teams. The command is called ".groupby" and we will us
# to obtain the sum of wins and runs (scored and conceded) and also the counter varia
# (in MLB the teams do not necessarily play the same number of games in the regular s
# Finally we rename the columns.

MLBhome = MLB18.groupby('HomeTeam')['hwin','HomR','VisR','count'].sum().reset_index()
MLBhome = MLBhome.rename(columns={'HomeTeam':'team','VisR':'VisRh','HomR':'HomRh','co
MLBhome
```

Out[13]:

| | team | hwin | HomRh | VisRh | Gh |
|---|---|---|---|---|---|
| 0 | ANA | 42 | 355 | 355 | 81 |
| 1 | ARI | 40 | 359 | 328 | 81 |
| 2 | ATL | 43 | 391 | 357 | 81 |
| 3 | BAL | 28 | 339 | 411 | 81 |
| 4 | BOS | 57 | 468 | 322 | 81 |
| 5 | CHA | 30 | 321 | 409 | 81 |
| 6 | CHN | 51 | 385 | 349 | 82 |
| 7 | CIN | 37 | 385 | 418 | 81 |
| 8 | CLE | 49 | 443 | 334 | 81 |
| 9 | COL | 47 | 445 | 404 | 81 |
| 10 | DET | 38 | 330 | 363 | 81 |
| 11 | HOU | 46 | 373 | 288 | 81 |
| 12 | KCA | 32 | 333 | 424 | 81 |
| 13 | LAN | 45 | 366 | 297 | 82 |
| 14 | MIA | 38 | 279 | 323 | 81 |
| 15 | MIL | 51 | 384 | 322 | 81 |
| 16 | MIN | 49 | 397 | 361 | 81 |
| 17 | NYA | 53 | 453 | 352 | 81 |
| 18 | NYN | 37 | 274 | 310 | 81 |
| 19 | OAK | 50 | 369 | 310 | 81 |
| 20 | PHI | 49 | 370 | 347 | 81 |
| 21 | PIT | 44 | 326 | 318 | 80 |
| 22 | SDN | 31 | 313 | 390 | 81 |
| 23 | SEA | 45 | 299 | 337 | 81 |
| 24 | SFN | 42 | 321 | 334 | 81 |
| 25 | SLN | 43 | 351 | 346 | 81 |
| 26 | TBA | 51 | 371 | 284 | 81 |
| 27 | TEX | 34 | 432 | 479 | 81 |
| 28 | TOR | 40 | 361 | 393 | 81 |
| 29 | WAS | 41 | 409 | 363 | 81 |

In [ ]:
```
#Your Code Here
```

# 3 Self test - 1

Sometimes the code you write doesn't produce the result you want, and you need to go back and re-do it. Frequently it makes sense to go back to the beginning, rather than try to amend a df which isn't working the way you want it to. Re-starting is easy- just click on "Kernel" in the toolbar and then click "Restart and Clear Output". You can now begin again.

Copy the previous cell (first use "Insert" to add a extra cell, and then use copy and paste), and then delete ".reset_index()" and then run the code to see what happens differently. The extra headings would be a problem later on, which makes ".reset_index()" very useful in many situations.

```
In [14]: # Now we create a similar df for teams playing as visitors - To write this code all y
         # the previous cell and then change any reference to the home team into a reference t

         MLBaway = MLB18.groupby('VisitingTeam')['awin','HomR','VisR','count'].sum().reset_inde
         MLBaway = MLBaway.rename(columns={'VisitingTeam':'team','VisR':'VisRa','HomR':'HomRa'
         MLBaway
```

Out[14]:

| | team | awin | HomRa | VisRa | Ga |
|---|---|---|---|---|---|
| 0 | ANA | 38 | 367 | 366 | 81 |
| 1 | ARI | 42 | 316 | 334 | 81 |
| 2 | ATL | 47 | 300 | 368 | 81 |
| 3 | BAL | 19 | 481 | 283 | 81 |
| 4 | BOS | 51 | 325 | 408 | 81 |
| 5 | CHA | 32 | 439 | 335 | 81 |
| 6 | CHN | 44 | 296 | 376 | 81 |
| 7 | CIN | 30 | 401 | 311 | 81 |
| 8 | CLE | 42 | 314 | 375 | 81 |
| 9 | COL | 44 | 341 | 335 | 82 |
| 10 | DET | 26 | 433 | 300 | 81 |
| 11 | HOU | 57 | 246 | 424 | 81 |
| 12 | KCA | 26 | 409 | 305 | 81 |
| 13 | LAN | 47 | 313 | 438 | 81 |
| 14 | MIA | 25 | 486 | 310 | 80 |
| 15 | MIL | 45 | 337 | 370 | 82 |
| 16 | MIN | 29 | 414 | 341 | 81 |
| 17 | NYA | 47 | 317 | 398 | 81 |
| 18 | NYN | 40 | 397 | 402 | 81 |
| 19 | OAK | 47 | 364 | 444 | 81 |
| 20 | PHI | 31 | 381 | 307 | 81 |
| 21 | PIT | 38 | 375 | 366 | 81 |
| 22 | SDN | 35 | 377 | 304 | 81 |
| 23 | SEA | 44 | 374 | 378 | 81 |
| 24 | SFN | 31 | 365 | 282 | 81 |
| 25 | SLN | 45 | 345 | 408 | 81 |
| 26 | TBA | 39 | 362 | 345 | 81 |
| 27 | TEX | 33 | 369 | 305 | 81 |
| 28 | TOR | 33 | 439 | 348 | 81 |
| 29 | WAS | 41 | 319 | 362 | 81 |

```
In [15]: # We now merge MLBhome and MLBaway so that we have a list of all the clubs with home
         # We will be using pd.merge frequently during the course to combine dfs
         # Note that we've called this new df "MLB18", which is name we had already used for e
         # overwriting the old MLB18 - which is fine in this case since we don't need the data
         # If we did want to retain the daat in the old MLB18 df, we should have given this ne

         MLB18 = pd.merge(MLBhome,MLBaway,on='team')
         MLB18
```

Out[15]:

| | team | hwin | HomRh | VisRh | Gh | awin | HomRa | VisRa | Ga |
|---|---|---|---|---|---|---|---|---|---|
| 0 | ANA | 42 | 355 | 355 | 81 | 38 | 367 | 366 | 81 |
| 1 | ARI | 40 | 359 | 328 | 81 | 42 | 316 | 334 | 81 |
| 2 | ATL | 43 | 391 | 357 | 81 | 47 | 300 | 368 | 81 |
| 3 | BAL | 28 | 339 | 411 | 81 | 19 | 481 | 283 | 81 |
| 4 | BOS | 57 | 468 | 322 | 81 | 51 | 325 | 408 | 81 |
| 5 | CHA | 30 | 321 | 409 | 81 | 32 | 439 | 335 | 81 |
| 6 | CHN | 51 | 385 | 349 | 82 | 44 | 296 | 376 | 81 |
| 7 | CIN | 37 | 385 | 418 | 81 | 30 | 401 | 311 | 81 |
| 8 | CLE | 49 | 443 | 334 | 81 | 42 | 314 | 375 | 81 |
| 9 | COL | 47 | 445 | 404 | 81 | 44 | 341 | 335 | 82 |
| 10 | DET | 38 | 330 | 363 | 81 | 26 | 433 | 300 | 81 |
| 11 | HOU | 46 | 373 | 288 | 81 | 57 | 246 | 424 | 81 |
| 12 | KCA | 32 | 333 | 424 | 81 | 26 | 409 | 305 | 81 |
| 13 | LAN | 45 | 366 | 297 | 82 | 47 | 313 | 438 | 81 |
| 14 | MIA | 38 | 279 | 323 | 81 | 25 | 486 | 310 | 80 |
| 15 | MIL | 51 | 384 | 322 | 81 | 45 | 337 | 370 | 82 |
| 16 | MIN | 49 | 397 | 361 | 81 | 29 | 414 | 341 | 81 |
| 17 | NYA | 53 | 453 | 352 | 81 | 47 | 317 | 398 | 81 |
| 18 | NYN | 37 | 274 | 310 | 81 | 40 | 397 | 402 | 81 |
| 19 | OAK | 50 | 369 | 310 | 81 | 47 | 364 | 444 | 81 |
| 20 | PHI | 49 | 370 | 347 | 81 | 31 | 381 | 307 | 81 |
| 21 | PIT | 44 | 326 | 318 | 80 | 38 | 375 | 366 | 81 |
| 22 | SDN | 31 | 313 | 390 | 81 | 35 | 377 | 304 | 81 |
| 23 | SEA | 45 | 299 | 337 | 81 | 44 | 374 | 378 | 81 |
| 24 | SFN | 42 | 321 | 334 | 81 | 31 | 365 | 282 | 81 |
| 25 | SLN | 43 | 351 | 346 | 81 | 45 | 345 | 408 | 81 |
| 26 | TBA | 51 | 371 | 284 | 81 | 39 | 362 | 345 | 81 |
| 27 | TEX | 34 | 432 | 479 | 81 | 33 | 369 | 305 | 81 |
| 28 | TOR | 40 | 361 | 393 | 81 | 33 | 439 | 348 | 81 |
| 29 | WAS | 41 | 409 | 363 | 81 | 41 | 319 | 362 | 81 |

## 4 Self test - 2

When creating MLBhome and MLBaway we we renamed the variables using ".rename(columns ={'oldname':'newname'})". Copy and paste these cells and then re-run the code and see how the merge looks. Note that when Python encounters two variables with the same name in a merge it relabels the names with _x and _y.

Sometimes we can work with the data in this way, but usually renaming makes it easier to follow.

```
In [16]: # Now we create the total wins, games, played, runs scored and run conceded by summing

         MLB18['W']=MLB18['hwin']+MLB18['awin']
         MLB18['G']=MLB18['Gh']+MLB18['Ga']
         MLB18['R']=MLB18['HomRh']+MLB18['VisRa']
         MLB18['RA']=MLB18['VisRh']+MLB18['HomRa']
         MLB18
```

Out[16]:

| | team | hwin | HomRh | VisRh | Gh | awin | HomRa | VisRa | Ga | W | G | R | RA |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | ANA | 42 | 355 | 355 | 81 | 38 | 367 | 366 | 81 | 80 | 162 | 721 | 722 |
| 1 | ARI | 40 | 359 | 328 | 81 | 42 | 316 | 334 | 81 | 82 | 162 | 693 | 644 |
| 2 | ATL | 43 | 391 | 357 | 81 | 47 | 300 | 368 | 81 | 90 | 162 | 759 | 657 |
| 3 | BAL | 28 | 339 | 411 | 81 | 19 | 481 | 283 | 81 | 47 | 162 | 622 | 892 |
| 4 | BOS | 57 | 468 | 322 | 81 | 51 | 325 | 408 | 81 | 108 | 162 | 876 | 647 |
| 5 | CHA | 30 | 321 | 409 | 81 | 32 | 439 | 335 | 81 | 62 | 162 | 656 | 848 |
| 6 | CHN | 51 | 385 | 349 | 82 | 44 | 296 | 376 | 81 | 95 | 163 | 761 | 645 |
| 7 | CIN | 37 | 385 | 418 | 81 | 30 | 401 | 311 | 81 | 67 | 162 | 696 | 819 |
| 8 | CLE | 49 | 443 | 334 | 81 | 42 | 314 | 375 | 81 | 91 | 162 | 818 | 648 |
| 9 | COL | 47 | 445 | 404 | 81 | 44 | 341 | 335 | 82 | 91 | 163 | 780 | 745 |
| 10 | DET | 38 | 330 | 363 | 81 | 26 | 433 | 300 | 81 | 64 | 162 | 630 | 796 |
| 11 | HOU | 46 | 373 | 288 | 81 | 57 | 246 | 424 | 81 | 103 | 162 | 797 | 534 |
| 12 | KCA | 32 | 333 | 424 | 81 | 26 | 409 | 305 | 81 | 58 | 162 | 638 | 833 |
| 13 | LAN | 45 | 366 | 297 | 82 | 47 | 313 | 438 | 81 | 92 | 163 | 804 | 610 |
| 14 | MIA | 38 | 279 | 323 | 81 | 25 | 486 | 310 | 80 | 63 | 161 | 589 | 809 |
| 15 | MIL | 51 | 384 | 322 | 81 | 45 | 337 | 370 | 82 | 96 | 163 | 754 | 659 |
| 16 | MIN | 49 | 397 | 361 | 81 | 29 | 414 | 341 | 81 | 78 | 162 | 738 | 775 |
| 17 | NYA | 53 | 453 | 352 | 81 | 47 | 317 | 398 | 81 | 100 | 162 | 851 | 669 |
| 18 | NYN | 37 | 274 | 310 | 81 | 40 | 397 | 402 | 81 | 77 | 162 | 676 | 707 |
| 19 | OAK | 50 | 369 | 310 | 81 | 47 | 364 | 444 | 81 | 97 | 162 | 813 | 674 |
| 20 | PHI | 49 | 370 | 347 | 81 | 31 | 381 | 307 | 81 | 80 | 162 | 677 | 728 |
| 21 | PIT | 44 | 326 | 318 | 80 | 38 | 375 | 366 | 81 | 82 | 161 | 692 | 693 |
| 22 | SDN | 31 | 313 | 390 | 81 | 35 | 377 | 304 | 81 | 66 | 162 | 617 | 767 |
| 23 | SEA | 45 | 299 | 337 | 81 | 44 | 374 | 378 | 81 | 89 | 162 | 677 | 711 |
| 24 | SFN | 42 | 321 | 334 | 81 | 31 | 365 | 282 | 81 | 73 | 162 | 603 | 699 |
| 25 | SLN | 43 | 351 | 346 | 81 | 45 | 345 | 408 | 81 | 88 | 162 | 759 | 691 |
| 26 | TBA | 51 | 371 | 284 | 81 | 39 | 362 | 345 | 81 | 90 | 162 | 716 | 646 |
| 27 | TEX | 34 | 432 | 479 | 81 | 33 | 369 | 305 | 81 | 67 | 162 | 737 | 848 |
| 28 | TOR | 40 | 361 | 393 | 81 | 33 | 439 | 348 | 81 | 73 | 162 | 709 | 832 |
| 29 | WAS | 41 | 409 | 363 | 81 | 41 | 319 | 362 | 81 | 82 | 162 | 771 | 682 |

```
In [17]: # The last step in preparing the data is to define win percentage and the Pythagorean

         MLB18['wpc'] = MLB18['W']/MLB18['G']
         MLB18['pyth'] = MLB18['R']**2/(MLB18['R']**2 + MLB18['RA']**2)
         MLB18
```

Out[17]:

| | team | hwin | HomRh | VisRh | Gh | awin | HomRa | VisRa | Ga | W | G | R | RA | \ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | ANA | 42 | 355 | 355 | 81 | 38 | 367 | 366 | 81 | 80 | 162 | 721 | 722 | |
| 1 | ARI | 40 | 359 | 328 | 81 | 42 | 316 | 334 | 81 | 82 | 162 | 693 | 644 | |
| 2 | ATL | 43 | 391 | 357 | 81 | 47 | 300 | 368 | 81 | 90 | 162 | 759 | 657 | |
| 3 | BAL | 28 | 339 | 411 | 81 | 19 | 481 | 283 | 81 | 47 | 162 | 622 | 892 | |
| 4 | BOS | 57 | 468 | 322 | 81 | 51 | 325 | 408 | 81 | 108 | 162 | 876 | 647 | |
| 5 | CHA | 30 | 321 | 409 | 81 | 32 | 439 | 335 | 81 | 62 | 162 | 656 | 848 | |
| 6 | CHN | 51 | 385 | 349 | 82 | 44 | 296 | 376 | 81 | 95 | 163 | 761 | 645 | |
| 7 | CIN | 37 | 385 | 418 | 81 | 30 | 401 | 311 | 81 | 67 | 162 | 696 | 819 | |
| 8 | CLE | 49 | 443 | 334 | 81 | 42 | 314 | 375 | 81 | 91 | 162 | 818 | 648 | |
| 9 | COL | 47 | 445 | 404 | 81 | 44 | 341 | 335 | 82 | 91 | 163 | 780 | 745 | |
| 10 | DET | 38 | 330 | 363 | 81 | 26 | 433 | 300 | 81 | 64 | 162 | 630 | 796 | |
| 11 | HOU | 46 | 373 | 288 | 81 | 57 | 246 | 424 | 81 | 103 | 162 | 797 | 534 | |
| 12 | KCA | 32 | 333 | 424 | 81 | 26 | 409 | 305 | 81 | 58 | 162 | 638 | 833 | |
| 13 | LAN | 45 | 366 | 297 | 82 | 47 | 313 | 438 | 81 | 92 | 163 | 804 | 610 | |
| 14 | MIA | 38 | 279 | 323 | 81 | 25 | 486 | 310 | 80 | 63 | 161 | 589 | 809 | |
| 15 | MIL | 51 | 384 | 322 | 81 | 45 | 337 | 370 | 82 | 96 | 163 | 754 | 659 | |
| 16 | MIN | 49 | 397 | 361 | 81 | 29 | 414 | 341 | 81 | 78 | 162 | 738 | 775 | |
| 17 | NYA | 53 | 453 | 352 | 81 | 47 | 317 | 398 | 81 | 100 | 162 | 851 | 669 | |
| 18 | NYN | 37 | 274 | 310 | 81 | 40 | 397 | 402 | 81 | 77 | 162 | 676 | 707 | |
| 19 | OAK | 50 | 369 | 310 | 81 | 47 | 364 | 444 | 81 | 97 | 162 | 813 | 674 | |
| 20 | PHI | 49 | 370 | 347 | 81 | 31 | 381 | 307 | 81 | 80 | 162 | 677 | 728 | |
| 21 | PIT | 44 | 326 | 318 | 80 | 38 | 375 | 366 | 81 | 82 | 161 | 692 | 693 | |
| 22 | SDN | 31 | 313 | 390 | 81 | 35 | 377 | 304 | 81 | 66 | 162 | 617 | 767 | |
| 23 | SEA | 45 | 299 | 337 | 81 | 44 | 374 | 378 | 81 | 89 | 162 | 677 | 711 | |
| 24 | SFN | 42 | 321 | 334 | 81 | 31 | 365 | 282 | 81 | 73 | 162 | 603 | 699 | |
| 25 | SLN | 43 | 351 | 346 | 81 | 45 | 345 | 408 | 81 | 88 | 162 | 759 | 691 | |
| 26 | TBA | 51 | 371 | 284 | 81 | 39 | 362 | 345 | 81 | 90 | 162 | 716 | 646 | |
| 27 | TEX | 34 | 432 | 479 | 81 | 33 | 369 | 305 | 81 | 67 | 162 | 737 | 848 | |
| 28 | TOR | 40 | 361 | 393 | 81 | 33 | 439 | 348 | 81 | 73 | 162 | 709 | 832 | |
| 29 | WAS | 41 | 409 | 363 | 81 | 41 | 319 | 362 | 81 | 82 | 162 | 771 | 682 | |

| | wpc | pyth |
|---|---|---|
| 0 | 0.493827 | 0.499307 |
| 1 | 0.506173 | 0.536600 |
| 2 | 0.555556 | 0.571662 |
| 3 | 0.290123 | 0.327161 |
| 4 | 0.666667 | 0.647037 |
| 5 | 0.382716 | 0.374388 |
| 6 | 0.582822 | 0.581946 |
| 7 | 0.413580 | 0.419344 |
| 8 | 0.561728 | 0.614423 |
| 9 | 0.558282 | 0.522939 |
| 10 | 0.395062 | 0.385147 |
| 11 | 0.635802 | 0.690171 |
| 12 | 0.358025 | 0.369726 |
| 13 | 0.564417 | 0.634665 |
| 14 | 0.391304 | 0.346435 |

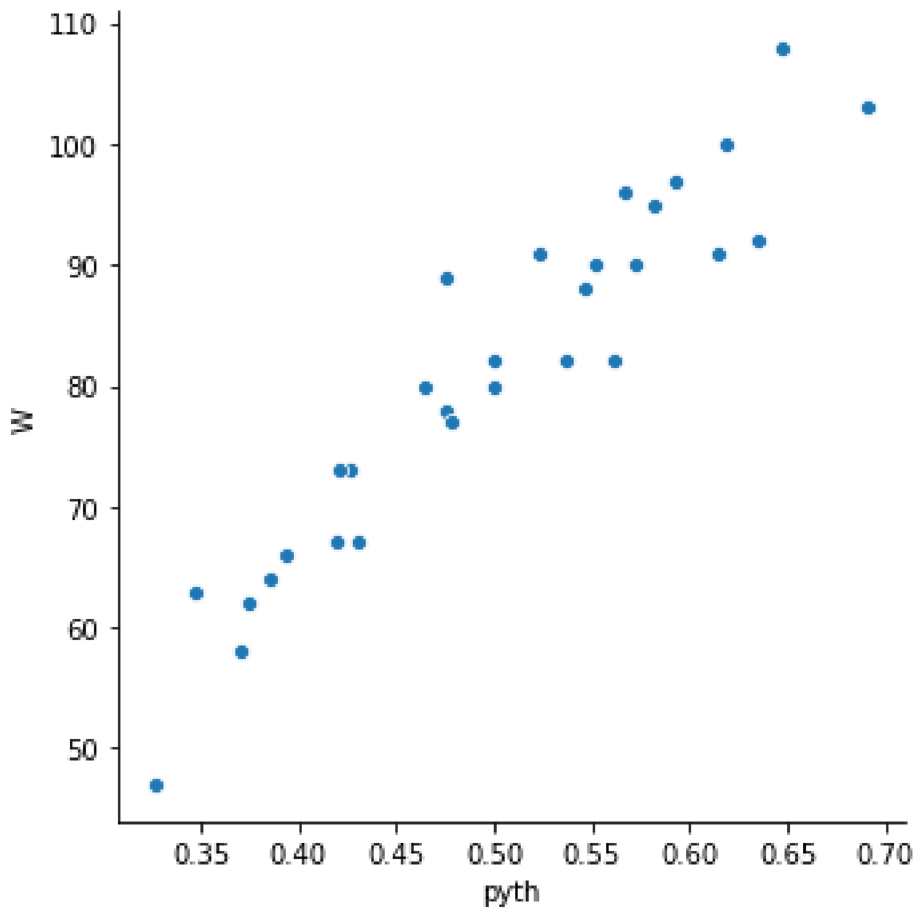
```
15  0.588957  0.566930
16  0.481481  0.475560
17  0.617284  0.618044
18  0.475309  0.477596
19  0.598765  0.592667
20  0.493827  0.463749
21  0.509317  0.499278
22  0.407407  0.392877
23  0.549383  0.475519
24  0.450617  0.426666
25  0.543210  0.546794
26  0.555556  0.551260
27  0.413580  0.430310
28  0.450617  0.420687
29  0.506173  0.561024
```

In [18]:

```
# Having prepared the data, we are now ready to examine it. First, we generate and xy
# This illustrates nicely the close correlation between win percentage and the Pythag

sns.relplot(x="pyth", y="wpc", data = MLB18)
```

Out[18]: <seaborn.axisgrid.FacetGrid at 0x7fcd94e0dac8>



### 4.1 Self test - 3

run sns.relplot again, but this time write y="W" instead of y="wpc". What do you find? Does it make a difference?

## 5 Finally we generate a regression.

The regression output tells you many things about the fitted relationship between win percentage and the Pythagorean Expectation. Regression is a method for identifying an equation which best fits the data. In this case that relationship is

wpc = Intercept + coef x pyth

You can see the value of Intercept is 0.0609 and coef is .8770. It's this latter value were interested in. It means that for every one unit increase in pyth, the value of wpc goes up by 0.887.

Two other points to note:

(i) The standard error (std err) gives us an idea of the precision of the estimate. The ratio of the coefficient (coef) to the standard error is called the t statistic (t) and its value informs us about statistical significance. This is illustrated by the p-value (P > |t|) - this is the probability that we would observe the value .8770 by chance, if the true value were really zero. This probability here is 0.000 - (this is not exactly zero, but the table doesn't include enough decimal places to show this) which means we can confident it is not zero. By convention, it is usual to conclude that we cannot be confident that the value of the coefficient is not zero if the p-value is greater than .05

(ii) in the top right hand corner of the table is the R-squared. This statistic tells you the percentage of variation in the y-variable (wpc) which can be accounted for by the variation in the x variables (pyth). R-squared can be thought of as a percentage - here the Pythagorean Expectation can account for 89.4% of the variation in win percentage.

```
In [ ]: # Finally we generate a regression.

        pyth_lm = smf.ols(formula = 'wpc ~ pyth', data=MLB18).fit()
        pyth_lm.summary()
```

### 5.1 Self test - 4

Run the regression above but instead write 'wpc ~ W' instead of 'wpc ~ pyth' in the line starting pyth_lm. What difference does this make?

## 6 Conclusion

This example was intended to get you started- don't worry if some things seem unclear - we're now going to conduct the same analysis for cricket, basketball, soccer and hockey. This will extend your understanding and help to make clear what we have just looked at.

A Useful Tip: when working in Python you will often come across problems that can be solved using methods you have encountered previously. It is often a good idea to return to an old notebook at a later stage to remind yourself how to code a particular problem.

```
In [ ]:
```