C3_M2 SportsStats Desc Stats

May 25, 2025

1 Descriptive Statistics

1.1 Step 1: Perform Initial Statistics

1.1.1 Hypothesis

1. Is there an advantage for athletes being from the host country? Do they win more?
2. Does it help or hurt for an athlete to compete in multiple events?
3. Is there a correlation between physical attributes and winning medals?

[1]:

```python
# Import all necessary libraries library

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

# Import the SQL library

from pandasql import sqldf
pysqldf = lambda q: sqldf(q, globals())
```

[2]:

```python
# Import the datasets

events = pd.read_csv('athlete_events.csv')
regions = pd.read_csv('noc_regions.csv')
```

[3]:

```python
events.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 271116 entries, 0 to 271115
Data columns (total 15 columns):
 #   Column  Non-Null Count   Dtype
---  ------  --------------   -----
 0   ID      271116 non-null  int64
 1   Name    271116 non-null  object
 2   Sex     271116 non-null  object
 3   Age     261642 non-null  float64
 4   Height  210945 non-null  float64
 5   Weight  208241 non-null  float64
 6   Team    271116 non-null  object
 7   NOC     271116 non-null  object
 8   Games   271116 non-null  object
 9   Year    271116 non-null  int64
 10  Season  271116 non-null  object
 11  City    271116 non-null  object
 12  Sport   271116 non-null  object
 13  Event   271116 non-null  object
 14  Medal   39783 non-null   object
dtypes: float64(3), int64(2), object(10)
memory usage: 31.0+ MB
```

1.2 Desciptive Stats Examples: Athlete Age, Height, Weight, and Sex

```
[3]: # Generate descriptive stats for the events table

     events.describe()
```

```
[3]:                   ID            Age         Height         Weight  \
     count  271116.000000  261642.000000  210945.000000  208241.000000
     mean    68248.954396      25.556898     175.338970      70.702393
     std     39022.286345       6.393561      10.518462      14.348020
     min         1.000000      10.000000     127.000000      25.000000
     25%     34643.000000      21.000000     168.000000      60.000000
     50%     68205.000000      24.000000     175.000000      70.000000
     75%    102097.250000      28.000000     183.000000      79.000000
     max    135571.000000      97.000000     226.000000     214.000000

                     Year
     count  271116.000000
     mean     1978.378480
     std        29.877632
     min      1896.000000
     25%      1960.000000
     50%      1988.000000
     75%      2002.000000
     max      2016.000000
```

```
[36]: # Descriptive stats for Age

      # Count
      ages_df = pysqldf('SELECT COUNT(DISTINCT Age) AS ages FROM events')
      ages = ages_df['ages'].iloc[0]

      # Mean
```

```python
mean_df = pysqldf('SELECT ROUND(AVG(Age), 0) AS avg_age FROM events')
mean=mean_df['avg_age'].iloc[0]

# Median
median_df = pysqldf('SELECT Age AS med_age FROM events ORDER BY Age LIMIT 1 OFFSET (SELECT COUNT(*) FROM events) / 2')
median=median_df['med_age'].iloc[0]

# Mode
mode_df = pysqldf('SELECT Age, COUNT(*) AS age_freq FROM events GROUP BY Age ORDER BY COUNT(*) DESC LIMIT 1')
mode_age = mode_df['Age'].iloc[0]
mode_freq = mode_df['age_freq'].iloc[0]

# Min & Max
min_df = pysqldf('SELECT MIN(Age) AS min_age FROM events')
min = min_df['min_age'].iloc[0]

max_df = pysqldf('SELECT MAX(Age) AS max_age FROM events')
max = max_df['max_age'].iloc[0]

print('Number of Ages: ', ages)
print('Average Age: ', mean)
print('Median Age: ', median)
print('Most Common Age: ', mode_age, ' (occurs', mode_freq, 'times)')
print('Youngest: ', min)
print('Oldest: ', max)
```

```
Number of Ages:  74
Average Age:  26.0
Median Age:  24.0
Most Common Age:  23.0  (occurs 21875 times)
Youngest:  10.0
Oldest:  97.0
```

[45]:

```python
# Descriptive stats for Height

# Count
hgts_df = pysqldf('SELECT COUNT(DISTINCT Height) AS hgts FROM events')
hgts = hgts_df['hgts'].iloc[0]

# Mean
mean_df = pysqldf('SELECT ROUND(AVG(Height), 0) AS avg_hgt FROM events')
mean=mean_df['avg_hgt'].iloc[0]

# Median
```

```
median_df = pysqldf('SELECT Height AS med_hgt FROM events ORDER BY Height LIMIT␣
 ↪1 OFFSET (SELECT COUNT(*) FROM events) / 2')
median=median_df['med_hgt'].iloc[0]

# Mode
mode_df = pysqldf('SELECT Height, COUNT(*) AS hgt_freq FROM events GROUP BY␣
 ↪Height ORDER BY COUNT(*) DESC LIMIT 1')
mode_hgt = mode_df['Height'].iloc[0]
mode_freq = mode_df['hgt_freq'].iloc[0]

# Min & Max
min_df = pysqldf('SELECT MIN(Height) AS min_hgt FROM events')
min = min_df['min_hgt'].iloc[0]

max_df = pysqldf('SELECT MAX(Height) AS max_hgt FROM events')
max = max_df['max_hgt'].iloc[0]

print('Number of Heights: ', hgts)
print('Average Height: ', mean)
print('Median Height: ', median)
print('Most Common Height: ', mode_hgt, ' (occurs', mode_freq, 'times)')
print('Shortest: ', min)
print('Tallest: ', max)
```

```
Number of Heights:  95
Average Height:  175.0
Median Height:  171.0
Most Common Height:  None  (occurs 60171 times)
Shortest:  127.0
Tallest:  226.0
```

[51]:

```
# Descriptive stats for Weight

# Count
wgts_df = pysqldf('SELECT COUNT(DISTINCT Weight) AS wgts FROM events')
wgts = wgts_df['wgts'].iloc[0]

# Mean
mean_df = pysqldf('SELECT ROUND(AVG(Weight), 0) AS avg_wgt FROM events')
mean=mean_df['avg_wgt'].iloc[0]

# Median
median_df = pysqldf('SELECT Weight AS med_wgt FROM events ORDER BY Weight LIMIT␣
 ↪1 OFFSET (SELECT COUNT(*) FROM events) / 2')
median=median_df['med_wgt'].iloc[0]

# Mode
```

```
mode_df = pysqldf('SELECT Weight, COUNT(*) AS wgt_freq FROM events GROUP BY␣
↪Weight ORDER BY COUNT(*) DESC LIMIT 1')
mode_wgt = mode_df['Weight'].iloc[0]
mode_freq = mode_df['wgt_freq'].iloc[0]

# Min & Max
min_df = pysqldf('SELECT MIN(Weight) AS min_wgt FROM events')
min = min_df['min_wgt'].iloc[0]

max_df = pysqldf('SELECT MAX(Weight) AS max_wgt FROM events')
max = max_df['max_wgt'].iloc[0]

print('Number of Weights: ', wgts)
print('Average Weight: ', mean)
print('Median Weight: ', median)
print('Most Common Weight: ', mode_wgt, ' (occurs', mode_freq, 'times)')
print('Lightest: ', min)
print('Heaviest: ', max)
```

```
Number of Weights:  220
Average Weight:  71.0
Median Weight:  64.0
Most Common Weight:  None  (occurs 62875 times)
Lightest:  25.0
Heaviest:  214.0
```

[5]:

```
# Descriptive stats for Sex

pysqldf('SELECT Sex, COUNT(Sex) AS athletes FROM events GROUP BY Sex ORDER BY␣
↪Sex')
```

[5]:

```
  Sex  athletes
0   F     74522
1   M    196594
```

1.3 Exploring Hypothesis 1

[5]:

```
# Match host City to its corresponding NOC

host_city_noc_map_data = {
    'Host_City': ['London', 'Athina', 'Sydney', 'Atlanta', 'Rio de Janeiro',␣
↪'Beijing', 'Barcelona', 'Los Angeles',
                  'Seoul', 'Munich', 'Montreal', 'Mexico City', 'Helsinki',␣
↪'Roma', 'Tokyo', 'Moskva', 'Paris',
                  'Berlin', 'Amsterdam', 'Sochi', 'Melbourne', 'Vancouver',␣
↪'Torino', 'Stockholm', 'Antwerpen',
                  'Salt Lake City', 'Innsbruck', 'Nagano', 'Albertville',␣
 ↪'Lillehammer', 'Calgary', 'Sarajevo',
                  'Lake Placid', 'Grenoble', 'Sankt Moritz', 'Sapporo',␣
 ↪"Cortina d'Ampezzo", 'St. Louis',
                  'Squaw Valley', 'Oslo', 'Garmisch-Partenkirchen', 'Chamonix'],
    'Host_NOC': ['GBR', 'GRE', 'ANZ', 'USA', 'BRA', 'CHN', 'ESP', 'USA',
                 'KOR', 'GER', 'CAN', 'MEX', 'FIN', 'ITA', 'JPN', 'RUS', 'FRA',
                 'GER', 'NED', 'RUS', 'ANZ', 'CAN', 'ITA', 'SWE', 'BEL',
                 'USA', 'AUT', 'JPN', 'FRA', 'NOR', 'CAN', 'BIH',
                 'USA', 'FRA', 'SWZ', 'JPN', 'ITA', 'USA',
                 'USA', 'NOR', 'GER', 'FRA']
}
host_city_noc_map = pd.DataFrame(host_city_noc_map_data)


# Return records where the host City is in an athlete's NOC

athletes_in_host_country = pysqldf('SELECT e.*, h.Host_NOC FROM events AS e␣
 ↪JOIN host_city_noc_map AS h ON e.City = h.Host_City WHERE e.NOC = h.
 ↪Host_NOC')

print("\nAthletes whose Home NOC matches the Host City's NOC:")
athletes_in_host_country
```

```
Athletes whose Home NOC matches the Host City's NOC:
```

[5]:

```
           ID                            Name Sex   Age  Height  Weight  \
0          10  Einar Ferdinand "Einari" Aalto   M  26.0     NaN     NaN
1          17         Paavo Johannes Aaltonen   M  32.0   175.0    64.0
2          17         Paavo Johannes Aaltonen   M  32.0   175.0    64.0
3          17         Paavo Johannes Aaltonen   M  32.0   175.0    64.0
4          17         Paavo Johannes Aaltonen   M  32.0   175.0    64.0
…       …                              …  ..   …     …      …
18511  135485          Stepan Olegovich Zuyev   M  25.0   189.0    90.0
18512  135485          Stepan Olegovich Zuyev   M  25.0   189.0    90.0
18513  135485          Stepan Olegovich Zuyev   M  25.0   189.0    90.0
18514  135539           Marius Edmund Zwiller   M  18.0     NaN     NaN
18515  135560               Stavroula Zygouri   F  36.0   171.0    63.0

          Team  NOC        Games  Year  Season      City          Sport  \
0      Finland  FIN  1952 Summer  1952  Summer  Helsinki       Swimming
1      Finland  FIN  1952 Summer  1952  Summer  Helsinki     Gymnastics
2      Finland  FIN  1952 Summer  1952  Summer  Helsinki     Gymnastics
3      Finland  FIN  1952 Summer  1952  Summer  Helsinki     Gymnastics
4      Finland  FIN  1952 Summer  1952  Summer  Helsinki     Gymnastics
…        …  …          …   …     …       …            …
```

```
18511   Russia  RUS  2014 Winter  2014  Winter     Sochi  Alpine Skiing
18512   Russia  RUS  2014 Winter  2014  Winter     Sochi  Alpine Skiing
18513   Russia  RUS  2014 Winter  2014  Winter     Sochi  Alpine Skiing
18514   France  FRA  1924 Summer  1924  Summer     Paris       Swimming
18515   Greece  GRE  2004 Summer  2004  Summer    Athina      Wrestling

                                           Event   Medal Host_NOC
0             Swimming Men's 400 metres Freestyle    None      FIN
1          Gymnastics Men's Individual All-Around    None      FIN
2                Gymnastics Men's Team All-Around  Bronze      FIN
3                 Gymnastics Men's Floor Exercise    None      FIN
4                    Gymnastics Men's Horse Vault    None      FIN
…                                               …      …      …
18511                   Alpine Skiing Men's Super G    None      RUS
18512              Alpine Skiing Men's Giant Slalom    None      RUS
18513                    Alpine Skiing Men's Slalom    None      RUS
18514        Swimming Men's 200 metres Breaststroke    None      FRA
18515     Wrestling Women's Middleweight, Freestyle    None      GRE

[18516 rows x 16 columns]
```

[6]:

```
# How many medals have been won in total?

pysqldf('SELECT COUNT(Medal) AS medals FROM events WHERE Medal IS NOT NULL')
```

[6]:

```
   medals
0   39783
```

[7]:

```
home_medal_pct = (18516/39783)*100
home_medal_pct
```

[7]:

```
46.542493024658775
```

We see that 18,516 of the 39,783 medals were won by athletes in their home country, or 46.5%. This is a significant correlation given the number of countries that participate in the Olympic Games.

1.4 Exploring Hypothesis 2

[9]:

```
# Find athletes who were in multiple events in the same year

mea = pysqldf('SELECT Name, Year, COUNT(*) AS event_count FROM events GROUP BY␣
 ↪Name, Year HAVING COUNT(*) > 1')
mea
```

[9]:

```
                                        Name  Year  event_count
0        Eleonora Margarida Josephina Scmitt  1948            2
```

```
1       Luis ngel Fernando de los Santos Grossi  1952            4
2                                   Th Ngn Thng  2008            5
3                                   Th Ngn Thng  2012            2
4                               A. Abdul Razzak  1960            2
…                                          …   …          …
47537                                yvind Berg  1994            3
47538                              yvind Tveter  1980            2
47539                                 zcan Ediz  1994            2
47540                              zdemir Akbal  2000            2
47541                                  zer Atei  1968            3

[47542 rows x 3 columns]
```

[10]:

```
# Which ones medaled?

mea_medals = pysqldf('SELECT Name, Year, COUNT(*) AS event_count, Medal FROM␣
 ↪events WHERE Medal IS NOT NULL GROUP BY Name, Year, Medal HAVING COUNT(*) >␣
 ↪1')
mea_medals
```

[10]:

```
                                   Name  Year  event_count   Medal
0      Aagje "Ada" Kok (-van der Linden)  1964            2  Silver
1                    Aaron Wells Peirsol  2004            3    Gold
2                    Aaron Wells Peirsol  2008            2    Gold
3                       Abelardo Olivier  1920            2    Gold
4                      Adam Henryk Maysz  2010            2  Silver
…                                 …   …          …     …
1856                   scar Cristi Gallo  1952            2  Silver
1857                        sten stensen  1920            2  Bronze
1858                        sten stensen  1920            2  Silver
1859        tienne Nol Henri Vandernotte  1936            2  Bronze
1860                       va Grard-Novk  1952            2  Silver

[1861 rows x 4 columns]
```

[12]:

```
# What percentage of them medaled?

mea_medals_pct = (1861/47542)*100
mea_medals_pct
```

[12]: 3.9144335534895465

[18]:

```
# Compare that to the percentage of all athletes who medaled

athletes = pysqldf('SELECT COUNT(*) AS athletes FROM events')
medals = pysqldf('SELECT COUNT(Medal) AS medals FROM events WHERE Medal IS NOT␣
 ↪NULL')
```

```
percent = (39783/271116)*100

print('Total Athletes: ', athletes)
print('Medals: ', medals)
print('Medal %: ', percent)
```

```
Total Athletes:     athletes
0    271116
Medals:     medals
0   39783
Medal %:  14.673792767671404
```

Competing in multiple events seems to lead to a medal rate of just 3.9%, whereas the general medal rate for all athletes is 14.7%.

1.5 Step 2: Evaluations

1.5.1 1. Provide a summary of the different descriptive statistics you looked at and WHY?

The descriptive stats used were count, mean, median, mode, min and max on all appropriate variables to learn the distribution and extremes of the data. Percentages and aggregates were used for non-numerical variables to check frequency and correlation.

1.5.2 2. Submit 2-3 key points you may have discovered about the data, i.e. new relationships? Aha's! Did you come up with additional ideas for other things to review?

1. The age range of athletes is much wider than expected: 10 - 97!
2. There is an apparent advantage to competing in your home country.
3. A significant number of athletes - near 1 in 6 - compete in multiple events at the same Games.

1.5.3 3. Did you prove or disprove any of your initial hypotheses? If so, which one(s) and what you plan to do next?

1. Is there an advantage for athletes being from the host country? Do they win more?

The data shows a significant correlation between medal winners competing in their home country. This hypothesis does bear out with the data.

2. Does it help or hurt for an athlete to compete in multiple events?

Preliminary findings indicate athletes who compete in multiple events medal at near 1/5 the average medal winning rate.

3. Is there a correlation between physical attributes and winning medals?

This is inconclusive so far. Further analysis is required.

1.5.4 4. What additional questions are you seeking to answer?

1. What are the demographic trends of Olympic athletes over time? Is the average height, weight or age changing?
2. What does country participation look like?
3. What are the medal count trends?
4. What events are most popular (have the most athletes competing)?

```
[ ]:
```