Beyond Descriptive Stats

In [63]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from pandasql import sqldf
pysqldf = lambda q: sqldf(q, globals())

%store -r athlete_events
%store -r summer_games
%store -r winter_games
%store -r female
%store -r male
```

In [64]:
```
#Calculating the ratios for Men vs Women
# Total Ratio
total_ratio = pysqldf('''SELECT Sex,
                                COUNT(*) Total_Count,
                                COUNT(*) * 100.00 / SUM(COUNT(*)) OVER () AS ratio
                                FROM athlete_events
                                GROUP BY sex''')

#Summer games ratio
summer_ratio = pysqldf('''SELECT Sex,
                                COUNT(*) Total_Count,
                                COUNT(*) * 100.00 / SUM(COUNT(*)) OVER () AS ratio
                                FROM summer_games
                                GROUP BY sex''')

#Winter games ratio
winter_ratio = pysqldf('''SELECT Sex,
                                COUNT(*) Total_Count,
                                COUNT(*) * 100.00 / SUM(COUNT(*)) OVER () AS ratio
                                FROM winter_games
                                GROUP BY sex''')
```

In [65]:
```
total_ratio.head()
```

Out[65]:

	Sex	Total_Count	ratio
0	F	74522	27.487127
1	M	196594	72.512873

In [66]:
```
summer_ratio.head()
```

Out[66]:

	Sex	Total_Count	ratio
0	F	59443	26.709713
1	M	163109	73.290287

In [67]:
```
winter_ratio.head()
```

Out[67]:

	Sex	Total_Count	ratio
0	F	15079	31.049749
1	M	33485	68.950251

In [68]:
```
#Calculating the total average for Men and Women
total_averages = pysqldf('''SELECT Sex,
                                    AVG(Weight) Average_Weight,
                                    AVG(Height) Average_Height,
                                    AVG(Age) Average_Age
                                FROM athlete_events
                                GROUP BY Sex''')

#Calculating the Summer average for Men and Women
summer_averages = pysqldf('''SELECT Sex,
                                    AVG(Weight) Average_Weight,
                                    AVG(Height) Average_Height,
                                    AVG(Age) Average_Age
                                FROM summer_games
                                GROUP BY Sex''')

#Calculating the Winter average for Men and Women
winter_averages = pysqldf('''SELECT Sex,
                                    AVG(Weight) Average_Weight,
                                    AVG(Height) Average_Height,
                                    AVG(Age) Average_Age
                                FROM winter_games
```

```
                                GROUP BY Sex''')
```

In [69]:
```
total_averages.head()
```

Out[69]:

	Sex	Average_Weight	Average_Height	Average_Age
0	F	60.021252	167.839740	23.732881
1	M	75.743677	178.858463	26.277562

In [70]:
```
summer_averages.head()
```

Out[70]:

	Sex	Average_Weight	Average_Height	Average_Age
0	F	60.087644	168.169025	23.660997
1	M	75.604195	178.901874	26.443944

In [71]:
```
winter_averages.head()
```

Out[71]:

	Sex	Average_Weight	Average_Height	Average_Age
0	F	59.755156	166.528250	24.014398
1	M	76.357058	178.668699	25.504261

In [72]:
```
#Total view of each year showing Participants and the number of medals awarded
pysqldf('''SELECT Year,
                  COUNT(*) Participants,
                  SUM(CASE WHEN Medal IS NOT NULL THEN 1 ELSE 0 END) AS Total_Medals,
                  SUM(CASE WHEN Medal = 'Gold' THEN 1 ELSE 0 END) AS Gold_Medals,
                  SUM(CASE WHEN Medal = 'Silver' THEN 1 ELSE 0 END) AS Silver_Medals,
                  SUM(CASE WHEN Medal = 'Bronze' THEN 1 ELSE 0 END) AS Bronze_Medals
           FROM athlete_events
           GROUP BY Year''')
```

Out[72]:

	Year	Participants	Total_Medals	Gold_Medals	Silver_Medals	Bronze_Medals
0	1896	380	143	62	43	38
1	1900	1936	604	201	228	175
2	1904	1301	486	173	163	150
3	1906	1733	458	157	156	145
4	1908	3101	831	294	281	256
5	1912	4040	941	326	315	300
6	1920	4292	1308	493	448	367
7	1924	5693	962	332	319	311
8	1928	5574	823	275	267	281
9	1932	3321	739	261	246	232
10	1936	7401	1025	348	347	330
11	1948	7480	987	330	332	325
12	1952	9358	1033	351	335	347
13	1956	6434	1043	353	342	348
14	1960	9235	1058	359	342	357
15	1964	9480	1215	408	406	401
16	1968	10479	1256	425	410	421
17	1972	11959	1414	474	455	485
18	1976	10502	1531	508	505	518
19	1980	8937	1602	529	531	542
20	1984	11588	1698	571	551	576
21	1988	14676	1845	607	601	637
22	1992	16413	2030	663	657	710
23	1994	3160	331	110	109	112
24	1996	13780	1842	608	605	629
25	1998	3605	440	145	145	150
26	2000	13821	2004	663	661	680
27	2002	4109	478	162	157	159
28	2004	13443	2001	664	660	677
29	2006	4382	526	176	175	175
30	2008	13602	2048	671	667	710
31	2010	4402	520	174	175	171
32	2012	12920	1941	632	630	679
33	2014	4891	597	202	197	198
34	2016	13688	2023	665	655	703

In [73]:
```
#Shows the % of medals won compared to the total participants e.g
# 2016 - 14.77% of participants received a medal
# 2016 - 4.85% of participants received a Gold medal
# 2016 - 4.78% of participants received a Silver medal
# 2016 - 5.13% of participants received a Bronze medal
total_ratio_medals = pysqldf('''SELECT Year,
                                       CAST(Total_Medals AS FLOAT) * 100 / Participants AS Total_Medals,
                                       CAST(Gold_Medals AS FLOAT) * 100 / Participants AS Gold_Medals,
                                       CAST(Silver_Medals AS FLOAT) * 100 / Participants AS Silver_Medals,
                                       CAST(Bronze_Medals AS FLOAT) * 100 / Participants AS Bronze_Medals
                                FROM
                                   (
                                   SELECT Year,
                                          COUNT(*) Participants,
                                          SUM(CASE WHEN Medal IS NOT NULL THEN 1 ELSE 0 END) AS Total_Medals,
                                          SUM(CASE WHEN Medal = 'Gold' THEN 1 ELSE 0 END) AS Gold_Medals,
                                          SUM(CASE WHEN Medal = 'Silver' THEN 1 ELSE 0 END) AS Silver_Medals,
                                          SUM(CASE WHEN Medal = 'Bronze' THEN 1 ELSE 0 END) AS Bronze_Medals
                                   FROM athlete_events
                                   GROUP BY Year) Medals_Table
                                   ''')
```

In [74]:
```
total_ratio_medals
```

Out[74]:

	Year	Total_Medals	Gold_Medals	Silver_Medals	Bronze_Medals
0	1896	37.631579	16.315789	11.315789	10.000000
1	1900	31.198347	10.382231	11.776860	9.039256
2	1904	37.355880	13.297463	12.528824	11.529593
3	1906	26.428159	9.059435	9.001731	8.366994
4	1908	26.797807	9.480813	9.061593	8.255401
5	1912	23.292079	8.069307	7.797030	7.425743
6	1920	30.475303	11.486486	10.438024	8.550792
7	1924	16.897945	5.831723	5.603373	5.462849
8	1928	14.764980	4.933620	4.790097	5.041263
9	1932	22.252334	7.859079	7.407407	6.985848
10	1936	13.849480	4.702067	4.688556	4.458857
11	1948	13.195187	4.411765	4.438503	4.344920
12	1952	11.038683	3.750801	3.579825	3.708057
13	1956	16.210755	5.486478	5.315511	5.408766
14	1960	11.456416	3.887385	3.703303	3.865728
15	1964	12.816456	4.303797	4.282700	4.229958
16	1968	11.985877	4.055731	3.912587	4.017559
17	1972	11.823731	3.963542	3.804666	4.055523
18	1976	14.578176	4.837174	4.808608	4.932394
19	1980	17.925478	5.919212	5.941591	6.064675
20	1984	14.653089	4.927511	4.754919	4.970659
21	1988	12.571545	4.136004	4.095121	4.340420
22	1992	12.368245	4.039481	4.002925	4.325839
23	1994	10.474684	3.481013	3.449367	3.544304
24	1996	13.367199	4.412192	4.390421	4.564586
25	1998	12.205270	4.022191	4.022191	4.160888
26	2000	14.499674	4.797048	4.782577	4.920049
27	2002	11.633001	3.942565	3.820881	3.869555
28	2004	14.885070	4.939374	4.909618	5.036078
29	2006	12.003651	4.016431	3.993610	3.993610
30	2008	15.056609	4.933098	4.903691	5.219821
31	2010	11.812812	3.952749	3.975466	3.884598
32	2012	15.023220	4.891641	4.876161	5.255418
33	2014	12.206093	4.130035	4.027806	4.048252
34	2016	14.779369	4.858270	4.785213	5.135885

In [75]:
```
summer_games.head()
```

Out[75]:

	ID	Name	Sex	Age	Height	Weight	Team	NOC	Games	Year	Season	City	Sport	Event	Medal
0	1	A Dijiang	M	24.0	180.0	80.0	China	CHN	1992 Summer	1992	Summer	Barcelona	Basketball	Basketball Men's Basketball	None
1	2	A Lamusi	M	23.0	170.0	60.0	China	CHN	2012 Summer	2012	Summer	London	Judo	Judo Men's Extra-Lightweight	None
2	3	Gunnar Nielsen Aaby	M	24.0	NaN	NaN	Denmark	DEN	1920 Summer	1920	Summer	Antwerpen	Football	Football Men's Football	None
3	4	Edgar Lindenau Aabye	M	34.0	NaN	NaN	Denmark/Sweden	DEN	1900 Summer	1900	Summer	Paris	Tug-Of-War	Tug-Of-War Men's Tug-Of-War	Gold
4	8	Cornelia "Cor" Aalten (-Strannood)	F	18.0	168.0	NaN	Netherlands	NED	1932 Summer	1932	Summer	Los Angeles	Athletics	Athletics Women's 100 metres	None

In [76]:
```
#Calculating the % of medals won for the Summer Olympic Games
summer_ratio_medals = pysqldf('''SELECT Year,
                                        CAST(Total_Medals AS FLOAT) * 100 / Participants AS Total_Medals,
                                        CAST(Gold_Medals AS FLOAT) * 100 / Participants AS Gold_Medals,
```

```
                                        CAST(Silver_Medals AS FLOAT) * 100 / Participants AS Silver_Medals,
                                        CAST(Bronze_Medals AS FLOAT) * 100 / Participants AS Bronze_Medals
                                    FROM
                                       (
                                       SELECT Year,
                                              COUNT(*) Participants,
                                              SUM(CASE WHEN Medal IS NOT NULL THEN 1 ELSE 0 END) AS Total_Medals,
                                              SUM(CASE WHEN Medal = 'Gold' THEN 1 ELSE 0 END) AS Gold_Medals,
                                              SUM(CASE WHEN Medal = 'Silver' THEN 1 ELSE 0 END) AS Silver_Medals,
                                              SUM(CASE WHEN Medal = 'Bronze' THEN 1 ELSE 0 END) AS Bronze_Medals
                                       FROM summer_games
                                       GROUP BY Year) Medals_Table
                                       ''')
```

In [77]:
```
#Calculating the % of medals won for the Winter Olympic Games
winter_ratio_medals = pysqldf('''SELECT Year,
                                        CAST(Total_Medals AS FLOAT) * 100 / Participants AS Total_Medals,
                                        CAST(Gold_Medals AS FLOAT) * 100 / Participants AS Gold_Medals,
                                        CAST(Silver_Medals AS FLOAT) * 100 / Participants AS Silver_Medals,
                                        CAST(Bronze_Medals AS FLOAT) * 100 / Participants AS Bronze_Medals
                                    FROM
                                       (
                                       SELECT Year,
                                              COUNT(*) Participants,
                                              SUM(CASE WHEN Medal IS NOT NULL THEN 1 ELSE 0 END) AS Total_Medals,
                                              SUM(CASE WHEN Medal = 'Gold' THEN 1 ELSE 0 END) AS Gold_Medals,
                                              SUM(CASE WHEN Medal = 'Silver' THEN 1 ELSE 0 END) AS Silver_Medals,
                                              SUM(CASE WHEN Medal = 'Bronze' THEN 1 ELSE 0 END) AS Bronze_Medals
                                       FROM winter_games
                                       GROUP BY Year) Medals_Table
                                       ''')
```

In [78]:
```
summer_ratio_medals
```

Out[78]:

	Year	Total_Medals	Gold_Medals	Silver_Medals	Bronze_Medals
0	1896	37.631579	16.315789	11.315789	10.000000
1	1900	31.198347	10.382231	11.776860	9.039256
2	1904	37.355880	13.297463	12.528824	11.529593
3	1906	26.428159	9.059435	9.001731	8.366994
4	1908	26.797807	9.480813	9.061593	8.255401
5	1912	23.292079	8.069307	7.797030	7.425743
6	1920	30.475303	11.486486	10.438024	8.550792
7	1924	15.899102	5.293331	5.369769	5.236002
8	1928	14.703526	4.907853	4.787660	5.008013
9	1932	21.791849	7.713035	7.207814	6.871000
10	1936	14.094682	4.795573	4.764832	4.534276
11	1948	13.302108	4.512100	4.434036	4.355972
12	1952	10.846433	3.700121	3.518742	3.627570
13	1956	17.417593	5.890384	5.714843	5.812366
14	1960	11.220594	3.805887	3.621136	3.793571
15	1964	13.360166	4.505323	4.401454	4.453389
16	1968	12.307871	4.180252	3.959013	4.168607
17	1972	11.791537	3.920807	3.804348	4.066382
18	1976	15.276010	5.068858	5.022567	5.184585
19	1980	19.246280	6.355166	6.369072	6.522041
20	1984	15.612439	5.257034	5.045483	5.309922
21	1988	13.142810	4.320013	4.261859	4.560937
22	1992	13.192571	4.307621	4.230562	4.654389
23	1996	13.367199	4.412192	4.390421	4.564586
24	2000	14.499674	4.797048	4.782577	4.920049
25	2004	14.885070	4.939374	4.909618	5.036078
26	2008	15.056609	4.933098	4.903691	5.219821
27	2012	15.023220	4.891641	4.876161	5.255418
28	2016	14.779369	4.858270	4.785213	5.135885

In [79]:
```
winter_ratio_medals.head()
```

Out[79]:

	Year	Total_Medals	Gold_Medals	Silver_Medals	Bronze_Medals
0	1924	28.260870	11.956522	8.260870	8.043478
1	1928	15.292096	5.154639	4.810997	5.326460
2	1932	26.136364	9.090909	9.090909	7.954545
3	1936	12.067039	4.022346	4.134078	3.910615
4	1948	12.558140	3.813953	4.465116	4.279070

As you can see here the winter Olympic games did not start until 1924 whereas the summer games started on 1896. I will have to create a new dataframe with them both commencing at the same time to help figure out and determine any correlation.

In [80]:
```
summer_medals_count = pysqldf('''SELECT Year,
                                        COUNT(*) Participants,
                                        SUM(CASE WHEN Medal IS NOT NULL THEN 1 ELSE 0 END) AS Total_Medals,
                                        SUM(CASE WHEN Medal = 'Gold' THEN 1 ELSE 0 END) AS Gold_Medals,
                                        SUM(CASE WHEN Medal = 'Silver' THEN 1 ELSE 0 END) AS Silver_Medals,
                                        SUM(CASE WHEN Medal = 'Bronze' THEN 1 ELSE 0 END) AS Bronze_Medals
                                FROM summer_games
                                GROUP BY Year''')

winter_medals_count = pysqldf('''SELECT Year,
                                        COUNT(*) Participants,
                                        SUM(CASE WHEN Medal IS NOT NULL THEN 1 ELSE 0 END) AS Total_Medals,
                                        SUM(CASE WHEN Medal = 'Gold' THEN 1 ELSE 0 END) AS Gold_Medals,
                                        SUM(CASE WHEN Medal = 'Silver' THEN 1 ELSE 0 END) AS Silver_Medals,
                                        SUM(CASE WHEN Medal = 'Bronze' THEN 1 ELSE 0 END) AS Bronze_Medals
                                FROM winter_games
                                GROUP BY Year''')
```

In [81]:
```
summer_medals_count.head()
```

Out[81]:

	Year	Participants	Total_Medals	Gold_Medals	Silver_Medals	Bronze_Medals
0	1896	380	143	62	43	38
1	1900	1936	604	201	228	175
2	1904	1301	486	173	163	150
3	1906	1733	458	157	156	145
4	1908	3101	831	294	281	256

In [82]:
```
winter_medals_count.head()
```

Out[82]:

	Year	Participants	Total_Medals	Gold_Medals	Silver_Medals	Bronze_Medals
0	1924	460	130	55	38	37
1	1928	582	89	30	28	31
2	1932	352	92	32	32	28
3	1936	895	108	36	37	35
4	1948	1075	135	41	48	46

In [83]:
```
summer_medals_count_new = summer_medals_count[7:]
```

In [84]:
```
summer_medals_count_new
```

Out[84]:

	Year	Participants	Total_Medals	Gold_Medals	Silver_Medals	Bronze_Medals
7	1924	5233	832	277	281	274
8	1928	4992	734	245	239	250
9	1932	2969	647	229	214	204
10	1936	6506	917	312	310	295
11	1948	6405	852	289	284	279
12	1952	8270	897	306	291	300
13	1956	5127	893	302	293	298
14	1960	8119	911	309	294	308
15	1964	7702	1029	347	339	343
16	1968	8588	1057	359	340	358
17	1972	10304	1215	404	392	419
18	1976	8641	1320	438	434	448
19	1980	7191	1384	457	458	469
20	1984	9454	1476	497	477	502
21	1988	12037	1582	520	513	549
22	1992	12977	1712	559	549	604
23	1996	13780	1842	608	605	629
24	2000	13821	2004	663	661	680
25	2004	13443	2001	664	660	677
26	2008	13602	2048	671	667	710
27	2012	12920	1941	632	630	679
28	2016	13688	2023	665	655	703

In [85]:
```
x = winter_medals_count.Total_Medals
y = summer_medals_count_new.Total_Medals
cor = np.corrcoef(x, y)
```

In [86]:
```
print(cor)
```

```
[[1.         0.94141801]
 [0.94141801 1.        ]]
```

The Pearon correlation coefficient between the total number of medals in the winter and summer olympics from 1924 to 2016, is 0.94, which is highly positive. Therefore, the performance of a country in winter olympics is highly correlated to that in summer olympics

In [87]:
```
std_medal_counts_summer_olympics = np.std(y)
std_medal_counts_winter_olympics = np.std(x)
```

In [88]:
```
print("Standard Deviation for Summer Olympics Medals =",std_medal_counts_summer_olympics)
print("Standard Deviation for Winter Olympics Medals =",std_medal_counts_winter_olympics)
```

```
Standard Deviation for Summer Olympics Medals = 475.323015441357
Standard Deviation for Winter Olympics Medals = 152.56899942903493
```

From 1924 to 2016, as the standard deviation in the summer olympics is about 3 times that in the winter olympics, country performance by year change more in Summer Olympics.