Homework3

September 27, 2024

1 Homework 3 -OLYMPICS

[1]:
```
# Please provide the information to identify this assignment in this comment␣
 ↪cell/section
# Student name: Mrudu lahari Malayanur
# Assignment submisision date:
```

[2]:
```
#importing necessary libraries
import pandas as pd
import numpy as np
import matplotlib
import matplotlib.pyplot as plt
```

[3]:
```
# NOTE: Please make sure that the olympics1992_2008.zip file has been uploaded␣
 ↪to the same folder where you have this
#       notebook file
odata = pd.read_csv('olympics1992_2008.zip',skiprows=4)
```

[4]:
```
# Start exploratory data analysis
odata.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9619 entries, 0 to 9618
Data columns (total 10 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   City          9619 non-null   object
 1   Edition       9619 non-null   int64
 2   Sport         9619 non-null   object
 3   Discipline    9619 non-null   object
 4   Athlete       9619 non-null   object
 5   NOC           9619 non-null   object
 6   Gender        9619 non-null   object
 7   Event         9619 non-null   object
 8   Event_gender  9619 non-null   object
 9   Medal         9619 non-null   object
dtypes: int64(1), object(9)
memory usage: 751.6+ KB
```

[5]:
```
odata.head()
```

[5]:

	City	Edition	Sport	Discipline	Athlete	NOC	Gender	Event	Event_gender	Medal
0	Barcelona	1992	Aquatics	Diving	XIONG, Ni	CHN	Men	10m platform	M	Bronze
1	Barcelona	1992	Aquatics	Diving	SUN, Shuwei	CHN	Men	10m platform	M	Gold
2	Barcelona	1992	Aquatics	Diving	DONIE, Scott R.	USA	Men	10m platform	M	Silver
3	Barcelona	1992	Aquatics	Diving	CLARK, Mary Ellen	USA	Women	10m platform	W	Bronze
4	Barcelona	1992	Aquatics	Diving	FU, Mingxia	CHN	Women	10m platform	W	Gold

[6]:
```
# Add cells with any additional exploratory data analysis commands/functions
 ↪that you think are necessary. This will
# not be graded but will help you in solving this homework's tasks
# Hint.. get the unique entries for columns of interest
```

[7]:
```
#understanding different Disciplines of participants and number of people under
 ↪each category
odata['Discipline'].value_counts()
```

[7]:
```
Discipline
Swimming           920
Athletics          902
Rowing             732
Hockey             481
Football           455
Handball           443
Artistic G.        373
Volleyball         359
Basketball         358
Canoe / Kayak F    345
Baseball           335
Fencing            321
Water polo         311
Judo               280
Sailing            261
Boxing             232
Shooting           231
Cycling Track      229
Weightlifting      194
Softball           180
Wrestling Free.    161
```

```
Synchronized S.    135
Wrestling Gre-R    132
Diving             132
Badminton          120
Archery            120
Table Tennis       102
Tennis              94
Rhythmic G.         87
Eventing            81
Taekwondo           80
Canoe / Kayak S     75
Jumping             74
Dressage            72
Cycling Road        65
Beach volley.       48
Modern Pentath.     33
Mountain Bike       24
Trampoline          18
Triathlon           18
BMX                  6
Name: count, dtype: int64
```

[8]:
```
#finding the category of participants based
odata['Event_gender'].value_counts()
#there are 5259 male participants, 3992 female participants, and 368 others
```

[8]:
```
Event_gender
M    5259
W    3992
X     368
Name: count, dtype: int64
```

Solve the following tasks. You can add as many additional cells as you need to solve each one of them.

1.1 Task #1

a) List the 5 countries that accumulated the most medals across all the olympic game editions covered in the dataset

b) List the 5 countries that accumulated the most GOLD medals across all the olympic game editions covered in the dataset

[9]:
```
#a) List the 5 countries that accumulated the most medals across all the␣
 ↪olympic game editions covered in the dataset
countries=odata['NOC'].value_counts()
countries.head(5)
```

```
#the value_counts gives the unique values in the decending order. Hence taking␣
 ↪the first 5 entries using head(5) to get top 5 countries with most medals
```

[9]:
```
NOC
USA    1311
GER     691
AUS     678
RUS     638
CHN     550
Name: count, dtype: int64
```

[10]:
```
#b) List the 5 countries that accumulated the most GOLD medals across all the␣
 ↪olympic game editions covered in the dataset
gold_medalists=odata[odata['Medal']=='Gold']
top_gold_medalist_countries=gold_medalists['NOC'].value_counts().head(5)
top_gold_medalist_countries

#applied boolean check operation on the dataset using odata['Medal']=='Gold'␣
 ↪condition and getting the countries that has achieved gold medals.
```

[10]:
```
NOC
USA    620
GER    237
CHN    202
RUS    192
AUS    186
Name: count, dtype: int64
```

[11]:
```
#different types of events
odata['Event'].value_counts()
```

[11]:
```
Event
hockey                                  481
football                                455
handball                                443
volleyball                              359
basketball                              358
                                       ...
90 - 100kg, total (first-heavyweight)     3
82.5 - 90kg, total (middle-heavyweight)   3
67.5 - 75kg, total (middleweight)         3
56 - 60kg, total (featherweight)          3
75 - 82.5kg, total (light-heavyweight)    2
Name: count, Length: 277, dtype: int64
```

1.2 Task #2

List the number of Gold, Silver and Bronze medals obtained by Women and Men across all the olympic game editions covered in the dataset

[12]:
```
#filtering the dataset into 2, women category, men category
women_winners=odata[odata['Event_gender']=='W']
men_winners = odata[odata['Event_gender']=='M']

#extracting medal categories for men and women
print("WOMEN Category ","\n",women_winners['Medal'].value_counts())
print("\n")
print("MEN Category","\n",men_winners['Medal'].value_counts())
```

```
WOMEN Category
 Medal
Bronze    1353
Gold      1326
Silver    1313
Name: count, dtype: int64


MEN Category
 Medal
Bronze    1829
Gold      1715
Silver    1715
Name: count, dtype: int64
```

1.3 Task #3

List the names of the 5 male athletes and 5 female athletes that obtained the most medals across all the olympic game editions covered in the dataset

[13]:
```
#getting the list of top 5 Athletes using head(5) in value_counts that give␣
 ↪data in decending order. Using index.tolist() to get only the list of names␣
 ↪and not the numbers associated with them

top_5_women_Athletes=women_winners['Athlete'].value_counts().head(5).index.
 ↪tolist()

top_5_men_Athletes=men_winners['Athlete'].value_counts().head(5).index.tolist()


print("Top women Athletes: ","\n",top_5_women_Athletes)
print("\n")
print("Top men Athletes: ","\n",top_5_men_Athletes)
```

```
Top women Athletes:
 ['THOMPSON, Jenny', 'COUGHLIN, Natalie', 'VAN ALMSICK, Franziska', 'TORRES,
Dara', 'JONES, Leisel']


Top men Athletes:
 ['PHELPS, Michael', 'NEMOV, Alexei', 'HALL, Gary Jr.', 'SCHERBO, Vitaly',
'THORPE, Ian']
```

1.4 Task #4

Provide two additional analysis results that you can derive from the dataset (they must be different than those obtained in tasks 1 to 3). The results can include graphs (but it is not required). Describe the results obtained in the cell provided for that purpose

[14]:
```
#Given that USA is the country with most medals, which event did it win the␣
 ↪most medals for?

USA_medals = odata[odata['NOC']=='USA']
USA_medals['Event'].value_counts().head(1)
#USA won most, 120 medals in basketball game
```

[14]:
```
Event
basketball    120
Name: count, dtype: int64
```

[15]:
```
#RESULT:
#From the above result, basketball stands to be the most winning sport for USA␣
 ↪with 120 medals
```

[16]:
```
#seeing least 5 sports in the year 1992, to analyse the trend and understand if␣
 ↪they have become popular in 2008?
edition_1992=odata[odata['Edition']==1992]
edition_1992['Sport'].value_counts()
```

[16]:
```
Sport
Aquatics            228
Athletics           178
Rowing              156
Hockey               96
Canoe / Kayak        84
Handball             84
Gymnastics           80
Volleyball           72
Fencing              72
Basketball           72
Wrestling            60
Baseball             60
```

```
Judo                56
Sailing             51
Cycling             50
Boxing              48
Equestrian          45
Shooting            39
Football            37
Weightlifting       29
Archery             24
Badminton           24
Table Tennis        24
Tennis              24
Modern Pentathlon   12
Name: count, dtype: int64
```

RESULTS ANALYSIS Summarize your findings here.

[17]:
```
# getting data for 2008
edition_2008=odata[odata['Edition']==2008]
edition_2008['Sport'].value_counts()
```

[17]:
```
Sport
Aquatics            347
Athletics           177
Rowing              144
Football            108
Gymnastics           99
Hockey               98
Handball             85
Volleyball           84
Canoe / Kayak        84
Basketball           72
Baseball             72
Wrestling            71
Cycling              71
Fencing              62
Judo                 56
Sailing              54
Softball             45
Equestrian           45
Weightlifting        45
Shooting             45
Boxing               44
Taekwondo            32
Table Tennis         24
Archery              24
Badminton            24
```

```
Tennis                18
Triathlon              6
Modern Pentathlon      6
Name: count, dtype: int64
```

RESULT

From the above analysis, we can see that, the least popular sport Modern Pentathlon has not become quite popular. However, Football has become one amongst the top 5 popular sports by 2008. Aquatics remains to be the most popular and Modern Pentathlon the least.

2 PART 2

Explore the dataset of Paris Olympics medallists. See https://www.kaggle.com/datasets/piterfm/paris-2024-olympic-summer-games?select=medallists.csv for more information. Describe your exploration and observations of **TWO** unique findings.

[18]:
```
#reading paris dataset
paris_olympics=pd.read_csv("Parismedallists.csv")
paris_olympics.head()
```

[18]:

	medal_date	medal_type	medal_code	name	gender	country_code	…
0	2024-07-27	Gold Medal	1.0	EVENEPOEL Remco	Male	BEL	
1	2024-07-27	Silver Medal	2.0	GANNA Filippo	Male	ITA	
2	2024-07-27	Bronze Medal	3.0	van AERT Wout	Male	BEL	
3	2024-07-27	Gold Medal	1.0	BROWN Grace	Female	AUS	
4	2024-07-27	Silver Medal	2.0	HENDERSON Anna	Female	GBR	

	country	country_long	nationality_code	nationality	…	team
0	Belgium	Belgium	BEL	Belgium	…	NaN
1	Italy	Italy	ITA	Italy	…	NaN
2	Belgium	Belgium	BEL	Belgium	…	NaN
3	Australia	Australia	AUS	Australia	…	NaN
4	Great Britain	Great Britain	GBR	Great Britain	…	NaN

	team_gender	discipline	event	event_type
0	NaN	Cycling Road	Men's Individual Time Trial	ATH
1	NaN	Cycling Road	Men's Individual Time Trial	ATH
2	NaN	Cycling Road	Men's Individual Time Trial	ATH
3	NaN	Cycling Road	Women's Individual Time Trial	ATH
4	NaN	Cycling Road	Women's Individual Time Trial	ATH

	url_event	birth_date	code_athlete
0	/en/paris-2024/results/cycling-road/men-s-indi…	2000-01-25	1903136
1	/en/paris-2024/results/cycling-road/men-s-indi…	1996-07-25	1923520
2	/en/paris-2024/results/cycling-road/men-s-indi…	1994-09-15	1903147
3	/en/paris-2024/results/cycling-road/women-s-in…	1992-07-07	1940173

```
4  /en/paris-2024/results/cycling-road/women-s-in…  1998-11-14      1912525

   code_team is_medallist
0        NaN         True
1        NaN         True
2        NaN         True
3        NaN         True
4        NaN         True

[5 rows x 21 columns]
```

[19]:
```
#using info to learnt the data columns
paris_olympics.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2315 entries, 0 to 2314
Data columns (total 21 columns):
 #   Column            Non-Null Count  Dtype
---  ------            --------------  -----
 0   medal_date        2315 non-null   object
 1   medal_type        2315 non-null   object
 2   medal_code        2314 non-null   float64
 3   name              2315 non-null   object
 4   gender            2315 non-null   object
 5   country_code      2315 non-null   object
 6   country           2315 non-null   object
 7   country_long      2315 non-null   object
 8   nationality_code  2314 non-null   object
 9   nationality       2314 non-null   object
 10  nationality_long  2314 non-null   object
 11  team              1555 non-null   object
 12  team_gender       1555 non-null   object
 13  discipline        2315 non-null   object
 14  event             2315 non-null   object
 15  event_type        2315 non-null   object
 16  url_event         2294 non-null   object
 17  birth_date        2315 non-null   object
 18  code_athlete      2315 non-null   int64
 19  code_team         1555 non-null   object
 20  is_medallist      2315 non-null   bool
dtypes: bool(1), float64(1), int64(1), object(18)
memory usage: 364.1+ KB
```

[20]:
```
paris_olympics.describe()
```

[20]:
```
         medal_code  code_athlete
count  2314.000000  2.315000e+03
```

```
mean      2.023336  1.893321e+06
std       0.820390  2.628276e+05
min       1.000000  1.532872e+06
25%       1.000000  1.896552e+06
50%       2.000000  1.924464e+06
75%       3.000000  1.950498e+06
max       3.000000  4.980004e+06
```

[21]:
```
#out of 2315 participants, what % of them were medallists?
len(paris_olympics[paris_olympics['is_medallist']])/
 ↪len(paris_olympics['name'])*100
```

[21]:
```
97.96976241900649
```

about 97% of participants are medallists

[22]:
```
#lets see the distribution of participants based on medals

(paris_olympics['medal_type'].value_counts()/paris_olympics['medal_type'].
 ↪count())*100
```

[22]:
```
medal_type
Bronze Medal    34.859611
Silver Medal    32.656587
Gold Medal      32.483801
Name: count, dtype: float64
```

[23]:
```
#get the youngest and oldest athlet
```

[24]:
```
#getting youngest participant by taking the min value on birth date
youngest = paris_olympics['birth_date'].min()
paris_olympics[paris_olympics['birth_date']==youngest]
```

[24]:
```
      medal_date    medal_type  medal_code         name  gender country_code  \
1126  2024-08-02  Silver Medal         2.0  KRAUT Laura  Female          USA

            country              country_long nationality_code    nationality  \
1126  United States  United States of America             USA  United States

      …                      team team_gender  discipline         event  \
1126  …  United States of America           0  Equestrian  Jumping Team

     event_type                                          url_event  \
1126       TEAM  /en/paris-2024/results/equestrian/jumping-team…

      birth_date code_athlete          code_team is_medallist
1126  1965-11-14      1951840  EQUOJUMPTEAMUSA01         True
```

```
[1 rows x 21 columns]
```

[25]:
```
#getting oldest participant by taking the min value on birth date
oldest = paris_olympics['birth_date'].max()
paris_olympics[paris_olympics['birth_date']==oldest]
```

[25]:
```
     medal_date  medal_type  medal_code        name  gender country_code  \
432  2024-08-06  Gold Medal         1.0  TREW Arisa  Female          AUS

       country country_long nationality_code nationality  … team  \
432  Australia    Australia              AUS   Australia  …  NaN

    team_gender     discipline         event event_type  \
432         NaN  Skateboarding  Women's Park        ATH

                                           url_event  birth_date  \
432  /en/paris-2024/results/skateboarding/women-s-p…  2010-05-12

    code_athlete  code_team is_medallist
432      1946064        NaN         True

[1 rows x 21 columns]
```