Rodriguez_Felipe_DSC530_1.2Exercise

March 14, 2023

Week 1 Assignment

Load the dataset as a Pandas data frame.

[1]:
```
# Read in Pandas Library
import pandas as pd
```

[118]:
```
# Establish Data Variable
data = 'Video_Games_Sales_as_at_22_Dec_2016.csv'
```

[4]:
```
# Loads the data as Pandas Data Frame
df = pd.read_csv(data)
```

Display the first ten rows of data.

[5]:
```
# Displays first ten rows
df.head(10)
```

[5]:

	Name	Platform	Year_of_Release	Genre	Publisher	NA_Sales	EU_Sales	JP_Sales	Other_Sales	Global_Sales
0	Wii Sports	Wii	2006.0	Sports	Nintendo	41.36	28.96	3.77	8.45	82.53
1	Super Mario Bros.	NES	1985.0	Platform	Nintendo	29.08	3.58	6.81	0.77	40.24
2	Mario Kart Wii	Wii	2008.0	Racing	Nintendo	15.68	12.76	3.79	3.29	35.52
3	Wii Sports Resort	Wii	2009.0	Sports	Nintendo	15.61	10.93	3.28	2.95	32.77
4	Pokemon Red/Pokemon Blue	GB	1996.0	Role-Playing	Nintendo	11.27	8.89	10.22	1.00	31.37
5	Tetris	GB	1989.0	Puzzle	Nintendo	23.20	2.26	4.22	0.58	30.26
6	New Super Mario Bros.	DS	2006.0	Platform	Nintendo	11.28	9.14	6.50	2.88	29.80
7	Wii Play	Wii	2006.0	Misc	Nintendo	13.96	9.18	2.93	2.84	28.92
8	New Super Mario Bros. Wii	Wii	2009.0	Platform	Nintendo					
9	Duck Hunt	NES	1984.0	Shooter						

```
8  Nintendo     14.44      6.94      4.70         2.24         28.32
9  Nintendo     26.93      0.63      0.28         0.47         28.31

   Critic_Score  Critic_Count User_Score  User_Count Developer Rating
0          76.0          51.0          8       322.0  Nintendo      E
1           NaN           NaN        NaN         NaN       NaN    NaN
2          82.0          73.0        8.3       709.0  Nintendo      E
3          80.0          73.0          8       192.0  Nintendo      E
4           NaN           NaN        NaN         NaN       NaN    NaN
5           NaN           NaN        NaN         NaN       NaN    NaN
6          89.0          65.0        8.5       431.0  Nintendo      E
7          58.0          41.0        6.6       129.0  Nintendo      E
8          87.0          80.0        8.4       594.0  Nintendo      E
9           NaN           NaN        NaN         NaN       NaN    NaN
```

Find the dimensions (number of rows and columns) in the data frame. What do these two numbers represent in the context of the data?

[117]:
```
# Dimensions of df
df.shape
```

[117]:
```
(16719, 17)
```

Starting with the first number 16719, this represents the number of records that are found in the dataset. The number 16, are the variables or charteristics of each record.

Find the top five games by critic score.

[116]:
```
# df sorted by Critic_Score
df.sort_values('Critic_Score', ascending=False).head(5)
```

[116]:
```
                         Name Platform  Year_of_Release     Genre  \
227   Tony Hawk's Pro Skater 2       PS           2000.0    Sports
57         Grand Theft Auto IV      PS3           2008.0    Action
51         Grand Theft Auto IV     X360           2008.0    Action
5350               SoulCalibur       DC           1999.0  Fighting
165         Grand Theft Auto V     XOne           2014.0    Action

                Publisher  NA_Sales  EU_Sales  JP_Sales  Other_Sales  \
227             Activision      3.05      1.41      0.02         0.20
57    Take-Two Interactive      4.76      3.69      0.44         1.61
51    Take-Two Interactive      6.76      3.07      0.14         1.03
5350   Namco Bandai Games       0.00      0.00      0.34         0.00
165   Take-Two Interactive      2.81      2.19      0.00         0.47

      Global_Sales  Critic_Score  Critic_Count User_Score  User_Count  \
227           4.68          98.0          19.0        7.7       299.0
57           10.50          98.0          64.0        7.5      2833.0
```

```
51           11.01          98.0          86.0        7.9      2951.0
5350          0.34          98.0          24.0        8.8       200.0
165           5.48          97.0          14.0        7.9       764.0

                     Developer Rating  NA_Sales_pct
227   Neversoft Entertainment      T         65.17
57              Rockstar North      M         45.33
51              Rockstar North      M         61.40
5350                     Namco      T          0.00
165             Rockstar North      M         51.28
```

Find the number of video games in the data frame in each genre.

[115]:
```
# Count of Genres
df['Genre'].value_counts()
```

[115]:
```
Action          3370
Sports          2348
Misc            1750
Role-Playing    1500
Shooter         1323
Adventure       1303
Racing          1249
Platform         888
Simulation       874
Fighting         849
Strategy         683
Puzzle           580
Name: Genre, dtype: int64
```

Find the first five games in the data frame on the SNES platform.

[114]:
```
# Selecting first five games where platform is SNES
df[df['Platform'] == 'SNES'].head(5)
```

[114]:
```
                                    Name Platform  Year_of_Release     Genre  \
18                     Super Mario World     SNES           1990.0  Platform
56                 Super Mario All-Stars     SNES           1993.0  Platform
71                   Donkey Kong Country     SNES           1994.0  Platform
76                      Super Mario Kart     SNES           1992.0    Racing
137  Street Fighter II: The World Warrior     SNES           1992.0  Fighting

    Publisher  NA_Sales  EU_Sales  JP_Sales  Other_Sales  Global_Sales  \
18   Nintendo     12.78      3.75      3.54         0.55         20.61
56   Nintendo      5.99      2.15      2.12         0.29         10.55
71   Nintendo      4.36      1.71      3.00         0.23          9.30
76   Nintendo      3.54      1.24      3.81         0.18          8.76
137    Capcom      2.47      0.83      2.87         0.12          6.30
```

```
     Critic_Score  Critic_Count User_Score  User_Count Developer Rating  \
18            NaN           NaN        7.5         NaN       NaN    NaN
56            NaN           NaN        7.5         NaN       NaN    NaN
71            NaN           NaN        7.5         NaN       NaN    NaN
76            NaN           NaN        7.5         NaN       NaN    NaN
137           NaN           NaN        7.5         NaN       NaN    NaN

     NA_Sales_pct
18          62.01
56          56.78
71          46.88
76          40.41
137         39.21
```

Find the five publishers with the highest total global sales. Note: You will need to calculate the total global sales for each publisher to do this.

[113]:
```
# df grouped by publisher with all columns summed
df.groupby('Publisher').sum().sort_values('Global_Sales', ascending=False).
  ↪head(5)
```

[113]:
```
                            Year_of_Release  NA_Sales  EU_Sales  JP_Sales  \
Publisher
Nintendo                          1402730.0    816.97    419.01    458.15
Electronic Arts                   2696650.0    599.50    373.91     14.35
Activision                        1959140.0    432.59    215.90      6.71
Sony Computer Entertainment       1375104.0    266.17    186.56     74.15
Ubisoft                           1867400.0    252.74    161.99      7.52

                            Other_Sales  Global_Sales  Critic_Score  \
Publisher
Nintendo                          94.68       1788.81       23413.0
Electronic Arts                  128.96       1116.96       76636.0
Activision                        75.81        731.16       39641.0
Sony Computer Entertainment       79.67        606.48       25827.0
Ubisoft                           49.18        471.61       38231.0

                            Critic_Count  User_Count  NA_Sales_pct
Publisher
Nintendo                         13029.0     58157.0      26095.34
Electronic Arts                  28218.0    169765.0      77949.02
Activision                       15380.0    121404.0      65849.00
Sony Computer Entertainment      12980.0     88341.0      27043.05
Ubisoft                          14109.0     85994.0      53530.09
```

Create a new column in the data frame that calculates the percentage of global sales from North America. Display the first five rows of the new data frame.

[108]:
```
# Function to calculate percentage
def pct(sales, global_sales):
    pct = (sales/global_sales)*100
    return pct
```

[109]:
```
# Calculates percentage of NA_Sales
NA_Sales_pct = pct(df.NA_Sales, df.Global_Sales)
```

[111]:
```
# Adds NA_Sales_pct to df and rounds to 2 decimal places
df['NA_Sales_pct'] = NA_Sales_pct.apply(lambda x:round(x,2))
# Displays first five rows
df.head(5)
```

[111]:

	Name	Platform	Year_of_Release	Genre	Publisher	NA_Sales	EU_Sales	JP_Sales	Other_Sales	Global_Sales	Critic_Score	Critic_Count	User_Score	User_Count	Developer	Rating	NA_Sales_pct
0	Wii Sports	Wii	2006.0	Sports	Nintendo	41.36	28.96	3.77	8.45	82.53	76.0	51.0	8	322.0	Nintendo	E	50.12
1	Super Mario Bros.	NES	1985.0	Platform	Nintendo	29.08	3.58	6.81	0.77	40.24	NaN	NaN	7.5	NaN	NaN	NaN	72.27
2	Mario Kart Wii	Wii	2008.0	Racing	Nintendo	15.68	12.76	3.79	3.29	35.52	82.0	73.0	8.3	709.0	Nintendo	E	44.14
3	Wii Sports Resort	Wii	2009.0	Sports	Nintendo	15.61	10.93	3.28	2.95	32.77	80.0	73.0	8	192.0	Nintendo	E	47.64
4	Pokemon Red/Pokemon Blue	GB	1996.0	Role-Playing	Nintendo	11.27	8.89	10.22	1.00	31.37	NaN	NaN	7.5	NaN	NaN	NaN	35.93

Find the number NaN entries (missing data values) in each column.

[107]:
```
# Calculates the sum of Nan Values by Column Name
df.isna().sum()
```

[107]:
```
Name                 2
Platform             0
Year_of_Release    269
Genre                2
Publisher           54
NA_Sales             0
EU_Sales             0
JP_Sales             0
```

```
Other_Sales           0
Global_Sales          0
Critic_Score       8582
Critic_Count       8582
User_Score            0
User_Count         9129
Developer          6623
Rating             6769
NA_Sales_pct          0
dtype: int64
```

Try to calculate the median user score of all the video games. You will likely run into an error because some of the user score entries are a non-numerical string that cannot be converted to a float. Find and replace this string with NaN and then calculate the median. Then, replace all NaN entries in the user score column with the median value.

[106]:
```
# Read in numpy library
import numpy as np
# Replace string value with NaN
df['User_Score'] = df['User_Score'].replace('tbd', np.nan)
# Calculate and display median
median = df.User_Score.median()
median
```

[106]:
```
7.5
```

[112]:
```
# Replace NaN values with Median
df['User_Score'] = df['User_Score'].replace(np.nan, median)
# Displays first five rows
df.head(5)
```

[112]:
```
                      Name Platform  Year_of_Release         Genre Publisher  \
0               Wii Sports      Wii           2006.0        Sports  Nintendo
1         Super Mario Bros.      NES           1985.0      Platform  Nintendo
2            Mario Kart Wii      Wii           2008.0        Racing  Nintendo
3         Wii Sports Resort      Wii           2009.0        Sports  Nintendo
4  Pokemon Red/Pokemon Blue       GB           1996.0  Role-Playing  Nintendo

   NA_Sales  EU_Sales  JP_Sales  Other_Sales  Global_Sales  Critic_Score  \
0     41.36     28.96      3.77         8.45         82.53          76.0
1     29.08      3.58      6.81         0.77         40.24           NaN
2     15.68     12.76      3.79         3.29         35.52          82.0
3     15.61     10.93      3.28         2.95         32.77          80.0
4     11.27      8.89     10.22         1.00         31.37           NaN

   Critic_Count User_Score  User_Count Developer Rating  NA_Sales_pct
```

0	51.0	8	322.0	Nintendo	E	50.12
1	NaN	7.5	NaN	NaN	NaN	72.27
2	73.0	8.3	709.0	Nintendo	E	44.14
3	73.0	8	192.0	Nintendo	E	47.64
4	NaN	7.5	NaN	NaN	NaN	35.93