Video Games Sales Analysis

This is a video games sales dataset. I got this dataset from kaggle community which was generated by vgcgartz.com. This Dataset contains a list of video games with sales greater than 100,000 copies. Fields Include:

1. Rank - Ranking of overall sales
2. Name - The games name
3. Platform - Platform of the games release(i.e. PC, PSA, etc.)
4. Year - Year of the game's release
5. Genre - Genre of the game
6. Publisher - Publisher of the game
7. NA_Sales - Sales in North America(in millions)
8. JP_Sales - Sales in Japan
9. Other_Sales - Sales in rest of the world
10. EU_Sales - Sales in Europe
11. Global_Sales - Total worldwide sales There are 16,598 records. Here is the link which you can get - https://www.kaggle.com/datasets/gregorut/videogamesales . The analysis of this dataset is based on the learning which I have done from the course Data Analysis with Python: Zero to Pandas. Following topics are covered in this course:
12. Selecting and downloading a dataset
13. Data preparation and cleaning
14. Exploratory analysis and visualization
15. Asking and answering interesting questions
16. Summarizing inferences and drawing conclusions

How to run the code

This is an executable *Jupyter notebook* hosted on Jovian.ml, a platform for sharing data science projects. You can run and experiment with the code in a couple of ways: *using free online resources* (recommended) or *on your own computer*.

Option 1: Running using free online resources (1-click, recommended)

The easiest way to start executing this notebook is to click the "Run" button at the top of this page, and select "Run on Binder". This will run the notebook on mybinder.org, a free online service for running Jupyter notebooks. You can also select "Run on Colab" or "Run on Kaggle".

Option 2: Running on your computer locally

1. Install Conda by following these instructions. Add Conda binaries to your system `PATH`, so you can use the `conda` command on your terminal.
2. Create a Conda environment and install the required libraries by running these commands on the terminal:

```
conda create -n zerotopandas -y python=3.8
conda activate zerotopandas
pip install jovian jupyter numpy pandas matplotlib seaborn opendatasets --upgrade
```

3. Press the "Clone" button above to copy the command for downloading the notebook, and run it on the terminal. This will create a new directory and download the notebook. The command will look something like this:

   ```
   jovian clone notebook-owner/notebook-id
   ```

4. Enter the newly created directory using `cd directory-name` and start the Jupyter notebook.

   ```
   jupyter notebook
   ```

You can now access Jupyter's web interface by clicking the link that shows up on the terminal or by visiting http://localhost:8888 on your browser. Click on the notebook file (it has a `.ipynb` extension) to open it.

Downloading the Dataset

> Instructions for downloading the dataset (delete this cell)
>
> - Find an interesting dataset on this page: https://www.kaggle.com/datasets?fileType=csv
> - The data should be in CSV format, and should contain at least 3 columns and 150 rows
> - Download the dataset using the `opendatasets` Python library

```
!pip install jovian opendatasets --upgrade --quiet
```

Let's begin by downloading the data, and listing the files within the dataset.

```
# Change this
dataset_url = 'https://www.kaggle.com/datasets/gregorut/videogamesales'
```

```
import opendatasets as od
od.download(dataset_url)
```

```
Please provide your Kaggle credentials to download this dataset. Learn more:
http://bit.ly/kaggle-creds
Your Kaggle username: tanishaagrawal945
Your Kaggle Key: ·····
Downloading videogamesales.zip to ./videogamesales
100%|██████████| 381k/381k [00:00<00:00, 98.3MB/s]
```

The dataset has been downloaded and extracted.

```
# Change this
data_dir = './videogamesales'
```

```
import os
os.listdir(data_dir)
```

```
['vgsales.csv']
```

Let us save and upload our work to Jovian before continuing.

```
project_name = "video-game-sales-analysis" # change this (use lowercase letters and hyp
```

```
!pip install jovian --upgrade -q
```

```
import jovian
```

```
jovian.commit(project=project_name)
```

```
[jovian] Updating notebook "tannu945/video-game-sales-analysis" on https://jovian.ai
[jovian] Committed successfully! https://jovian.ai/tannu945/video-game-sales-analysis
```

```
'https://jovian.ai/tannu945/video-game-sales-analysis'
```

Data Preparation and Cleaning

Load the dataset in the dataframe using pandas and perform various operations on different columns of the dataset. Let's explore the different columns of the dataset, check whether they contain null values, cleaning the data by removing unwanted rows and columns, fill null values using mean or other methods.

Import Pandas package

```
import pandas as pd
```

Extracting csv file to dataframe using pandas read_csv function.

```
vgsales_df = pd.read_csv('./videogamesales/vgsales.csv')
```

let's check the datatype of the columns, the number of entries, memory usage, Column names.

```
vgsales_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 16598 entries, 0 to 16597
Data columns (total 11 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
```

```
 0   Rank          16598 non-null  int64
 1   Name          16598 non-null  object
 2   Platform      16598 non-null  object
 3   Year          16327 non-null  float64
 4   Genre         16598 non-null  object
 5   Publisher     16540 non-null  object
 6   NA_Sales      16598 non-null  float64
 7   EU_Sales      16598 non-null  float64
 8   JP_Sales      16598 non-null  float64
 9   Other_Sales   16598 non-null  float64
 10  Global_Sales  16598 non-null  float64
dtypes: float64(6), int64(1), object(4)
memory usage: 1.4+ MB
```

The dataset contains 10 columns named Rank int64 dtype, Name object dtype, Platform object dtype, Year float64 dtype, Gere object dtype, Publisher object dtype, NA_Sales, float64 dtype, EU_Sales float64 dtype, JP_Sales float64 dtype, Other_Sales float64 dtype and Global_Sales float64 dtype. And it seems like Year and Publisher contains some null values. The float64 dytpe has 6 columns, int64 dtype has 1 column and object dtype has 4 columns.

Let's analyse the starting rows of the dataset.

```
vgsales_df.head()
```

	Rank	Name	Platform	Year	Genre	Publisher	NA_Sales	EU_Sales	JP_Sales	Other_Sales	Global_Sa
0	1	Wii Sports	Wii	2006.0	Sports	Nintendo	41.49	29.02	3.77	8.46	82
1	2	Super Mario Bros.	NES	1985.0	Platform	Nintendo	29.08	3.58	6.81	0.77	40
2	3	Mario Kart Wii	Wii	2008.0	Racing	Nintendo	15.85	12.88	3.79	3.31	35
3	4	Wii Sports Resort	Wii	2009.0	Sports	Nintendo	15.75	11.01	3.28	2.96	33
4	5	Pokemon Red/Pokemon Blue	GB	1996.0	Role-Playing	Nintendo	11.27	8.89	10.22	1.00	31

After analysng first five rows it seems like that the top publisher if Nintendo and the global sale of Nintendo is the highest. Nintendo produces games in different genre in different year. It updated it's games it seems like.

Now, let's check if the dataset contains null values or not i.e. if the dataset has some cells empty or not.

```
vgsales_df.isnull().sum()
```

```
Rank            0
Name            0
Platform        0
Year          271
Genre           0
Publisher      58
NA_Sales        0
```

```
EU_Sales          0
JP_Sales          0
Other_Sales       0
Global_Sales      0
dtype: int64
```

Year column of the dataset contains 271 null values and the Publisher column of the dataset contains 58 null values. we need to fill them before analysing the data.

Let fill the null values of the dataset to perform various statistical calculations without any difficulty.

```
vgsales_df['Year'].fillna(0, inplace=True), vgsales_df['Publisher'].fillna('Unknown', i
```

```
(None, None)
```

Let's calculate the various statistical functions of numerical columns of the dataset.

```
vgsales_df.describe()
```

	Rank	Year	NA_Sales	EU_Sales	JP_Sales	Other_Sales	Global_Sales
count	16598.000000	16598.000000	16598.000000	16598.000000	16598.000000	16598.000000	16598.000000
mean	8300.605254	1973.647307	0.264667	0.146652	0.077782	0.048063	0.537441
std	4791.853933	254.346809	0.816683	0.505351	0.309291	0.188588	1.555028
min	1.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.010000
25%	4151.250000	2003.000000	0.000000	0.000000	0.000000	0.000000	0.060000
50%	8300.500000	2007.000000	0.080000	0.020000	0.000000	0.010000	0.170000
75%	12449.750000	2010.000000	0.240000	0.110000	0.040000	0.040000	0.470000
max	16600.000000	2020.000000	41.490000	29.020000	10.220000	10.570000	82.740000

On looking the statistical values, the highest average sales of video games is in the Japan and the lowest average sale is in the Europe. The min and max sales is given and it also shows the 25%, 50% and 75% values and the standard of each numerical column.

```
import jovian
```

```
jovian.commit()
```

```
[jovian] Updating notebook "tannu945/video-game-sales-analysis" on https://jovian.ai
[jovian] Committed successfully! https://jovian.ai/tannu945/video-game-sales-analysis
```

```
'https://jovian.ai/tannu945/video-game-sales-analysis'
```

Exploratory Analysis and Visualization

Compute the mean, sum, range and other interesting statistics for numeric columns Explore distributions of numeric columns using histograms etc. Explore relationship between columns using scatter plots, bar charts etc. Make a note of interesting insights from the exploratory analysis

Let's begin by importing `matplotlib.pyplot and seaborn .`

```
import seaborn as sns
import matplotlib
import matplotlib.pyplot as plt
%matplotlib inline

sns.set_style('darkgrid')
matplotlib.rcParams['font.size'] = 14
matplotlib.rcParams['figure.figsize'] = (30, 5)
matplotlib.rcParams['figure.facecolor'] = '#00000000'
```

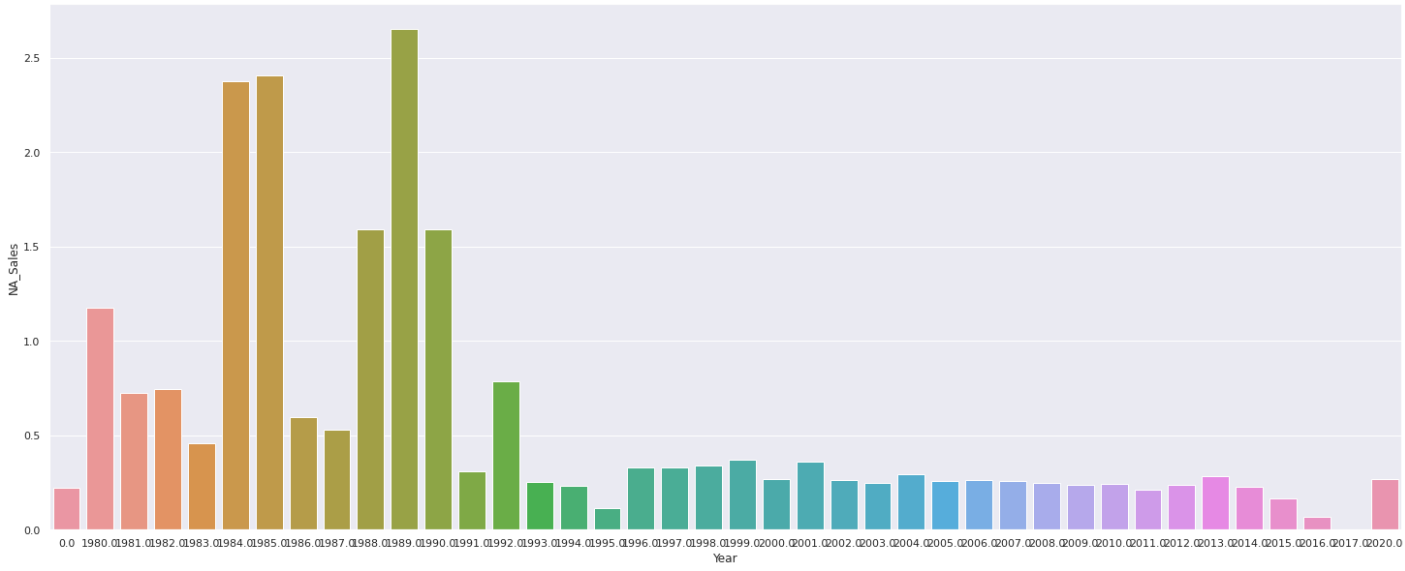
Let's plot a bar graph for the average of NA_Sales in respect of their years to get the highest amount of sales.

```
sns.set(rc={"figure.figsize":(25,10)})
```

```
df = vgsales_df.groupby('Year').mean()
df['Year'] = df.index
```

```
sns.barplot(data=df, x='Year', y='NA_Sales')
```

```
<AxesSubplot:xlabel='Year', ylabel='NA_Sales'>
```

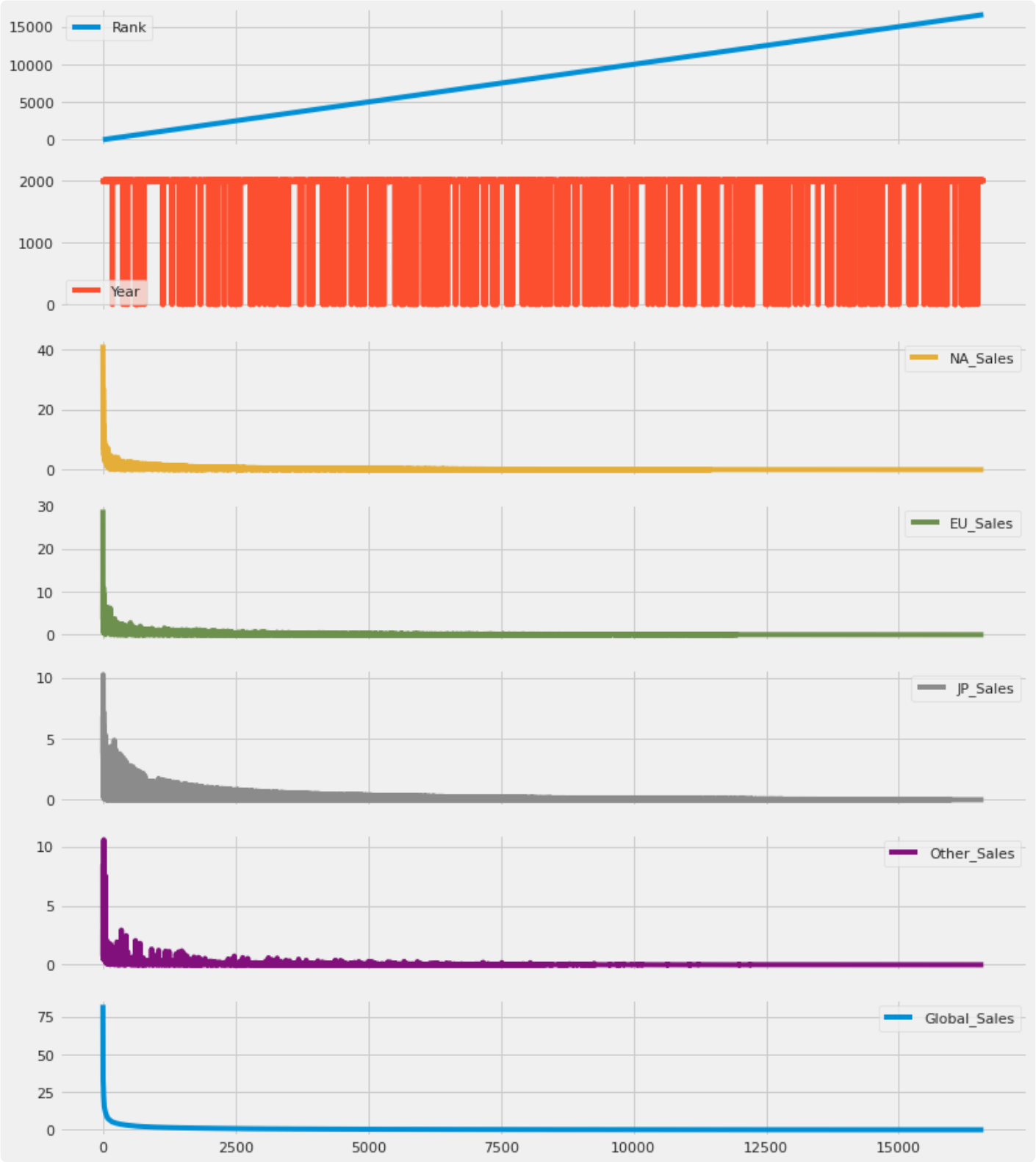


The plotted bar graph shows that the North America sales was highest in 1989 about 2.8 millions of genre Puzzle.And lowest sales in 2017 about 0.1 millions. As we can see in the graph, the demand of Action games increases after 2005. Till then puzzle, strategy, shooting, simulation type of games are in demand.

Subplots of every column to analyse every column more precisely.

```
plt.style.use("fivethirtyeight")
vgsales_df.plot(subplots=True, figsize=(12, 15))
```

```
array([<AxesSubplot:>, <AxesSubplot:>, <AxesSubplot:>, <AxesSubplot:>,
       <AxesSubplot:>, <AxesSubplot:>, <AxesSubplot:>], dtype=object)
```

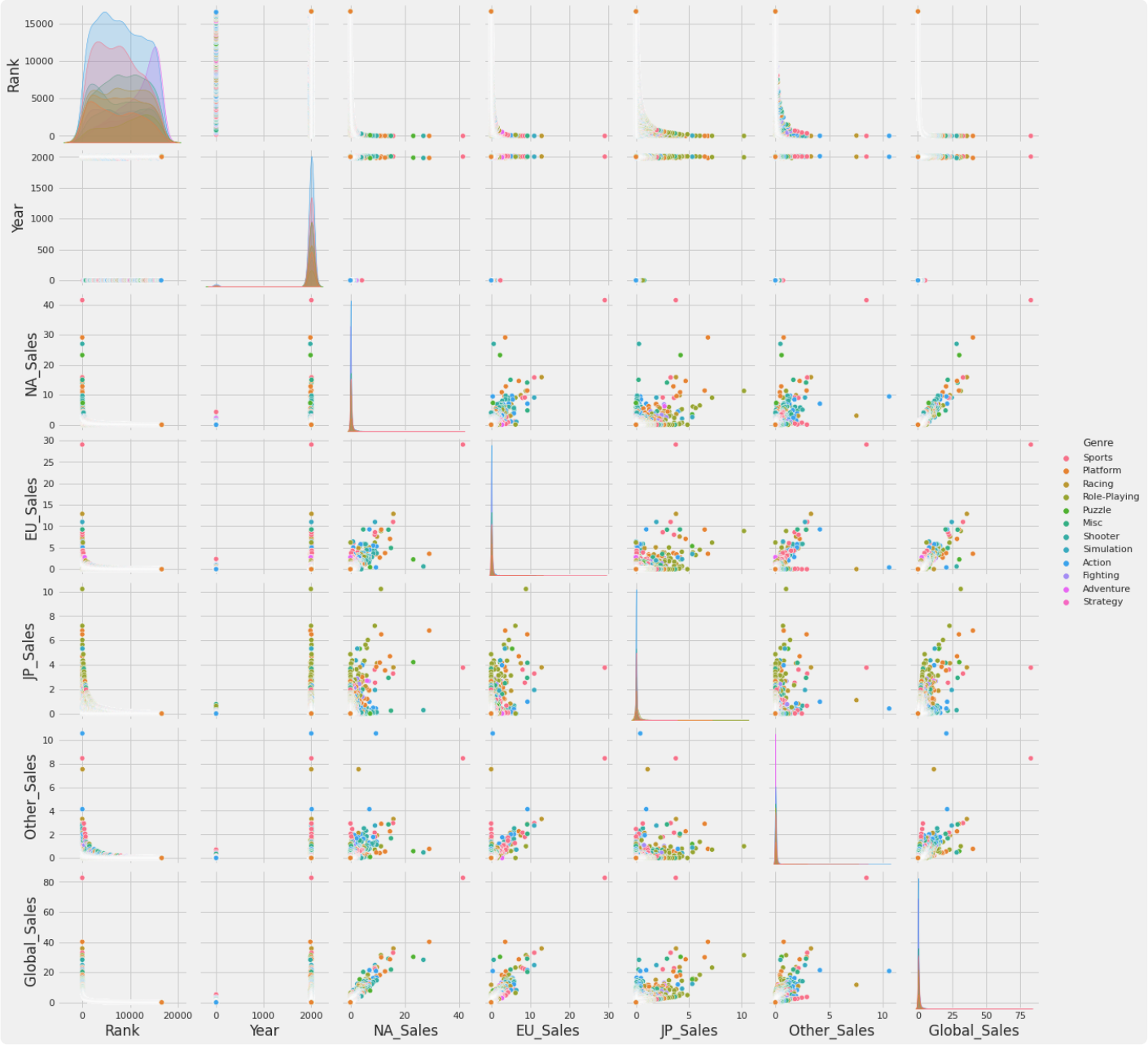


The first subplot is of the column rank and in that as you can see it is gradually increasing as the rank is increased by each row pf the data. The second subplot is of Year and it is constant. In the third subplot which is of NA_Sales indicates that the highest sales was about 42 millions. The fourth subplot shows the highest sales of Europe which is approx. 29 millions. In the fifth subplot, the highest sales was about 11 millions. The sixth subplot indicating the max sales of Other_sales which is about 12 millions. And the last subplot is indicating the global sales which is about 82 millions. By this we conclude that the global sales is highest amongst the all.

Let's plot a PairPlot for different columns of different genres to explore each column more precisely.

```
sns.pairplot(vgsales_df, hue='Genre')
```

```
<seaborn.axisgrid.PairGrid at 0x7fb0675732e0>
```

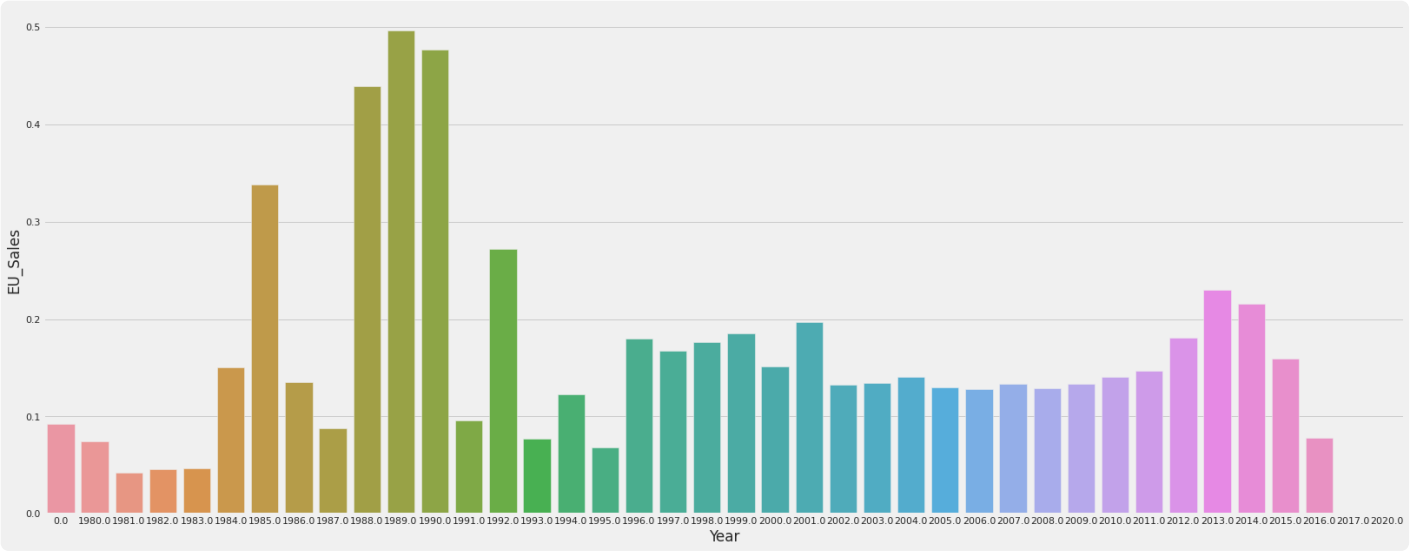


If we take axis[0][1] graph then you can clearly observe that after year 2000 the rank of Sports is higher the and after that we will analyse the row of NA_Sales then we can clearly stat that the sales of the Sports video games is highest after year 2000 and if we compare NA_Sales with other sales then we can conclude that in North America sports genre video games is in demand after 2000. Like that we can also conclude oother sales columns that in Europe also sports genre is in demand. In Japan Role-Playing games is in demand and in Other's Action video games is in demand. Globally Platform games gain more popularity.

Let's plot a Bar plot for the average of EU_Sales in respect of their years to get the highest amount of sales.

```
sns.barplot(data=df, x='Year', y='EU_Sales')
```

```
<AxesSubplot:xlabel='Year', ylabel='EU_Sales'>
```



The plotted bar graph shows that the Europe sales was highest in 1989 about 0.49 millions of genre Puzzle.And lowest sales in 2017 about 0.05 millions of Strategy genre that means it was less in demand. As we can see in the graph, the demand of Action games increases after 2005. Till then puzzle, strategy, shooting, simulation type of games are in demand. As the time passes, the demand of Action games shifted to the demand of Sports games.

Let us save and upload our work to Jovian before continuing

```
import jovian
```

```
jovian.commit()
```

```
[jovian] Updating notebook "tannu945/video-game-sales-analysis" on https://jovian.ai
[jovian] Committed successfully! https://jovian.ai/tannu945/video-game-sales-analysis
```

```
'https://jovian.ai/tannu945/video-game-sales-analysis'
```

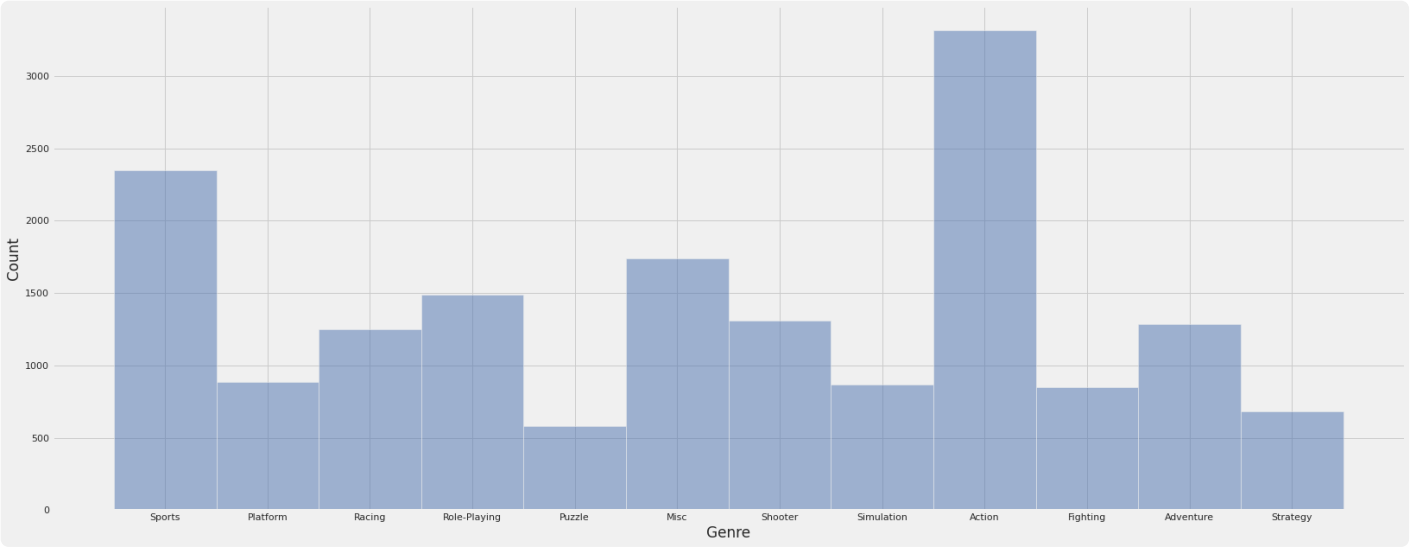
Asking and Answering Questions

We have analysed the different columns of the dataset with their visualizations and gained the several insights about the sales of the video games in different genre's. Let's ask some specific questions and try to visualize them.

Q1: Which genre is highest in demand?

```
sns.histplot(vgsales_df.Genre, alpha=0.5)
```

```
<AxesSubplot:xlabel='Genre', ylabel='Count'>
```



The histogram stats that the highest demand is of Action genre as it gets the highest count values and the lowest is of Puzzle genre. After the Action games, Sports games are in demand following Misc and role-playing. it shows that people are most interested in action games and least interested in puzzle games. To increase sales of video games, companies can focus on production of action games more.

Q2. Which publisher has published games in large amount ? Visualize the year-wise Global Sales of that publisher from the year 2000.

```
from numpy import mean
```

```
vgsales_df['Publisher'].value_counts()
```

```
Electronic Arts                 1351
Activision                       975
Namco Bandai Games               932
Ubisoft                          921
Konami Digital Entertainment     832
                                ...
Warp                               1
New                                1
Elite                              1
Evolution Games                    1
UIG Entertainment                  1
Name: Publisher, Length: 578, dtype: int64
```

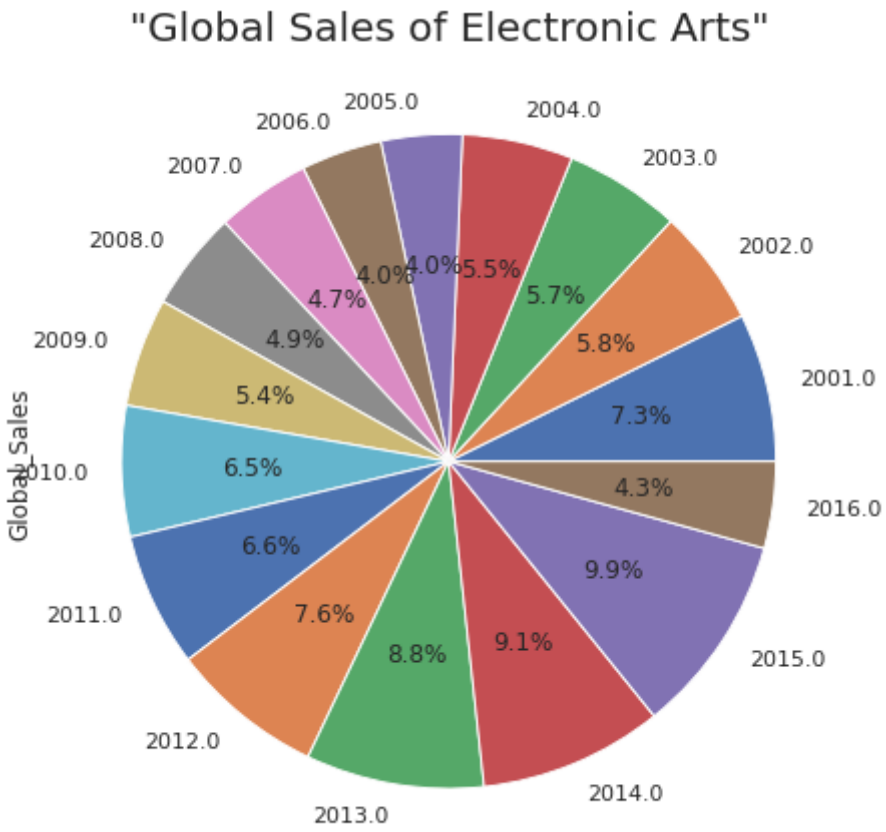
The Electronic Arts publishing company published highest amount of games.

```
df2=vgsales_df[vgsales_df.Publisher=='Electronic Arts']
```

```
new_df1 = df2.groupby('Year').mean()
new_df1=new_df1.drop('Rank', axis=1)
new_df1 = new_df1[new_df1.index>2000]
```

```
new_df1['Global_Sales'].plot(kind='pie', figsize=(25,7), subplots=True, autopct="%1.1f%
plt.title("\"Global Sales of Electronic Arts\"", fontsize=20)
```

```
Text(0.5, 1.0, '"Global Sales of Electronic Arts"')
```



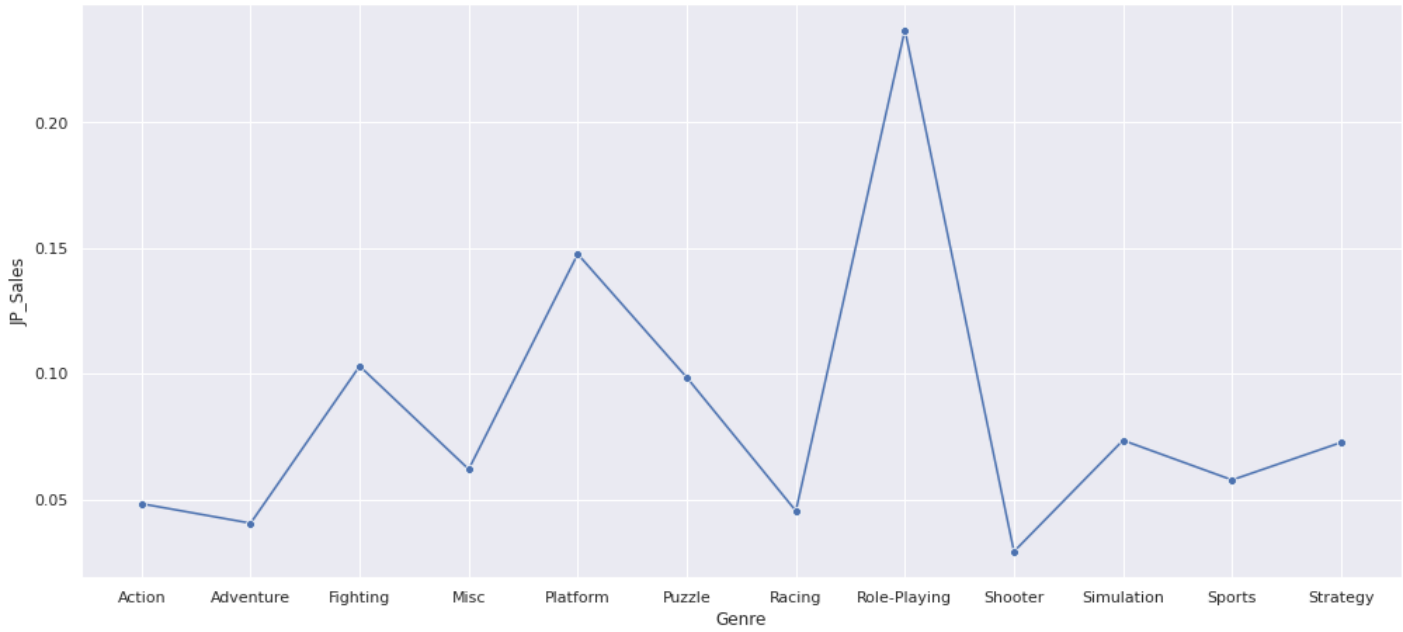
The global sale of Electronic Arts is maximum in 2015 and minimum in 2005 and 2006. It may be possible that the demand of video games is decreased at that time and again increased in 2015 or the company has less published games in 2005 and 2006 and publish more games in 2015.

Q3: Visualize the sale of different genre's in Japan.

```
new_df = vgsales_df.groupby('Genre').mean()
```

```
sns.set(rc={"figure.figsize":(15,7)})
sns.lineplot(data=new_df, x='Genre', y='JP_Sales', marker="o")
```

```
<AxesSubplot:xlabel='Genre', ylabel='JP_Sales'>
```



The graph stats that in Japan Playing video games are more popular than any other genre. And people are less interested in playing Shooter video games. After action games, a very population of Japan likes to play Platform Games.

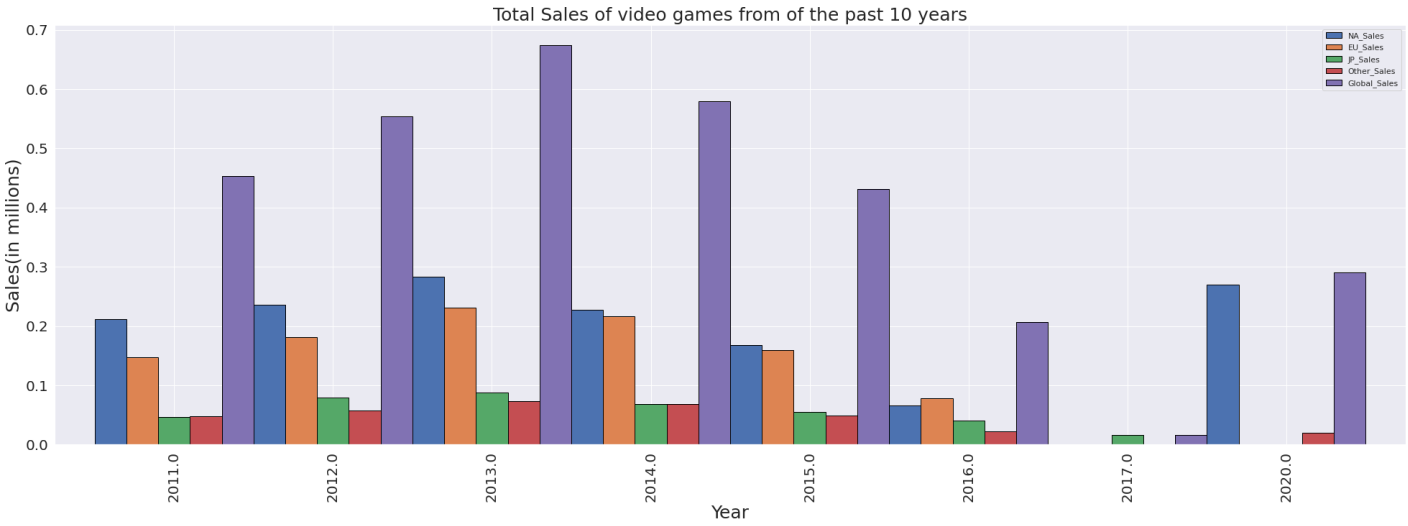
Q4: What was the total sales of video games of the past 10 years? Visualize them in a graph.

```
df = vgsales_df.groupby('Year').mean()
```

```
df=df[df.index>2010]
```

```
df.plot(kind='bar', width=1.0, edgecolor='black', figsize=(30,10), fontsize=20)
plt.title('Total Sales of video games from of the past 10 years', fontsize=25)
plt.xlabel('Year', fontsize=25)
plt.ylabel('Sales(in millions)', fontsize=25)
```

```
Text(0, 0.5, 'Sales(in millions)')
```



The global sales of video games was highest in the year 2013 and lowest in 2016. The North America sales was highest in 2013 and lowest in 2016.If we observe the chart then we can conclude that the overall sale was highest in 2013 and lowest in 2016. It may be possible that in 2016 some crisis may occur due to which the sale of video game is affected.

Let us save and upload our work to Jovian before continuing.

```
import jovian
```

```
jovian.commit()
```

```
[jovian] Updating notebook "tannu945/video-game-sales-analysis" on https://jovian.ai
[jovian] Committed successfully! https://jovian.ai/tannu945/video-game-sales-analysis
```

```
'https://jovian.ai/tannu945/video-game-sales-analysis'
```

Inferences and Conclusion

We have drawn different conclusions by ding this analysis. Here is summary of them:

1. North America Sales was highest in 1989 and at that time the demand of puzzle games was high.
2. The highest sale of North America of video games is 42 millions.
3. The highest sale of Europe is 29 millions.
4. The highest sale of Japan is 11 millions.
5. The highest sale in other countries is 12 millions.
6. The highest sale globally is 82 millions.
7. In North America and in Europe, sports video games are in demand whereas in Japan Role-Playing games are in demand. ANd if we watch globally people like platform games more.
8. Action video games are all way more popular than any other.
9. The "Electronic Arts" publisher published games in large amount.
10. Total Sale of video games was highest in 2013 and lowest in 2016.

```
import jovian
```

```
jovian.commit()
```

```
[jovian] Updating notebook "tannu945/video-game-sales-analysis" on https://jovian.ai
[jovian] Committed successfully! https://jovian.ai/tannu945/video-game-sales-analysis
```

```
'https://jovian.ai/tannu945/video-game-sales-analysis'
```

References and Future Work

Check out the following resources to learn more about the dataset and tools used in this notebook: Kaggle video game sales survey: https://www.kaggle.com/datasets/gregorut/videogamesales Pandas user guide: Seaborn

User guide: Opendatasets Python Library: As a next step, you can try out a project on another dataset of your choice:

Kaggle Video game sales survey: https://www.kaggle.com/datasets/gregorut/videogamesales Pandas user guide: https://pandas.pydata.org/docs/user_guide/index.html Matplotlib user guide: https://matplotlib.org/3.3.1/users/index.html Seaborn user guide & tutorial: https://seaborn.pydata.org/tutorial.html opendatasets Python library: https://github.com/JovianML/opendatasets As a next step, you can try out a project on another dataset of your choice: https://jovian.ml/aakashns/zerotopandas-course-project-starter .

```
import jovian
```

```
jovian.commit()
```

```
[jovian] Attempting to save notebook..
[jovian] Updating notebook "aakashns/zerotopandas-course-project-starter" on https://jovian.ml/
[jovian] Uploading notebook..
[jovian] Capturing environment..
[jovian] Committed successfully! https://jovian.ml/aakashns/zerotopandas-course-project-starter
```

```
'https://jovian.ml/aakashns/zerotopandas-course-project-starter'
```

```

```