Video Game Sales

Abstract

In this Project I will perform Exploratory data analysis, data visualizations and Modling of the video game sales.For EDA,I am using Python programming language, for data visualization,I am using Matplotlib and Seaborn and for data modling,I am using sketlearn.

Data

This project is one of the T5 Data Science BootCamp requirements. Data provided by Kaggle has been used in this project. The dataset is provided in .csv format.This dataset contains a list of video games with greater sales. It contains 16,598 records, each record has 11 features. The most relevant feature to this project is Total worldwide sales.This feature is extracted from other features such as Sales in North America,Sales in Europe,Sales in Japan, and Sales in the rest of the world.

Fields include

- Rank - Ranking of overall sales
- Name - The games name
- Platform - Platform of the games release (i.e. PC,PS4, etc.)
- Year - Year of the game's release
- Genre - Genre of the game
- Publisher - Publisher of the game
- NA_Sales - Sales in North America (in millions)
- EU_Sales - Sales in Europe (in millions)
- JP_Sales - Sales in Japan (in millions)
- Other_Sales - Sales in the rest of the world (in millions)
- Global_Sales - Total worldwide sales.

Step 1: Identify your Problem & Goal

The Goal: I will try to answer the following questions:

1. Global Sales Distribution.
2. The video games produced in a each year.
3. which genre sold the most globally?
4. The platform,game and the publisher which has the top sales in global sales
5. The platform,game and the publisher which has the low sales in global sales
6. How does the Genre affect the Global_Sales?
7. What are most games produced in a specific Gaming Platform?

EDA in Python

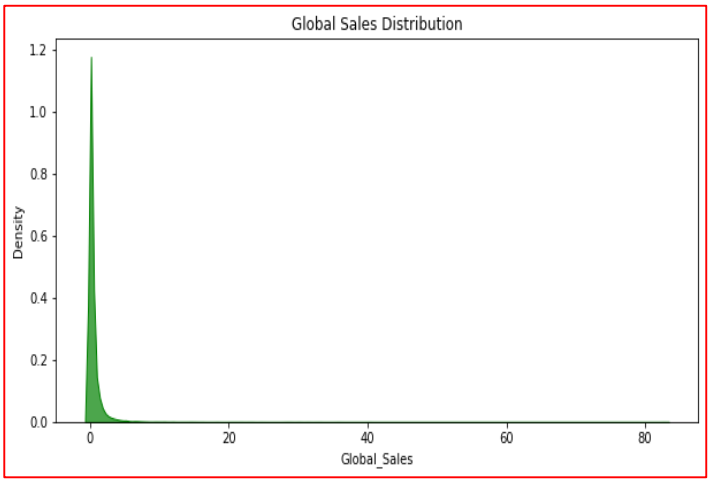
Multiple libraries are available to perform basic EDA but I am going to use pandas, matplotlib and seaborn for this project. Pandas for data manipulation and matplotlib and seaborn, for plotting graphs. Jupyter Nootbooks to write code and other findings. Jupyter notebooks is kind of diary for data analysis and scientists, a web based platform where you can mix Python, html and Markdown to explain your data insights.

Step 2: Gather and clean your data

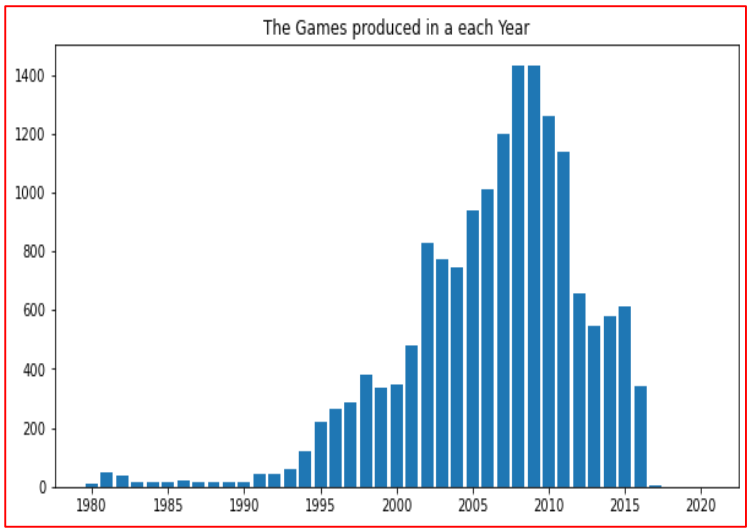
1. Import libraries and load dataset
2. summarize the data
3. check Concise info (columns data type) of dataset
4. check our data is clean or not
5. check the percentage of the data are missing column
6. Removing the missing value rows in the dataset

Step 3: Get to know your data and Visualizations

1. Global Sales Distribution.



2. The video games produced in an each year.



3. Which genre sold the most globally?

Genre	Global_Sales
Action	1722.84
Adventure	234.59
Fighting	444.05
Misc	789.87
Platform	829.13
Puzzle	242.21
Racing	726.76
Role-Playing	923.83
Shooter	1026.20
Simulation	389.98
Sports	1309.24
Strategy	173.27

4. The platform,game and the publisher which has the top sales in global sales

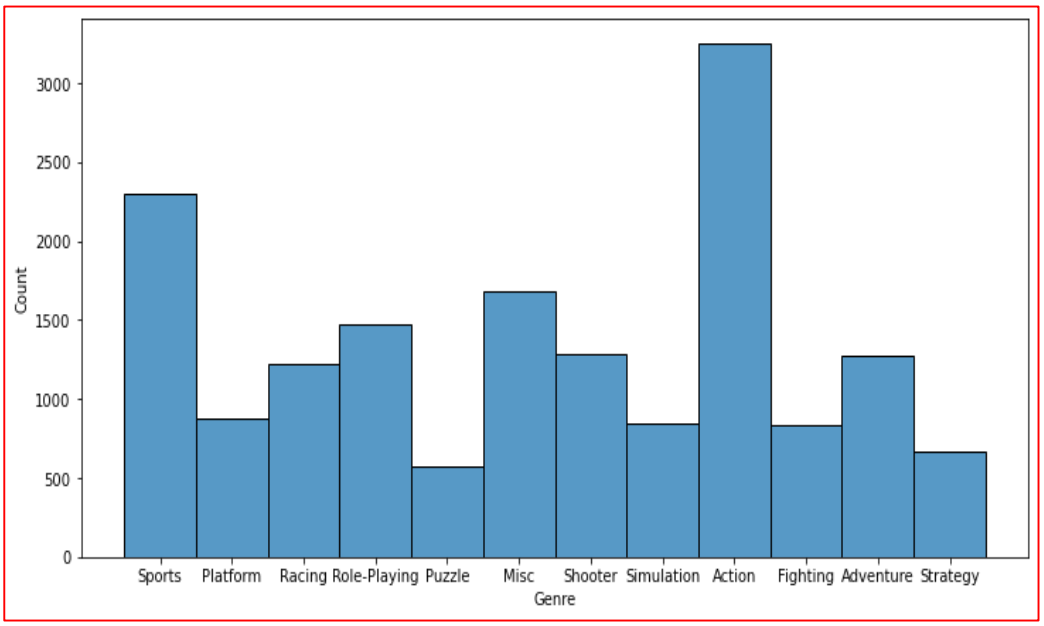
	Rank	Name	Platform	Year	Genre	Publisher	NA_Sales	EU_Sales	JP_Sales	Other_Sales	Global_Sales
0	1	Wii Sports	Wii	2006.0	Sports	Nintendo	41.49	29.02	3.77	8.46	82.74

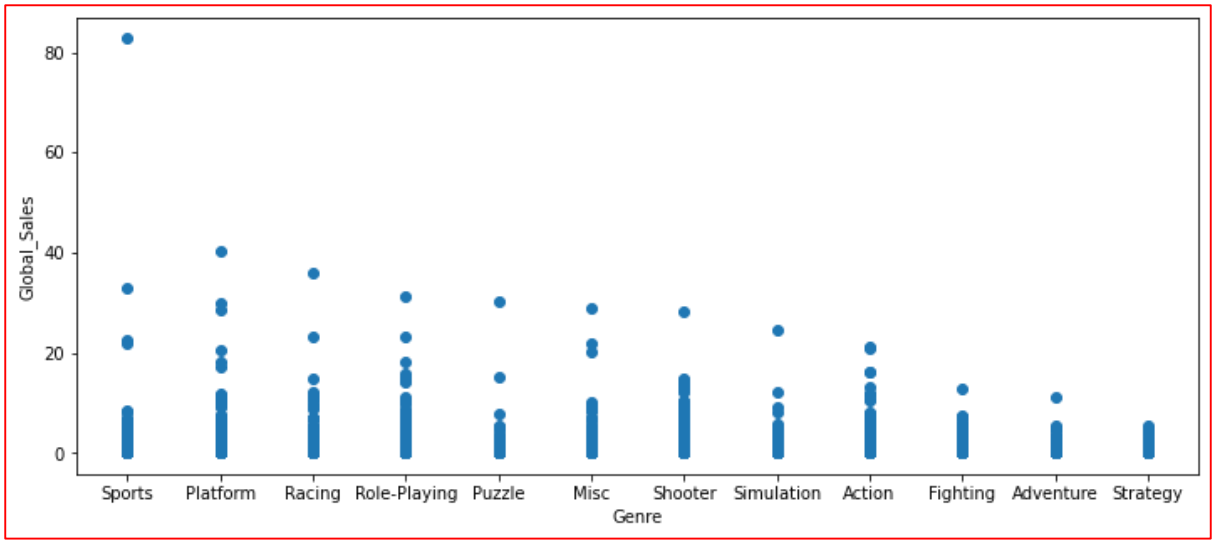
5. The platform,game and the publisher which has the low sales in global sales

	Rank	Name	Platform	Year	Genre	Publisher	NA_Sales	EU_Sales	JP_Sales	Other_Sales	Global_Sales
15980	15983	Turok	PC	2008.0	Action	Touchstone	0.00	0.01	0.00	0.0	0.01
15981	15984	Coven and Labyrinth of Refrain	PSV	2016.0	Action	Nippon Ichi Software	0.00	0.00	0.01	0.0	0.01
15982	15985	Super Battle For Money Sentouchuu: Kyuukyoku n...	3DS	2016.0	Action	Namco Bandai Games	0.00	0.00	0.01	0.0	0.01
15983	15986	Dragon Zakura DS	DS	2007.0	Misc	Electronic Arts	0.00	0.00	0.01	0.0	0.01
15984	15987	Chameleon: To Dye For!	DS	2006.0	Puzzle	505 Games	0.01	0.00	0.00	0.0	0.01
...	...	...	...	...	...	...	...	...	...	...	...
16593	16596	Woody Woodpecker in Crazy Castle 5	GBA	2002.0	Platform	Kemco	0.01	0.00	0.00	0.0	0.01
16594	16597	Men in Black II: Alien Escape	GC	2003.0	Shooter	Infogrames	0.01	0.00	0.00	0.0	0.01
16595	16598	SCORE International Baja 1000: The Official Game	PS2	2008.0	Racing	Activision	0.00	0.00	0.00	0.0	0.01
16596	16599	Know How 2	DS	2010.0	Puzzle	7G//AMES	0.00	0.01	0.00	0.0	0.01
16597	16600	Spirits & Spells	GBA	2003.0	Platform	Wanadoo	0.01	0.00	0.00	0.0	0.01

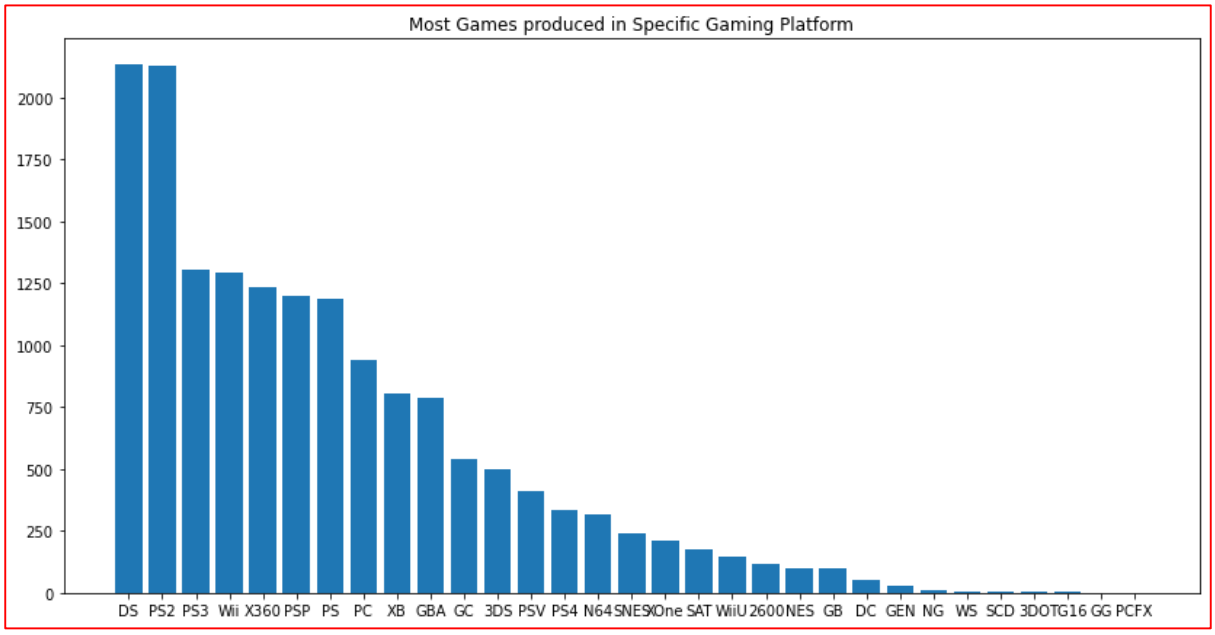
600 rows × 11 columns

6. How does the Genre affect the Global_Sales?





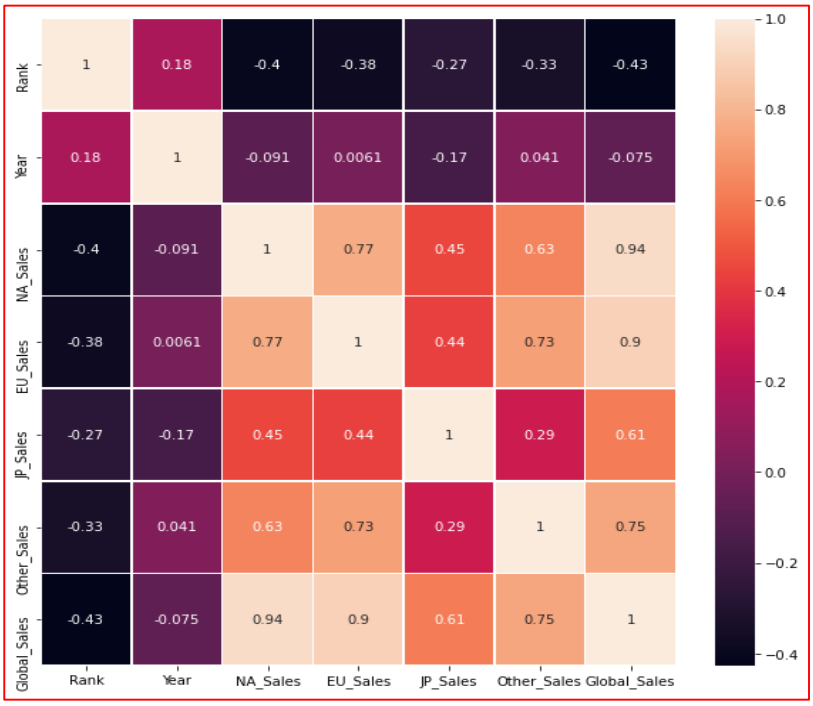
7. What are most games produced in a specific Gaming Platform?

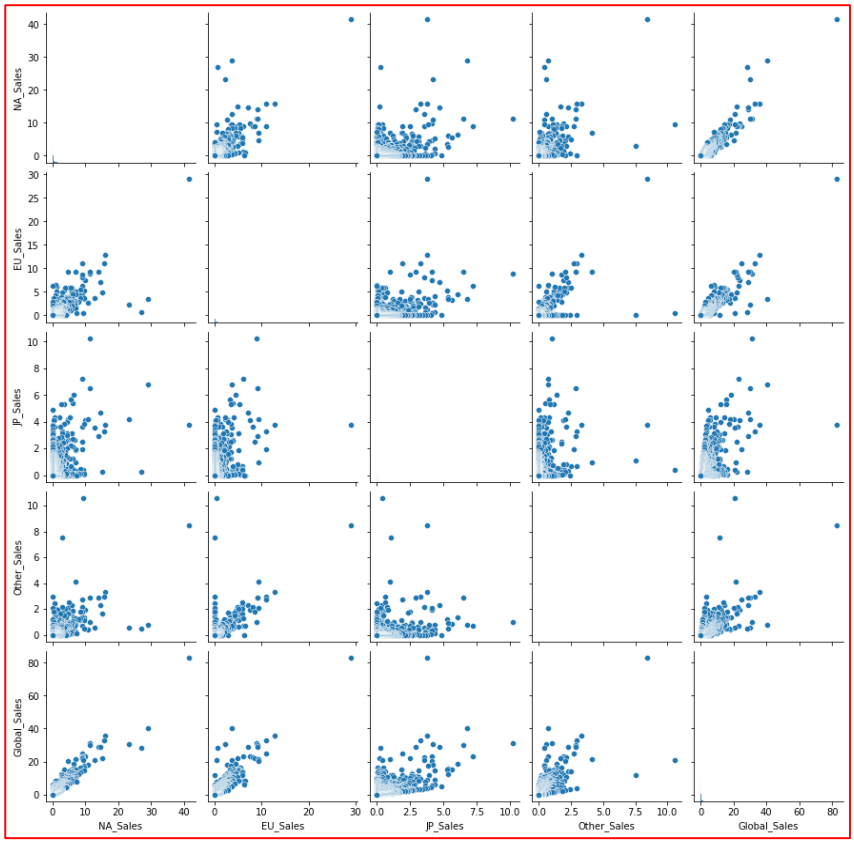


Step 4: Picking your model

1. Determining the relevancy of features using correlations,heatmap and pairplot

	Rank	Year	NA_Sales	EU_Sales	JP_Sales	Other_Sales	Global_Sales
Rank	1.000000	0.178027	-0.400315	-0.379137	-0.269323	-0.332735	-0.426975
Year	0.178027	1.000000	-0.091285	0.006108	-0.169387	0.041128	-0.074647
NA_Sales	-0.400315	-0.091285	1.000000	0.768923	0.451283	0.634518	0.941269
EU_Sales	-0.379137	0.006108	0.768923	1.000000	0.436379	0.726256	0.903264
JP_Sales	-0.269323	-0.169387	0.451283	0.436379	1.000000	0.290559	0.612774
Other_Sales	-0.332735	0.041128	0.634518	0.726256	0.290559	1.000000	0.747964
Global_Sales	-0.426975	-0.074647	0.941269	0.903264	0.612774	0.747964	1.000000





2. Train, Test Split For modelling Linear Regression

```
In [75]: # take only the useful features of the dataset
         ## Splitting the dataset into independent and dependent vaiables
         from sklearn.model_selection import train_test_split
         from sklearn.linear_model import LinearRegression
         from sklearn.metrics import r2_score
         X = dataset[['NA_Sales','EU_Sales','JP_Sales','Other_Sales']]
         y = dataset[['Global_Sales']]
         X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=0)

In [76]: X_train.shape
Out[76]: (13032, 4)

In [77]: X_test.shape
Out[77]: (3259, 4)

In [78]: y_train.shape
Out[78]: (13032, 1)

In [79]: y_test.shape
Out[79]: (3259, 1)
```

3. Predicting Test Data

```
predictions = Sales_R.predict(X_test)
r2_score(y_test, predictions)
```

```
0.9999863067614225
```

4. Evaluations

Step 5: How do I know if my model is good?

The model is fitting because adjusted_r_squared for training set close to testing set.

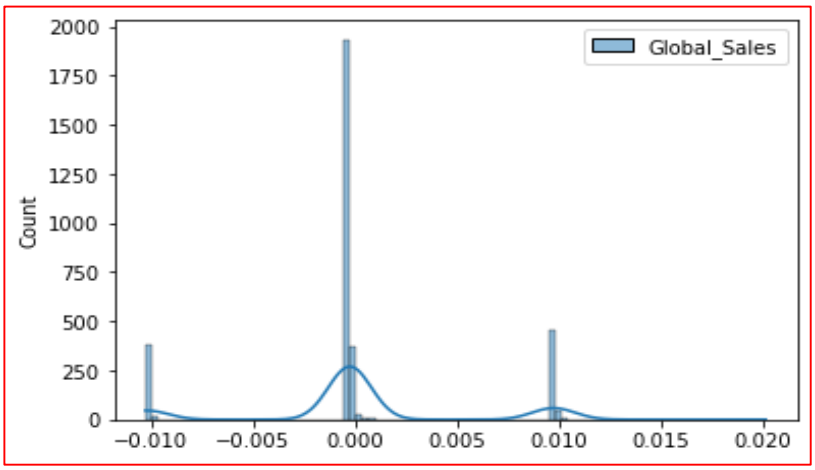
```
# The coefficients
print('Coefficients: \n', Sales_R.coef_)

Coefficients:
 [[0.99991848 0.99995408 0.99992023 0.99973702]]
```

Interpreting the coefficients:

- Holding all other features fixed, a 1 unit increase in NA_Sale is associated with an increase of 0.99 Global_Sales.
- Holding all other features fixed, a 1 unit increase in EU_Sales is associated with an increase of 0.99Global_Sales.
- Holding all other features fixed, a 1 unit increase in JP_Sales'is associated with an increase of 0.99 Global_Sales.
- Holding all other features fixed, a 1 unit increase in Other_Sales is associated with an increase of 0.99 Global_Sales.

I have gotten a very good model with a good fit. Let's quickly explore the residuals to make sure everything was okay with my data.



❖ **Check contain a constant or not:**

```
y-intercept:
 [0.00030232]
```

❖ **Check Multicollinearity**

	NA_Sales	EU_Sales	JP_Sales	Other_Sales	Global_Sales
NA_Sales	1.000000	0.768923	0.451283	0.634518	0.941269
EU_Sales	0.768923	1.000000	0.436379	0.726256	0.903264
JP_Sales	0.451283	0.436379	1.000000	0.290559	0.612774
Other_Sales	0.634518	0.726256	0.290559	1.000000	0.747964
Global_Sales	0.941269	0.903264	0.612774	0.747964	1.000000