

**数据挖掘**

**互评作业三**

**题　　目： 分类与预测**

**学 院： 计算机学院**

**专业名称： 计算机科学与技术**

**学 号： 3120191033**

**姓 名： 彭成**

**任课教师： 汤世平老师**

2020年5月29日星期五

# Video Game Sales 电子游戏销售分析

## 数据集介绍、探索方向及提交内容

该数据集包含游戏名称、类型、发行时间、发布者以及在全球各地的销售额数据。

数据量：11列共1.66W数据。

基于这个数据集，可进行以下问题的探索：

1. 电子游戏市场分析：受欢迎的游戏、类型、发布平台、发行人等；
2. 预测每年电子游戏销售额。
3. 可视化应用：如何完整清晰地展示这个销售故事。

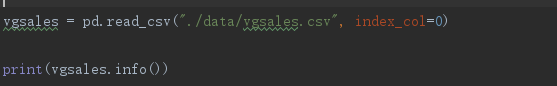
也可以自行发现其他问题，并进行相应的挖掘。

提交内容：

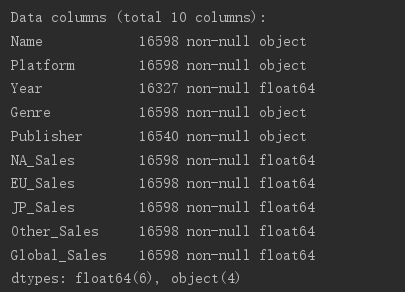
1. 对数据集进行处理的代码。
2. 数据挖掘代码。
3. 挖掘过程的报告：展示挖掘的过程、结果和你的分析。
4. 所选择的问题在README中说明，数据文件不要上传到Github中。

## 数据集预处理

首先，加载数据并观看每个属性的情况。

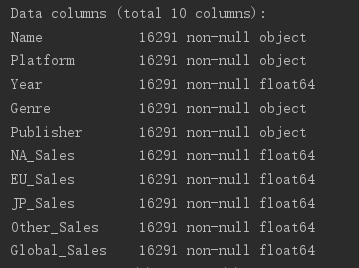


观察结果发现，数据缺失的情况并不严重。



由于数据missing的数量较少，我们可以用直接将缺失部分剔除的方法进行预处理。





最终要处理的数据集如上图所示。

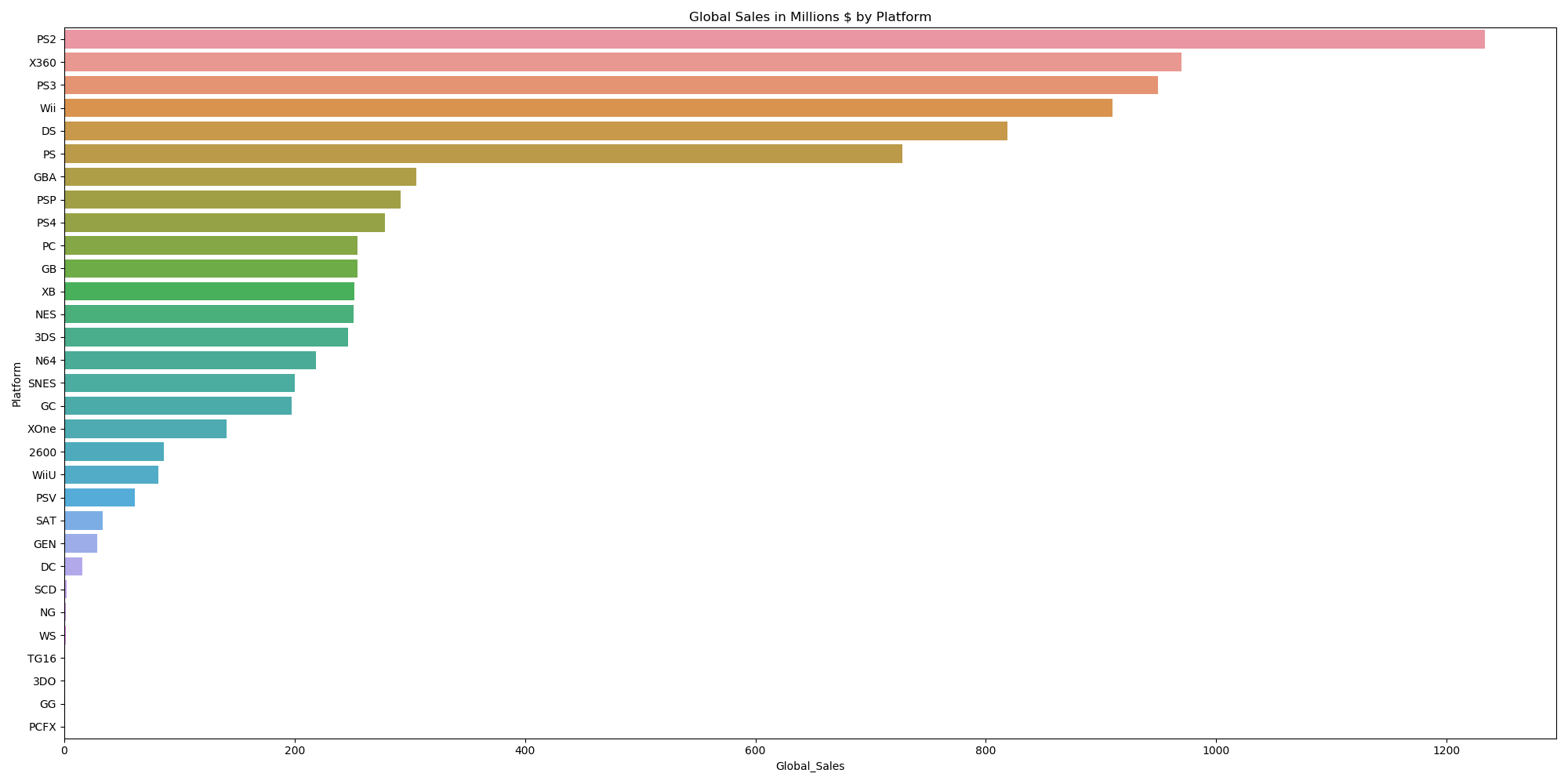
## 电子游戏市场分析

### 受欢迎的平台

我们通过将platform聚集在一起对数据进行分析，看那个平台的销售额最好，具体代码为：

|  |
| --- |
| plt.figure(figsize = (20,10))  sale\_for\_Platform = vgsales.groupby('Platform', as\_index = False).sum().sort\_values(by = 'Global\_Sales', ascending = False)  sns.barplot(x = 'Global\_Sales', y = 'Platform', data = sale\_for\_Platform)  plt.title('Global Sales in Millions $ by Platform')  plt.show() |

输出的图像为：



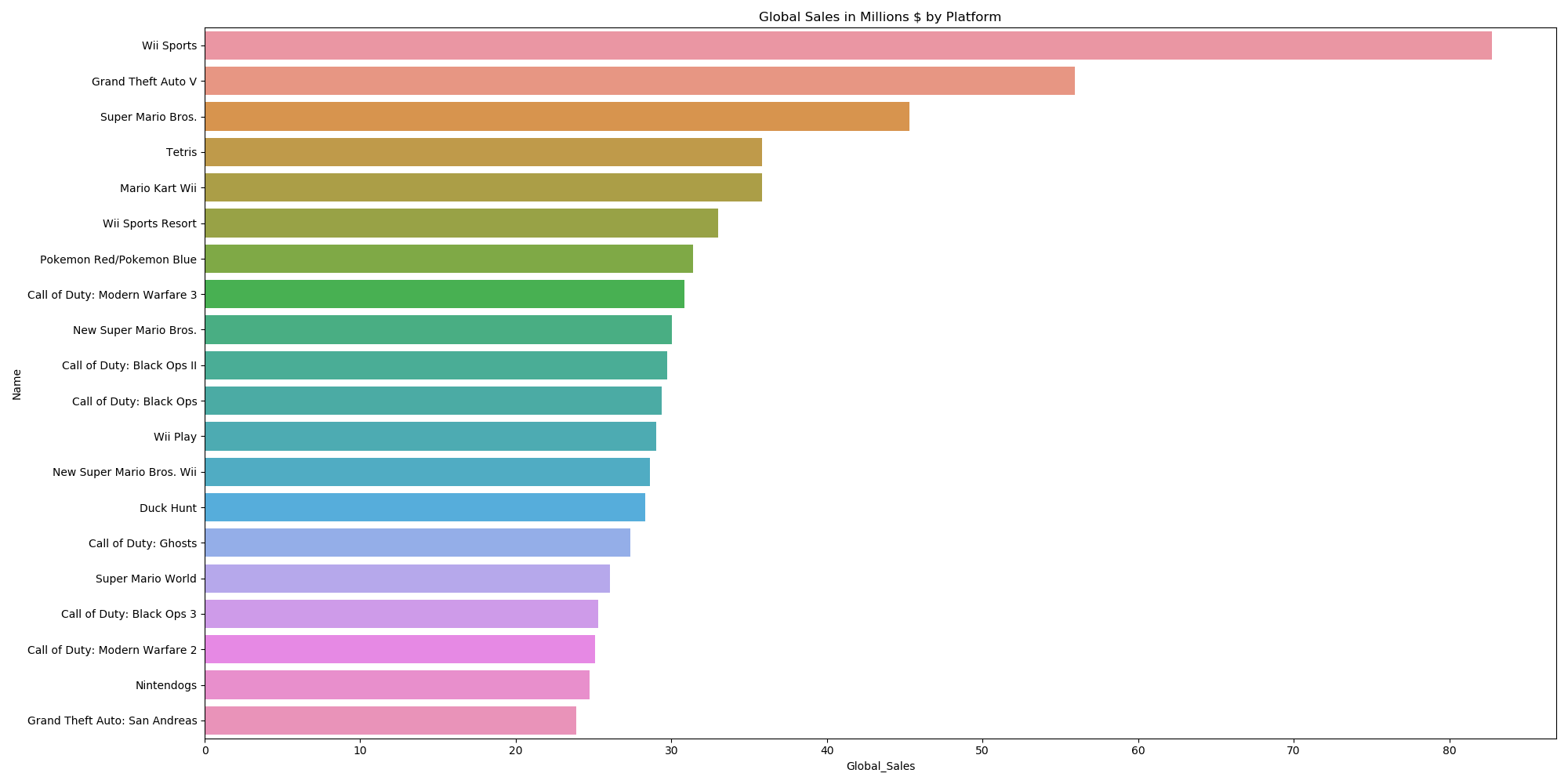
我们可以分析得出，最受欢迎平台为PS2，同时还有X360、PS3、Wii、DS等同样非常受欢迎。

### 受欢迎的游戏

我们通过将Name聚集在一起对数据进行分析，看那个平台的销售额最好，具体代码为：

|  |
| --- |
| plt.figure(figsize = (20,10))  sale\_for\_game = vgsales.groupby('Name', as\_index = False).sum().sort\_values(by = 'Global\_Sales', ascending = False).head(20)  sns.barplot(x = 'Global\_Sales',  y = 'Name', data = sale\_for\_game)  plt.title('Global Sales in Millions $ by Game')  plt.show() |

输出的图像为：



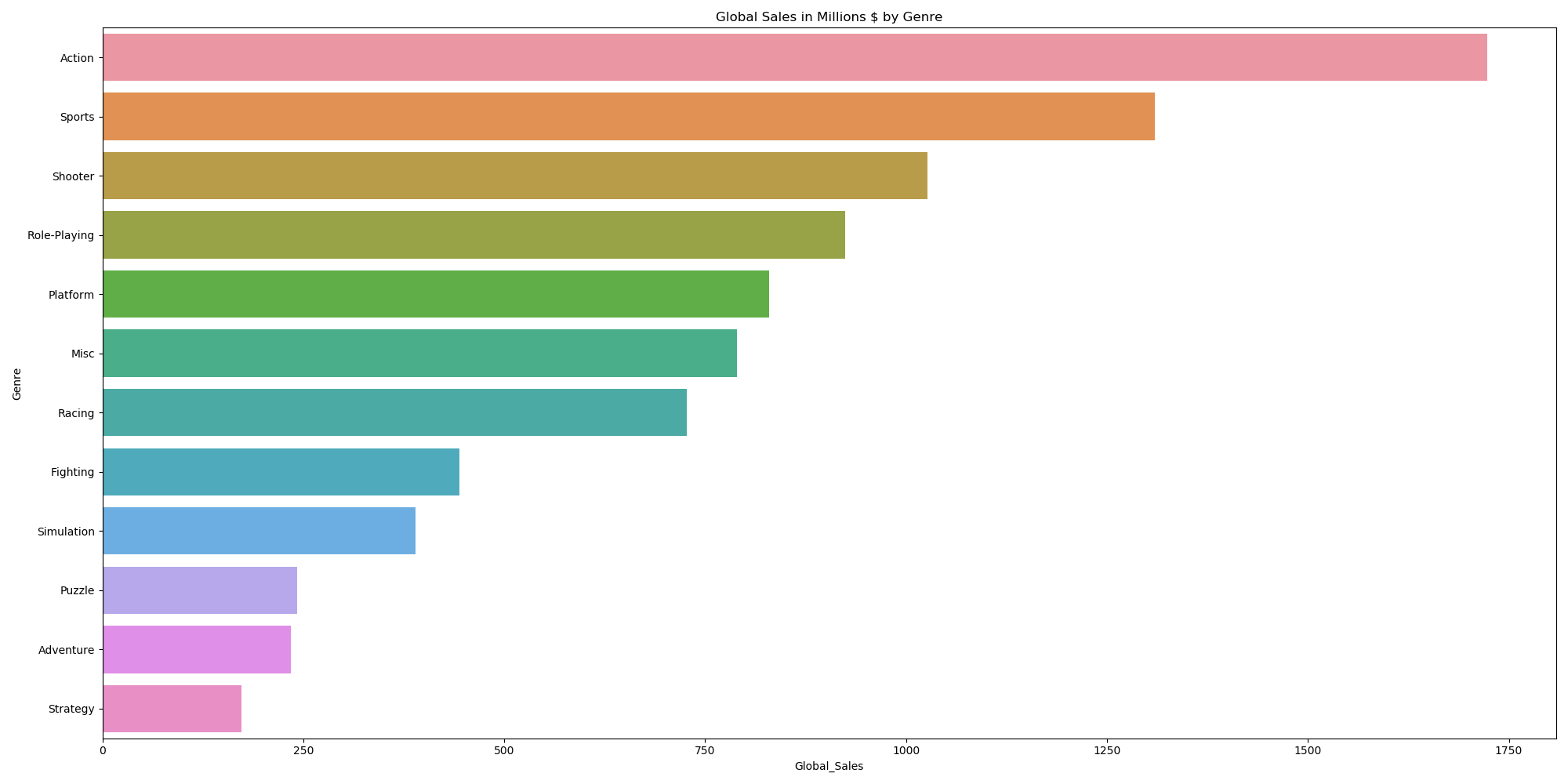
我们可以分析得出，最受欢迎的游戏为Wii Sports，同时还有GTA5、Super Mario Bros、Tetris等同样非常受欢迎。

### 受欢迎的游戏类型

我们通过将Genre聚集在一起对数据进行分析，看那个平台的销售额最好，具体代码为：

|  |
| --- |
| plt.figure(figsize = (20,10))  sale\_for\_Genre = vgsales.groupby('Genre', as\_index = False).sum().sort\_values(by = 'Global\_Sales', ascending = False).head(20)  sns.barplot(x = 'Global\_Sales', y = 'Genre', data = sale\_for\_Genre)  plt.title('Global Sales in Millions $ by Genre')  plt.show() |

输出图形为：



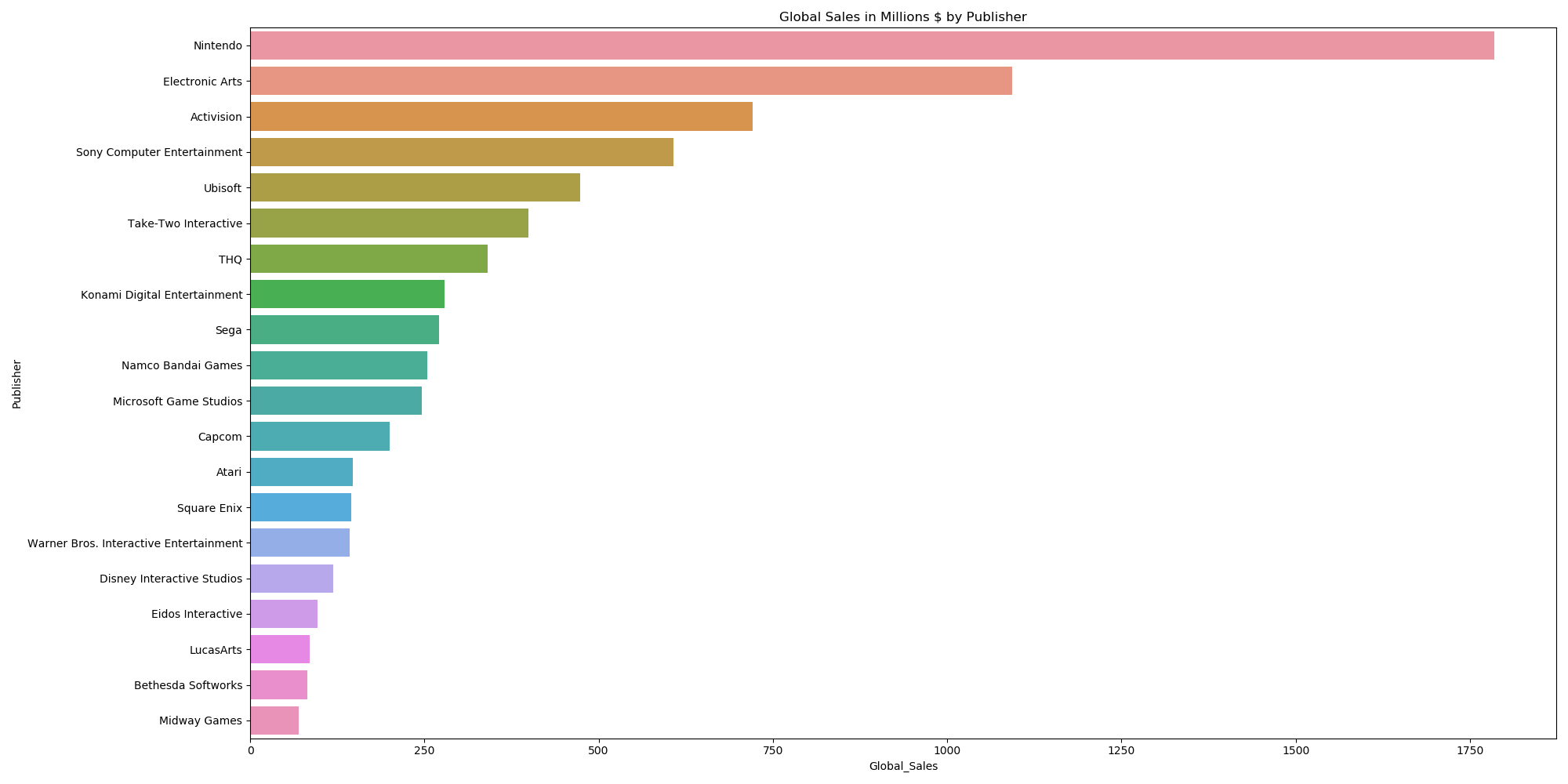
我们可以分析得出，最受欢迎的游戏类型为Action动作类游戏，同时还有运动类、设计类、角色扮演类等同样非常受欢迎。

### 受欢迎的游戏发行商

我们通过将Publisher聚集在一起对数据进行分析，看那个平台的销售额最好，具体代码为：

|  |
| --- |
| plt.figure(figsize = (20,10))  sale\_for\_Publisher = vgsales.groupby('Publisher', as\_index = False).sum().sort\_values(by = 'Global\_Sales', ascending = False).head(20)  sns.barplot(x = 'Global\_Sales', y = 'Publisher', data = sale\_for\_Publisher)  plt.title('Global Sales in Millions $ by Publisher')  plt.show() |

输出图形为：



我们可以分析得出，最受欢迎的游戏发行商为Nintendo动作类游戏，同时还有EA、Activision、Sony等同样非常受欢迎。

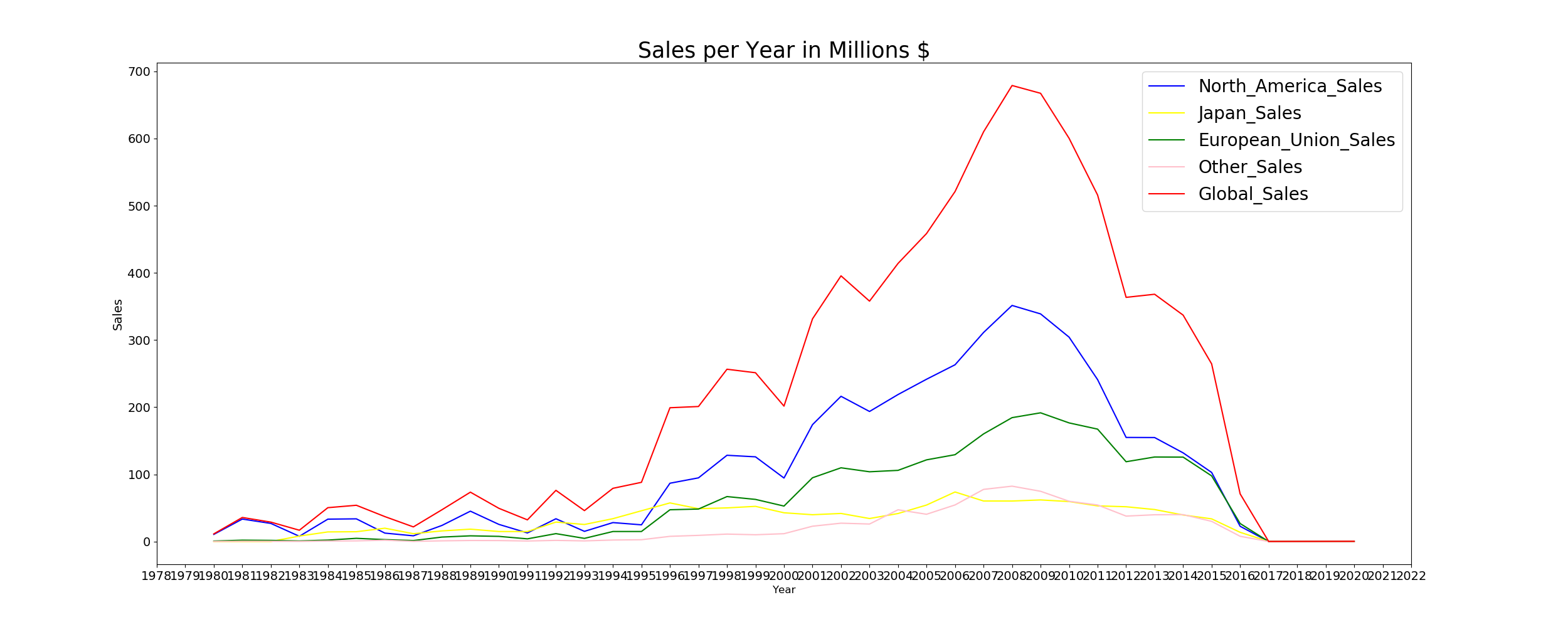
## 预测每年的电子游戏销售额

### 每年的销售金额分析

首先，我们来看每年的电子游戏销售情况，具体代码为：

|  |
| --- |
| left, bottom, width, height = 0.1, 0.1, 0.8, 0.8  fig = plt.figure(figsize = (25,10))  ax = fig.add\_axes([left, bottom, width, height])  sale\_for\_years = vgsales.groupby('Year', as\_index = False).sum()  sns.lineplot(x = 'Year', y = 'NA\_Sales', data = sale\_for\_years, color = 'blue', label = 'North\_America\_Sales', ax = ax)  sns.lineplot(x = 'Year', y = 'JP\_Sales', data = sale\_for\_years, color = 'yellow', label = 'Japan\_Sales', ax = ax)  sns.lineplot(x = 'Year', y = 'EU\_Sales', data = sale\_for\_years, color = 'green', label = 'European\_Union\_Sales', ax = ax)  sns.lineplot(x = 'Year', y = 'Other\_Sales', data = sale\_for\_years, color = 'pink', label = 'Other\_Sales', ax = ax)  sns.lineplot(x = 'Year', y = 'Global\_Sales', data = sale\_for\_years, color = 'red', label = 'Global\_Sales', ax = ax)  ax.set\_xlabel('Year', fontsize = 12)  ax.set\_ylabel('Sales', fontsize = 14)  ax.tick\_params(axis='both', which='major', labelsize=14)  ax.xaxis.set\_major\_locator(ticker.MultipleLocator(1))  ax.set\_title('Sales per Year in Millions $', fontsize = 25)  plt.legend(fontsize = 20)  plt.show() |

输出的图像为：



我们可以分析得出，自1978年到2008年，不管是全球的销售金额还是地区的销售金额都在稳步上升，但是2009年之后，电子游戏的销售额大幅度下降。

原因可能是市场不景气，但更大的可能是数据集中统计不完全。根据我对数据集的观察，2017年之后的数据十分不完整，所以图像中2017年之后的可以摒弃不看，但2009年至2016年的数据是否统计完整不得而知。

### 通过北美洲的销售额来预测全球的销售额

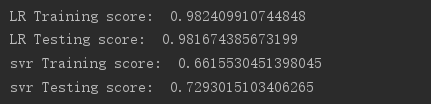
由上文分析可得，我们只使用2016年以前的数据进行预测。



然后将北美销售额和全球销售额提取出来作为输入和预测值。



采用LinearRegression和SVR(kernel="poly")两种回归模型分别对其进行预测，并得到它们预测的得分。结果如下：



代码为：

|  |
| --- |
| vgsales = vgsales[vgsales['Year'] <= 2016.0]  sale\_for\_years = vgsales.groupby('Year', as\_index = False).sum()  print(sale\_for\_years.info())  x = sale\_for\_years['NA\_Sales'].values.reshape(-1, 1)  y = sale\_for\_years['Global\_Sales']  X\_train, X\_test, y\_train, y\_test = train\_test\_split(x, y, test\_size=0.2, random\_state = 1)  LR = LinearRegression()  LR.fit(X\_train, y\_train)  LR\_score\_train = LR.score(X\_train, y\_train)  print('LR Training score: ',LR\_score\_train)  LR\_score\_test = LR.score(X\_test, y\_test)  print('LR Testing score: ',LR\_score\_test)  svr = SVR(kernel="poly")  svr.fit(X\_train, y\_train)  svr\_score\_train = svr.score(X\_train, y\_train)  print('svr Training score: ', svr\_score\_train)  svr\_score\_test = svr.score(X\_test, y\_test)  print('svr Testing score: ', svr\_score\_test) |

### 通过年份来预测某几年的销售额

将年份和全球销售额提取出来作为输入和预测值。



使用LinearRegression作为回归模型，得到结果如下：



代码为：

|  |
| --- |
| vgsales = vgsales[vgsales['Year'] <= 2016.0]  sale\_for\_years = vgsales.groupby('Year', as\_index = False).sum()  print(sale\_for\_years.info())  x = sale\_for\_years['Year'].values.reshape(-1, 1)  y = sale\_for\_years['Global\_Sales']  X\_train, X\_test, y\_train, y\_test = train\_test\_split(x, y, test\_size=0.05, random\_state = 1)  LR = LinearRegression()  LR.fit(X\_train, y\_train)  LR\_score\_train = LR.score(X\_train, y\_train)  print('LR Training score: ',LR\_score\_train)  LR\_score\_test = LR.score(X\_test, y\_test)  print('LR Testing score: ',LR\_score\_test) |

综上所述我们可以发现，依照局部地区来预测全球的销售额效果要比直接按照年份来预测的效果要好。