北京理工大学
BEIJING INSTITUTE OF TECHNOLOGY

数据挖掘
互评作业二

题　　目：频繁模式与关联规则挖掘

学　　院：计算机学院

专业名称：计算机科学与技术

学　　号：3220200998

姓　　名：余晓

任课教师：汤世平老师

本次作业仓库地址：

https://github.com/BruceYu-Bit/DataMiningSecond

一、 整体任务

任务名：Video Game Sales 电子游戏销售分析

数据集：Video Game Sales

数据集介绍：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。数据量约 11 列共 1.66W 条数据。

问题探索：

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。

二、数据集探索

2.1 数据集介绍

在本次作业中，我们所要处理的是对 Video Game Sales 进行电子游戏销售分析。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，数据量为 11 列共 1.66w 条数据：

数据集相关信息：

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import matplotlib.ticker as ticker
import seaborn as sns

from sklearn.linear_model import LinearRegression
from sklearn.svm import SVR
from sklearn.model_selection import train_test_split


vgsales = pd.read_csv("./vgsales.csv", index_col=0)
print('查看数据集信息:')
print(vgsales.info())
print('数据集的属性值如下：')
print(vgsales.columns.values)
print('数据集的前五列数据如下：')
vgsales.head()
```

相应的数据集信息结果如下：

```
查看数据集信息:
<class 'pandas.core.frame.DataFrame'>
Int64Index: 16598 entries, 1 to 16600
Data columns (total 10 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   Name          16598 non-null  object
 1   Platform      16598 non-null  object
 2   Year          16327 non-null  float64
 3   Genre         16598 non-null  object
 4   Publisher     16540 non-null  object
 5   NA_Sales      16598 non-null  float64
 6   EU_Sales      16598 non-null  float64
 7   JP_Sales      16598 non-null  float64
 8   Other_Sales   16598 non-null  float64
 9   Global_Sales  16598 non-null  float64
dtypes: float64(6), object(4)
memory usage: 1.4+ MB
None
数据集的属性值如下:
['Name' 'Platform' 'Year' 'Genre' 'Publisher' 'NA_Sales' 'EU_Sales'
 'JP_Sales' 'Other_Sales' 'Global_Sales']
数据集的前五列数据如下:
```

Rank	Name	Platform	Year	Genre	Publisher	NA_Sales	EU_Sales	JP_Sales	Other_Sales	Global_Sales
1	Wii Sports	Wii	2006.0	Sports	Nintendo	41.49	29.02	3.77	8.46	82.74
2	Super Mario Bros.	NES	1985.0	Platform	Nintendo	29.08	3.58	6.81	0.77	40.24
3	Mario Kart Wii	Wii	2008.0	Racing	Nintendo	15.85	12.88	3.79	3.31	35.82
4	Wii Sports Resort	Wii	2009.0	Sports	Nintendo	15.75	11.01	3.28	2.96	33.00
5	Pokemon Red/Pokemon Blue	GB	1996.0	Role-Playing	Nintendo	11.27	8.89	10.22	1.00	31.37

2.2 数据集预处理

由上面表格分析可以得知，总共包含 11 列数据：Name、Platform、Year、Genre、Publisher、NA_Sales、EU_Sales、JP_Sales、Other_Sales、Global_Sales。由于数据集缺失的数据集较少，我们采用对缺失数据值直接剔除的策略进行预处理。处理后的数据总量为 16291 条。

```
#数据集预处理
vgsales = vgsales.dropna(how='any')
print(vgsales.info())
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 16291 entries, 1 to 16600
Data columns (total 10 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   Name          16291 non-null  object
 1   Platform      16291 non-null  object
 2   Year          16291 non-null  float64
 3   Genre         16291 non-null  object
 4   Publisher     16291 non-null  object
 5   NA_Sales      16291 non-null  float64
 6   EU_Sales      16291 non-null  float64
 7   JP_Sales      16291 non-null  float64
 8   Other_Sales   16291 non-null  float64
 9   Global_Sales  16291 non-null  float64
dtypes: float64(6), object(4)
memory usage: 1.4+ MB
None
```

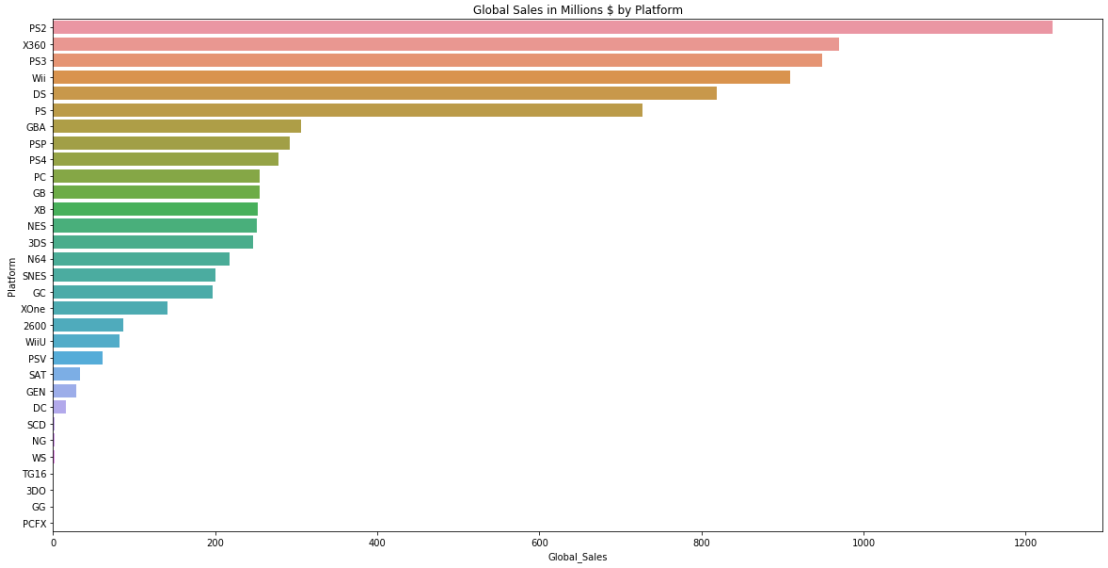
三、电子游戏市场分析

3.1 受欢迎的平台

通过对 platform 字段进行分组分析，可以得到最受欢迎的平台，具体量化为其销售额的高低，实现代码如下：

```
#各电子游戏平台销售额情况

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()
```

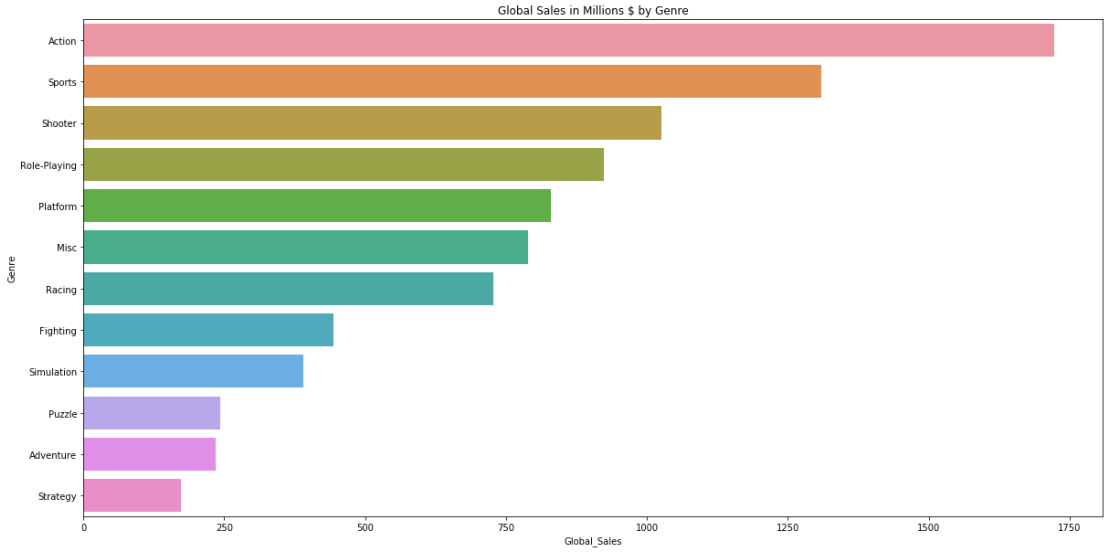


分析可得，最受欢迎的平台为 PS2，同时还有 GTA5、Super Mario Bros、Tetris 等同样非常受欢迎。

3.2 受欢迎的游戏类型

同理，我们对 Genre 字段进行分组，得到最受欢迎的游戏类型。

```
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()
```

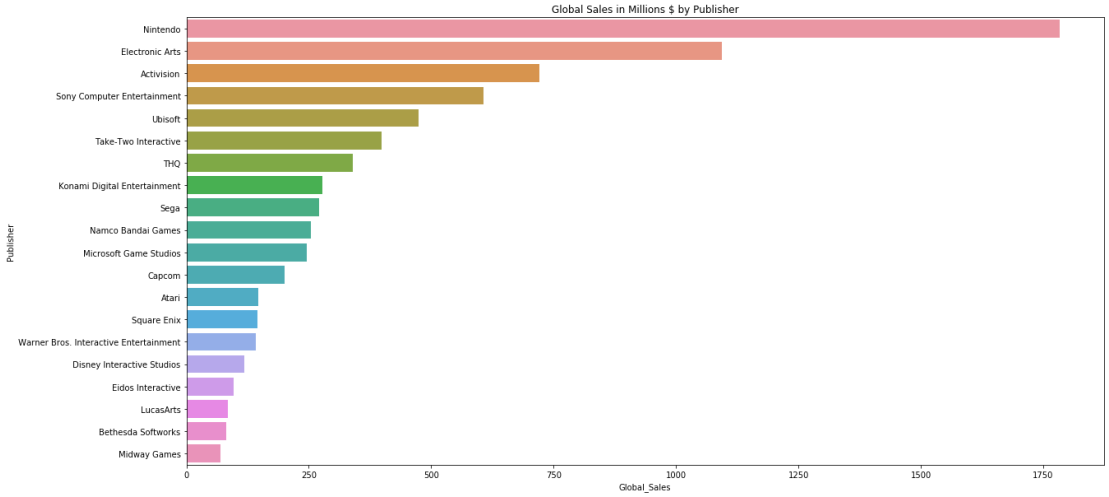


分析可得，最受欢迎的游戏类型为 Action 动作类游戏，同时还有运动类、设计类、角色扮演类等同等受欢迎。

3.3 受欢迎的游戏发行商

我们对 Publisher 字段进行分组，得到最受欢迎的游戏发行商。

```
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()
```



分析可得，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。

四、预测每年电子游戏销售额

4.1 可视化分析

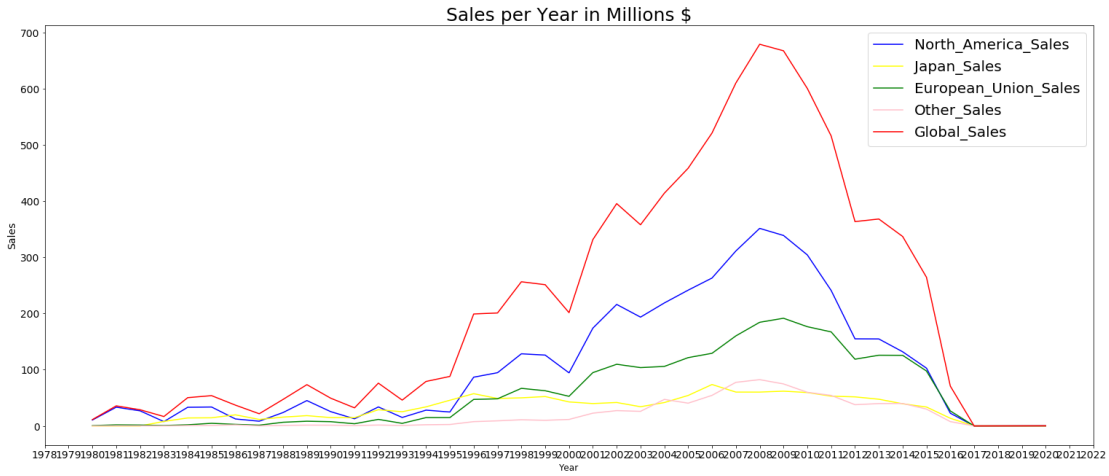
我们首先对每年的销售金额进行可视化，分析近些年的销售金额增长趋势。

```
#每年电子游戏销售额情况
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()
```



分析可得，从 1978 年到 2008 年，全球和地区的销售金额都在稳步上升，但在 2009 年后，电子游戏的销售额大幅度下降。原因可能是时长不景气，但更大的可能是数据统计不完全。

4.2 全球销售额预测

由上文的初步分析可得，我们只使用 2016 年以前的数据。将北美销售额和全球销售额作为输入和预测值，对其进行预测，并得到预测的得分，结果如下：

```
#通过北美洲的销售额来预测全球的销售额

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)
```

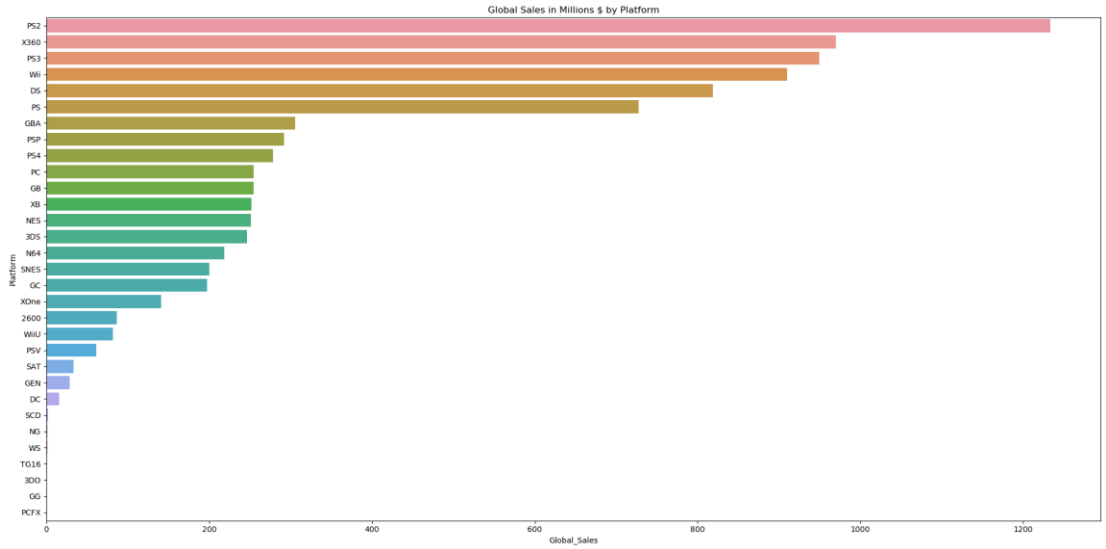
采用 SVR 和 LR（LinearRegression）两种模型分别对其进行回归预测，得到预测得分如下：

```
LR Training score:  0.982409910744848
LR Testing score:  0.981674385673199
svr Training score:  0.6615530451398045
svr Testing score:  0.7293015103406265
```

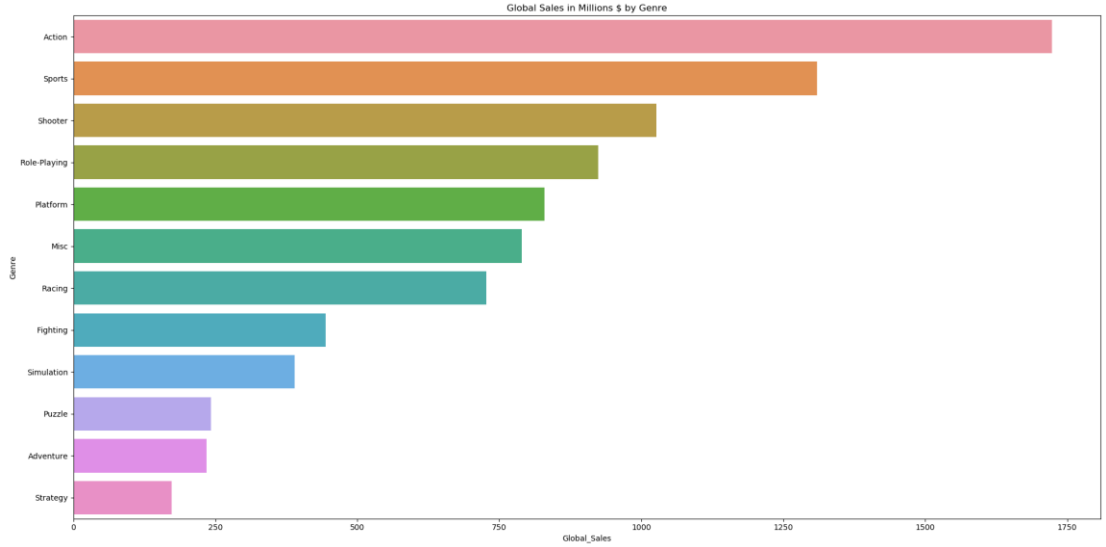
可以看出 LR 模型的预测效果较好。

五、可视化应用

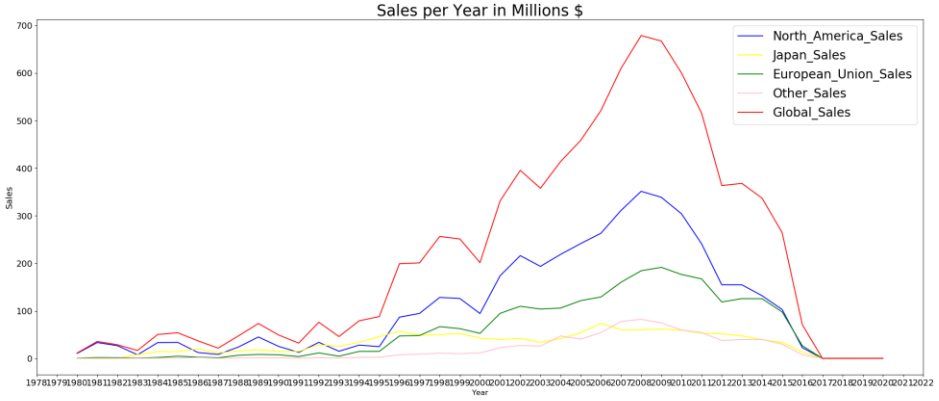
由上述对平台的分析可知，并不是只有一家平台独大，而是 PS2、X360 等平台齐头并进，所以推出的游戏尽量能支持多平台，才能推广向更广大的游戏受众。



同时，通过数据分析挖掘可知，应该向动作、运动、设计、角色扮演等受众较大的方向开发游戏。



电子游戏的销售总额在下降，证明现有的游戏种类已经让市场有些疲乏，应该做些有创新性的游戏，发明些新的玩法来刺激市场。


Sales per Year in Millions $
North_America_Sales
Japan_Sales
European_Union_Sales
Other_Sales
Global_Sales
Sales
Year