Assignment 12

March 7, 2021

0.0.1 Portfolio assignment 12

30 min: Perform a bivariate analysis on at least 3 combinations of a numerical column with a categorical column in the dataset that you chose in portfolio assignment 4. Use .groupby('columnname').mean() to calculate the means. Is there a difference between categories? Then use seaborn barplots to check if there is a statistically significant difference.

[1]:
```
import pandas as pd
import seaborn as sns
```

My previous dataset did not have much numerical data, so i'm using a different one.

[2]:
```
vg = pd.read_csv('vgsales.csv')
vg.head()
```

[2]:

	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
1	2	Super Mario Bros.	NES	1985.0	Platform	Nintendo	29.08	3.58	6.81	0.77	40.24
2	3	Mario Kart Wii	Wii	2008.0	Racing	Nintendo	15.85	12.88	3.79	3.31	35.82
3	4	Wii Sports Resort	Wii	2009.0	Sports	Nintendo	15.75	11.01	3.28	2.96	33.00
4	5	Pokemon Red/Pokemon Blue	GB	1996.0	Role-Playing	Nintendo	11.27	8.89	10.22	1.00	31.37

[3]:
```
vg.groupby('Genre').mean()
```

[3]:

Genre	Rank	Year	NA_Sales	EU_Sales	JP_Sales
Action	7973.879071	2007.909929	0.264726	0.158323	0.048236
Adventure	11532.787714	2008.130878	0.082271	0.049868	0.040490
Fighting	7646.511792	2004.630383	0.263667	0.119481	0.103007
Misc	8561.847039	2007.258480	0.235906	0.124198	0.061967
Platform	6927.251693	2003.820776	0.504571	0.227573	0.147596
Puzzle	9627.381443	2005.243433	0.212680	0.087251	0.098471

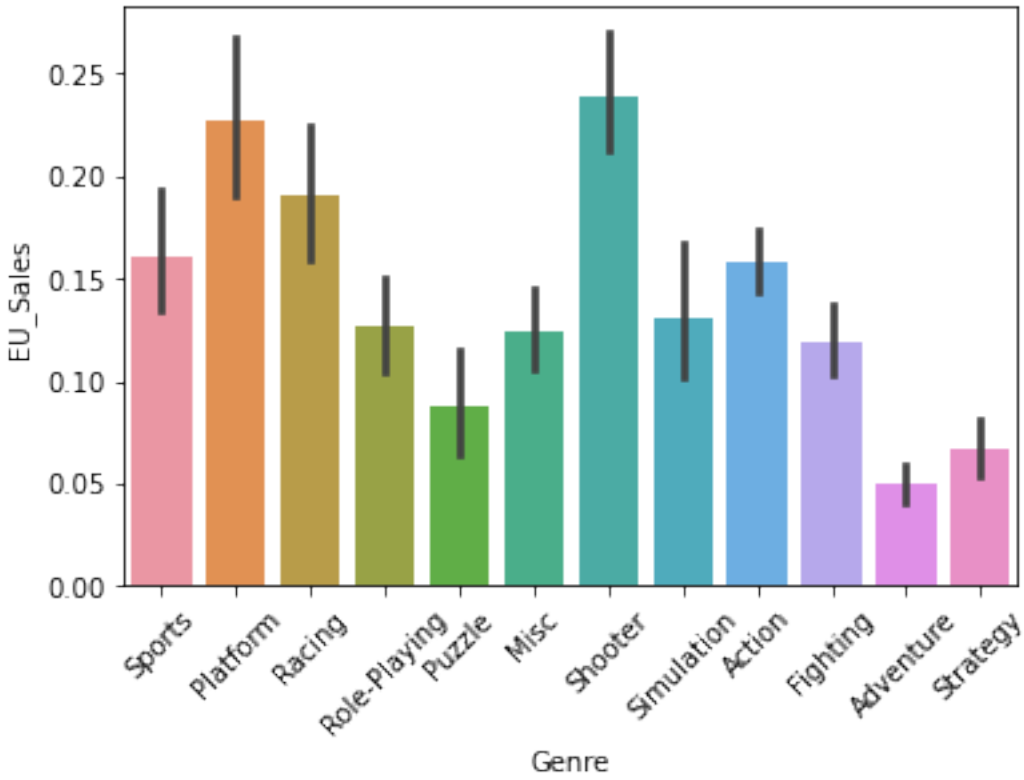
```
Racing          7961.515612  2004.840131  0.287766  0.190865  0.045388
Role-Playing    8086.174731  2007.055744  0.219946  0.126384  0.236767
Shooter         7369.367939  2005.918877  0.444733  0.239137  0.029221
Simulation      8626.085352  2006.567568  0.211430  0.130773  0.073472
Sports          7425.026428  2005.477865  0.291283  0.160635  0.057702
Strategy       10071.897210  2005.599106  0.100881  0.066579  0.072628

               Other_Sales  Global_Sales
Genre
Action            0.056508      0.528100
Adventure         0.013072      0.185879
Fighting          0.043255      0.529375
Misc              0.043312      0.465762
Platform          0.058228      0.938341
Puzzle            0.021564      0.420876
Racing            0.061865      0.586101
Role-Playing      0.040060      0.623233
Shooter           0.078389      0.791885
Simulation        0.036355      0.452364
Sports            0.057532      0.567319
Strategy          0.016681      0.257151
```

Ofcourse we can scrap datatypes like Year, since they don't serve much value here.

[4]:
```
chart = sns.barplot(x='Genre', y='EU_Sales', data=vg)
chart.set_xticklabels(chart.get_xticklabels(), rotation=45)
chart
```

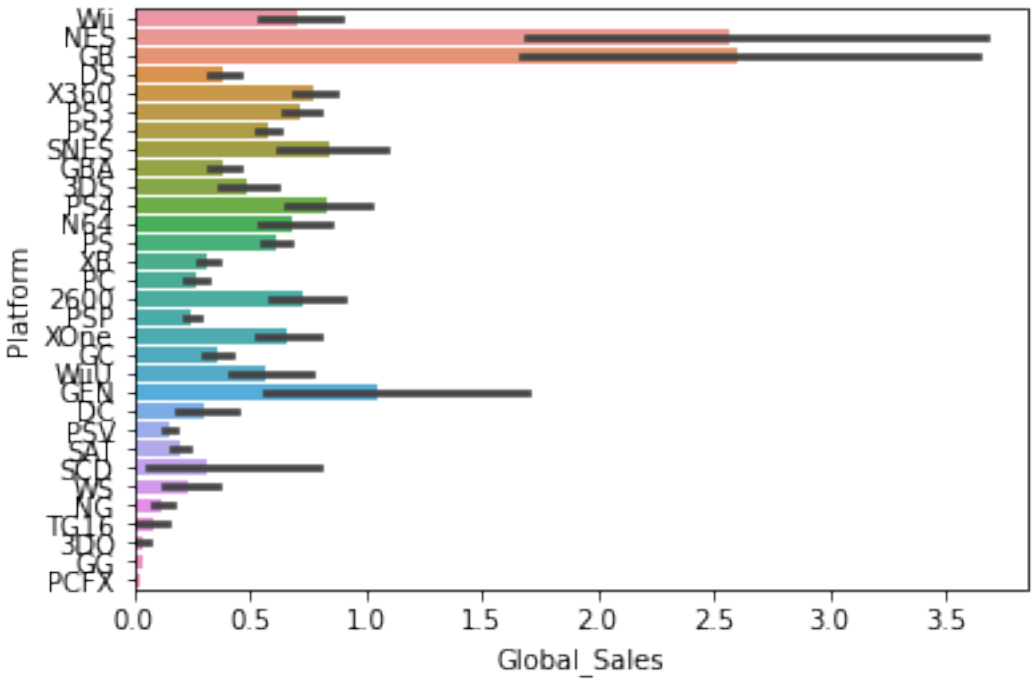
[4]:
```
<AxesSubplot:xlabel='Genre', ylabel='EU_Sales'>
```



We can conclude that shooter and platform are the most popular in EU.

[5]:
```
sns.barplot(y='Platform', x='Global_Sales', data=vg)
```

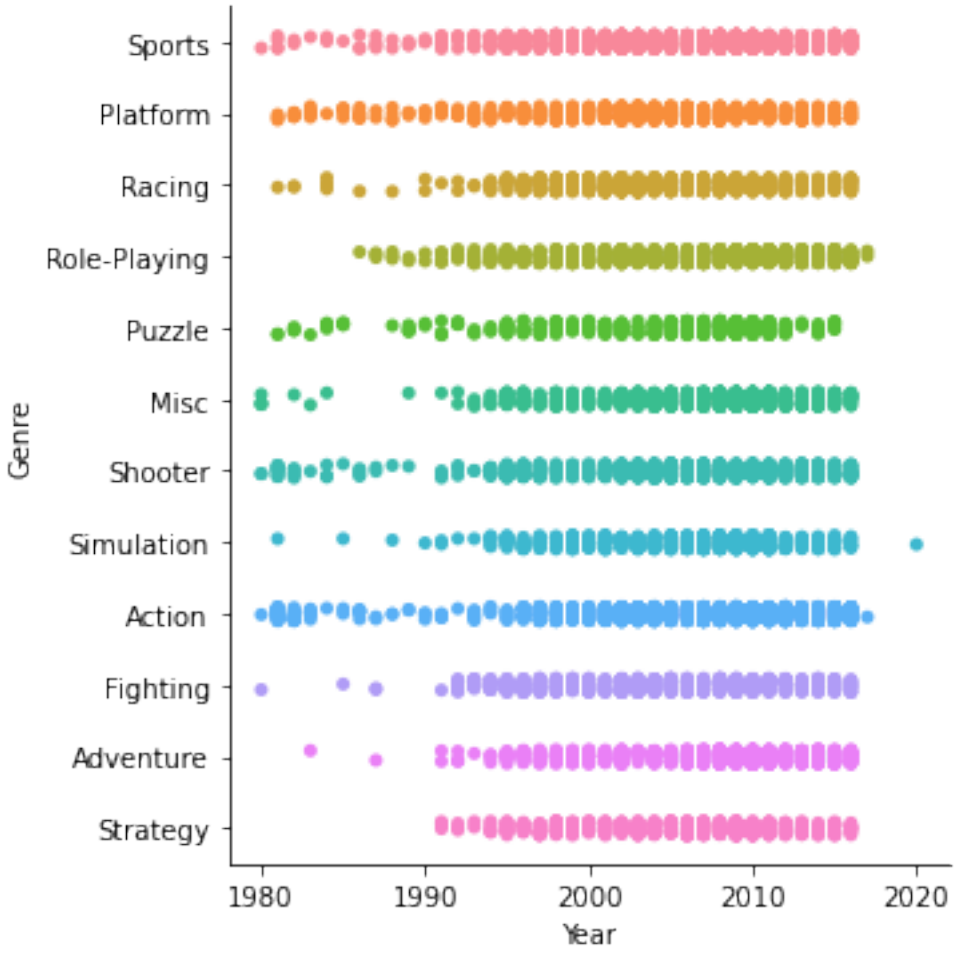
[5]:
```
<AxesSubplot:xlabel='Global_Sales', ylabel='Platform'>
```



We can conclude that the GameBoy made the most global sales

[6]:
```
sns.catplot(y='Genre', x='Year', data=vg)
```

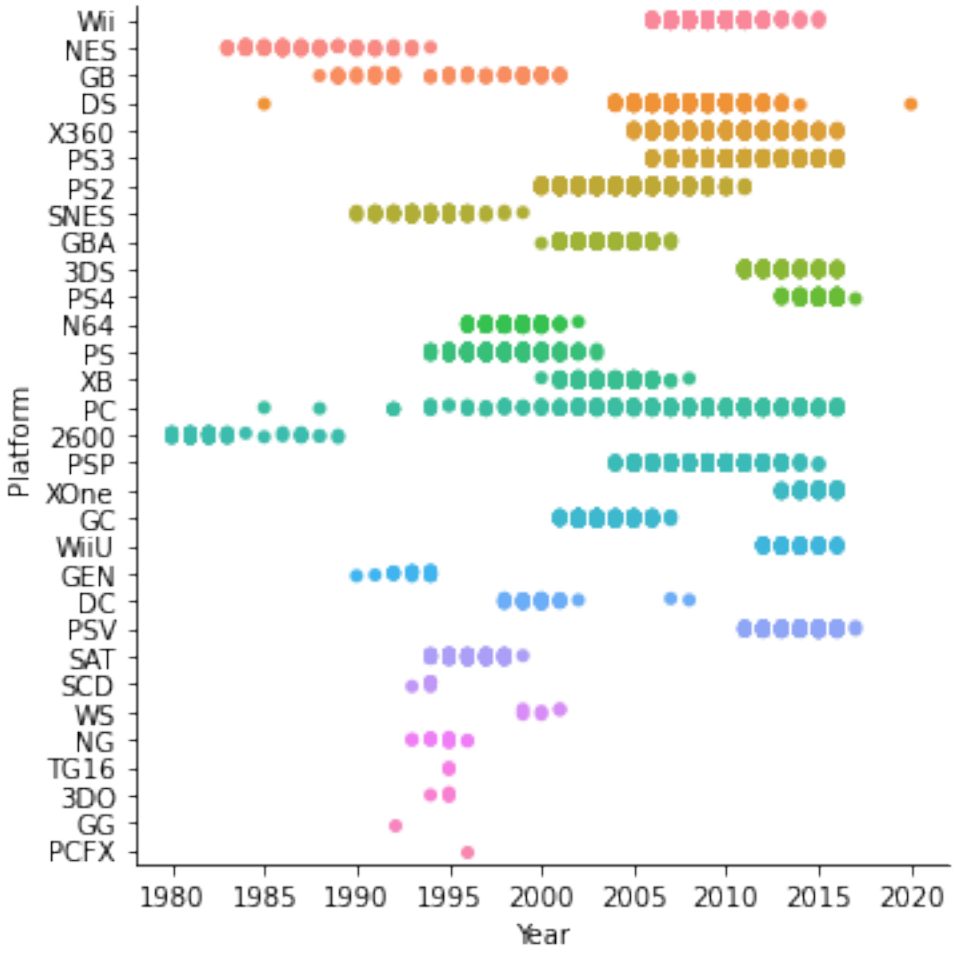
[6]:
```
<seaborn.axisgrid.FacetGrid at 0x1d0d5e862b0>
```



An accidental, but quite beautiful graph in my opinion. Here you can visualize when every genre started to become popular. I wanted to do the same with the different platforms.

[7]:
```
sns.catplot(y='Platform', x='Year', data=vg)
```

[7]:
```
<seaborn.axisgrid.FacetGrid at 0x1d0d63640d0>
```



Maybe I'm the only one, but I really love catplot's visualization here. You can even see when the NES started dying out, the GameBoy sales came in, and after that the Wii sales.