CSC 532 Machine Learning Project

Jerry Parng

Abstract

When it comes to video games, there are so many different genres such as action or role-playing that there is enough fun and enjoyment for everyone to discover. In the video game industry, trying to find the genre of video game that is the most popular with the consumers and producing those can make for a profitable product. This project will try and classify video game genres using Random forest, Gradient boosted tree, Neural network, SVM linear, and SVM radial models.

Problem Definition and Goals

The problem that this project is looking to solve is to classify video game genres based on several attributes of a video game such as sales, platform, and publisher. For this project, the dataset was taken from Kaggle.com and contains 11,493 observations with 11 features. These features are:

- Rank - Sales rank of each game, with number 1 being the most sold game (numerical)
- Name - Name of the game (categorical)
- Platform - System used for the game (categorical)
- Year - Year the game was developed (numerical)
- Genre - The genre of the game (categorical)
- Publisher - Company that published the game (categorical)
- Na_sales - Number of games sold in North America (numerical)
- Jp_sales - Number of games sold in Japan (numerical)
- Eu_sales - Number of games sold in EU (numerical)
- Other_sales - Number of games sold everywhere else (numerical)
- Global_sales - Total number of games sold worldwide (numerical)

The dataset requires some cleaning before any machine learning models can be trained. Some preprocessing needs to be done with the attributes and will be explained in more detail in the Data Exploration section. After the preparations for the dataset are made, the model's can

be trained and fitted with the dataset, and then their performances will be evaluated based on accuracy and AUC (Area Under Curve).

Related Work

Being able to predict the type of video game genre can be very important to a company in order for them to seek out the most profitable game that they can produce. In that sense, several other users have taken this dataset and made their own analyses. For example, one user by the name of NIHAR14 posted their analysis on Kaggle by using pie charts and bar graphs and concluded that the most popular genre in Japan was role-playing, while action would be the more popular genre in both Europe and North America.[2]

Another user by the name of Alfaradi Krisna Ocysta on Kaggle also reviewed the data and viewed the popularity of video game genres each year by sales. They found that the peak of video games sales was between 2007 and 2008 with the highest selling genres being action, shooter, and sports.[3]

Another study from Yuhang Jiang and Lukun Zheng also researched deep learning for the video game genre by using image-based, text-based, and multi-modal models to classify video game genres from a dataset containing 50,000 video games from IGDB.com (video game database). From their image-based models, they were able to get accuracies of 31.4% and 61.7%. With their text-based models, they achieved 44.3% and 72.1% accuracies. Finally, their multi-modal model had achieved accuracies of 49.9% and 79.9%. [4]

Data Exploration

To start with, the data needs to be preprocessed before the models are trained and fitted with training data. The name attribute is redundant to the data as the rank can still be used for the identity of the games so it shall be removed. There were 271 rows with some missing values and were removed as the small amount would not have a great effect on the data.The publisher attribute has a lot of unique levels that would become too complex if one-hot-encoded so instead embeddings will need to be used to encode this attribute when training and fitting specific models.

To find out whether our genre attribute is associated with the other attributes in the dataset, tests and plots will be employed. Due to how the dataset contains both categorical and numerical attributes, mosaic plots and chi-square tests would be used for the categorical attributes while the side-by-side boxplots and ANOVA tests would be used for the numerical attributes.

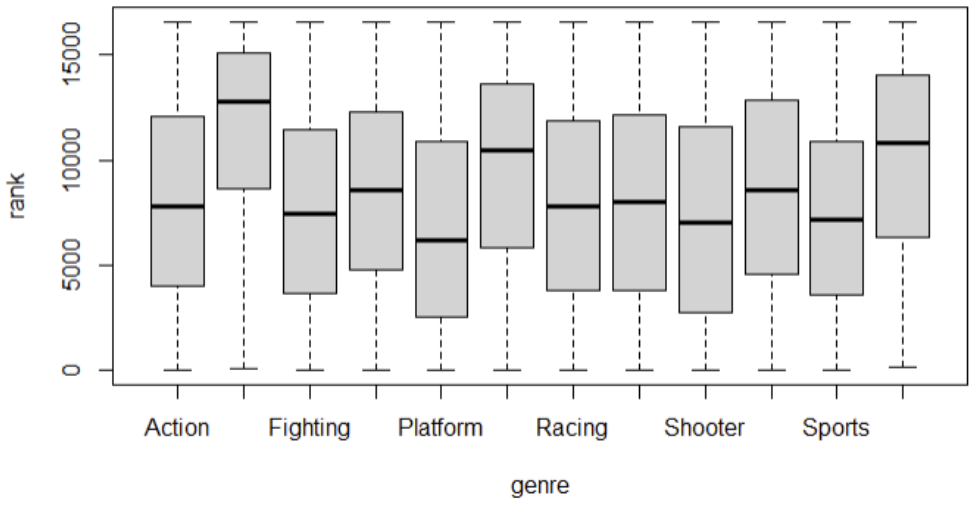

Figure 1: Boxplot of genre/rank

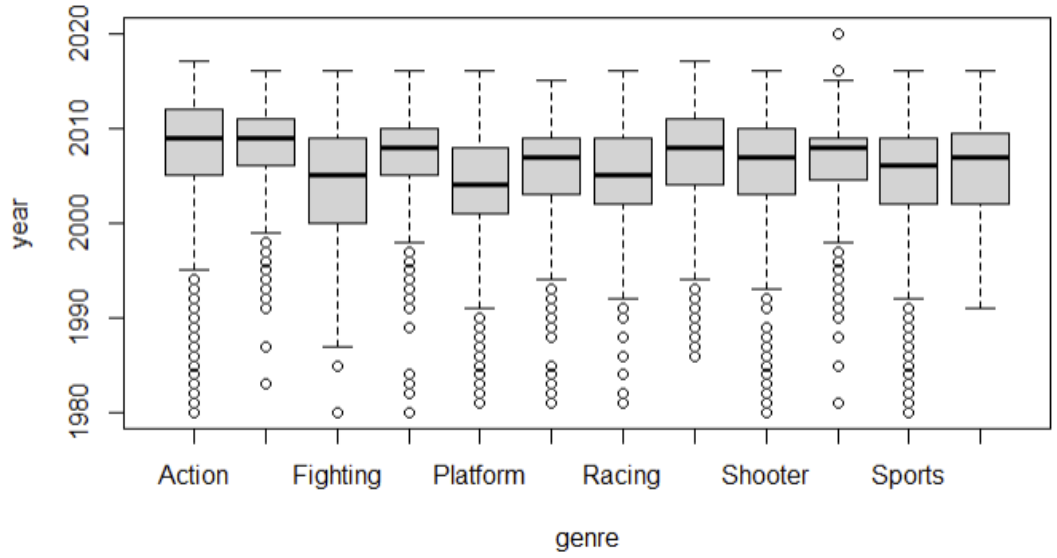

Figure 2: Boxplot of genre/year

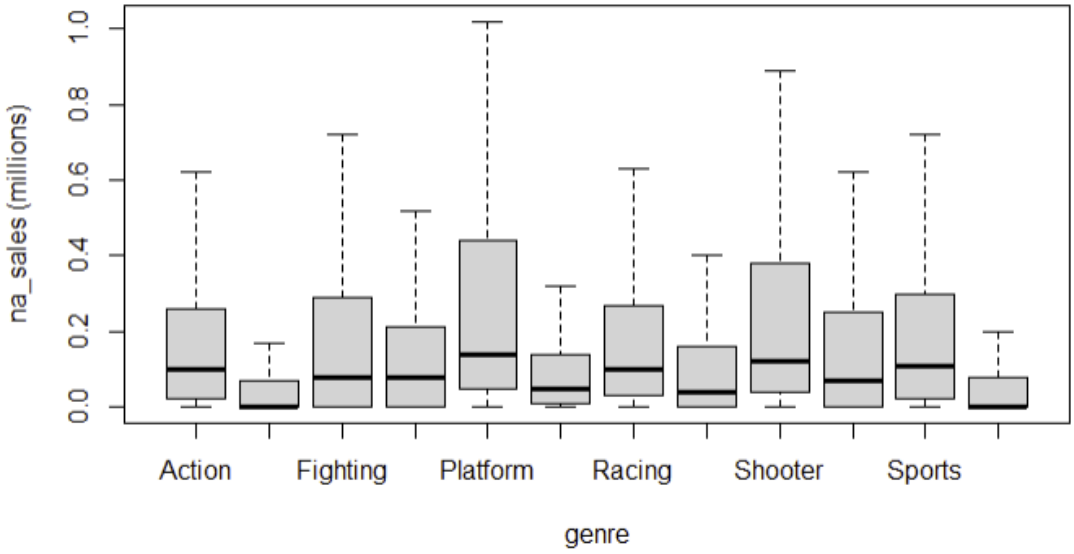

Figure 3: Boxplot of na_sales(millions)/genre

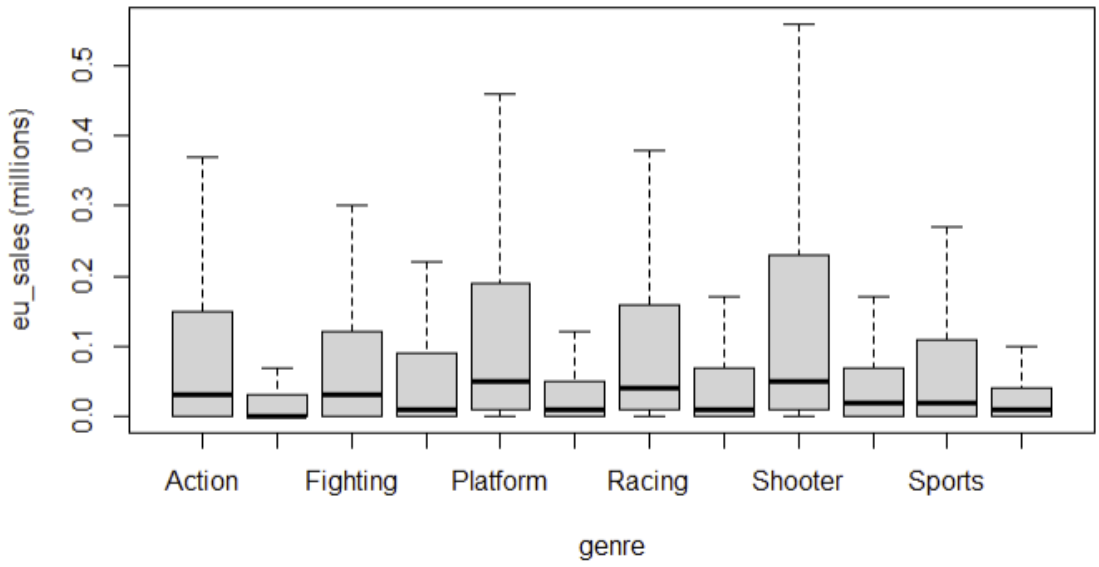

Figure 4: Boxplot of eu_sales(millions)/genre

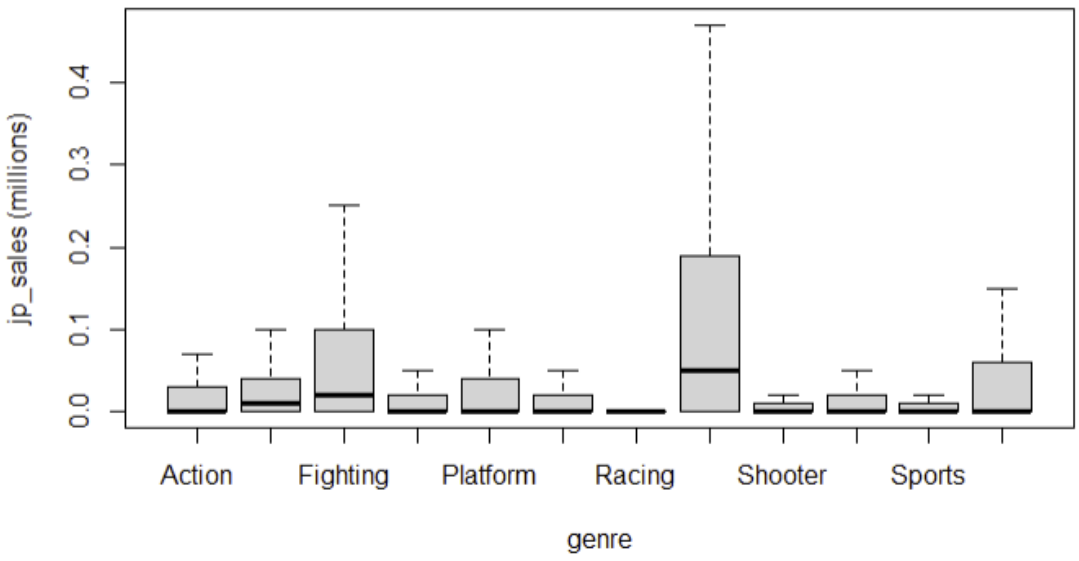

Figure 5: Boxplot of jp_sales(millions)/genre

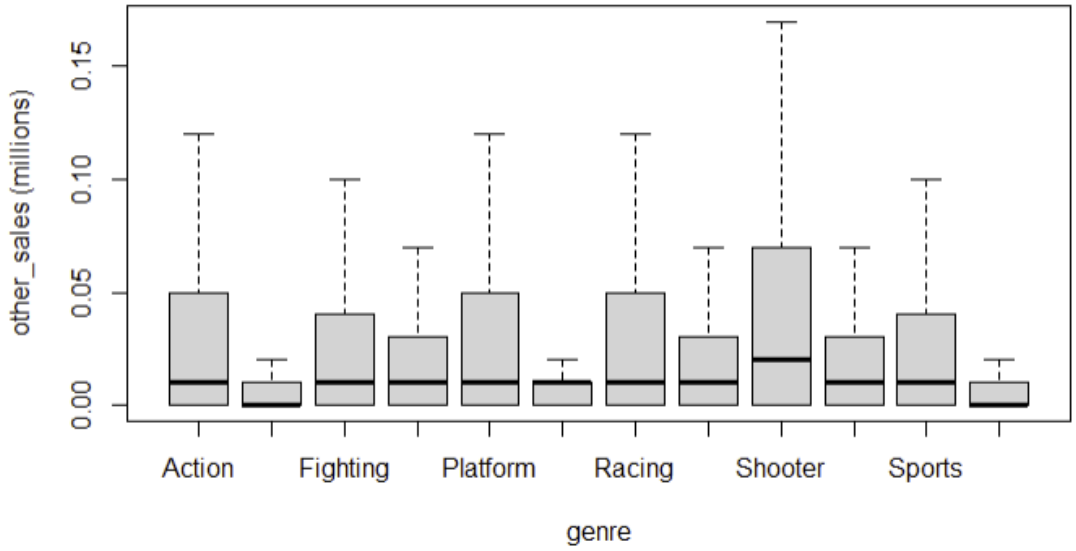

Figure 6: Boxplot of other_sales(millions)/genre

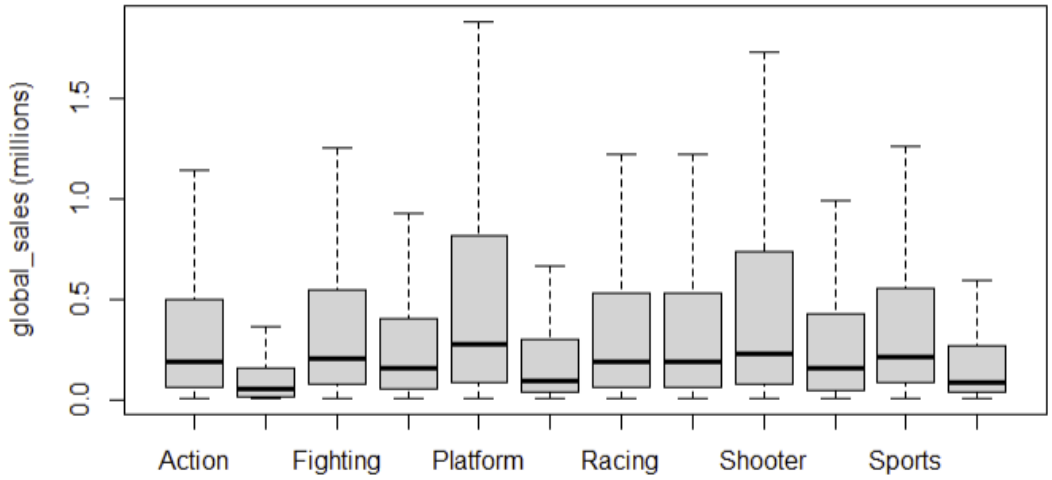

Figure 7: Boxplot of global_sales(millions)/genre

To get a better visualization of the data comparing genre and sales the outliers were omitted from figures 3-7, but the boxplots including the outliers will be available to view in the notebook. The boxplots and ANOVA tests showed that there is an association between genre and the categorical variables, with the ANOVA tests showing p-values less than 0 for each variable.

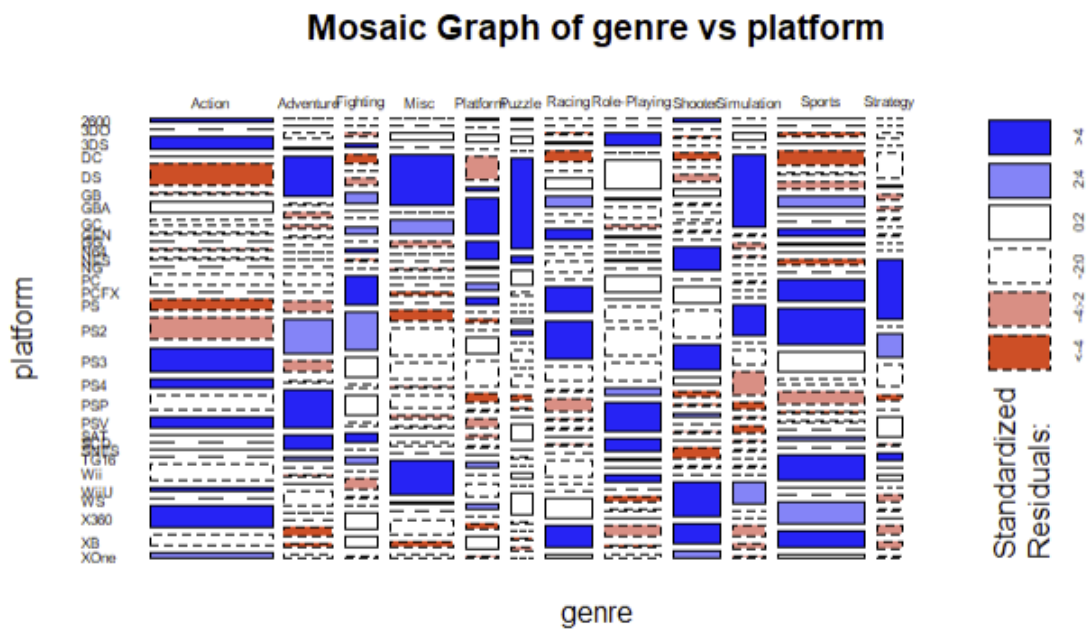

Figure 8: Mosaic plot of genre/platform

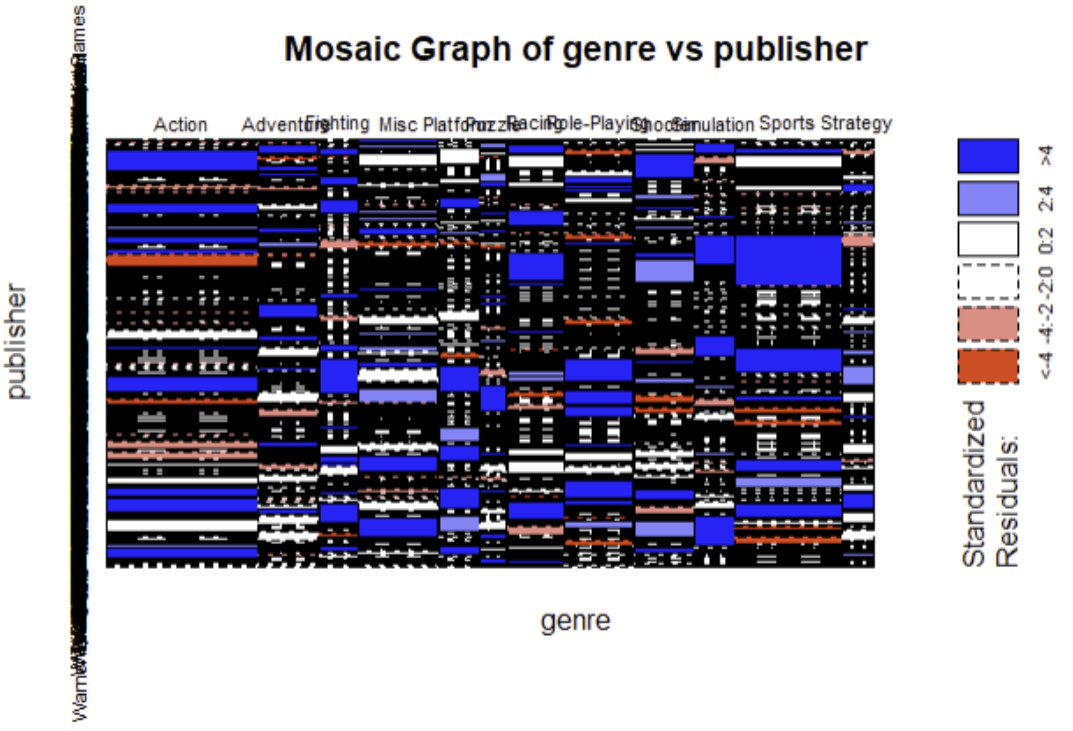

Figure 9: Mosaic plot of genre/publisher

For the categorical attributes of platform and publisher, mosaic plots and chi-square tests were performed to understand the relationship between genre. Platform had a p-value less than 0.5, and the mosaic plot shows that there is an association between platform and genre.

After data exploration, the next step would be to prepare the dataset for data analysis. In the next section, several models and algorithms will be used such as the Random Forest model, the Neural Network model, and the Gradient Boosted Tree model. The dataset would then be partitioned into training and test sets using the “createDataPartition” function from the “caret” package. The data did not need to be scaled and no other feature engineering was required.

Data Analysis and Experimental Results

For the data analysis, the models that were used were as follows:

- Random Forest
- Gradient Boosted
- Neural Network
- SVM
    - Linear Kernel
    - Radial Kernel

Accuracy and AUC were evaluated to compare the performances of each model.

Confusion Matrix and Statistics

	Reference											
Prediction	Action	Adventure	Fighting	Misc	Platform	Puzzle	Racing	Role-Playing	Shooter	Simulation	Sports	Strategy
Action	294	116	67	146	78	48	92	128	110	63	155	61
Adventure	0	0	0	0	0	0	0	0	0	0	0	0
Fighting	0	0	2	0	0	1	1	4	0	0	0	0
Misc	0	0	0	1	0	0	0	0	0	0	1	0
Platform	1	0	0	0	0	0	0	0	0	1	1	0
Puzzle	1	0	0	0	0	2	0	1	0	0	0	0
Racing	0	1	4	0	0	0	1	0	0	0	0	0
Role-Playing	0	0	0	0	0	0	0	0	0	0	0	0
Shooter	0	1	0	0	0	0	0	1	0	0	0	0
Simulation	1	0	0	0	0	0	0	1	0	2	0	0
Sports	28	8	10	24	9	6	27	12	18	19	73	6
Strategy	0	1	0	0	0	0	1	0	0	0	0	0

Overall Statistics

```
               Accuracy : 0.2302
                 95% CI : (0.21, 0.2514)
    No Information Rate : 0.1995
    P-Value [Acc > NIR] : 0.001263

                  Kappa : 0.0517
```

Figure 10: SVM Linear Confusion Matrix

```
Confusion Matrix and Statistics

              Reference
Prediction     Action Adventure Fighting Misc Platform Puzzle Racing Role-Playing Shooter Simulation Sports Strategy
  Action          286        94       79  158       83     48    114           134     119         76    218       57
  Adventure        39        33        4   13        4      9      8            13       9          9     12       10
  Fighting          0         0        0    0        0      0      0             0       0          0      0        0
  Misc              0         0        0    0        0      0      0             0       0          0      0        0
  Platform          0         0        0    0        0      0      0             0       0          0      0        0
  Puzzle            0         0        0    0        0      0      0             0       0          0      0        0
  Racing            0         0        0    0        0      0      0             0       0          0      0        0
  Role-Playing      0         0        0    0        0      0      0             0       0          0      0        0
  Shooter           0         0        0    0        0      0      0             0       0          0      0        0
  Simulation        0         0        0    0        0      0      0             0       0          0      0        0
  Sports            0         0        0    0        0      0      0             0       0          0      0        0
  Strategy          0         0        0    0        0      0      0             0       0          0      0        0

Overall Statistics

               Accuracy : 0.1958
                 95% CI : (0.1768, 0.2159)
    No Information Rate : 0.1995
    P-Value [Acc > NIR] : 0.6546

                  Kappa : 0.0104
```

Figure 11: SVM Radial Confusion Matrix

```
Confusion Matrix and Statistics

              Reference
Prediction     Action Adventure Fighting Misc Platform Puzzle Racing Role-Playing Shooter Simulation Sports Strategy
  Action          259        63       50   81       68     27     74            87      92         40    104       33
  Adventure        15        31        8   22        1      2      1            10       7          8      5        6
  Fighting          0         0        1    0        0      0      0             0       0          0      0        0
  Misc              5         8        1   27        2      9      3             4       4          8      5        3
  Platform          0         0        0    0        1      1      0             0       0          0      0        0
  Puzzle            0         0        0    0        0      0      0             0       0          0      0        0
  Racing            0         0        0    0        0      0      4             0       1          0      0        0
  Role-Playing      9         7        6    8        2      5      2            31       2          3      8        8
  Shooter           0         0        0    0        0      0      0             0       0          0      0        0
  Simulation        0         0        0    0        0      0      0             0       0          0      0        0
  Sports           34        15       17   32       12      9     37            11      20         23    106       12
  Strategy          3         3        0    1        1      4      1             4       2          3      2        5

Overall Statistics

               Accuracy : 0.2855
                 95% CI : (0.2636, 0.3081)
    No Information Rate : 0.1995
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.1443
```

Figure 12: Random Forest Confusion Matrix

Confusion Matrix and Statistics

Prediction	Reference: Action	Adventure	Fighting	Misc	Platform	Puzzle	Racing	Role-Playing	Shooter	Simulation	Sports	Strategy
Action	187	28	26	45	39	13	36	42	51	22	39	13
Adventure	13	40	4	20	0	2	2	4	5	7	4	4
Fighting	9	2	22	5	1	2	3	1	5	2	4	2
Misc	22	18	3	46	14	7	8	11	7	8	8	6
Platform	7	1	3	1	13	2	6	4	2	0	1	1
Puzzle	2	2	0	3	0	8	1	0	1	1	2	1
Racing	6	5	0	2	1	7	21	0	5	3	8	0
Role-Playing	14	7	5	11	1	4	4	64	2	2	6	6
Shooter	18	5	3	2	2	2	1	1	20	2	6	4
Simulation	10	2	0	8	0	2	2	3	2	19	5	6
Sports	34	14	17	26	15	4	36	11	26	15	140	10
Strategy	3	3	0	2	1	4	2	6	2	4	7	14

Overall Statistics

```
               Accuracy : 0.3646
                 95% CI : (0.3412, 0.3885)
    No Information Rate : 0.1995
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.2687
```

Figure 13: GBM Confusion Matrix

Confusion Matrix and Statistics

Prediction	Reference: 1	2	3	4	5	6	7	8	9	10	11	12
1	209	34	35	49	48	8	37	35	67	24	43	20
2	9	36	5	8	1	2	0	2	2	3	2	2
3	6	1	15	1	0	1	2	0	1	1	1	1
4	13	12	3	53	10	9	4	9	6	9	7	3
5	3	0	2	1	8	4	3	4	0	2	3	2
6	3	8	0	4	1	8	2	1	0	2	1	1
7	2	3	0	3	4	7	19	2	3	2	3	1
8	18	6	6	7	1	1	2	68	1	3	3	5
9	5	3	1	2	0	1	0	2	13	1	1	4
10	9	1	0	5	1	2	2	0	2	13	1	6
11	47	18	16	37	12	12	49	18	33	20	164	12
12	1	5	0	1	1	2	2	6	0	5	1	10

Overall Statistics

```
               Accuracy : 0.3781
                 95% CI : (0.3545, 0.4022)
    No Information Rate : 0.1995
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.2767
```

Figure 14: Neural Network Confusion Matrix (1- 12 : Action - Strategy)

Model	Accuracy	AUC
SVM Linear	23.0%	0.521
SVM Radial	19.6%	0.491
Random Forest	28.6%	.567
GBM	36.5%	.602
Neural Network	37.8%	.6379

Figure 15: Accuracy and AUC results

The dataset was randomized and split into testing and training data. Testing and training data were duplicated to be used for later models such as the neural network model. For SVM and the neural network, the dataset would have to be converted to numeric variables.
For the neural network model, categorical variables would need to be encoded using one-hot encoding. Hyper parameters for every model except for the neural network were tuned by caret, while the neural network model was tuned manually. 10-fold cross validation was used for every model other than the neural network model. The “tfruns” package was used to find the best tuned neural network model. From the confusion matrices, sports and action were misclassified the most. This could be due to how both genres are very similar, as you can have a sports game be action.

Conclusion

While the neural network model performed the best, overall accuracy was still low as it was below 50%. This could be attributed to how large the publisher attribute was with over 500 levels. The sales attributes could likely have been adjusted as the global sales totaled the same when North America, Europe, Japan, and other sale variables were combined.Each model took an extremely long time to finish, with SVM linear and radial taking upwards to 3-4 hours each.

Another suggestion could be to remove genres that might be similar to increase accuracy, like how Action and Sports were misclassified the most. Using embedding vectors may be useful as the publisher attribute due to having over 500 levels and could have been a factor in having too many dimensions from one hot encoding. For further research, it could be useful to include genres that are a combination of other genres as games can be more than one genre like a “Role-playing Action-Adventure”.

References

1. *Multiclass.roc: Multi-class auc*. RDocumentation. (n.d.). Retrieved May 4, 2023, from https://www.rdocumentation.org/packages/pROC/versions/1.18.0/topics/multiclass.roc
2. nihar14. (2023, April 5). Analysis on video games sales. Kaggle. Retrieved May 5, 2023, from https://www.kaggle.com/code/nihar14/analysis-on-video-games-sales
3. Alfaradikrisnaocsyta. (2023, April 5). *Video game sales analysis*. Kaggle. Retrieved May 5, 2023, from https://www.kaggle.com/code/alfaradikrisnaocsyta/video-game-sales-analysis
4. Jiang, Y., & Zheng, L. (2020, November 21). *Deep learning for video game genre classification*. arXiv.org. Retrieved May 5, 2023, from https://arxiv.org/abs/2011.12143
5. *To_categorical: Converts a class vector (integers) to binary class matrix.* RDocumentation. (n.d.). Retrieved May 6, 2023, from https://www.rdocumentation.org/packages/kerasR/versions/0.8.1/topics/to_categorical