Forecasting Video Game Sales

with Data Mining

Rez Ebrahimi1, Yuening He2, Ava Shearer3

Department of Computer Science, Georgia State University

Atlanta, Georgia 30302 USA

1rebrahimi1@student.gsu.edu

2yhe12@student.gsu.edu

3ashearer5@student.gsu.edu

***Abstract* — Growing numbers of video game players supports the likelihood of high potential growth in the gaming industry. This paper studies multiple factors that contribute to the sales of video games. The dataset was generated by scraping data from vgchartz.com. A prediction model, which predicts the sales of future video games, will be created based on the analysis of this dataset. The algorithms used in this project include multiple linear regression, random forest, and k-means clustering.**

*Index Terms —* data mining, k-means clustering, multiple linear regression, random forest, video games, sales, prediction, features

INTRODUCTION

Aside from employing data mining techniques learned in the Fall 2020 Data Mining course, the aim of this project is to get a better, hands-on understanding of how algorithms and data mining is applied to real-life scenarios. After generating the dataset by scraping data from **vgchartz.com,** the exploration of data could begin and allow our group to uncover hidden patterns and predict outcomes of future video game sales with multiple data mining algorithms. To achieve this, we looked at simple metrics like number of games sold, number of games released in a year, average sales, and a timeline of how sales have changed within the dataset’s given time period. Then, using data mining techniques and algorithms such as random forests, the insights we gain will help us determine future sales. Not only is this topic relevant for our project, but it could realistically be used and applied to similar real-world problems. Video games are not going anywhere, and only continue to grow in popularity each year. What we learn from this project will be useful for years to come.

RELATED WORK

Similar projects to ours would be very feasible, as the algorithms used are not very complicated and the data was easily available and accessible, and applicable to an interesting question. The exact dataset we used may be difficult to replicate, but the overall concept of the project could also be useful in real life.

One similar example we saw to our project was a video game forecasting project on Kaggle. Minor differences were observed, like some features in that project were not in our and vise versa. But they had a similar objective in that they wanted to predict the future of video game sales. They also employed linear regression and a random forest algorithm to their project, which yielded similar results to ours. But, they also used many other algorithms such as lasso, ridge regression, support vector regression, and gradient boosting regressor.

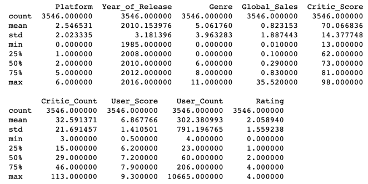
So, while we observed a very similar project on Kaggle, ours still had more in-depth insight into questions such as which games have the highest ratings, and how that affects global sales. We also had great visualization and analysis from the K-means algorithm, which the other project did not have. The other project seemed to run the algorithms without a very in-depth analysis afterwards, which sets our project apart.

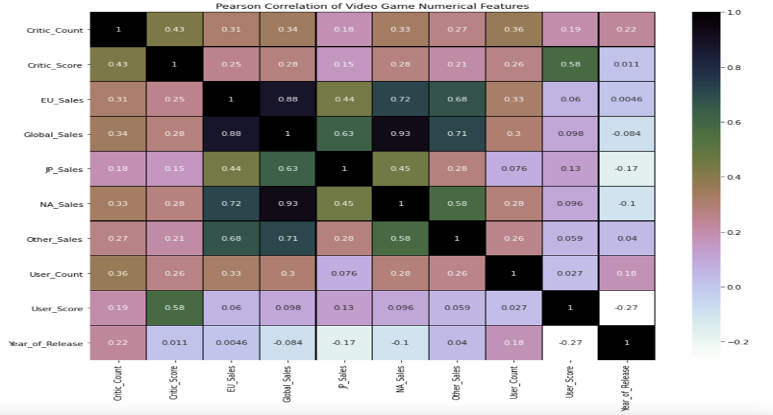
METHODS

The size of our dataset is 16719 rows and 16 columns. Its categorical features include Platform, Genre, Publisher, Developer, and Rating. Its numeric features include Year\_of\_Release, NA\_Sales, JP\_Sales, Other\_Sales, Global\_Sales, Critic\_Score, Critic\_Count, User\_Score, and User\_Count. Before we continued with other preprocessing methods, we checked for obvious outliers and removed one outlier from the dataset.

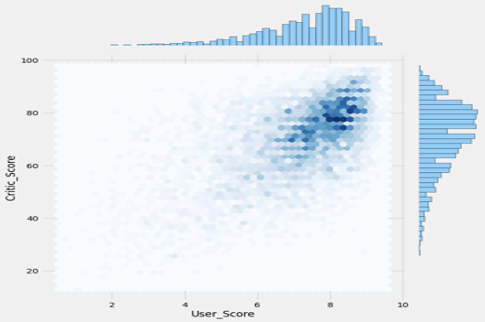
First, missing values were addressed in the dataset. We believed Critic\_Score may be the most important independent variable, but 52.5% of its values are missing. We then checked missing ratios for Critic\_Score across multiple major platforms, including X360, PS3, PC, Wii, etc. With 39.5% of values missing, we decided to drop all rows with missing Critic\_Score to get the best performance of the dataset. Missing values for other features became relatively small as well.

After removing rows with missing Critic\_Score, the size of our dataset was significantly reduced and became 5673 rows and 16 columns. We imputed the missing values for Publisher, Developer and Rating columns with mode. As for Year\_of\_Release, User\_Score, and User\_Count, we replaced their missing values with median. After considering our dataset and the distribution of the columns, we accordingly decided to impute with mode and median, since choosing to impute with the wrong metric can be bad for the model performance. The tables below show the visualizations we created of the descriptive statistics for continuous variables. There is a table showing the calculated mean, median, mode, and percentiles of data, and the correlation matrix, respectively.

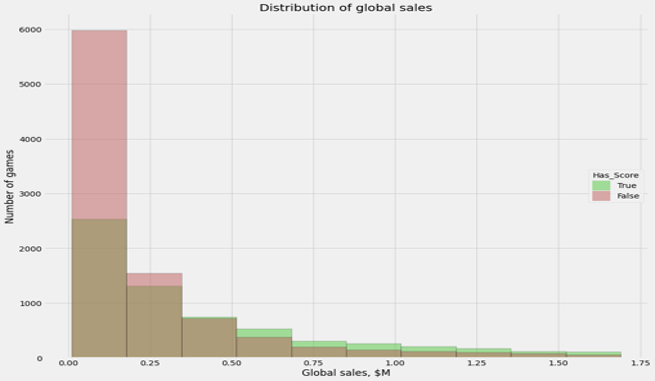




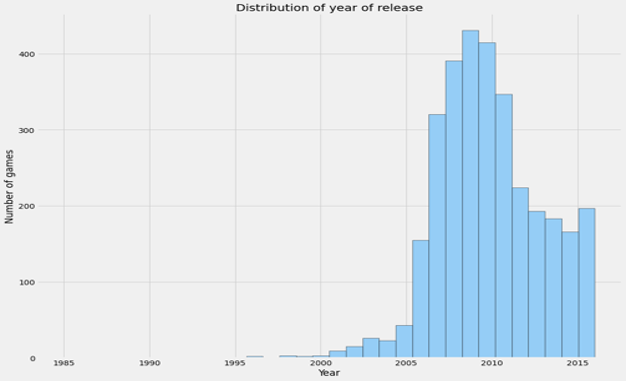
In terms of autocorrelation, the expected correlations of related variables in the correlation heat map are not very high. Only the sales figures are highly correlated with each other. Therefore, we can look at intercepts of regression models without worrying too much if we wish to do so later on. The high correlation between sales figures may explain the global nature of the video game industry: success on one continent usually means success on another continent. This provides greater confidence in using only Global\_Sales as the dependent variable.



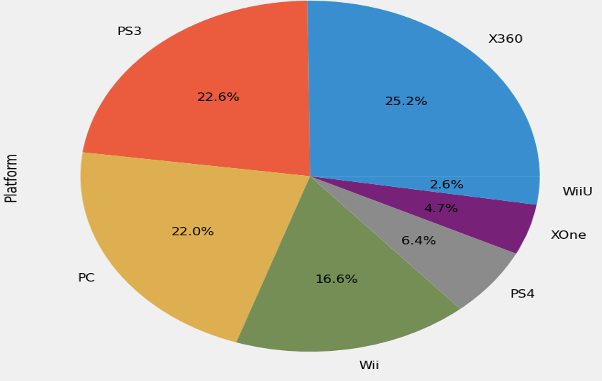
From the illustration above, we can conclude Critic\_Score and User\_Score for most of the games is greater than 70 and 7 respectively. This means most of the games in the dataset are reasonably popular and played by consumers, as there is a market for them since they have decent scores. This analysis alone may seem insignificant, but is more useful in our later conclusions.



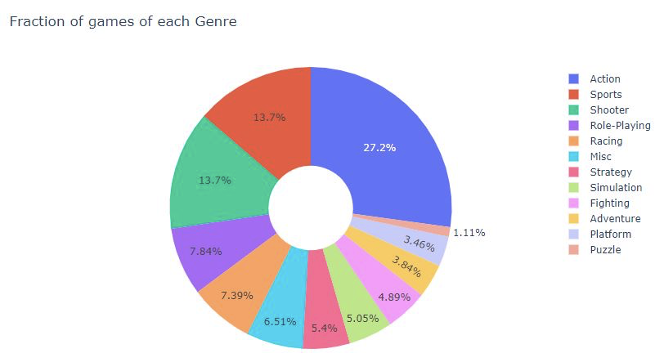
The stacked histogram above demonstrates how global sales was distributed among games with scores and games without scores. It is obvious that games with scores have a much higher quantity of global sales. Because of this, it is evident that customers are less likely to purchase a video game if they could not find scores or ratings for a particular game, or opt for purchasing a game that has ratings.



Based on the distribution of Year\_Of\_Release, most of the video games from the dataset were released between 2007 and 2011. PC’s, laptops, and other gaming consoles started gaining popularity around the 2007 mark, and it started to become common to see nice gaming consoles (like Wii, Xbox, PlayStation) in homes. Because of this, there was a boom in demand for games and developers started launching many games to keep up with the demand.



The pie graph above shows the distribution of video game sales across different platforms. The most popular platforms were X360, PS3, PC, Wii, PS4, XOne, and WiiU.



We used another pie graph to illustrate the distribution of video game sales across genres. Most customers chose to purchase an action video game. Sports and Shooter games were people’s second favorite.

RESULTS AND DISCUSSION

For this project, Data Mining algorithms we used are Random Forest (Categorical), Multiple Linear Regression (Numeric), and K-Means Clustering (Numeric).

1. Random Forest Classification

Our motivation for this algorithm was that since we wanted to work with Categorical variables and use the Classifier method, we decided to use Random Forest Classification to find out which genre of games were sold the most based on our features. Based on our assumption on data visualization, we assumed that Action should have the

highest global sales among other genres. So we used the Random Forest algorithm to prove our assumption.

First, we tried to take care of Data Preprocessing and choose our independent and dependent variables. Since we did not want to have too many categorical variables in order to have a better accuracy, we used LabelEncoder() to convert ‘Platform’ and ‘Rating’ columns from Categorical to Numerical. Next, we dropped the columns that we don’t really need to use because they have the least effect on our model.

Independent variables:

cols = ['Platform', 'Year\_of\_Release', 'Global\_Sales', 'Critic\_Score', 'Critic\_Count', 'User\_Score', 'User\_Count', 'Rating']

X = ds[list(cols)].values

Dependent variable:

Y = ds['Genre'].values

So our Dependent is ‘Genre’ since this is what we are looking for to analyze.

Next, we split our data to train and test and the shape of our train and test data is:

X\_train shape: (2659, 8)

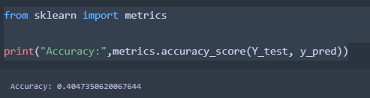
Y\_train shape: (2659,)

X\_test shape: (887, 8)

Y\_test shape: (887,)

After that, the most important process before we go to Random Forest is to standardize our data which helps our model to have better accuracy and prediction. When we standardized our data, we Used RandomForestClassifier and fit our train data in it, and predicted X\_test.

Finally, we were able to get the accuracy score of our model based on test data, as shown below:



Our Accuracy Score is 0.40 which is not a good score for our model.

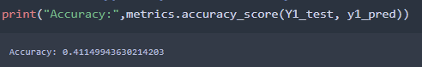
At the end, we predicted our model based on the Array number to see which genre has the highest Global Sales, and it is the Action genre, which proves our assumption.



In order to make our accuracy model better, we dropped ‘Year\_of\_Release’ from our features to see if we can get a better accuracy score.

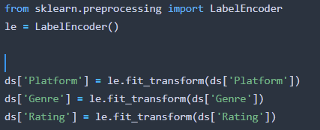
After dropping a feature, we got the accuracy of 0.41, which did not improve our model at all.

In conclusion, Random forest did not give us a good accuracy score for our model, however, we were able to find out that the Action genre has the highest global sales.



1. K-Means Clustering

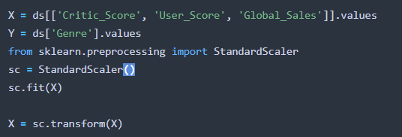
For K-means, we took care of Data Preprocessing to make sure everything is good to work with. We wanted to have all numeric variables so we used LabelEncoder () to convert our Categorical variables to Numeric.



After that, we dropped the columns that are not important for our model, just like what we did for Random Forest.

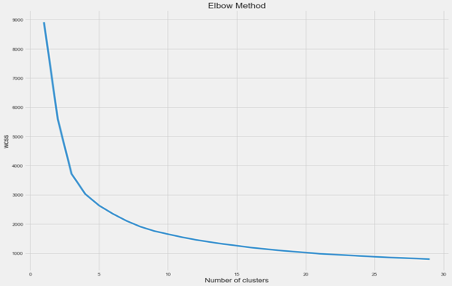
ds = ds.drop(['Name', 'Year\_of\_Release', 'Publisher', 'Developer', 'NA\_Sales', 'EU\_Sales', 'JP\_Sales', 'Other\_Sales'], axis=1)

Then we picked our features for our model and standard them:



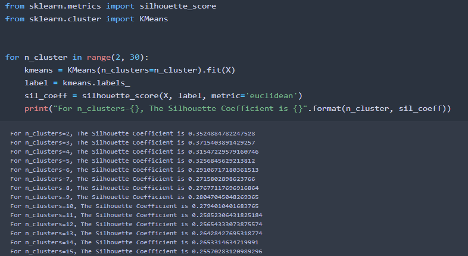
We used Critic\_Score, User\_Score, and Global\_Sales as features, since they are the most important columns in our dataset.

Next, we used elbow method to find the right number of clusters:

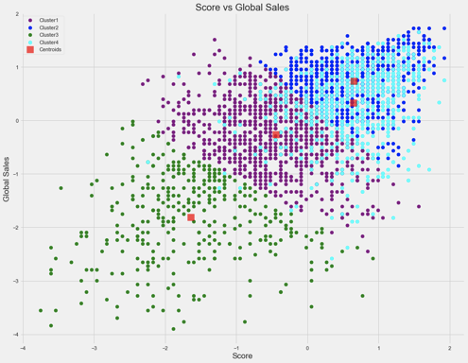


The Elbow method tells us that 4 is a good value for a number of clusters.

Next, we calculated Silhouette Coefficient to see how well our clusters are separated:



After plotting our model, K-means clusters are shown as below:



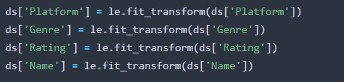
In our Data Visualization part, we assumed that games that have the highest score, will have the highest global sales, and we wanted to prove our assumption with K-means Clustering.

As you can see, when Clusters have higher scores, it means they have higher global sales as well. So there is a positive relationship between Score and Global Sales, Clusters 2 and 4 have the highest score, hence the highest sales.

1. Multiple Linear Regression

Since we wanted to have all numeric variables to work with, we picked a Linear Regression model to perform on our model. Like K-means Clustering, we wanted to prove our assumption that games that have the highest score will have highest global sales as well. At the beginning, we thought that it would be hard to visualize linear regression since we were using multiple dimensions. So, what we did was to use Linear regression for a single dimension and then add more features to make it multiple regression and compare our results to see which one gives a better accuracy for our model.

First, we took care of Data Preprocessing and converted Categorical values to Numerical:



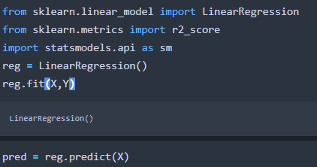
Then dropped columns that we do not need.

ds = ds.drop(['Year\_of\_Release', 'Publisher', 'Developer', 'NA\_Sales', 'EU\_Sales', 'JP\_Sales', 'Other\_Sales'], axis=1)

Next, we picked our independent and dependent:



Then we used LinearRegression() to fit our independent and dependent in it and predicted X:

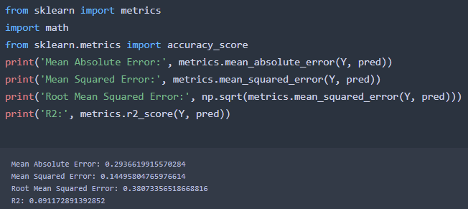


Next, we got our Linear model:



Our Coefficient of Critic Score is 0.008533 which tells us the mean increase of Global sales for every additional score in Critic Score.

Next, we calculated MAE, MSE, RMSE, R2 scores to see how good our model is:



RMSE is a key value and it is 0.38 which is not bad, and R2 is 0.091 which is not good, and it tells us our model is not good.

Next step is to add more features to our model and make it a multiple linear regression and see if our model improves, based on RMSE and R^2 scores.

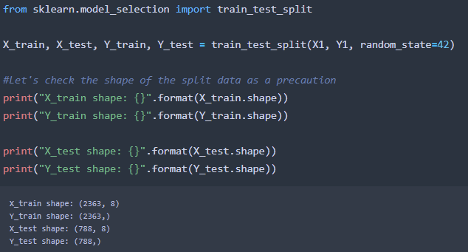
We added more features to our independent:

X1 = ds[['Critic\_Score', 'User\_Score', 'Critic\_Count', 'User\_Count', 'Rating', 'Genre', 'Name', 'Platform']].values

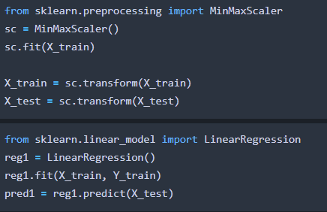
And our dependent is:

Y1 = ds['Global\_Sales'].values

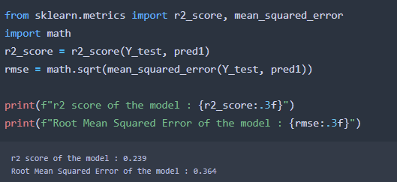
After that, we split data to train and test:



Then we used MixMaxScaler() instead of StandardScaler, and transformed X\_train and Y\_train. After that, we fit our train to LinearRegression and predicted X\_test.

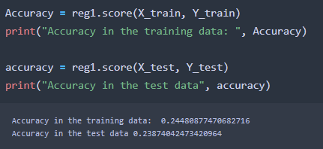


Finally, we calculate our R2 and RMSE:



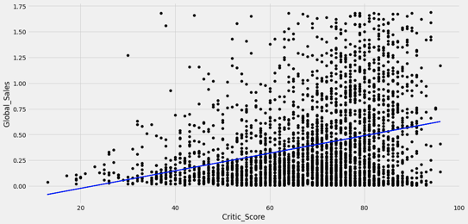
As you can see, by adding more features our model improved based on R2 and RMSE values, Since R2 value increased and RMSE value decreased compared to the first model.

Now let's try to get accuracy score for train and test data separately:



Trained data has better accuracy than the test data. In Conclusion, the Linear Regression model did not give us a good accuracy even though we tried to improve, it is still not good for our model.

But our Assumption was right, as we assumed that higher critic score for a game is equal to higher global sales, as you can see our Linear model:



CONCLUSION

At the beginning where we did Data Visualization, we realized that Games that don’t have score would not have any global sales or have less global sales compared to games that have score. Games that have a score will have global sales, but which games have the highest global sales? Well the games that have higher scores will have higher global sales, where there is a relationship between score and global sales, as the score increases, global sales increases too. So, it is very important for a game to have a score in order to be sold more. We performed Linear Regression and K-means Clustering to prove our point that when a score of a game/genre increases, its global sales increases as well, where these two algorithms proved our point.

Based on Data Visualization, we wanted to know which Genre has the most score, which means the most global sales. We realized that the Action genre has the highest score, which makes it to be the most sold genre globally. We performed Random Forest Classification to prove our point and assumption that Action has the highest global sales based on scores, and Random Forest proved our assumption that Action has the highest among all the genres. Between these algorithms that we used; Random forest has the highest accuracy which makes it a good algorithm to use for our model. From all the Data Visualization and Algorithms results, we can predict that Action games is the future of video games since they were sold the most globally, and for any games that want to compete to be sold more than other games, they should have highest scores in order to be sold more.