Report On Evaluation of Steam Game Performance

Carolyn Zhang, Cherie Zhang, Lukas Fullner, Diya Lakhani, Archit Pimple

Abstract

This report presents a comprehensive analysis of factors influencing game performance on Steam, which offers valuable insights for developers, publishers and marketers. Leveraging a rich dataset encompassing game attributes, user preferences, and market trends. This study has explored the intricate relationship between various factors and successful games. Through exploratory data analysis techniques such as heat map visualization and scatter plots, and also the statistical methodologies including linear regression, and logistic regression classifiers, this study uncovers nuanced insights into the dynamics of the gaming industry. The findings shed light on the impact of the game attributes such as pricing, difficulty, user reviews, and genre preferences on market performance, providing actionable insights for strategic decision making. This report will serve as a valuable resource for stakeholders in the gaming field, offering a deeper understanding of the factors driving game success on Stam and informing future development and marketing strategies.

Introduction

This project aims to dissect the various elements that contribute to a game's popularity and performance on Steam, providing a foundation for strategies that enhance game development and marketing efforts. By leveraging a comprehensive dataset that encapsulates game details, user preferences, and market trends, we intend to shed light on the intricate relationship between game features and their market success.

The digital gaming industry, and in particular game products and gaming communities on platforms such as Steam, are experiencing exponential growth and diversity. In this rapidly evolving environment, understanding the factors that drive the popularity and commercial success of games is increasingly important for developers, publishers, and marketers.Our main goal is to analyze and explain how various characteristics of a game (from genre and pricing to user reviews and playtime) affect its success on Steam. By delving into these factors, we aim to provide actionable insights that can guide future game development and marketing strategies.

The significance of this research is not only to improve the profitability and marketability of games, but also to contribute to a deeper understanding of consumer preferences and trends in the gaming world. Our research methodology aims to combine comprehensive dataset analysis with advanced analytics to decipher the secrets of success on the world's largest digital distribution platform for PC games.

Description of the Data

In the early 2000s video game development company Valve had an issue. As their catalog of games grew, they required a system to publish games and updates for those games. This led to the creation of the Steam platform in 2003, which was originally made to publish Valve's various titles, as well as push updates for those games. But these were more than just games, and each title Valve published had a large community of passionate fans surrounding it, including a vibrant modding community. *Team Fortress Classic* for example originated as a mod for Valves *Quake* before eventually being acquired by Valve and turned into a full fledged game. Valve saw the potential of mods and how they could improve their games, and so embraced that part of the community, making many of their tools including the Source Engine publicly available. Back to Steam, this embrace of the modding community led to Steam hosting mods, and eventually full games from third party developers. In the following twenty years Steam has grown from the publishing platform for Valve to one of the most used game platforms in gaming history!

Steam now hosts over 85 thousand games from many different developers. It acts as a convenient store for these games, allowing users to buy and install games while leaving feedback and reviews for the developers and other prospective buyers. And because of Valve's openness with its community we are able to access this data. Two separate Kaggle users have recently pulled this data from Steam, and gathered it in datasets which we will be using in our project.

The base dataset we used originates from Martin Bustos in his Kaggle dataset: https://www.kaggle.com/datasets/fronkongames/steam-games-dataset/data. It contains a large quantity of data for each game, and is the base dataset we used joining extra features to. And those extra features come from Sujay Kapadnis' Kaggle dataset: https://www.kaggle.com/datasets/sujaykapadnis/games-on-steam. Though both datasets come from Steam, the amount of data contained on each game means these two different datasets included different features of the games, so we pulled both then joined them by the common AppID, a unique identifier assigned to each Steam game. Martin Bustos' dataset contains 39 columns, and Sujay Kapadnis' dataset contains 46 rows. Many of these are shared or will not be used, so as part of our project we dropped many of the columns. The dataset we began analyzing contains 22 columns, with each row representing a game. Some of the columns relevant to our analysis are mentioned below:

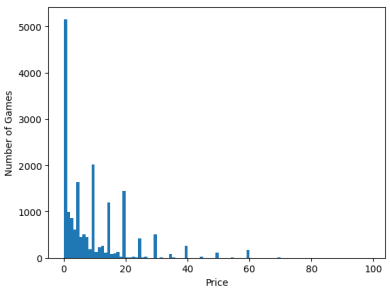
AppID	A unique ID assigned to each game
Name	The name of the game
Release date	The date the game was released
Estimated owners	A string containing the estimated number of owners as the exact values are not known (e.g. '0 - 20000')
Peak CCU	The maximum number of players playing the game at the same time.
Price	The price in USD the game sells for
Supported languages	A list of languages the game is localized for
Reviews	A string containing text reviews from users, this will be used for sentiment analysis
Windows	A boolean describing if the game is playable on a computer running Windows OS
Positive	The number of positive reviews a game got. Steam stores reviews as either positive or negative
Negative	The number of negative reviews a game got.
Median playtime forever	The median amount of time each user spent playing the game
Categories	A list of gameplay categories defined by the developer, e.g. singleplayer, multiplayer
Genres	A list of game genres defined by the developer, e.g. action, adventure
igdb_uscore	A game score from the game review site IGDB

Prior Work

There is one prior project on our dataset, Terenci Claramunt's Steam Games tag visualization: https://www.kaggle.com/code/terencicp/steam-games-data-transformation. This visualization is quite interesting to use as it provides a clean visualization for the large volume of data contained on Steam. It however does little analysis on the data, and so our project is the first to perform a deeper analysis on this data.

Exploratory Data Analysis

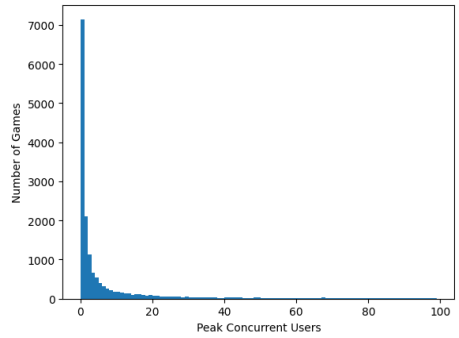
Because of the sheer volume of data we wanted to spend some time on Exploratory Data Analysis in order to get a better understanding of the relationship between various features in the data. In this section we will show some of the early exploratory graphs, and what they mean in the context of the data.



Price Histogram
This chart shows the distribution of games according to their price, shown in a histogram with bin width equal to 1 USD. The large spike at 0 shows the volume of free games, leading us to split our data in future

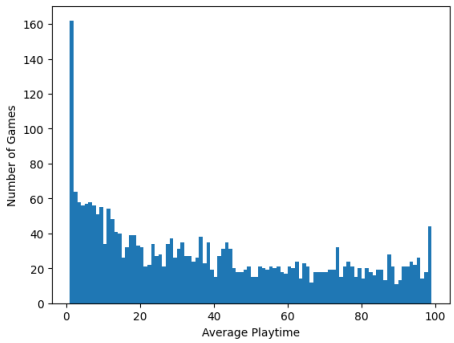
analysis by free and paid games. We also see spikes every five dollars, which aligns to societal standards of pricing goods on round numbers.

Peak Concurrent Users Histogram



This chart shows the distribution of peak concurrent users, and leads us to one of our insights about the data: a lot of games receive very little attention. We see that most games on Steam receive almost no attention whatsoever, so we needed to account for this in our analysis, where a large section of our data contains very little information. This manifested in tables that while not being filled with null values, were filled with a lot of zeros.

Average Playtime Histogram

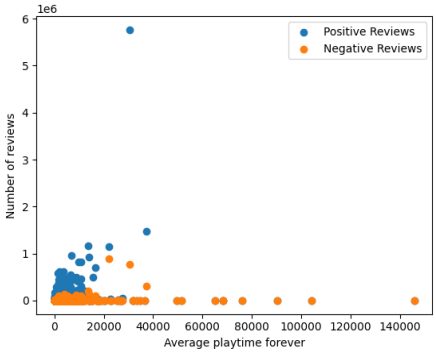


This histogram leads to a more surprising result. While the Peak CCU graph shows most games getting little peak users, the average playtime graph is more distributed from 0 to 100 minutes. This could be due to players trying out a larger volume of game for various times, so while the peak number of users on a game may be low the users that do try it still give it time.

Review Scatter Plot

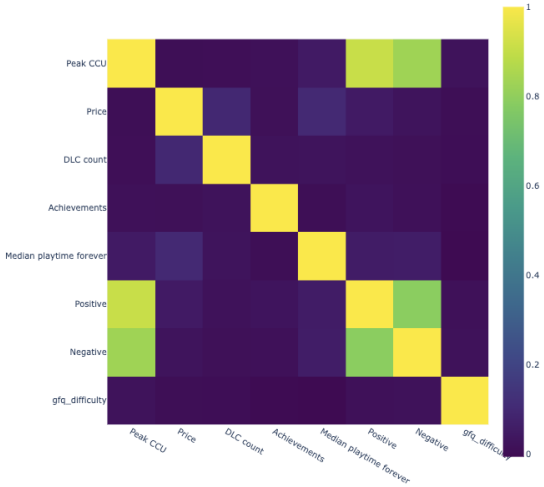
In order to assess game performance we wanted to look at the number of positive and negative reviews. There are varying reasons for why someone would leave various reviews, so looking at factors such as playtime can give us insights into underlying factors in the data. Here we see clusters of both positive and negative reviews. We can also see a few outliers, which reinforces the idea that only a few games get a lot of traction. Note the outliers along the bottom are both positive and negative, meaning games with lots of playtime have a similar positive and negative review count.

In our comprehensive project, we ventured into the intricate world of video game success analysis with a clear objective: to discern the defining features that contribute to a game's popularity and market success. This ambitious endeavor began with a meticulous process of data cleaning and feature selection, wherein we carefully sifted through a plethora of data points to focus on those we deemed most influential. These included Peak Concurrent Users (CCU), Price, DLC (Downloadable Content) count, Achievements, Median playtime forever, Positive ratings, Negative ratings, and the categorically unique feature, gfq_difficulty. In our pursuit of a refined dataset, we opted to eliminate features that offered redundant information, such as multiple ratings from different platforms and various metrics of playtime, which, while informative, overlapped significantly in the insights they provided.



Our methodology was grounded in rigorous data preprocessing techniques. Since each game might have several ‘genres’, we have used Multilabelbinarizer to create a new DataFrame where each genre is a separate column, with binary values indicating the

presence of that genre in each game. Recognizing the categorical nature of the gfq_difficulty variable, we embarked on encoding this feature into ordinal integers, a step that would later prove pivotal in our analytical models. The vast range of differences among our chosen features necessitated the normalization of the continuous quantitative variables, ensuring uniformity and comparability across our dataset.



The exploration phase brought us to employ heatmap analysis (shown aside), a visual tool that unveiled the high correlation between Positive and Negative ratings and the Peak CCU, shedding light on the interconnectedness of these variables. This discovery was instrumental in guiding the subsequent stages of our analysis. To quantify the elusive concept of game success, we introduced the success_ratio, a calculated metric derived from the proportion of Positive ratings to the total of Positive and Negative ratings. This continuous quantitative measure ranged from 0 to 1 and served as a cornerstone for our classification efforts.

We have also utilized the Variance Inflation Factor (VIF) as a tool to detect and measure the extent of multicollinearity among the predictor variables. By evaluating the VIF scores, we are provided with valuable insights that guide us in determining which features should be kept, adjusted, or excluded from the model. This analysis has been applied to both our logistic regression and multinomial logistic regression models. The VIF scores obtained for these models are as follows:

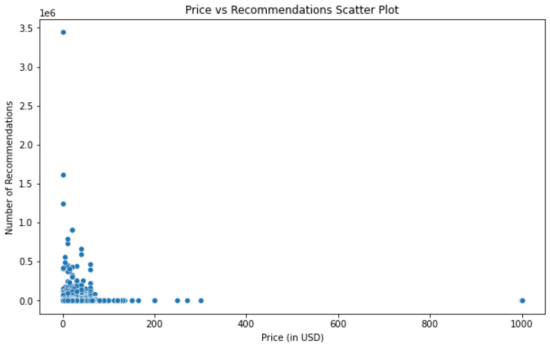
- VIF: Peak CCU: 1.003
- VIF: Price: 1.020
- VIF: DLC count: 1.011
- VIF: Achievements: 1.001
- VIF: Median playtime forever: 1.012
- VIF: gfq_difficulty: 1.000

With these VIF scores approximately equal to 1, it signals almost no multicollinearity between the independent variables, indicating that each variable is largely independent of the others in the models.

Analysis

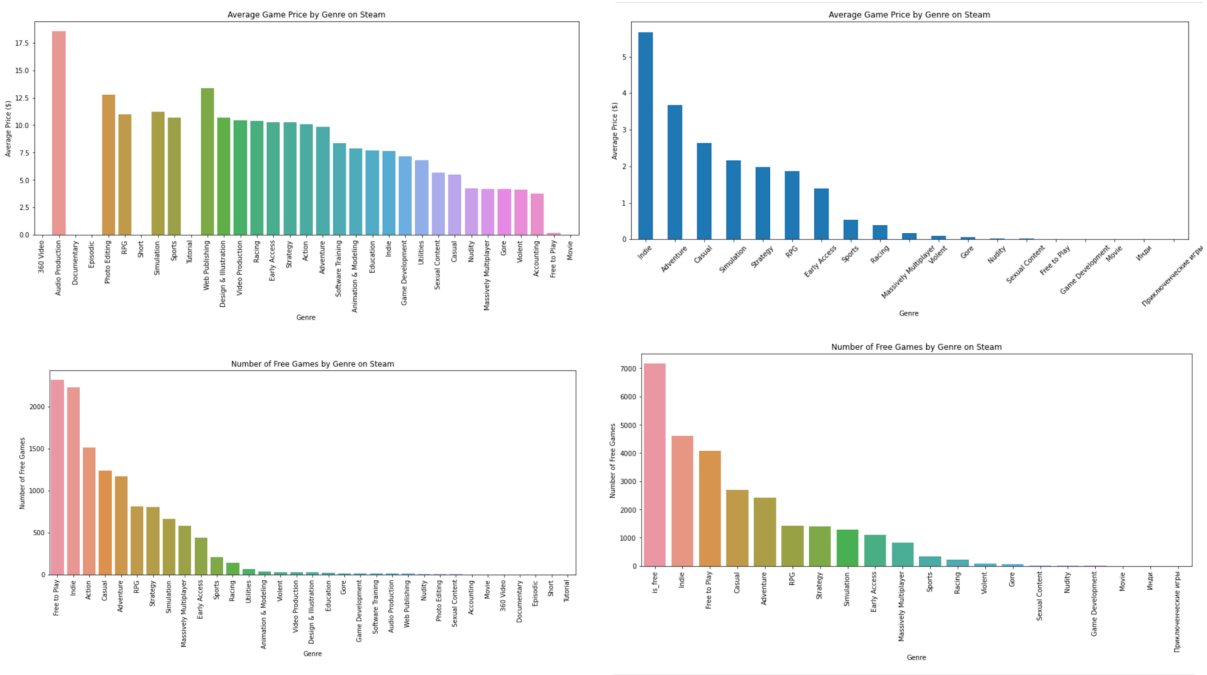
Relationships between ‘price’ and ‘recommendations’ analysis:

In our project, we first tried to find relationships between ‘price’ and ‘recommendations’ to see if this tells us anything about the success of a game. By calculating the coefficient of these two features we got 0.0430546313069667, which is approximately zero. The graph here showcase the relationship between ‘price’ and ‘recommendations’:



However, by getting the correlation coefficient and graph from these two features, it indicates a very weak positive linear relationship between the two variables. In practical terms, it suggests that there isn't a strong or significant linear association between the price of a game on Steam and the number of recommendations it receives.

We also tried to find the relationship between ‘price’ and ‘genres’. What we have done here is getting the average price for each genres and see what type of genres has the highest average price; and also the ‘free game’ vs. ‘genres’:

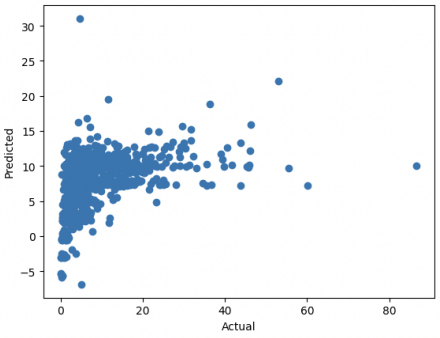


However, all of these graphs didn’t give us a strong relationship and results on what can cause a game success.As a result of this, we turned to find the relationship between ‘genres’ and other features to see what might affect the success of a game.

Linear Regression:

To begin our analysis, we chose to build a multivariate regression model to observe if the ratio to the positive to the negative reviews can be predicted through other quantitative features in our dataset. By predicting the ratio of the positive to the negative reviews, we essentially aimed to analyze if certain features of the game can impact the way it is received by the users. For our regression model, we chose to use the number of languages, price, Peak CCU (the maximum number of players the game received), and the igdb_score (a third party user determined score on a scale of 1 to 100).

The result of the regression model on the testing data is displayed on the left. It is clear that our model does not capture the patterns in the data since ideally we would want to observe a 45 degree line for the actual v/s predicted plot. The R2 was value, which indicates the quality of the linear fit was roughly 0.2, suggesting a poor linear fit. In addition, the RMSE was around 6.5 which is quite high. To understand the skewed distribution, we delved into the regression diagnostic tests.

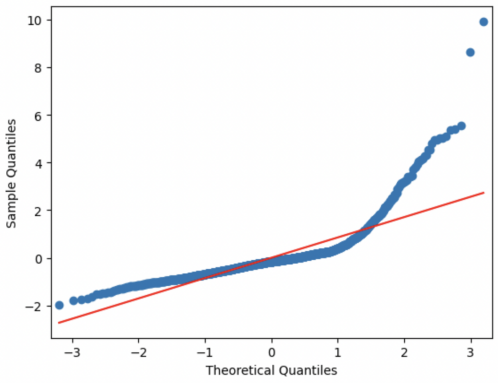


Regression Diagnostics:

Before we utilize our regression model for hypothesis testing, it is crucial that we test the validity of our model. We do so by performing the linear regression diagnostic test. Linear regression models depend on specific assumptions, namely the relationship between the

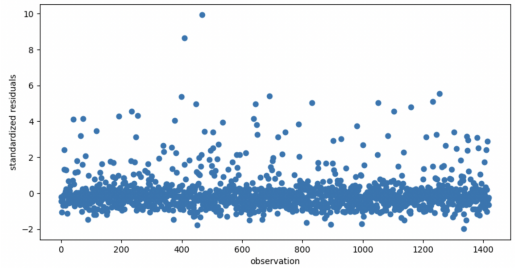
dependent and independent variables is linear, the residuals are normally distributed and are independent.

We first check to see if the residuals are normally distributed. We conduct this analysis by constructing a QQ-plot. The following plot was obtained:



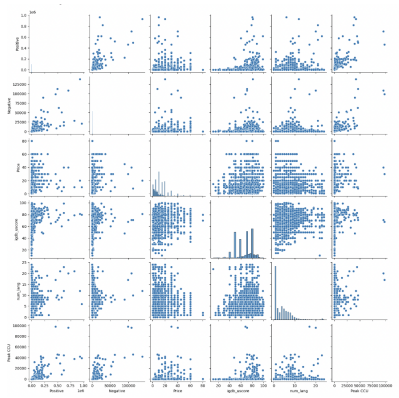
Ideally we would like to observe the points arranged in a straight line across the 45 degree line, however, there is an evident skew in the points. This suggests that the residuals are not normally distributed and the plot can be further validated by performing the Shapiro-Wilk test. We obtained a p-value of 3.1903057621861278e-43, which is extremely low, and hence, we can reject the null hypothesis. The null hypothesis is that the residuals are normally distributed.

The second diagnostic test we performed involved checking the independence of the residuals. For this we plot a scatter plot of the standardized residuals against the observations. The following plot was obtained:



Ideally we would have liked to observe a scatter plot with no evident pattern, however, it is quite evident that the residuals are clustered as a horizontal line across the x-axis. In conclusion, we have proven that our regression model is not suitable for hypothesis testing since it is not a valid model.

These skewed distributions led us back to data exploration and we produced the following pairplot.



The pairplot involves individual scatter plots of the predictor and response variables. Positive and negative are the response variables and essentially we observe the lack of a linear relationship between our dependent and independent variables. In addition, certain features seem to appear more discrete than continuous, like the number of languages and price, which could also lead to skewed results. After trying a couple of other regression models, however not being able to obtain a low RMSE and higher R2 value, we transitioned to conducting analysis involving more categorical variables.

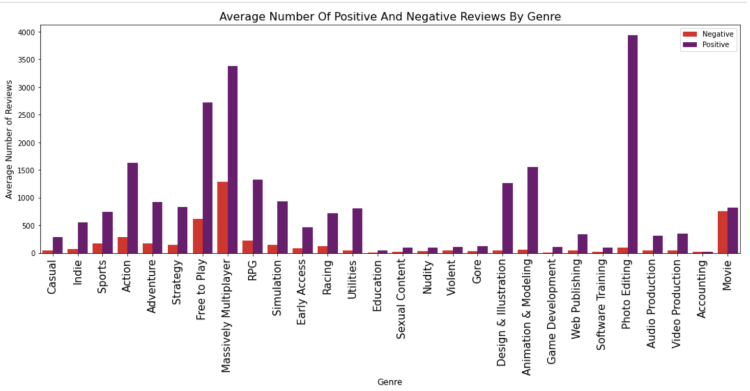
Relationship between ‘genres’ and other features analysis:

Firstly, we used Multilabelbinarizer for one hot-encoding the ‘genres’, which gives us the binary values and better indicate the presence of that genre for each game:

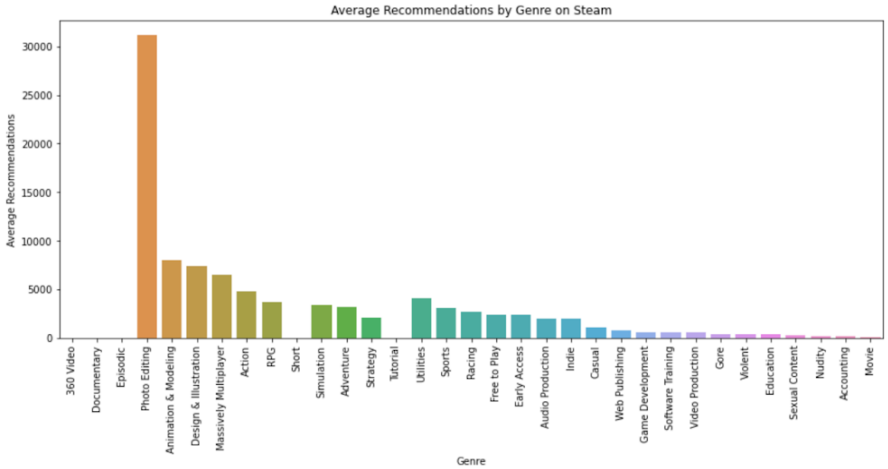
e	Negative	Recommendations	360 Video	Accounting	Action	...	Short	Simulation	Software Training	Sports	Strategy	Tutorial	Utilities	Video Production	Violent	Web Publishing
7	49	0	0	0	0	...	0	0	0	0	1	0	0	0	0	0
5	45	0	0	0	1	...	0	0	0	0	0	0	0	0	0	0
9	212	427	0	0	0	...	0	0	0	0	0	0	0	0	0	0
7	58	0	0	0	0	...	0	0	0	0	1	0	0	0	0	0
6	11	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0

Moving further from this, we start finding the relationship between ‘genres’ and reviews. Since the dataset we choose includes ‘positive’ and ‘negative’ reviews, we can use these to find what type of genre has the best review, and this might help us predict the elements of a successful game.

We first initialize dictionaries to store total positive and negative reviews for each genre. Then sum up the reviews for each genre. After that we normalized by the number of games in each genre. Then we sorted normalized_positive_reviews and normalized_negative_reviews by values in descending order. After all these steps, we construct histograms for ‘positive’ and ‘negative’ reviews, shown on the left.



Based on the graphs we have got, it shows that ‘photo editing’ is the genre that has the most positive review; and ‘Massively Multiplayer’ has the most negative reviews. We might predict a game with genres of ‘photo editing’ could be a successful game. And in our subsequent reports, we can use this data to further confirm that this prediction is correct.

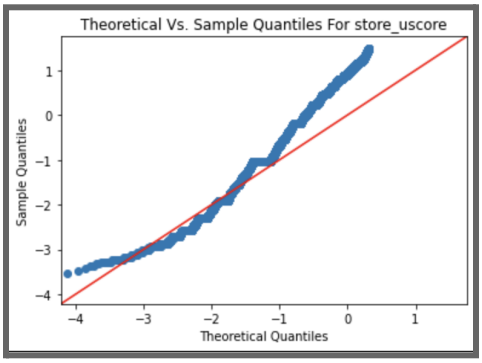


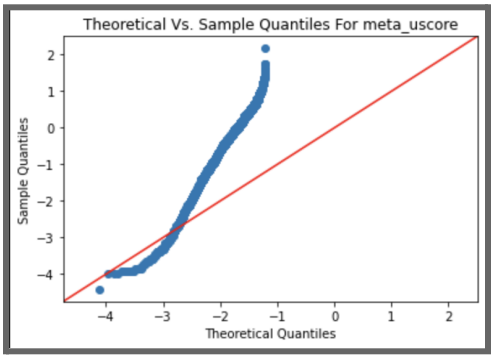
In addition to this, in order to further predict what games are likely to be successful, we likewise went to the relationship between genres and recommendations. By using a similar way of doing this: we initialized a dictionary to store the total recommendations for each genre; Summed up the recommendations for each genre; then normalized by the number of games in each genre; After that, we sorted normalized_recommendations in descending order.

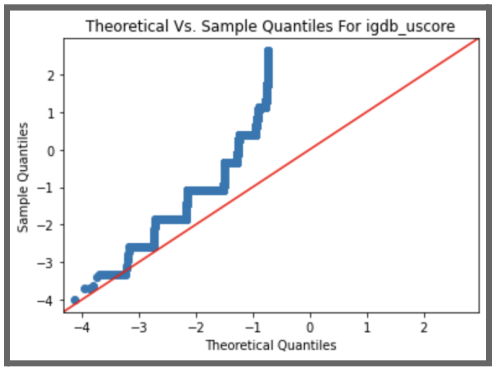
Based on what the histogram shows about the relationship between ‘genres’ and ‘recommendations’, it also indicates that ‘photo editing’ is the genre that has the most recommendations. This result gives us further certainty about the idea that ’successful games have photo editing in their genre‘.

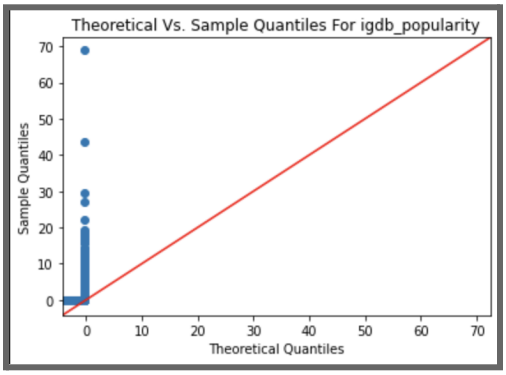
Quantitative Analysis Of Game Ratings:

Next, we analyzed the distribution of quantitative variables in our second dataset, focusing on columns ‘store_uscore’, ‘meta_uscore’, ‘igdb_popularity’, and ‘igdb_uscore’. These columns represented the ratings of Steam games for each rating platform. We created Q–Q Plots for each column, which compares the theoretical and sample quantiles for a quantitative variable to identify whether the distributions of the quantiles are similar. The following Q–Q plots were created:









As previously stated in “Regression Diagnostics”, we expected the trend of each plot to follow that of a straight line, indicating that the quantiles are generated from the same distribution. However, the plots show nonlinear trends for all quantitative columns. The Q–Q Plots for ‘store_uscore’ and ‘meta_uscore’ show exponential growth, and the steepest exponential growth can be observed in the Q–Q plot for ‘igdb_popularity’, which has a steep increase for theoretical quantiles greater than -0.5. A unique pattern can be seen in the graph for ‘igdb_uscore’, which depicts an exponential “staircase” pattern. All of these patterns indicate that there is skewness present for each store category and that the data is not normally distributed. These results can be correlated to the ineffectiveness of multivariate regression and the discrepancies in residual values.

Homogeneity Test:

As mentioned previously, due to the skewed diagnostic tests of the regression model and distribution of quantitative variables we transitioned to analysis involving the categorical features. One part of the categorical analysis involved a homogeneity test to analyze if the distribution of games by ranks per genre is different. The ranks are determined by the igdb_score (third-party user determined game score) and are divided into categories as follows {[10, 20) : 1, [20, 30) : 3 … [90, 100) : 9, [100] : 10} where every key corresponds to the range of the score and value to the respective rank. In addition, we took a subset of the top eight genres available in our dataset.

The purpose of the homogeneity is to analyze if certain genres have a higher or lower proportion of games in a higher ranking category compared to other genres. If the test results are significant it suggests that the genre of the game could be a factor that determines the score a game is more likely to receive.

Null hypothesis: The distribution of igdb_score ranks across all genres is the same.
Alternative hypothesis: The distribution of igdb_score ranks across all genres is different.

The crosstab data frame produced for the test is illustrated below.

ranks / genres	1	2	3	4	5	6	7	8	9	10
Action	0.003364	0.009672	0.024811	0.040370	0.095038	0.215307	0.320437	0.243902	0.045416	0.001682
Adventure	0.004530	0.007928	0.023783	0.041903	0.096829	0.214609	0.323330	0.248018	0.037373	0.001699
Casual	0.001350	0.012146	0.029690	0.040486	0.106613	0.229420	0.313090	0.228070	0.033738	0.005398
Indie	0.003956	0.007911	0.020965	0.038370	0.095332	0.223497	0.327532	0.242484	0.037975	0.001978
RPG	0.002294	0.005734	0.014908	0.032110	0.091743	0.217890	0.305046	0.280963	0.047018	0.002294
Racing	0.004975	0.024876	0.014925	0.029851	0.124378	0.268657	0.263682	0.238806	0.029851	0.000000
Simulation	0.001621	0.009724	0.032415	0.034036	0.089141	0.247974	0.307942	0.236629	0.035656	0.004862
Strategy	0.000000	0.003363	0.016816	0.032511	0.082960	0.198430	0.323991	0.288117	0.049327	0.004484
total	0.003102	0.008504	0.022711	0.038119	0.095148	0.220110	0.319260	0.249725	0.040820	0.002501

Unlike the methodology adopted in the homework assignment, for this experiment, we use the stats chi2 contingency package to run the homogeneity test. The degree of freedom for this test was 63, we obtained a large p-value, close to 1. Due to these results, we fail to reject the null hypothesis since we do not have enough evidence to suggest that there is a difference in the distribution of the genres across ratings. We continue our analysis and conduct a multinomial regression analysis to make predictions about our categorical variables.

Logistic Regression and Multinomial Logistic Regression:

In our data exploration phase, we tackled the challenge of defining game success by devising the "success_ratio" metric. This metric is the result of calculating the proportion of positive ratings to the aggregate count of both positive and negative ratings. It's a continuous value that varies from 0 to 1, providing a nuanced scale of player reception.

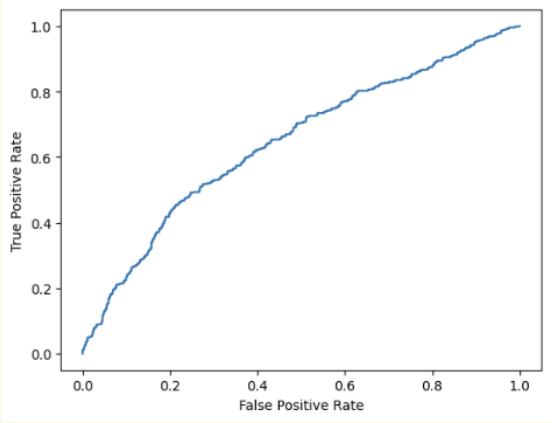
The centerpiece of our analysis was the deployment of logistic regression to differentiate video games into two discrete categories of success based on this "success_ratio": games were deemed "not success" if the ratio was below 0.8, and "success" if above.

To train and test our model, we partitioned the data into an 80/20 split, respectively. Here, 'X' denotes the array of input features, while 'Y' corresponds to the binary categorical outcomes of "success" and "not success."

The effectiveness of our classification model was evaluated using accuracy, which reflects the proportion of correct predictions in the test dataset. Although we utilized the statsmodels package to develop our

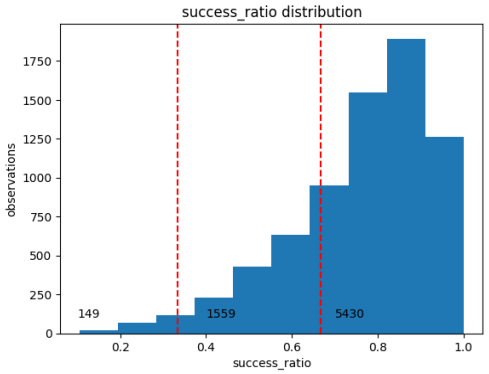
logistic regression framework, the accuracy reached was approximately 0.61, indicating room for model improvement or perhaps a need for more sophisticated metrics or additional features that could enhance performance.

To further assess our model's predictive capabilities, we visualized its performance using the Receiver Operating Characteristic (ROC) curve, shown below, applied it to the test dataset, and computed the area under the ROC curve (AUC) to obtain a more quantitative insight. The calculated roc_auc_score was approximately 0.617, suggesting that while our model performs better than random guessing (which would result in an AUC of 0.5), it does so modestly. An AUC score of 0.617 indicates that the model possesses a degree of ability to distinguish between the positive and negative classes; however, the closeness of this value to the threshold of randomness underscores a substantial potential for enhancing the model's discriminative power. This highlights the necessity for further model refinement or exploring alternative modeling approaches to increase its predictive accuracy and reliability in classifying positive instances.

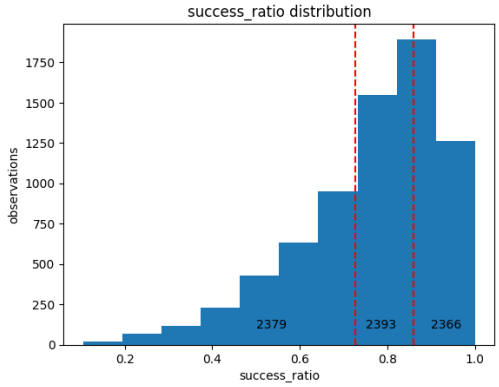


Furthermore, we analyzed multinomial logistic regression. Similar to logistic regression, we aimed at classifying video games into three distinct success categories: "not success," "ok performance," and "success" based upon "success_ratio," which we then segmented into the aforementioned categories. The categorization process was iteratively refined through multiple attempts to balance data distribution and ensure representativeness across categories.

The first attempt with the success_ratio distribution is shown:
The initial categorization effort for the multinomial logistic regression model involved dividing the "success_ratio" of video games into three equal parts to define the success categories: "not success" (0 to 0.333), "ok performance" (0.333 to 0.666), and "success" (0.666 to 1). This equal partitioning was grounded in the intent to impose an unbiased structure on the success metric. Despite achieving a promising accuracy of 77%, a closer examination revealed a data imbalance with an overwhelming majority of observations being classified as "success." This suggested that the model might disproportionately favor this outcome, thus skewing the accuracy metric and potentially overstating the model's predictive prowess.

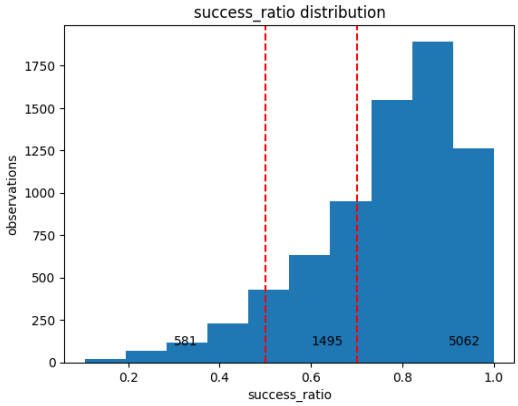


The second attempt with the success_ratio distribution is shown:
In response to the skewed distribution encountered in the first attempt, the second iteration sought to evenly distribute observations across the three success categories. This was achieved by adjusting the thresholds so that each category contained a similar number of observations: "not success" (0 to 0.727), "ok performance" (0.727 to 0.860), and "success" (0.860 to 1). Although this adjustment successfully balanced the categories, the model's accuracy dropped markedly to 41%. The



reduction highlighted a critical issue—uniform distribution of data points across categories does not inherently equate to effective classification. It revealed that the model struggled to discern the nuances between categories when the success ratio was not correlated with clearly distinct success states.

The finalized model for multinomial logistic regression with the success_ratio distribution is shown below:



Ultimately, the final model emerged as a pragmatic synthesis of the first two attempts. The success ratio thresholds were redefined to "not success" (0 to 0.5), "ok performance" (0.5 to 0.7), and "success" (0.7 to 1), resulting in a more representative data distribution among the categories while achieving a robust accuracy of 71%. This compromise sought to mitigate the data imbalance issue of the initial model while recapturing a significant portion of the predictive accuracy lost in the second attempt. By iteratively refining the category thresholds, the final model offered a more realistic and balanced classification, embodying an improved understanding of the varied definitions of success in the context of video game performance.

Random Forest Classifier and Decision Tree Classifier:
In the final phase of our analysis, we began experimenting with the Random Forest Classifier for this problem and achieved an initial accuracy of approximately 71%. We have not yet engaged in fine-tuning the model, indicating a potential for improvement once we deepen our understanding of its underlying mathematics and optimization techniques. Concurrently, we applied a Decision Tree Classifier and observed results closely aligned with those of the Random Forest, suggesting similar performance levels between the two models at this preliminary stage. Moving forward, it's crucial for us to further investigate and refine these models. However, we also entertain the possibility that the maximum achievable accuracy for this dataset might inherently be around 71% accuracy.

Interpretation of the Results and Inference

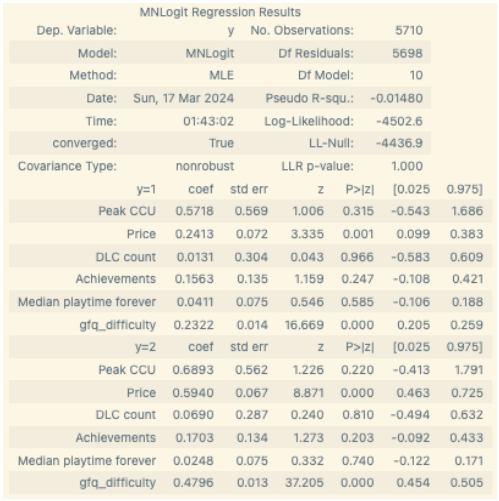
From our initial exploration, it is clear that due to sparsity in the dataset and unsuccessful diagnostic tests linear regression was not a good choice to model our data. Our model was not suitable since the residuals were not normally distributed and were not independent of each other. From here we transitioned to analyzing other features, in particular focusing on the qualitative features.

By analyzing quantitative features through the construction of Q–Q Plots, we were able to gain insight into the distribution of score data for different rating platforms. This provided insight into the skewness of the quantitative variables and guided us to look further into the similarity between review ratios for each genre.

The distributions of genres and reviews provide valuable insights into the landscape of video game preferences and their reception among players. Analyzing the relationship between reviews and genres allows us to understand which types of games tend to receive higher or lower ratings, which can inform game development and marketing strategies.

To analyze the distributions of various genres against the user scores we conducted a homogeneity test. The result of the test was that we failed to reject our null hypothesis which stated that the distributions of each genre are the same. Hence, we do not have sufficient evidence to conclude that the genre of a game can help predict what score category it is more likely to fall in.

Next, we proceed to draw inferences from the multinomial logistic regression model by examining the summary of the model, shown on the left picture.



MNLogit Regression Results

Dep. Variable:		y	No. Observations:		5710	
Model:		MNLogit	Df Residuals:		5698	
Method:		MLE	Df Model:		10	
Date:	Sun, 17 Mar 2024		Pseudo R-squ.:		-0.01480	
Time:		01:43:02	Log-Likelihood:		-4502.6	
converged:		True	LL-Null:		-4436.9	
Covariance Type:		nonrobust	LLR p-value:		1.000	
y=1	coef	std err	z	P>|z|	[0.025	0.975]
Peak CCU	0.5718	0.569	1.006	0.315	-0.543	1.686
Price	0.2413	0.072	3.335	0.001	0.099	0.383
DLC count	0.0131	0.304	0.043	0.966	-0.583	0.609
Achievements	0.1563	0.135	1.159	0.247	-0.108	0.421
Median playtime forever	0.0411	0.075	0.546	0.585	-0.106	0.188
gfq_difficulty	0.2322	0.014	16.669	0.000	0.205	0.259
y=2	coef	std err	z	P>|z|	[0.025	0.975]
Peak CCU	0.6893	0.562	1.226	0.220	-0.413	1.791
Price	0.5940	0.067	8.871	0.000	0.463	0.725
DLC count	0.0690	0.287	0.240	0.810	-0.494	0.632
Achievements	0.1703	0.134	1.273	0.203	-0.092	0.433
Median playtime forever	0.0248	0.075	0.332	0.740	-0.122	0.171
gfq_difficulty	0.4796	0.013	37.205	0.000	0.454	0.505

In our multinomial logistic model, y=2 represents the success category, y=1 represents ok performance, and y=0 represents not success.

We make inferences for each feature's coefficient, their statistical significance (P values), and confidence intervals for both outcome categories y=1 and y=2 (another y=0 category is for baseline, not included below):

For category y = 1, which means “ok performance” category for 0.7 > “success_ratio” ≥ 0.5:

Peak CCU:

- Coefficient: 0.5718 implies that for a one-unit increase in Peak CCU, the log-odds of being in category 1 relative to the reference category (likely 0) increase by 0.5718, holding other variables constant.
- P-value: 0.315 suggests that the relationship between Peak CCU and the outcome is not statistically significant since the p-value is above the common threshold of 0.05.
- Confidence Interval: [-0.543, 1.686] includes zero, which further indicates that the coefficient is not statistically significant.
- Inference: Peak CCU does not appear to be a strong predictor for category 1 in this model.

Price:

- Coefficient: 0.2413 suggests that as the price increases by one unit, the log-odds of being in category 1 increase by 0.2413, holding other variables constant.
- P-value: 0.001 indicates a statistically significant relationship between Price and the outcome for category 1.
- Confidence Interval: [0.099, 0.383] does not include zero, confirming the significance of this variable.
- Inference: Price is a statistically significant predictor for the likelihood of being in category 1.

DLC count:

- Coefficient: 0.1313 indicates a positive relationship between DLC count and the outcome, but the effect size is relatively small.
- P-value: 0.966 shows that this is not a statistically significant predictor.
- Confidence Interval: [-0.583, 0.609] includes zero.
- Inference: DLC count does not significantly influence the likelihood of the outcome being in category 1.

Achievements:

- Coefficient: 0.1563 shows a slight positive relationship with the outcome.
- P-value: 0.247 is above the significance threshold.
- Confidence Interval: [-0.108, 0.421] includes zero.
- Inference: Achievements do not significantly impact the prediction of category 1.

Median playtime forever:

- Coefficient: 0.0411 indicates that higher median playtime is weakly associated with the likelihood of being in category 1.
- P-value: 0.585 suggests non-significance.
- Confidence Interval: [-0.106, 0.188] includes zero.
- Inference: Median playtime forever is not a significant factor in predicting category 1.

gfq_difficulty:

- Coefficient: 0.2322 implies that greater difficulty is associated with an increased likelihood of being in category 1.
- P-value: < 0.001 indicates a highly significant predictor.
- Confidence Interval: [0.205, 0.259] does not cross zero.
- Inference: gfq_difficulty is a significant and strong predictor for category 1.

For category y = 2, which means “success” category for “success_ratio” > 0.7:

Peak CCU:

- Coefficient: 0.6893 suggests that for a one-unit increase in Peak CCU, the log-odds of being in category 2 relative to the reference category increase by 0.6893, holding other variables constant.
- P-value: 0.220 means the variable is not statistically significant for category 2.
- Confidence Interval: [-0.413, 1.791] includes zero.
- Inference: Peak CCU is not a significant predictor for category 2.

Price:

- Coefficient: 0.5940 indicates a strong positive relationship between price and the outcome for category 2.
- P-value: < 0.001 signifies strong statistical significance.
- Confidence Interval: [0.463, 0.725] excludes zero, confirming its significance.
- Inference: Price is a significant predictor for category 2, suggesting that higher prices increase the likelihood of the outcome being in this category.

DLC count:

- Coefficient: 0.0690 indicates a small positive relationship with the outcome for category 2.
- P-value: 0.810 shows non-significance.
- Confidence Interval: [-0.494, 0.632] includes zero.
- Inference: DLC count is not a significant predictor for category 2.

Achievements:

- Coefficient: 0.1703 suggests a slight positive relationship with the outcome for category 2.
- P-value: 0.203 is above the significance threshold.
- Confidence Interval: [-0.092, 0.433] includes zero.
- Inference: Achievements do not significantly affect the prediction for category 2.

Median playtime forever:

- Coefficient: 0.0248 indicates a very small positive association with the outcome for category 2.
- P-value: 0.740 suggests non-significance.
- Confidence Interval: [-0.122, 0.171] includes zero.
- Inference: Median playtime forever is not significant in predicting category 2.

gfq_difficulty:

- Coefficient: 0.4796 implies a strong positive relationship with the outcome for category 2.
- P-value: < 0.001 indicates a highly significant relationship.
- Confidence Interval: [0.454, 0.505] does not include zero.
- Inference: gfq_difficulty is a highly significant predictor for category 2, with a strong positive effect on the likelihood of the outcome being in this category.

Our analysis of the multinomial logistic regression model revealed two standout features that significantly impact a video game's market success: gfq_difficulty and price. The model highlighted a trend where games classified as "success" (category y =2) typically possess a median level of difficulty (shown in Figure 1) and are priced between free to $20 (shown in Figure 2). This finding suggests a market preference for games that offer a moderate challenge and are affordably priced, striking a sweet spot that

likely appeals to a broad audience. Essentially, games that balance engaging difficulty with accessible pricing appear to have a higher likelihood of success in the competitive gaming market.

Figure 1

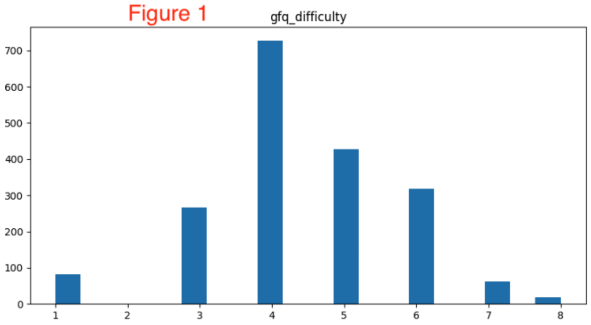
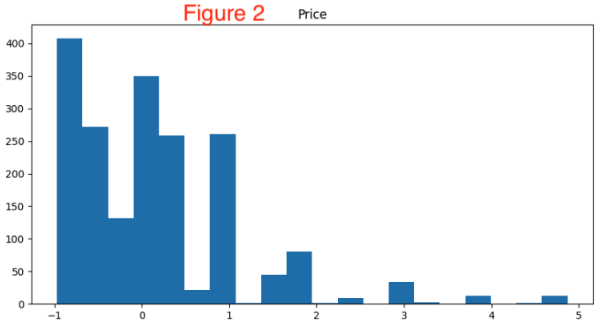


Figure 2



In evaluating the robustness of our logistic regression model, we paid particular attention to the potential issue of multicollinearity among the independent variables. To this end, we calculated the Variance Inflation Factors (VIF) for each predictor. The VIF scores obtained were as follows:
VIF: Peak CCU: 1.003
VIF: Price: 1.020
VIF: DLC count: 1.011
VIF: Achievements: 1.001
VIF: Median playtime forever: 1.012
VIF: gfq_difficulty: 1.000
As VIF scores close to 1 indicate a minimal level of multicollinearity, we can deduce that our predictors are demonstrating low intercorrelations. Consequently, this suggests that each variable in our model is largely independent of the others, and we can be more confident in the validity of our coefficient estimates. The absence of significant multicollinearity supports the reliability and interpretability of our model, enhancing the credibility of the insights drawn from our analysis.

Conclusion and Future Work

Our analysis of the relationship between game genres and reviews provides valuable insights into the factors driving game success. While 'photo editing' emerges as a genre with the highest count of positive reviews, suggesting player satisfaction and engagement with its gameplay mechanics, 'Massively Multiplayer' games stand out with the highest count of negative reviews, indicating potential challenges in this genre related to technical issues or community management. These findings underscore the significance of genre selection in shaping player perceptions and highlight opportunities for developers to leverage player sentiment to optimize game offerings and enhance player experiences, ultimately increasing their chances of success in the competitive gaming market.

Through multivariate regression analysis, we attempted to predict the ratio of the positive to the negative scores with quantitative features (like peak CCU, Price, etc). The results were invalid since we failed the regression diagnostic tests. Through further analysis, we concluded that the quantitative features were not quite suitable for regression analysis and we shifted analysis to the categorical variables. Furthermore, the multinomial logistic regression model achieves an accuracy of approximately 71%, indicating its effectiveness in classifying a game's success based on carefully selected features. Further analysis reveals that successful games are commonly characterized by a median difficulty level and a price range of free to $20, helping us identify critical features that contribute towards a game’s popularity and long-term success.

As a result, our future efforts will be centered around improving classification methods. We have begun our initial exploration into Decision Tree classifiers, and also Random Forest classifiers, which make use of multiple Decision Trees and can prove to be a more robust model. By using these models and aiming for further improvement, we hope to achieve an accuracy higher than 71 percent.