EXECUTIVE SUMMARY

Video game is considered as one of the most popular leisure activities in the world since 1970s. Every year, there are more than hundreds of new video games coming up in the public. As the industry is growing faster and faster, accurate analysis is of becoming vital importance for videogame companies as well as publishers. For example, based on customer analysis such as age distribution and category preference, the company can generate a better marketing strategy specifically for target customers. And based on sales region analysis, it is clear which kind of videogame is more popular in which area. However, analysis is not enough in today's serve competition. We cannot wait for the result of market selection but instead, we have to predict the results to take actions ahead of time.

Here we chose a data set about videogames sales situation in different areas with additional information such as Platform, Genre, Critic Score and so on. Based on these information, we can have a prediction of videogames sales number as well as the popularity in North American Region in the future. The following table is the explanation of each variable in our data set.

VARIABLE	DESCRIPTION
Name	The Name of the Video game
Platform	Console on which the game is running
Year of Release	On which year the video game came out to the public

Genre	Categories of the video game
Publisher	The company which released the video game
NA_Sales	Sales number in north America region
EU_Sales	Sales number in Europe region
JP_Sales	Sales number in Japan region
Other_Sales	Sales number in other regions
Global_Sales	Sales number in global market, which is the sum of sales number in NA, EU, JP and others
Critic_Score	Score graded by testers before release
Critic_Count	Number of people who have graded the video game before release
User_Score	Score graded by users after the video game came to the market
User_Count	Number of users who have graded the videogame
Developer	Party responsible for creating the game
Rating	The ESRB ratings

Based on SEMMA schema, firstly we sampled the data set with 50% in training, 25% in validation and 25% in test. During the data preprocessing and cleaning process, we excluded unrelated variables such as User Score and User Count since these two variables are the value after videogames releasing. What's more, we removed all the sales number in other regions such as Europe Sales and Japan Sales. In the end, we chose five predictors which are Platform, Genre, Critic Score, Critic Count and Rating. For the missing values and outliers, we simply excluded 9129 missing values and 3 values which are far from others. . The problem we found here is that the distribution of NA Sales was too crowded between 0 to 2, even after standardization the distribution wasn't changed, then we transformed that variable to the log value of NA Sales.

The main prediction we did is about NA sales number, so here we tried both continuous predictions to predict the actual sales number and categorical prediction to predict the sales range. The models we applied for numeric model include linear regression, Neural, Naïve Bayes, Bootstrap Forest and K-NN. Among all these methodologies, linear regression performed best with an interaction variable calculated by Critic Score * Critic Count. After applying this interaction variable, R Square and RMSE performed better than before. For the sales range estimation, we chose a 0.5 binning width for each range and tried K-NN and Naïve Bayes. Compared with Naïve Bayes, K-NN owned a lower misclassification rate, which is 0.42.

When it comes to the popularity, the first thing we did is how to define whether the videogame will be popular or not. Among all the variables, we used Critic Score and Critic Count as measures of division. If Critic Score * Critic Count > 2400, then we define it as popular, or we consider it as unpopular. The methodology we applied here is Naïve Bayes and we selected

Platform, Genre and Rating as predictors. Misclassification rate is 0.32 and accuracy for unpopular is 0.72, which can be considered as a good model for categorical prediction.

Based on all those models we tried and results we mentioned above, we conclude that: : (1) Critic Score and Critic Count are the most important predictors when we predict the sales number (2) Genre contribute a lot in a consideration of popularity since different people have different preferences (3) Advanced platforms are more popular (4) Rating would basically affect the population of certain age group that has access to games meant for them. According to the conclusions, we recommend that for designers, they can design more games in category of action, shooting and role-playing since those are top three popular video games among all genres. What's more, they should pay attention to update videogames for a more advanced platform since it would be more popular for customers. Based on our models, the company could predict the sales number of each game before release so that they could take different marketing strategies to maximize profits and save costs.

METHODOLOGY

SAMPLE

OUTLIERS

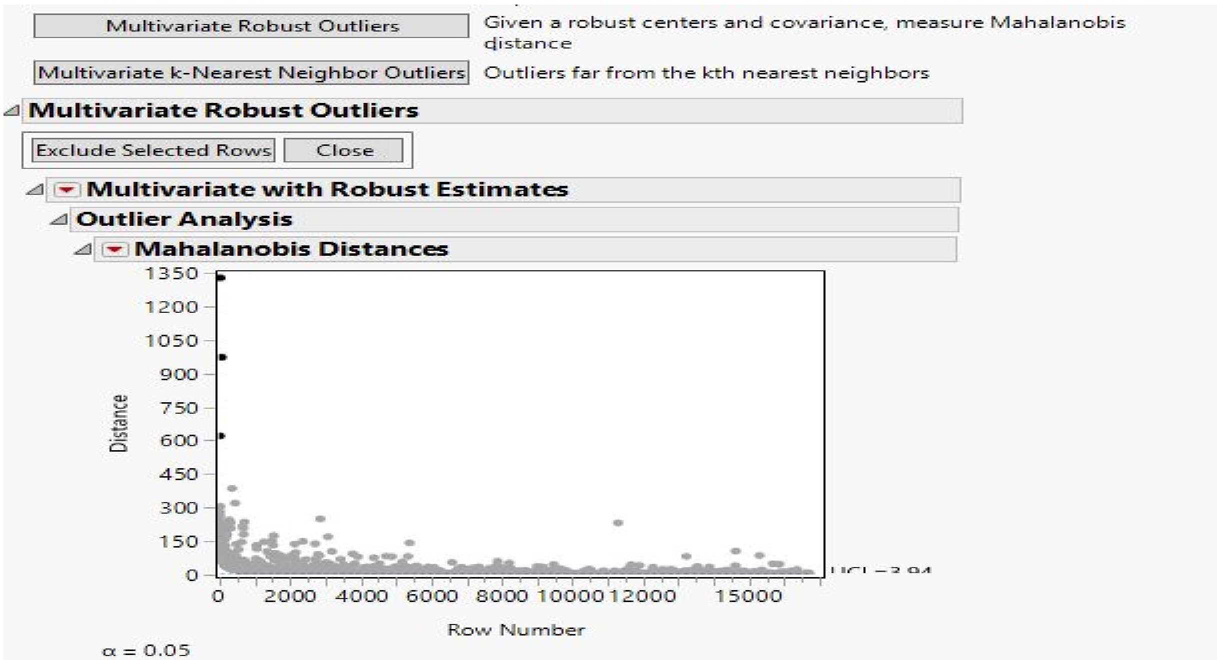


Figure 1 Outlier Analysis

While exploring outliers, we found that 3 records were significantly far but these are results of either poor sales, less number of user or critic counts.

MISSING VALUES

Since this a Retail Commodity, we have chosen to exclude the rows with missing values as we have enough data to perform analysis.

Also, there is no case of rare events or outcomes in this data set.

No. of missing values in User count = 9129

After removing missing values from user count, 573 rows of data have missing values in both critic score and critic count.

Also, 70 records did not have any rating which were excluded along with the 573 rows.

There are rows with NA_Sales = 0. These rows belong to the games which were not released in North America but in other regions. We omit these rows as including them would affect the model and there might be a situation where a particular game sale is predicted negatively after considering the previous data of 0 sales.

No. of rows with 0 NA Sales = 550

Final data= 6397 rows

PARTITIONING:

The data is partitioned into following proportion :

- Training - 0.50
- Validation – 0.25
- Test – 0.25

Fixed random option was used with seed equal to 5.

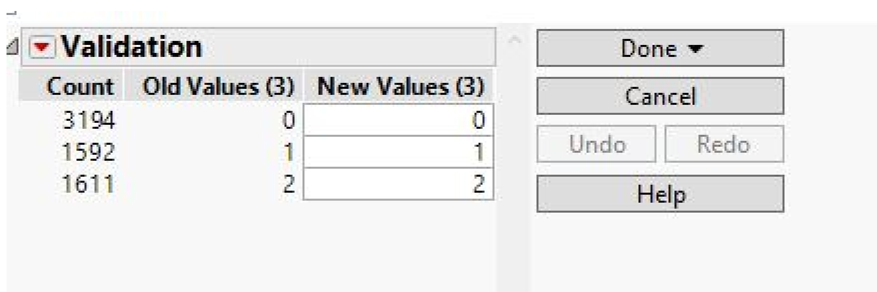


Figure 2 Create Validation Column

POTENTIAL PREDICTORS

By looking at the data and considering the Gaming & Retail field 'Critic Score' is the strongest Predictor among the available variables.

The predictors that will be used for analysis are:

- Platform
- Genre
- Critic Score
- Critic Count
- Rating

INTERACTION TERMS:

Since Critic Score and Critic Count are very good predictors combining these two variables can be very significant.

Response NA_Sales

Validation: Validation

Singularity Details

Intercept = Platform[3DS] - 16*Platform[DC] + Platform[DS] + Platform[GBA] + Platform[GC] + Platform[PC] + Platform[PS] + Platform[PS2] + Platform[PS3] + Platform[PS4] + Platform[PSP] + Platform[PSV] + Platform[Wii] + Platform[WiiU] + Platform[X360] + Platform[XB] = - 5*Rating[AO] + Rating[E] + Rating[E10+] + Rating[K-A] + Rating[M]

Effect Summary

Source	LogWorth	PValue
Critic_Count	35.918	0.00000
Critic_Score	21.204	0.00000
Platform	16.021	0.00000
Rating	7.807	0.00000
Genre	5.926	0.00000

Remove Add Edit FDR

Lack Of Fit

Source	DF	Sum of Squares	Mean Square	F Ratio
Lack Of Fit	3130	2177.0957	0.695558	2.7780
Pure Error	31	7.7619	0.250382	Prob > F
Total Error	3161	2184.8575		0.0004*
				Max RSq
				0.9971

Summary of Fit

RSquare	0.182858
RSquare Adj	0.174586
Root Mean Square Error	0.83138
Mean of Response	0.421597
Observations (or Sum Wgts)	3194

Crossvalidation

Source	RSquare	RASE	Freq
Training Set	0.1870	0.7978	3474
Validation Set	0.1790	0.5905	1737
Test Set	0.1005	1.2380	1736

Figure 3 Test Regression Result for NA_Sales

The above screenshot shows the RSq and RMSE values, which is low and requires an interaction term in order to improve the model.

Response NA_Sales

Validation: Validation

Singularity Details

Intercept = Platform[3DS] - 16*Platform[DC] + Platform[DS] + Platform[GBA] + Platform[GC] + Platform[PC] + Platform[PS] + Platform[PS2] + Platform[PS3] + Platform[PS4] + Platform[PSP] + Platform[PSV] + Platform[Wii] + Platform[WiiU] + Platform[X360] + Platform[XB] = - 5*Rating[AO] + Rating[E] + Rating[E10+] + Rating[K-A] + Rating[M]

Effect Summary

Source	LogWorth	PValue
Critic_Score	41.277	0.00000
Critic_Score*Critic_Count	41.269	0.00000
Platform	15.641	0.00000
Critic_Count	12.452	0.00000 ^
Rating	5.905	0.00000
Genre	5.705	0.00000

Remove Add Edit FDR ('^' denotes effects with containing effects above them)

Lack Of Fit

Source	DF	Sum of Squares	Mean Square	F Ratio
Lack Of Fit	3129	2053.2475	0.656199	2.6208
Pure Error	31	7.7619	0.250382	Prob > F
Total Error	3160	2061.0093		0.0008*
				Max RSq
				0.9971

Summary of Fit

RSquare	0.229178
RSquare Adj	0.221128
Root Mean Square Error	0.8076
Mean of Response	0.421597
Observations (or Sum Wgts)	3194

Crossvalidation

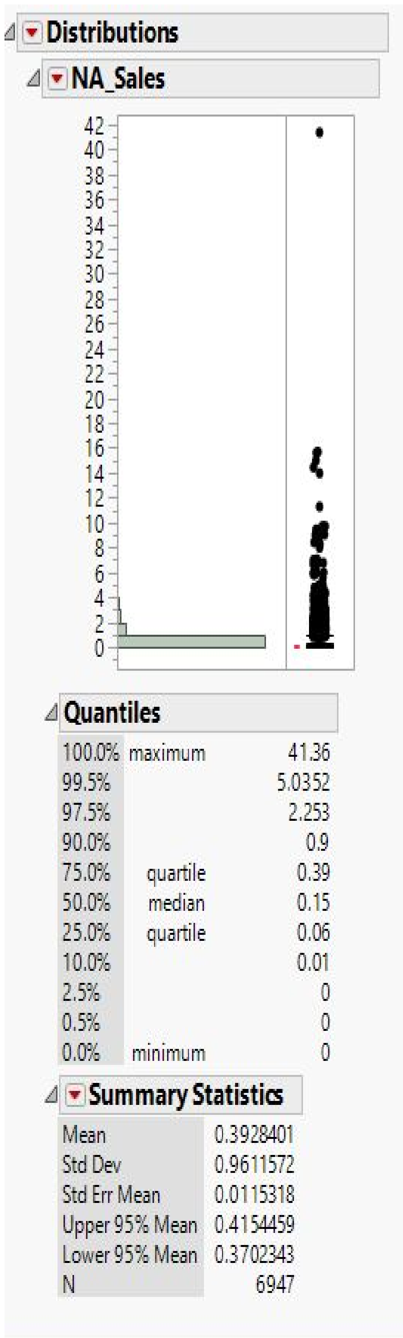
Source	RSquare	RASE	Freq
Training Set	0.2324	0.7752	3474
Validation Set	0.2177	0.5764	1737
Test Set	0.1170	1.2266	1736

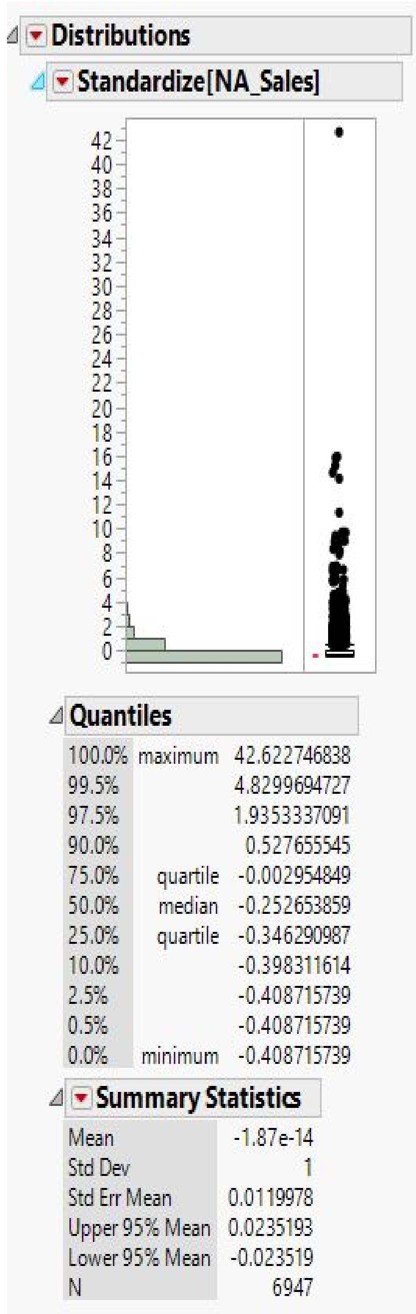
Figure 4 Test Regression Result for log(NA_Sales) with interaction term

After adding the interaction term, the model's RSq and RMSE improved.

<u>TRANSFORMATION</u>

Looking at the NA_Sales column, the numbers are read in million dollars and the value which might get a bit difficult when we have to set them to ranges when the management requires to know the group of games that belong to a particular sales range.





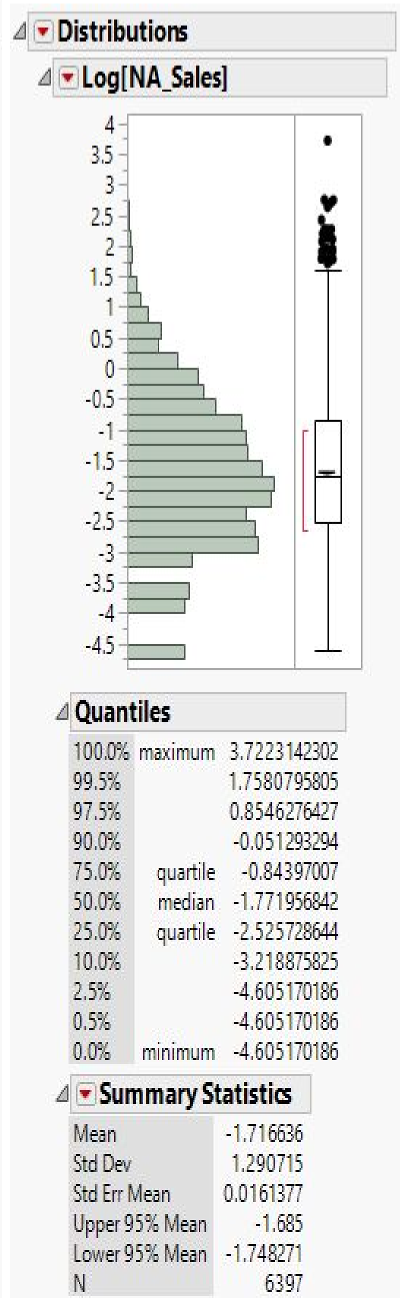
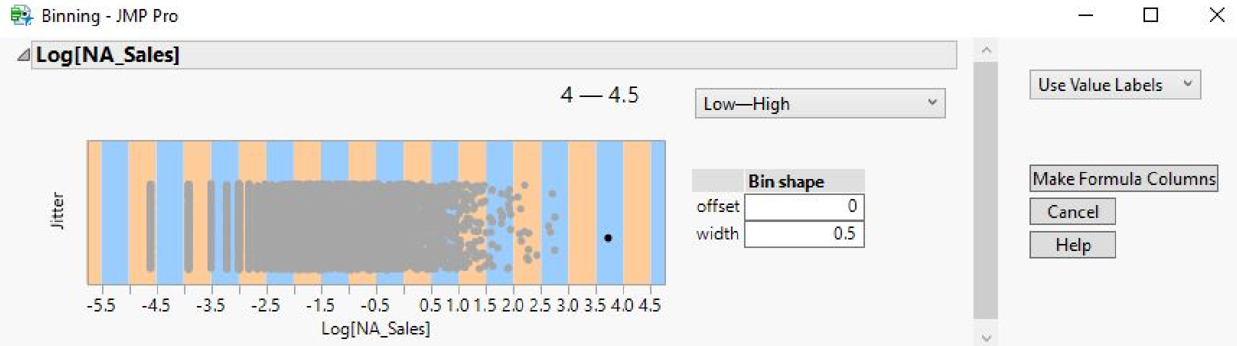


Figure 5 DIstribution of NA_Sales   Figure 6 Dist. of Standardized NA_Sales   Figure 7 Distribution of log(NA_Sales)

Standardizing the variable also does not change the distribution and hence, log transformation was used in order to bin the NA_Sales into required ranges.

<u>BINNING:</u>



	Global_Sales	Critic_Score	Critic_Count	User_Score	User_Count	Developer	Rating	Validation	Log[NA_Sales]	Residual Log[NA_Sales]	Log[NA_Sales] Binned
1	82.53	76	51	8	322	Nintendo	E	Test	3.7223142302	4.1175000381	3.5 — 4
2	35.52	82	73	8.3	709	Nintendo	E	Training	2.7523860149	2.6113298316	2.5 — 3
3	32.77	80	73	8	192	Nintendo	E	Training	2.7479117345	2.5855741643	2.5 — 3
4	29.8	89	65	8.5	431	Nintendo	E	Test	2.4230312461	2.5726232587	2 — 2.5
5	28.92	58	41	6.6	129	Nintendo	E	Training	2.6361960973	3.4682190427	2.5 — 3
6	28.32	87	80	8.4	594	Nintendo	E	Test	2.6700021335	2.2161285608	2.5 — 3
7	23.21	91	64	8.6	464	Nintendo	E	Test	2.2731562823	2.4249518918	2 — 2.5
8	22.7	80	63	7.7	146	Nintendo	E	Test	2.1882959466	2.2305897977	2 — 2.5
9	21.81	61	45	6.3	106	Good Science Stu...	E	Training	2.7080502011	3.6697104085	2.5 — 3
10	21.79	80	33	7.4	52	Nintendo	E	Test	2.1983350716	2.8545231867	2 — 2.5
11	21.04	97	50	8.2	3994	Rockstar North	M	Training	1.948763218	2.3183020503	1.5 — 2
12	20.81	95	80	9	1588	Rockstar North	M	Training	2.2438960966	2.1544710346	2 — 2.5
13	20.15	77	58	7.9	50	Nintendo	E	Training	1.5560371357	1.8805345504	1.5 — 2
14	16.27	97	58	8.1	3711	Rockstar North	M	Validation	2.2679936482	2.4238283323	2 — 2.5
15	16.15	95	62	8.7	730	Rockstar North	M	Training	2.129421474	2.4083329703	2 — 2.5
16	15.29	77	37	7.1	19	Nintendo	E	Validation	1.2325602612	2.8971382439	1 — 1.5
17	14.98	95	54	8.4	314	Polyphony Digital	E	Training	1.9242486523	2.2664086271	1.5 — 2
18	14.73	88	81	3.4	8713	Infinity Ward, Sle...	M	Training	2.2016591744	2.1033417203	2 — 2.5
19	14.61	87	89	6.3	1454	Treyarch	M	Training	2.2721258855	2.0369368571	2 — 2.5
20	13.79	83	21	5.3	922	Treyarch	M	Validation	1.6074359098	2.9210738088	1.5 — 2
21	13.67	83	73	4.8	2256	Treyarch	M	Training	2.1102132003	2.3097686973	2 — 2.5
22	13.47	94	100	6.3	2698	Infinity Ward	M	Training	2.1424163408	1.4942978093	2 — 2.5
23	13.32	88	39	3.2	5234	Infinity Ward, Sle...	M	Training	1.7119945008	2.5231280273	1.5 — 2
24	13.1	97	56	8.5	664	DMA Design	M	Test	1.9444805562	2.2925037798	1.5 — 2
25	12.84	93	81	8.9	1662	Game Arts	T	Validation	1.8900953699	1.4578694756	1.5 — 2
26	12.66	85	73	8.2	632	Retro Studios, En...	E	Training	1.6154199841	1.9593333045	1.5 — 2
27	12.63	88	58	6.4	1094	Treyarch	M	Validation	1.7900914121	2.2124252382	1.5 — 2

Figure 8 Binning for log(NA_Sales)

MODEL

<u>For continuous NA sales prediction</u>

1. **REGRESSION:** With NA_Sales being the target variable, we can build a regression model where we assess the p-value or significance of the variables and also the RSq and root mean square error.

Response NA_Sales

Validation: Validation

Singularity Details

Intercept = Platform[3DS] - 16*Platform[DC] + Platform[DS] + Platform[GBA] + Platform[GC] + Platform[PC] + Platform[PS] + Platform[PS2] + Platform[PS3] + Platform[PS4] + Platform[PSP] + Platform[PSV] + Platform[Wii] + Platform[WiiU] + Platform[X360] + Platform[XB] = - 5*Rating[AO] + Rating[E] + Rating[E10+] + Rating[K-A] + Rating[M]

Effect Summary

Source	LogWorth	PValue	
Critic_Score	41.277	0.00000	
Critic_Score*Critic_Count	41.269	0.00000	
Platform	15.641	0.00000	
Critic_Count	12.452	0.00000	^
Rating	5.905	0.00000	
Genre	5.705	0.00000	

Remove Add Edit FDR ('^' denotes effects with containing effects above them)

Lack Of Fit

Source	DF	Sum of Squares	Mean Square	F Ratio
Lack Of Fit	3129	2053.2475	0.656199	2.6208
Pure Error	31	7.7619	0.250382	**Prob > F**
Total Error	3160	2061.0093		0.0008*
				Max RSq
				0.9971

Summary of Fit

RSquare	0.229178
RSquare Adj	0.221128
Root Mean Square Error	0.8076
Mean of Response	0.421597
Observations (or Sum Wgts)	3194

Crossvalidation

Source	RSquare	RASE	Freq
Training Set	0.2292	0.8033	3194
Validation Set	0.2062	0.5971	1592
Test Set	0.1131	1.2710	1611

Figure 9 Regression Result for Log(NA_Sales) with interaction term

In this regression model, with the help of interaction term Critic score*Critic count, R-square and RMSE improved to 0.21/0.59 and 0.11/1.27in validation and test set,respectively.

2. **NEURAL NETWORKS:**

This model is used mainly to check if there was a huge difference in the RSq and RMSE as this model would not help us in explaining the contribution of each variable, which we know are very good predictors from business standpoint.

Model 1: With Tanh – 5, 1 layer

Neural

Validation Column: Validation

Model Launch

Model NTanH(5)

Measures	Training NA_Sales Value	Validation NA_Sales Value	Test NA_Sales Value
RSquare	0.3297376	0.2205764	0.1610107
RMSE	0.7490621	0.5917069	1.2362222
Mean Abs Dev	0.3213523	0.3265951	0.3533208
-LogLikelihood	3609.2362	1423.5579	2627.5388
SSE	1792.1341	557.38637	2462.0031
Sum Freq	3194	1592	1611

Figure 10 Neural Result 1 for Log(NA_Sales)

Model 2: 2-TanH, 2-Linear, 1-Gaussian

Neural

Validation Column: Validation

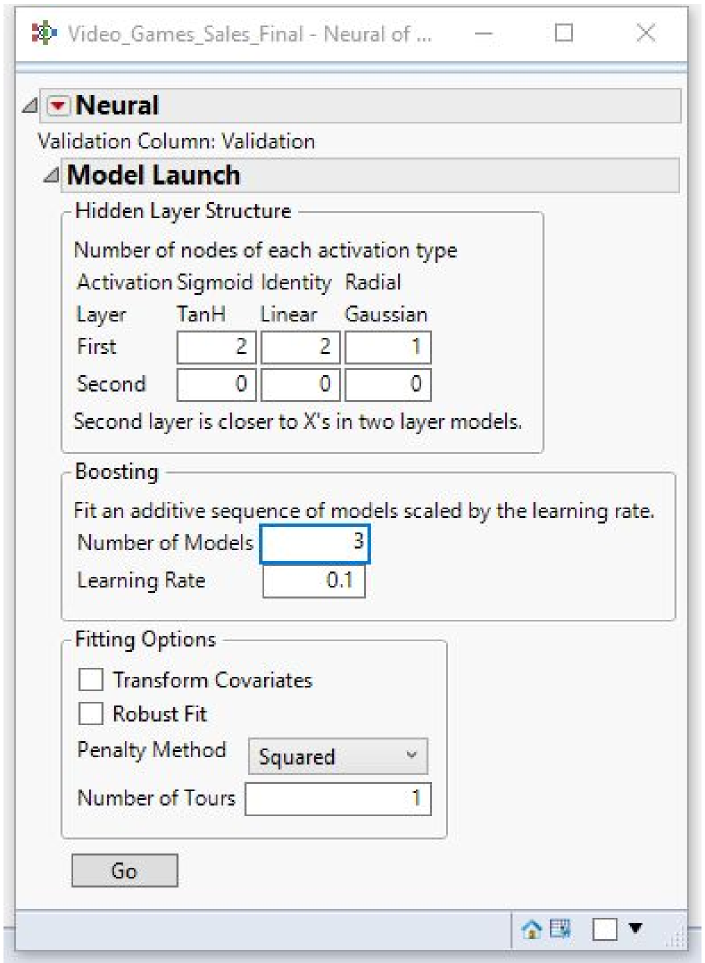
Model Launch

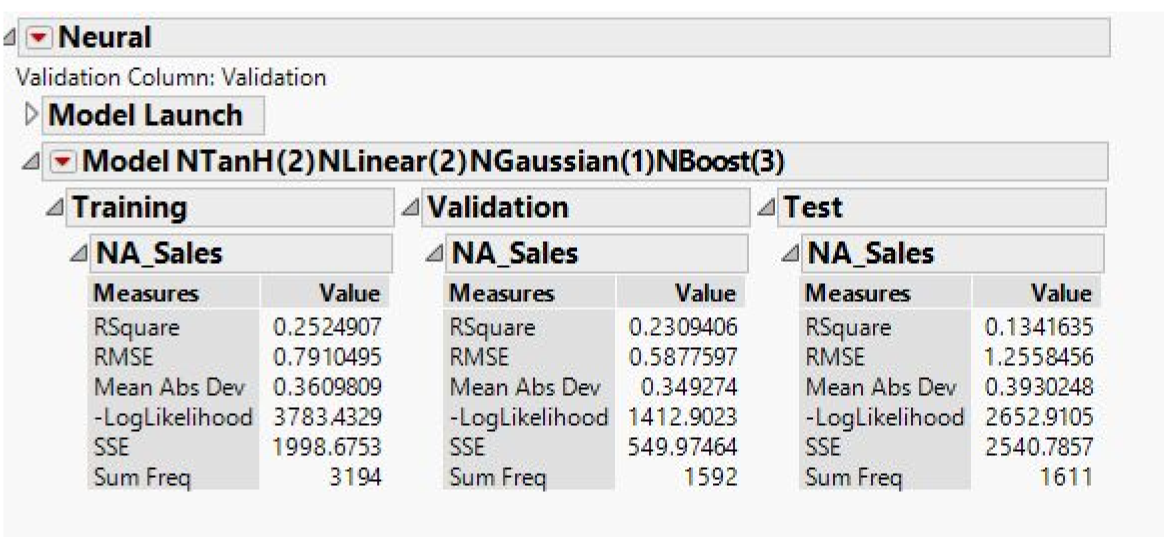
Model NTanH(2)NLinear(2)NGaussian(1)

Measures	Training NA_Sales Value	Validation NA_Sales Value	Test NA_Sales Value
RSquare	0.2698538	0.2330157	0.1024906
RMSE	0.7818083	0.5869662	1.2786092
Mean Abs Dev	0.3413259	0.3399999	0.3795747
-LogLikelihood	3745.9004	1410.7516	2681.8501
SSE	1952.2503	548.49067	2633.7296
Sum Freq	3194	1592	1611

Figure 11 Neural Result 2 for Log(NA_Sales)

Model 3: 2-TanH, 2-Linear, 1-Gaussian, with Boosted tree where No. of trees = 3





Training — NA_Sales

Measures	Value
RSquare	0.2524907
RMSE	0.7910495
Mean Abs Dev	0.3609809
-LogLikelihood	3783.4329
SSE	1998.6753
Sum Freq	3194

Validation — NA_Sales

Measures	Value
RSquare	0.2309406
RMSE	0.5877597
Mean Abs Dev	0.349274
-LogLikelihood	1412.9023
SSE	549.97464
Sum Freq	1592

Test — NA_Sales

Measures	Value
RSquare	0.1341635
RMSE	1.2558456
Mean Abs Dev	0.3930248
-LogLikelihood	2652.9105
SSE	2540.7857
Sum Freq	1611

Figure 12 Neural Result 3 for Log(NA_Sales)

We can tell R-square/RMSE in validation and test set from model 1 is 0.22/0.59 and 0.16/1.23,respectively. While model 2 yields 0.23/0.59 and 0.10/1.27. Model 1 is better than model 2 in general. Model 3 has R-square/RMSE in validation and test set 0.23/0.59 and 0.13/1.26,respectively. Using rule of thumb and considering nodes as the number of predictors, five nodes with TanH function produces better results among the other tested combinations.

3. **BOOTSTRAP:** To check if the RSq and RMSE values can improved we built bootstrap tree as it is assuming that a complex model would improve the performance.

Number of trees in forest 10; Number of terms sampled per split: 2

Bootstrap Forest for NA_Sales

Specifications

Target Column:	NA_Sales	Training Rows:	3194
Validation Column:	Validation	Validation Rows:	1592
		Test Rows:	1611
Number of Trees in the Forest:	10	Number of Terms:	5
Number of Terms Sampled per Split:	2	Bootstrap Samples:	3194
		Minimum Splits per Tree:	10
		Minimum Size Split:	6

Overall Statistics

Individual Trees	RMSE
In Bag	0.4953514
Out of Bag	0.8837926

	RSquare	RMSE	N
Training	0.541	0.6198656	3194
Validation	0.230	0.5882306	1592
Test	0.144	1.2485544	1611

Cumulative Validation

Per-Tree Summaries

Figure 13 Bootstrap Result for Log(NA_Sales)

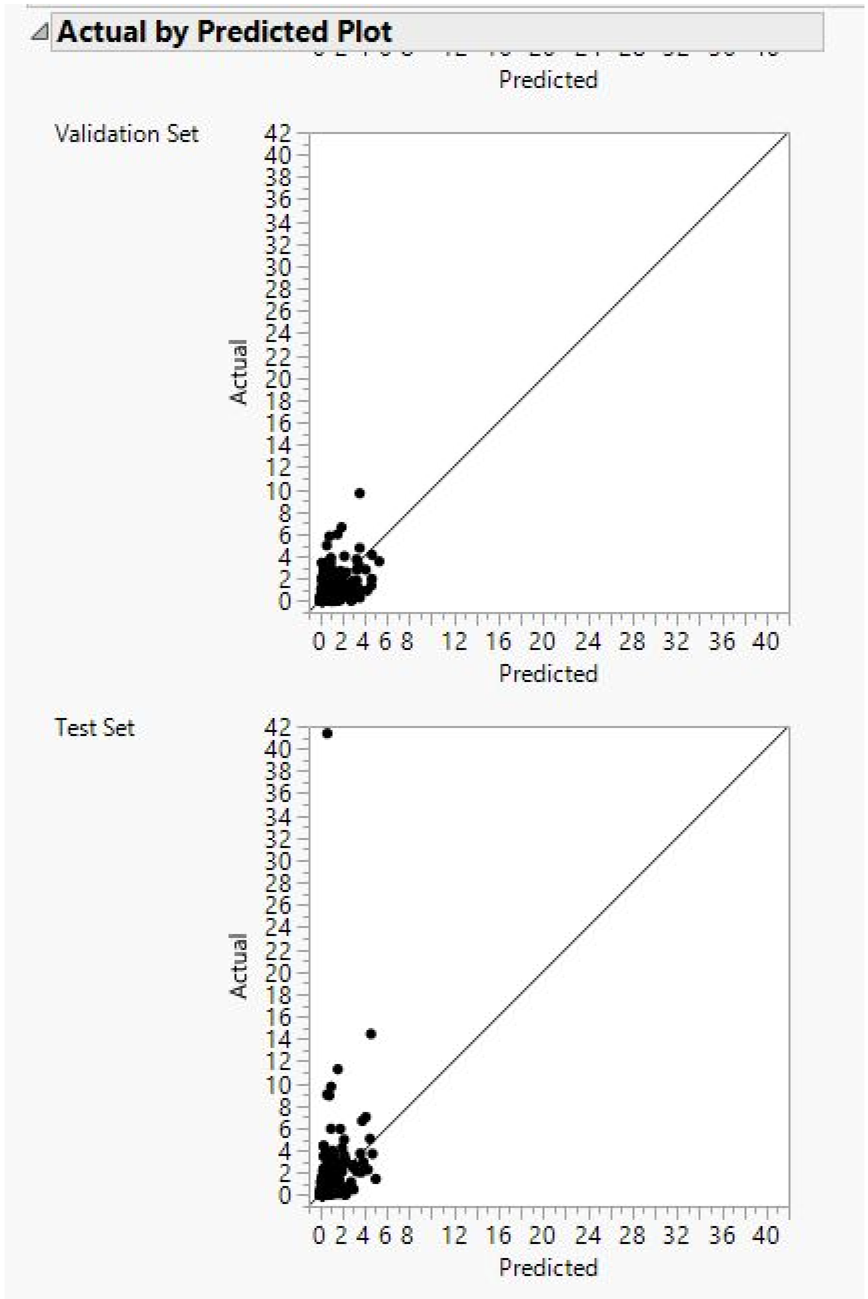


Figure 14 Bootstrap Prediction Result for Log(NA_Sales)

Bootstrap forest model yields R-square/RMSE 0.23/0.59 and 0.14/1.25 in validation and test set, respectively.

The Bootstrap has very low RSq values for the test data and is quite far from the validation data RSq values.

4. **K-NN**

When NA_Sales is the target variable, the model improves from having k=10 in the Training Data to k=8 in the Test data.

K Nearest Neighbors

NA_Sales

Training Set				Validation Set				Test Set			
K	Count	RMSE	SSE	K	Count	RMSE	SSE	K	Count	RMSE	SSE
1	3194	1.0580	3575.16	1	1592	0.90840	1313.71	1	1611	1.3750	3045.72
2	3194	0.8484	2299.07	2	1592	0.78595	983.405	2	1611	1.2904	2682.54
3	3194	0.8322	2211.95	3	1592	0.74174	875.893	3	1611	1.2853	2661.3
4	3194	0.8086	2088.17	4	1592	0.72057	826.599	4	1611	1.2804	2641.17
5	3194	0.8054	2071.96	5	1592	0.69777	775.115	5	1611	1.2685	2592.14
6	3194	0.8014	2051.41	6	1592	0.67784	731.479	6	1611	1.2525	2527.09
7	3194	0.8048	2068.79	7	1592	0.67612	727.774	7	1611	1.2454	2498.72
8	3194	0.7988	2037.97	8	1592	0.66082	695.208	8	1611	1.2115	2364.55 *
9	3194	0.7818	1952.33	9	1592	0.65087	674.427	9	1611	1.2194	2395.28
10	3194	0.7788	1937.03 *	10	1592	0.64051	653.129 *	10	1611	1.2165	2384.14

Figure 15 K-NN Result for Log(NA_Sales)

K-NN model yields RMSE 0.64 /1.21 in validation/test set ,respectively. k-NN model has not very good evidence to explain the variable significance and also the higher RMSE would not be suitable.

Model for categorical binned NA_Sales - Sales range prediction

Due to the non existence of binary output we test two models: KNN and Naive Bayes to predict the sales range that a particular game falls into.

1. **k- NN mode**l: This model was chosen as the first model to start the categorical prediction among other models that predicts categories as more than 2 categories exist in our data.

At k=8, the misclassification rate is the least, which is 40% in test set.

K Nearest Neighbors

Log[NA_Sales] Binned 2

Training Set

K	Count	Misclassification Rate	Misclassifications
1	3194	0.48090	1536
2	3194	0.49280	1574
3	3194	0.46556	1487
4	3194	0.45022	1438
5	3194	0.43456	1388
6	3194	0.43519	1390
7	3194	0.42642	1362
8	3194	0.42173	1347
9	3194	0.41703	1332 *
10	3194	0.42110	1345

Validation Set

K	Count	Misclassification Rate	Misclassifications
1	1592	0.51884	826
2	1592	0.49309	785
3	1592	0.47990	764
4	1592	0.46608	742
5	1592	0.45163	719
6	1592	0.44347	706 *
7	1592	0.44912	715
8	1592	0.44849	714
9	1592	0.44786	713
10	1592	0.44912	715

Test Set

K	Count	Misclassification Rate	Misclassifications
1	1611	0.47114	759
2	1611	0.47176	760
3	1611	0.45748	737
4	1611	0.45251	729
5	1611	0.44196	712
6	1611	0.42210	680
7	1611	0.41962	676
8	1611	0.40472	652 *
9	1611	0.41341	666
10	1611	0.40658	655

Confusion Matrix for Best K=6

Training Set

Actual Log[NA_Sales] Binned 2	Predicted Count -6 — -4	-4 — -2	-2 — 0	0 — 2	2 — 4
-6 — -4	3	55	34	4	0
-4 — -2	14	718	528	18	0
-2 — 0	3	443	1031	58	2
0 — 2	0	33	183	52	4
2 — 4	0	0	6	5	0

Validation Set

Actual Log[NA_Sales] Binned 2	Predicted Count -6 — -4	-4 — -2	-2 — 0	0 — 2	2 — 4
-6 — -4	2	31	18	1	0
-4 — -2	5	339	264	5	0
-2 — 0	3	213	505	33	1
0 — 2	0	19	112	40	0
2 — 4	0	0	1	0	0

Test Set

Actual Log[NA_Sales] Binned 2	Predicted Count -6 — -4	-4 — -2	-2 — 0	0 — 2	2 — 4
-6 — -4	2	36	16	3	0
-4 — -2	4	358	244	5	0
-2 — 0	3	226	541	30	0
0 — 2	0	21	84	30	2
2 — 4	0	0	4	2	0

Figure 16 K-NN Result for NA_Sales range

2. **NAÏVE BAYES**

Misclassification rate is 48% in test set.

Naive Bayes

Log[NA_Sales] Binned 2

	Training Set Count	Misclassification Rate	Misclassifications	Validation Set Count	Misclassification Rate	Misclassifications	Test Set Count	Misclassification Rate	Misclassifications
	3194	0.47026	1502	1592	0.48618	774	1611	0.48479	781

Confusion Matrix

Training Set

Actual Log[NA_Sales] Binned 2	Predicted Count -6 — -4	-4 — -2	-2 — 0	0 — 2	2 — 4
-6 — -4	8	58	30	0	0
-4 — -2	7	562	709	0	0
-2 — 0	1	414	1122	0	0
0 — 2	0	66	206	0	0
2 — 4	0	2	9	0	0

Validation Set

Actual Log[NA_Sales] Binned 2	Predicted Count -6 — -4	-4 — -2	-2 — 0	0 — 2	2 — 4
-6 — -4	3	32	17	0	0
-4 — -2	4	257	352	0	0
-2 — 0	1	197	557	0	0
0 — 2	1	47	122	1	0
2 — 4	0	0	1	0	0

Test Set

Actual Log[NA_Sales] Binned 2	Predicted Count -6 — -4	-4 — -2	-2 — 0	0 — 2	2 — 4
-6 — -4	1	40	16	0	0
-4 — -2	3	253	355	0	0
-2 — 0	4	220	576	0	0
0 — 2	0	34	103	0	0
2 — 4	0	0	6	0	0

Figure 17 NAÏVE BAYES Result for NA_Sales range

Comparing the two models above, we determine K-NN model is better predicting the sales range. Since we predicted the Log(NA sales), it is simple to transform back to normal NA sales : NA sales = exp^(log(NA_sales)

Model for categorical value " Popularity" prediction

NAÏVE BAYES

In order to classify if a game is popular or not we create a new column formula where the interaction term "Critic Score*Critic Count" is defined more than 2400 in order for it to be popular.

The output is either 0 (if not popular) or 1(if popular).

2400 is chosen as a metric to define the popularity by taking into the consideration the critic score even when the critic count is very less. In this model, we focus more on the accuracy of 0’s as the focus is on those games which require more attention in terms of marketing and promotion to increase it’s popularity.

For Naïve Bayes, we only want to know how the categorical variables Platform, Genre and Rating predict the popularity.

Naive Bayes

popularity

	Training Set			Validation Set			Test Set	
Count	Misclassification Rate	Misclassifications	Count	Misclassification Rate	Misclassifications	Count	Misclassification Rate	Misclassifications
3474	0.29822	1036	1737	0.30570	531	1736	0.31624	549

Confusion Matrix

Training Set

Actual popularity	Predicted Count 0	1
0	1972	343
1	693	466

Validation Set

Actual popularity	Predicted Count 0	1
0	993	176
1	355	213

Test Set

Actual popularity	Predicted Count 0	1
0	947	182
1	367	240

Figure 18 NAÏVE BAYES Result for Popularity Prediction

Misclassification Rate: 31.6%

Accuracy of 0’s: 947/1314 = 72.07%

Naïve Bayes was specifically used as the need is to evaluate the popularity only using the categorical variables Platform, Genre and Rating.

ASSESS

COMPARISON OF RESIDUALS (for NA_Sales)

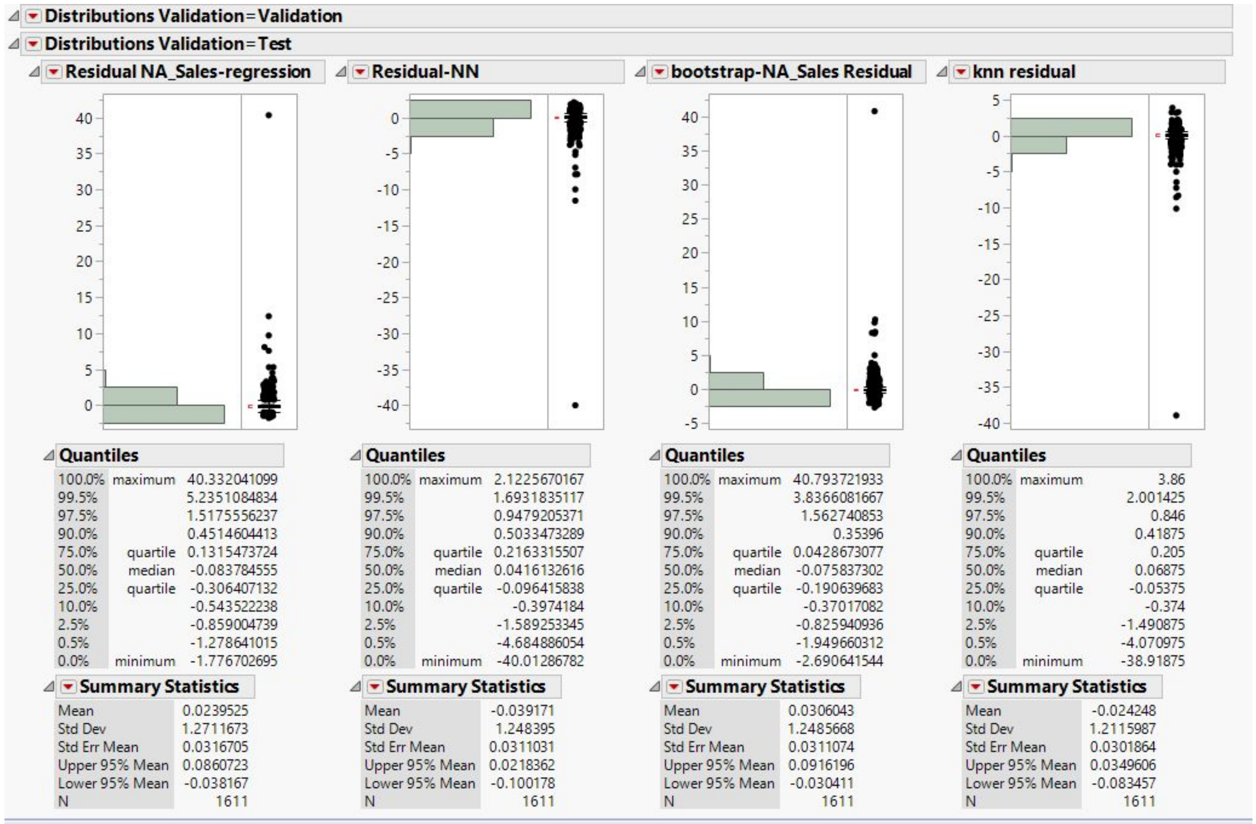


Figure 19 Residual Comparison for NA_Sales

In order to understand the distribution properly, the same models were built to predict the Log[NA_Sales] and the residuals for each model displayed the below distribution.

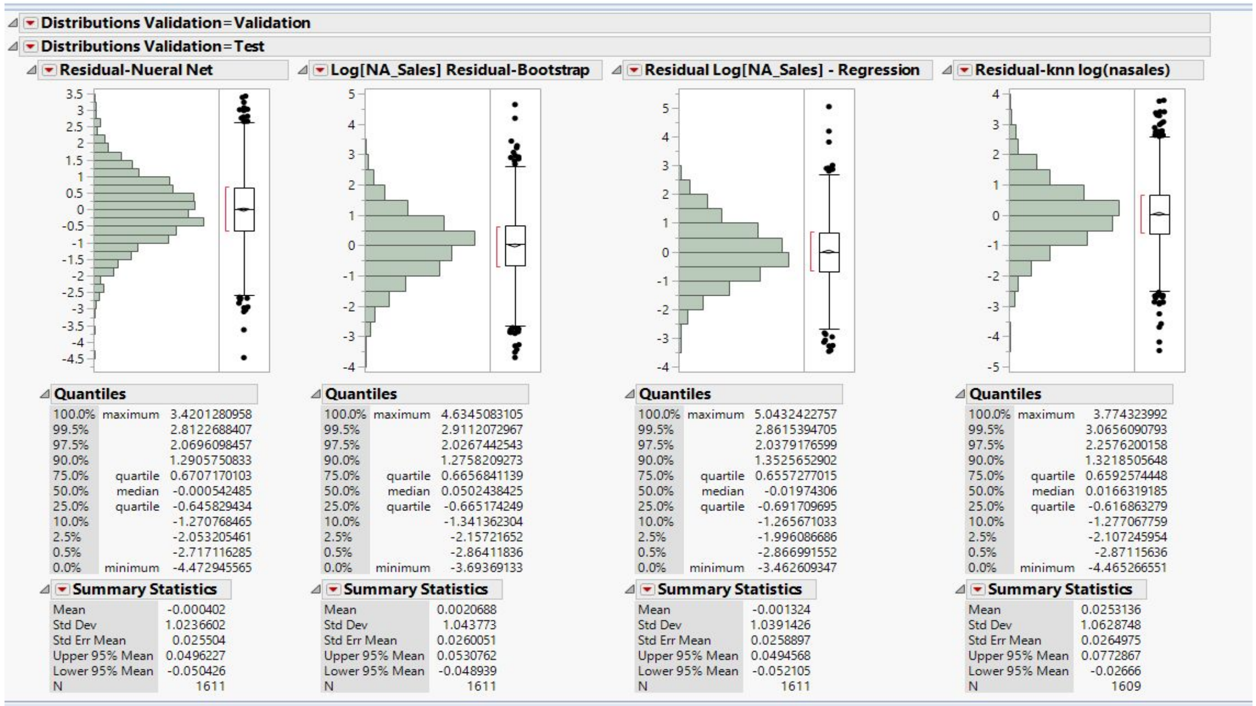


Figure 20 Residual Comparison for Log(NA_Sales)

RESULTS

We chose linear regression model for prediction of NA_Sales and KNN model for the sales potential range and Naïve bayes for popularity prediction.

1. Prediction of NAsales

We chose the regression model for prediction of NAsales. Although the residuals of the four models are similar, the regression model has a highest RSq than the others'. And in business context, we know that the predictors for regression are reasonable and good for prediction. So we choose linear regression model for prediction of NAsales.

The average error of regression is slightly more compared to Neural Networks and Bootstrap model but since the explainability is high in regression this was picked. Coming to Bootstrap, the inclusion of trees would make the model complicated and also when compared to the minimum Sales it would predict wrong when compared to regression model, it was high.

2. The NAsales potential range

We did two different models, KNN and NAÏVE BAYES.

We chose KNN for its lower misclassification rate than NAÏVE BAYES, which is 0.422<0.484.

3. Popularity prediction

We believe that NAÏVE BAYES is a good model to predict popularity because in popularity, we focus on the "0", which means not popular. We have to make some changes to the not popular ones and the zero accuracy of our model for prediction is 0.7207.

CONCLUSION

- Looking at the results, we can conclude that the model to predict NA_Sales was chosen to be a Regression model. One of the key takeaways from our analysis was that, even though the RSq and RMSE values did not fall in the ideal range that one would expect, the basic Business knowledge in retail/ video game industry assured the predictors used were significant.

- Critic Score and Critic count remain the most important predictors as we know, just like in movies, attract customers when a person with high expertise in that field reviews the product. Using both these variables as interaction term improved the model as there could be a scenario where there are less number of critics who have given high scores or high number of critics who have given low scores.

- Genre significantly contributes towards sales as there would be existing customer base who are regular users, they might belong to certain age group. Similarly, certain names in Platform would have gained popularity which could affect the Sales.

- Rating would basically affect the population of certain age group that has access to games meant for them.

- We binned the NA_sales to ranges in order to find the games that would fall in a certain group. This would help the marketing team to improve promotions and campaigns for those games which generate low sales. The binning was done in such a way that used the product of critic core and critic count and a optimal threshold was fixed based on research of the ways games are scored. The results were validated by checking the popularity of random games.

REFERENCES

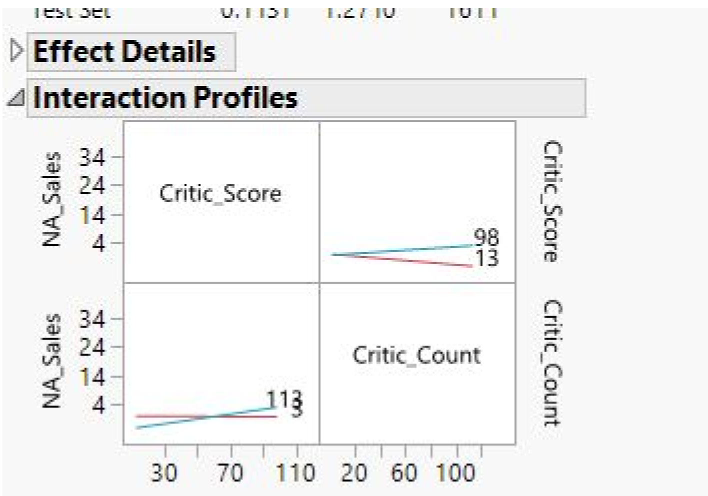
1) https://www.researchgate.net/post/Whats_the_best_model_to_predict_a_categorical_outcome_with_41_levels_in_R
2) www.google.com
3) https://www.sas.com/jmpstore/products-solutions/cSoftware-p1.html
4) Data Mining for Business Analytics

APPENDIX

The screenshots of all the models that were tried.

NA_Sales Prediction

Interaction Plot



Neural Nets

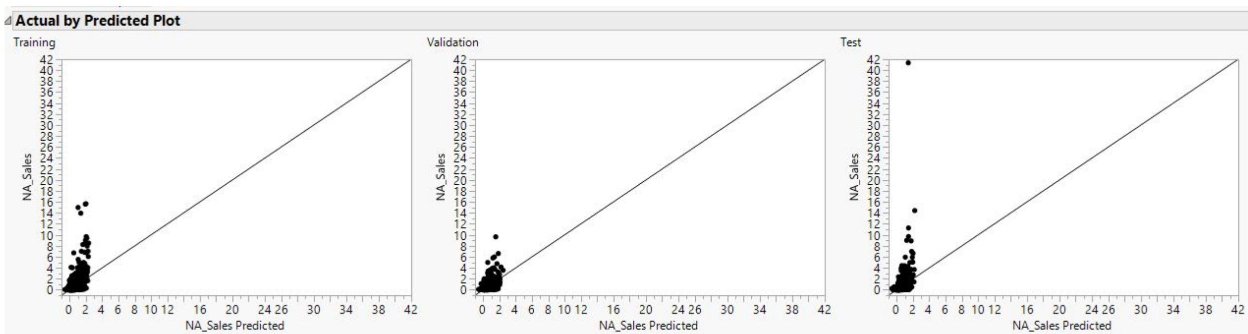
Neural

Validation Column: Validation

Model Launch

Model NTanH(2)NLinear(2)NGaussian(1)NBoost(3)

Training: NA_Sales Measures	Value	Validation: NA_Sales Measures	Value	Test: NA_Sales Measures	Value
RSquare	0.2524907	RSquare	0.2309406	RSquare	0.1341635
RMSE	0.7910495	RMSE	0.5877597	RMSE	1.2558456
Mean Abs Dev	0.3609809	Mean Abs Dev	0.349274	Mean Abs Dev	0.3930248
-LogLikelihood	3783.4329	-LogLikelihood	1412.9023	-LogLikelihood	2652.9105
SSE	1998.6753	SSE	549.97464	SSE	2540.7857
Sum Freq	3194	Sum Freq	1592	Sum Freq	1611



Other required screenshots are included in the "MODEL" section.

The work contained and presented here is our team's work and our work alone