Multiple Linear Regression of Video Game Sales

Consumers have many choices when seeking a particular good or service. This is certainly true among video gamers, who often have preferences regarding their favorite video game console, developer, and genre. The gaming industry, nearing close to $100 billion just last year, seeks to capture the attention of gamers with ever changing interests. Observing market trends to capture unsaturated markets within the industry is important for businesses to maintain revenue growth. The analysis below will shed light on the characteristics that determine the success, or lack thereof, of a given video game.

Problem Statement

Develop a regression model based on an observed set of explanatory variables that can be utilized to predict future yearly global sales of a video game.

Constraints and Limitations

The analysis was completed on a Kaggle data set which was collected via web scraping of VGChartz, and Metacritic, two websites that provide tracking and analysis of video game sales. Since the data set is a compilation of observational data, causal inferences cannot be drawn between the explanatory and response variables. Moreover, the data set contains missing observations due to the fact that Metacritic covers a limited segment of video game platforms. The conducted analysis pertains to only complete observations, which may lend bias to certain explanatory variables having more weight than they should in forecasting sales. However, this concern is eased by the randomness of the missing data. Additionally, it is possible that the data set does not include other explanatory variables which more accurately predict the sales of a given video game.

Data Set Description

The data covers 16,719 unique video game titles that were released as far back as 1980, to even games that have yet to be released up until 2020. Subsetting the data to complete observations reduces the number of observations to 6,826 distinct titles. Before delving into the descriptions of the variables, it is helpful to gain a glimpse of the data set:

Head of Video Game Sales Data

Obs	Name	Platform	Year_of_Release	Genre	Publisher	NA_Sales	EU_Sales	JP_Sales	Other_Sales	Global_Sales	Critic_Score	Critic_Count	User_Score	User_Count	Developer	Rating
1	Wii Sports	Wii	2006	Sports	Nintendo	41.36	28.96	3.77	8.45	82.53	76	51	8	322	Nintendo	E
2	Super Mario Bros.	NES	1985	Platform	Nintendo	29.08	3.58	6.81	0.77	40.24	.	.	.	.		
3	Mario Kart Wii	Wii	2008	Racing	Nintendo	15.68	12.76	3.79	3.29	35.52	82	73	8.3	709	Nintendo	E
4	Wii Sports Resort	Wii	2009	Sports	Nintendo	15.61	10.93	3.28	2.95	32.77	80	73	8	192	Nintendo	E
5	Pokemon Red/Pokemon Blue	GB	1996	Role-Playing	Nintendo	11.27	8.89	10.22	1	31.37	.	.	.	.		
6	Tetris	GB	1989	Puzzle	Nintendo	23.2	2.26	4.22	0.58	30.26	.	.	.	.		
7	New Super Mario Bros.	DS	2006	Platform	Nintendo	11.28	9.14	6.5	2.88	29.8	89	65	8.5	431	Nintendo	E
8	Wii Play	Wii	2006	Misc	Nintendo	13.96	9.18	2.93	2.84	28.92	58	41	6.6	129	Nintendo	E
9	New Super Mario Bros. Wii	Wii	2009	Platform	Nintendo	14.44	6.94	4.7	2.24	28.32	87	80	8.4	594	Nintendo	E
10	Duck Hunt	NES	1984	Shooter	Nintendo	26.93	0.63	0.28	0.47	28.31	.	.	.	.		

Figure 1: Head of Data Set

Variable Name	Variable Type	Summary
Name	Character	Title of the game
Platform	Character	The console of the game
Year_of_Release	Numeric	The year the game was released, ranging from 1980-2020, with a mean of 2007
Genre	Character	The category or play style of the game

Publisher	Character	The entity responsible for the manufacturing and marketing of the game.
NA_Sales	Numeric	The cumulative sales in millions from North America
EU_Sales	Numeric	The cumulative sales in millions from Europe
JP_Sales	Numeric	The cumulative sales in millions from Japan
Other_Sales	Numeric	The cumulative sales in millions from other regions
Global_Sales	Numeric	The cumulative global sales in millions
Critic_Score	Numeric	Average score compiled by Metacritic's staff
Critic_Count	Numeric	The number of critics used to calculate the critic score
User_Score	Numeric	Average score by Metacritic's subscribers
User_Count	Numeric	The number of subscribers used to calculate the user score
Developer	Character	Party responsible for creating the game
Rating	Character	The ESRB rating of the game

Exploratory Data Analysis

As one can see from Figure 2, logarithmic transformation of the data set was necessary. After transformation, the data now resembles a more normal distribution, which allows for the possibility of regression analysis.

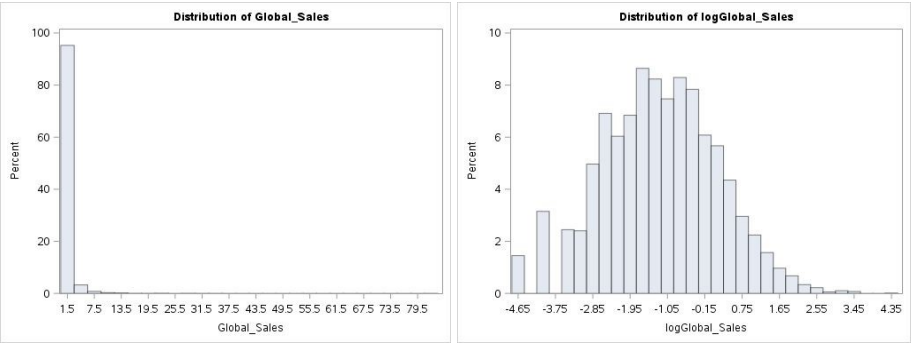

Figure 2: Histograms of Global Sales, before and after transformation

With a median of 0.29, mean of 0.777499, and a range of 82.52 global sales (in millions) is extremely right-skewed before transformation. This is in part due to Wii Sports with a global sales figure of 82.52 million. This observation is much larger compared to other video games because of Nintendo's decision to bundle Wii Sports with the sale of the Wii itself.

By plotting global sales of video games by genre per year of the title's release, we can see in Figure 3A that action games developed a strong foothold around 2002, and has continued its

dominance of market share throughout the years even with an overall market downtrend in 2010.

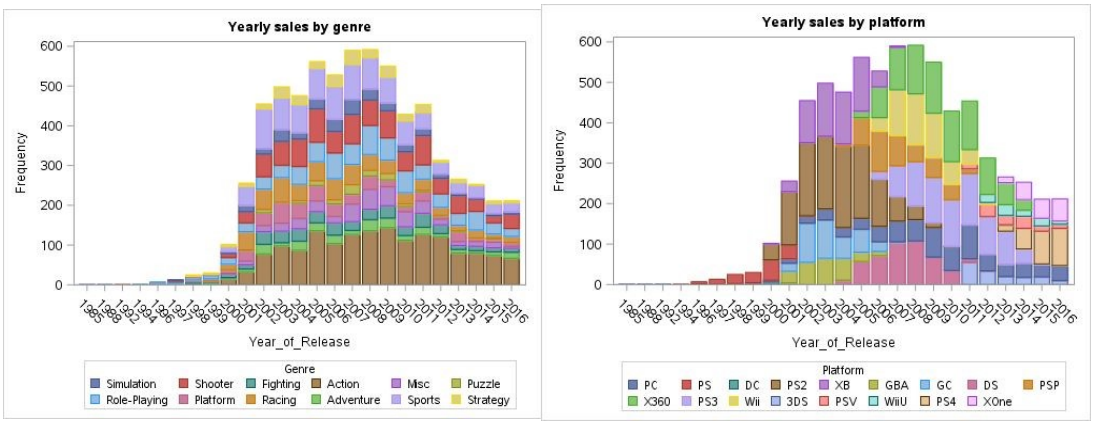

Figure 3: Histograms of a.) Sales per year of release by genre b.) Sales per year of release by platform

Continuing the exploratory data analysis, it is visually apparent in Figure 3B that there have been many different platforms throughout the years. At first glance it would appear that PlayStation experienced a monopoly during the late 1990's, but we must keep in mind the limitations of the data set. The Nintendo N64 was released in 1996, yet does not appear on the graph because of how the data set was collected.

However, this does not mean the graph is incapable of providing insight. The transition of sales from an older generation console to the next generation console is exemplified by the initially small market share of Xbox 360 sales that gradually consumed and reversed proportions with its older counterpart, the Xbox.

Plotting the sales of video games per genre by platform in Figure 4 sheds light on the nature of each platform's consumers. For example, shooting games make up a significant portion of Sony's 2nd and 3rd generation of PlayStation. Likewise, this is also true for Microsoft's Xbox, whereas Nintendo's Wii consoles have instead a proportionally significantly larger amount of games categorized as miscellaneous. One possible explanation could be Nintendo's tendencies to develop family and social oriented games, such as Mario Party, which fails to qualify for any other genre besides miscellaneous.

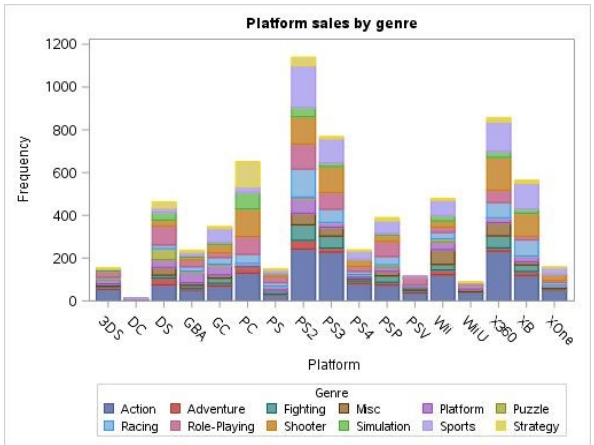

Figure 4: Histogram of sales per genre by platform

Variable Screening

A heat map correlation matrix was implemented over the standard SAS scatterplot matrix simply due to the limitation of the graphical size of the plot while maintaining label integrity. Additionally, we can discern the correlation between variables with much greater easy in this manner. For example, Figure 5 shows that sales in North America and Europe have a strong positive correlation with overall global sales, whereas sales in Japan is only weakly positively correlated with global sales. Users' scores of a given video game appears to have a moderately negative correlation with the game's release date, which gives some credence to the belief that "they don't make them like they used to anymore."

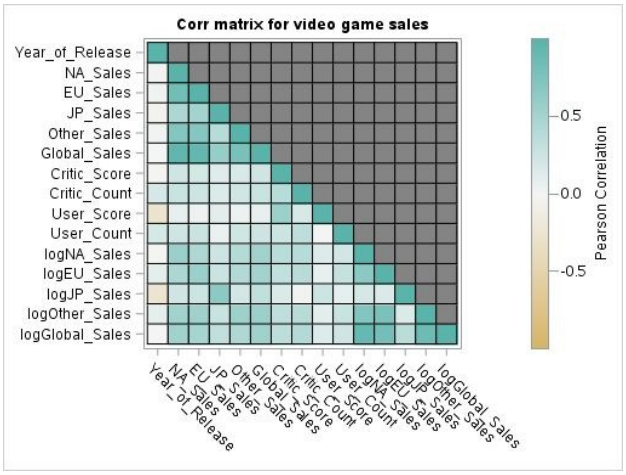

Figure 5: Heat map correlation matrix

Global sales is comprised as the aggregate sum of all other sales variables. Consequently, to reduce redundancy, our model will only include global sales as the single response variable. Among all the numeric explanatory variables shown in Figure 6, the Variance Inflation Factor is fairly low. Unfortunately, dummy coding most of the categorical variables is unfeasible due to the sheer number of developers and so on. Although there is likely to be some collinearity between genre and rating, it is unlikely to be enough to justify removing either from the model. The scope of this project is to most accurately predict global sales of a video game, rather than which variables most accurately explain the response variable. With this in mind, all explanatory variables will be kept in the model, and regional sales will be implicitly included as global sales are modelled.

Parameter Estimates

Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|	Variance Inflation
Intercept	1	61.47811	7.64632	8.04	<.0001	0
Year_of_Release	1	-0.03223	0.00380	-8.48	<.0001	1.17453
Critic_Score	1	0.02589	0.00143	18.05	<.0001	1.81351
Critic_Count	1	0.02427	0.00089078	27.24	<.0001	1.34405
User_Score	1	-0.07897	0.01334	-5.92	<.0001	1.69383
User_Count	1	0.00017723	0.00002785	6.36	<.0001	1.22648

Figure 6: Variance Inflation Factor of numeric variables

Model Selection

Analysis of worldwide video game sales can now be carried out with the following model:

$$Log(Global\ Sales) = \beta_0 + \beta Platform + \beta Year\ of\ Release + \beta Genre + \beta Publisher + \beta Critic_{Score} + \beta Critic_{Count} + \beta User_{Score} + \beta User_{Count} + \beta Developer + \beta Rating$$

To avoid overfitting the model a computational variable selection method will be utilized. Specifically, Least Absolute Shrinkage and Selection (LASSO) will discern from all the explanatory variables which can be dropped due to statistical insignificance. The output in Figure 7 displays the explanatory variables that LASSO selection kept. Local minimum of the SBC criterion was found at Platform_PS2 with an SBC value of 1523.2218 being greater than the comparison SBC value of 1515.5290, upon which variable selection for the model was halted. 37.02% of the variation of global video game sales is explained by the explanatory variables in Figure 7. Performing a general linear modelling with no explanatory variables excluded resulted in a R-square value of 0.701744, which would indicate that the model was indeed overfitting the response variable.

Analysis of Variance				
Source	DF	Sum of Squares	Mean Square	F Value
Model	10	4958.39532	495.83953	402.16
Error	6815	8402.54686	1.23295	
Corrected Total	6825	13361		

Root MSE	1.11038
Dependent Mean	-1.23349
R-Square	0.3711
Adj R-Sq	0.3702
AIC	8268.41554
AICC	8268.46133
SBC	1515.52898

Parameter Estimates		
Parameter	DF	Estimate
Intercept	1	-3.008015
Platform_PC	1	-1.302380
Platform_Wii	1	0.018077
Platform_XB	1	-0.161491
Publisher_Electronic Arts	1	0.210444
Publisher_Nintendo	1	0.220964
Developer_Nintendo	1	0.332513
Genre_Strategy	1	-0.040385
Critic_Score	1	0.019357
Critic_Count	1	0.016133
User_Count	1	0.000251

Figure 7: ANOVA of model

In order to confidently use this regression model, we must first validate its residuals against assumptions of normality. The histogram in Figure 8 reflects a standard bell curve one would expect from a normally distributed data set. Moreover, the Q-Q plot does a moderately good job a aligning with a constant diagonal linearity, even though there is some presence of left tail skewness. While the residual scatter plot has some visual stratification, there is no evidence to suggest that the data set is nonlinear.

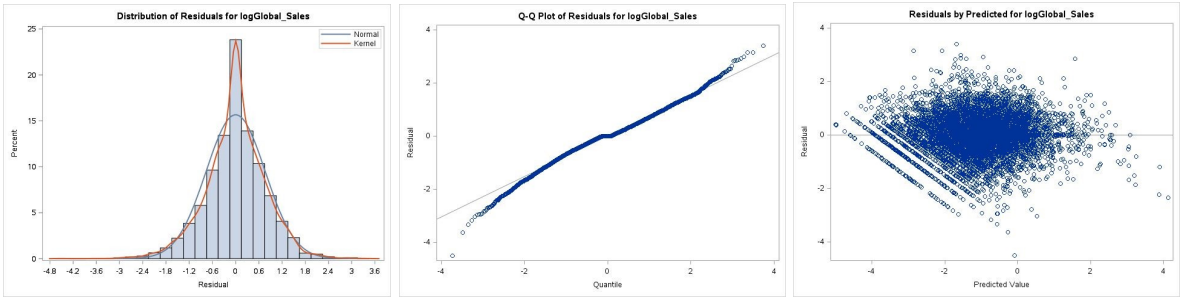

Figure 8: Histogram, Q-Q Plot, and Scatter Plot of Residuals

With the assumptions of normality met, we will turn our attention towards locating any presence of outliers or high leverage points. The Figure 9A identifies numerous outliers and leverage points, as well as a handful of observations that qualify as both. Leverage

points themselves are not usually a case for concern unless they are also outliers. Fortunately, the observations identified as both an outlier and a leverage point appear to be relatively close to the general distribution of the other residuals.

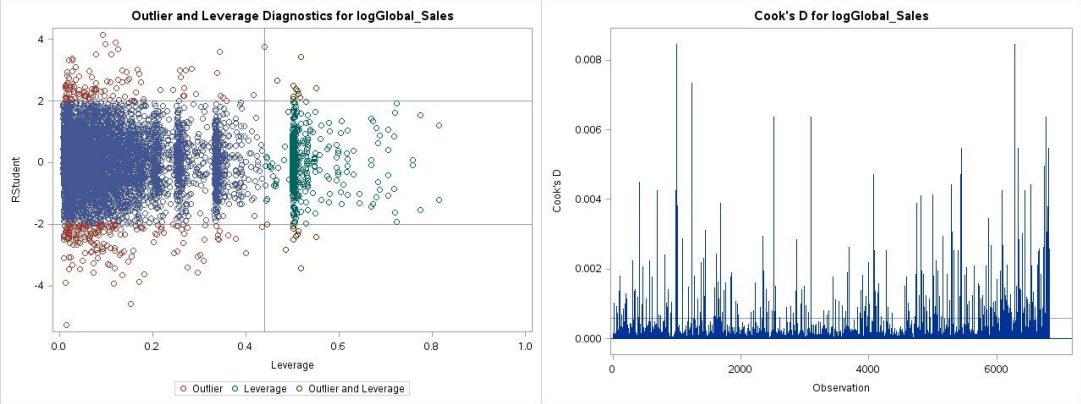

Figure 9: a.) Studentized Residuals and b.) Cook's Distance of Log(Global Sales)

Under the standard procedure, all Cook's D values in Figure 9B are well under the cut off mark of 1. Alternatively, using $D_i > 4 / n$ as a benchmark for identifying highly influential points leads to a threshold of 0.000586, which the majority of the observations seem to fall under. However, strictly relying on either threshold mark can lead to too much rigidity when attempting to discern which data points are highly influential. Ultimately some of Cook's D values in Figure 9B are relatively high, but when viewed in the conjunction with the studentized residuals, we can see that no single observation is extremely far from the general distribution of studentized residuals.

With assumptions of normality, linearity, and constant variance met, combined with no presence of extreme high leverage point outliers, the model can be employed to predict future global sales of video games. After variable selection, the model is statistically significant at an alpha level of 5% (n = 6825, F-value = 402.16, P-value < 0.0001). Ultimately the regression analysis predicts a model in which median sales are expected increase by: 0.2718839 million if on the Personal Computer platform, 1.0182414 million if on the Wii platform, 0.85087351 million if on the Xbox platform, 1.234226 million if published by Electronic Arts, 1.247279 million if published by Nintendo, 1.39447 million if developed by Nintendo, 0.9604196 million if a strategy game, and 1.0195456 for a Critic Score of 100%. Likewise, median sales are predicted to increase by 5.01935 million if 100 critics review the given video game, and 1.0254177 million if 100 users review the given video game on Metacritic.

$$Global\ Sales = 0.04938962 + e^{-1.302380*(PlatformPC)} + e^{0.018077*(PlatformWii)} + e^{-0.1614918*(PlatformXB)} + e^{0.210444*(PublisherEA)} + e^{0.220964*(PublisherNintendo)} + e^{0.332513*(DeveloperNintendo)} + e^{-0.040385*(GenreStrategy)} + e^{0.019357*(CriticScore)} + e^{0.016133*(CriticCount)} + e^{0.000251*(\mathrm{UserCount})}$$

Serial Correlation

6,826 observations are more than sufficient justify the use of the Durbin-Watson test for serial correlation. The analysis has a lag of 1 to determine if there is autocorrelation within the data. Figure 10 reflects a Durbin-Watson value of 0.4472, which is near to zero, indicating a presence of positive serial correlation.

The AUTOREG Procedure

Ordinary Least Squares Estimates			
SSE	10154.2837	DFE	6820
MSE	1.48890	Root MSE	1.22020
SBC	22135.3121	AIC	22094.3412
MAE	0.95761525	AICC	22094.3535
MAPE	221.078815	HQC	22108.477
Durbin-Watson	0.4618	Total R-Square	0.2400

Durbin-Watson Statistics			
Order	DW	Pr < DW	Pr > DW
1	0.4618	<.0001	1.0000

Figure 10: Auto Regression Output and Durbin-Watson Test

The result of the Durbin-Watson test is further solidified upon examining the Partial Autocorrelation Function (PACF) plot in Figure 11. The residuals of the PACF initially is extremely outside the 2-standard error area of the plot, and then gradually decreases towards zero without ever becoming negative. This behavior is a textbook description of positive autocorrelation, and can potentially be explained by both monetary inflation, and the general increased popularity of video games beginning in the early 1990's to 2000's.

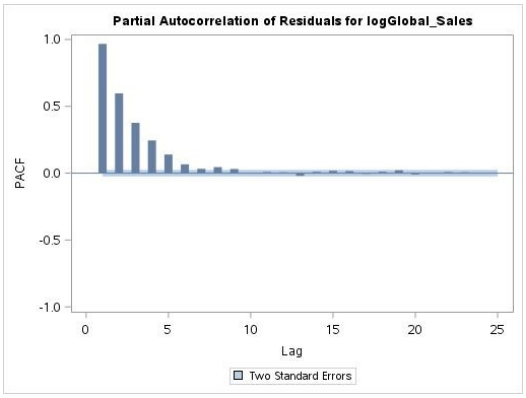

Figure 11: Partial Autocorrelation Function

Conclusion

The linear regression model developed by this analysis ascertained that not all explanatory variables present in the original data set were statistically significant at the $\alpha = 0.05$ level for predicting global sales of a given video game. The model explains 37.02% ($R^2 = 0.3702$) of variation among global sales. While this R-square value is lower than expected, it is still of worth in providing context for further analysis.

In particular, additional research into forecasting video game sales would do well in focusing on developing regional models, as the statistically significant explanatory variables might change region-to-region. On the other hand, the original data set could simply not be collecting the pertinent information for explaining the level of sales for a given video game.

Ways to improve the analysis include imputation of missing data. Imputation would require dummy coding the categorical variables in the original data set, some of which contain upwards of a thousand unique string values. Additionally, random forest techniques could be implemented increase the accuracy of the regression model. However, this is currently outside the scope our knowledge base at this point in time in the data science program.

Appendix

```
/*
 * Import the video game sales with ratings data set.
 * Data set is sourced from Kaggle:
 * https://www.kaggle.com/rush4ratio/video-game-sales-with-ratings
 */
proc import out=sales
datafile='/home/joebarrystraume0/6372/project1/Video_Games_Sales_as_at_22_Dec_2016.csv'
dbms=csv;
getnames=yes;
run;

/*
 * Check out the head of the data set.
 */
proc print data=sales (firstobs=1 obs=10);
title 'Head of Video Game Sales Data';
run;

/*
 * Get a general snapshot of the data set.
 */
proc contents data=sales;
title 'Snapshot of video game sales data set';
run;

/*
 * Create a correlations matrix heat map.
 * This portion of code is repurposed from the following SAS blog:
 * https://blogs.sas.com/content/sasdummy/2013/06/12/correlations-matrix-heatmap-with-sas/
 */
ods path work.mystore(update) sashelp.tmplmst(read);

proc template;
        define statgraph corrHeatmap;
  dynamic _Title;
                begingraph;
      entrytitle _Title;
                        rangeattrmap name='map';
                        /* select a series of colors that represent a "diverging"  */
                        /* range of values: stronger on the ends, weaker in middle */
                        /* Get ideas from http://colorbrewer.org                   */
                        range -1 - 1 / rangecolormodel=(cxD8B365 cxF5F5F5 cx5AB4AC);
                        endrangeattrmap;
                        rangeattrvar var=r attrvar=r attrmap='map';
                        layout overlay /
                                xaxisopts=(display=(line ticks tickvalues))
                                yaxisopts=(display=(line ticks tickvalues));
                                heatmapparm x = x y = y colorresponse = r / xbinaxis=false ybinaxis=false
                                        colormodel=THREECOLORRAMP name = "heatmap" display=all;
```

```
                                continuouslegend "heatmap" /
                                        orient = vertical location = outside title="Pearson Correlation";
                        endlayout;
                endgraph;
        end;
run;

/* Prepare the correlations coeff matrix: Pearson's r method */
%macro prepCorrData(in=,out=);
        /* Run corr matrix for input data, all numeric vars */
        proc corr data=&in. noprint
                pearson
                outp=work._tmpCorr
                vardef=df
        ;
        run;

/* prep data for heatmap */
data &out.;
        keep x y r;
        set work._tmpCorr(where=(_TYPE_="CORR"));
        array v{*} _numeric_;
        x = _NAME_;
        do i = dim(v) to 1 by -1;
                y = vname(v(i));
                r = v(i);
                /* creates a diagonally sparse matrix */
                if (i<_n_) then
                        r=.;
                output;
        end;
run;

proc datasets lib=work nolist nowarn;
        delete _tmpcorr;
quit;
%mend;

/* Build the graphs */
ods graphics /height=600 width=800 imagemap;

%prepCorrData(in=sales,out=sales_r);
proc sgrender data=sales_r template=corrHeatmap;
  dynamic _title="Corr matrix for video game sales";
run;

/*
 * Inspect the character variables of platform and genre
 * and their respective value frequencies.
 */
```

```
proc freq data=sales;
tables Platform Genre Rating / nocum;
title 'Freq table for platform, genre, and rating variables';
run;

/*
 * Subset the original data set
 * to include only complete observations.
 */
data completeSales;
set sales;
if cmiss(of _all_)=0; /* complete cases for all vars */
logNA_Sales = log(NA_Sales);
logEU_Sales = log(EU_Sales);
logJP_Sales = log(JP_Sales);
logOther_Sales = log(Other_Sales);
logGlobal_Sales = log(Global_Sales);
run;

/*
 * Snapshot of updated data set.
 */
proc contents data=completeSales;
title 'Snapshot of data set with complete observations';
run;

proc univariate data=completeSales;
run;

/*
 * Request a variety of statistics for summarizing
 * the data distribution of each analysis variable.
 */
proc univariate data=completeSales;
title 'Univariate output for each variable';
run;

/*
 * New correlation heat map matrix for updated data set.
 */
%prepCorrData(in=completeSales,out=completeSales_r);
proc sgrender data=completeSales_r template=corrHeatmap;
   dynamic _title="Corr matrix for video game sales";
run;

proc sgscatter data = completeSales;
matrix _numeric_ / diagonal=(Histogram);
title 'Scatter Matrix of Sales Data Set';
run;
```

```
/*
 * Plot sales of each year by genre.
 */
proc sgplot data=completeSales;
vbar Year_of_Release / group=Genre;
title 'Yearly sales by genre';
run;

/*
 * Plot sales of each year by platform.
 */
proc sgplot data=completeSales;
vbar Year_of_Release / group=Platform;
title 'Yearly sales by platform';
run;

/*
 * Plot sales by platform and genre.
 */
proc sgplot data=completeSales;
vbar Platform / group=Genre;
title 'Platform sales by genre';
run;

proc sgplot data=completeSales;
scatter y=logGlobal_Sales x=Year_of_Release;
title 'Distribution of global sales by year';
run;

proc univariate data=completeSales;
histogram logGlobal_Sales;
run;

proc sgplot data=completeSales;
histogram;

proc anova data=completesales;
class Platform Publisher Developer Genre Rating;
model logGlobal_Sales = Platform Publisher Developer Genre Rating Year_of_Release Critic_Score
Critic_Count User_Score User_Count;
run;

proc glm data=completeSales PLOTS(UNPACK MAXPOINTS=10000)=DIAGNOSTICS;
class Platform Publisher Developer Genre Rating;
model logGlobal_Sales = Platform Publisher Developer Genre Rating Year_of_Release Critic_Score
Critic_Count User_Score User_Count;
run;

proc glmselect data=completeSales;
class Platform Publisher Developer Genre Rating;
```

```
model logGlobal_Sales = Platform Publisher Developer Genre
Rating Year_of_Release Critic_Score Critic_Count User_Score User_Count /selection=LASSO;
run;

proc reg data=completeSales;
model logGlobal_Sales = Year_of_Release Critic_Score Critic_Count User_Score User_Count / vif;
run;

proc autoreg data=completeSales plots(unpack)=(all acf pacf);
model logGlobal_Sales = Year_of_Release Critic_Score Critic_Count User_Score User_Count / nlag=1
dwprob;
run; quit;
```