**Predictive Analytics using SAS – Homework 3 Report**

**Question 1: -**

1. The dataset “videogamesales\_main.sas7bdat” was loaded into SAS in a directory named HW3. It consisted of 4413 observations and 12 variables. These 12 variables were a combination of both numeric and categorical variables as follows,

A screenshot of a computer

Description automatically generated

Following this, we were asked to run a regression model that linked the global sales to video game reviews. Initially, I started out with the numeric variables as predictors in my model and in order to improve the fit of the model, I started including categorical variables too.

However, since I was using PROC REG to run my regression and get detailed diagnostics for my model, I had to create dummy (indicator) variables for the various levels of the respective categorical variables in order to incorporate them into my model. The reasons why I added such numeric and categorical variables in my model are explained in detail in (b).

**Final Model –**

Global \_Sales = critic\_score+ critic\_count+ user\_score+ user\_count+ critic+ user+ r1+ r2+ r3+ r4 + g1 + g2+ g3+ g4+ g5+ g6+ g7+ g8+ + g10+ g11+ g12+ p1+ p2+ p3+ p4+ p5+ p6+ p7+ p8+ p9 + p10+ r1\_ucount+ r2\_ucount + r3\_ucount + r4\_ucount + r1\_ccount + r2\_ccount +r3\_ccount + r4\_ccount + p1\_ucount+ p2\_ucount + p3\_ucount + p4\_ucount + p5\_ucount + p6\_ucount + p7\_ucount + p8\_ucount + p9\_ucount + p10\_ucount + p1\_ccount+ p2\_ccount + p3\_ccount + p4\_ccount + p5\_ccount + p6\_ccount + p7\_ccount + p8\_ccount + p9\_ccount + p10\_ccount + g1\_uscore+ g2\_uscore + g3\_uscore + g4\_uscore + g5\_uscore + g6\_uscore + g7\_uscore + g8\_uscore + g9\_uscore + g10\_uscore + g11\_uscore + g12\_uscore + g1\_cscore+ g2\_cscore + g3\_cscore + g4\_cscore + g5\_cscore + g6\_cscore + g7\_cscore + g8\_cscore + g9\_cscore + g10\_cscore + g11\_cscore + g12\_cscore

where,

critic: Interaction term between ‘critic\_score’ and ‘critic\_count’

user: Interaction term between ‘user\_score’ and ‘user\_count’

r1 – r4: Dummy (indicator) variable for ‘rating’

g1 – g12: Dummy (indicator) variable for ‘genre’

p1 – p10: Dummy (indicator) variable for ‘platform’

r1\_ucount – r4\_ucount: Interaction term between ‘rating’ and ‘user\_count’

r1\_ccount – r4\_ccount: Interaction term between ‘rating’ and ‘critic\_count’

p1\_ucount – p10\_ucount: Interaction term between ‘platform’ and ‘user\_count’

p1\_ccount – p10\_ccount: Interaction term between ‘platform’ and ‘critic\_count’

g1\_ucount – g12\_ucount: Interaction term between ‘genre’ and ‘user\_count’

g1\_ccount – g12\_ccount: Interaction term between ‘genre’ and ‘critic\_count’

**Results of Final Model –**

1. **ANOVA results:**

A screenshot of a cell phone

Description automatically generated

1. **Categorical variables and Interaction terms in the model:**

A screenshot of a social media post

Description automatically generated

1. **Parameter Estimates –**

A large room

Description automatically generatedA picture containing window, large, people, white

Description automatically generated

A picture containing large

Description automatically generatedA picture containing large, white, room, keyboard

Description automatically generated

1. **Fit Diagnostics:**

A close up of a piece of paper

Description automatically generated

1. The initial model I started out with is as follows,

global \_sales = critic\_score + critic\_count + user\_score + user\_count

Following this, I made numerous changes and variations to my initial model by including different categorical variables and interaction terms. I have tabulated my thought process below,

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No.** | **Variations** | **Ideology** | **Decision** |
| 1. | Predictor: Name | I assumed that certain games are so popular that their name alone is sufficient to generate great reviews and sales and so I decided to include ‘name’ in my model. | The problem here was that ‘name’ had too many levels. This would require the creation of as many dummy variables in order to incorporate ‘name’ into my model. This would make the model difficult to interpret and increase computation time significantly. Hence, I decided not to include name in my model. |
| 2. | Predictors: Publisher and Developer | Similar to ‘name’, there are certain publishers and developers of games who are popular in the gaming community compared to others. I assumed these variables would in turn impact the model and affect global sales and so I decided to include both ‘publisher’ and ‘developer’ in my model. | Surprisingly, when I included ‘publisher’ in the model, it was found to be insignificant and thus, I decided not to include it.  However, though ‘developer’ was significant, including it in the model would once again be cumbersome as it had too many levels like ‘name’. |
| 3. | Predictor: Year of Release | I assumed that ‘year\_of\_release’ would play a major role in global sales of a game as I assumed that games that were released later would have more users since gaming evolved drastically in the recent years compared to the earlier years and decided to include it in my model. | However, when including ‘year\_of\_release’ in the model, it was found to be insignificant and thus, I decided not to include it in my final model. |
| 4. | Predictor: Rating | I assumed that ‘rating’ would be a contributing factor to global sales since different rated games would have different communities of users giving reviews. For ex: The rating ‘E’ would have a wider reach compared to a rating ‘M’ and hence, would result in more reviews and thus, more sales. | Like I presumed, ‘rating’ did have some certain significance to it and thus, I decided to retain it in my final model. |
| 5. | Predictor: Genre | Similar to ‘rating’, certain genres of games are more popular compared to others. This would mean more users prefer the popular genres, giving more reviews, resulting in more sales of that game compared to genres that are less popular. | Despite not being very significant, I decided to retain genre in my final model as I felt it would be a key predictor. |
| 6. | Predictor: Platform | Some games are released only in certain popular platforms and not in others. This would mean that these platforms would contribute to more sales of that game compared to less popular platforms and so I decided to include it in my model. | A few platforms were indeed significant as observed in the results and so I retained ‘platform’ in my final model. |
| 7. | Interaction terms: ‘critic’ and ‘user’ | ‘critic’ and ‘user’ are interaction terms between the respective scores and counts. I assumed that greater the count, greater would be the impact on score and decided to include these interaction terms in my model. | These interaction terms were mildly significant and thus, I retained them in my model. |
| 8. | Interaction terms: ‘r#\_ucount’ and ‘r#\_ccount’ | These are interactions between ‘rating’, ‘user\_count’ and ‘critic\_count’ respectively. I assumed that since different genres would have different number of users and critics, there would be an interaction that would in turn affect the global sales. | As observed in the results, certain interactions were indeed significant and thus, I decided to retain them in my final model. |
| 9. | Interaction terms:  ‘p#\_ucount’ and ‘p#\_ccount’ | These are interactions between ‘platform’, ‘user\_count’ and ‘critic\_count’. Similar to the previous interaction terms, I assumed that more popular platforms would have a larger community giving more reviews compared to less popular platforms which in turn would affect global sales. | Despite not being very significant, I felt this would be a key interaction that should be inspected further. Thus, I decided to retain it in my final model. |
| 10. | Interaction terms: ‘g#\_uscore’ and ‘g#\_cscore’ | These are interactions between ‘genre’, ‘user\_score’ and ‘critic\_score’. This seemed pretty straightforward as I assumed that more popular genres would generate higher scores compared to less popular genres thereby resulting in more sales. | To my surprise, this was not the case owing to its insignificance However, once again, I felt this would be a key interaction term and decided to retain it in my final model. |

1. The interpretation of the results of my final model are as follows,

* **Hypothesis testing –**

**Null Hypothesis H0**: The coefficients of all the predictors used in the model are equal to zero.

**Alternate Hypothesis H1**: The coefficients of all the predictors used in the model are significantly different from zero.

Based on the ANOVA results, we observe that,

1. The p-value for the model is < 0.0001, meaning we reject the null hypothesis. Thus, we can say that the coefficients of all the predictors used in the model are indeed significantly different from zero.
2. The R2 value is 0.3203 meaning that nearly 32% of the variation in the target variable is captured by the predictors and interactions used in the final model.
3. The adjusted R2 value 0.3085 which is close to R2 means that the variation captured in the target variable is indeed due to the predictors used and not simply because of the addition of variables into the model which may provide better fit but may have no relationship with the target variable.
4. The RMSE is also quite low with a value of 1.781.

* **Interpretation of significant individual predictors in the model –**

1. **user\_count**: The p-value of <0.0001 indicates that it is indeed significant pertaining to the model, and it’s estimate of 0.005 means that when the user count increases by 1, the global sales increases by 0.005 million units.
2. **r1**: It has a p-value of 0.01 meaning it is significantly different from r4 and its estimate of 0.451 indicates that compared to the sales of Rating M games which is the reference level, Rating E games have 0.451 million units more sale.
3. **g6**: It has a p-value of 0.01 meaning it is significantly different from g12 and its estimate of 1.61 indicates that compared to the sales of ‘fighting genre’ games which is the reference level, ‘role-playing genre’ games have 1.61 million units more sale.
4. **p2**: It has a p-value of 0.026 meaning it is significantly different from p10 and its estimate of 0.706 indicates that compared to the sales of games in the XS platform, which is the reference level, games in the GBA platform have 1.61 million units more sale.

* **Interpretation of significant interaction terms in the model –**

1. **r2\_ucount**: It has a p-value of 0.022 and its estimate of -0.0006 means that compared to rating ‘M’, rating ‘E10+’ had 0.0006 million lesser units’ sale due to the influence of ‘user\_count’.
2. **r1\_ccount**: It has a p-value of <0.0001 and its estimate of 0.033 means that compared to rating ‘M’, rating ‘E’ had 0.033 million more units’ sale due to the influence of ‘critic\_count’.
3. **p1\_ucount**: It has a p-value of <0.0001 and its estimate of 0.019 means that compared to platform ‘XS’, platform ‘DS’ had 0.019 million more units’ sale due to the influence of ‘user\_count’.
4. **p4\_ucount**: It has a p-value of <0.0001 and its estimate of -0.005 means that compared to platform ‘XS’, platform ‘PC’ had 0.005 million lesser units’ sale due to the influence of ‘user\_count’.
5. **p5\_ucount**: It has a p-value of 0.02 and its estimate of 0.002 means that compared to platform ‘XS’, platform ‘PS2’ had 0.002 million more units’ sale due to the influence of ‘user\_count’.
6. **p6\_ucount**: It has a p-value of 0.001 and its estimate of -0.003 means that compared to platform ‘XS’, platform ‘PS3’ had 0.003 million lesser units’ sale due to the influence of ‘user\_count’.
7. **p9\_ucount**: It has a p-value of 0.003 and its estimate of -0.003 means that compared to platform ‘XS’, platform ‘X360’ had 0.003 million lesser units’ sale due to the influence of ‘user\_count’.
8. **p8\_ccount**: It has a p-value of <0.0001 and its estimate of 0.04 means that compared to platform ‘XS’, platform ‘Wii’ had 0.04 million more units’ sale due to the influence of ‘critic\_count’.

**Question 2: -**

1. **Outliers and influential observations -**

Typically, the outliers and influential observations in the dataset are analyzed by using the Cook’s D: measure. Upon running the model with options for the Cook’s D values and Studentized residuals, we get the following plots,

A close up of a piece of paper

Description automatically generated

* From the plot between the predicted value and residuals, we observe that one observation is quite far away from zero.
* From the plot between the predicted value and Studentized residuals we observe that there are some points that fall outside the base condition of 2- standard deviations.
* From the plot showing the leverage of each observation, we see that certain observations have higher leverage compared to others.
* From the Q-Q plot, we observe that some points, lie away from the normal distribution line.
* From the plot showing the Cook’s D measure, we see that one particular observation has an exceptionally high Cook’s D value.
* From the histogram plot, we also observe that the data seems to be right-skewed.

Using the thumb’s rule: Cook’s D value > 4/n, we find the influential observations in the dataset. A sample of such observations is shown below,

A close up of a window

Description automatically generated

**Correction of Violation -**

In order to correct this violation, I run the regression model which only considers the observations whose Cook’s D value is < 4 /n. This led to more meaningful coefficients for critic\_score etc. as shown below,

A screenshot of a cell phone

Description automatically generated

As an alternate approach I used PROC REG which assigns weight to the outliers and influential observations, thereby solving the problem.

1. **Multicollinearity –**

To check for multicollinearity among the independent variables, I first generate a correlation matrix as shown below,

A picture containing large, sink

Description automatically generated

However, seeing above, we observe that owing to too many predictors, it would be difficult to determine whether multicollinearity exists.

Therefore, as an easier approach, I use the variance inflation factor (VIF) to determine if multicollinearity exists. As a base condition, I assume that if VIF > 10, it causes collinearity problems. Thereby, I dropped that kind of variables.

* Parameter estimates before dropping correlated variables –

A picture containing large, people

Description automatically generated

* Parameter estimates after dropping correlated variables –

A screenshot of a computer

Description automatically generated

Though dropping variables may create bias in the model, we decide to go ahead with this model. We could opt for PCA, but the problem is that PCA might not be able to properly interpret which indicators are significant and which are not.

1. **Heteroscedasticity in error term –**

Heteroscedasticity in error term is usually found out by conducting the White’s test. Observing the Heteroscedasticity consistent values, we see that most of the t-values are in fact lower than the actual t-values as shown below. This means that the problem of heteroscedasticity is absent in the model, thereby not requiring any transformation to the dependent variable ‘global\_sales’ as it would only worsen the model.

A picture containing large

Description automatically generated

1. **Normality of error term –**

If the error term did not follow a normal distribution, the p-values would be incorrect. Thus, we check for the normality of the error term before dropping the outliers through the PROC UNIVARIATE procedure as shown below,

A screenshot of a cell phone

Description automatically generated

A screenshot of a cell phone

Description automatically generated

Following this, we drop the outliers and generate the histogram. This is shown below,

A screenshot of a cell phone

Description automatically generated

A screenshot of a social media post

Description automatically generated

An important observation here is that, even after dropping the outliers, the residuals have a normal distribution, meaning this assumption was not violated and hinting that the normality issue may not be caused by the outliers.

**Final interpretation post correction of assumption violations –**

A screenshot of a cell phone

Description automatically generated

These show that key predictors like ‘critic\_score’ and ‘user\_count’ are now significant and have sensible estimates which is practically significant. Thus, the model is now slightly improved before correction of the violations.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*