Wish Shopping!

paper submitted for the completion of the project requirement in Stat 102B, Summer 2021

Group members : Yunjing Mai 3716,
Mengyu Zhang 0138

I. INTRODUCTION

The data set is called ”Sales Summer Clothes in E-commerce Wish”. The data was scraped from a platform called “Wish”(An American online e-commerce platform that facilitates transactions between sellers and buyers). It is the product listing with rating and performance during August 2020. The products listed in the dataset are those that would appear if you type "summer" in the search field of the platform Wish. Studying top products requires more than just product listings.

The reason we are interested in this topic is that we are big fans of online shopping. E-commerce is developing rapidly nowadays. Especially under the circumstances of the Covid-19, having people stick to the home. In this case, it creates tons of opportunities for merchants for E-commerce.

In this report, we would do the clustering to label different groups of products with key variables by the K-means algorithm. Since there are tons of variables affecting each other, we are trying to reduce the dimensions by principal components analysis so that we can focus on those dimensions that can explain the most variance of the data. Also, what we want to address from the data set in the further analysis are the factors that affect the rating of the product significantly. In other words, we want to figure out the underlying reasons to make a well-sell product in E-commerce by principal component regression. Before the analysis, we come up with a hypothesis to assume that the units sold and rating for the merchants.

II. DATA

For this original dataset, there are **1457** observations in the dataset with **35** variables. It includes **21** quantitative variables and **14** categorical variables. There are not any missing values in the data set.

The source of the dataset is Kaggle.
(link:https://www.kaggle.com/jmmvutu/summer-products-and-sales-in-ecommerce-wish?select=summer-products-with-rating-and-performance_2020-08.csv)

Quantitative variables:	
Price	The price you would pay to get the product
retail_price	the price before discount.
Updated_price	price/ retail_price -1 The percentage from retail_price changing into price
unites_sold	Number of units sold
Rating_count	Total number of ratings of the product
Rating	Mean product rating
Rating_five_count	Number of ratings in 5-star
rating_four_count	Number of ratings in 4-star
rating_three_count	Number of ratings in 3-star
rating_two_count	Number of ratings in 2_star
Rating_one_count	Number of ratings in 1-star
badges_count	Number of badges the product or the seller have
Product_variation_invertory	Inventory the seller has. Max allowed quantity is 50
Shipping_option_price	Shipping price
Inventory_total	Total inventory for all the product's variations
Merchant_rating_count	Number of rating of this seller
rating_five_percentage	Percentage of rating in 5-star
rating_four_percentage	Percentage of rating in 4-star
rating_three_percentage	Percentage of rating in 3-star
rating_two_percentage	Percentage of rating in 2_star
Rating_one_percentage	Percentage of rating in 1-star

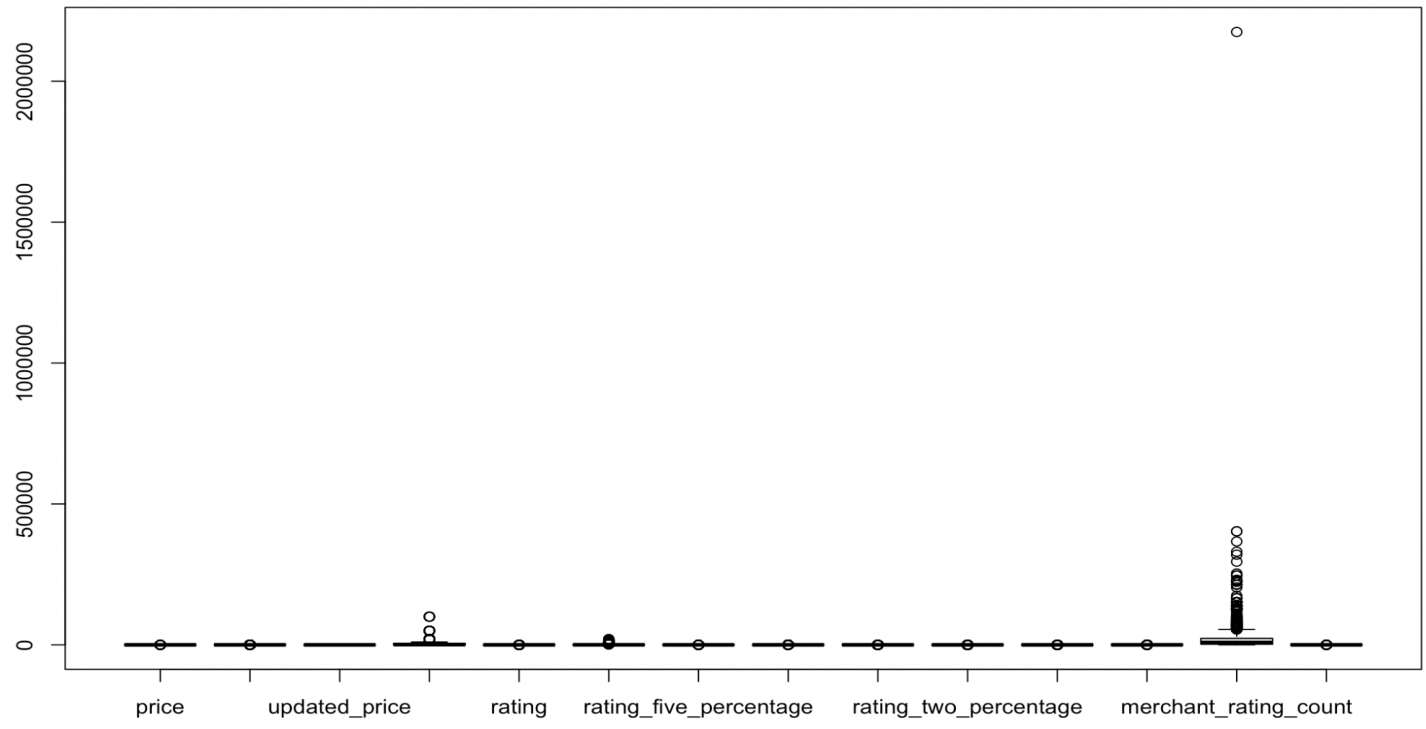
Categorical variables：	**Notes： 1 - YES 0 - NO**
title_orig （unique id）	Original English title of the product
badge_local_product	A badge that denotes the product is a local product. Conditions may vary (being produced locally, or something else). Some people may prefer buying local products rather than. 1 means Yes, has the badge. Otherwise 0 means No.
badge_product_quality	Badge awarded when many buyers consistently gave good evaluations 1 means Yes, has the badge
badge_fast_shipping	Badge awarded when this product's order is consistently shipped rapidly. 1 means Yes, has the badge
product_color	Product's main color
Shipping_option_name	The options for different shipping
Shipping_is_express	Whether the shipping is express or not
countries_shipped_to	Number of countries this product is shipped to

Has_urgency_banner	Whether there was an urgency banner with an urgency
origin_country	Original countries
Merchant's rating	Merchant's rating
merchant_has_profile_picture	whether there is a merchant profile picture
uses_ad_boosts	Whether the seller paid to boost his product within the platform (better_place/ highlighting)

*Cleaning the data *

The quantitative variables we would mainly focus on are
[1] "price" [2] "retail_price" [3] "updated_price"
[4] "units_sold" [5] "rating" [6] "rating_count"
[7] "rating_five_percentage" [8]"rating_four_percentage" [9] "rating_three_percentage"
[10] "rating_two_percentage" [11] "rating_one_percentage" [12] "countries_shipped_to"
[13] "merchant_rating_count" [14] "merchant_rating"

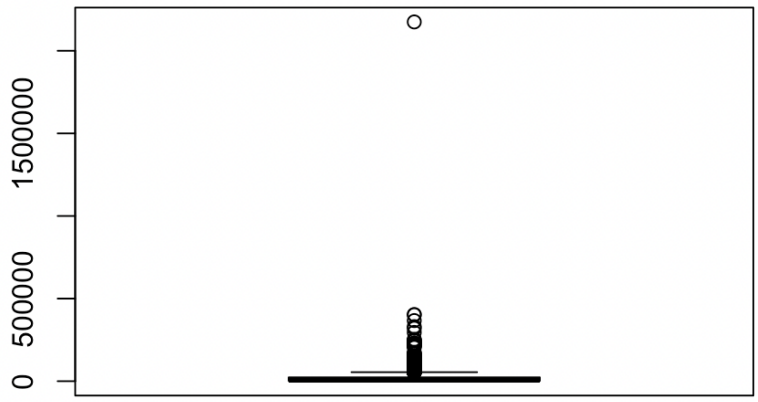
Since the dataset has not any missing values, we can focus on the outliers for cleansing the data. Firstly, let's take a look at the boxplot including all the quantitative variables.



At the first glance, we can notice that there are outstanding outliers for the 4th and 13th variables(represents units_sold and merchants_rating_count). Let's take a closer look at those variables.

```
> # check outlier for merchant_rating_count
> OutVals = boxplot(data$merchant_rating_count)$out
> length(OutVals) # 164 outliers
[1] 164
> sd(data$merchant_rating_count) # 69927.11
[1] 69927.11
```

fg.1



fg.2

As for fg.1 and fg.2 for the variable merchant_rating_count. There are 164 outliers and pretty high standard deviation of 69927.11.

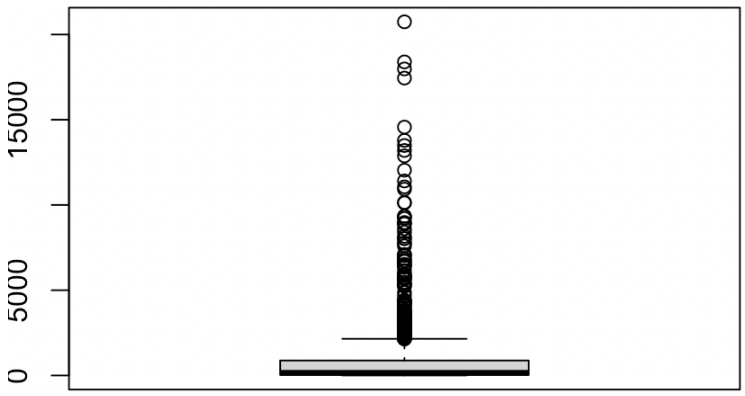
```
> # check outlier for units_sold
> OutVals = boxplot(data$units_sold)$out
> length(OutVals) # 116 outliers
[1] 116
> summary(data$units_sold)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
      2     100    1000    4339    5000  100000
> sd(data$units_sold) #9037.937
```

fg.3

For the units sold, there are 116 outliers and a standard deviation with 9037.937

```
> # check outlier for rating_count
> OutVals = boxplot(data$rating_count)$out
> length(OutVals) # 165 outliers
[1] 165
> sd(data$rating_count) # 1928.282
[1] 1928.282
```

fg.4

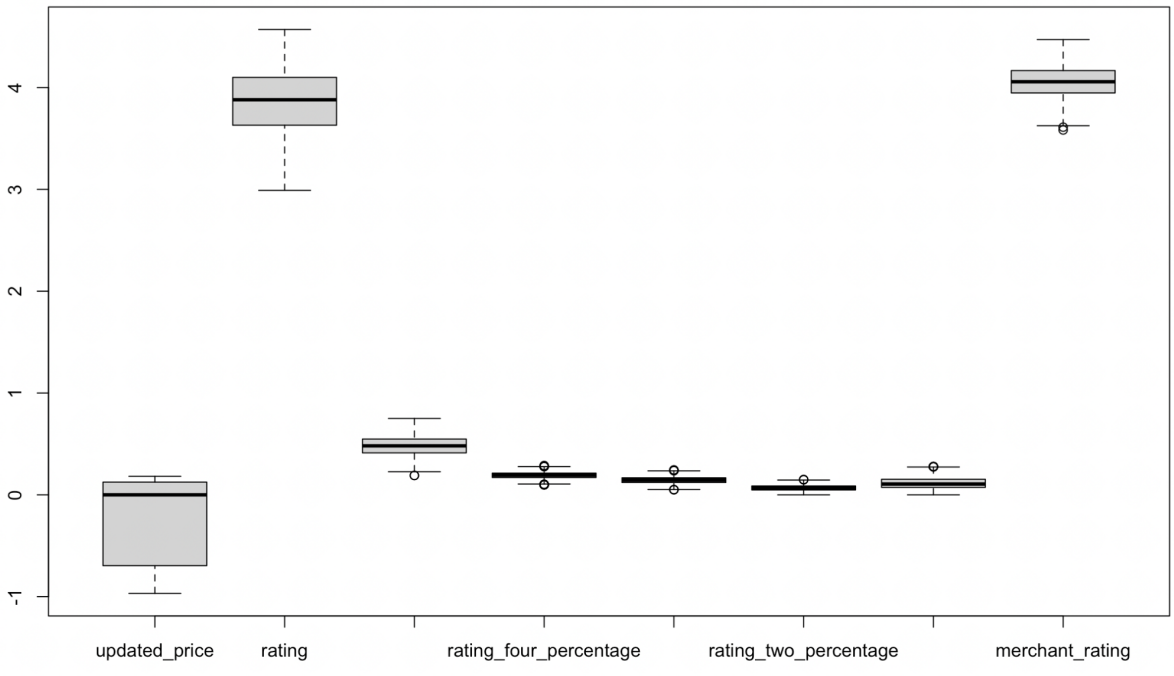


fg.5

Similarly, for the rating_count, there are 165 outliers and a standard deviation with 1928.282

Those variables have a high amount of outliers and high standard deviation which are not friendly for further study. As we went further, the variables Price, retail price, countries ship to also have the same issues. Finally, we decide to keep the variables "updated_price" "rating" "rating_five_percentage" "rating_four_percentage" "rating_three_percentage" "rating_two_percentage" "rating_one_percentage" "merchant_rating" .

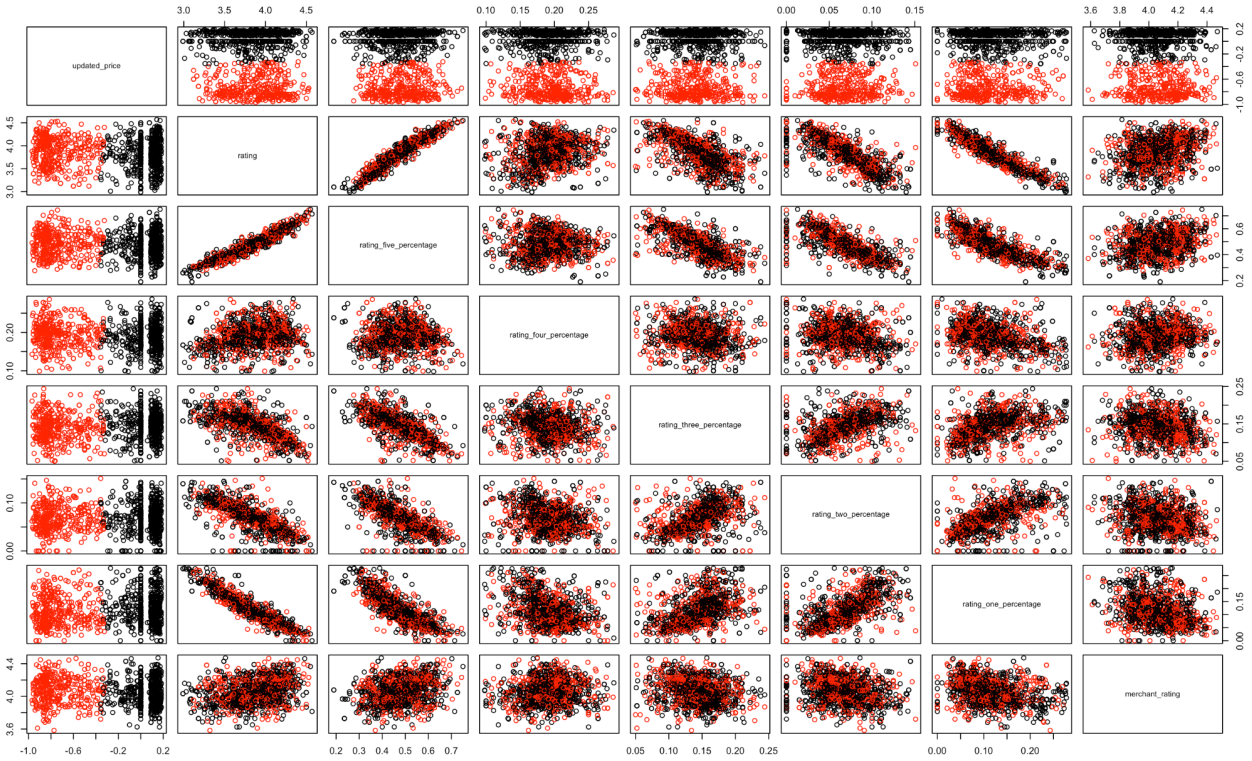
dataWith these remaining variables, we removed the rows with outliers. After we cleansing the outliers, the dimension of cleaned data is 1082 x 8. The plot6 is the final boxplot after removing outliers with much fewer outliers. We will do the analysis and regression with this cleaned .



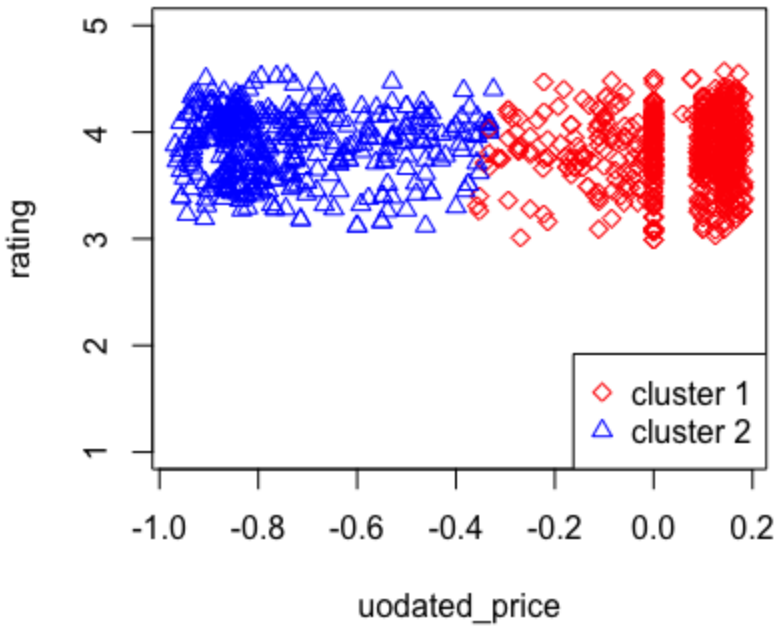
Fg.6

III. CLUSTER ANALYSIS

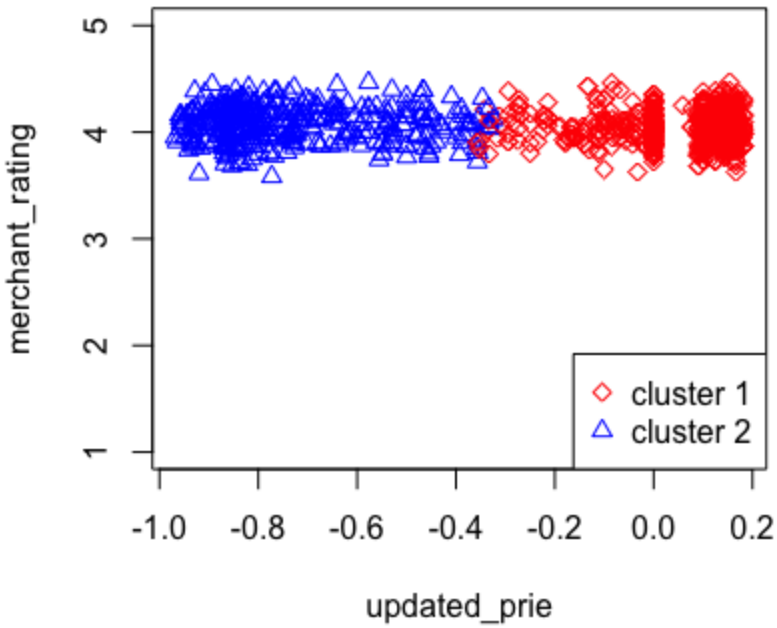
Since we have multiple variables, we applied k-means clustering on most related variables to view results of different pairs of variables. From the plot which displays in pairs of variables, we can see most of the variables do not perform well in clustering. The variables for which the mean is different in the two clusters will be more helpful in visualizing them than the other variables. We would compare two pairs of variables **(rating&merchant_rating)** by kmeans in 2 clusters. See how it could cluster the product by these two variables.




```
> K$centers
  updated_price   rating rating_five_percentage rating_four_percentage rating_three_percentage
1     0.0549657 3.834174              0.4764115              0.1899263               0.1456401
2    -0.7320502 3.877673              0.4866955              0.1945421               0.1400965
  rating_two_percentage rating_one_percentage merchant_rating
1            0.06736578             0.1206460        4.048699
2            0.06718564             0.1115074        4.073599
```



From the mean table above, we compared the variables between price difference(percentage of price increasement) and rating of product where we can see a clear partition. With a similar rating, Cluster 2 has the larger average price difference while cluster1 has the lower average price difference. These 2 clusters are separated clearly with few overlapping.



From the mean table above, we compared the variables between price difference and rating of merchants where we can see a clear partition. With a similar rating, Cluster 2 has the larger average price difference while cluster1 has the lower average price difference. These 2 clusters are separated clearly with few overlapping.

```
> table(K$cluster)

  1   2
678 404
```

There are 678 in cluster 1 and it takes 62.7% , cluster 2 has 404 and takes 33.3% overall, which indicates that cluster 1 is the majority. So judging by the visual inspection, I would conclude that there is one cluster of summer products on the Wish platform with a large rating count and high price difference and a group of small rating counts and small price differences. The price difference and rating count seem to be key variables in separating the groups.

IV. PRINCIPAL COMPONENTS FOR DIMENSION REDUCTION

```
            X1cs        X2cs        X3cs        X4cs        X5cs        X6cs        X7cs        X8cs
X1cs  1.00000000 -0.05911244 -0.04513399 -0.06641501  0.06525837  0.01019644  0.06712854 -0.07135636
X2cs -0.05911244  1.00000000  0.95434984  0.20304297 -0.59939844 -0.73106266 -0.93457564  0.26266988
X3cs -0.04513399  0.95434984  1.00000000 -0.05763943 -0.68371534 -0.71063135 -0.81696515  0.24388825
X4cs -0.06641501  0.20304297 -0.05763943  1.00000000 -0.13133488 -0.19633268 -0.30792136  0.07346419
X5cs  0.06525837 -0.59939844 -0.68371534 -0.13133488  1.00000000  0.37892731  0.39189180 -0.16838844
X6cs  0.01019644 -0.73106266 -0.71063135 -0.19633268  0.37892731  1.00000000  0.52988199 -0.14963787
X7cs  0.06712854 -0.93457564 -0.81696515 -0.30792136  0.39189180  0.52988199  1.00000000 -0.26151504
X8cs -0.07135636  0.26266988  0.24388825  0.07346419 -0.16838844 -0.14963787 -0.26151504  1.00000000
```

As indicated in the above statement, we could predict that just by looking at the correlations of the variables in the data set. We have a certain amount of correlations larger than 0.5 which means some of the variables have relatively high correlations.Therefore, we expect to do dimension reduction.

```
> cumsum(100*eigenvalues/(sum(eigenvalues)))
[1]  48.58002  62.24007  74.57124  85.75165  93.94482  99.99888  99.99968 100.00000
```

With 1 component	With 2 component	With 3 component	With 4 component	With 5 component
48.58002%	62.24007%	74.57124%	85.75165%	93.94482%

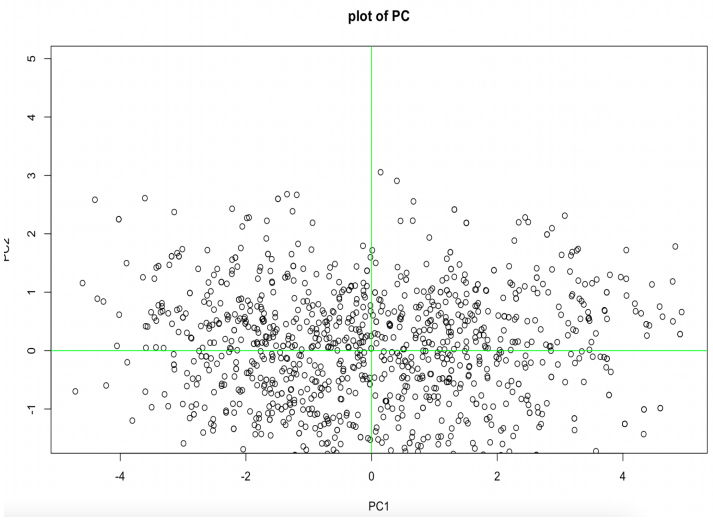
If we account for 90% of the variability in the measurement, we need the first five PCs.

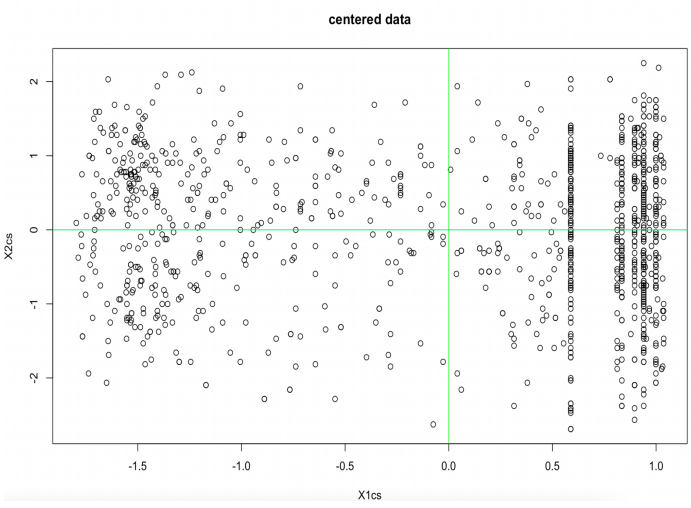
Because because the cumulative percentage of the variability in the data accounted for by four components is only 85.75165% while it has the percentage of 93.94482% for five components.Therefore, we selected the first 5 principal components which can explain about 93.935% of variance so that we can lower the dimensionality of 5 instead of 7.

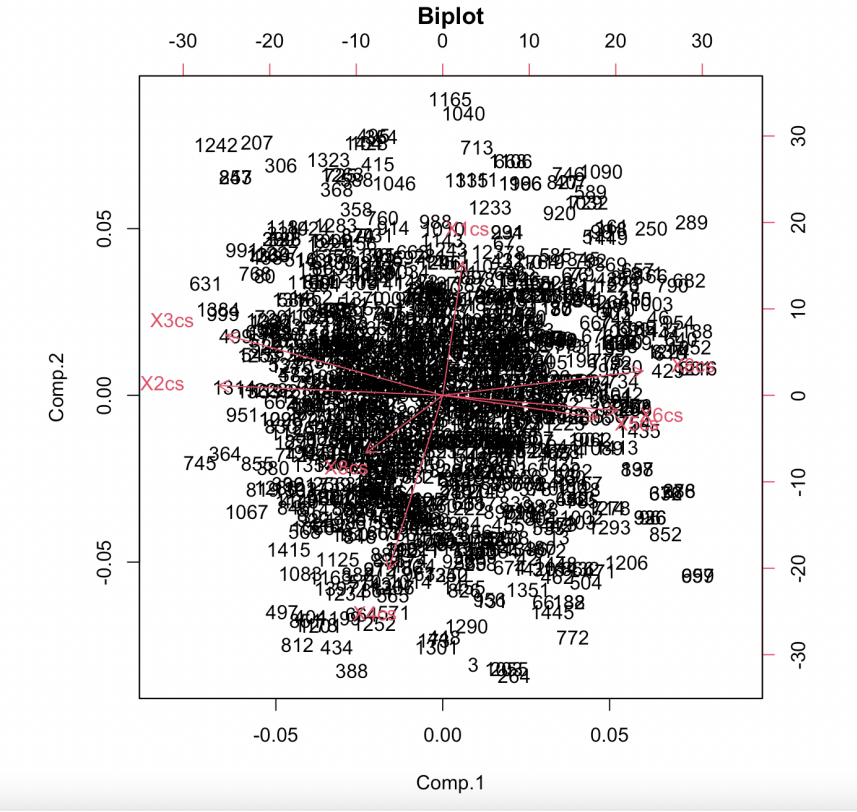
To do the PCA algorithm, we transformed the dataset into 8 principal components.

	V1	V2	V3	V4	V5	V6
updated_price	0.08750798	0.58723819	0.6652755	0.44150172	0.09354851	-0.03502918
rating	-0.98533251	0.04344672	0.0371135	-0.03925709	-0.10475181	-0.11530957
rating_five_percentage	-0.95114337	0.26504397	-0.1321537	-0.06741036	-0.01809682	-0.05218582
rating_four_percentage	-0.23820452	-0.76534286	0.5766199	0.01499739	0.15535549	0.02561867
rating_three_percentage	0.68289351	-0.09841939	0.1119561	0.12061144	-0.70376763	-0.03996657
rating_two_percentage	0.76898658	-0.06739282	-0.1549664	0.09839851	0.24112592	-0.55881681
rating_one_percentage	0.88017912	0.10749682	-0.1085434	-0.02247241	0.23952766	0.37953875
merchant_rating	-0.33771269	-0.25354629	-0.3797881	0.81759685	0.02149573	0.09218843

	V7	V8
updated_price	0.0000016945627	0.0000009780659
rating	-0.0066322434080	-0.0006884186973
rating_five_percentage	0.0015104461637	0.0040118034624
rating_four_percentage	-0.0001662460442	0.0013609286071
rating_three_percentage	-0.0009000235705	0.0013344905891
rating_two_percentage	-0.0013847923379	0.0010684812993
rating_one_percentage	-0.0039295130156	0.0019642200386
merchant_rating	-0.0000003531617	0.0000007565688







V. PRINCIPAL COMPONENTS REGRESSION FOR PREDICTION

In this part, the dependent variable I want to predict is the rating for the product. The independent variables we would use "updated_price" "rating_five_percentage" "rating_four_percentage" "rating_three_percentage" "rating_two_percentage" "rating_one_percentage" "merchant_rating" .The data set is splitted into two groups for testing and training. We have 10 rows for the testing set and the rest is the training test with 1072 rows.

```
> cor(y.cs,X.tilde )
          [,1]       [,2]       [,3]       [,4]       [,5]       [,6]         [,7]
[1,] -0.9704164 0.05842187 0.05532862 0.05535238 -0.1314285 -0.1771723 -0.001480625
```

From the correlation matrix between scared response variable rating and PC of scaled independent variables, we notice that PC(1), PC(2), PC(3), PC(5), and PC(6) are the most highly correlated with the dependent variable. (-0.9704164, 0.05842187, 0.05532862, -0.1314285, -0.1771723 respectively)

Note : PC(1), PC(2), PC(3), PC(5), and PC(6) represents "updated_price" , "rating_five_percentage" ,"rating_four_percentage" , "rating_two_percentage" "rating_one_percentage" respectively .

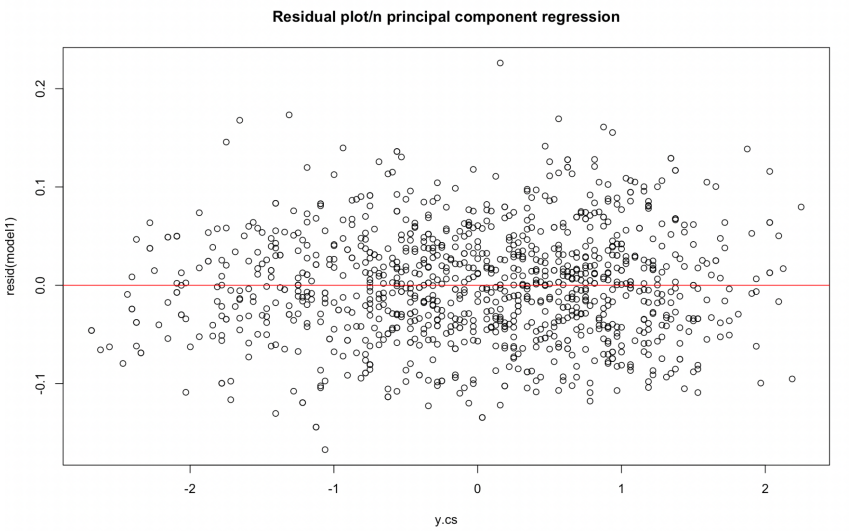
We run the regression with lm() with normalized PC with PC(1), PC(2), PC(3), PC(5), and PC(6). Below is the summary for the regression model, we can notice that all p-value for selected PC is much less than 0.05 which means those are the variables that significantly affect the response variable rating.

```
Call:
lm(formula = y.cs ~ pc1 + pc2 + pc3 + pc5 + pc6 - 1, data = df_train)

Residuals:
      Min        1Q    Median        3Q       Max
-0.167257 -0.038335  0.000059  0.036279  0.226136

Coefficients:
     Estimate Std. Error t value            Pr(>|t|)
pc1 -31.75798    0.05627 -564.42 <0.0000000000000002 ***
pc2   1.91192    0.05627   33.98 <0.0000000000000002 ***
pc3   1.81069    0.05627   32.18 <0.0000000000000002 ***
pc5  -4.30115    0.05627  -76.44 <0.0000000000000002 ***
pc6  -5.79816    0.05627 -103.05 <0.0000000000000002 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.05627 on 1067 degrees of freedom
Multiple R-squared:  0.9968,    Adjusted R-squared:  0.9968
F-statistic: 6.744e+04 on 5 and 1067 DF,  p-value: < 0.00000000000000022
```

From the summary table, we can get formula:
Y = -31.75798pc1 +1.91192pc2 + 1.81069pc3 - 4.30115pc5 - 5.79816pc6

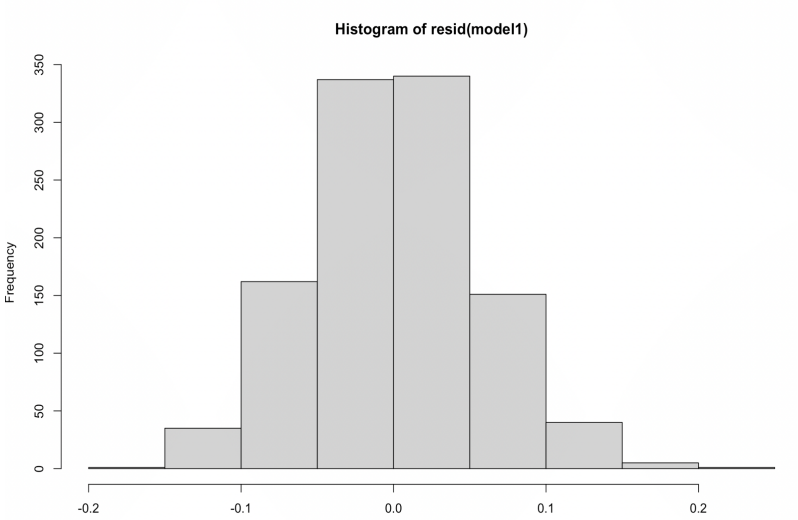
Rating = -31.75798 updated_price +1.91192 rating_five_percentage + 1.81069rating_four_percentage - 4.30115rating_two_percentage - 5.79816rating_one_percentage

This regression formula is describing how other independent variables affect the rating. The coefficient for updated price is the largest with 31.74798 units of rating decreses as one unit of updated_price increases.



Residual plot

From the residual plot, we can not see a clear pattern from the plot. The points are randomly dispersed around the horizontal axis. Therefore, a linear regression model is appropriate for the data.



The Histogram of the Residual

To check whether the variance is normally distributed, we plot the histogram of the residual. It shows us an Asymmetric bell-shaped histogram that is evenly distributed around zero. It indicates that the normality assumption is likely to be true.

```
> predicteds
     1360       793       291      1447       472       556       561       487       417      1253
7.968707 22.978300 -4.795853  5.757552 -5.450434 -2.270712 -5.090472 -3.980526 22.569649 -2.636295
```

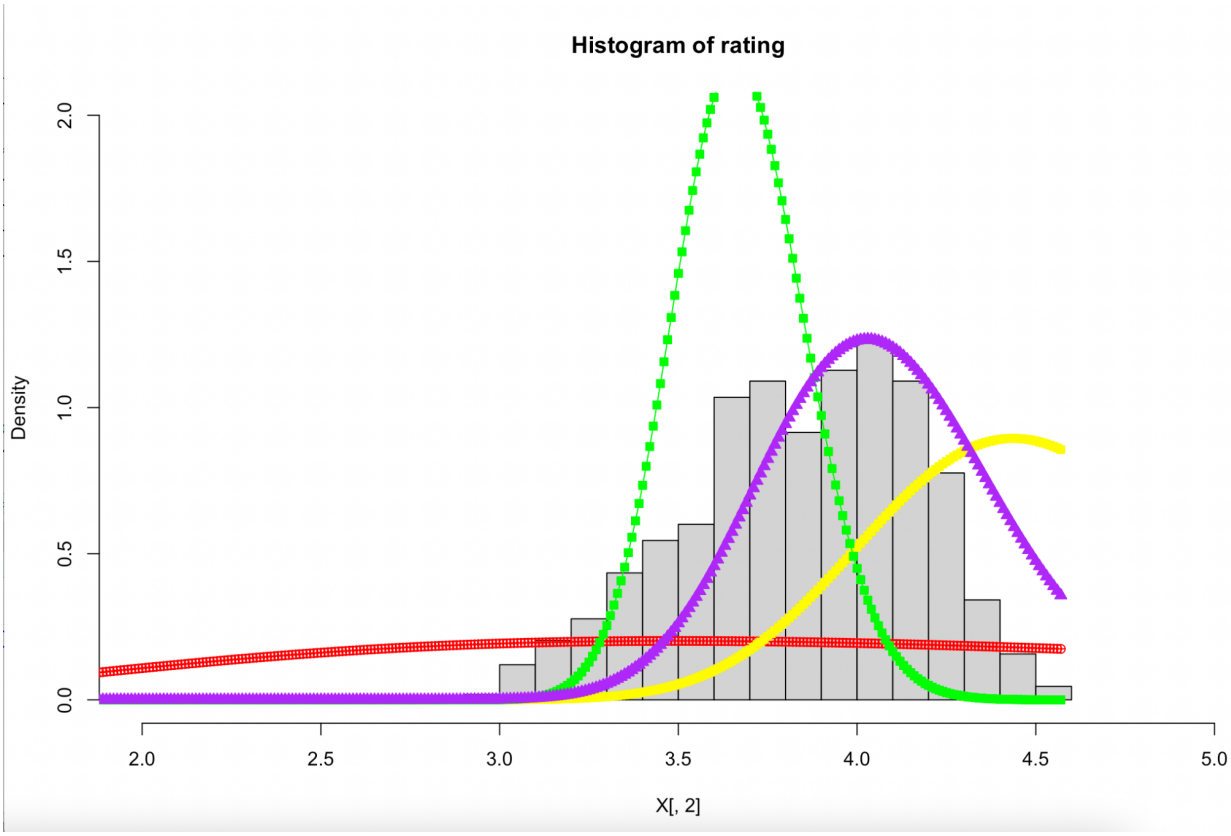
This is the predicted value for rating regarding the principal component regression. We can see the prediction does not perform well as most of the values are beyond the range of 0 to 5. Also, the negative values should not appear for the variables range.

```
> #root mean square error
> sqrt(mean((testing_set[,2] - predicteds)^2))
[1] 10.55311
```

The value of root mean square error between observed values and predicted values is 10.5531.Since the RMSE is beyond the ideal range between 0.2 and 0.5, the model can not relatively predict the data accurately. Further work is still needed.

From this research result, we can draw a conclusion that percentages of price increasement is the most significant factor affecting the rating of a product. The rating is sensitive to the price differences.

VI. MAXIMUM LIKELIHOOD ESTIMATION (UNCONSTRAINED)



After simulating the log normal distribution, we could see from the above plot that the purple line fit the histogram better than all others. We start from mean = 1.5 and sd = 0.5 to get the red line. Then, we try mean =1.5 and sd = 0.1and get the yellow line. After that we try mean = 1.3 and sd = 0.05, so we have the green line. Finally, we try the mean= 1.4 and sd =0.08 to get the purple line, which fits better than all others.

```
> mu.mle
[1] 1.34464
> sigma.mle
[1] 0.0847654
>
```

```
eigen() decomposition
$values
[1] -150587.9 -301175.8

$vectors
                             [,1]                        [,2]
[1,] -1.0000000000000000000000 0.0000000000000003631496
[2,]  0.0000000000000003631496 1.0000000000000000000000
```

As we can see both the eigenvalues of -150787.9 and - 301175.8 are negative, we can conclude that it is the maximum.

```
> CI_mu
[1] 1.339590 1.349691
> CI_sigma
[1] 0.0811940 0.0883368
```

The confidence interval of 95% for the parameters is (1.339590,1.349691) and (0.0811940,0.0883368) respectively.

VII. CONCLUSION

In this research, our group explored the dataset regarding the data for E-commerce products on Wish. We did the data cleansing by removing the outliers. With the cleaned data, we have done a k-means algorithm to cluster the product into two groups.We found out the groups can be labeled into the one with higher price increases and other one with lower price increases from retailed price to selling price. Based on the two clusters, I would target these two groups of products to differ in price increases.

Since the dimensionality and correlation with each variable are relatively high, we did the principle component analysis to reduce the dimension. With this algorithm, we transformed the dataset into 8 principal components. According to the cumulative percentage of the variability in the data, we selected the first 5 principal components which can explain about 93.935% of variance so that we can lower the dimensionality of 5 instead of 7.

In addition, we tried to predict rating as response values by other independent variables by principal component regression. We got the formula to predict rating:
Rating = -31.75798 updated_price +1.91192 rating_five_percentage + 1.81069rating_four_percentage - 4.30115rating_two_percentage - 5.79816rating_one_percentage
Using the formula,we predicted the rating from the testing set. The predicted values did not perform very well with relatively high root mean square error around 10. We consider it could be caused by insufficient data from another dimension. From this research result, we can draw a conclusion that percentages of price increasement is the most significant factor affecting the rating of a product. The rating is sensitive to the price differences. The discount of the product is larger, the rating would be higher. It encourages E-commerce merchants to spend more time on pricing strategy to improve the rating for their products.

Finally, we did the Maximum likelihood estimation by using the log normal model since the points from this model fit well. We calculated the mle estimators of the log normal model's parameter mean and standard deviation of 1.34464 and 0.0848046 respectively. The confidence interval of 95% for the parameters is (1.339590,1.349691) and (0.0811940,0.0883368) respectively.

VIII. ACKNOWLEDGMENTS

We want to acknowledged that we have used the class notes and code provided by Sanchez, J. Stat 102B lecture notes and R code.

IX. REFERENCES

[1] Data set from Kaggle:
https://www.kaggle.com/jmmvutu/summer-products-and-sales-in-ecommerce-wish?select=summer-products-with-rating-and-performance_2020-08.csv
[2] Sanchez, J. Stat 102B lecture notes and R code.
[3] R Core Team (2020). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/.