Analyzing Online Retail Datasets: a way to understand buyer's consuming behavior

Rian Dwi Putra
School of Computing

National College of Ireland

Dublin, Ireland

x22108637@student.ncirl.ie

ABSTRACT

The goal of data mining is to find and extract hidden information from data, such as relationships and patterns, which is where computer science, machine learning, and statistics meet. It is possible to identify variable associations, predict shopping patterns, classify large collections of e-books, and identify fraudulent bank transactions, among other things, depending on the method used and the data set being analyzed. The primary objective of this study is to evaluate methods for automatically classifying data in structured personal registries and gain a deeper understanding of the field of data mining. The study will first describe various data mining algorithms and methods, then apply a few well-known algorithms to three data sets and discuss the results.

INTRODUCTION

The extremely increasing popularity of online retail has attracted attention from tremendous amount of buyer in the world. Online retail has developed bigger than anything that anyone can possibly imagine.

Retailers provide online stores and sell goods to consumers or businesses. They buy from certain suppliers and then sell them online. For digital goods, they don't need a physical store, they just need a large storage capacity. Purchased goods are then shipped directly over the Internet for digital goods and through logistics providers for physical goods.

In addition, the retail business is also aimed at the ultimate consumer, which means that the goods or services traded in the retail business are goods that are used directly for consumers. Customers should have access to a computer, reliable internet connection and a payment method.

Like any other terminology which starts before the rise of online retail, everything starts from the very basic, such as traditional transaction and barter.

Therefore, by the advancement of technology, retail has been attractive. It is very difficult to foresee the outcome of online retail without understanding the influence it brings to the consumers. This influence will be defined by various aspect from consumer's point of view. This standpoint would be collected as information or data.

Between online retail and interpreting its data are correlated in various ways. The incredibly huge amount of data used in this business would be very exciting topic to be discussed. The use of data, which in this study referred as retail dataset, has become a guidance how this phenomenon will move.

The following are the research questions that require answers:

1. Which variable has the greatest impact on the sales?
2.Mention every independent variable for Retail Dataset Scenario.
3. Which is the most accurate regression model for Online Retail Dataset Scenario?
4. Which is the most suitable classification model for Online Retail Dataset Scenario?

RELATED WORK

Related works in this study are;

- Orogun and B. Onyekwelu, "Predicting consumer behaviour in digital market: a machine learning approach," International Journal of Innovative Research in Science, Engineering and Technology, vol. 8, no. 8, pp. 8391–8402, 2019.
- R. Gupta and C. Pathak, "A machine learning framework for predicting purchase by online customers based on dynamic pricing," Procedia Computer Science, vol. 36, pp. 599–605, 2014.
- X. S. Wang, J. H. J. Ryoo, N. Bendle, and P. K. Kopalle, "The role of machine learning analytics and metrics in retailing research," Journal of Retailing, vol. 97, no. 4, pp. 658–675, 2021.

Based on those works, we can conclude that, Retail is a business that sells products or services to individual consumers or end consumers for their own use or not for resale. The practice of selling products to customers who are either interested in using them or are interested in buying them from a single point of purchase is known as retail. A catalog, a shopping website, or a physical retail store can all serve as the point of purchase. A retail business involves selling products or services to customers in retail units.

Basically, the goal of a retail business is to make it easier for customers to buy products in sufficient stock and packaged into smaller sizes. Retail is also a link between producers and consumers. The following are some retail destinations.

- Make it easy for consumers

The retail business makes it easier for end-level consumers to get the products they need. Without a retail business, consumers will find it difficult to meet their needs. Because consumer must buy it directly from the manufacturer or wholesaler and must be in large quantities

- Favorable manufacturers and wholesalers

The existence of a retail business provides benefits for producers and wholesalers. Retailers will usually shop for stock in large quantities from wholesalers. Later, the funds obtained are used again for capital and producing goods.

- Promote products directly

Retailers will usually directly promote the products sold to consumers in various ways. This activity aims to create a trend of products produced from the producer's side.

- Offers a variety of prices

Various types of goods provided by retailers can come from different manufacturers, so the prices offered also vary. This will provide market variations in line with increasing consumer satisfaction.

Retailers rely on systems to supply merchandise to markets and to consumers. To achieve this goal, a retail supply chain process is needed. This chain consists of producers, wholesalers, retailers, and end consumers. At each step there is a profit margin. The following are part of the retail chain.

- Producer. Producing raw goods using machines and having a workforce.
- Wholesaler. Buying finished goods and selling them to retailers in bulk.
- Retailers. Selling goods in small quantities to end users at higher prices.
- Consumer. Buying goods from retail for personal needs.

DATA MINING METHODOLOGY

In this study, the KDD Methodology will be the primary distribution method. A set of procedures known as Knowledge Discovery in Databases (KDD) is used to explore, analyze, and extract useful knowledge from large data sets.

The entire non-trivial process of finding and identifying patterns in the data is included in KDD, provided that the patterns found are valid, novel, useful, and comprehensible. KDD is about scientific discovery, interpretation, and visualization of patterns in multiple data sets as well as integration techniques.

The Data Mining (KDD) Process includes several phases.

1. Understand the application domain

   The application domain will be subject to Retail Domain. The retail domain is collection of classes that appear for business objects used by Point-of-Sales. The Retail Domain is collection of business logic variables that construct retail-oriented business functionality in Point-of-Sales.

2. Identify data sources and select target data

   In this study, the dataset used is all about online retail dataset. The experiment presented will focus on the retail dataset based on 3 (three) online stores; macys.com, hanky-panky.com and ae.com. The dataset used in this study is from:

   https://www.kaggle.com/datasets/PromptCloudHQ/innerwear-data-from-victorias-secret-and-others

   There are some available datasets in this source, which will be used in this study. They were extracted data from the popular retail sites for data extraction solutions. Datasets included: Hanky Panky, Macy's, American Eagle.

Table 1. Table Template for Retail Dataset

Column Name	Description
product_name	Name of Product
mrp	Maximum Retail Price
price	Price of Product
pdp_url	URL for Product Detail Pages
brand_name	Name of Product Brand
product_category	Category of Product
retailer	Name of Retailer
description	Description of Product
rating	Rating of Product
review_count	Count of Product Review
style_attribute	Attribute of Product Style
total_sizes	Total of Product Size
available_size	Size Availability
color	Color
purchased	Has this Product already been Purchased or not

3. Pre-process: cleaning, attribute selection

The primary focus of data preprocessing in this study are numerical variable (such as price, rating, review count) and categorical variable (such as purchased history).

The process in this step related to set working directory, importing the dataset, encoding the target feature as factor and taking care of missing data.

```
# set working directory
setwd("~/R/DMML I Project/I")
```

Figure 1. set working directory

When building an RStudio Project, the working directory is the location where R will store, and load files associated with this project. Once done setting up the working directory, it will automatically set project files to the related location.

```
# Importing the dataset
dataset = read.csv('[DATASET].csv', sep = ';', header=T, stringsAsFactors=T)
dataset = dataset[1:4]
# Encoding the target feature as factor
dataset$purchased = factor(dataset$purchased, levels = c(0, 1))
```

Figure 2. importing dataset

importing data in R means that reading files, writing it to external files, and make the data accessible to be used within R environment

```
#taking care of missing data
dataset$rating = ifelse(is.na(dataset$rating),
                 ave(dataset$rating, FUN = function(x) mean(x, na.rm = TRUE)),
                 dataset$rating)
dataset$price = ifelse(is.na(dataset$price),
                ave(dataset$price, FUN = function(x) mean(x, na.rm = TRUE)),
                dataset$price)
dataset$review_count =ifelse(is.na(dataset$review_count),
                      ave(dataset$review_count, FUN = function(x) mean(x, na.rm = TRUE)),
                      dataset$review_count)
```

Figure 3. taking care of missing data

4. Data mining to extract patterns or models

Before building the machine learning model, this study will split the dataset into test set and training set.

```
# Splitting the dataset into the Training set and Test set
# install.packages('caTools')

library(caTools)
set.seed(123)
split = sample.split(dataset$purchased, SplitRatio = 0.75)
training_set = subset(dataset, split == TRUE)
test_set = subset(dataset, split == FALSE)
```

Figure 4. splitting data into training and testing

The training dataset is the subset of the original data that will be used to train the model; testing data will be used to verify the accuracy of the model. This is the difference between the training dataset and the testing dataset. Typically, the training data is larger than the testing data.

The scaling of features is the next step. Using feature scaling, numerical data in a dataset can be made to have the same range of values (scale). There is no longer a single data variable that is more important than the others.

```
# Feature Scaling
training_set[,-4] = scale(training_set[,-4])
test_set[,-4] = scale(test_set[,-4])
```

Figure 5. feature scaling for logistic regression

Models which will be built for this study are revolved around Regression and Classification technique.
Classification technique includes:

- Logistic Regression

This study will use logistic regression, a type of statistical analysis, for predictive modeling. The dependent variable is either finite or categorical in this analytical method, and it can be either A or B (binary regression) or any combination of A, B, C, or D (multinomial regression).

In statistical software, this kind of statistical analysis is used to estimate the probability of the relationship between the dependent variable and one or more independent variables. Predicting the odds can be made easier with this type of analysis.

For building this model, we need to import the dataset. In this model, we will use ae_com.csv as dataset.

```
# Importing the dataset
dataset = read.csv('ae_com.csv', sep = ';', header=T, stringsAsFactors=T)
dataset = dataset[1:4]
# Encoding the target feature as factor
dataset$purchased = factor(dataset$purchased, levels = c(0, 1))
```

Figure 6. importing ae_com.csv dataset

After reading dataset, we will be taking care of missing data, split the data into training set and test set and doing feature scaling (as described earlier).

```
# Fitting [Logistic Regression] to the [Training set]
classifier = glm(formula = as.factor(training_set$purchased) ~ price + review_count + rating,
             family = binomial,
             data = training_set)

# predicting the [Test set] results
prob_prediction = predict(classifier, type = 'response', newdata = test_set[-4])


# grouping the prediction result from previous level
y_prediction = ifelse(prob_prediction > 0.5, 1, 0)
```

Figure 7. building model for logistic regression

In building machine learning model for Logistic Regression, the steps are :

- Fitting Logistic Regression to the Training set

  The flexible generalization of conventional linear regression is the generalized linear model (GLM). By allowing the linear model to be linked to the response variable via a link function and allowing the magnitude of each measurement's variance to be a function of its predicted value, the GLM makes linear regression more general.[6]
- Predicting the Test set results
- Grouping the prediction result from previous level

- SVM

Support Vector Machine–hereinafter referred to as SVM– is a method in machine learning that can be used to analyze data and sort it into one of two categories.

A learning system known as the Support Vector Machine (SVM) makes use of a hypothetical space represented by linear functions in a high-dimensional feature and is trained with an optimization-theoretic learning algorithm.

For building this model, we need to import the dataset. In this model, we will use hankypanky_com.csv as dataset

```
# Importing the dataset
dataset = read.csv('hankypanky_com.csv', sep = ';', header=T, stringsAsFactors=T)
dataset = dataset[1:4]
```

Figure 8. importing hankypanky.csv dataset

After reading dataset, we will be taking care of missing data, split the data into training set and test set and doing feature scaling (as described earlier).

```
# Fitting SVM to the [Training set]
# install.packages('e1071')
library(e1071)
classifier = svm(formula = purchased ~ price + review_count + rating,
                 data = training_set,
                 type = 'C-classification',
                 kernel = 'linear')

# Predicting the Test set results
y_prediction = predict(classifier, newdata = test_set[-4])
```

Figure 9. building model for SVM

In building machine learning model for SVM, the steps are :

- Fitting SVM to the Training set

SVM works to find the best hyperplane or separation function (decision boundary) to separate two or more classes in the input space. The hyperplane can be a line or a line in two dimensions and can be a flat plane in multiple planes.

- Predicting the Test set results

- Naïve Bayes

The Naive Bayes method works well for both binary and multiclass classification. Naive Bayes Classifier, another name for this approach, uses conditional probabilities to assign class labels to instances or records, putting the supervised method of future object classification into practice. A measure of the probability of an event occurring based on other events that have (assuming, presumed, stated, or proven) to occur is known as conditional probability.

For building this model, we need to import the dataset. In this model, we will use macys_com.csv as dataset

```
# Importing the dataset
dataset = read.csv('macys_com.csv', sep = ';', header=T, stringsAsFactors=T)
dataset = dataset[1:4]
```

Figure 10. importing macys_com.csv dataset

After reading dataset, we will be taking care of missing data, split the data into training set and test set and doing feature scaling (as described earlier).

```
# Fitting Naive Bayes to the Training set
# install.packages('e1071')
library(e1071)
classifier = naiveBayes(x = training_set[-4],
                        y = training_set$purchased)


# Predicting the Test set results
y_prediction = predict(classifier, newdata = test_set[-4])
```

Figure 11. building model for naïve bayes

In building machine learning model for Naïve Bayes, the steps are :

- Fitting Naïve Bayes to the Training set

  The workings of the Naïve Bayes Classifier function calculate the probability of one class from each existing attribute group and determine which class is the most optimal, meaning that grouping can be done based on the categories.

- Predicting the Test set results

In the meantime, the method of regression is similar to linear regression (simple, multiple, or polynomial regression): In quantitative research, linear regression is used to estimate or predict the relationship between two variables. where linear regression can make one more assumption that links the independent and dependent variables by following the best line from the straight-line data point. That is, there is not a curve or any clustering factor.

However, linear regression has its limitations, because even the best data do not tell the full story. Regression analysis is usually used in research to establish that there is a correlation between variables.

For building this model, we need to import the dataset. In this model, we will use ae_com.csv as dataset.

```
# Importing the dataset
dataset = read.csv('ae_com.csv', sep = ';', header=T, stringsAsFactors=T)
dataset = dataset[1:3]
```

Figure 12. importing ae_com.csv dataset

After reading dataset, we will be taking care of missing data, split the data into training set and test set and doing feature scaling (as described earlier).

```
# Fitting Linear Regression to the dataset
lin_reg = lm(formula = rating ~ price,
             data = dataset)
summary(lin_reg)
```

Figure 13. building simple linear regression model

```
poly_reg = lm(formula = rating  ~ price+review_count,
              data = dataset)
```

Figure 14. building multiple linear regression model

In building machine learning model for Simple Linear Regression, the steps are :

- Fitting Linear Regression to the Training set
- Predicting the Test set results

5. Post-process: identifying useful patterns

It classifies unseen data. And it will make valuable predictions, by employing learning techniques. It recognizes and identify the result at various point of view.

6. Incorporate patterns in real world tasks

It creates suggestion or predictions, by using machine learning model, of the result analyzed and provide practical, actionable suggestions.

EVALUATION

For the Classification Model, the performance measured on this study will focus on the result of

- Accuracy of average classification.

Accuracy is a test method based on the level of closeness between the predicted value and the actual value. By knowing the amount of data that is classified correctly, the accuracy of the prediction results can be known.

```
               Accuracy : 0.4881
                 95% CI : (0.4764, 0.4999)
    No Information Rate : 0.5473
    P-Value [Acc > NIR] : 1

                  Kappa : -0.0235

 Mcnemar's Test P-Value : 1.345e-08

            Sensitivity : 0.4881
            Specificity : 0.4881
         Pos Pred Value : 0.4410
         Neg Pred Value : 0.5355
             Prevalence : 0.4527
         Detection Rate : 0.2210
   Detection Prevalence : 0.5011
      Balanced Accuracy : 0.4881

       'Positive' Class : 0
```

Figure 15. Accuracy for Logistic Regression Model

```
               Accuracy : 0.5028
                 95% CI : (0.4923, 0.5133)
    No Information Rate : 0.9928
    P-Value [Acc > NIR] : 1

                  Kappa : -0.0015

 Mcnemar's Test P-Value : <2e-16

            Sensitivity : 0.444444
            Specificity : 0.503223
         Pos Pred Value : 0.006446
         Neg Pred Value : 0.992058
             Prevalence : 0.007199
         Detection Rate : 0.003200
   Detection Prevalence : 0.496400
      Balanced Accuracy : 0.473834

       'Positive' Class : 0
```

Figure 16. Accuracy for SVM Model

```
               Accuracy : 0.4988
                 95% CI : (0.489, 0.5085)
    No Information Rate : 0.6763
    P-Value [Acc > NIR] : 1

                  Kappa : -0.0035

 Mcnemar's Test P-Value : <2e-16

            Sensitivity : 0.4958
            Specificity : 0.5002
         Pos Pred Value : 0.3220
         Neg Pred Value : 0.6745
             Prevalence : 0.3237
         Detection Rate : 0.1605
   Detection Prevalence : 0.4985
      Balanced Accuracy : 0.4980

       'Positive' Class : 0
```

Figure 17. Accuracy for Naïve Bayes Model

- Confusion matrix, which is table showing the number of true negatives, true positives, false negative, false positives.

```
Confusion Matrix and Statistics

          Reference
Prediction    0    1
         0 1565 1984
         1 1641 1892
```

Figure 18. Confusion Matrix for Logistic Regression

```
Confusion Matrix and Statistics

          Reference
Prediction    0    1
         0   28 4316
         1   35 4372
```

Figure 19. Confusion Matrix for SVM

```
Confusion Matrix and Statistics

          Reference
Prediction    0    1
         0 1641 3456
         1 1669 3459
```

Figure 20. Confusion Matrix for Naïve Bayes

- Precision, Recall and Specificity. major performance metrics for describing and analyzing classification model.
    - Precision is a test method by comparing the amount of relevant information obtained by the system with the total amount of information taken by the system, whether relevant or not.
    - Recall is a test method that compares the amount of relevant information obtained by the system with the total amount of relevant information in the collection of information (either taken or not taken by the system).

```
> accuracy
[1] 0.4881389
>
> precision = diag / colsums
> recall = diag / rowsums
> f1 = 2 * precision * recall / (precision + recall)
>
> data.frame(precision, recall, f1)
  precision    recall        f1
0 0.4881472 0.4409693 0.4633605
1 0.4881321 0.5355222 0.5107302
```

Figure 21. Precision and recall for Logistic Regression

```
> accuracy
[1] 0.5027997
>
> precision = diag / colsums
> recall = diag / rowsums
> f1 = 2 * precision * recall / (precision + recall)
>
> data.frame(precision, recall, f1)
  precision      recall         f1
0 0.4444444 0.006445672 0.01270706
1 0.5032228 0.992058089 0.66773578
>
```

Figure 22. Precision and recall for SVM

```
> accuracy
[1] 0.4987775
>
> precision = diag / colsums
> recall = diag / rowsums
> f1 = 2 * precision * recall / (precision + recall)
>
> data.frame(precision, recall, f1)
  precision    recall        f1
0 0.4957704 0.3219541 0.3903890
1 0.5002169 0.6745320 0.5744416
>
```

Figure 23. Precision and recall for Naïve Bayes

And for Regression model, RMSE, MAE, MSE, and R-Squared metrics will be used to evaluate the result.

```
Call:
lm(formula = rating ~ price, data = dataset)

Residuals:
    Min      1Q  Median      3Q     Max
-3.5530 -0.2819  0.1181  0.4928  1.7400

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  4.8072837  0.0127243  377.80   <2e-16 ***
price       -0.0315836  0.0006999  -45.13   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.7036 on 21123 degrees of freedom
  (7203 observations deleted due to missingness)
Multiple R-squared:  0.08793,   Adjusted R-squared:  0.08789
F-statistic:  2036 on 1 and 21123 DF,  p-value: < 2.2e-16

> AIC(lin_reg)
[1] 45098.72
> BIC(lin_reg)
[1] 45122.59
>
> library(modelr)
> data.frame(
+   R2 = rsquare(lin_reg, data = dataset),
+   RMSE = rmse(lin_reg, data = dataset),
+   MAE = mae(lin_reg, data = dataset)
+ )
         R2      RMSE       MAE
1 0.08792984 0.7035226 0.4922254
>
> glance(lin_reg) %>%
+   dplyr::select(adj.r.squared, sigma, AIC, BIC, p.value)
# A tibble: 1 × 5
  adj.r.squared sigma    AIC    BIC p.value
          <dbl> <dbl>  <dbl>  <dbl>   <dbl>
1        0.0879 0.704 45099. 45123.       0
```

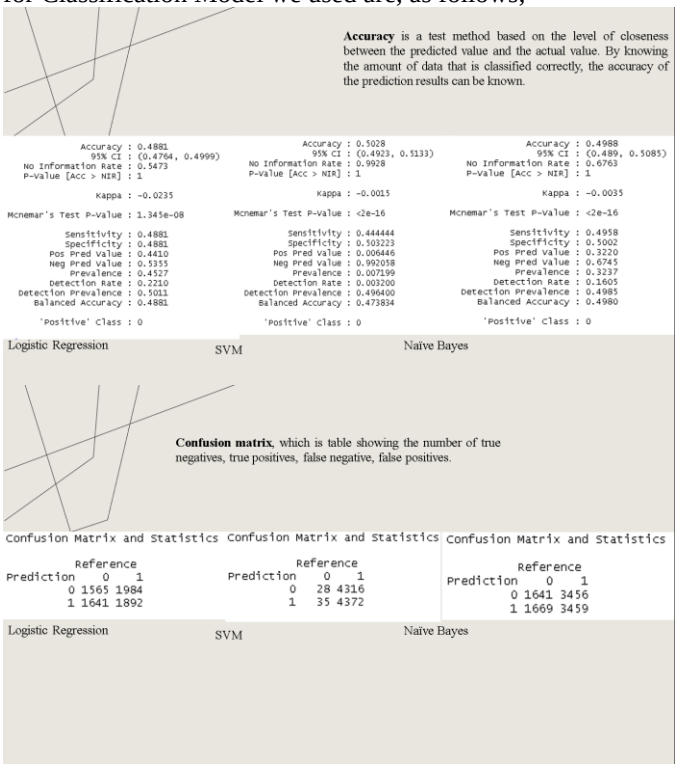
Figure 24. Summary of Simple Linear Regression

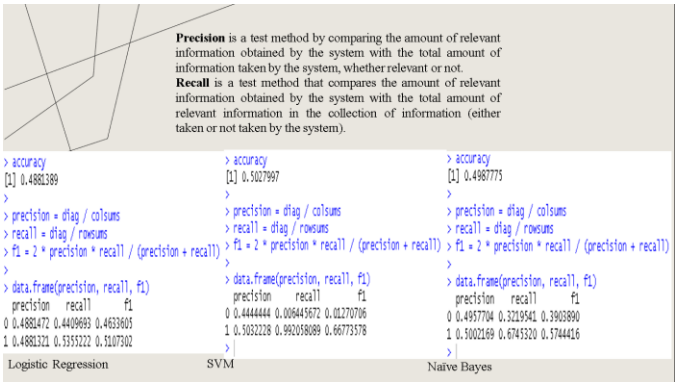
```
lm(formula = rating ~ price + review_count, data = dataset)

Residuals:
    Min      1Q  Median      3Q     Max
-3.5756 -0.2224  0.2289  0.4884  2.2329

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)   4.9258548  0.0230791 213.434   <2e-16 ***
price        -0.0441763  0.0014669 -30.114   <2e-16 ***
review_count  0.0054140  0.0005462   9.913   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.8394 on 9587 degrees of freedom
  (18738 observations deleted due to missingness)
Multiple R-squared:  0.09163,   Adjusted R-squared:  0.09145
F-statistic: 483.6 on 2 and 9587 DF,  p-value: < 2.2e-16

> AIC(poly_reg)
[1] 23862.88
> BIC(poly_reg)
[1] 23891.56
>
> library(modelr)
>
> data.frame(
+   R2 = rsquare(poly_reg, data = dataset),
+   RMSE = rmse(poly_reg, data = dataset),
+   MAE = mae(poly_reg, data = dataset)
+ )
         R2      RMSE       MAE
1 -0.2981328 0.8392883 0.5942489
>
> glance(poly_reg) %>%
+   dplyr::select(adj.r.squared, sigma, AIC, BIC, p.value)
# A tibble: 1 × 5
  adj.r.squared sigma    AIC    BIC   p.value
          <dbl> <dbl>  <dbl>  <dbl>     <dbl>
1        0.0914 0.839 23863. 23892. 8.35e-201
```

Figure 25. Summary of Multiple Linear Regression Model

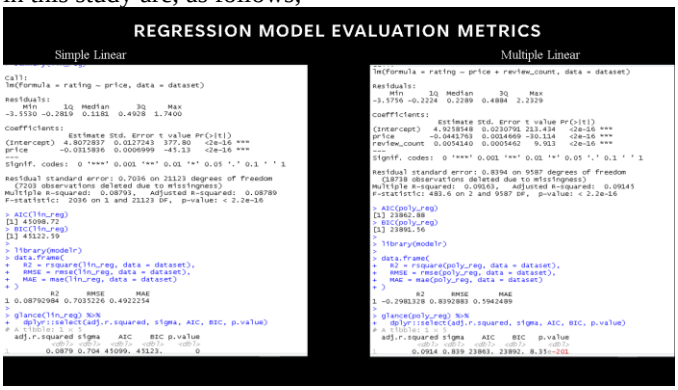
CONCLUSIONS AND FUTURE WORK

Based on evaluation drawn, we can interpret the conclusions for Classification Model we used are, as follows;





And we can conclude from Linear Regression model we used in this study are, as follows;



Hence, we can answer this study objective, are as follows;

1. What variable has the most impact to the sales of business?

Rating, Price, and Review_Count are the variable which affect the most to the Sales of Business.

2.Mention every independent variable for Retail Dataset Scenario.

Dependent variable in this Retail Dataset are Rating, Price, and Review_Count.

3. Which is the most accurate regression model for Online Retail Dataset Scenario?

Based on this study, a more precise calculation than simple linear regression is multiple linear regression. For straight-forward connections, basic direct relapse may effectively catch the connection between the two factors. Multiple linear regression is frequently superior for relationships that are more intricate and necessitate more thought.

4. Which is the most suitable classification model for Online Retail Dataset Scenario?

This study concludes that every model has its own advantages. This is a brief explanation for every single model.

- Since Naive Bayes assumes that the features are independent, it basically calculates the posterior probability product for each feature. Since it is almost never the case that features are completely independent, it is evident that this is an absurd assumption. In any case, the fundamental benefit of this is adaptability. Maximum-likelihood training can be performed by evaluating a closed-form expression in linear time, and the number of parameters in this model is proportional to the number of features. However, this study has observed that Naive Bayes is rarely utilized when performance matters.
- Continue with Logistic Regression, there are a few reasons this model is useful:
  - It yields all around aligned class probabilities
  - It has an unconstrained, smooth misfortune capability (contrasted with SVM)
  - It plays well with Bayesian strategies
- There are a few reasons why SVM is useful:
  - Correctly labeled examples are not penalized by SVMs, which is often helpful for generalization.
  - Gives meager arrangements while utilizing the piece stunt (pleasant for adaptability)

REFERENCES

[1] P. Kalia, A. Zia, and K. Kaur, "Social influence in online retail: A review and research agenda," European Management Journal, 2022.

[2] T. L. Childers, C. L. Carr, J. Peck, and S. Carson, "Hedonic and utilitarian motivations for online retail shopping behavior," Journal of retailing, vol. 77, no. 4, pp. 511–535, 2001.

[3] V. Insley and D. Nunan, "Gamification and the online retail experience," International Journal of Retail & Distribution Management, 2014.

[4] P. W. Ballantine, Effects of interactivity and product information on consumer satisfaction in an online retail setting. International journal of retail & distribution management. 2005.

[5] P. Soni, Revisiting the role of relationship benefits in online retail. Marketing Intelligence & Planning. 2019.

[6] Wikipedia contributors, "Generalized linear model," Wikipedia, The Free Encyclopedia, 26-Oct-2022. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Generalized_linear_model&oldid=1118341387.