Project Report

Rushikesh K Khankar

Analysis of Sales Report of a Clothes Manufacturing Outlet

Background and Objective:

A high-end fashion retail store is looking to expand its products. It wants to understand the market and find the current trends in the industry. It has a database of all products with attributes, such as style, material, season, and the sales of the products over a period of two months.

Domain: Retail

Dataset Description:

There are two files provided, and the detailed description of each is given below:

Attribute	Description
Dress_ID	A unique identifier for each dress
Style	Style of dress can belong to one of 12 styles, including casuals, novelty, etc.
Price	Price category of the dress (low, average, medium, high, and very high)
Rating	A number between 0 and 5, specifying the rating of the dress
Size	Size of the dress (small, medium, large, XL, and free)
Season	Season category of the dress, i.e., summer, spring, etc
Neckline	Type of neckline, for example, V-neck, collar, etc
Sleeve length	Length of the sleeve—full, three-quarters, etc
Waistline	The waistline of the dress
Material	The material of the dress, for example, silk, cotton, etc
Fabric type	Fabric type of dress
Decoration	The decoration of the dress, like ruffles, embroidery, etc
Pattern Type	Pattern type of the dress—dot, animal print, etc
Recommendation	A binary value suggesting a recommendation (1) or not (0)

The remaining columns depict the sales for each dress on a particular date. Date ranges from 29/8/2013 to 12/10/2013, and the sales are registered for alternative days.

Analysis Tasks:

The goals of this project are:

- To automate the process of recommendations, the store needs to analyze the given attributes of the product, like the style, season, etc., and come up with a model to predict the recommendation of products (in binary output – 0 or 1) accordingly.
- In order to stock the inventory, the store wants to analyze the sales data and predict the trend of total sales for each dress for an extended period of three more alternative days.
- To decide the pricing for various upcoming clothes, they wish to find how the style, season, and material affect the sales of a dress and if the style of the dress is more influential than its price.
- Also, to increase sales, the management wants to analyze the attributes of dresses and find which are the leading factors affecting the sale of a dress.
- To regularize the rating procedure and find its efficiency, the store wants to find if the rating of the dress affects the total sales

1. Project Motivation

The aim of this project to analysis of sales report of a Clothes Manufacturing Outlet. . And achieve the following goals:

1. Build a classifier for dress recommendation prediction.
2. Show the effect of dress attributes on the product price.

2. Installation

- R
- R Libraries:
  - readxl
  - plyr
  - caTools
  - e1071
  - caret
  - randomForest

3. Data

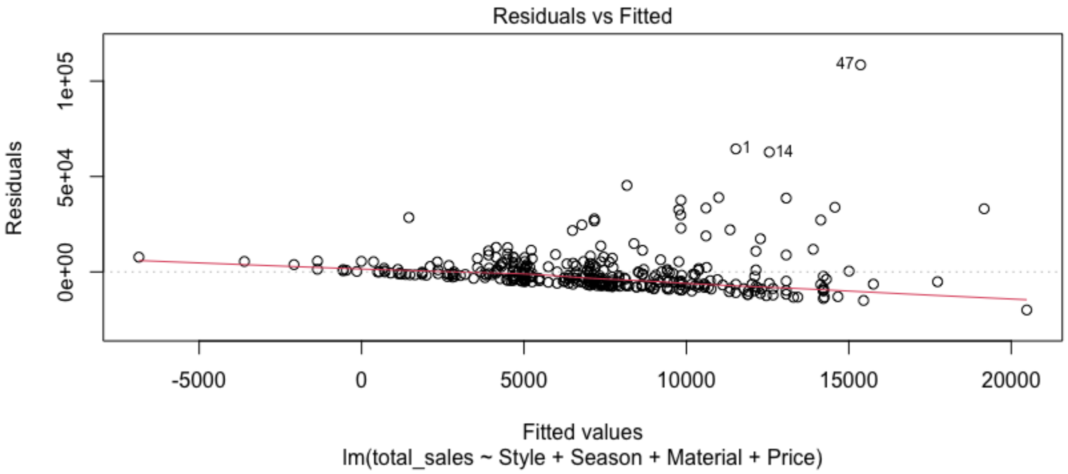
The dataset was provided from Simplilearn platform.

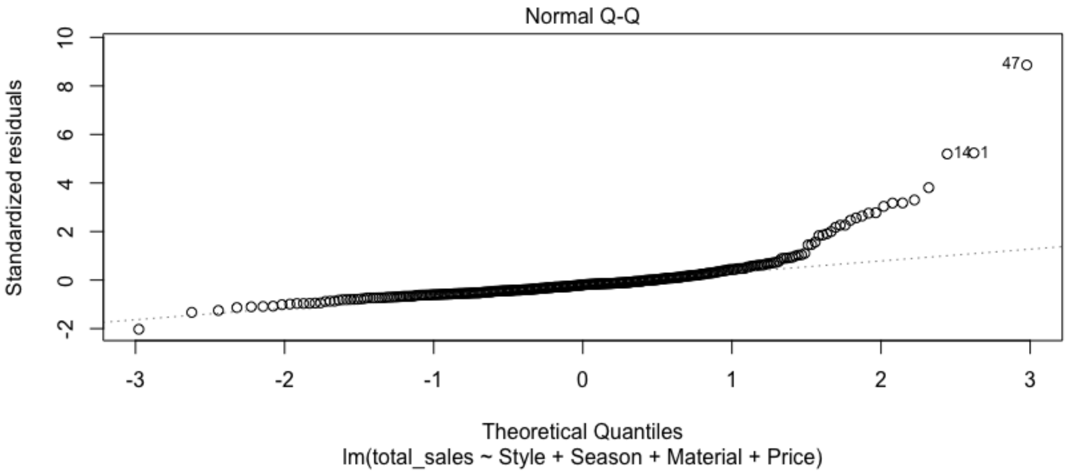
4. Implementation

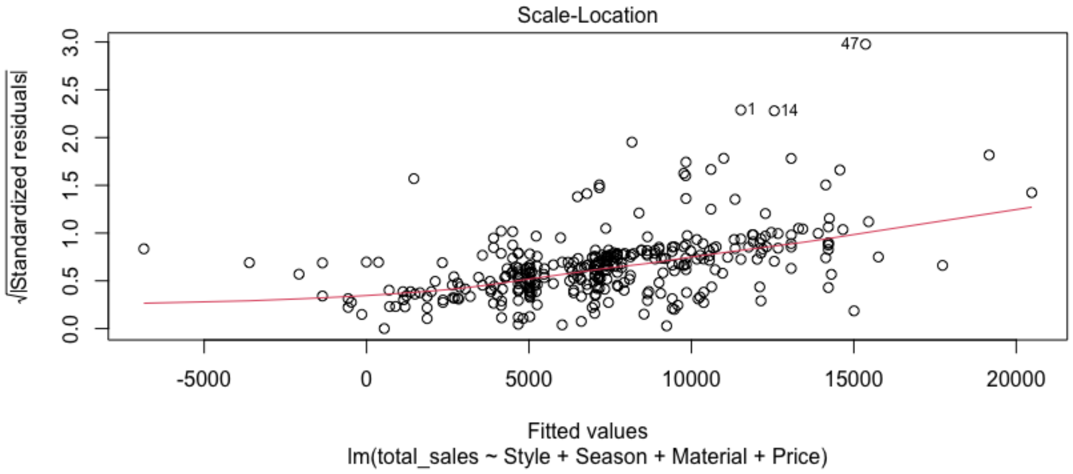
In this project, we built three classifier for dress recommendation prediction by use of Naive Bayes, Random Forest, and Support Vector Machine. Also, we used a Linear regression model to show the effect of dress attributes on the product price.

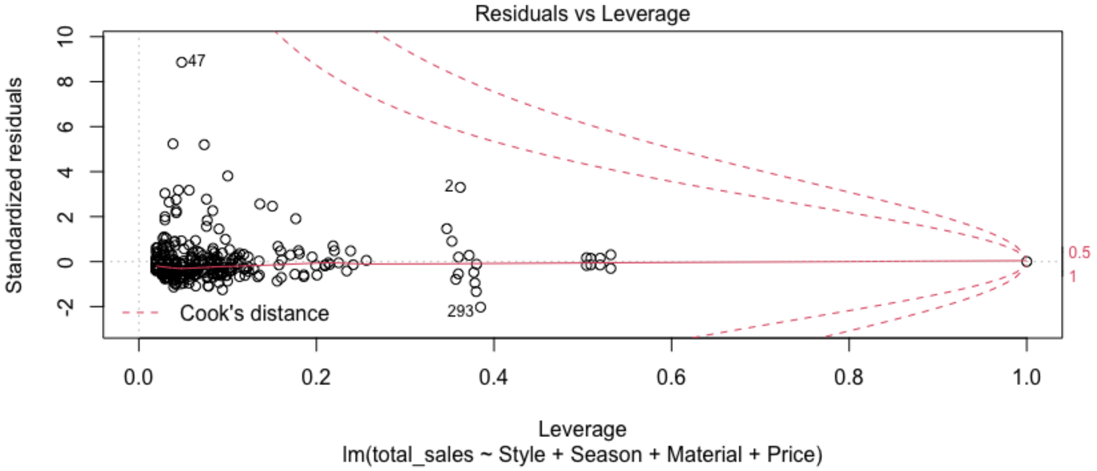
5. Result

The details of the results show in the code.


Residuals vs Fitted
Residuals
1e+05
5e+04
0e+00
-5000
0
5000
10000
15000
20000
47
1
14
Fitted values
lm(total_sales ~ Style + Season + Material + Price)


Normal Q-Q
Standardized residuals
10
8
6
4
2
0
-2
-3
-2
-1
0
1
2
3
47
14
1
Theoretical Quantiles
lm(total_sales ~ Style + Season + Material + Price)


Scale-Location
√|Standardized residuals|
3.0
2.5
2.0
1.5
1.0
0.5
0.0
-5000
0
5000
10000
15000
20000
47
1
14
Fitted values
lm(total_sales ~ Style + Season + Material + Price)


Residuals vs Leverage
Standardized residuals
10
8
6
4
2
0
-2
47
2
293
0.5
1
Cook's distance
0.0
0.2
0.4
0.6
0.8
1.0
Leverage
lm(total_sales ~ Style + Season + Material + Price)

My Code:{Solution}

```
> setwd("/Users/rushikeshkhankar/Desktop/R/1555052405_datasets")
> getwd()
[1] "/Users/rushikeshkhankar"
> library(e1071)
> library(caret)
Loading required package: lattice

Attaching package: ‘caret’

The following object is masked from ‘package:purrr’:

    lift

> library(readxl)
> library(plyr)
> library(caTools)
> library(randomForest)
> AttDS <- read_excel("Desktop/R/1555052405_datasets/Attribute DataSet.xlsx")
> Dress_Sales <- read_excel("Desktop/R/1555052405_datasets/Dress Sales.xlsx")
> View(AttDS)
> View(Dress_Sales)
> #remove dress id column
> AttDS_ = AttDS[2:14]
> Dress_Sales_ = Dress_Sales[2:24]
> View(Dress_Sales)
> #values checking
> #Style
> AttDS_$Style[AttDS_$Style == 'sexy'] = 'Sexy'
> #Price
> AttDS_$Price[AttDS_$Price == 'low'] = 'Low'
> AttDS_$Price[AttDS_$Price == 'high'] = 'High'
> #Size
> AttDS_$Size[AttDS_$Size == 's'] = 'S'
> AttDS_$Size[AttDS_$Size == 'small'] = 'S'

> #Season

> AttDS_$Season[AttDS_$Season == 'spring'] = 'Spring'

> AttDS_$Season[AttDS_$Season == 'summer'] = 'Summer'

> AttDS_$Season[AttDS_$Season == 'Automn'] = 'Autumn'

> AttDS_$Season[AttDS_$Season == 'winter'] = 'Winter'

> #NeckLine

> AttDS_$NeckLine[AttDS_$NeckLine == 'sweetheart'] = 'Sweetheart'

> #SleeveLenght

> AttDS_$SleeveLength[AttDS_$SleeveLength == 'sleevless'] = 'sleeveless'

> AttDS_$SleeveLength[AttDS_$SleeveLength == 'sleeevless'] = 'sleeveless'

> AttDS_$SleeveLength[AttDS_$SleeveLength == 'threequater'] = 'threequarter'

> AttDS_$SleeveLength[AttDS_$SleeveLength == 'thressqatar'] = 'threequarter'

> AttDS_$SleeveLength[AttDS_$SleeveLength == 'urndowncollor'] = 'turndowncollor'

> #Decoration

> AttDS_$Decoration[AttDS_$Decoration == 'embroidary'] = 'embroidery'

> AttDS_$Decoration[AttDS_$Decoration == 'sequined'] = 'sequins'

> AttDS_$Decoration[AttDS_$Decoration == 'ruched'] = 'ruche'

> AttDS_$Decoration[AttDS_$Decoration == 'none'] = 'null'

> #Pattern Type

> AttDS_$`Pattern Type`[AttDS_$`Pattern Type` == 'none'] = 'null'

> AttDS_$`Pattern Type`[AttDS_$`Pattern Type` == 'leapord'] = 'leopard'

> #factoring

> AttDS_$Style = factor(AttDS_$Style,

+                        levels =  c( 'bohemian', 'Brief','Casual','cute', 'fashion', 'Flare','Novelty','OL','party',
'Sexy','vintage', 'work'),

+                        labels =  c(0,1,2,3,4,5,6,7,8,9,10,11))

> AttDS_$Price = factor(AttDS_$Price,

+                          levels =  c('Low','Medium', 'Average','High','very-high'),

+                          labels =  c(0,1,2,3,4))

> AttDS_$Size = factor(AttDS_$Size,

+                         levels =  c('free', 'L' ,'M','S' ,'XL'),

+                         labels =  c(0,1,2,3,4))

> AttDS_$Season = factor(AttDS_$Season,

+                           levels =  c('Autumn', 'Spring', 'Summer', 'Winter'),
+                              labels =  c(0,1,2,3))
> AttDS_$NeckLine = factor(AttDS_$NeckLine,
+                                levels =  c("o-neck","v-neck","boat-neck","peterpan-collor","ruffled","turndowncollor","slash-neck","mandarin-collor","open", "sqare-collor","Sweetheart", "Scoop","halter","backless","bowneck","NULL" ),
+                                labels =  c(0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15))
> AttDS_$SleeveLength = factor(AttDS_$SleeveLength,
+                                    levels =  c("sleevless","Petal", "full","butterfly" ,"short","threequarter","halfsleeve","cap-sleeves","turndowncollor","capsleeves","half","NULL"  ),
+                                    labels =  c(0,1,2,3,4,5,6,7,8,9,10,11))
> AttDS_$waiseline = factor(AttDS_$waiseline,
+                                 levels =  c("empire","natural","null","princess","dropped" ),
+                                 labels =  c(0,1,2,3,4))
> AttDS_$Material = factor(AttDS_$Material,
+                                levels = c("null","microfiber","polyster","silk","chiffonfabric","cotton","nylon","other","milksilk","linen","rayon","lycra","mix","acrylic","spandex","lace","modal","cashmere","viscos","knitting","sill","wool","model","shiffon" ),
+                                labels =  c(0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23))
> AttDS_$FabricType = factor(AttDS_$FabricType,
+                                  levels = c("chiffon","null","broadcloth","jersey","other","batik","satin","flannel","worsted","woolen","poplin","dobby","knitted","tulle","organza","lace","Corduroy","terry" ),
+                                  labels =  c(0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17))
> AttDS_$Decoration = factor(AttDS_$Decoration,
+                                  levels = c("ruffles","null","embroidary","bow","lace","beading","sashes","hollowout","pockets","sequined" ,"applique","button","Tiered","rivet","feathers","flowers","pearls","pleat","crystal","ruched","draped","tassel","plain","cascading" ),
+                                  labels =  c(0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23))
> AttDS_$`Pattern Type` = factor(AttDS_$`Pattern Type`,
+                                      levels = c("animal","print","dot","solid","null","patchwork","striped","geometric","plaid","leopard","floral","character","splice","leapord","none" ),
+                                      labels =  c(0,1,2,3,4,5,6,7,8,9,10,11,12,13,14))
> AttDS_$Recommendation = sapply(AttDS_$Recommendation, factor)
> #count of missing values in attribset _ dataset
> colSums(is.na(AttDS_))
        Style        Price        Rating         Size        Season       NeckLine   SleeveLength
```

```
          0          2          0          0          2          1        232
     waiseline      Material     FabricType     Decoration   Pattern Type Recommendation
          1          1         20         23          1          0
> #create the function
> getmode <- function(v) {
+   uniqv <- unique(v)
+   uniqv[which.max(tabulate(match(v, uniqv)))]
+ }
> #Fill missing values with mode
> AttDS_$Price[is.na(AttDS_$Price) == TRUE] <- getmode(AttDS_$Price)
> AttDS_$Season[is.na(AttDS_$Season) == TRUE] <- getmode(AttDS_$Season)
> AttDS_$NeckLine[is.na(AttDS_$NeckLine) == TRUE] <- getmode(AttDS_$NeckLine)
> AttDS_$waiseline[is.na(AttDS_$waiseline) == TRUE] <- getmode(AttDS_$waiseline)
> AttDS_$Material[is.na(AttDS_$Material) == TRUE] <- getmode(AttDS_$Material)
> AttDS_$FabricType[is.na(AttDS_$FabricType) == TRUE] <- getmode(AttDS_$FabricType)
> AttDS_$Decoration[is.na(AttDS_$Decoration) == TRUE] <- getmode(AttDS_$Decoration)
> AttDS_$`Pattern Type`[is.na(AttDS_$`Pattern Type`) == TRUE] <- getmode(AttDS_$`Pattern
Type`)
> AttDS_data <- data.frame(AttDS_)
> str(AttDS_)
tibble [500 × 13] (S3: tbl_df/tbl/data.frame)
 $ Style         : Factor w/ 12 levels "0","1","2","3",..: 10 3 11 2 4 1 3 7 6 1 ...
 $ Price         : Factor w/ 5 levels "0","1","2","3",..: 1 1 4 3 1 1 3 3 3 1 ...
 $ Rating        : num [1:500] 4.6 0 0 4.6 4.5 0 0 0 0 0 ...
 $ Size          : Factor w/ 5 levels "0","1","2","3",..: 3 2 2 2 3 3 5 1 1 1 ...
 $ Season        : Factor w/ 4 levels "0","1","2","3": 3 3 1 2 3 3 3 1 2 3 ...
 $ NeckLine      : Factor w/ 16 levels "0","1","2","3",..: 1 1 1 1 1 2 1 1 2 2 ...
 $ SleeveLength  : Factor w/ 12 levels "0","1","2","3",..: NA 2 3 3 4 NA 3 5 5 NA ...
 $ waiseline     : Factor w/ 5 levels "0","1","2","3",..: 1 2 2 2 2 1 3 2 1 2 ...
 $ Material      : Factor w/ 24 levels "0","1","2","3",..: 1 2 3 4 5 1 6 3 6 7 ...
 $ FabricType    : Factor w/ 18 levels "0","1","2","3",..: 1 2 2 1 1 2 2 3 3 1 ...
 $ Decoration    : Factor w/ 24 levels "0","1","2","3",..: 1 1 2 2 4 2 2 5 6 2 ...
 $ Pattern Type  : Factor w/ 15 levels "0","1","2","3",..: 1 1 2 2 3 2 4 5 4 5 ...
 $ Recommendation: Factor w/ 2 levels "1","0": 1 2 2 1 2 2 2 2 1 1 ...
> Dress_Sales_ = rename(Dress_Sales_, c('41314'='2/9/2013'))
> Dress_Sales_ = rename(Dress_Sales_, c('41373'='4/9/2013'))

> Dress_Sales_ = rename(Dress_Sales_, c('41434'='6/9/2013'))

> Dress_Sales_ = rename(Dress_Sales_, c('41495'='8/9/2013'))

> Dress_Sales_ = rename(Dress_Sales_, c('41556'='10/9/2013'))

> Dress_Sales_ = rename(Dress_Sales_, c('41617'='12/9/2013'))

> Dress_Sales_ = rename(Dress_Sales_, c('41315'='2/10/2013'))

> Dress_Sales_ = rename(Dress_Sales_, c('41374'='4/10/2013'))

> Dress_Sales_ = rename(Dress_Sales_, c('41435'='6/10/2013'))

> Dress_Sales_ = rename(Dress_Sales_, c('40400'='8/10/2013'))

> Dress_Sales_ = rename(Dress_Sales_, c('41557'='10/10/2013'))

> Dress_Sales_ = rename(Dress_Sales_, c('41618'='12/10/2013'))

> #Covert all variable types to numeric

> Dress_Sales_ <- as.data.frame(apply(Dress_Sales_, 2, as.numeric))

Warning messages:

1: In apply(Dress_Sales_, 2, as.numeric) : NAs introduced by coercion

2: In apply(Dress_Sales_, 2, as.numeric) : NAs introduced by coercion

3: In apply(Dress_Sales_, 2, as.numeric) : NAs introduced by coercion

4: In apply(Dress_Sales_, 2, as.numeric) : NAs introduced by coercion

5: In apply(Dress_Sales_, 2, as.numeric) : NAs introduced by coercion

6: In apply(Dress_Sales_, 2, as.numeric) : NAs introduced by coercion

> #Mean row

> Dress_Sales_ = as.matrix(Dress_Sales_)

> k <- which(is.na(Dress_Sales_), arr.ind = TRUE)

> Dress_Sales_[k] <- rowMeans(Dress_Sales_, na.rm = TRUE)[k[,1]]

> Dress_Sales_ = as.data.frame(Dress_Sales_)

> #sum all values on row on (total sales)

> Dress_Sales_$total_sales = rowSums(Dress_Sales_)

> head(Dress_Sales_)

  29/8/2013 31/8/2013 2/9/2013 4/9/2013 6/9/2013 8/9/2013 10/9/2013 12/9/2013 14/9/2013
16/9/2013 18/9/2013

1      2114      2274     2491     2660     2727     2887      2930      3119      3204      3277      3321

2       151       275      570      750      813     1066      1164      1558      1756      1878      1985

3         6         7        7        7        8        8         9        10        10        10        10

4      1005      1128     1326     1455     1507     1621      1637      1723      1746      1783      1796

5       996      1175     1304     1396     1432     1559      1570      1638      1655      1681      1743
```

```
6     4     5    11    13    13    13    16    18    19    20    20
  20/9/2013 22/9/2013 24/9/2013 26/9/2013 28/9/2013 30/9/2013 2/10/2013 4/10/2013 6/10/2013
8/10/2013
1    3386    3479    3554    3624    3706    3746    3795    3832    3897    3923
2    2106    2454    2710    2942    3258    3354    3475    3654    3911    4024
3      10      11      11      11      11      11      11      11      11      11
4    1812    1845    1878    1892    1914    1924    1929    1941    1952    1955
5    1824    1919    2032    2156    2252    2312    2387    2459    2544    2614
6      21      22      25      25      26      26      26      26      27      27
  10/10/2013 12/10/2013 total_sales
1      3985       4048       75979
2      4125       4277       52256
3        11         11         223
4      1959       1963       39691
5      2693       2736       44077
6        27         27         457
> #merge data
> merged_data <- data.frame(AttDS_, Dress_Sales_)
> merged_data
   Style Price Rating Size Season NeckLine SleeveLength waiseline Material FabricType Decoration
1     9     0    4.6    2      2        0         <NA>         0        0          0          0
2     2     0    0.0    1      2        0            1         1        1          1          0
3    10     3    0.0    1      0        0            2         1        2          1          1
4     1     2    4.6    1      1        0            2         1        3          0          1
5     3     0    4.5    2      2        0            3         1        4          0          3
6     0     0    0.0    2      2        1         <NA>         0        0          1          1
7     2     2    0.0    4      2        0            2         2        5          1          1
8     6     2    0.0    0      0        0            4         1        2          2          4
9     5     2    0.0    0      1        1            4         0        5          2          5
10    0     0    0.0    0      2        1         <NA>         1        6          0          1
11    8     2    5.0    0      2        0            2         1        2          2          4
12    5     2    0.0    0      1        1            4         2        6          1          1
13    9     0    4.7    2      3        0            5         2        0          0          4
14   10     2    4.8    2      2        0            4         0        5          3          1
15    2     0    5.0    2      2        2            4         2        5          1          6
```

```
16   2   0   0.0   0   3    2        2     2    7     4     4
17   3   2   4.7   1   1    0        4     2    5     1     6
18   0   1   5.0   0   0    0        2     1    0     1     7
19   1   2   0.0   2   3    3        5     1    5     1     1
20   9   2   5.0   2   0    0     <NA>     0    8     1     1
21   9   2   4.5   1   0    0        2     2    5     1     5
22   2   0   4.3   2   2    0     <NA>     1    0     0     1
23   1   0   4.0   4   2    1        4     1    5     1     8
24   9   2   4.7   3   2    1     <NA>     0    5     1     1
25   9   2   0.0   0   0    1     <NA>     1    2     1     4
26   9   2   0.0   2   0    0     <NA>     0    5     2     1
27   9   2   4.7   2   1    0     <NA>     1    0     1     1
   Pattern.Type Recommendation X29.8.2013 X31.8.2013 X2.9.2013 X4.9.2013 X6.9.2013
X8.9.2013 X10.9.2013
1          0          1     2114     2274     2491     2660     2727     2887     2930
2          0          0      151      275      570      750      813     1066     1164
3          1          0        6        7        7        7        8        8        9
4          1          1     1005     1128     1326     1455     1507     1621     1637
5          2          0      996     1175     1304     1396     1432     1559     1570
6          1          0        4        5       11       13       13       13       16
7          3          0       45       61      131      165      176      209      216
8          4          0        4       13       55       73       76       89       94
9          3          1        5        6       10       12       13       15       16
10         4          1        9       11       12       12       12       12       12
11         3          0       15       28       42       49       49       55       56
12         0          0       23       38       54       59       65       72       77
13         1          1     1235     1333     1471     1568     1602     1722     1756
14         0          1     2498     2545     2627     2656     2669     2738     2769
15         3          0       22       28       40       43       44       45       45
16         4          0        4        8       19       25       26       31       35
17         3          1     1587     1669     1729     1764     1782     1817     1833
18         5          1       10       13       17       17       17       17       17
19         5          0        2        4       10       11       11       12       12
20         4          1      122      143      243      279      299      322      331
21         3          0      104      106      142      163      171      192      196
```

22	3	0	856	967	1057	1109	1124	1173	1198
23	3	0	145	189	219	229	229	233	233
24	3	1	1292	1319	1405	1479	1512	1578	1592
25	5	0	2	3	4	4	4	5	5
26	1	0	6	7	15	19	19	19	20
27	3	1	986	1048	1224	1358	1401	1571	1612

	X12.9.2013	X14.9.2013	X16.9.2013	X18.9.2013	X20.9.2013	X22.9.2013	X24.9.2013	X26.9.2013	X28.9.2013
1	3119	3204	3277	3321	3386	3479	3554	3624	3706
2	1558	1756	1878	1985	2106	2454	2710	2942	3258
3	10	10	10	10	10	11	11	11	11
4	1723	1746	1783	1796	1812	1845	1878	1892	1914
5	1638	1655	1681	1743	1824	1919	2032	2156	2252
6	18	19	20	20	21	22	25	25	26
7	251	262	289	295	325	353	379	400	417
8	117	121	125	135	142	150	160	165	169
9	17	17	19	21	25	27	27	28	29
10	13	13	13	13	14	14	14	14	14
11	65	66	68	68	78	89	93	98	106
12	90	102	106	117	122	133	153	162	170
13	1906	1994	2077	2105	2192	2284	2395	2423	2537
14	2934	3001	3068	3125	3239	3339	3445	3561	3667
15	49	50	51	51	52	53	52	52	52
16	45	48	58	74	88	97	120	132	158
17	1898	1909	1954	2005	2096	2120	2163	2241	2301
18	17	17	17	17	18	18	18	19	19
19	12	13	13	13	13	13	13	13	16
20	381	401	413	442	458	470	496	517	539
21	227	247	258	264	285	291	316	244	365
22	1248	1284	1300	1326	1366	1388	1431	1482	1533
23	233	233	233	233	233	233	233	233	233
24	1671	1689	1738	1778	1820	1905	1962	1946	2077
25	5	5	5	5	5	5	5	5	5
26	22	22	23	23	24	25	25	26	26
27	1762	1852	1901	2041	2115	2217	2358	2452	2580

```
   X30.9.2013 X2.10.2013 X4.10.2013 X6.10.2013 X8.10.2013 X10.10.2013 X12.10.2013 total_sales
1        3746       3795       3832       3897       3923        3985        4048       75979
2        3354       3475       3654       3911       4024        4125        4277       52256
3          11         11         11         11         11          11          11         223
4        1924       1929       1941       1952       1955        1959        1963       39691
5        2312       2387       2459       2544       2614        2693        2736       44077
6          26         26         26         27         27          27          27         457
7         424        445        462        474        494         503         552        7328
8         172        196        212        224        241         252         263        3248
9          29         29         29         29         29          29          33         494
10         14         14         14         14         14          14          14         300
11        114        119        124        128        130         133         138        1911
12        172        176        181        185        188         195         202        2842
13       2580       2645       2704       2748       2813        2914        2979       49983
14       3680       3745       3842       3952       3999        4077        4170       75346
15         52         52         52         52         52          53          53        1095
16        163        174        182        195        204         213         220        2319
17       2325       2375       2421       2479       2492        2512        2526       47998
18         19         19         19         19         19          19          19         401
19         16         17         17         17         17          17          17         299
20        542        562        575        596        602         621         644        9998
21        368        369        370        372        374         375         376        6175
22       1564       1572       1621       1682       1699        1726        1761       31467
23        233        233        234        235        235         236         236        5216
24       2095       2115       2179       2243       2286        2314        2352       42347
25          5          5          5          5          5           5           5         107
26         27         27         28         28         29          29          29         518
27       2621       2678       2792       2882       2923        2995        3071       48440
 [ reached 'max' / getOption("max.print") -- omitted 473 rows ]
> str(merged_data)
'data.frame':	500 obs. of  37 variables:
 $ Style         : Factor w/ 12 levels "0","1","2","3",..: 10 3 11 2 4 1 3 7 6 1 ...
 $ Price         : Factor w/ 5 levels "0","1","2","3",..: 1 1 4 3 1 1 3 3 3 1 ...
 $ Rating        : num  4.6 0 0 4.6 4.5 0 0 0 0 0 ...
$ Size          : Factor w/ 5 levels "0","1","2","3",..: 3 2 2 2 3 3 5 1 1 1 ...
$ Season        : Factor w/ 4 levels "0","1","2","3": 3 3 1 2 3 3 3 1 2 3 ...
$ NeckLine      : Factor w/ 16 levels "0","1","2","3",..: 1 1 1 1 1 2 1 1 2 2 ...
$ SleeveLength  : Factor w/ 12 levels "0","1","2","3",..: NA 2 3 3 4 NA 3 5 5 NA ...
$ waiseline     : Factor w/ 5 levels "0","1","2","3",..: 1 2 2 2 2 1 3 2 1 2 ...
$ Material      : Factor w/ 24 levels "0","1","2","3",..: 1 2 3 4 5 1 6 3 6 7 ...
$ FabricType    : Factor w/ 18 levels "0","1","2","3",..: 1 2 2 1 1 2 2 3 3 1 ...
$ Decoration    : Factor w/ 24 levels "0","1","2","3",..: 1 1 2 2 4 2 2 5 6 2 ...
$ Pattern.Type  : Factor w/ 15 levels "0","1","2","3",..: 1 1 2 2 3 2 4 5 4 5 ...
$ Recommendation: Factor w/ 2 levels "1","0": 1 2 2 1 2 2 2 2 1 1 ...
$ X29.8.2013    : num  2114 151 6 1005 996 ...
$ X31.8.2013    : num  2274 275 7 1128 1175 ...
$ X2.9.2013     : num  2491 570 7 1326 1304 ...
$ X4.9.2013     : num  2660 750 7 1455 1396 ...
$ X6.9.2013     : num  2727 813 8 1507 1432 ...
$ X8.9.2013     : num  2887 1066 8 1621 1559 ...
$ X10.9.2013    : num  2930 1164 9 1637 1570 ...
$ X12.9.2013    : num  3119 1558 10 1723 1638 ...
$ X14.9.2013    : num  3204 1756 10 1746 1655 ...
$ X16.9.2013    : num  3277 1878 10 1783 1681 ...
$ X18.9.2013    : num  3321 1985 10 1796 1743 ...
$ X20.9.2013    : num  3386 2106 10 1812 1824 ...
$ X22.9.2013    : num  3479 2454 11 1845 1919 ...
$ X24.9.2013    : num  3554 2710 11 1878 2032 ...
$ X26.9.2013    : num  3624 2942 11 1892 2156 ...
$ X28.9.2013    : num  3706 3258 11 1914 2252 ...
$ X30.9.2013    : num  3746 3354 11 1924 2312 ...
$ X2.10.2013    : num  3795 3475 11 1929 2387 ...
$ X4.10.2013    : num  3832 3654 11 1941 2459 ...
$ X6.10.2013    : num  3897 3911 11 1952 2544 ...
$ X8.10.2013    : num  3923 4024 11 1955 2614 ...
$ X10.10.2013   : num  3985 4125 11 1959 2693 ...
$ X12.10.2013   : num  4048 4277 11 1963 2736 ...
$ total_sales   : num  75979 52256 223 39691 44077 ...
```

```
> #splitting dataset
> set.seed(100)
> spl = sample.split(merged_data$Recommendation, SplitRatio = 0.7)
> train = subset(merged_data, spl==TRUE)
> test = subset(merged_data, spl==FALSE)
> print(dim(train)) ; print(dim(test))
[1] 350  37
[1] 150  37
> #Naive Bayes Model
> #build model
> Naive_Model = naiveBayes(Recommendation ~., data = train)
> #Create Confusion Matrix
> confusionMatrix(train$Recommendation, predict(Naive_Model, train), positive = '1')
Confusion Matrix and Statistics

          Reference
Prediction   1   0
         1 108  39
         0  70 133

               Accuracy : 0.6886
                 95% CI : (0.6372, 0.7367)
    No Information Rate : 0.5086
    P-Value [Acc > NIR] : 6.449e-12

                  Kappa : 0.3788

 Mcnemar's Test P-Value : 0.00406

            Sensitivity : 0.6067
            Specificity : 0.7733
         Pos Pred Value : 0.7347
         Neg Pred Value : 0.6552
             Prevalence : 0.5086
         Detection Rate : 0.3086
   Detection Prevalence : 0.4200
      Balanced Accuracy : 0.6900

       'Positive' Class : 1

> #predict test set
> Naive_Predict = predict(Naive_Model, test)
> table(Naive_Predict, test$Recommendation)

Naive_Predict  1  0
            1 35 34
            0 28 53
> #regression
> regress_Sales = lm(formula = total_sales ~ Style+Season+Material+Price, data = train)
> summary(regress_Sales)

Call:
lm(formula = total_sales ~ Style + Season + Material + Price,
    data = train)

Residuals:
   Min     1Q Median     3Q    Max
-19936  -6113  -2230   1381 108508

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   3703.0     4044.0   0.916   0.3605
Style1        4763.6     4824.1   0.987   0.3242
Style2        2317.9     3465.8   0.669   0.5041
Style3        2543.0     4098.6   0.620   0.5354
Style4       -2163.5    13067.0  -0.166   0.8686
Style5       -4908.1    13127.9  -0.374   0.7088
Style6         152.3     6713.6   0.023   0.9819
```

```
Style8        3040.3     4162.6   0.730   0.4657
Style9        7057.2     3738.5   1.888   0.0600 .
Style10        9843.9     4594.4   2.143   0.0329 *
Style11        3802.6     4893.5   0.777   0.4377
Season1        3455.8     2427.4   1.424   0.1555
Season2         761.5     2339.6   0.325   0.7450
Season3         233.9     2382.8   0.098   0.9219
Material1   12382.7     7516.0   1.648   0.1005
Material2      790.2     2243.6   0.352   0.7249
Material3   -2454.3     3316.7  -0.740   0.4599
Material4     3595.3     3141.2   1.145   0.2533
Material5   -2111.8     1884.6  -1.121   0.2633
Material6   -3568.7     5518.6  -0.647   0.5183
Material7   -2426.9     9130.3  -0.266   0.7906
Material8   -2429.2     9068.5  -0.268   0.7890
Material9     2010.3     7588.1   0.265   0.7912
Material10   -6025.8     5056.8  -1.192   0.2343
Material11    -637.5     7475.0  -0.085   0.9321
Material12   -1602.8     4285.8  -0.374   0.7087
Material13   -2853.4     7500.8  -0.380   0.7039
Material14   -7213.0    12932.0  -0.558   0.5774
Material15   -2242.1    12843.7  -0.175   0.8615
Material16   -6416.6    12731.2  -0.504   0.6146
Material17   -5257.7     9108.9  -0.577   0.5642
Material19   -1842.8    12767.8  -0.144   0.8853
Material20   -7843.1    12790.6  -0.613   0.5402
Material23   -3215.7     9269.8  -0.347   0.7289
Price1       -4172.0     3571.0  -1.168   0.2436
Price2         355.2     1627.8   0.218   0.8274
Price3       -2030.6     3912.1  -0.519   0.6041
Price4       -8335.1     4055.2  -2.055   0.0407 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

```
Residual standard error: 12550 on 312 degrees of freedom
Multiple R-squared:  0.09296,	Adjusted R-squared:  -0.01461 
F-statistic: 0.8642 on 37 and 312 DF,  p-value: 0.6971

> plot(regress_Sales)
Hit <Return> to see next plot: #Regression (total sales n ratings)
Hit <Return> to see next plot: regress_Rating = lm(formula = total_sales ~ Rating, data = train)
Hit <Return> to see next plot: summary(regress_Rating)
Hit <Return> to see next plot: plot(regress_Rating, pch = 16, col = "blue")
Warning messages:
1: not plotting observations with leverage one:
  8, 153, 162, 202, 257, 271 
2: In sqrt(crit * p * (1 - hh)/hh) : NaNs produced
3: In sqrt(crit * p * (1 - hh)/hh) : NaNs produced
```