**Analysis of Sales Report of a Clothes Manufacturing Outlet**

**Source Code:**

library(readxl)

setwd('F:/Data science project/Analysis of Sales Report clothing')

getwd()

attribset <- read\_excel('Attribute DataSet.xlsx')

attribset1 <- attribset[2:14]

dresssale <- read\_excel('Dress Sales.xlsx')

dresssale1 <- dresssale[2:24]

#rename

library(plyr)

dresssale1 <- rename(dresssale1, c("41314"="2/9/2013"))

dresssale1 <- rename(dresssale1, c("41373"="4/9/2013"))

dresssale1 <- rename(dresssale1, c("41434"="6/9/2013"))

dresssale1 <- rename(dresssale1, c("41495"="8/9/2013"))

dresssale1 <- rename(dresssale1, c("41556"="10/9/2013"))

dresssale1 <- rename(dresssale1, c("41617"="12/9/2013"))

dresssale1 <- rename(dresssale1, c("41315"="2/10/2013"))

dresssale1 <- rename(dresssale1, c("41374"="4/10/2013"))

dresssale1 <- rename(dresssale1, c("41435"="6/10/2013"))

dresssale1 <- rename(dresssale1, c("40400"="8/10/2013"))

dresssale1 <- rename(dresssale1, c("41557"="10/10/2013"))

dresssale1 <- rename(dresssale1, c("41618"="12/10/2013"))

dresssale1[8:13] <- data.frame(sapply(dresssale1[8:13], function(x) as.numeric(as.character(x))))

#mean row wise

as.matrix(dresssale1)

k <- which(is.na(dresssale1), arr.ind=TRUE)

dresssale1[k] <- rowMeans(dresssale1, na.rm=TRUE)[k[,1]]

as.data.frame(dresssale1)

##Total sales

Totalsale<-rowSums(dresssale1[1:23])

dresssale1<-data.frame(dresssale1, Totalsale)

attribset1$Style[attribset1$Style =='sexy'] <- 'Sexy' #manipulating sexy to Sexy

attribset1$Price[attribset1$Price =='high'] <- 'High'

attribset1$Price[attribset1$Price =='low'] <- 'Low'

attribset1$Size[attribset1$Size =='s'] <- 'S'

attribset1$Size[attribset1$Size =='small'] <- 'S'

attribset1$Season[attribset1$Season =='Automn'] <- 'Autumn'

attribset1$Season[attribset1$Season =='spring'] <- 'Spring'

attribset1$Season[attribset1$Season =='summer'] <- 'Summer'

attribset1$Season[attribset1$Season =='winter'] <- 'Winter'

attribset1$NeckLine[attribset1$NeckLine =='sweetheart'] <- 'Sweetheart'

attribset1$SleeveLength[attribset1$SleeveLength =='sleeevless'] <- 'sleevless'

attribset1$SleeveLength[attribset1$SleeveLength =='sleveless'] <- 'sleevless'

attribset1$SleeveLength[attribset1$SleeveLength =='sleeveless'] <- 'sleevless'

attribset1$SleeveLength[attribset1$SleeveLength =='threequater'] <- 'threequarter'

attribset1$SleeveLength[attribset1$SleeveLength =='thressqatar'] <- 'threequarter'

attribset1$SleeveLength[attribset1$SleeveLength =='urndowncollor'] <- 'turndowncollor'

attribset1$FabricType[attribset1$FabricType =='shiffon'] <- 'chiffon'

attribset1$FabricType[attribset1$FabricType =='sattin'] <- 'satin'

attribset1$FabricType[attribset1$FabricType =='wollen'] <- 'woolen'

attribset1$FabricType[attribset1$FabricType =='flannael'] <- 'flannel'

attribset1$FabricType[attribset1$FabricType =='knitting'] <- 'knitted'

attribset1$Decoration[attribset1$Decoration =='none'] <- 'null'

attribset1$`Pattern Type`[attribset1$`Pattern Type` =='none'] <- 'null'

attribset1$`Pattern Type`[attribset1$`Pattern Type` =='leopard'] <- 'leapord'

#Factoring

attribset1$Style = factor(attribset1$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))

attribset1$Price = factor(attribset1$Price,

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

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

attribset1$Size = factor(attribset1$Size,

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

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

attribset1$Season = factor(attribset1$Season,

levels = c('Autumn', 'Spring', 'Summer', 'Winter'),

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

attribset1$NeckLine = factor(attribset1$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))

attribset1$SleeveLength = factor(attribset1$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))

attribset1$waiseline = factor(attribset1$waiseline,

levels = c("empire","natural","null","princess","dropped" ),

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

attribset1$Material = factor(attribset1$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))

attribset1$FabricType = factor(attribset1$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))

attribset1$Decoration = factor(attribset1$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))

attribset1$`Pattern Type` = factor(attribset1$`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))

#Missing Value with mode

attribset1$Price[is.na(attribset1$Price) ==TRUE] <- 2

attribset1$Season[is.na(attribset1$Season) ==TRUE] <- 2

attribset1$NeckLine[is.na(attribset1$NeckLine) ==TRUE] <- 0

attribset1$waiseline[is.na(attribset1$waiseline) ==TRUE] <- 1

attribset1$Material[is.na(attribset1$Material) ==TRUE] <- 5

attribset1$FabricType[is.na(attribset1$FabricType) ==TRUE] <- 1

attribset1$Decoration[is.na(attribset1$Decoration) ==TRUE] <- 1

attribset1$`Pattern Type`[is.na(attribset1$`Pattern Type`) ==TRUE] <- 3

attribset1$SleeveLength[is.na(attribset1$SleeveLength) ==TRUE] <- 0

mergedset <- data.frame(attribset1, dresssale1)

#split data into test set and trainin set

install.packages('caTools')

library(caTools)

set.seed(123)

split = sample.split(mergedset$Recommendation, SplitRatio = 0.80)

training\_set = subset(mergedset, split == TRUE)

test\_set = subset(mergedset, split == FALSE)

#convert data frame to numeric

training\_set <- data.frame(sapply(training\_set, function(x) as.numeric(as.character(x))))

test\_set <- data.frame(sapply(test\_set, function(x) as.numeric(as.character(x))))

#Feature Scaling

training\_set[-13] = scale(training\_set[-13])

test\_set[-13] = scale(test\_set[-13])

#Multiple Linear Regrression for how the style, season, and material affect the sales of a dress

regressor = lm(formula = Totalsale ~ Style+Season+Material+Price,

data = training\_set)

summary(regressor)

# Price is more influential than style on sales

#Multiple Linear Regrression for atributes affecting sales

regressor = lm(formula = Totalsale ~ . ,

data = training\_set[-13:-36])

regressor = lm(formula = Totalsale ~ .-Material-Style-FabricType-NeckLine-Size-Pattern.Type-Decoration ,

data = training\_set[-13:-36])

summary(regressor)

#Linear regression for finding effect of rating on total sales

library(caTools)

set.seed(123)

split = sample.split(mergedset$Recommendation, SplitRatio = 0.80)

lin\_training\_set = subset(mergedset, split == TRUE)

lin\_test\_set = subset(mergedset, split == FALSE)

regressor = lm(formula = Totalsale ~ Rating,

data = lin\_training\_set)

y\_pred = predict(regressor, newdata = lin\_test\_set)

library(ggplot2)

ggplot() +

geom\_point(aes(x = lin\_training\_set$Rating, y = lin\_training\_set$Totalsale),

colour = 'red') +

geom\_line(aes(x = lin\_training\_set$Rating, y = predict(regressor, newdata = lin\_training\_set)),

colour = 'blue') +

ggtitle('Rating vs Totalsales (Training set)') +

xlab('Rating') +

ylab('TotalSales')

ggplot() +

geom\_point(aes(x = lin\_test\_set$Rating, y = lin\_test\_set$Totalsale),

colour = 'red') +

geom\_line(aes(x = lin\_test\_set$Rating, y = predict(regressor, newdata = lin\_test\_set)),

colour = 'blue') +

ggtitle('Rating vs Totalsales (Test set)') +

xlab('Rating') +

ylab('TotalSales')

#Random Forest for prediciting Recomendation

install.packages('randomForest')

library(randomForest)

set.seed(123)

classifier = randomForest(x = training\_set[-13],

y = training\_set$Recommendation,

ntree =800)

## Random forest prediction

y\_pred = predict(classifier, newdata = test\_set[-13])

y\_pred = ifelse(y\_pred > 0.5, 1, 0)

cm = table(test\_set[, 13], y\_pred )