Comparing Dress Sales Classifiers

Joong Ho Choi
joonghoc

Contents

Introduction: **1**

Exploratory Data Analysis: **1**
Background and Variables . . . 1
Summary of the Response Labels in the Training Dataset . . . 2
Univariate Exploration . . . 2
Bivariate Exploration . . . 14
Some EDA on relationships between recommendation and the categorical variables . . . 14

Modeling **24**
Linear Discriminant Analysis (LDA) . . . 24
Quadratic Discriminant Analysis (QDA) . . . 24
Classification Trees . . . 25
Binary Logistic Regression . . . 26
Final Recommendation . . . 27

Discussion **27**

```
######################
### Loading the data
######################
dress_train <- readr::read_csv("http://stat.cmu.edu/~gordonw/dress_train.csv")
dress_test <- readr::read_csv("http://stat.cmu.edu/~gordonw/dress_test.csv")
```

Introduction:

In fashion industry, it is paramount to keep up with the latest trends when deciding what kind of clothes to make and sell. Stocking up on clothes, which do not correspond to any style demand,does not only waste resources and incur opportunity costs, but also will result in losses. Therefore, as it is for other industries, it is important to understand the market's demands in fashion industry.

In this paper, we will train and evaluate machine learning classification techniques for predicting the sales of dresses, based on various aspects of dresses. Then, we will decide which classification model would be most desirable, if we were to choose only one for predicting whether a dress will sell well or not.

Exploratory Data Analysis:

Background and Variables

We have data from the store's last year sales to train our classifiers, and we will use also this available data to verify our classifiers' accuracy. We will pick the best model to be used next year to decide if a dress with pre-determiend characteristics should be included in next year's collection.

In the data, we have the following predictor variables:

- `Style`: dress style (cute, work, casual, fashion, party)
- `Price`: price range (low, average, high)
- `Rating`: average customer rating from dress factory market survey (average of stars, 0-5)
- `Season`: which season is the dress appropriate for (summer, fall, winter, spring)
- `NeckLine`: type of neckline (O-neck, V-neck, other)
- `Material`: if it is a cotton dress or not
- `Decoration`: if it has any decoration or not
- `Pattern`: if the fabric has a pattern (yes) or of it's a solid color (no)
- `Sleeve`: if the dress has a sleeve
- `Waistline`: type of waistline (other, empire, natural)

and our response labels that we want to predict with our classifiers:

- `Recommendation`: binary outcome if the dress sells well (1) or not (0).

Summary of the Response Labels in the Training Dataset

We first note that in the training set, we have 303 observations. We have 158 clothes selling well, comprising 45.53% of the clothes. On the other, we have 189 clothes not selling well, comprising 54.47% of the clothes, as shown in the following tables:

```
table(dress_train$Recommendation)
```

```
##
##   0   1
## 189 158
```

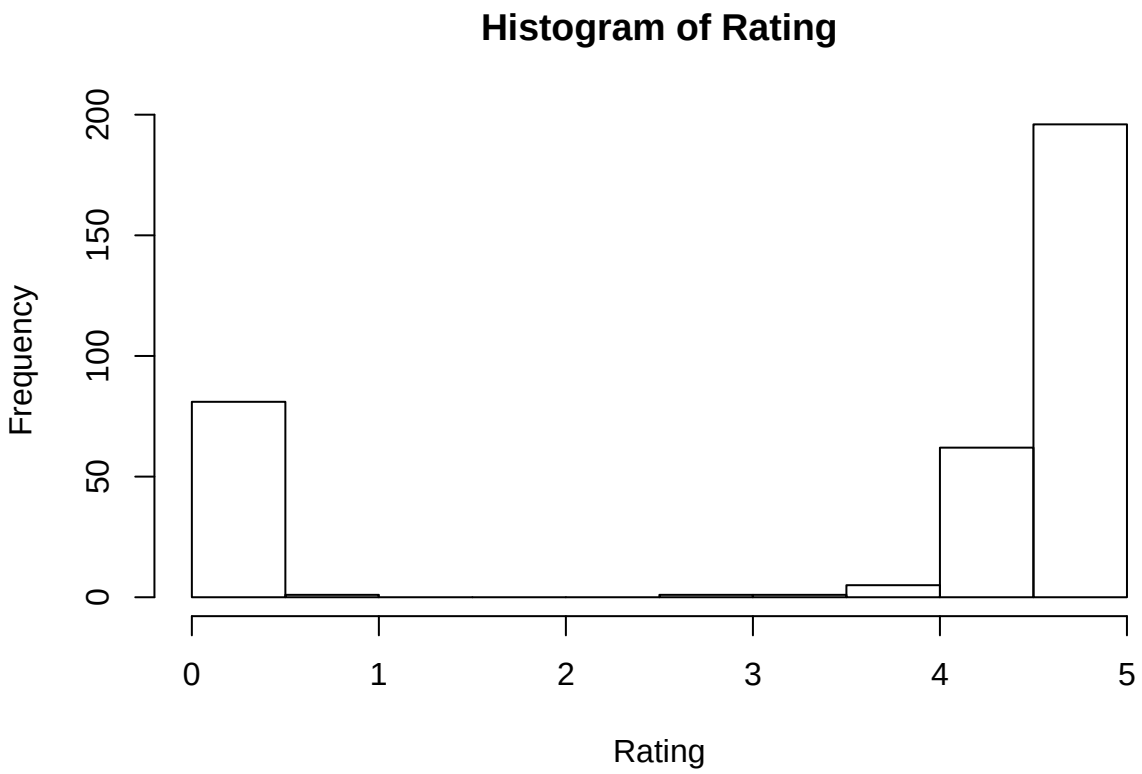
```
prop.table(table(dress_train$Recommendation))
```

```
##
##         0         1
## 0.5446686 0.4553314
```

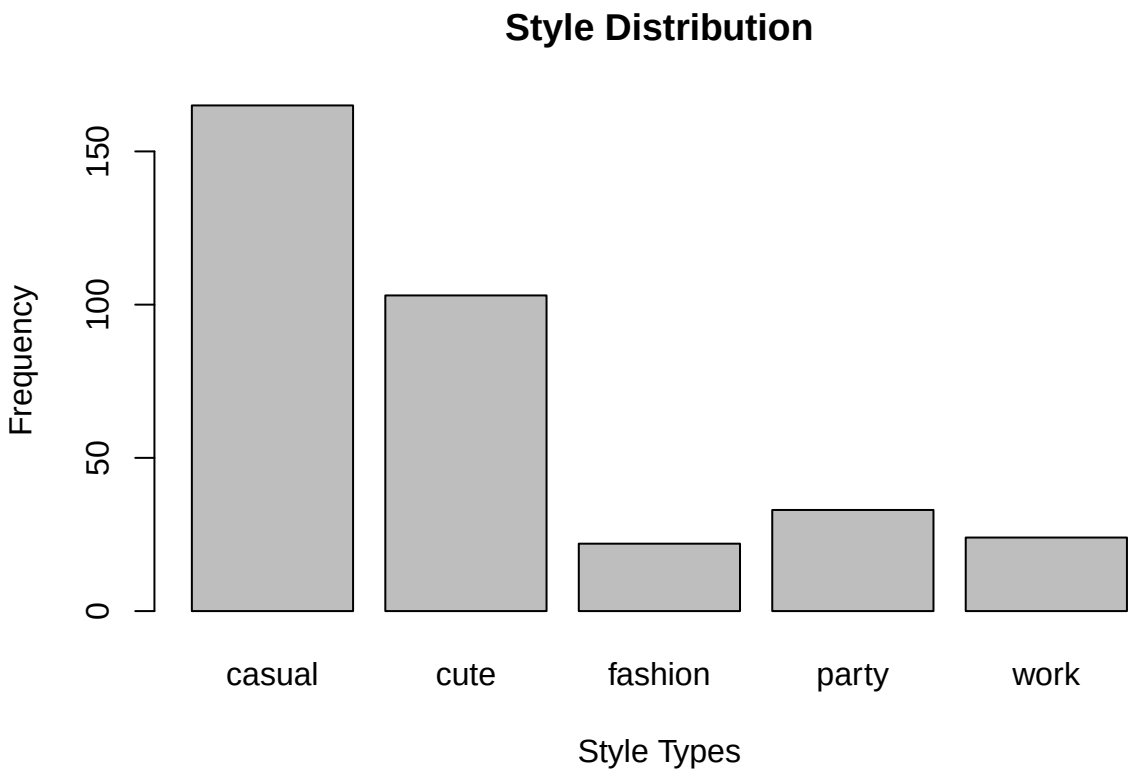
Univariate Exploration

As a first step in the analysis, we explore each variable individually. We use a histogram to explore the distribution of continuous variable and barplots to explore our categorical variables.

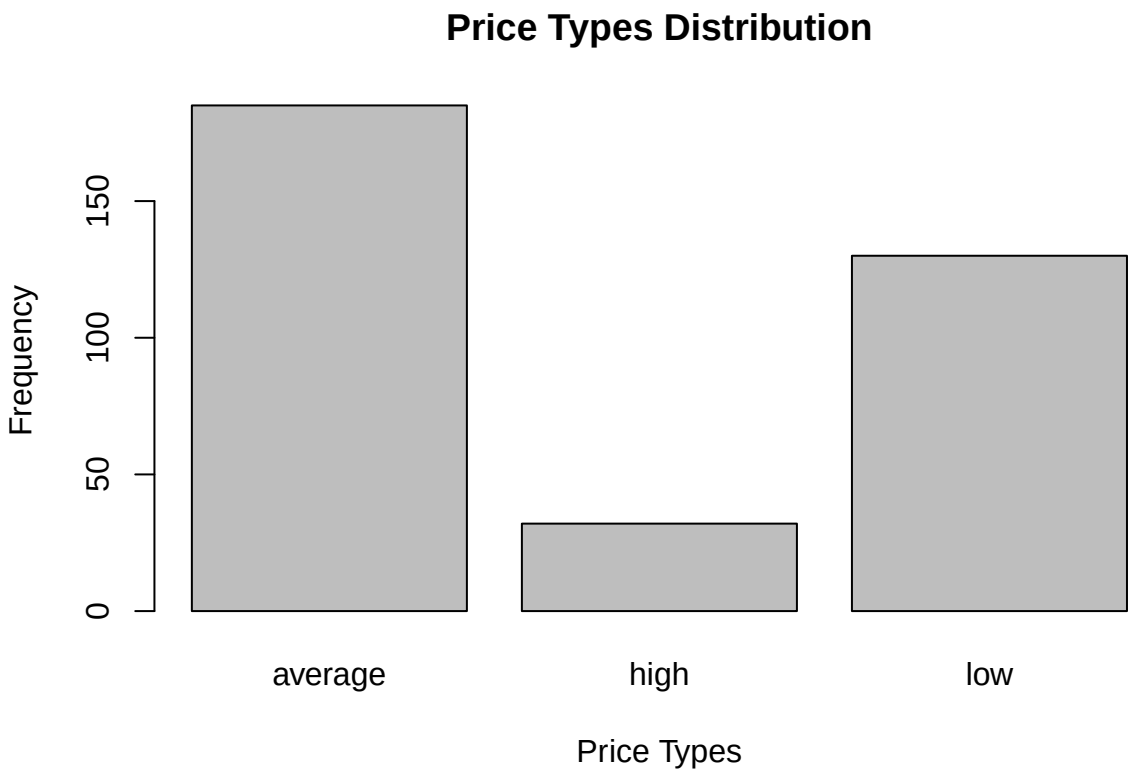
```
hist(dress_train$Rating,
     main = "Histogram of Rating",
     xlab = "Rating",
     ylab = "Frequency"
     )
```



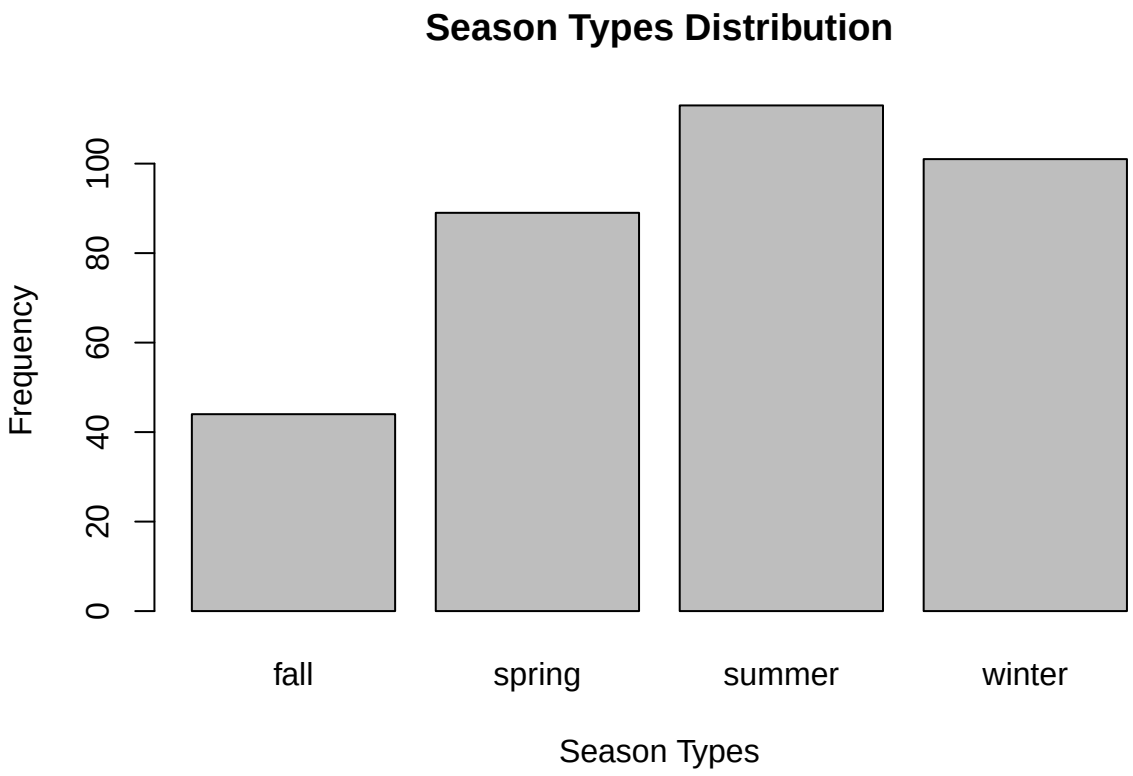

```
barplot(table(dress_train$Style),main="Style Distribution",
        xlab = "Style Types", ylab = "Frequency")
```



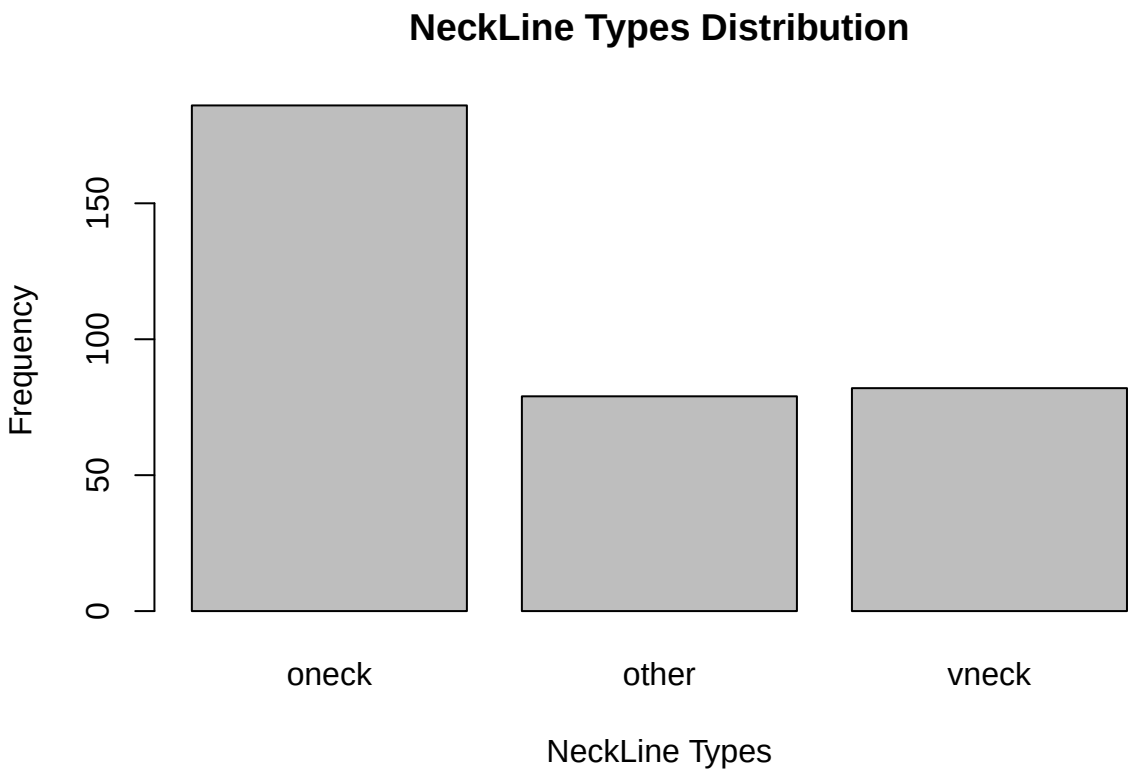

```
barplot(table(dress_train$Price),main="Price Types Distribution",
        xlab = "Price Types", ylab = "Frequency")
```



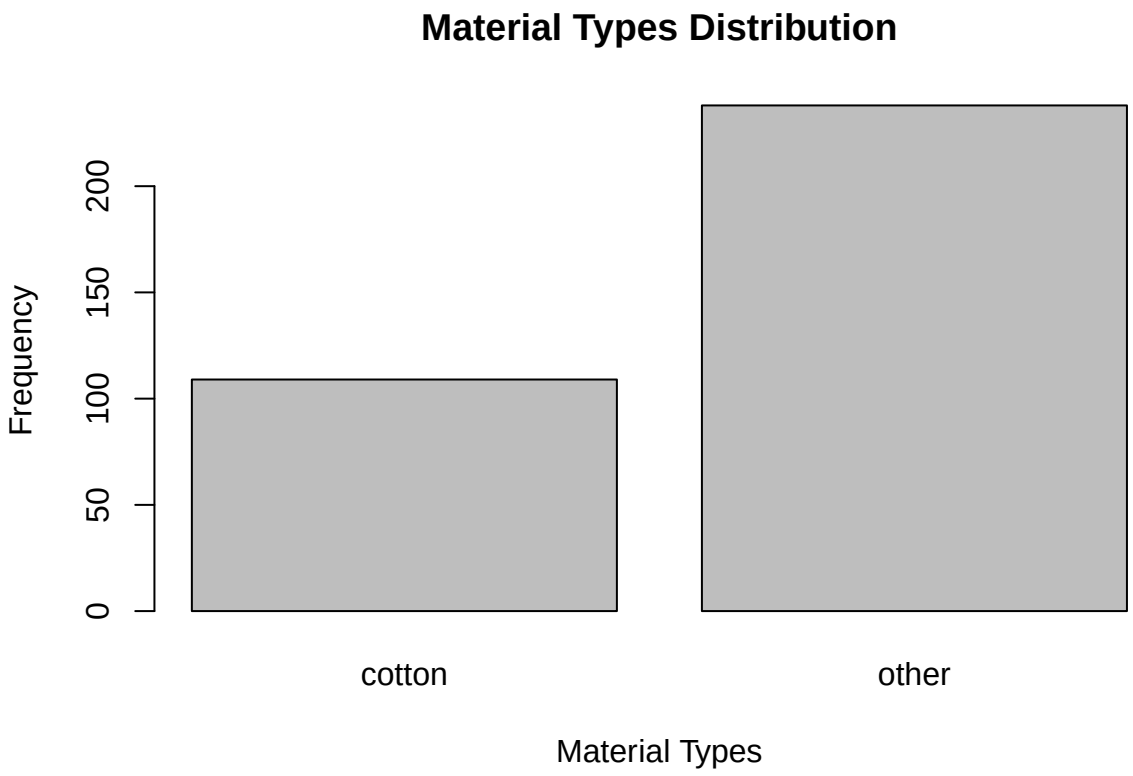

```
barplot(table(dress_train$Season),main="Season Types Distribution",
        xlab = "Season Types", ylab = "Frequency")
```



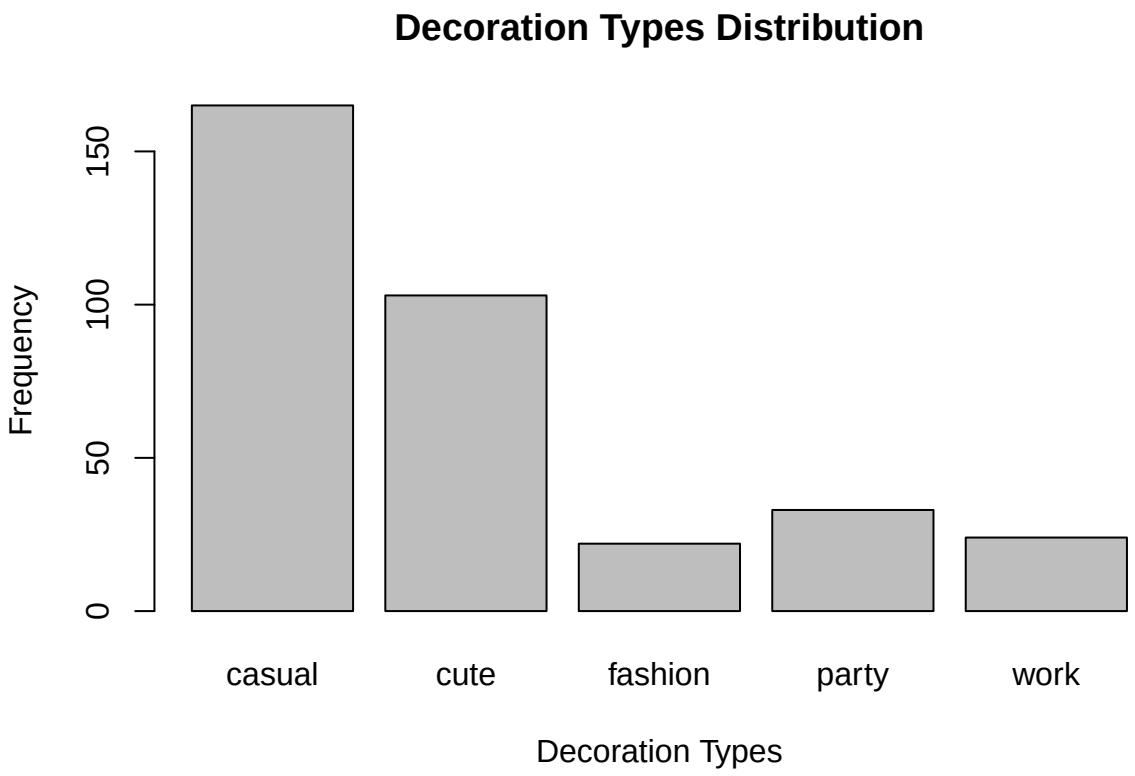

```
barplot(table(dress_train$NeckLine),main="NeckLine Types Distribution",
        xlab = "NeckLine Types", ylab = "Frequency")
```



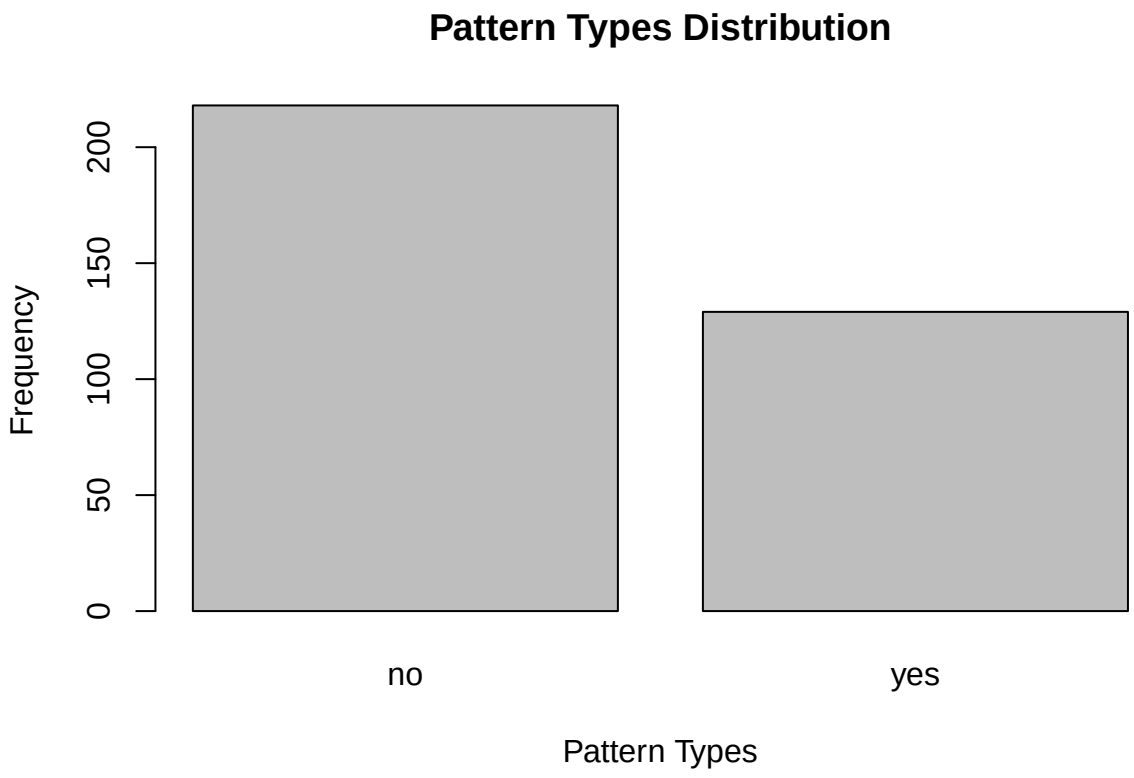

```
barplot(table(dress_train$Material),main="Material Types Distribution",
        xlab = "Material Types", ylab = "Frequency")
```



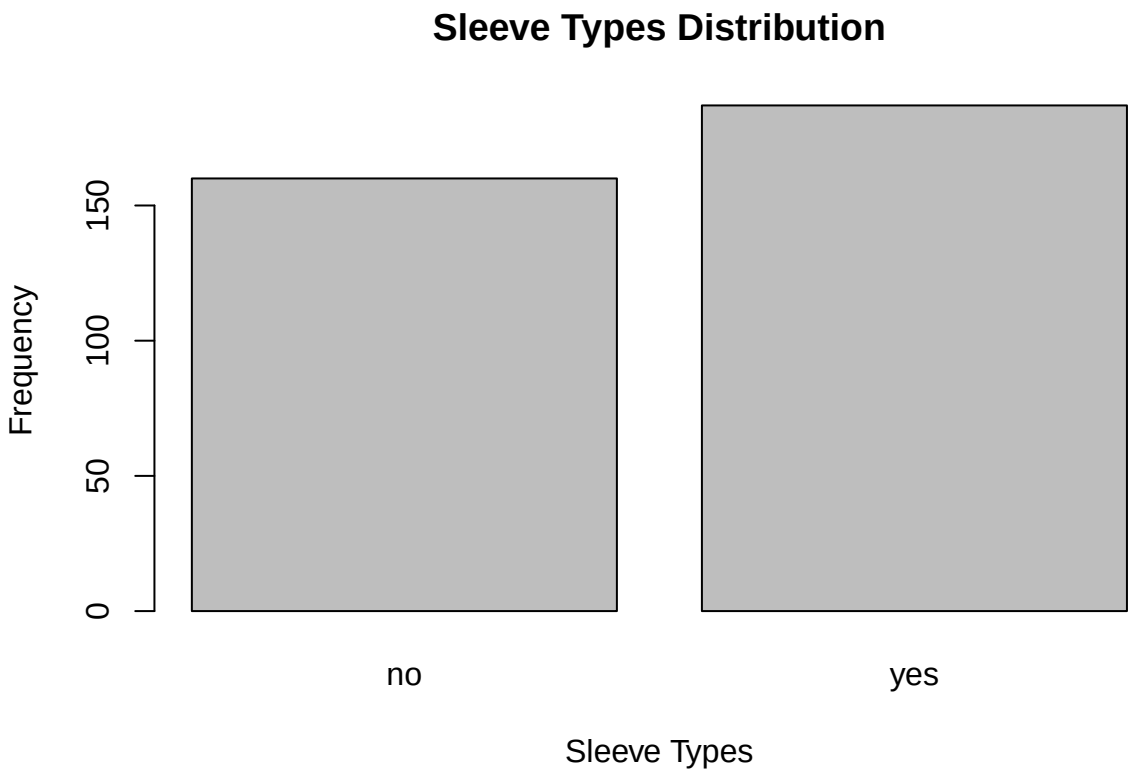

```
barplot(table(dress_train$Style),main="Decoration Types Distribution",
        xlab = "Decoration Types", ylab = "Frequency")
```



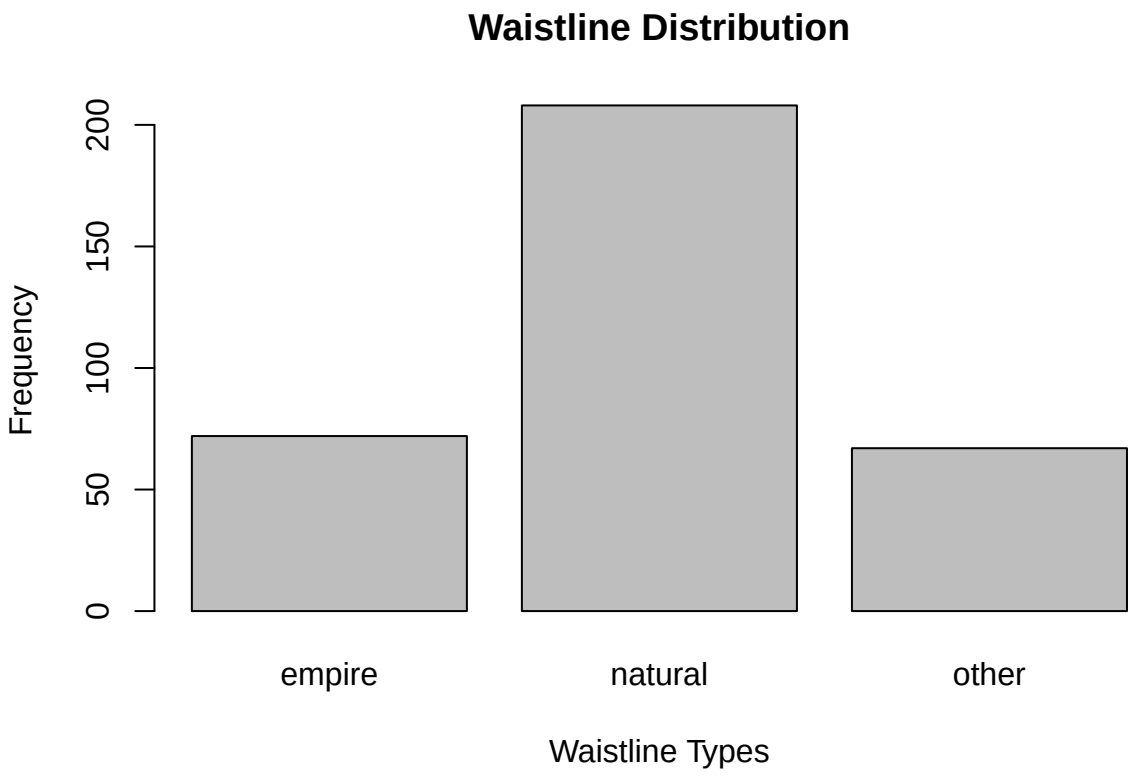

```
barplot(table(dress_train$Pattern),main="Pattern Types Distribution",
        xlab = "Pattern Types", ylab = "Frequency")
```




```
barplot(table(dress_train$Sleeve),main="Sleeve Types Distribution",
        xlab = "Sleeve Types", ylab = "Frequency")
```




```
barplot(table(dress_train$Waistline),main="Waistline Distribution",
        xlab = "Waistline Types", ylab = "Frequency")
```




```
summary(dress_train$Rating)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   4.000   4.600   3.563   4.800   5.000
```

```
table(dress_train$Style)
```

```
##
##  casual    cute fashion   party    work
##     165     103      22      33      24
```

```
table(dress_train$Price)
```

```
##
## average    high     low
##     185      32     130
```

```
table(dress_train$Season)
```

```
##
##   fall spring summer winter
##     44     89    113    101
```

```
table(dress_train$NeckLine)
```

```
##
## oneck other vneck
##   186    79    82
```

```r
table(dress_train$Material)
```

```
##
## cotton  other
##    109    238
```

```r
table(dress_train$Decoration)
```

```
##
##  no yes
## 160 187
```

```r
table(dress_train$Pattern)
```

```
##
##  no yes
## 218 129
```

```r
table(dress_train$Sleeve)
```

```
##
##  no yes
## 160 187
```

```r
table(dress_train$Waistline)
```

```
##
##  empire natural   other
##      72     208      67
```

From the visualizations, summary and tables above, we note the followings:

1.The graph is uni-modal and was skewed to the left. The frequency was the highest when the rating was between 4.5 and 5. The median rating was 4.6, while the mean was 3.563.

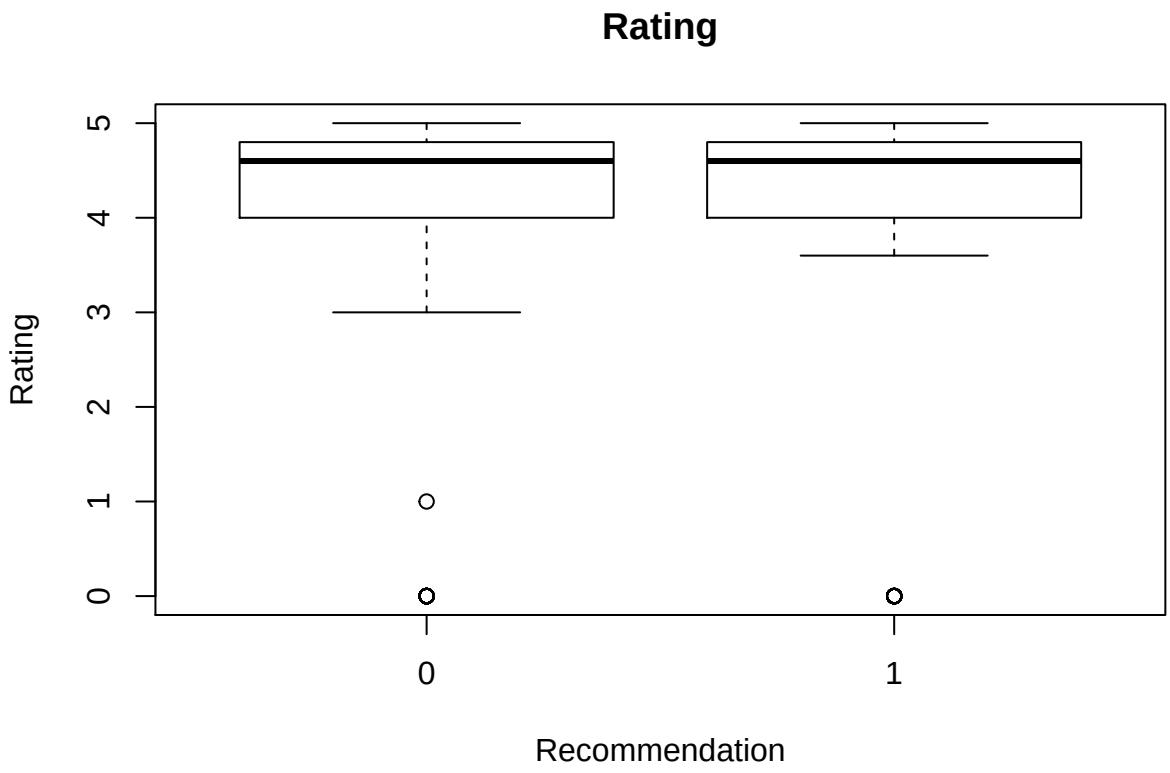
2. While casual style takes the highest proportion of the training data set, fashion takes the smallest proportion.
3. While average price type takes the highest proportion of the training data set, high price type takes the smallest proportion.
4. While summer season type takes the highest proportion of the training data set, fall season type takes the smallest proportion.
5. While O-neck Neckline type takes the highest proportion of the training data set, other neckline type takes the smallest proportion.
6. While other material type takes the highest proportion of the training data set, cotton material type takes the smallest proportion.
7. While casual decoration takes the highest proportion of the training data set, fashion decoration takes the smallest proportion.
8. Clothes with no pattern takes higher proportion of the training data set than the clothes with patterns.
9. Clothes with sleeve takes higher proportion of the training data set than the clothes with no sleeve.
10. While natural waistline type takes the highest proportion of the training data set, other waistline type takes the smallest proportion.

Bivariate Exploration

Now that we understand the distribution of the individual variables in this data, we can graphically how each predictor is associated with the response Recommendation.

We first move toward visualizing the relationship between the response (Recommendation) and the only quantiative predictor(Rating).For visually exploring whether we expect the quantitative predictor to be useful in helping to classify the recommendation of dresses, we show a boxplot, which appear as follows:

```
boxplot(Rating~Recommendation,main="Rating",data = dress_train)
```



In the above boxplots, we note that if the boxplots show differences between the two types of recommendation, we have some evidence of a relationship and a variable that might be useful in our classifiers (although note that this is not the same as a statistically significant relationship). With that in mind, we note that there seems to be similar median rating; although there's a slightly wider spread/variance for the recommendation 0, both remmendations are roughly the same. We note that for both boxplots, there are a few outliers at the ratings of 0 and 1.

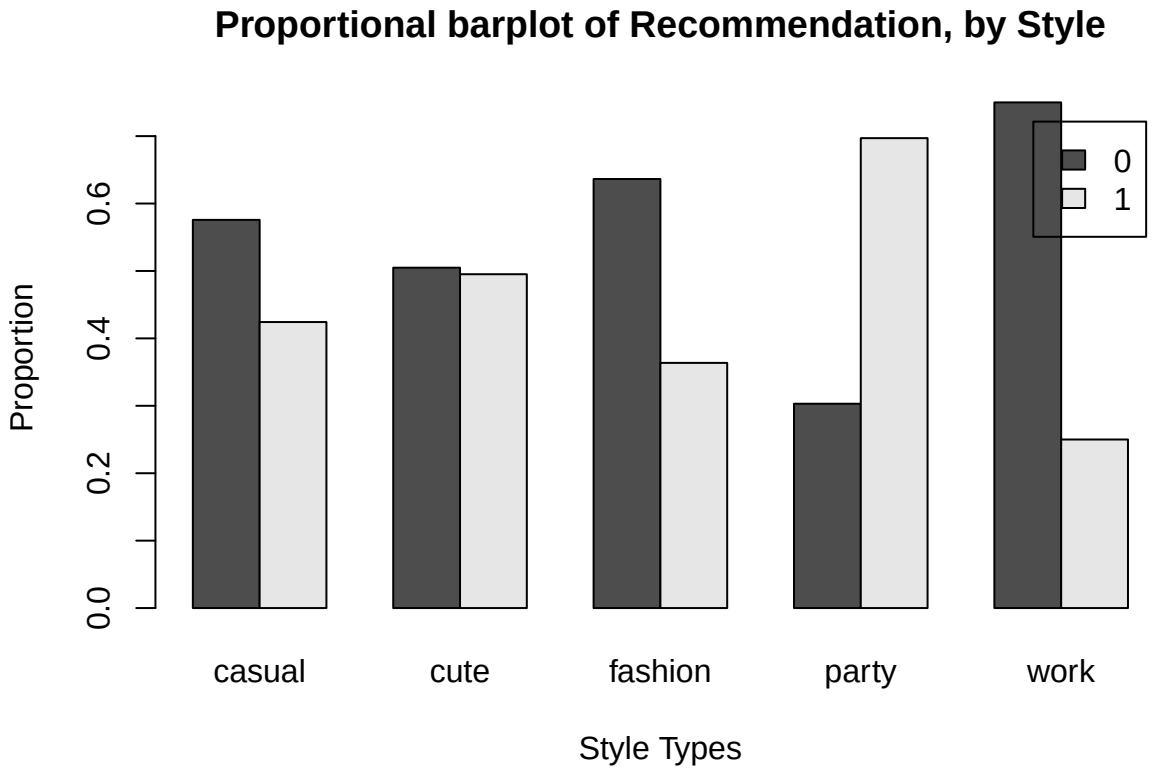
Some EDA on relationships between recommendation and the categorical variables

To explore the relationship between type and the categorical predictor quality, we can look at the conditional proportions of recommendation, conditioned on the categorical variables, shown as follows (we note that recommendation is when Recommendation=1 (white column) and no recommendation is when Recommendation=0(black column)):

```
prop.table(table(dress_train$Recommendation, dress_train$Style),margin = 2)
```

```
##
##        casual      cute   fashion     party      work
##   0 0.5757576 0.5048544 0.6363636 0.3030303 0.7500000
##   1 0.4242424 0.4951456 0.3636364 0.6969697 0.2500000
```

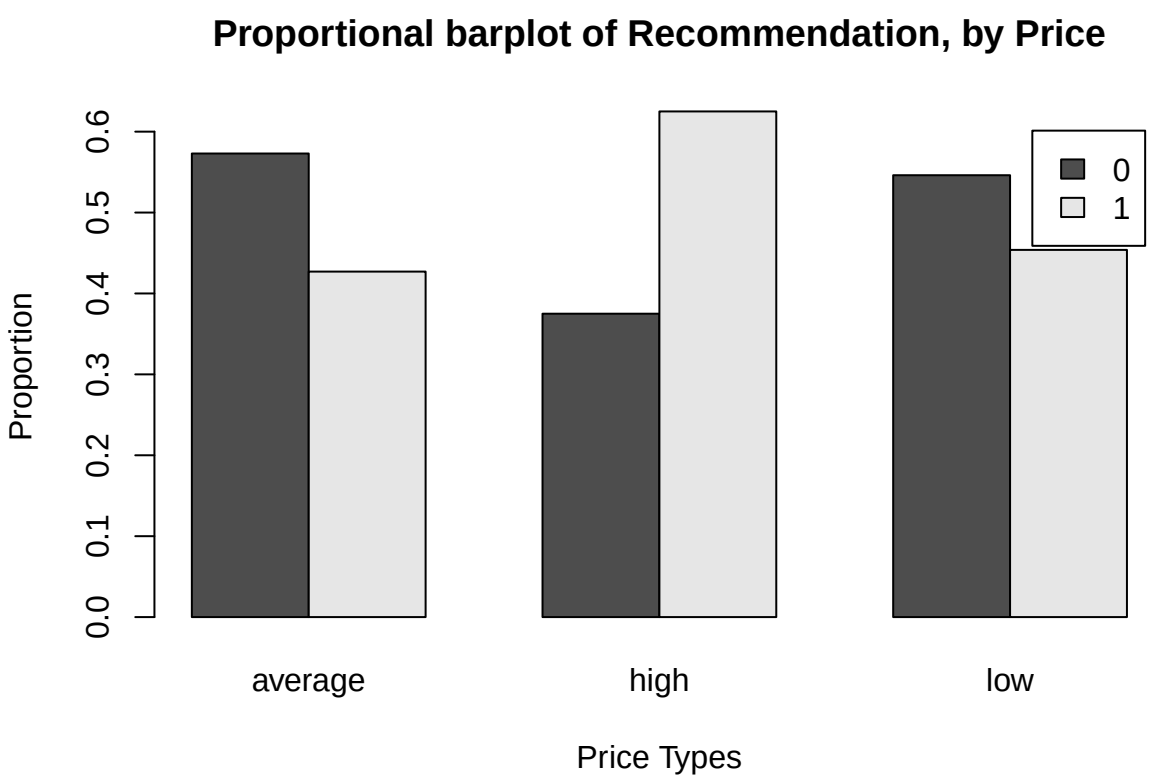
```
barplot(prop.table(table(dress_train$Recommendation, dress_train$Style),
margin = 2), beside = TRUE,xlab="Style Types",ylab="Proportion",
main = "Proportional barplot of Recommendation, by Style",legend=TRUE)
```




```
prop.table(table(dress_train$Recommendation, dress_train$Price),margin = 2)
```

```
##
##       average      high       low
##   0 0.5729730 0.3750000 0.5461538
##   1 0.4270270 0.6250000 0.4538462
```

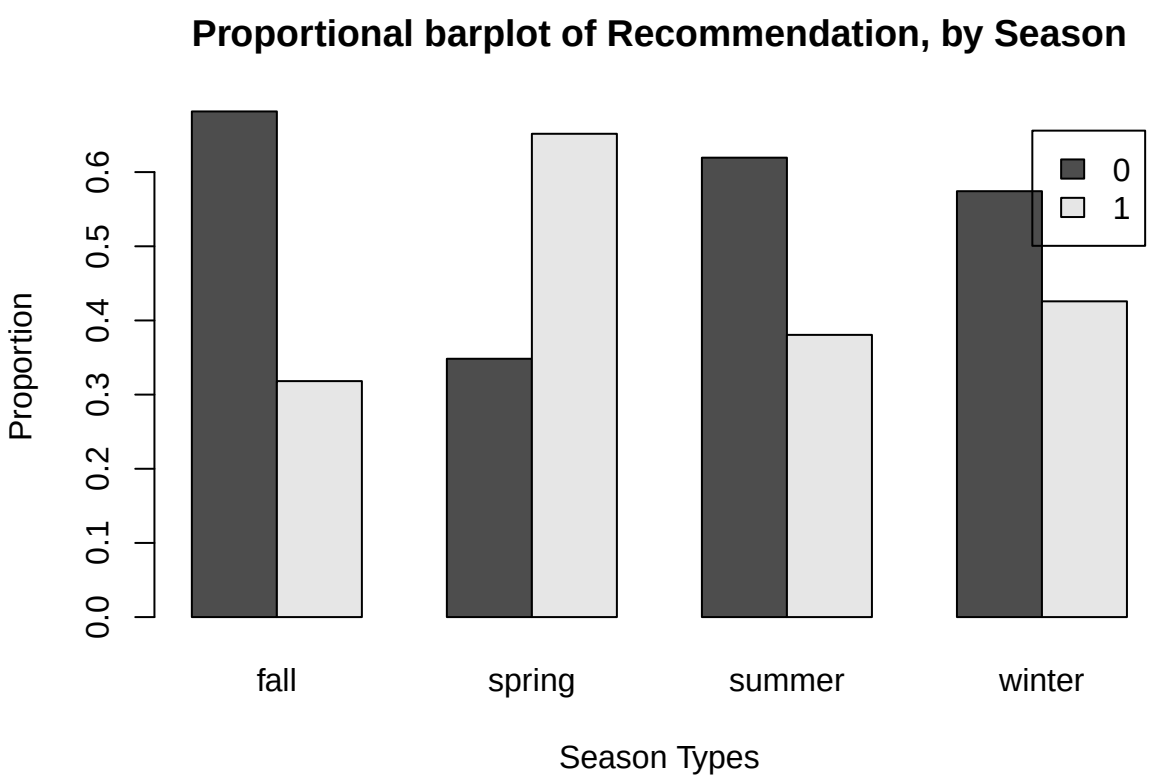
```
barplot(prop.table(table(dress_train$Recommendation, dress_train$Price),
margin = 2), beside = TRUE,xlab="Price Types",ylab="Proportion",
main = "Proportional barplot of Recommendation, by Price",legend=TRUE)
```




```
prop.table(table(dress_train$Recommendation, dress_train$Season),margin = 2)
```

```
##
##          fall    spring    summer    winter
##   0 0.6818182 0.3483146 0.6194690 0.5742574
##   1 0.3181818 0.6516854 0.3805310 0.4257426
```

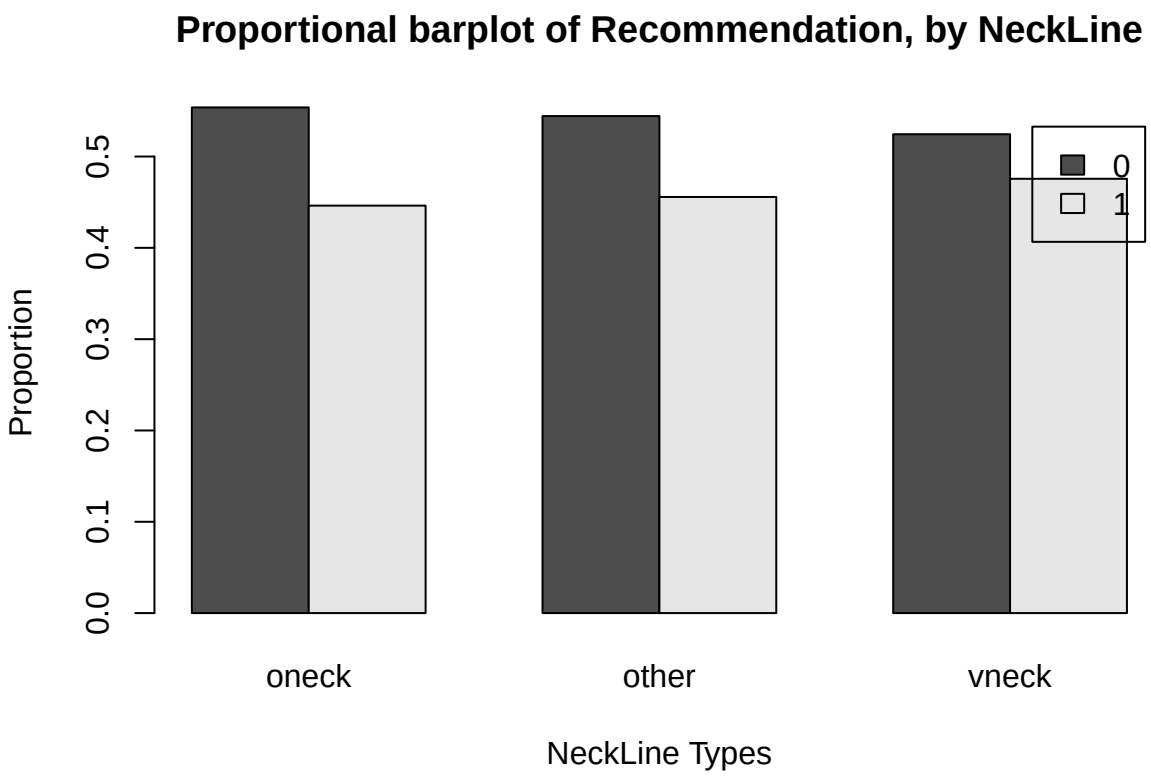
```
barplot(prop.table(table(dress_train$Recommendation, dress_train$Season),
margin = 2), beside = TRUE,xlab="Season Types",ylab="Proportion",
main = "Proportional barplot of Recommendation, by Season",legend=TRUE)
```




```
prop.table(table(dress_train$Recommendation, dress_train$NeckLine),margin = 2)
```

```
##
##         oneck     other     vneck
##   0 0.5537634 0.5443038 0.5243902
##   1 0.4462366 0.4556962 0.4756098
```

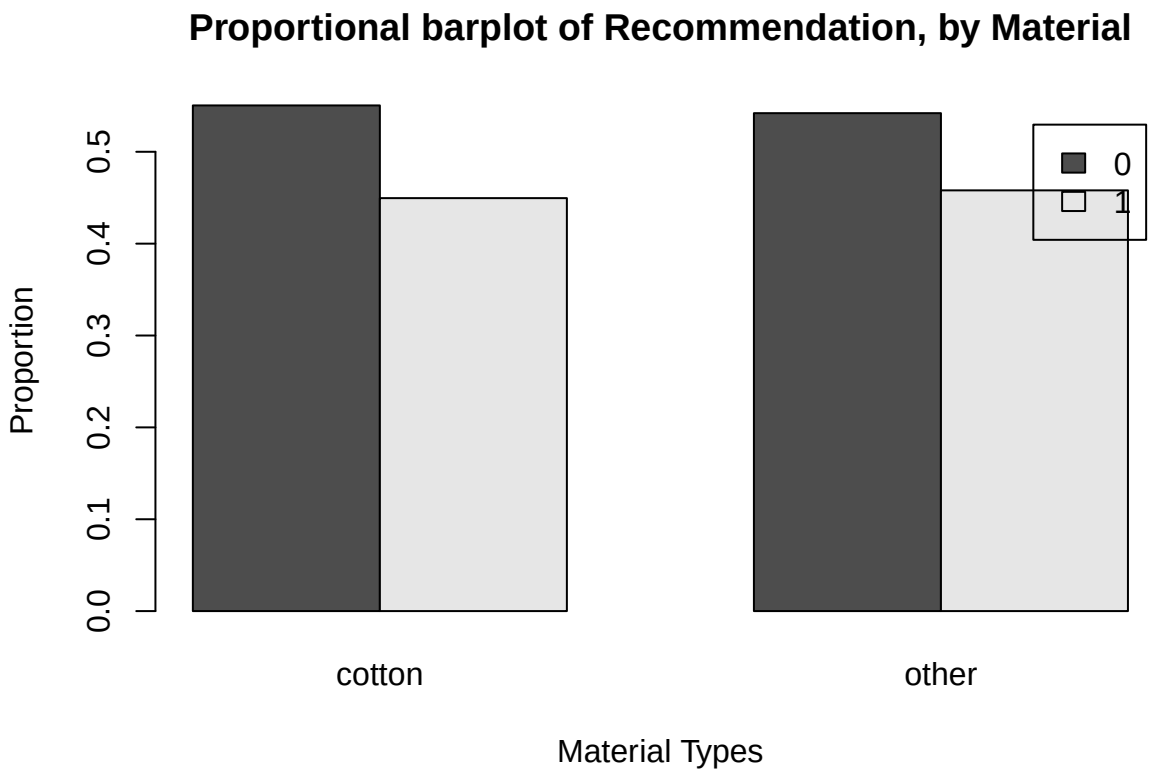
```
barplot(prop.table(table(dress_train$Recommendation, dress_train$NeckLine),
margin = 2), beside = TRUE,xlab="NeckLine Types",ylab="Proportion",
main = "Proportional barplot of Recommendation, by NeckLine",legend=TRUE)
```




```
prop.table(table(dress_train$Recommendation, dress_train$Material),margin = 2)
```

```
##
##        cotton     other
##   0 0.5504587 0.5420168
##   1 0.4495413 0.4579832
```

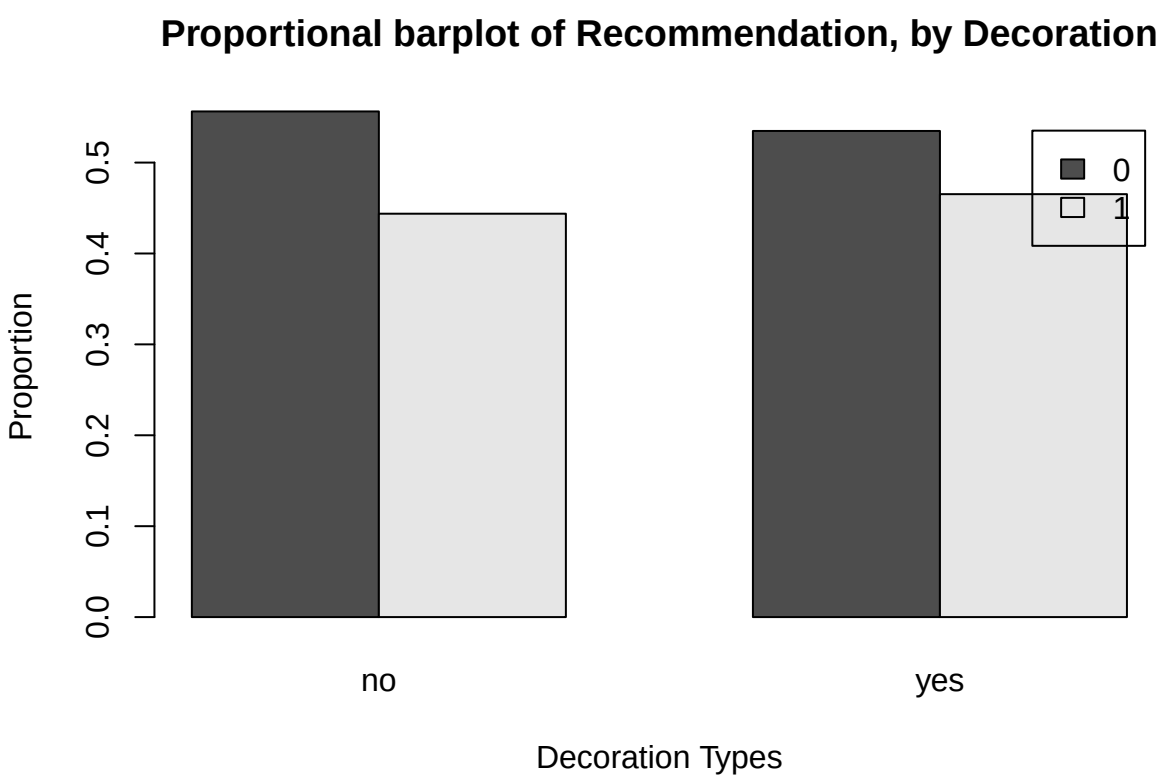
```
barplot(prop.table(table(dress_train$Recommendation, dress_train$Material),
margin = 2), beside = TRUE,xlab="Material Types",ylab="Proportion",
main = "Proportional barplot of Recommendation, by Material",legend=TRUE)
```




```
prop.table(table(dress_train$Recommendation, dress_train$Decoration),margin = 2)
```

```
##
##            no       yes
##   0 0.5562500 0.5347594
##   1 0.4437500 0.4652406
```

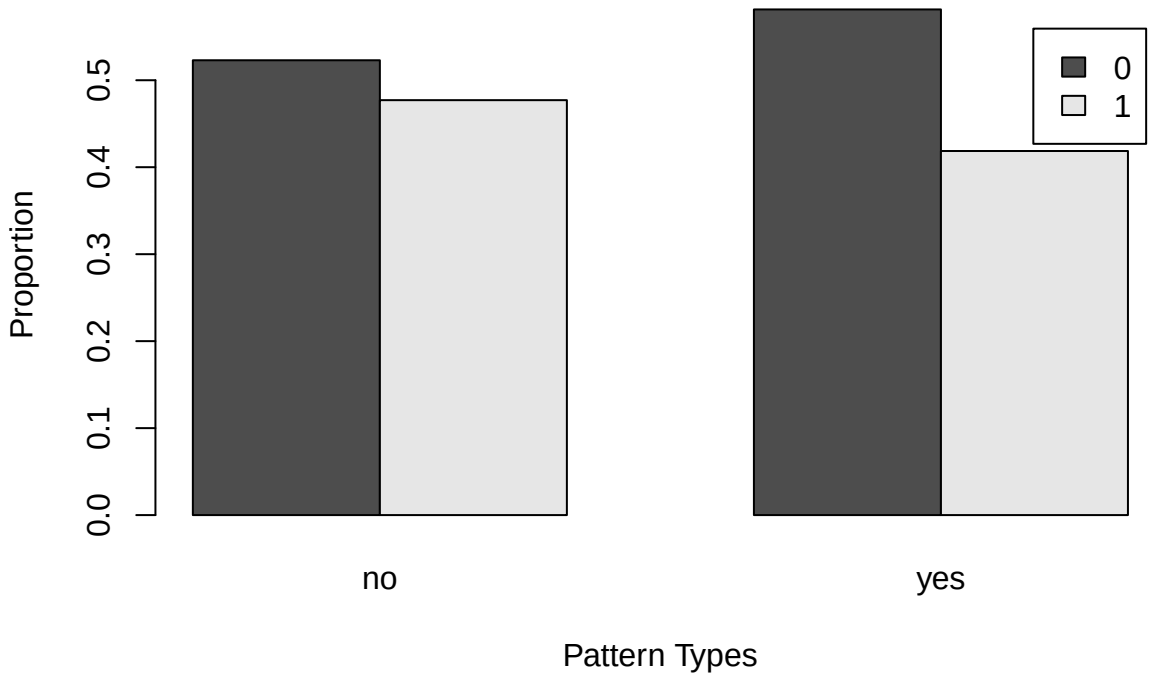
```
barplot(prop.table(table(dress_train$Recommendation, dress_train$Decoration),
margin = 2), beside = TRUE,xlab="Decoration Types",ylab="Proportion",
main = "Proportional barplot of Recommendation, by Decoration",legend=TRUE)
```




```
prop.table(table(dress_train$Recommendation, dress_train$Pattern),margin = 2)
```

```
##
##            no       yes
##   0 0.5229358 0.5813953
##   1 0.4770642 0.4186047
```

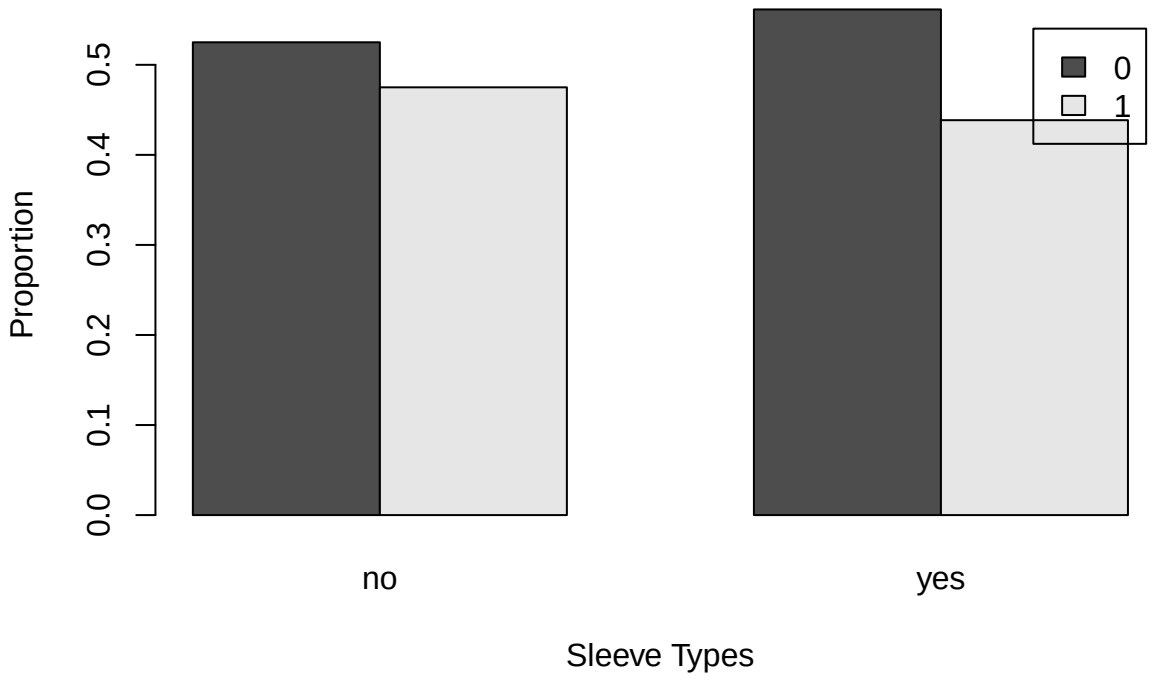
```
barplot(prop.table(table(dress_train$Recommendation, dress_train$Pattern),
margin = 2), beside = TRUE,xlab="Pattern Types",ylab="Proportion",
main = "Proportional barplot of Recommendation, by Pattern",legend=TRUE)
```




```
prop.table(table(dress_train$Recommendation, dress_train$Sleeve),margin = 2)
```

```
##
##            no       yes
##   0 0.5250000 0.5614973
##   1 0.4750000 0.4385027
```

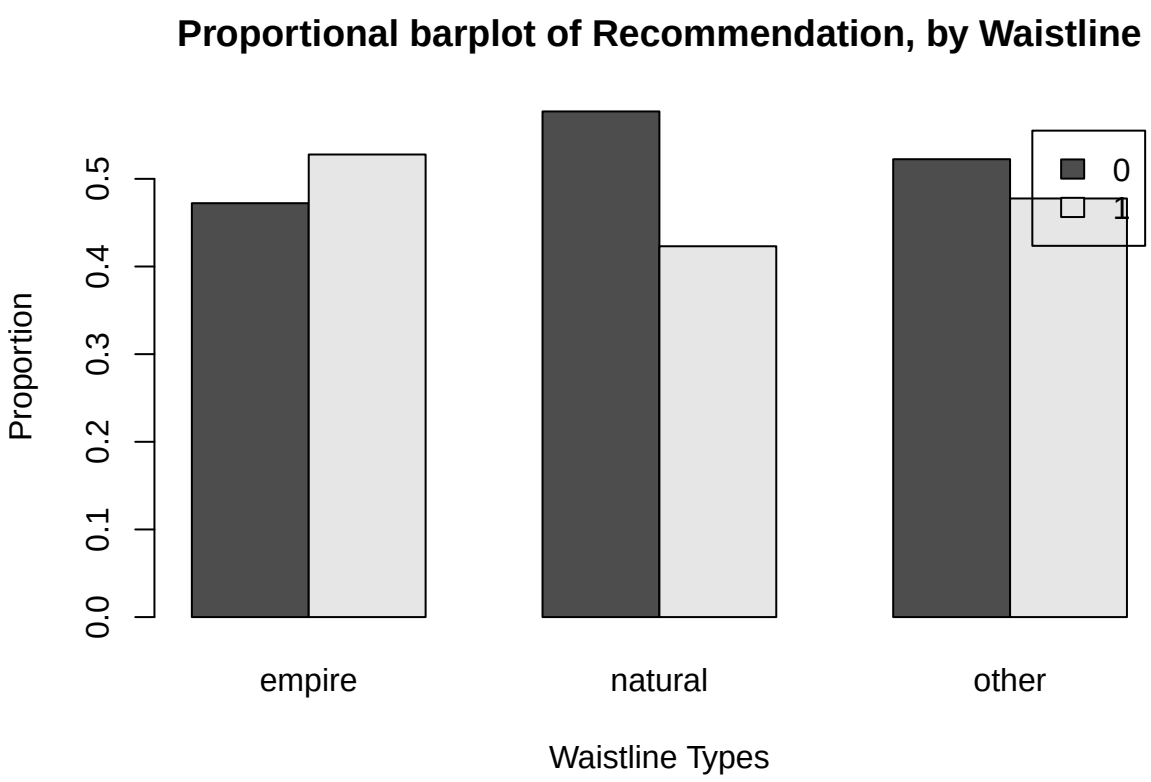
```
barplot(prop.table(table(dress_train$Recommendation, dress_train$Sleeve),
margin = 2), beside = TRUE,xlab="Sleeve Types",ylab="Proportion",
main = "Proportional barplot of Recommendation, by Sleeve",legend=TRUE)
```




```
prop.table(table(dress_train$Recommendation, dress_train$Waistline),margin = 2)
```

```
##
##        empire   natural     other
##   0 0.4722222 0.5769231 0.5223881
##   1 0.5277778 0.4230769 0.4776119
```

```
barplot(prop.table(table(dress_train$Recommendation, dress_train$Waistline),
margin = 2), beside = TRUE,xlab="Waistline Types",ylab="Proportion",
main = "Proportional barplot of Recommendation, by Waistline",legend=TRUE)
```



From the summaries above, it appears that

1. Recommendation is highest for party style and lowest for work style; on the other hand, for not getting recommendation, work style is highest and party style is the lowest.
2. Recommendation is similar level for both low and average prices and highest for high price clothes. On the other hand, no recommendation is highest for average prices and lowest for high prices.
3. Recommendation is highest for dresses appropriate for spring and lowest for those appropriate for fall; on the other hand, no recommendation is highest for dresses appropriate for fall and lowest for spring.
4. Both recommendation and no recommendation seem to remain roughly the same for clothes with different types of necklines respectively.. Generally, no recommendation level is higher than recommendation level. While no recommendation is highest for O-neck, recommendation is highest for V-neck.
5. Both recommendation and no recommendation seem to remain roughly the same for clothes, regardless of whether their materials are cotton or not.Same relationship seems to apply for when accounting for patterns and sleeves
6. Similar relationship seems to apply for whether the clothes have any decoration or not. No recommendation level is slightly higher when the dresses have no decoration.
7. No recommendation is the highest for dresses of natural waistline and lowest for those of empire waistline; recommendation level is the highest for dresses of empire waistline and lowest for those natural waistline

Modeling

We now turn to building and assessing our classifiers for predicting the type of wine. Our four classifiersare: linear discriminant analysis (lda), quadratic discriminant analysis (qda), classification trees, and binary logistic regression.To ensure that our models are not overfitting to our sample, we randomly split our observations into training and test sets. All four models were built using the same training observations and assessed on the same set oftest observations.

In order to design LDA and QDA, I decided to use the one quantitative variable I have. Also, for each categorical variable, the bar plots would vary at least slightly between different types, so I decided to use all of my categorical variables when I'm modelling a classification tree and binary logistic regression.

Linear Discriminant Analysis (LDA)

For our LDA and QDA models, we use only the continuous variable(rating). The LDA classifier is built on the training data as follows:

```
dress.lda<-lda(Recommendation~Rating,data=dress_train)
```

Then we investigate the performance of the LDA classifier on our test data as follows:

```
dress.lda.pred<-predict(dress.lda,as.data.frame((dress_test)))
```

```
table(dress.lda.pred$class,dress_test$Recommendation)
```

```
##
##       0   1
##   0 100  49
##   1   0   0
```

On the test data, LDA gave an overall error rate of (0+49)/149 = 0.329 which is quite high)

In particular,we do best at predicting dresses that will not sell well (error rate of only 0/149 = 0). Our LDA has a higher error rate for predicting dresses that will sell well (49/149 = 0.329).

Quadratic Discriminant Analysis (QDA)

Similarly, we use our quantitative variables for training a QDA classifier as follows:

```
dress.qda<-qda(Recommendation~Rating,data=dress_train)
```

And we investigate the performance of the QDA classifier on our test data as follows:

```
dress.qda.pred <-predict(dress.qda,as.data.frame(dress_test))
```

```
table(dress.qda.pred$class, dress_test$Recommendation)
```

```
##
##       0   1
##   0 100  49
##   1   0   0
```

We note that QDA is same as LDA at predicting dresses that will and won't sell.

QDA gave an overall error rate of (0+49)/149 = 0.329 which is quite high)

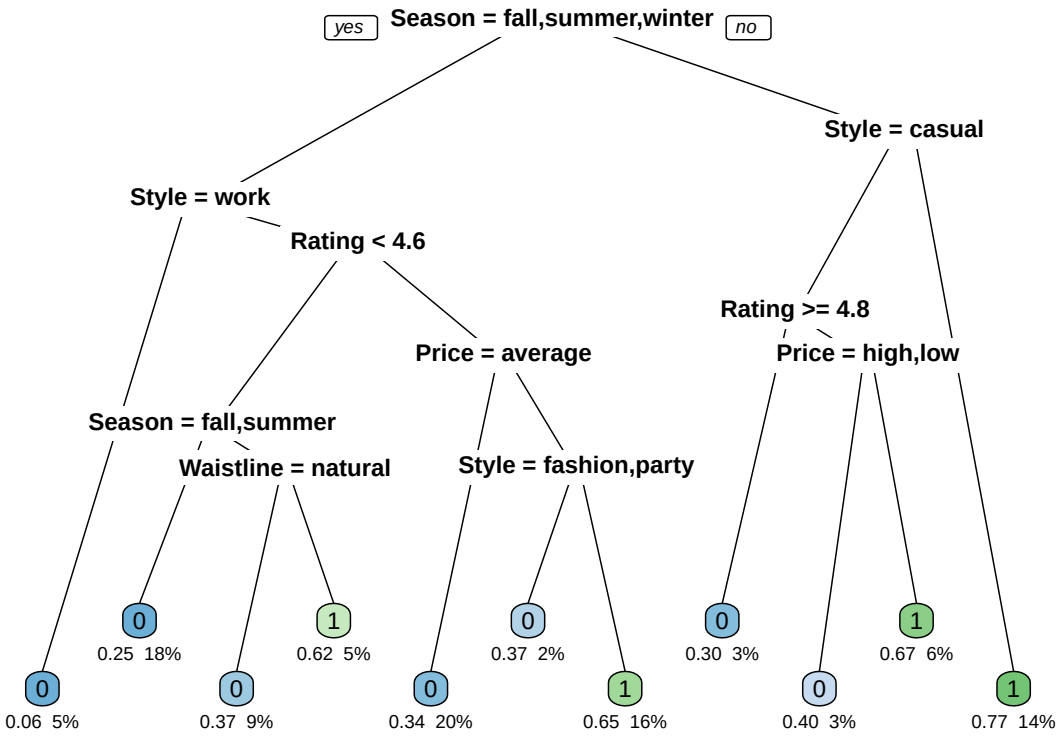
In particular,we do best at predicting dresses that will not sell well (error rate of only 0/149 = 0). Like our LDA, our QDA also has a higher error rate for predicting dresses that will sell well (49/149 = 0.329).

Classification Trees

While we could only take into account the quantitative variable in the LDA and QDA classifiers, we can also account for the categorical variables (Style,price, season,neckline,material,decoration,pattern,sleeve,waistline) in a classification tree. We fit a classification tree on the training data and plot it, as follows:

```
dress.tree <-rpart(Recommendation~Style+Price+Rating+Season+NeckLine+Material
                   +Decoration+Pattern+Sleeve+Waistline,
                   data=dress_train,method="class")
```

```
rpart.plot(dress.tree,type = 0,clip.right.labs = FALSE,branch = 0.1,under = TRUE)
```



We note that for the leaves, the numbers in the centers(0,1) represent the outcome for recommendation); the numbers at the bottom left represent the conditional probability, while the numbers at the bottom right represent the proportion of dress_train data.For example, if the dress is not made for fall, summer,winter and its style is casual, the conditional probability of the dress being recommended is 0.77, and 14% of dress_train data goes under this category.

We note that the classification tree selected did not use NeckLine, Material, Decoration, Pattern and Sleeve to predict whether the dress will sell well. [In general,the "most important" variables that the tree determines for classification will be indicated from top down onthe tree.] Also, less variables are used to determine if the clothes will sell when the clothes are meant for spring.

We then investigate the performance of the tree classifier on our test data as follows:

```
dress.tree.pred <-predict(dress.tree,as.data.frame(dress_test),type="class")
```

```
table(dress.tree.pred, dress_test$Recommendation)
```

```
##
```

```
## dress.tree.pred  0  1
##                0 71 31
##                1 29 18
```

Our tree gave an overall error rate of (29+31)/149 = 0.403 (which is quite high)

In particular,we do best at predicting dresses that will not sell well (error rate of only 29/149 =0.195). Our tree has a slightly higher error rate for predicting dresses that will sell well (31/149 = 0.208).

Binary Logistic Regression

Finally, we consider binary logistic regression to model the type of wine. Similarly to the classification trees,a logistic classifier can use all the variables including the categorical variables(Style, price, season,neckline,material,decoration,pattern,sleeve,waistline).

We train a logistic classifier on the training data, and then inspect the resulting confusion matrix from thetest data, as follows: We first fit a binary logistic regression to the data as follows:

```
dress.logit<-glm(factor(Recommendation)~Rating+factor(Style)+factor(Price)
                 +factor(Season)+factor(NeckLine)+factor(Material)
                 +factor(Decoration)+factor(Pattern)+factor(Sleeve)
                 +factor(Waistline),data=dress_train,
                 family=binomial(link="logit"))
```

We then apply the logistic model to the test data:

```
dress.logit.prob <-predict(dress.logit,as.data.frame(dress_test),type = "response")
```

Since the logistic model applied to the test data yields probabilities (not just 0 or 1), we will convert the logistic probabilities into classification predictions by thresholding the probability, so that if prob>0.5, we will classify it as recommended (else, classify as the otherwise. Basically 0).

In order to associate the correct direction of probability with the appropriate recommendation level , we need to see how "Recommendation" is default ordered. We do that by running "levels" on the factored response variable, as follows:

```
dress.logit.pred <-ifelse(dress.logit.prob>0.5,"1","0")
```

We then evaluate how the the logistic classifier performed on our test data with a confusion matrix as shown:

```
table(dress.logit.pred, dress_test$Recommendation)
```

```
##
## dress.logit.pred  0  1
##                0 77 27
##                1 23 22
```

The logistic model as a classifier (using threshold probability of 0.5) performs better than the classification tree and nearly as well as the LDA and QDA, with overall error rate of ((23+27)/149)=0.336. For predicting dresses that will sell well, it gives an error rate of (27/149)=0.181. For predicting dresses that won't sell well, it gives an error rate of (23/149)=0.154.

We note that as for LDA and QDA, the logistic regression classifier and even the tree performed better on predicting dresses that will sell.

Final Recommendation

Of the four classifiers we tested, the LDA and QDA performed the best (and equally) on the test data). Our final recommendation is either LDA or QDA.

However, as the available data from last year only had rating as the quantiative variable, these models only used one quantitative variable. If the store feels uncomfortable relying on a model with such limitation, then the logistic classifier seems to be a good secondary recommendation and potentially more stable.Logistic regression performed nearly as well as them overall.

We note that both logicstic and the tree showed better pformance with respect to predicting clothes that will sell.

Discussion

Overall, our models did reasonable at predicting clothes that would sell well. We note that LDA and QDA had exactly same overall error rates; they also had identical error rates for predicting clothes that will sell well and not well respectively.This is most likely due to both models only having one same quantitative variable to use: rating.

It would probably be helpful to see if there is anymore available data than just last year's data. In addition, having more variables—especially quantiative variables—could help improve and stabilize the classification routines.

Other areas for future research that could be of greater interest to the industry would be to build models that can predict what combinations of quantitative and categorical variables would be most optimal for making clothes get sold more. Through such models, the industry could cut down on the resource wastage, financial losses and opportunity cost incurred by producing clothes that will not sell well.