Predicting Dress Sales

Oskar Eisgruber

April 22, 2020

```
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 !

The dress insdustry is cut throat. There are a lot of people who are looking to buy dresses, and a lot of money can be made in the industry. However, not all dresses sell. What amkes dresses more attractable to buyers? How can one capitilize on this lucrative market and cash out? The answer is you need to know what the right dresses to sell are. Those dresses are rthe ones people will buy. If one cannot sell a dress they lose all of hte money they paid for it.

In this paper, we will train and evaluate machine learning classification techniques for classifying dresses based on various dress characteristics and use the techniques to predict whether a dress will sell or not. This could be a valuable tool to the dress insdustry if one of the classifiers works well.

Exploratory Data Analysis

Background ! and Variables

A clothing store dress sales have not been very successful. Although some styles sold out very fast, others were still there after sales events. The store wants to know which dresses they should order for next year so they have more successful sales.
Our sample of dresses include the stores sales from last year. The dress characteristics that will be used as predictor varibales are below.

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 347 observations, with 189 dresses that sell well comprising 54.47 percent of the dresses, and 158 dresses that do not sell well comprising 45.53 percent of the dresses, as shown in the following tables:

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

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

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

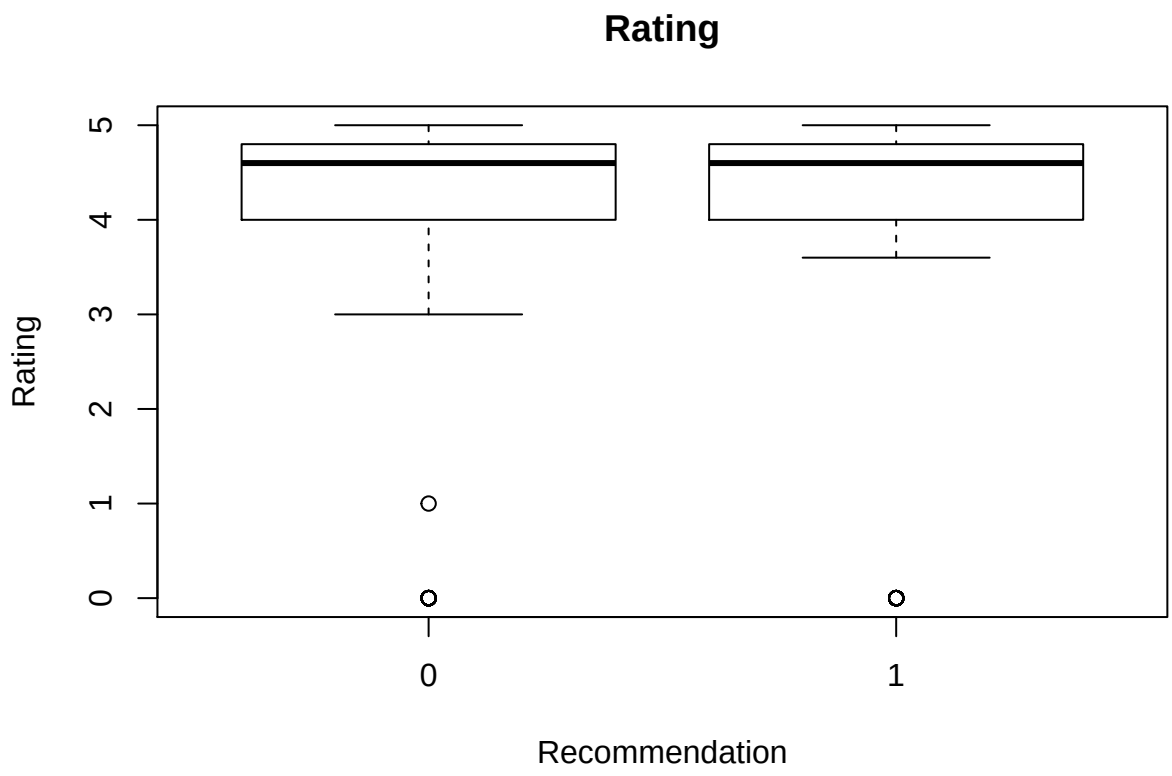
```
##
##         0         1
## 0.5446686 0.4553314
```

Some EDA on relationships between Reccomendation and the quantitative variables

We then move toward visualizing the relationship between the response (Recommendation) and the various predictors (dress characteristics).

For visually exploring whether we expect the quantitative predictors to be useful in helping to classify the dress Recommendation, we show boxplots, which appear as follows:

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



In the above boxplot, we note that if the boxplot shows a difference between the two dress Recommendations, 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 the boxplot of Rating above does not show a difference between the two dress Recommendations, so there is no evidence of a relationship.

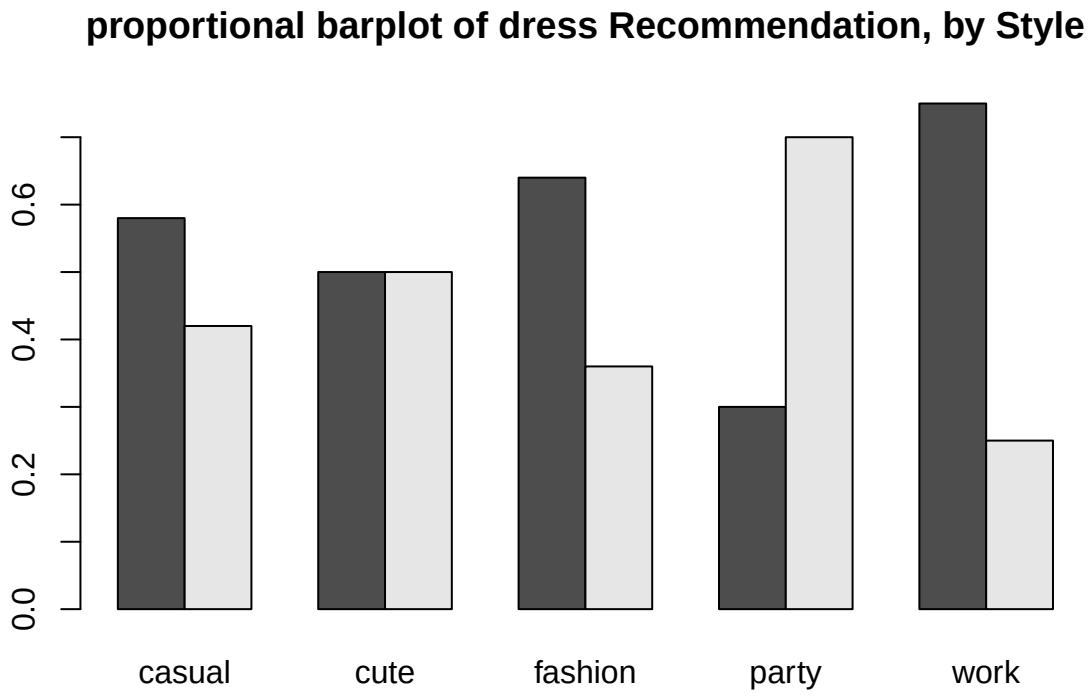
EDA on relationships between Recommendation and the categorical variables

To explore the relationship between Reccomndation and the categorical predictor variables, we can look at the conditional proportions of Reccomendation, conditioned on the categorical predictor variables, shown as follows:

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

```
##
##     casual cute fashion party work
##   0   0.58 0.50    0.64  0.30 0.75
##   1   0.42 0.50    0.36  0.70 0.25
```

```
barplot(
  round(prop.table(
  table(dress_train$Recommendation, dress_train$Style),
           margin = 2),2)
  , beside = TRUE,
  main = "proportional barplot of dress Recommendation, by Style")
```



```
legend = c("dark = not recommended","light")
```

```
par(mfrow = c(3,3),
    mai= c(0.3,0.3,0.1,0.1))

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

```
##
##     casual cute fashion party work
##   0   0.58 0.50    0.64  0.30 0.75
##   1   0.42 0.50    0.36  0.70 0.25
```

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

```
##
##     average high  low
##   0    0.57 0.38 0.55
##   1    0.43 0.62 0.45
```

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

```
##
##     fall spring summer winter
##   0 0.68   0.35   0.62   0.57
##   1 0.32   0.65   0.38   0.43
```

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

```
##
##     oneck other vneck
##   0  0.55  0.54  0.52
##   1  0.45  0.46  0.48
```

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

```
##
##     cotton other
##   0   0.55  0.54
##   1   0.45  0.46
```

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

```
##
##       no  yes
##   0 0.56 0.53
##   1 0.44 0.47
```

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

```
##
##       no  yes
##   0 0.52 0.58
##   1 0.48 0.42
```

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

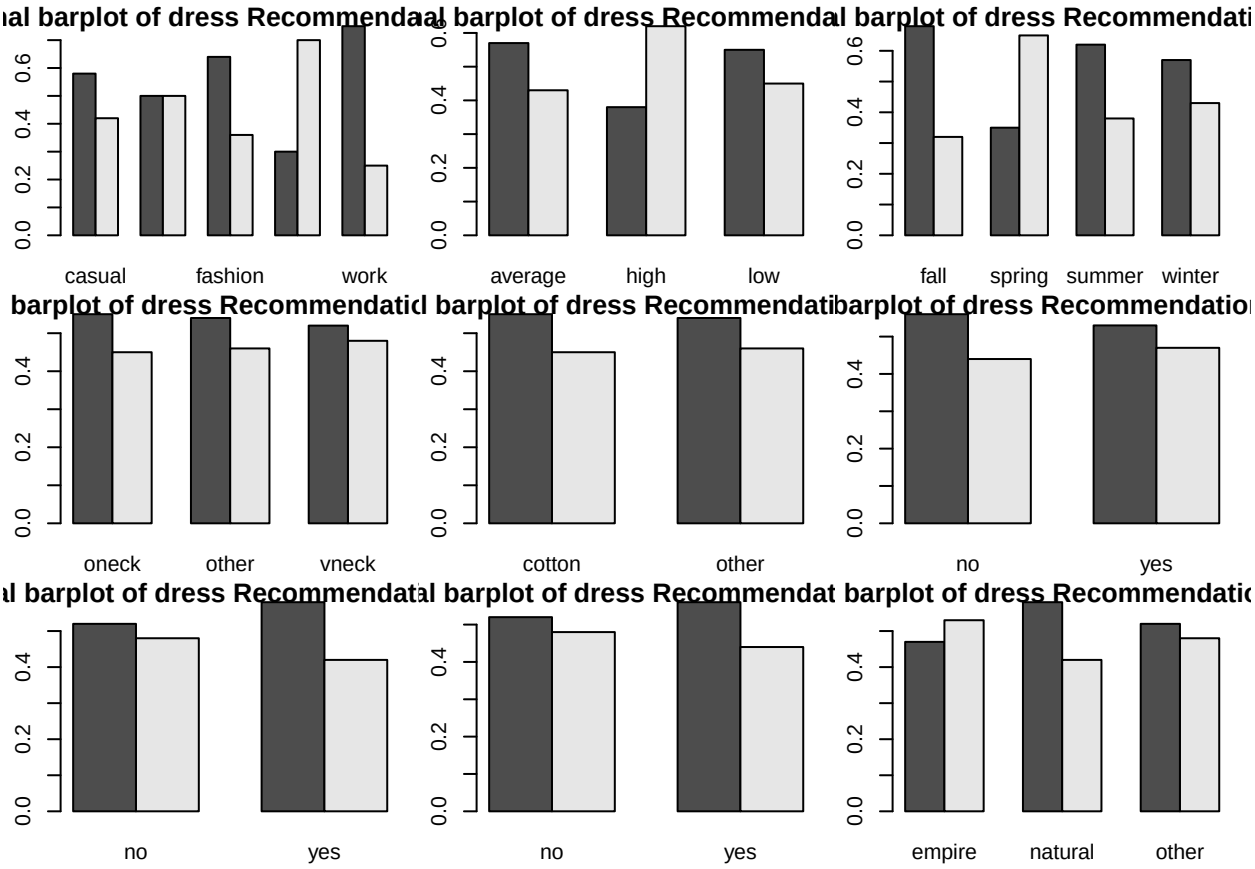
```
##
##       no  yes
##   0 0.52 0.56
##   1 0.48 0.44
```

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

```
##
##     empire natural other
##   0   0.47    0.58  0.52
##   1   0.53    0.42  0.48
```

```
par(mfrow = c(3,3),
    mai= c(0.3,0.3,0.1,0.1))

barplot(
  round(prop.table(
  table(dress_train$Recommendation, dress_train$Style),
           margin = 2),2)
  , beside = TRUE,
  main = "proportional barplot of dress Recommendation, by Style")
legend = c("dark = not recommended","light")

barplot(
  round(prop.table(
  table(dress_train$Recommendation, dress_train$Price),
           margin = 2),2)
  , beside = TRUE,
  main = "proportional barplot of dress Recommendation, by Price")
legend = c("dark = not recommended","light")

barplot(
  round(prop.table(
  table(dress_train$Recommendation, dress_train$Season),
           margin = 2),2)
  , beside = TRUE,
  main = "proportional barplot of dress Recommendation, by Season")
legend = c("dark = not recommended","light")

barplot(
```

```
  round(prop.table(
  table(dress_train$Recommendation, dress_train$NeckLine),
           margin = 2),2)
  , beside = TRUE,
  main = "proportional barplot of dress Recommendation, by NeckLine")
legend = c("dark = not recommended","light")

barplot(
  round(prop.table(
  table(dress_train$Recommendation, dress_train$Material),
           margin = 2),2)
  , beside = TRUE,
  main = "proportional barplot of dress Recommendation, by Material")
legend = c("dark = not recommended","light")

barplot(
  round(prop.table(
  table(dress_train$Recommendation, dress_train$Decoration),
           margin = 2),2)
  , beside = TRUE,
  main = "proportional barplot of dress Recommendation, by Decoration")
legend = c("dark = not recommended","light")

barplot(
  round(prop.table(
  table(dress_train$Recommendation, dress_train$Pattern),
           margin = 2),2)
  , beside = TRUE,
  main = "proportional barplot of dress Recommendation, by Pattern")
legend = c("dark = not recommended","light")

barplot(
  round(prop.table(
  table(dress_train$Recommendation, dress_train$Sleeve),
           margin = 2),2)
  , beside = TRUE,
  main = "proportional barplot of dress Recommendation, by Sleeve")
legend = c("dark = not recommended","light")

barplot(
  round(prop.table(
  table(dress_train$Recommendation, dress_train$Waistline),
           margin = 2),2)
  , beside = TRUE,
  main = "proportional barplot of dress Recommendation, by Waistline")
```




```
legend = c("dark = not recommended","light")
```

From the summaries above, a greater percentage of Casual, Fasion, and Work dresses do not sell well. Whereas, a greater percentage of Party dresses do sell well. An equal percentage of cute dresses sell well and dont sell well.

From the summaries above, a greater percentage of dresses in the average and low price range do not sell well. Whereas, a greater percentage of dresses in the high preice range do sell well.

From the summaries above, a greater percentage of dresses appropriate for the fall, summer, and winter seasons do not sell well. Whereas, a greater percentage of dresses appropriate for the spring season do sell well.

From the summaries above, a greater proportion of dresses with a vneck neckline sell well than dresses with an oneck or other neckline.

From the summaries above, a slightly smaller percentage of cotton dresses sell well than other dresses.

From the summaries above, a slightly greater percentage of dresses that have decoration sell well than dresses that do not have decoration.

From the summaries above, a slightly greater percentage of dresses that do not have a pattern sell well than dresses that do have a pattern.

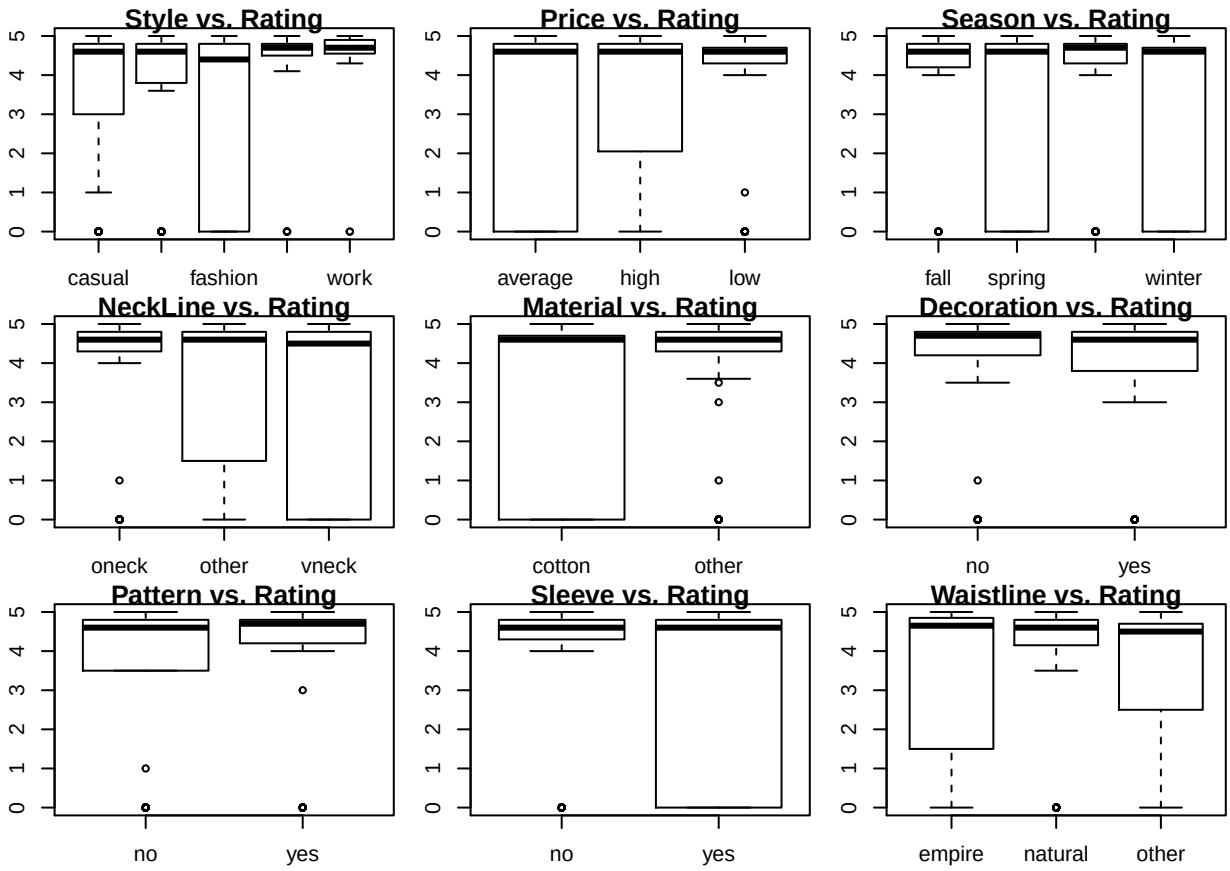
From the summaries above, a slightly greater percentage of dresses that do not have a sleeve sell well than dresses that have a sleeve

From the summaries above, a greater percentage of dresses with an empire waistline sell well than a natural or other waistline.

Some visual EDA on classification pairs

Finally, to get a sense of which pairs of quantitative predictors might help classify type, we can inspect labeled bivariate plots. We do that in a pairs plot:

```
par(mfrow = c(3,3),
    mai= c(0.3,0.3,0.1,0.1))


boxplot(Rating ~ Style,
        main="Style vs. Rating",
        data = dress_train)

boxplot(Rating ~ Price,
        main="Price vs. Rating",
        data = dress_train)

 boxplot(Rating ~ Season,
        main="Season vs. Rating",
        data = dress_train)

 boxplot(Rating ~ NeckLine,
        main="NeckLine vs. Rating",
        data = dress_train)

 boxplot(Rating ~ Material,
        main="Material vs. Rating",
        data = dress_train)

 boxplot(Rating ~ Decoration,
        main="Decoration vs. Rating",
        data = dress_train)

 boxplot(Rating ~ Pattern,
        main="Pattern vs. Rating",
        data = dress_train)

 boxplot(Rating ~ Sleeve,
        main="Sleeve vs. Rating",
        data = dress_train)

 boxplot(Rating ~ Waistline,
        main="Waistline vs. Rating",
        data = dress_train)
```



In the above boxplots, we note that if the boxplot shows a difference between the categorical variable, we have some evidence of a relationship between the two predictor variables that are plotted in the boxplot (although note that this is not the same as a statistically significant relationship).

In our boxplots above we see the only quantitative variable, Rating, plotted against all of the other variables which are categorical. We see that there dont seem to be any clear suggestions of a relationship between any of the predictor variables based off of these box plots. The plots that seem to suggest the smallest relationship are Decoration vs. Rating and Pattern vs. Rating. All of the other plots seem to suggest more of a relationship than the two prevoisuly mentioned, but nothing convincing, at least from a general visual interpretiation.

We do note, however, that we have only looked at single or pairs of variables, and that the true relationship in higher-dimensional space is likely more complicated.

Modeling

We now turn to building and assessing our classifiers for predicting the Recomendation of a dress. Our four classifiers are: 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 of test observations.

Linear Discriminant Analysis (LDA)

For our LDA and QDA models, we use only the continuous variables (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 $49/149 = 32.89\%$ which is quite high. In particular, we do best at finding the dresses that will not sell (error rate of only $0/100 = 0\%$). Our LDA has a higher error rate for classifying red wines ($49/49 = 100\%$).

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
```

With QDA, we might expect slightly better performance than LDA, given that QDA is more flexible at finding nonlinear, curved decision boundaries.

In this situation this is not the case, our results tabuled above no change in the overall error rate from LDA: $49/149 = 0.3289$. We do continue to do better at properly classifying dress that do not sell well (error rate of $0/100 = 0\%$) than with dresses that sell well (error rate of $49/49 = 100\%$).

We note that the LDA model is just as good as the QDA model at idenitfying the dresses that do and do not sell well. It's possible that QDA is overfitting for the red wines.

We note that the QDA and LDA models are both modeled using only the quantitative predictor variables from the dataset. Accordingly only 1 of the 10 predictor variables was used from the dataset, Rating, and the EDA suggested that it did not even show a strong relationship to the response variable, Recommendation.

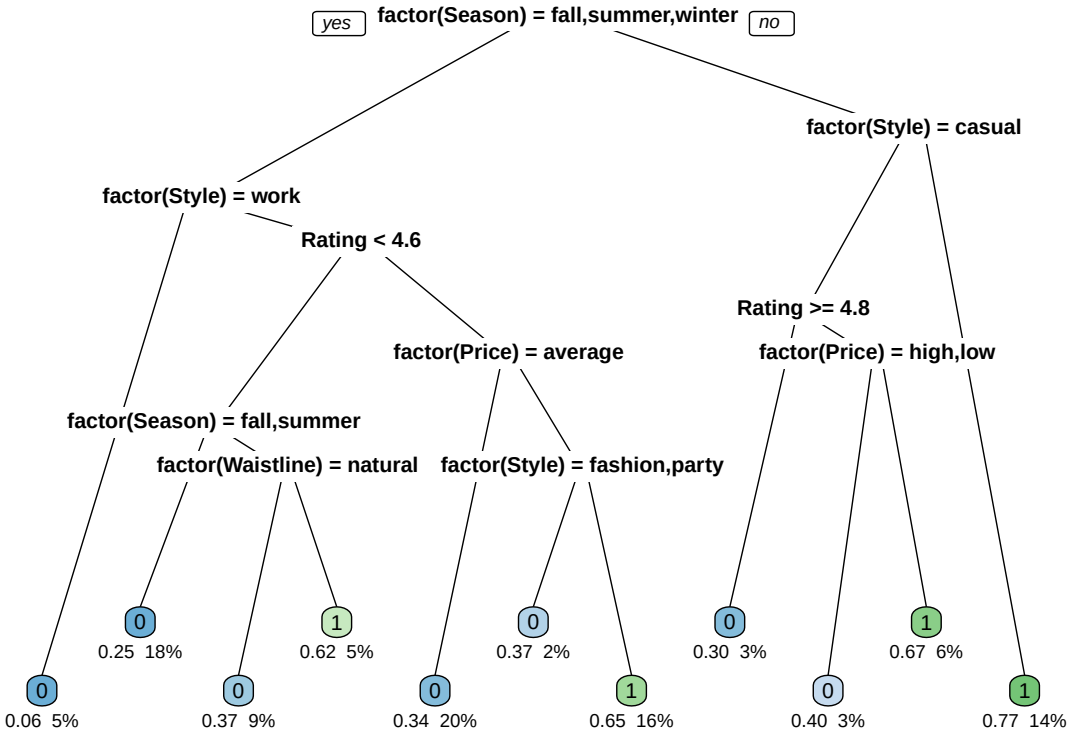
Classification Trees

While we could only take into account the quantitative variables in the LDA and QDA classifiers, we can also account for the categorical variable in a classification tree.

We fit a classification tree on the training data and plot it, as follows:

```
dress.tree <- rpart(factor(Recommendation) ~ factor(Style) + factor(Price) + Rating + factor(Season) + 
                      data=dress_train,
                      method="class")
```

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



We note that the classification tree selected Season as the most important variable to use to classify the wine type. [In general, the "most important" variables that the tree determines for classification will be indicated from top down on the tree.]

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
```

On the test data, Classification Trees gave an overall error rate of (29+31=60)/149 = 40.26% which is quite high, in fact it it is higher than the error rates given by the LDA and QDA models. In particular we do best at finding the dresses that will not sell well (error rate of 29/100 = 29%), and we do the worst at finding the dresses that will sell well (error rate of 31/49 = 63.27%).

Binary Logistic Regression

Finally, we consider binary logistic regression to model the dress Recommendation. Similarly to the classification trees, a logistic classifier can use all the variables including the categorical variables.

We train a logistic classifier on the training data, and then inspect the resulting confusion matrix from the test data, as follows:

We first fit a binary logistic regression to the data as follows:

```
dress.logit <- glm(factor(Recommendation) ~
                     factor(Style) + factor(Price) + Rating + factor(Season) + factor(NeckLine) + facto
              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 red/white classification), we will convert the logistic probabilities into classification predictions by thresholding the probability, so that if prob>0.5 we will classify it as the dress will sell well (else, classify as the other Recommendation).

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

```
levels(factor(dress_test$Recommendation))
```

```
## [1] "0" "1"
```

We then obtain test classification from the logistic model using a threshold probability of 0.5, 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 bettwe than classification tree, with overall error rate of only 0.3356 ((27+23)/149). For dresses that do not sell well, it gives an error rate of only 0.2300 (23/100), and for dresses that do sell well, it gives an error rate of only 0.5510 (27/49).

We note that as for LDA and QDA, the logistic regression classifier performed better on dresses that sell well.

Final Recommendation

Of the four classifiers we tested, the LDA and QDA models performed the best in terms of overall error rate.

LDA and QDA performed the same, and since the QDA is more suseptible to overfitting, and the LDA is simpler than the QDA, it is reasonable to trust the LDA prediction over the prediciton of the QDA.

However, we note that the QDA and LDA models are both modeled using only the quantitative predictor variables from the dataset. Accordingly only 1 of the 10 predictor variables was used from the dataset, Rating, and the EDA suggested that it did not even show a strong relationship to the response variable, Recommendation.

The overall error rate of the Logistic regression classifier was close to that of the LDA and QDA classifiers. Additionaly, it used all of the predictor variables to make its prediction.

As a result it might be safest to trust the logistic regression classifiers predictions.

We note that all of the classifiers showed better performance with respect to dress that do not sell than dresses that sell.

Our final recommendation is the logistic regression classifier but with the caveat that it did not have the lowest overall error rate. The LDA and QDA models produced the lowest overall error rate, but since they only used one of the ten predictor variables, and EDA showed that the varibale that was used did not show a convincing relationship to the response varibale, we reccomend the logistic regression classifier which used all ten of the predictor varibales in its prediction, and had an overal error rate close to that of the LDA and QDA models.

Discussion

Overall, our models did not do that well at classifying dress Recommendation. We note the realistic nature of the data, which most likely accounts for the high error rates.

Furthermore we note thatthe QDA and LDA models are both modeled using only the quantitative predictor variables from the dataset. Accordingly only 1 of the 10 predictor variables was used from the dataset, Rating, and the EDA suggested that it did not even show a strong relationship to the response variable, Recommendation.

In order to further pursue how to accurately predict if a dress will or will not sell well, one could find more data to include in the analsis. It would especially be useful to include more quantitative variables in order to rely more heavily on the prediction from the LDA and QDA models. Some futher variables to pursue could be location in the store where the dresses are positioned, color of the dress, length of the dress, time of year dress is sold.

Other areas for future research that could be of greater interest to the industry would be to test whether there are underlying or lurking varibales that are causing some dresses to sell more than others that are not reflected in characteristics of the dress that were considered in the data set analyzed in this report. In order to do so, other variables such as demographics of the customers, employees working on shift when dresses are sold could be considered.

Furthermore, retail sales are going down as technology improves and more and more stores rely on online sales. Maybe data could be collected to analyze the impact the transition of retail from the store to online can be analyzed and it might reflect a reason why some dresses are sold over others.