**Car Evaluation Machine Learning Analysis**

1. **Introduction**

This report was prepared using RStudio. The data used is *Car Evaluation Data Set* and was obtained from the UC Irvine Machine Learning Repository (archive.ics.uci.edu/ml/). It’s a multivariate set of data with 1,728 records/instances in total, each with 7 attributes. The data was provided to UC Irvine on 1 June 1997 by Marko Bohanec and Blaz Zupan.

|  |  |  |
| --- | --- | --- |
| Variable Name | Type | Description |
| buying | Categorical | Original buying price of the car |
| maint | Categorical | Price of maintenance on the vehicle |
| doors | Categorical | Number of doors |
| persons | Categorical | Capacity of the vehicle in terms of persons to carry |
| lug\_boot | Categorical | Size of the vehicle’s luggage boot |
| Safety | Categorical | Estimated level of safety of the vehicle |
| accept† | Categorical | Acceptability rating of the vehicle by its owner |
|  |  | † prediction classes/response variable |

1. **Diagnostics**

The data set and necessary R packages for our analysis are loaded into RStudio.

library(rmarkdown)

library(readr)

library(ggplot2)

library(gridExtra)

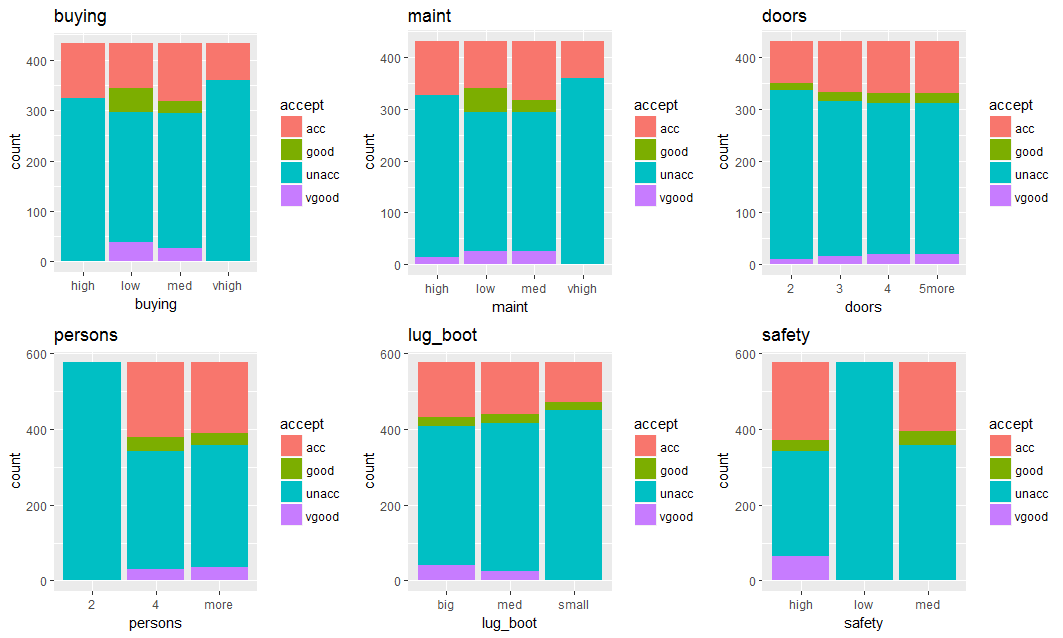
library(rattle)

car\_eval <- read\_csv("~/Projects/Car Evaluation/Data/car\_eval.csv")

View(car\_eval)  
str(car\_eval, vec.len=1)

## Classes 'tbl\_df', 'tbl' and 'data.frame': 1728 obs. of 7 variables:  
## $ buying : chr "vhigh" ...  
## $ maint : chr "vhigh" ...  
## $ doors : chr "2" ...  
## $ persons : chr "2" ...  
## $ lug\_boot: chr "small" ...  
## $ safety : chr "low" ...  
## $ accept : chr "unacc" ...

Next, we want to make some diagnostic plots to see if we can identify any important features that might predict a vehicle’s acceptability rating. We will use the information gathered to select which class of machine learning model to apply to the data and which features to include in our model.

plot1 <- ggplot(car\_eval, aes(buying)) + geom\_bar(aes(fill=accept))  
plot2 <- ggplot(car\_eval, aes(maint)) + geom\_bar(aes(fill=accept))  
plot3 <- ggplot(car\_eval, aes(doors)) + geom\_bar(aes(fill=accept))  
plot4 <- ggplot(car\_eval, aes(persons)) + geom\_bar(aes(fill=accept))  
plot5 <- ggplot(car\_eval, aes(lug\_boot)) + geom\_bar(aes(fill=accept))  
plot6 <- ggplot(car\_eval, aes(safety)) + geom\_bar(aes(fill=accept))  
grid.arrange(plot1, plot2, plot3, plot4, plot5, plot6, nrow=2)

Based on the diagnostic data, one interpretation that could be gathered is that **safety** is an important feature in rating a car’s acceptability. Seating capacity (**persons**) appears to be a relevant feature, as well. We will use the features **safety** and **persons** as independent variables to explain car acceptability. Because our goal is to predict a categorical outcome with four possible choices and our independent variables are also categorical, I chose to use a multiclass decision tree to model the data.

Decision trees are advantageous for a few reasons. First, they require relatively little effort in data preparation compared to most other machine learning algorithms. Nonlinear relationships don’t affect the performance of decision trees and finally, they are easier than most machine learning algorithms to explain to someone who is not a subject matter expert (i.e. a department VP, coworkers, etc.)

1. **The Decision Tree Algorithm**

To build the model, we must first know the counts and proportions of the categories for **accept**. We also want to assess the importance of the two explanatory features.

##  
## unacc acc good vgood  
## 1210 384 69 65  
##   
## unacc acc good vgood

## 0.70023148 0.22222222 0.03993056 0.03761574

varImp(car\_eval, scale=FALSE)

## persons safety

## 85.35115 69.87636

rpart(accept ~ safety+persons, data=car\_eval, method="class", minsplit=2)  
## Confusion Matrix and Statistics

##

## Reference

## Prediction unacc acc good vgood

## unacc 516 82 0 0

## acc 47 142 0 0

## good 9 29 0 0

## vgood 0 39 0 0

##

## Overall Statistics

##

## Accuracy : 0.7616

## 95% CI : (0.7317, 0.7896)

## No Information Rate : 0.662

## P-Value [Acc > NIR] : 1.242e-10

##

## Kappa : 0.4904

## Mcnemar's Test P-Value : NA

##

## Statistics by Class:

##

## Class: unacc Class: acc Class: good Class: vgood

## Sensitivity 0.9021 0.4863 NA NA

## Specificity 0.7192 0.9178 0.95602 0.95486

## Pos Pred Value 0.8629 0.7513 NA NA

## Neg Pred Value 0.7895 0.7778 NA NA

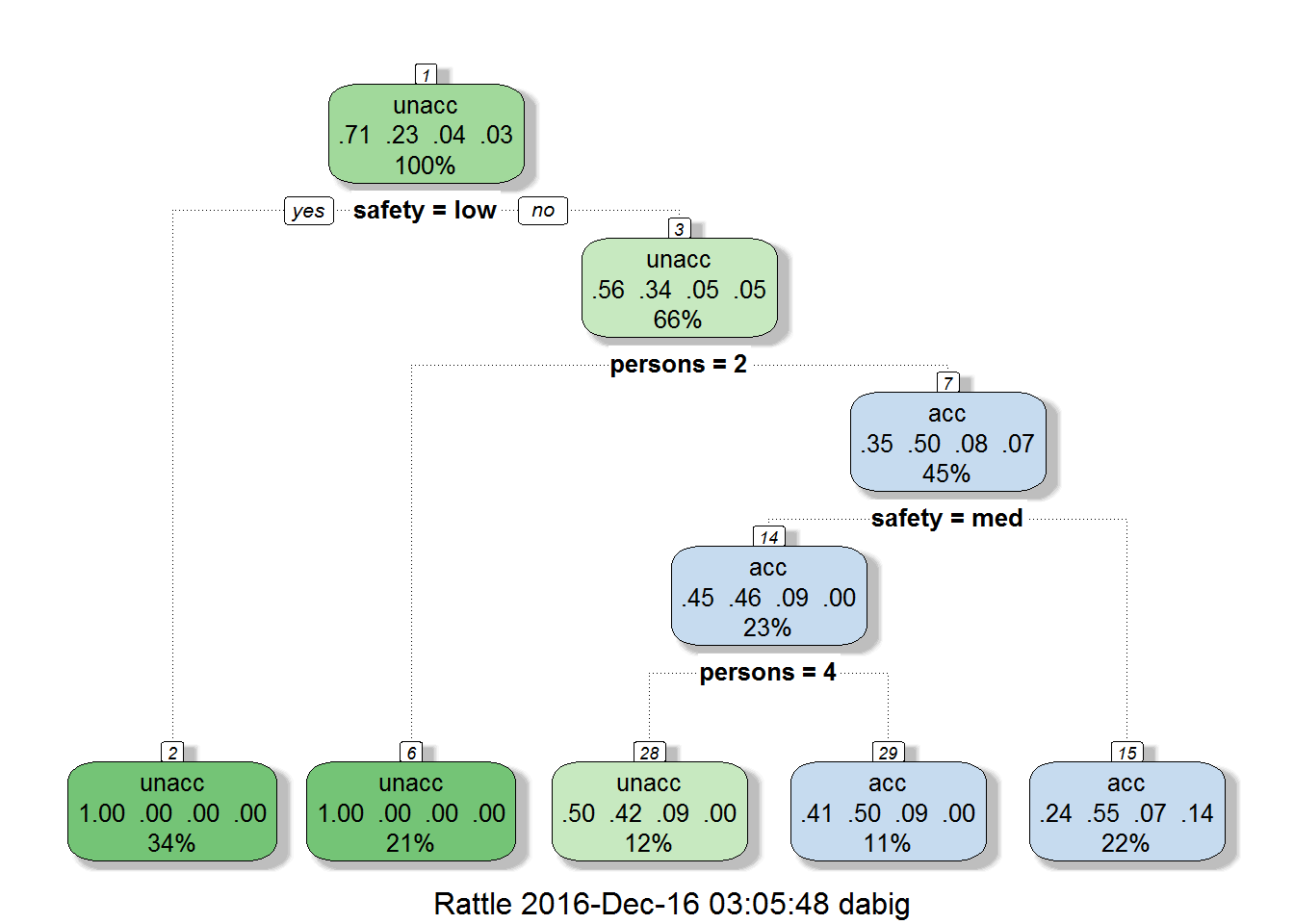
## Prevalence 0.6620 0.3380 0.00000 0.00000

## Detection Rate 0.5972 0.1644 0.00000 0.00000

## Detection Prevalence 0.6921 0.2188 0.04398 0.04514

## Balanced Accuracy 0.8106 0.7021 NA NA

1. **The Improved Algorithm**

As you can see, the initial model has about a 76% accuracy. Let’s see if that can be improved on by running another decision tree, this time using *all* the available explanatory features.

rpart(accept ~ safety + persons + buying + maint + doors + lug\_boot, data=car\_eval, method="class", minsplit=2)

## Confusion Matrix and Statistics

##

## Reference

## Prediction unacc acc good vgood

## unacc 596 3 0 0

## acc 3 184 7 2

## good 0 0 34 6

## vgood 0 0 0 29

##

## Overall Statistics

##

## Accuracy : 0.9757

## 95% CI : (0.9631, 0.9849)

## No Information Rate : 0.6933

## P-Value [Acc > NIR] : < 2.2e-16

##

## Kappa : 0.9479

## Mcnemar's Test P-Value : NA

##

## Statistics by Class:

##

## Class: unacc Class: acc Class: good Class: vgood

## Sensitivity 0.9950 0.9840 0.82927 0.78378

## Specificity 0.9887 0.9823 0.99271 1.00000

## Pos Pred Value 0.9950 0.9388 0.85000 1.00000

## Neg Pred Value 0.9887 0.9955 0.99150 0.99042

## Prevalence 0.6933 0.2164 0.04745 0.04282

## Detection Rate 0.6898 0.2130 0.03935 0.03356

## Detection Prevalence 0.6933 0.2269 0.04630 0.03356

## Balanced Accuracy 0.9918 0.9831 0.91099 0.89189

So, it appears that by using all the explanatory features in our data set, we improved the model’s accuracy by **21%**. We can now predict a vehicle’s acceptability rating 97% of the time if we know the other variables. Since our new model’s accuracy is so close to 100%, it’s unnecessary to continue to test alternative models.

1. **Summary**

|  |  |  |
| --- | --- | --- |
| Algorithm | Variables | Accuracy |
| decisionTree | safety + persons | 0.7616 |
| decisionTree | all | 0.9757 |