DataVizA Tutorial: Decision Trees: Solutions

Department of Econometrics and Business Statistics, Monash University

Tutorial 11

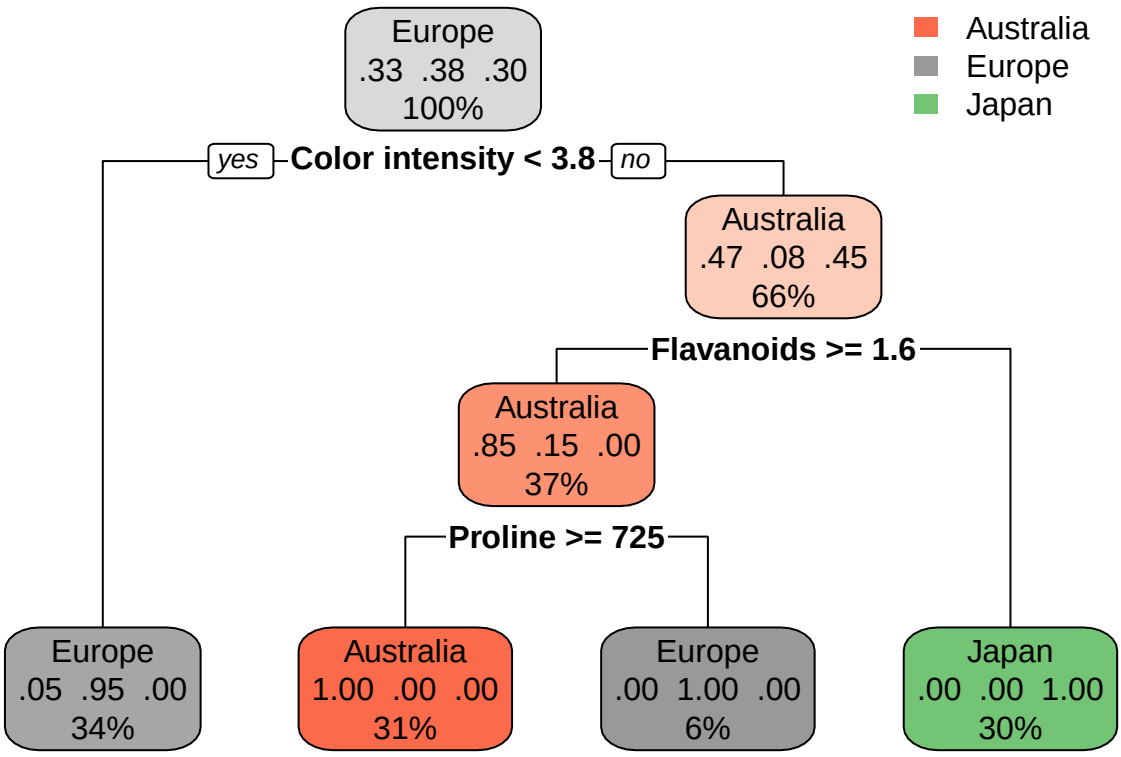
Wine Data

1. Construct a decision tree using all data in *ExistingWines.rds* in the training set and predict the data in *NewWines.rds*.

```
#Use MASS package
library(rpart)
#Later tidyverse also used
library(tidyverse)
#Load in data
ExistingWines<-readRDS('ExistingWines.rds')
#Load in New Wine Data
NewWines<-readRDS('NewWines.rds')
tree<-rpart(BestMarket~.,data = ExistingWines)
yhat<-predict(tree,newdata = NewWines,type='class')
```

2. Create a visual representation of the tree selected in Question 1. Use the default settings of `rpart.plot`

```
library(rpart.plot)
rpart.plot(tree)
```



3. A wine has a Color Intensity of 3.9, Flavanoids of 1.8 and Proline of 800. What is your prediction for the market this wine is suited to? Use the output from the previous question to answer this rather than R.

Move right at the root node, left at the next node and left at the following node. The prediction is Australia.

4. What are the predicted probabilities that the wine in question 3 belongs to each class?

Since all wines in this partition are classified as Australia the predicted probability is 1 for Australia and 0 for Europe and Japan.

5. A wine has a Color Intensity of 2.4, Flavanoids of 1.2 and Proline of 700. What is your prediction for the market this wine is suited to? Use the output from the previous question to answer this rather than R.

Move left at the root node. A leaf is reached. The prediction is Europe.

6. What are the predicted probabilities that the wine in question 3 belongs to each class?

This time the predicted probablities are 0.95 for Europe, 0.05 for Australia and 0 for Japan.

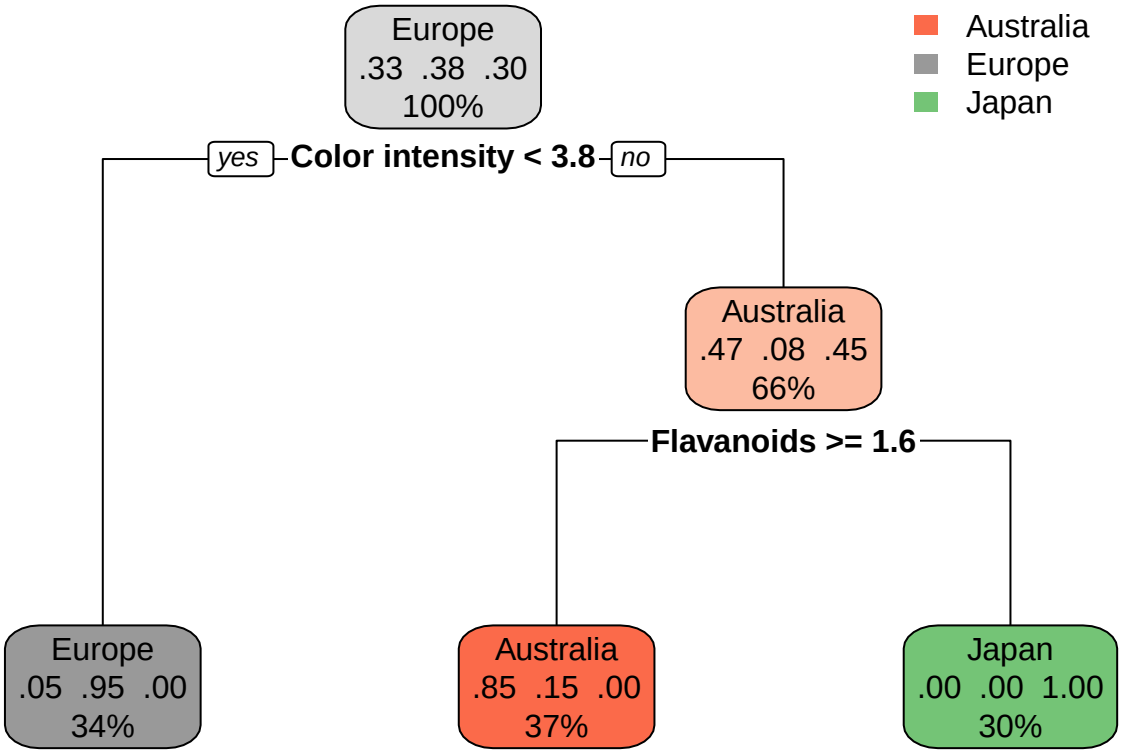
7. Using R, find predictions for the first ten wines in the *NewWines.rds*

```
#The first ten predictions can be checked using the head function
head(yhat,10)
```

```
##         1         2         3         4         5         6         7
## Australia Australia Australia Australia Australia Australia Australia
##         8         9        10
##    Europe Australia Australia
## Levels: Australia Europe Japan
```

8. Construct a different tree by requiring at least 15 training observations to be within each partition. Create a visual representation of this tree

```
tree15<-rpart(BestMarket~.,
                  data = ExistingWines,
                  control = rpart.control(minbucket = 15))
rpart.plot(tree15)
```



9. Using R, find predictions for the first ten wines in the *NewWines.rds* using the tree obtained in the Question 8.

```
#Get predictions
yhat15<-predict(tree15,NewWines,type='class')
#The first ten predictions can be checked using the head function
head(yhat15,10)
```

```
##         1         2         3         4         5         6         7
## Australia Australia Australia Australia Australia Australia Australia
##         8         9        10
##    Europe Australia Australia
## Levels: Australia Europe Japan
```

10. Split the data in *ExistingWines.rds* into a training sample (of roughly 70%) and a test sample (of roughly 30%).

```
#This is the same problem as last week.  However since the lda and qda functions take in the
#data differently to the knn function
#Create an indicator that determines whether it is training or test sample.
ind<-ifelse(runif(125)<0.7,"Training Sample","Test Sample")

#A data set augmented with sample information
Data_with_Sample<-add_column(ExistingWines,Sample=ind)

#Get Training data
train_data<-Data_with_Sample%>%
  filter(Sample=="Training Sample")%>%
  select(-Sample) #Can remove Sample variable

#Get Test data
test_data<-Data_with_Sample%>%
  filter(Sample=="Test Sample")%>%
  select(-Sample) #Can remove Sample variable
```

11. Which tree is better for this data? How do these compare to the results from kNN and discriminant analysis from previous tutorials?

```
# Tree with minimum 15 observations per bin
tree<-rpart(BestMarket~.,
                  data = train_data)

yhat<-predict(tree,test_data,type='class')


# Tree with minimum 15 observations per bin
tree15<-rpart(BestMarket~.,
                  data = train_data,
                  control = rpart.control(minbucket = 15))

yhat15<-predict(tree15,test_data,type='class')

#Compare misclassification rate

#Default tree
mean(yhat!=test_data$BestMarket)
```

```
## [1] 0.1025641
```

```
#Default tree
mean(yhat15!=test_data$BestMarket)
```

```
## [1] 0.1794872
```

```
#For this particular example the more complicated tree performs better.  Both perform worse
#than LDA and QDA and k-NN with k=5.  The more complicated tree performs equally
#to kNN with k=1.
```