Practical Machine Learning - Week 3 - Random Forests

Jeff Leek, notes by Bruno Fischer Colonimos

24 juin 2017

Contents

---

1 Install first

randomForest, e1071

2 Random forests principles

1. Bootstrap samples
2. At each split, bootstrap variables
3. Grow multiple trees and **vote**

Pros:

1. Accuracy

Cons:

1. Speed
2. Interpretability
3. Overfitting

3 Auxiliairy material

```
# palette color-blind-friendly: The palette with black
cbbPalette <- c("#000000", "#E69F00", "#56B4E9", "#009E73",
                "#F0E442", "#0072B2", "#D55E00", "#CC79A7")
```

4 Iris Example

4.1 data

```
data(iris)
inTrain <- createDataPartition(y=iris$Species,
                               p=0.7, list=FALSE)
training <- iris[inTrain,]
testing <- iris[-inTrain,]
```

4.2 Train a Random forests model

```
set.seed(975)
modFit <- train(Species~ ., data=training,method="rf", prox=TRUE) # prox?
modFit
```

```
## Random Forest
##
## 105 samples
##   4 predictor
##   3 classes: 'setosa', 'versicolor', 'virginica'
##
## No pre-processing
## Resampling: Bootstrapped (25 reps)
## Summary of sample sizes: 105, 105, 105, 105, 105, 105, ...
## Resampling results across tuning parameters:
##
##   mtry  Accuracy   Kappa
##   2     0.9601194  0.9394334
##   3     0.9611181  0.9409730
##   4     0.9600655  0.9393591
##
## Accuracy was used to select the optimal model using  the largest value.
## The final value used for the model was mtry = 3.
```

4.3 Getting a single tree from the model

```
treetwo <- getTree(modFit$finalModel,k=2) # Tree no 2
# treetwo
kable(treetwo, row.names = TRUE, caption = "Tree 2")
```

Table 1: Tree 2

	left daughter	right daughter	split var	split point	status	prediction
1	2	3	4	1.75	1	0
2	4	5	4	0.75	1	0
3	0	0	0	0.00	-1	3
4	0	0	0	0.00	-1	1
5	6	7	3	5.05	1	0
6	8	9	4	1.60	1	0
7	10	11	4	1.55	1	0
8	0	0	0	0.00	-1	2
9	12	13	2	2.75	1	0
10	0	0	0	0.00	-1	3
11	0	0	0	0.00	-1	2
12	0	0	0	0.00	-1	3
13	0	0	0	0.00	-1	2

```
# next tree
treethree <- getTree(modFit$finalModel,k=3) # Tree no 3
kable(treethree, row.names = TRUE, caption = "Tree 3")
```

Table 2: Tree 3

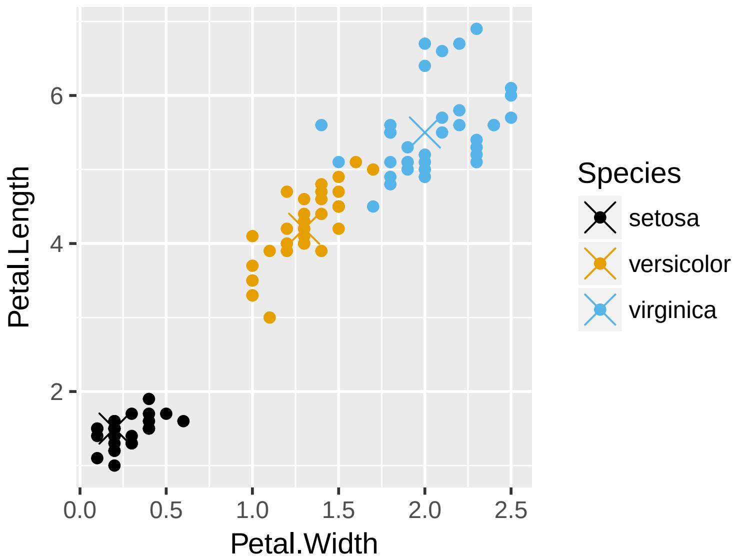
	left daughter	right daughter	split var	split point	status	prediction
1	2	3	3	2.35	1	0
2	0	0	0	0.00	-1	1
3	4	5	4	1.75	1	0
4	6	7	3	5.05	1	0
5	0	0	0	0.00	-1	3
6	0	0	0	0.00	-1	2
7	8	9	1	6.05	1	0
8	0	0	0	0.00	-1	2
9	0	0	0	0.00	-1	3

```
# library(rattle)
# library(rpart)
# fancyRpartPlot(treetwo,
#                main = "Tree 2",
#                palettes=c("Greys", "Oranges", "Reds") ) #==> Does not work
# These trees are differents
```

4.4 Getting Class "centers"

```
# get centers
irisP <- classCenter(training[,c(3,4)], training$Species, modFit$finalModel$prox) # return
irisP <- as.data.frame(irisP)
irisP$Species <- rownames(irisP)
```

```
p <- ggplot(data = training) +
        geom_point(aes(x=Petal.Width,y=Petal.Length,col=Species)) +
        geom_point(aes(x=Petal.Width,y=Petal.Length,col=Species),
                   size=5,shape=4,data=irisP)
p + scale_color_manual(values = cbbPalette)
```



4.5 Predicting new values

4.5.1 testing sample

```
pred <- predict(modFit,testing)
testing$predRight <- pred == testing$Species
# table(pred,testing$Species)
kable(table(pred,testing$Species), caption = "predicted (rows) vs species (columns)")
```

Table 3: predicted (rows) vs species (columns)

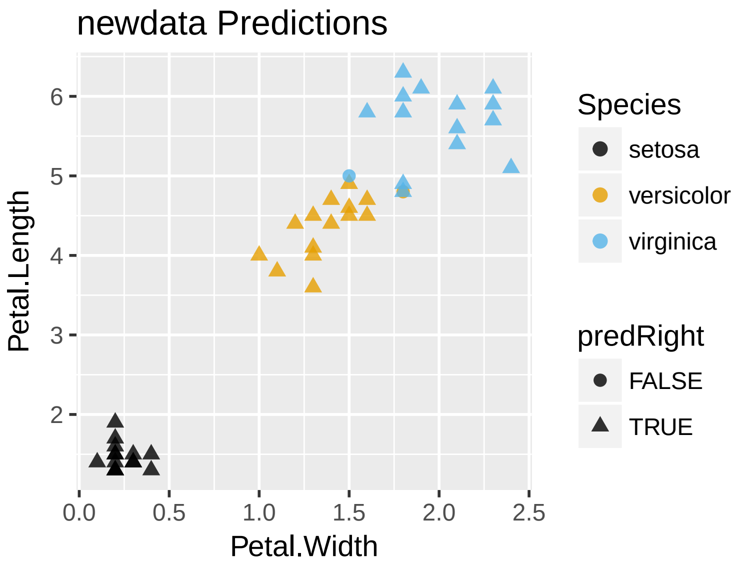
	setosa	versicolor	virginica
setosa	15	0	0
versicolor	0	14	1
virginica	0	1	14

4.5.2 Predicting new values : plot and check if true

```
# qplot(Petal.Width,Petal.Length,colour=predRight,data=testing,main="newdata Predictions

ggplot(data = testing,
       aes(Petal.Width,Petal.Length, colour = Species, shape = predRight)) +
        geom_point(size = 2, alpha = 0.8) +
```

```
scale_color_manual(values = cbbPalette) +
labs(title="newdata Predictions")
```



5 Notes and further resources

Notes:

- Random forests are usually one of the two top performing algorithms along with boosting in prediction contests.
- Random forests are difficult to interpret but often very accurate.
- Care should be taken to avoid overfitting (see rfcv funtion)

Further resources:

- Random forests
- Random forest Wikipedia
- Elements of Statistical Learning