RandomForest Notes

The flowers which we want to study for classification purposes are included below



Figure 1:

*#Install required packages*

**options**(repos='http://cran.rstudio.org')

have.packages <- **installed.packages**()

cran.packages <- **c**('devtools','plotrix','randomForest','tree','fifer')

to.install <- **setdiff**(cran.packages, have.packages[,1])

if(**length**(to.install)>0) **install.packages**(to.install)

**library**(devtools)

if(!('reprtree' %in% **installed.packages**())){

**install\_github**('araastat/reprtree')

}

for(p in **c**(cran.packages, 'reprtree')) **eval**(**substitute**(**library**(pkg), **list**(pkg=p)))

## randomForest 4.6-12

## Type rfNews() to see new features/changes/bug fixes.

## Warning: package 'fifer' was built under R version 3.4.3

## Loading required package: MASS

**library**(reprtree)

**library**(randomForest)

**require**(fifer)

**data**(iris)

Splitting the iris data set into equal parts (train, test) and check the result of splits

iris\_split<-**stratified**(df=iris,group="Species",replace=FALSE,size=25,bothSets = TRUE)

**str**(iris\_split)

1

## List of 2

## $ SET1:'data.frame': 75 obs. of 5 variables:

## ..$ Sepal.Length: num [1:75] 4.7 5.1 4.3 5.4 5.1 5.1 5 4.5 5 5.8 ...

## ..$ Sepal.Width : num [1:75] 3.2 3.5 3 3.7 3.4 3.8 3.5 2.3 3.4 4 ...

## ..$ Petal.Length: num [1:75] 1.3 1.4 1.1 1.5 1.5 1.6 1.6 1.3 1.6 1.2 ...

## ..$ Petal.Width : num [1:75] 0.2 0.2 0.1 0.2 0.2 0.2 0.6 0.3 0.4 0.2 ...

## ..$ Species : Factor w/ 3 levels "setosa","versicolor",..: 1 1 1 1 1 1 1 1 1 1 ...

## $ SET2:'data.frame': 75 obs. of 5 variables:

## ..$ Sepal.Length: num [1:75] 4.9 4.6 5 5.4 4.4 4.9 4.8 4.8 5.1 5.4 ...

## ..$ Sepal.Width : num [1:75] 3 3.1 3.6 3.9 2.9 3.1 3.4 3 3.5 3.4 ...

## ..$ Petal.Length: num [1:75] 1.4 1.5 1.4 1.7 1.4 1.5 1.6 1.4 1.4 1.7 ...

## ..$ Petal.Width : num [1:75] 0.2 0.2 0.2 0.4 0.2 0.1 0.2 0.1 0.3 0.2 ...

## ..$ Species : Factor w/ 3 levels "setosa","versicolor",..: 1 1 1 1 1 1 1 1 1 1 ...

Recode the names of stratified samples

iris\_train<-iris\_split$SET1

iris\_test<-iris\_split$SET2

Build the random forest model with the training set; the default number of trees generated are 500; make

sure you have installed randomForest package

**library**("randomForest", lib.loc="~/R/win-library/3.4")

rf\_train <- **randomForest**(Species ~ ., data=iris\_train)

Let us view the last 5 trees in text form as a table of the tree information as well as the graphic plots of

those last 5 trees from the list of 500 trained trees

for(i in 496:500){

tr\_struct = **getTree**(rf\_train, i, labelVar=TRUE)

**print**(tr\_struct)

**print**(iris[1,])

reprtree:::**plot.getTree**(rf\_train,k = i)

}

## left daughter right daughter split var split point status prediction

## 1 2 3 Petal.Length 2.60 1 <NA>

## 2 0 0 <NA> 0.00 -1 setosa

## 3 4 5 Petal.Width 1.75 1 <NA>

## 4 6 7 Petal.Length 5.05 1 <NA>

## 5 8 9 Sepal.Length 6.00 1 <NA>

## 6 0 0 <NA> 0.00 -1 versicolor

## 7 0 0 <NA> 0.00 -1 virginica

## 8 0 0 <NA> 0.00 -1 versicolor

## 9 0 0 <NA> 0.00 -1 virginica

## Sepal.Length Sepal.Width Petal.Length Petal.Width Species

## 1 5.1 3.5 1.4 0.2 setosa