```
##########classification################>
library(randomForest)
randomForest 4.6-12
> data(iris)
> help(iris)
```
This famous (Fisher's or Anderson's) iris data set gives the measurements in centimeters of the variables sepal length and width and petal length and width, respectively, for 50 flowers from each of 3 species of iris. The species are *Iris setosa*, *versicolor*, and *virginica*.

```
> nrow(iris)
[1] 150
> head(iris)
  Sepal.Length Sepal.Width Petal.Length Petal.Width Species
1          5.1         3.5          1.4         0.2  setosa
2          4.9         3.0          1.4         0.2  setosa
3          4.7         3.2          1.3         0.2  setosa
4          4.6         3.1          1.5         0.2  setosa
5          5.0         3.6          1.4         0.2  setosa
6          5.4         3.9          1.7         0.4  setosa

set.seed(71)
iris.rf <- randomForest(Species ~ ., data=iris, ntree=500, importance=TRUE,
                        proximity=TRUE)  #default is 500 trees


> iris.rf

Call:
 randomForest(formula = Species ~ ., data = iris, ntree = 500,      importance = TRUE,
proximity = TRUE)
               Type of random forest: classification
                     Number of trees: 500
No. of variables tried at each split: 2

        OOB estimate of  error rate: 4.67%
Confusion matrix:
           setosa  versicolor virginica class.error
setosa         50         0         0        0.00
versicolor      0        47         3        0.06
virginica       0         4        46        0.08

> predict(iris.rf)
     1      2      3      4      5      6      7
setosa setosa setosa setosa setosa setosa setosa
     8      9     10     11     12     13     14
setosa setosa setosa setosa setosa setosa setosa
    15     16     17     18     19     20     21
setosa setosa setosa setosa setosa setosa setosa
```

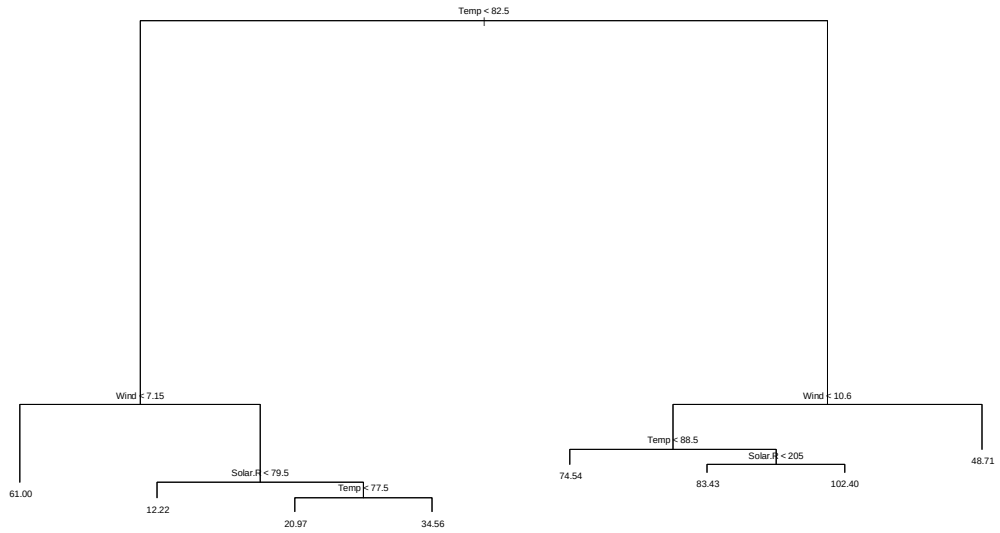
```
> importance(iris.rf) ##large values means more important
                setosa      versicolor virginica     MeanDecreaseAccuracy
Sepal.Length  6.51273   7.589459  7.470820           11.64546
Sepal.Width   4.48604   1.943485  5.783887           5.96269
Petal.Length 21.07643  32.630501 28.111405          32.11120
Petal.Width  23.62011  32.393569 31.696603          33.78858
                MeanDecreaseGini
Sepal.Length        9.529515
Sepal.Width         2.285106
Petal.Length       41.472200
Petal.Width        45.941726
>

> getTree(randomForest(iris[,-5], iris[,5]), 3, labelVar=TRUE)
   left daughter right daughter    split var         split point status prediction
1              2              3       Sepal.Length       5.55      1       <NA>
2              4              5       Petal.Length       2.45      1       <NA>
3              6              7       Petal.Width        1.70      1       <NA>
4              0              0       <NA>               0.00     -1     setosa
5              8              9       Petal.Width        1.60      1       <NA>
6             10             11       Sepal.Width        3.70      1       <NA>
7             12             13       Petal.Length       4.85      1       <NA>
8              0              0       <NA>               0.00     -1 versicolor
9              0              0       <NA>               0.00     -1  virginica
10            14             15       Petal.Length       4.95      1       <NA>
11             0              0       <NA>               0.00     -1     setosa
12            16             17       Sepal.Width        3.10      1       <NA>
13             0              0       <NA>               0.00     -1  virginica
14             0              0       <NA>               0.00     -1 versicolor
15             0              0       <NA>               0.00     -1  virginica
16             0              0       <NA>               0.00     -1  virginica
17             0              0       <NA>               0.00     -1 versicolor


> #########Regression:#########################
> data(airquality)
> nrow(airquality)
[1] 153
> head(airquality)
  Ozone   Solar.R  Wind Temp Month Day
1    41     190       7.4     67     5        1
2    36     118       8.0     72     5        2
3    12     149      12.6    74     5        3
4    18     313      11.5    62     5       4
5    NA      NA       14.3    56     5        5
6    28      NA       14.9    66     5        6
```

```
> library(tree)
> rt1 = tree(Ozone ~ .,data=airquality)
> rt1
node), split, n, deviance, yval   * denotes terminal node

 1) root 111 121800.0  42.10
   2) Temp < 82.5   77  42140.0  26.78
     4) Wind < 7.15 9  19320.0     61.00 *
     5) Wind > 7.15 68  10890.0  22.25
      10) Solar.R < 79.5 18    777.1  12.22 *
      11) Solar.R > 79.5 50   7648.0  25.86
        22) Temp < 77.5 32   2413.0  20.97 *
        23) Temp > 77.5 18   3108.0  34.56 *
   3) Temp > 82.5 34  20660.0  76.79
     6) Wind < 10.6 27  12530.0  84.07
      12) Temp < 88.5 13   7307.0  74.54 *
      13) Temp > 88.5 14   2939.0  92.93
        26) Solar.R < 205 7    413.7  83.43 *
        27) Solar.R > 205 7   1262.0 102.40 *
     7) Wind > 10.6 7   1183.0  48.71 *
>
plot(rt1)
```




```
> #random forest
> #random forest
> set.seed(131)
> newdata<-na.omit(airquality)#construct a new data without the missing value
> ozone.rf <- randomForest(Ozone ~ ., data=airquality, mtry=3,
+                          importance=TRUE, na.action=na.omit)
#mtry: number of variables randomly sampled as candidates at each split. Default
#value for classification is sqrt(p), p is the number of variables in x
```

```
> print(ozone.rf)

Call:
 randomForest(formula = Ozone ~ ., data = airquality, mtry = 3,      importance =
TRUE, na.action = na.omit)
               Type of random forest: regression
                     Number of trees: 500
No. of variables tried at each split: 3

         Mean of squared residuals: 299.5463
                   % Var explained: 72.7
> predict(ozone.rf)
       1        2        3        4        7        8        9
30.90665 23.65048 23.37471 22.12600 25.26299 17.17940 14.81550

attr(,"na.action")
 5  6 10 11 25 26 27 32 33 34 35 36 37 39 42 43 45 46 52


> ## Show "importance" of variables: higher value mean more important:
> round(importance(ozone.rf), 2)
        %IncMSE IncNodePurity
Solar.R   10.85      10627.11
Wind      22.82      44182.07
Temp      43.65      55002.66
Month      3.81       1691.60
Day        4.49       6672.37
>
#calculate mse
yhat<-ozone.rf$predicted
yhatnoob<-predict(ozone.rf,airquality[,-1])
res<-newdata$Ozone-yhat
mse<-sum(res^2)/nrow(newdata)
#calculate r^2
ssto<-sum((newdata$Ozone-mean(newdata$Ozone))^2)
sse<-sum(res^2)
rsq<-(ssto-sse)/ssto
rsq

> rsq
[1] 0.7270187
```