

# RF

```
library(mlbench)
data(Glass)
library(randomForest)
```

```
## randomForest 4.6-14
```

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

```
library(MASS)
library(caret)
```

```
## Loading required package: lattice
```

```
## Loading required package: ggplot2
```

```
## Registered S3 methods overwritten by 'ggplot2':
##   method      from
##   [.quosures   rlang
##   c.quosures   rlang
##   print.quosures rlang
```

```
##
## Attaching package: 'ggplot2'
```

```
## The following object is masked from 'package:randomForest':
##
##   margin
```

```
Glass$Type2 <- as.factor(c(rep('Window', 163), rep('Non-Window', 51)))
set.seed(42)
train <- createDataPartition(Glass$Type2, p=0.75, list = FALSE)
```

```
# sample size
ceiling(.632*nrow(Glass[-train,]))
```

```
## [1] 33
```

```
# number of vars at each split
floor(sqrt(ncol(Glass)))
```

```
## [1] 3
```

```
set.seed(42)
rf_class <- randomForest(Type2 ~ ., data = Glass,
  subset = train,
  mtry = 3,
  sampsize = 33,
  importance = T)
rf_class
```

```
##
## Call:
## randomForest(formula = Type2 ~ ., data = Glass, mtry = 3, sampsize = 33, importance = T, subset = train)
##      Type of random forest: classification
##      Number of trees: 500
##      No. of variables tried at each split: 3
##
##      OOB estimate of  error rate: 0.62%
##      Confusion matrix:
##      Non-Window Window class.error
## Non-Window      38      1 0.02564103
## Window          0     123 0.00000000
```

```
# test
est_medv <- predict(rf_class, newdata = Glass[-train,])
mean(est_medv != Glass$Type2[-train])
```

```
## [1] 0
```

```
set.seed(42)
ntrees <- 500
rf_class <- randomForest(Type2 ~ ., data = Glass,
  subset = train,
  ntree = ntrees,
  mtry = 3,
  sampsize = 33,
  importance = T,
  do.trace = ntrees/10)
```

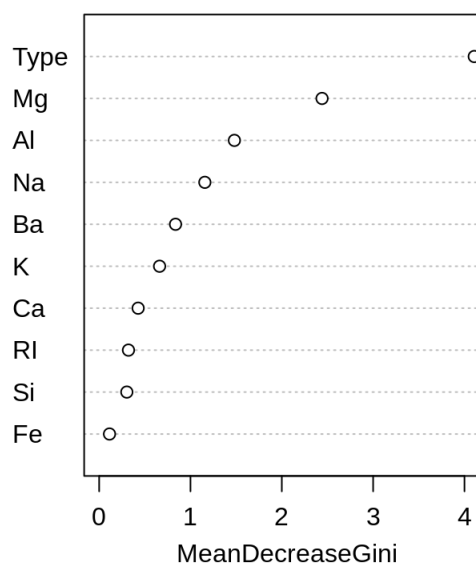
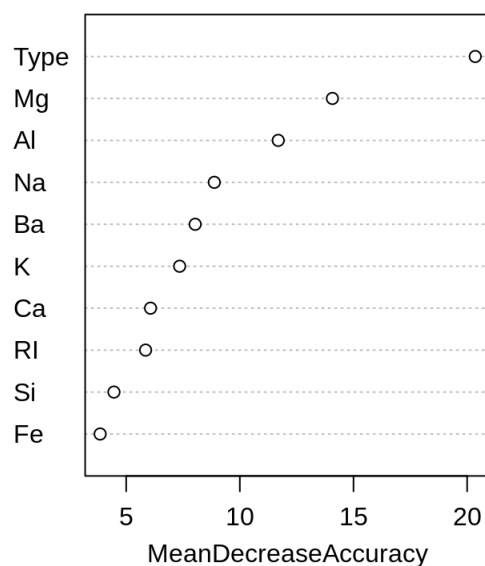
```
## ntree   OOB    1    2
##  50:  1.23%  2.56%  0.81%
## 100:  0.62%  2.56%  0.00%
## 150:  0.62%  2.56%  0.00%
## 200:  1.23%  2.56%  0.81%
## 250:  0.62%  2.56%  0.00%
## 300:  0.62%  2.56%  0.00%
## 350:  0.62%  2.56%  0.00%
## 400:  0.62%  2.56%  0.00%
## 450:  0.62%  2.56%  0.00%
## 500:  0.62%  2.56%  0.00%
```

```
rf_class
```

```
##
## Call:
## randomForest(formula = Type2 ~ ., data = Glass, ntree = ntrees,      mtry = 3, sampsize = 33, importance = T, do.trace = ntrees/10,      subset = train)
##      Type of random forest: classification
##      Number of trees: 500
## No. of variables tried at each split: 3
##
##      OOB estimate of  error rate: 0.62%
## Confusion matrix:
##      Non-Window Window class.error
## Non-Window      38    1 0.02564103
## Window           0   123 0.00000000
```

```
varImpPlot(rf_class)
```

rf\_class



```
set.seed(42)
ntrees <- 100
rf_class <- randomForest(Type2 ~ ., data = Glass,
  subset = train,
  ntree = ntrees,
  mtry = 3,
  sampsize = 33,
  importance = T,
  do.trace = ntrees/10)
```

```
## ntree   OOB    1    2
##   10: 2,47% 7,69% 0,81%
##   20: 1,85% 2,56% 1,63%
##   30: 1,85% 2,56% 1,63%
##   40: 1,23% 2,56% 0,81%
##   50: 1,23% 2,56% 0,81%
##   60: 0,62% 2,56% 0,00%
##   70: 0,62% 2,56% 0,00%
##   80: 0,62% 2,56% 0,00%
##   90: 0,62% 2,56% 0,00%
##  100: 0,62% 2,56% 0,00%
```

rf\_class

```
##
## Call:
## randomForest(formula = Type2 ~ ., data = Glass, ntree = ntrees, mtry = 3, sampsize = 33, importance = T, do.trace = ntrees/10, subset = train)
##      Type of random forest: classification
##      Number of trees: 100
## No. of variables tried at each split: 3
##
##      OOB estimate of  error rate: 0,62%
## Confusion matrix:
##      Non-Window Window class.error
## Non-Window      38    1 0,02564103
## Window          0   123 0,00000000
```