Lab 13

Daniel Tshiani

2025-06-14

```
library(randomForest)
```

```
## randomForest 4.7-1.2
```

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

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.1     v tibble    3.2.1
## v lubridate 1.9.4     v tidyr     1.3.1
## v purrr     1.0.2
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::combine()  masks randomForest::combine()
## x dplyr::filter()   masks stats::filter()
## x dplyr::lag()      masks stats::lag()
## x ggplot2::margin() masks randomForest::margin()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
load("../data/Auto-3.rda")
attach(Auto)
```

```
## The following object is masked from package:lubridate:
##
##     origin
```

```
## The following object is masked from package:ggplot2:
##
##     mpg
```

a

```
glimpse(Auto)
```

```
## Rows: 392
## Columns: 9
## $ mpg          <dbl> 18, 15, 18, 16, 17, 15, 14, 14, 14, 15, 15, 14, 15, 14, 2~
## $ cylinders    <dbl> 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 4, 6, 6, 6, 4, ~
## $ displacement <dbl> 307, 350, 318, 304, 302, 429, 454, 440, 455, 390, 383, 34~
## $ horsepower   <dbl> 130, 165, 150, 150, 140, 198, 220, 215, 225, 190, 170, 16~
## $ weight       <dbl> 3504, 3693, 3436, 3433, 3449, 4341, 4354, 4312, 4425, 385~
## $ acceleration <dbl> 12.0, 11.5, 11.0, 12.0, 10.5, 10.0, 9.0, 8.5, 10.0, 8.5, ~
```

```
## $ year         <dbl> 70, 70, 70, 70, 70, 70, 70, 70, 70, 70, 70, 70, 70, 70, 7~
## $ origin       <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 3, 1, 1, 1, 3, ~
## $ name         <fct> chevrolet chevelle malibu, buick skylark 320, plymouth sa~
```

```
Auto <- Auto%>%
  select(-name)
```

```
rf <- randomForest(mpg ~ ., data = Auto)
rf
```

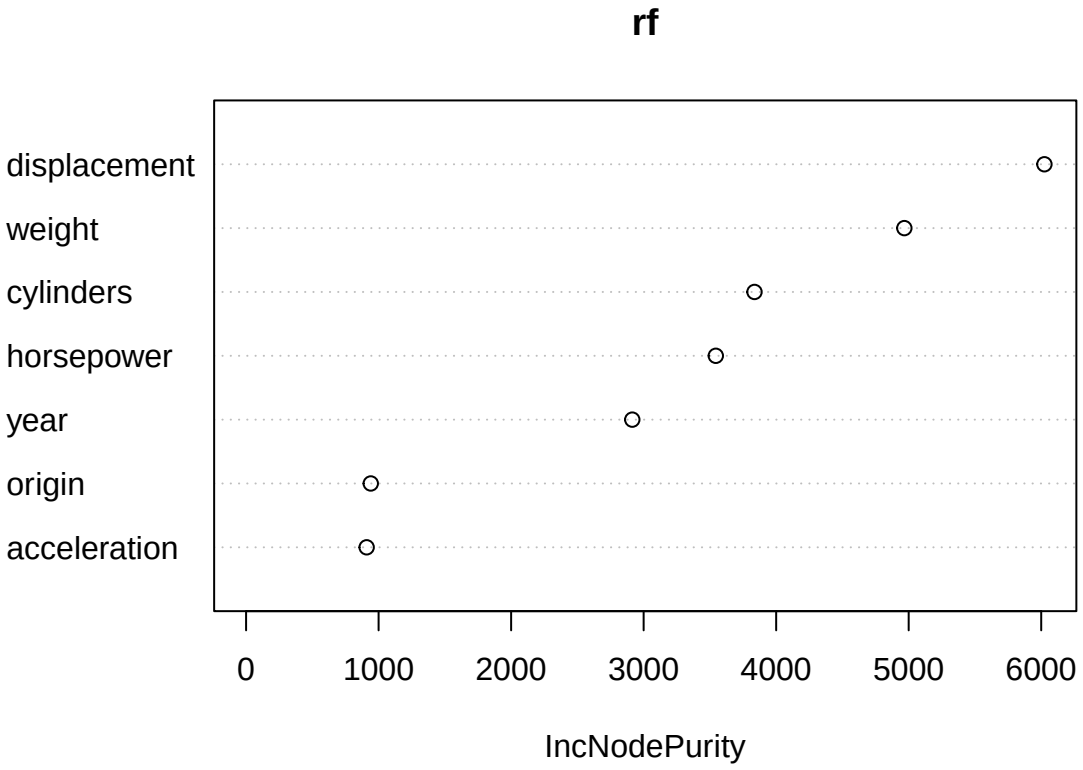
```
##
## Call:
##  randomForest(formula = mpg ~ ., data = Auto)
##                Type of random forest: regression
##                      Number of trees: 500
## No. of variables tried at each split: 2
##
##           Mean of squared residuals: 7.513397
##                     % Var explained: 87.63
```

b

```
importance(rf)
```

```
##              IncNodePurity
## cylinders        3836.4444
## displacement     6024.2771
## horsepower       3544.4831
## weight           4967.1548
## acceleration      909.8520
## year             2914.1633
## origin            940.8072
```

```
varImpPlot(rf)
```



displacement is considered the most important predictor.

c

```
set.seed(123)
train_index <- sample(1:nrow(Auto), 200)
train_data <- Auto[train_index, ]
test_data <- Auto[-train_index, ]
```

```
rf_model <- randomForest(mpg ~ . , data = train_data)
preds <- predict(rf_model, newdata = test_data)
mse <- mean((preds - test_data$mpg)^2)
mse
```

```
## [1] 7.64434
```

d

```
which.min(rf$mse)
```

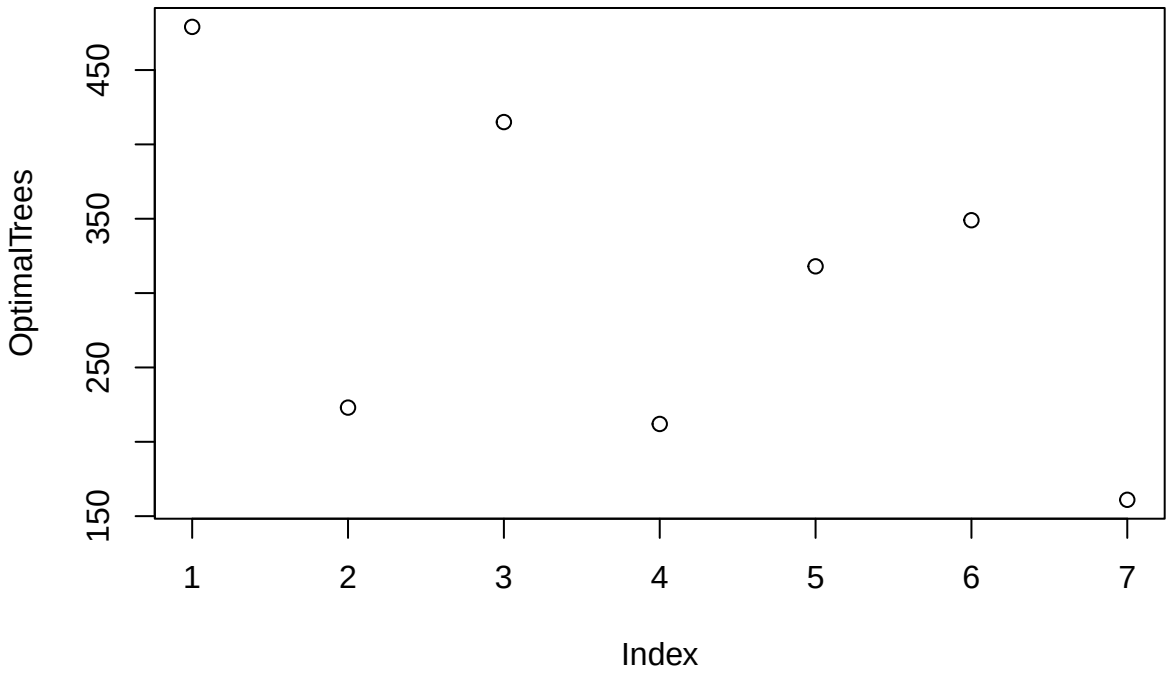
```
## [1] 474
```

e

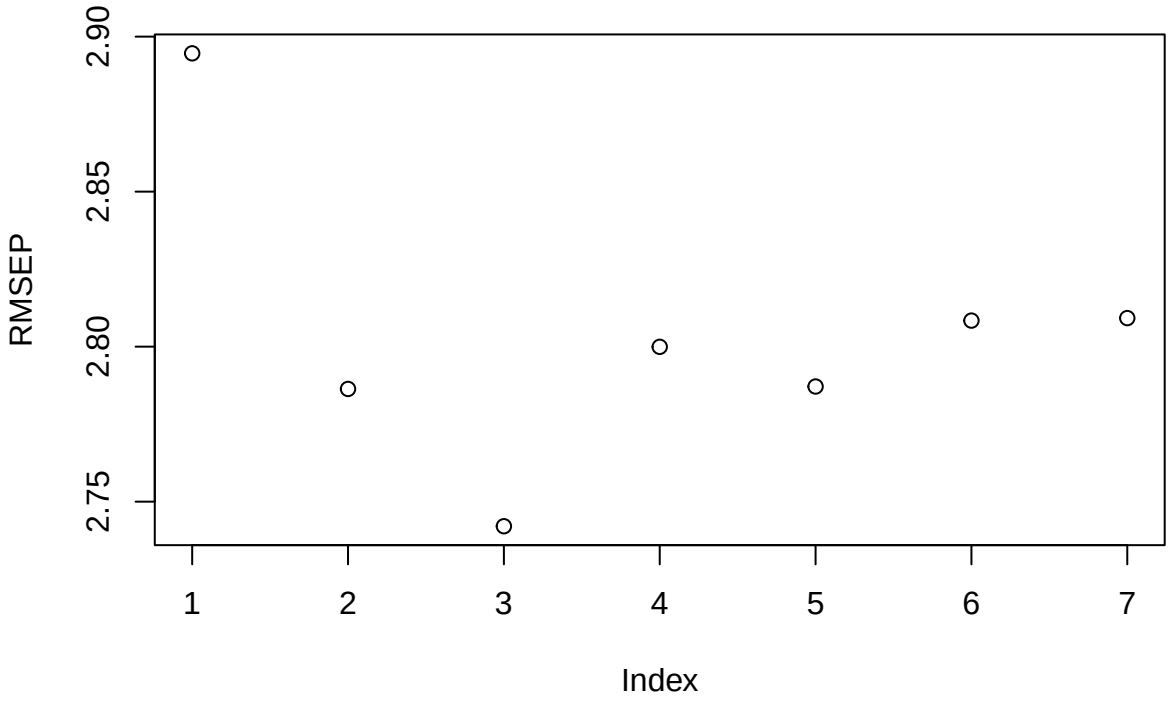
```
p <- length(Auto) - 1
RF <- OptimalTrees <- Yhat <- RMSEP <- vector(mode = "double", length = p)
```

```r
set.seed(123)
for (k in 1:p) {
  rf <- randomForest(mpg ~., data = train_data)
  OptimalTrees[k] <- which.min(rf$mse)
  rf <- randomForest(mpg ~ ., data = train_data, mtry = k, ntree = OptimalTrees[k])
  Yhat <- predict(rf, newdata = test_data)
  RMSEP[k] <- sqrt(mean((Yhat - test_data$mpg)^2))
}
```

```r
plot(OptimalTrees)
```




```r
plot(RMSEP)
```




```
which.min(RMSEP)
```

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

```
RMSEP[3]
```

```
## [1] 2.742069
```

```
OptimalTrees[3]
```

```
## [1] 415
```

f

```
RF <- randomForest(mpg ~ ., data = Auto, mtry = 7,
                   ntree = which.min(rf$mse))
rf
```

```
##
## Call:
##  randomForest(formula = mpg ~ ., data = train_data, mtry = k,      ntree = OptimalTrees[k])
##                Type of random forest: regression
##                      Number of trees: 161
## No. of variables tried at each split: 7
##
##           Mean of squared residuals: 9.039971
##                     % Var explained: 85.85
```