데이터과학을 위한 R프로그래밍

11주차. 의사결정나무와 랜덤포레스트

이혜선 교수
포항공과대학교 산업경영공학과

목차

11주차. 의사결정나무와 랜덤포레스트

11주차

3차시

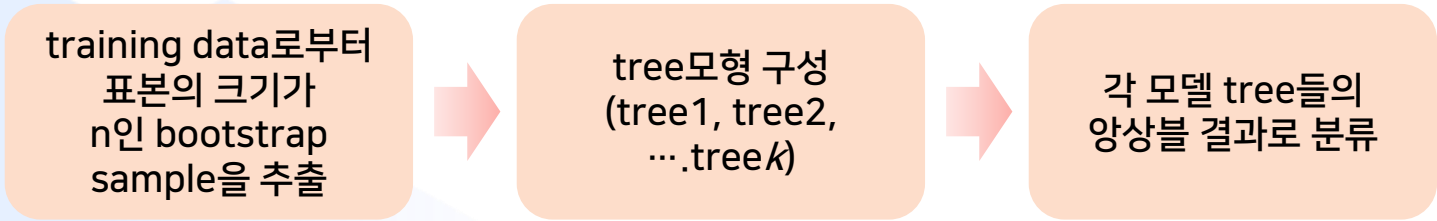
랜덤포레스트

POSTECH

랜덤포레스트(Random Forest) - 모형설명

랜덤포레스트(Random Forest)

- 2001년에 Leo Breiman에 의해 제안된 기법
- 의사결정나무의 단점(과적합)을 개선한 알고리즘
- 주어진 데이터의 리샘플링을 통해 생성한 다수의 의사결정나무 모델을 결합하여 정확도를 높이는 방법인 ensemble 기법

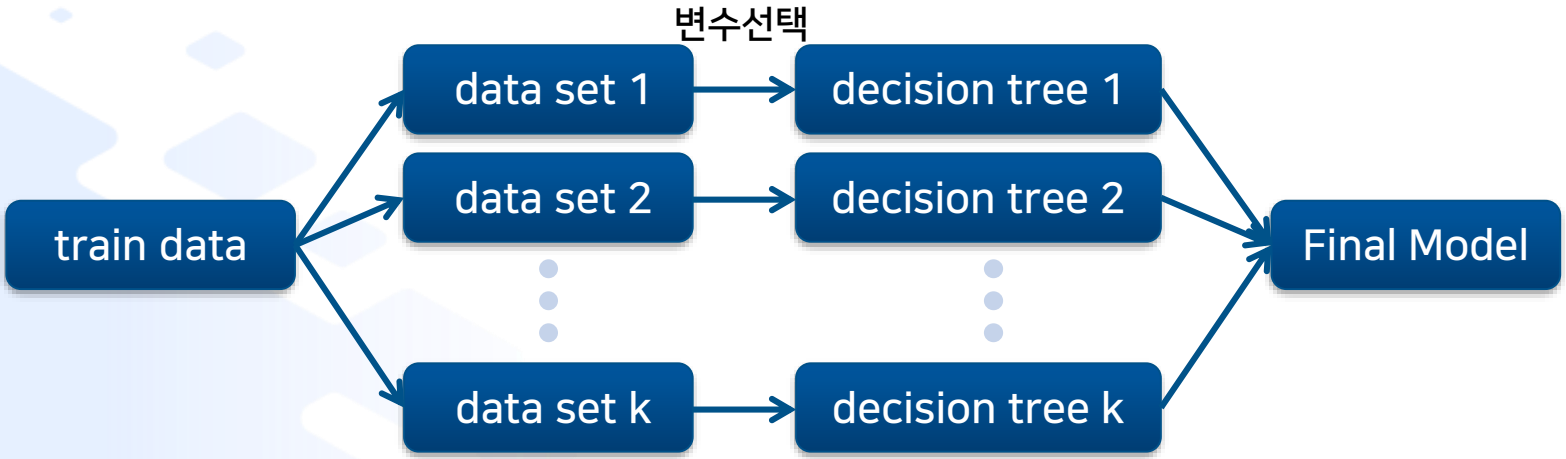


POSTECH

랜덤포레스트(Random Forest) - 모형설명

Bagging(Bootstrap Aggregating)

- 전체 데이터에서 샘플을 여러 번 추출(Bootstrap)하고 각 모델을 Aggregation하여 최종 모델을 도출하는 방법
- Training Data에서 Random Sampling



POSTECH

랜덤포레스트(Random Forest)

☑ 랜덤포레스트(Random forest) 패키지 : randomForest

```
# lec11_3_rf.R
# Random Forest using R

# random forest package
install.packages("randomForest")
library(randomForest)
help(randomForest)

# load caret package for confusion matrix
library(caret)
```

randomForest : 패키지 설치

caret : confusionMatrix

POSTECH

랜덤포레스트(Random Forest)

랜덤포레스트 실행 패키지 : randomForest 패키지

```
help(randomForest)
```

randomForest {randomForest} R Documentation

Classification and Regression with Random Forest

Description

`randomForest` implements Breiman's random forest algorithm (based on Breiman and Cutler's original Fortran code) for classification and regression. It can also be used in unsupervised mode for assessing proximities among data points.

Usage

```
## S3 method for class 'formula'
randomForest(formula, data=NULL, ..., subset, na.action=na.fail)
## Default S3 method:
randomForest(x, y=NULL,  xtest=NULL, ytest=NULL, ntree=500,
             mtry=if (!is.null(y) && !is.factor(y))
             max(floor(ncol(x)/3), 1) else floor(sqrt(ncol(x))),
             replace=TRUE, classwt=NULL, cutoff, strata,
             sampsize = if (replace) nrow(x) else ceiling(.632*nrow(x)),
             nodesize = if (!is.null(y) && !is.factor(y)) 5 else 1,
             maxnodes = NULL,
             importance=FALSE, localImp=FALSE, nPerm=1,
             proximity, oob.prox=proximity,
             norm.votes=TRUE, do.trace=FALSE,
             keep.forest=!is.null(y) && is.null(xtest), corr.bias=FALSE,
             keep.inbag=FALSE, ...)
## S3 method for class 'randomForest'
print(x, ...)
```

mtry : 독립변수의 수 (Number of variables randomly sampled, default = sqrt(k))

ntree : tree의 수 (number of decision trees to be grown

sampsize : 샘플사이즈 (sample size to be drawn from the input data for growing decision tree)

Importance :변수중요도

POSTECH

랜덤포레스트(Random Forest)

iris 데이터(iris.csv)

input변수(독립변수)

output변수(종속변수, 타겟변수)

Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
5.1	3.5	1.4	0.2	setosa
4.9	3	1.4	0.2	setosa
4.7	3.2	1.3	0.2	setosa
4.6	3.1	1.5	0.2	setosa
5	3.6	1.4	0.2	setosa
5.4	3.9	1.7	0.4	setosa
4.6	3.4	1.4	0.3	setosa
5	3.4	1.5	0.2	setosa
4.4	2.9	1.4	0.2	setosa
4.9	3.1	1.5	0.1	setosa
5.4	3.7	1.5	0.2	setosa
4.8	3.4	1.6	0.2	setosa
4.8	3	1.4	0.1	setosa

타겟변수(y) : setosa, versicolor, virginica

Iris setosa Iris versicolor Iris virginica

POSTECH

랜덤포레스트(Random Forest)

iris 데이터(학습데이터와 검증데이터의 분할)

```
# set working directory
setwd("D:/tempstore/moocr")

# read csv file
iris<-read.csv("iris.csv")
attach(iris)

# training (n=100)/ test data(n=50)
set.seed(1000)
N<-nrow(iris)
tr.idx<-sample(1:N, size=N*2/3, replace=FALSE)
# split train data and test data
train<-iris[tr.idx,]
test<-iris[-tr.idx,]
#dim(train)
#dim(test)
```

데이터분할
(학습데이터 2/3, 검증데이터 1/3)

train(100개의 데이터)
test(50개의 데이터)

POSTECH

랜덤포레스트(Random Forest)

☑ 랜덤포레스트 : randomForest(종속변수~x1+x2+x3+x4, data=)

mtry=임의적으로 선택되는 독립변수의 수 (default=sqrt(k))

```
#Random Forest : mtry=2 (default=sqrt(p))
rf_out1<-randomForest(Species~.,data=train,importance=T)
rf_out1
```

```
> rf_out1

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

        OOB estimate of  error rate: 6%
Confusion matrix:
           setosa versicolor virginica class.error
setosa         34          0         0  0.00000000
versicolor      0         32         3  0.08571429
virginica       0          3        28  0.09677419
```

POSTECH

랜덤포레스트(Random Forest)

☑ 랜덤포레스트 : randomForest(종속변수~x1+x2+x3+x4, data=)

mtry=임의적으로 선택되는 독립변수의 수 (default=sqrt(k))

```
#Random Forest : mtry=4
rf_out2<-randomForest(Species~.,data=train,importance=T, mrty=4)
rf_out2
```

```
> rf_out2

Call:
 randomForest(formula = Species ~ ., data = train, importance = T,
 mrty = 4)
               Type of random forest: classification
                     Number of trees: 500
No. of variables tried at each split: 2

        OOB estimate of  error rate: 6%
Confusion matrix:
           setosa versicolor virginica class.error
setosa         34          0         0  0.00000000
versicolor      0         32         3  0.08571429
virginica       0          3        28  0.09677419
```

mtry=4, mtry=2 결과 동일

POSTECH

랜덤포레스트(Random Forest)

☑ 변수의 중요도 : random forest결과로부터 중요변수 확인

```
# important variables for RF
round(importance(rf_out2), 2)
```

```
> round(importance(rf_out2), 2)
             setosa versicolor virginica MeanDecreaseAccuracy
Sepal.Length   6.07       3.14      8.11                 9.90
Sepal.Width    3.60       0.41      7.06                 5.71
Petal.Length  20.91      22.80     25.39                28.42
Petal.Width   22.05      29.08     41.10                37.07
             MeanDecreaseGini
Sepal.Length             5.58
Sepal.Width              2.71
Petal.Length            25.76
Petal.Width             31.82
```

분류의 정확도에 기여도가 높은 변수

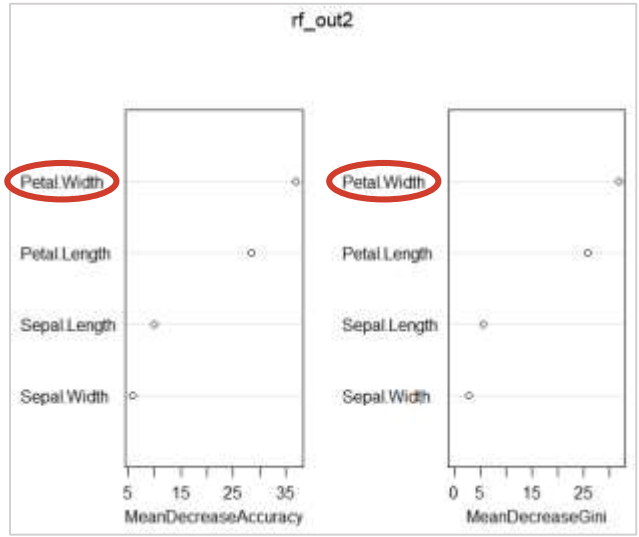
POSTECH

랜덤포레스트(Random Forest)

변수의 중요도 : random forest결과로부터 중요변수 확인

```
# graph
randomForest::importance(rf_out2)
varImpPlot(rf_out2)
```

```
> randomForest::importance(rf_out2)
                setosa versicolor virginica
Sepal.Length  6.073126  3.1392373  8.111493
Sepal.Width   3.600410  0.4050135  7.055566
Petal.Length 20.906314 22.7970178 25.388461
Petal.Width  22.045235 29.0785381 41.100485
             MeanDecreaseAccuracy MeanDecreaseGini
Sepal.Length             9.899308          5.583104
Sepal.Width              5.705388          2.706694
Petal.Length            28.416660         25.755489
Petal.Width             37.073551         31.816306
```



POSTECH

랜덤포레스트(Random Forest)

☑ 랜덤포레스트 결과 정확도 : test data에 대한 정확도

```
#measuring accuracy(rf)
rfpred<-predict(rf_out2,test)
confusionMatrix(rfpred,test$Species)
```

```
> confusionMatrix(rfpred,test$Species)
Confusion Matrix and Statistics

            Reference
Prediction    setosa versicolor virginica
  setosa          16          0         0
  versicolor       0         15         2
  virginica        0          0        17

Overall Statistics

               Accuracy : 0.96
```

정확도 96%