앙상블

- 앙상블(ensemble)은 여러 분류모형에 의한 결과를 종합하여 분류의 정확도를 높이는 방법
- 이는 적절한 표본추출법으로 데이터에서 여러 개의 훈련용 데이터 집합을 만들어 각각의 데이터 집합에서 하나의 분류기를 만들어 앙상블 하는 방법
  - 즉, 새로운 자료에 대해 분류기 예측값들의 가중 투표(weighted vote)를 통해 분류를 수행
- 데이터를 조절하는 가장 대표적인 방법에는 배깅(bagging)과 부스팅(boosting)
- 랜덤포리스트(random forest) 방법은 배깅의 개념과 속성(또는 변수)의 임의 선택(random
- selection)을 결합한 앙상블 기법

- 장점
  - 평균을 취함으로써 편의를 제거
    - 치우침이 있는 여러 모형의 평균을 취하면, 어느 쪽에도 치우치지 않는 결과(평균)를 얻게 됨
  - 분산이 감소
    - 한 개 모형으로부터의 단일 의견보다 여러 모형의 의견을 결합하면 변동이 축소
  - 과적합의 가능성 감소
    - 과적합이 없는 각 모형으로부터 예측을 결합(평균, 가중 평균, 로지스틱 회귀)하면 과적합의 여지가 감소

배깅

- 배깅(bagging : bootstrap aggregating)
  - 원 데이터로 부터 크기가 같은 표본을 여러 번 단순임의 복원 추출하여 각 표본(이를 붓스트랩 표본이라 함)에 대해 분류기(classifiers)를 생성한 후 그 결과를 앙상블 하는 방법
  - 반복추출 방법을 사용하기 때문에 같은 데이터가 한 표본에 여러 번 추출될 수도 있고, 어떤 데이터는 추출되지 않을 수도 있음
  - 데이터가 충분히 큰 경우, 각 데이터가 하나의 붓스트랩 표본에서 제외될 확률은 36.78%

$$\lim_{n\to\infty}\left(1-\frac{1}{n}\right)^{n} = e^{-1} = 0.3678$$

- **배깅(adabag 패키지)**

```
> bagging(formula, data, mfinal=100)
```

  - **formula : 모형식**
    - **class ~ $x_1+x_2+\cdots$**
  - **data : 분석 데이터(data frame)**
  - **mfinal : 반복수 또는 트리의 수**

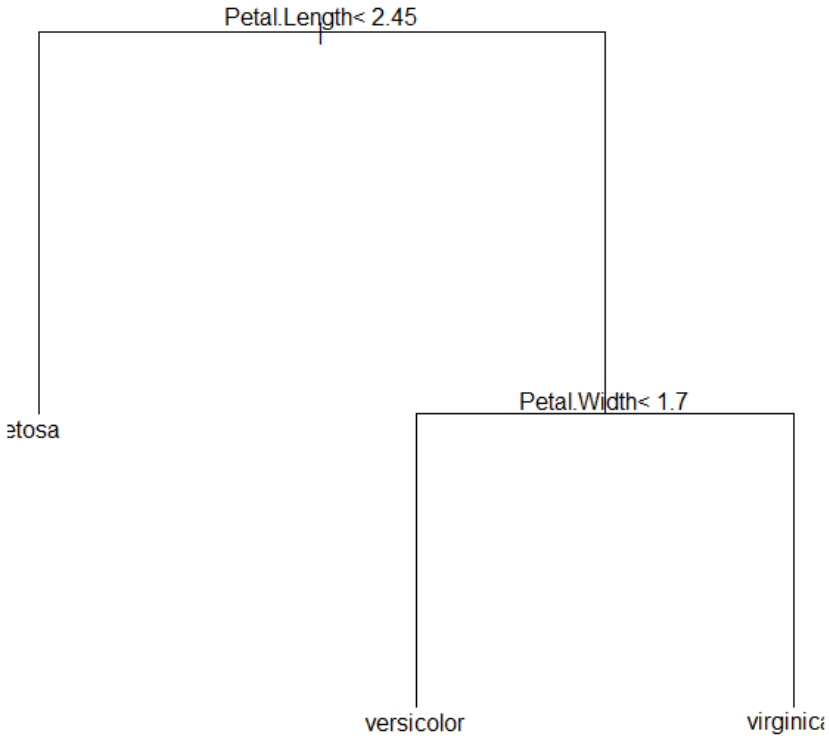
예제 4.1

- **다음 붓꽃(iris.csv)의 종(Species)을 배깅 방법을 이용하여 분류하시오.**
  - **종 : setosa, versicolor, virginica**

```
> library(adabag)
> library(gmodels) # for CrossTable
>
> iris.data<-read.csv("iris.csv")
> head(iris.data)
  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
```

```
> iris.bagging<-bagging(Species~., data=iris.data,
+                                 mfinal=10) # 배깅
> # 변수의 상대적인 중요도
> iris.bagging$importance
Petal.Length  Petal.Width Sepal.Length  Sepal.Width
    76.26441     23.73559       0.00000       0.00000
> # 10번째 분류결과 트리
> windows()
> plot(iris.bagging$trees[[10]])
> text(iris.bagging$trees[[10]])
>
```

- **변수의 중요도는 각 트리에서 변수에 의해 주어지는 지니지수의 이익(gain) (또는 불확실성의 감소량)을 고려한 측도**




```
> iris.bagging$trees[[10]]
n= 150

node), split, n, loss, yval, (yprob)
      * denotes terminal node

1) root 150 96 setosa (0.36000000 0.29333333 0.34666667)
  2) Petal.Length< 2.45 54  0 setosa (1.00000000 0.00000000 0.00000000) *
  3) Petal.Length>=2.45 96 44 virginica (0.00000000 0.45833333 0.54166667)
    6) Petal.Width< 1.7 48  4 versicolor (0.00000000 0.91666667 0.08333333) *
    7) Petal.Width>=1.7 48  0 virginica (0.00000000 0.00000000 1.00000000) *
> pred<-predict(iris.bagging,
+                         newdata=iris.data[,-5]) # 예측
> # 분류표
> CrossTable(pred$class, iris.data$Species,
+                  prop.chisq=FALSE, prop.t=FALSE,
+                  prop.r=FALSE,
+                  dnn=c('predicted', 'actual'))
```

```
 Cell Contents
|-------------------------|
|                       N |
|           N / Col Total |
|-------------------------|

Total Observations in Table:  150

             | actual
   predicted |     setosa | versicolor |  virginica |  Row Total |
-------------|------------|------------|------------|------------|
      setosa |         50 |          0 |          0 |         50 |
             |      1.000 |      0.000 |      0.000 |            |
-------------|------------|------------|------------|------------|
  versicolor |          0 |         47 |          1 |         48 |
             |      0.000 |      0.940 |      0.020 |            |
-------------|------------|------------|------------|------------|
   virginica |          0 |          3 |         49 |         52 |
             |      0.000 |      0.060 |      0.980 |            |
-------------|------------|------------|------------|------------|
Column Total |         50 |         50 |         50 |        150 |
             |      0.333 |      0.333 |      0.333 |            |
-------------|------------|------------|------------|------------|
```

```
> # 분류율
> acc<-mean(pred$class==iris.data$Species)
> acc
[1] 0.9733333
> detach("package:gmodels", unload=TRUE)
> detach("package:adabag", unload=TRUE)
```

부스팅

- 부스팅(boosting)은 배깅의 과정과 유사하나 붓스트랩 표본을 구성하는 재표본(re-sampling) 과정에서 각 자료에 동일한 확률을 부여하는 것이 아니라, 분류가 잘못된 데이터에 더 큰 가중을 주어 표본을 추출
- 부스팅에서는 붓스트랩 표본을 추출하여 분류기를 만든 후, 그 분류결과를 이용하여 각 데이터가 추출될 확률을 조정한 후, 다음 붓스트랩 표본을 추출하는 과정을 반복
- 아다부스팅(AdaBoosting: adaptive boosting)은 가장 많이 사용되는 부스팅 알고리즘

- 배깅(adabag 패키지)

```
> boosting(formula, data, boos=TRUE, mfinal=100)
```

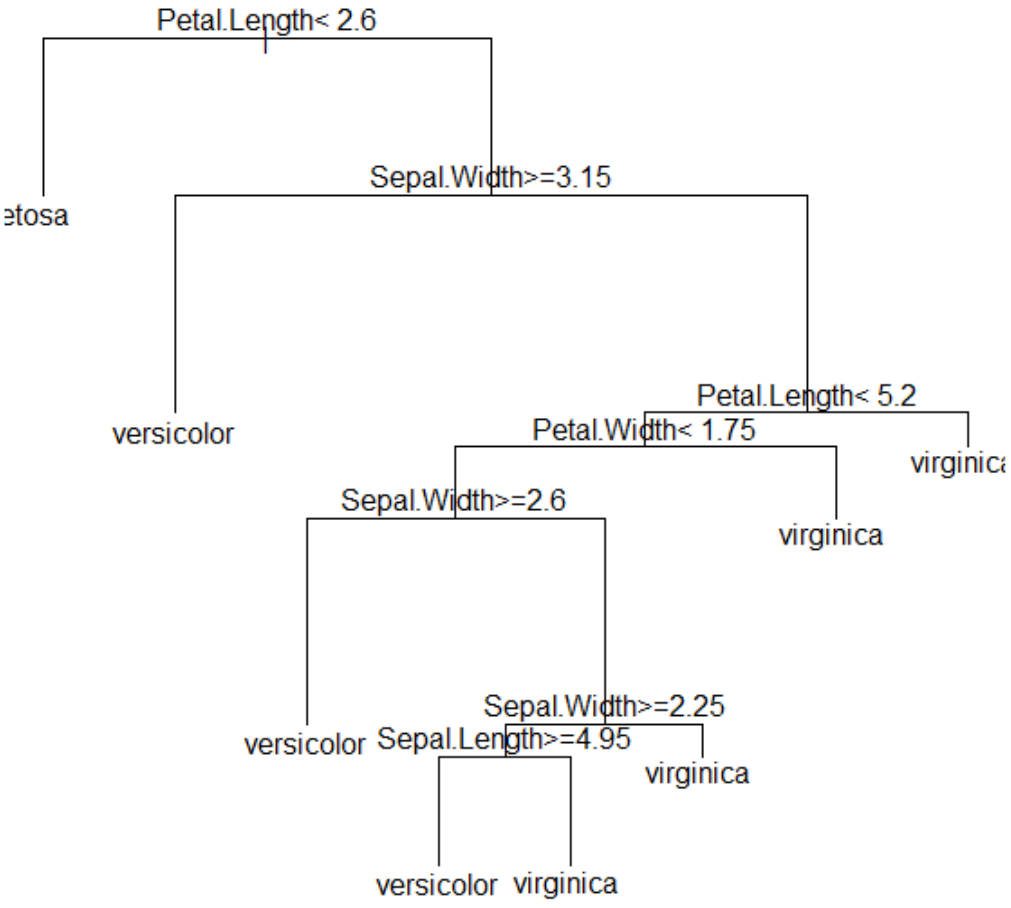
  - formula : 모형식
    - class ~ $x_1+x_2+\cdots$
  - data : 분석 데이터(data frame)
  - mfinal : 반복수 또는 트리의 수

예제 4.2

- 다음 붓꽃(iris.csv)의 종(Species)을 배깅 방법을 이용하여 분류하시오.
  - 종 : setosa, versicolor, virginica

```
> library(adabag)
> library(gmodels) # for CrossTable
>
> iris.data<-read.csv("iris.csv")
> head(iris.data)
  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
```

```
> boo.adabag<-boosting(Species~., data=iris.data,
+                      boos=TRUE, mfinal=10)  # 부스팅
> # 변수의 상대적인 중요도
> boo.adabag$importance
Petal.Length  Petal.Width Sepal.Length  Sepal.Width
   63.291999    23.092401     4.450154     9.165445
> # 10번째 분류결과 트리
> windows()
> plot(boo.adabag$trees[[10]])
> text(boo.adabag$trees[[10]])
```



```
> boo.adabag$trees[[10]]
n= 150

node), split, n, loss, yval, (yprob)
      * denotes terminal node

1) root 150 68 virginica (0.08666667 0.36666667 0.54666667)
  2) Petal.Length< 2.6 13  0 setosa (1.00000000 0.00000000 0.00000000) *
  3) Petal.Length>=2.6 137 55 virginica (0.00000000 0.40145985 0.59854015)
    6) Sepal.Width>=3.15 24  3 versicolor (0.00000000 0.87500000 0.12500000) *
    7) Sepal.Width< 3.15 113 34 virginica (0.00000000 0.30088496 0.69911504)
     14) Petal.Length< 5.2 82 34 virginica (0.00000000 0.41463415 0.58536585)
       28) Petal.Width< 1.75 62 28 versicolor (0.00000000 0.54838710 0.45161290)
         56) Sepal.Width>=2.6 27  2 versicolor (0.00000000 0.92592593 0.07407407) *
         57) Sepal.Width< 2.6 35  9 virginica (0.00000000 0.25714286 0.74285714)
          114) Sepal.Width>=2.25 21  9 virginica (0.00000000 0.42857143 0.57142857)
            228) Sepal.Length>=4.95 9  0 versicolor (0.00000000 1.00000000 0.00000000)
            229) Sepal.Length< 4.95 12  0 virginica (0.00000000 0.00000000 1.00000000)*
          115) Sepal.Width< 2.25 14  0 virginica (0.00000000 0.00000000 1.00000000) *
       29) Petal.Width>=1.75 20  0 virginica (0.00000000 0.00000000 1.00000000) *
     15) Petal.Length>=5.2 31  0 virginica (0.00000000 0.00000000 1.00000000) *
```

```
> # 예측
> pred<-predict(boo.adabag,
+                         newdata=iris.data[,-5])
> # 분류표
> CrossTable(pred$class, iris.data$Species,
+                 prop.chisq=FALSE, prop.t=FALSE,
+                 prop.r=FALSE,
+                 dnn=c('predicted', 'actual'))
> # 분류율
> acc<-mean(pred$class==iris.data$Species)
> acc
[1] 1
> detach("package:gmodels", unload=TRUE)
> detach("package:adabag", unload=TRUE)
```

Cell Contents

|-------------------------|
| N |
N / Col Total

Total Observations in Table: 150

	actual			
predicted	setosa	versicolor	virginica	Row Total
setosa	50	0	0	50
	1.000	0.000	0.000	
versicolor	0	50	0	50
	0.000	1.000	0.000	
virginica	0	0	50	50
	0.000	0.000	1.000	
Column Total	50	50	50	150
	0.333	0.333	0.333	

랜덤포리스트

- 랜덤포리스트(random forest)는 배깅에 랜덤 과정을 추가한 방법
  - 여러 개의 의사결정나무가 내놓은 예측결과를 투표방식으로 예측하는 알고리즘
- 원 자료로부터 붓스트랩 샘플을 추출하고, 각 붓스트랩 샘플에 대해 트리를 형성해 나가는 과정은 배깅과 유사하나, 각 노드마다 모든 예측변수 안에서 최적의 분할(split)을 선택하는 방법 대신 예측변수들을 임의로 추출하고, 추출된 변수 내에서 최적의 분할을 만들어 나가는 방법을 사용
- 따라서 나무의 수와 변수의 수가 중요 모수

- 새로운 자료에 대한 예측은 분류(classification)의 경우는 다수결(majority votes)로, 회귀(regression)의 경우에는 평균을 취하는 방법을 사용
  - 이는 다른 앙상블 모형과 동일
- 이 같은 방식은 기존 하나의 의사결정나무를 사용할 때보다 overfitting(과적합)문제는 해결
- 하지만 의사결정나무를 사용할 때는 예측 과정을 이해하기 쉽지만, 랜덤포레스트를 사용하게 되면 더 이상 어떤 과정을 통해 예측이 이뤄지는지 알기 어려움(블랙박스)

- 배깅(adabag 패키지)

```
> randomForest(formula, data=NULL, ntree=100,
+                      mtry)
```

  - formula : 모형식
    - class ~ $x_1+x_2+\cdots$
  - data : 분석 데이터(data frame)
  - ntree : 반복수 또는 트리의 수
  - mtry : 각각의 tree마다 몇 개의 예측변수를 사용할 것인지를 정하는 것
    - regression의 경우 변수개수/3
    - classification의 경우 sqrt(변수개수)

예제 4.3

- 다음 붓꽃(iris.csv)의 종(Species)을 배깅 방법을 이용하여 분류하시오.
  - 종 : setosa, versicolor, virginica

```
> # install.packages("randomForest")
> library(randomForest)
> library(gmodels) # for CrossTable
>
> iris.data<-read.csv("iris.csv")
> head(iris.data)
  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
```

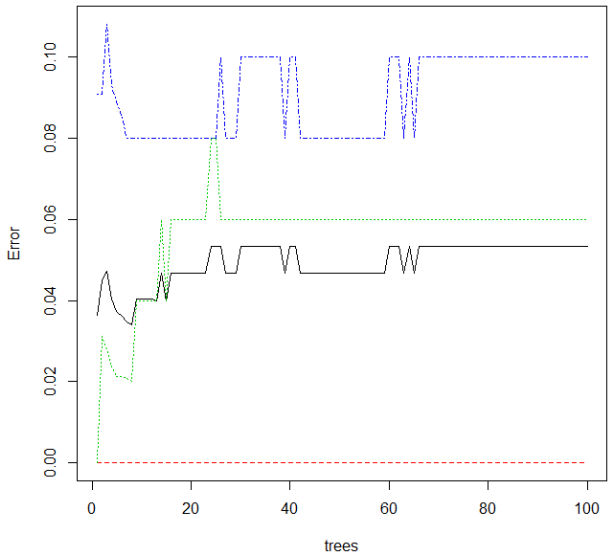
```
> # 랜덤포리스트
> rf<-randomForest(Species~., data=iris.data,
+                  ntree=100, mtry=2, importance=TRUE)
> rf  # 랜덤포리스트 결과

Call:
 randomForest(formula = Species ~ ., data = iris.data, ntree =
100,       mtry = 2, importance = TRUE)
               Type of random forest: classification
                     Number of trees: 100
No. of variables tried at each split: 2

        OOB estimate of  error rate: 5.33%
Confusion matrix:
           setosa versicolor virginica class.error
setosa         50          0         0        0.00
versicolor      0         47         3        0.06
virginica       0          5        45        0.10
```

- **앞의 결과는 정오분류표(confusion matrix)와 함께, 오류율에 대한 OOB(out-of-bag) 추정치를 제공**
- **랜덤포리스트에서는 별도의 검증용 데이터를 사용하지 않더라도 붓스트랩 샘플과정에서 제외된(out-of-bag)자료를 사용하여 검증을 실시**

```
> head(rf$err.rate)   # 에러율
          OOB      setosa versicolor     virginica
[1,] 0.03636364     0    0.00000000 0.09090909
[2,] 0.04494382     0    0.03125000 0.09090909
[3,] 0.04716981     0    0.02777778 0.10810811
[4,] 0.04032258     0    0.02380952 0.09302326
[5,] 0.03731343     0    0.02127660 0.08888889
[6,] 0.03623188     0    0.02127660 0.08510638
> plot(rf)
```

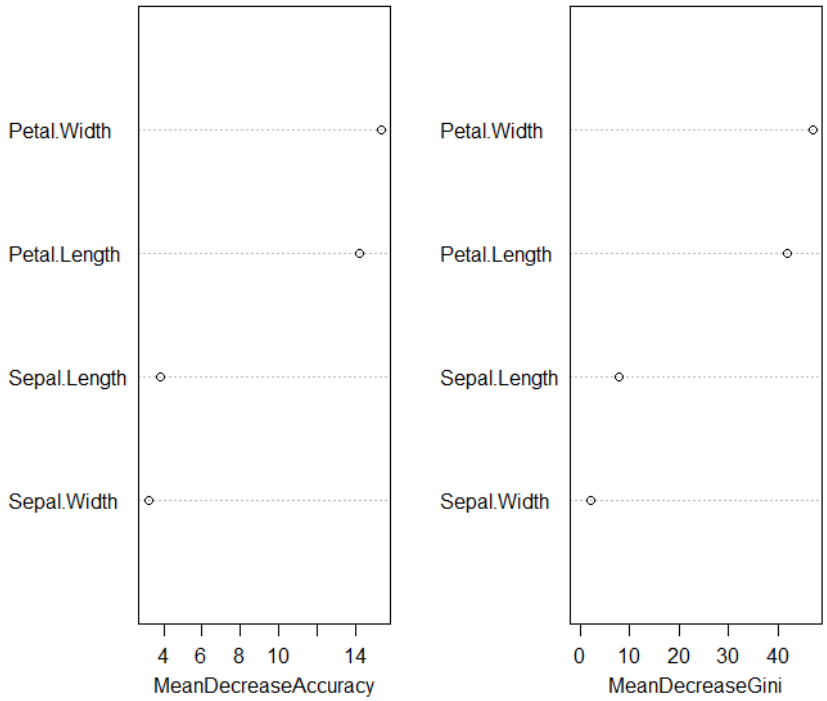


- **트리 수에 따른 종속변수의 범주별 오류율**
  - **검은색 : 전체 오류율**
  - **파랑색 : virginica 오류율**
  - **녹색 : versicolor 오류율**
  - **빨강색 : setosa 오류율**

```
> round(importance(rf),3) # 변수의 중요도
                setosa    versicolor virginica
Sepal.Length   2.332       3.128      2.582
Sepal.Width    1.617       1.204      3.343
Petal.Length 10.020      13.936     12.723
Petal.Width    9.932      15.963     14.672
             MeanDecreaseAccuracy MeanDecreaseGini
Sepal.Length                3.805            7.959
Sepal.Width                 3.187            2.229
Petal.Length               14.241           41.946
Petal.Width                15.369           47.057
> varImpPlot(rf)
```

- 앞의 그림은 각 변수의 중요도를 나타내는 그림으로, 해당 변수로부터 분할이 일어날 때 불순도(impurity)의 감소가 얼마나 일어나는지를 나타내는 값
  - 불순도의 감소가 클수록 순수도가 증가
  - 지니 지수(Gini index)는 노드의 불순도를 나타내는 값
  - 회귀의 경우에는 잔차제곱합(residual sum of square)을 통해 측정
- 노드 불순도 개선에 중요한 변수는 MeanDecreaseGini값으로 판단 Petal.Width>Petal.Length>Sepal.Length> Sepal.Width 순서

- **정확도 개선에 중요한 변수는 MeanDecreaseAccuracy으로 판단 Petal.Width>Petal.Length>Sepal.Length> Sepal.Width 순서**




```
> pred<-predict(rf) # 예측
> # 분류표
> CrossTable(pred, iris.data$Species,
+                    prop.chisq=FALSE, prop.t=FALSE,
+                    prop.r=FALSE,
+                    dnn=c('predicted', 'actual'))
> acc<-mean(pred==iris.data$Species) # 분류율
> acc
[1] 0.9466667
>
> detach("package:gmodels", unload=TRUE)
> detach("package:randomForest", unload=TRUE)
```

Cell Contents

	N
	N / Col Total

Total Observations in Table: 150

predicted \ actual	setosa	versicolor	virginica	Row Total
setosa	50 1.000	0 0.000	0 0.000	50
versicolor	0 0.000	47 0.940	5 0.100	52
virginica	0 0.000	3 0.060	45 0.900	48
Column Total	50 0.333	50 0.333	50 0.333	150

예제 4.4

- 연어는 강의 상류천에서 부화한 후 바다로 나아가 생활하게 된다. 그러다 산란기가 되면 알을 낳기 위하여 다시 자신이 태어난 곳으로 되돌아와 산란 후 최후의 죽음을 맞이하게 된다. 아래 표 (salmon.txt)는 알래스카와 캐나다 두 지역에서 부화한 연어의 크기를 측정한 결과로서 X_1은 강물에서, X_2는 바다물에서 성장한 길이를 각각 나타낸다. 랜덤포리스트를 이용하여 분류하시오.
  - 훈련자료 : 60%
  - 검정자료 : 40%

알래스카(group1)		캐나다(group2)		알래스카(group1)		캐나다(group2)	
X1	X2	X1	X2	X1	X2	X1	X2
108	368	129	420	102	429	145	376
131	355	148	371	101	469	115	354
105	469	179	409	85	444	134	383
86	506	152	381	109	397	117	355
99	402	166	377	106	442	126	345
87	423	124	389	82	431	118	379
94	440	156	419	118	381	120	369
117	489	131	345	105	388	153	403
79	432	140	362	121	403	150	354
99	403	144	345	85	451	154	390
114	428	149	393	83	453	155	349
123	372	108	330	53	427	109	325
123	372	135	355	95	411	117	344
109	420	170	386	76	442	128	400
112	394	152	301	95	426	144	403
104	407	153	397	87	402	163	370
111	422	152	301	70	397	145	355
126	423	136	438	84	511	133	375
105	434	122	306	91	469	128	383
119	474	148	383	74	451	123	349
114	396	90	385	101	474	144	373
100	470	145	337	80	398	140	388
84	399	123	364				

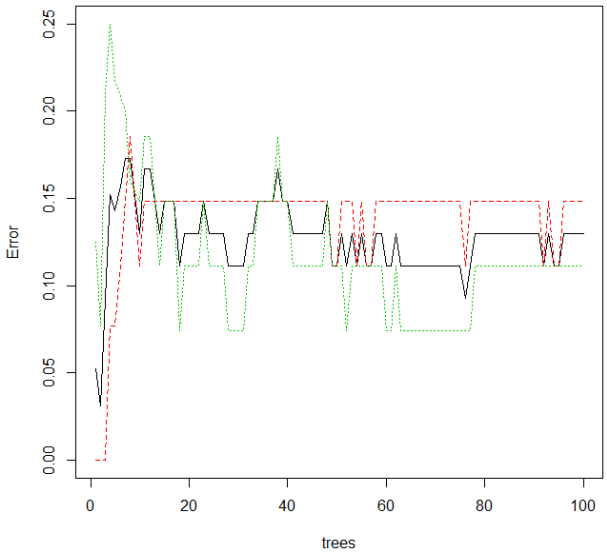
```
> library(randomForest)
> library(gmodels) # for CrossTable
>
> s.data<-read.table('salmon.txt',
+                    skip=4, header=T)
> head(s.data)
    Area  X1  X2
1 Alaska 108 368
2 Alaska 131 355
3 Alaska 105 469
4 Alaska  86 506
5 Alaska  99 402
6 Alaska  87 423
```

```
> m1<-sample(1:45, 27)  # Alaska 훈련자료 번호
> m2<-sample(46:90, 27)  # Canada 훈련자료 번호
> # 훈련데이터 60%
> tr.data<-s.data[sort(c(m1,m2)),]
> # 검정데이터 40%
> ts.data<-s.data[-sort(c(m1,m2)),]
> # 랜덤포리스트
> rf<-randomForest(Area~., data=tr.data,
+                  ntree=100, mtry=2, importance=TRUE)
> rf  # 랜덤포리스트 결과
```

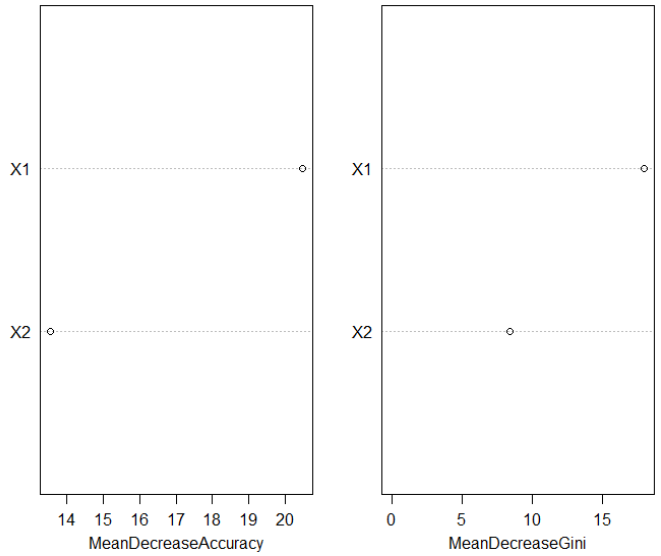
```
Call:
 randomForest(formula = Area ~ ., data = tr.data,
ntree = 100,       mtry = 2, importance = TRUE)
               Type of random forest: classification
                     Number of trees: 100
No. of variables tried at each split: 2

         OOB estimate of  error rate: 12.96%
Confusion matrix:
           Alaska Canada class.error
Alaska      23          4     0.1481481
Canada      3         24     0.1111111
```

```
> head(rf$err.rate)   # 에러율
            OOB      Alaska      Canada
[1,] 0.05263158 0.00000000 0.12500000
[2,] 0.03125000 0.00000000 0.07692308
[3,] 0.09302326 0.00000000 0.21052632
[4,] 0.15217391 0.07692308 0.25000000
[5,] 0.14285714 0.07692308 0.21739130
[6,] 0.15686275 0.11111111 0.20833333
> plot(rf)
> round(importance(rf),3) # 변수의 중요도
    Alaska  Canada MeanDecreaseAccuracy MeanDecreaseGini
X1 19.864 12.114              20.493            18.008
X2 12.214  7.818              13.551             8.434
> varImpPlot(rf)
```



검은색 : 전체 오류율
빨강색 : Alaska 오류율
녹색 : Canada 오류율



노드 불순도 개선에 중요한 변수
X1>X2 순서
정확도 개선에 중요한 변수
X1>X2 순서

```
> # 훈련데이터의 분류 예측
> tr.pred<-predict(rf) # 예측
> # 분류표
> CrossTable(tr.pred, tr.data$Area,
+                    prop.chisq=FALSE, prop.t=FALSE,
+                    prop.r=FALSE,
+                    dnn=c('predicted', 'actual'))
> tr.acc<-mean(tr.pred==tr.data$Area) # 분류율
> tr.acc
[1] 0.8703704
```

```
Cell Contents
|-------------------------|
|                       N |
|           N / Col Total |
|-------------------------|

Total Observations in Table:  54

             | actual
   predicted |    Alaska |    Canada | Row Total |
-------------|-----------|-----------|-----------|
      Alaska |        23 |         3 |        26 |
             |     0.852 |     0.111 |           |
-------------|-----------|-----------|-----------|
      Canada |         4 |        24 |        28 |
             |     0.148 |     0.889 |           |
-------------|-----------|-----------|-----------|
Column Total |        27 |        27 |        54 |
             |     0.500 |     0.500 |           |
-------------|-----------|-----------|-----------|
```

```
> # 검정 데이터의 분류 예측
> ts.pred<-predict(rf, newdata=ts.data) # 예측
> # 분류표
> CrossTable(ts.pred, ts.data$Area,
+                  prop.chisq=FALSE, prop.t=FALSE,
+                  prop.r=FALSE,
+                  dnn=c('predicted', 'actual'))
> ts.acc<-mean(ts.pred==ts.data$Area) # 분류율
> ts.acc
[1] 0.8888889
>
> detach("package:gmodels", unload=TRUE)
> detach("package:randomForest", unload=TRUE)
```

```
 Cell Contents
|-------------------------|
|                       N |
|           N / Col Total |
|-------------------------|

Total Observations in Table:  36

             | actual
   predicted |    Alaska |    Canada | Row Total |
-------------|-----------|-----------|-----------|
      Alaska |        17 |         3 |        20 |
             |     0.944 |     0.167 |           |
-------------|-----------|-----------|-----------|
      Canada |         1 |        15 |        16 |
             |     0.056 |     0.833 |           |
-------------|-----------|-----------|-----------|
Column Total |        18 |        18 |        36 |
             |     0.500 |     0.500 |           |
-------------|-----------|-----------|-----------|
```