의사결정나무 - decision tree

Jinseog Kim
Dongguk University
jskim1986@gmail.com

2018-04-18

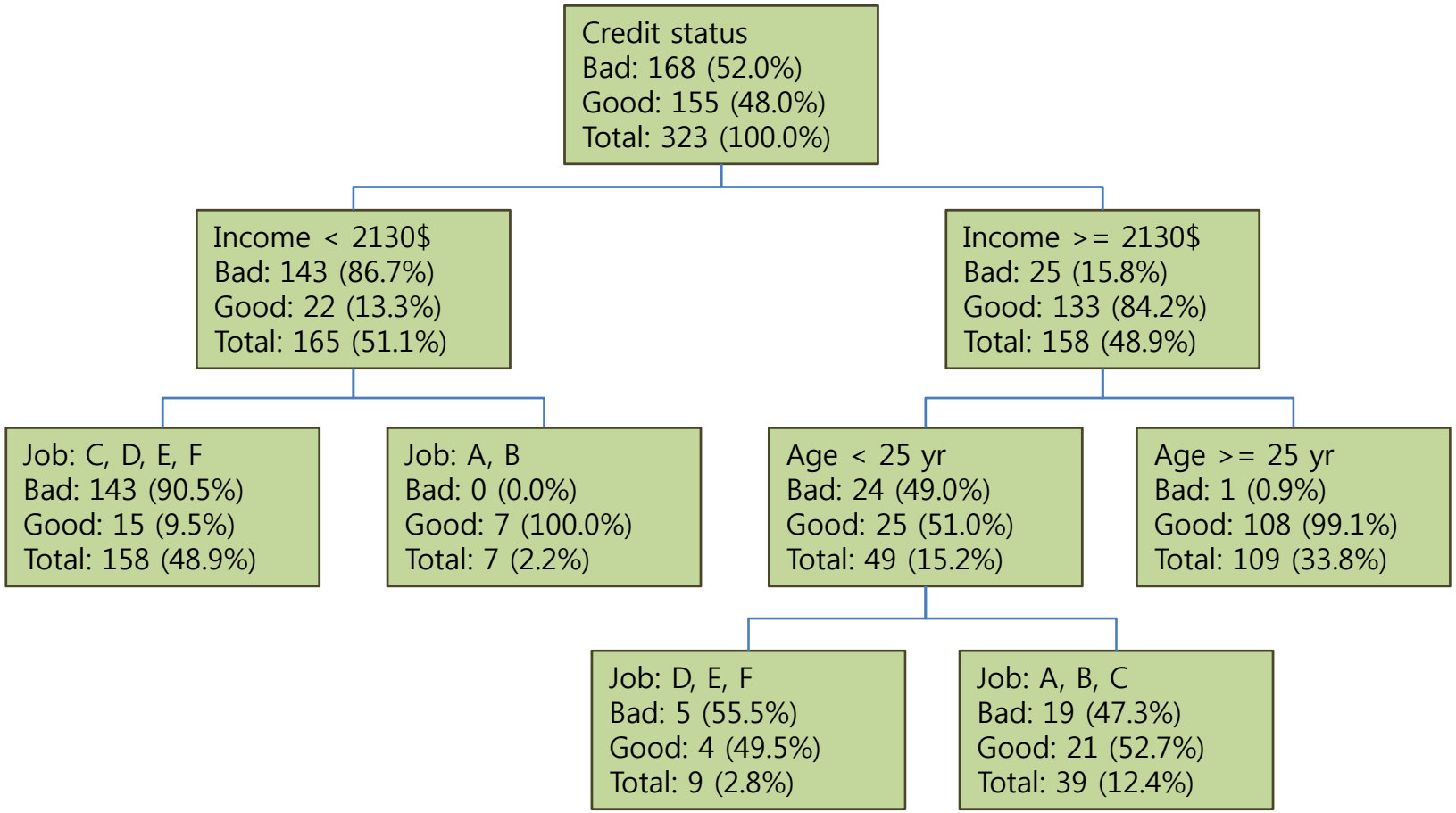
Contents

1 의사결정나무

1.1 의사결정나무란?

- 의사결정나무(decision trees)는 주어진 입력값에 대하여 출력값을 예측하는 모형
- 나무형태의 그래프로 표현
- 예측력은 다른 지도학습 기법들에 비해 대체로 떨어지나 해석력이 좋음
- 분류나무(classification trees)와 회귀나무(regression trees)

1.2 의사결정나무의 예시



1.3 의사결정나무모형에 사용되는 용어

- 뿌리마디(root node): 시작되는 마디로 전체 자료를 포함
- 자식마디(child node): 하나의 마디로부터 분리되어 나간 2개 이상의 마디들
- 부모마디(parent node): 주어진 마디의 상위마디
- 끝마디(terminal node): 자식마디가 없는 마디
- 중간마디(internal node): 부모마디와 자식마디가 모두 있는 마디
- 가지(branch): 뿌리마디로부터 끝마디까지 연결된 마디들
- 깊이(depth): 뿌리마디부터 끝마디까지의 중간마디들의 수

1.4 의사결정나무모형의 절차

1. 성장: 최대 크기의 나무 모형 형성
2. 가지치기: 최대 크기 나무모형에서 불필요한 가지를 제거하여 부분 나무모형(subtrees)의 집합을 탐색
3. 타당성 평가: 가지치기의 결과인 나무모형의 집합에서 최적 모형을 선택
4. 예측

1.5 의사결정나무모형의 종류

1. 회귀나무 (regression tree): 출력변수가 연속형
2. 분류나무 (classification tree): 출력변수가 범주형

2 나무모형의 성장

 의사결정나무의 형성과정은 크게 아래의 네 단계로 이루어진다.

1. 성장(growing): 각 마디에서 적절한 최적의 분리규칙을 찾아서 나무를 성장시키는 과정으로서 적절한 정지규칙을 만족하면 중단
2. 가지치기(pruning): 오차를 크게 할 위험이 높거나 부적절한 추론규칙을 가지고 있는 가지 또는 불필요한 가지를 제거
3. 타당성 평가: 이익도표(gain chart), 위험도표(risk chart), 혹은 시험자료를 이용하여 의사결정나무를 평가
4. 해석 및 예측: 구축된 나무모형을 해석하고 예측모형을 설정한 후 예측에 적용

2.1 데이터와 모형

 훈련자료: $(x_i, y_i), i = 1, \ldots, n, x_i = (x_{i1}, \ldots, x_{ip})^T$
 전체 입력 공간의 분할: M개의 영역 $R_1, \ldots, R_M$
 출력값(y) 적합

 연속형인 경우 : 분할된 영역(R_m)별로 상수(c_m)로 적합 (예측)

$$f(x) = \sum_{m=1}^{M} c_m I(x \in R_m).$$

 범주형은 : 가장 많은 범주값(최빈값)으로 적합

2.2 성장 절차

- 입력 공간의 분할 - 최적 분리 기준의 탐색
  - R_m, c_m의 결정: 분리변수(split variable)와 분리값(split value)의 결정

- 불순도(impurity): 오차제곱합

$$Q_m(T) = \sum_{i=1}^{n} (y_i - f(x_i))^2$$

  - 분리변수(split variable) x_j가 연속형
    - $R_1(j, s) = \{x : x_j \leq s\}$와 $R_2(j, s) = \{x : x_j > s\}$를 정의
    - s: 분리점, x_j: 분리변수
  - 분리변수가 범주형
    - 전체 범주값을 두 개의 부분집합으로 나눔
    - 전체 범주 = $\{1, 2, 3, 4\}$
      - $R_1(j) = \{1, 2, 4\}$와 $R_2(j) = \{3\}$

- 분리 기준

$$\min_{j,s} \left(\min_{c_1} \sum_{x_i \in R_1(j,s)} (y_i - c_1)^2 + \min_{c_2} \sum_{x_i \in R_2(j,s)} (y_i - c_2)^2 \right).$$

1. 지니(Gini)지수

$$2\,(p(\text{Left에서 Good})p(\text{Left에서 Bad})p(\text{Left}) \\ +p(\text{Right에서 Good})p(\text{Right에서 Bad})p(\text{Right}))$$

2. 엔트로피(entropy)지수

엔트로피지수 = 엔트로피(Left)p(Left) + 엔트로피(Right)p(Right)

여기서

$$\begin{aligned} \text{엔트로피(Left)} \;=\; & -p(\text{Left에서 Good})\log_2 p(\text{Left에서 Good}) \\ & -p(\text{Left에서 Bad})\log_2 p(\text{Left에서 Bad}) \end{aligned}$$

2.3 분류나무의 성장 예제

 자료가 아래의 표와 같이 분리된다고 하자.

	Good	Bad	Total
Left	32 (56)	48 (24)	80
Right	178 (154)	42 (66)	220
Total	210	90	300

1. 지니(Gini)지수에 의한 성장

$$2\,(p(\text{Left에서 Good})p(\text{Left에서 Bad})p(\text{Left}) \\ +p(\text{Right에서 Good})p(\text{Right에서 Bad})p(\text{Right}))$$

$$2\left(\frac{32}{80}\times\frac{48}{80}\times\frac{80}{300}+\frac{178}{220}\times\frac{42}{220}\times\frac{220}{300}\right)=0.355$$

2. 엔트로피(entropy)지수

엔트로피지수 = 엔트로피(Left)p(Left) + 엔트로피(Right)p(Right)

여기서

$$\begin{aligned}\text{엔트로피(Left)} &= -p(\text{Left에서 Good})\log_2 p(\text{Left에서 Good}) \\ &\quad -p(\text{Left에서 Bad})\log_2 p(\text{Left에서 Bad})\end{aligned}$$

$$-\left(\frac{32}{80}\log_2\left(\frac{32}{80}\right)+\frac{48}{80}\log_2\left(\frac{48}{80}\right)\right)\frac{80}{300}$$
$$-\left(\frac{178}{220}\log_2\left(\frac{178}{220}\right)+\frac{42}{220}\log_2\left(\frac{42}{220}\right)\right)\frac{220}{300}=.7747$$

3 가지치기: 비용-복잡도 가지치기(cost-complexity pruning)

- why?
  - 너무 큰 나무모형은 자료를 과대적합
  - 너무 작은 나무모형은 자료를 과소적합할 위험
  - 최적의 나무 모형의 크기를 결정해야 함

- 방법 및 절차
  - T_0: 최대로 성장시킨 나무모형
  - $T_0 > T_1 > T_2 > \ldots > T_{root}$: 가지치기하여 얻을 수 있는 나무모형
  - $|T|$: 나무모형 T의 끝마디(terminal node) 개수
  - N_m: T의 영역 R_m에 속하는 자료수
  - $\hat{c}_m$은 영역 R_m에 속하는 자료 y값들의 평균
  - 불순도는 $Q_m(T) = \frac{1}{N_m} \Sigma_{x_i \in R_m} (y_i - \hat{c}_m)^2$
  - 비용-복잡도 가지치기(cost-complexity pruning)

$$C_\alpha(T) = \sum_{m=1}^{|T|} N_m Q_m(T) + \alpha |T|$$

4 의사결정나무의 특징

- 장점
  - 의사결정나무(특히 CART)는 if-then 형식의 이해하기 쉬운 규칙
  - 연속형 변수와 범주형 변수를 모두 취급할 수 있음
  - 모형에 대한 가정(예: 선형회귀의 선형성, 등분산성 등)이 필요 없는 비모수적 방법
  - 가장 설명력이 있는 변수에 대하여 최초로 분리가 일어남
- 의사결정나무의 단점
  - 출력변수가 연속형인 회귀모형에서는 그 예측력 저하
  - 일반적으로 복잡한 나무모형은 예측력이 저하되고 해석 또한 어려움
  - 상황에 따라 계산량이 많을 수도 있음
  - 베이즈 분류경계가 사각형(rectangle)이 아닌 경우에는 좋지 않은 결과
  - 자료에 약간의 변화가 있는 경우에 전혀 다른 결과를 줄 수도 있는 (즉, 분산이 매우 큰) 불안정한 방법
    - cf: 배깅(bagging)과 같은 앙상블(ensemble) 알고리즘

5 모형의 평가

5.1 검증오차

1. 자료(데이터)를 훈련자료($\mathcal{L}$)와 검증자료($\mathcal{T}$)로 분할
2. 훈련자료를 이용 예측모형 $\hat{f}_k(), k = 1, ..., M$를 적합
3. 검증자료를 이용해 검증오차 계산

$$E_k = \sum_{(x_i, y_i) \in \mathcal{T}} l(\hat{f}_k(x_i), y_i)/m$$

4. 모든 모형들에 대해 검증오차를 계산하고 그 중 최소가 되는 모형을 선택

 검증오차법은 주어진 데이터의 갯수가 충분히 많은 경우에 사용

5.2 교차확인법

 교차확인법은 검증오차법의 일반화

1. 자료를 서로 배반(disjoint)이 되도록 무작위로 K개의 묶음(fold)들 $\mathcal{L}_1, \cdots, \mathcal{L}_K$로 분할
2. $k = 1, \ldots, K$에 대하여 $\cup_{l \neq k} \mathcal{L}_l$을 이용하여 $\hat{f}_k(x)$를 구하고
3. $\mathcal{L}_k$를 검증자료로 사용하여 검증오차를 구함

- K개의 검증오차의 산술평균을 K-묶음 교차확인오차(k-fold cross validation)

4. 각 모형들에 대한 교차확인오차를 계산하고 그 값이 최소가 되는 모형을 선택

- 1-표준편차 규칙(1-standard deviation rule): 교차확인오차의 분산이 큰 경우, 최소값을 가지는 모형 근방(흔히 표준편차의 범위내에서)에서 가장 단순한 모형을 찾는 방법

5.3 모형 평가

- confusion matrix (오분류표)

		Predicted($\hat{y}$)	
		true($\hat{y}=1$)	false($\hat{y}=0$)
Actural(y)	positive (1)	True Positive (TP)	False Negative (FN) (type II error)
	negative (0)	False Positive (FP) (Type I error)	True Negative (TN)

- True positive rate (Recall:재현율, Sensitivity: 민감도): $P(\hat{Y}=+|Y=+)$
- Error rate(오분류율): $P(\hat{Y}!=Y)$
- Precision (정확률): $P(Y=+|\hat{Y}=+)$
- Specificity (특이도) : $P(\hat{Y}=-|Y=-)$

6 R을 이용한 의사결정나무모형의 적합

6.1 R 패키지 및 함수

1. R packages: rpart, rpart.plot, tree, party, C5.0, …
2. R Functions
    1. rpart(formula, data=, method=,control=)
        - formula: response ~ input1 + input2 + …
        - data: data frame
        - method: “class” - classification tree, “anova” for a regression tree
        - control: rpart.control(minsplit=30, cp=0.001)
    2. printcp(fit), plotcp(fit): print or plot cp-table(cost-complexity-parameters) of fit
    3. prune(fit, cp) : pruning with cp value
    4. summary(fit): summary of fit model
    5. predict(fit, newdata, type): predict of type from fit model given newdata

7 의사결정나무모형 예제

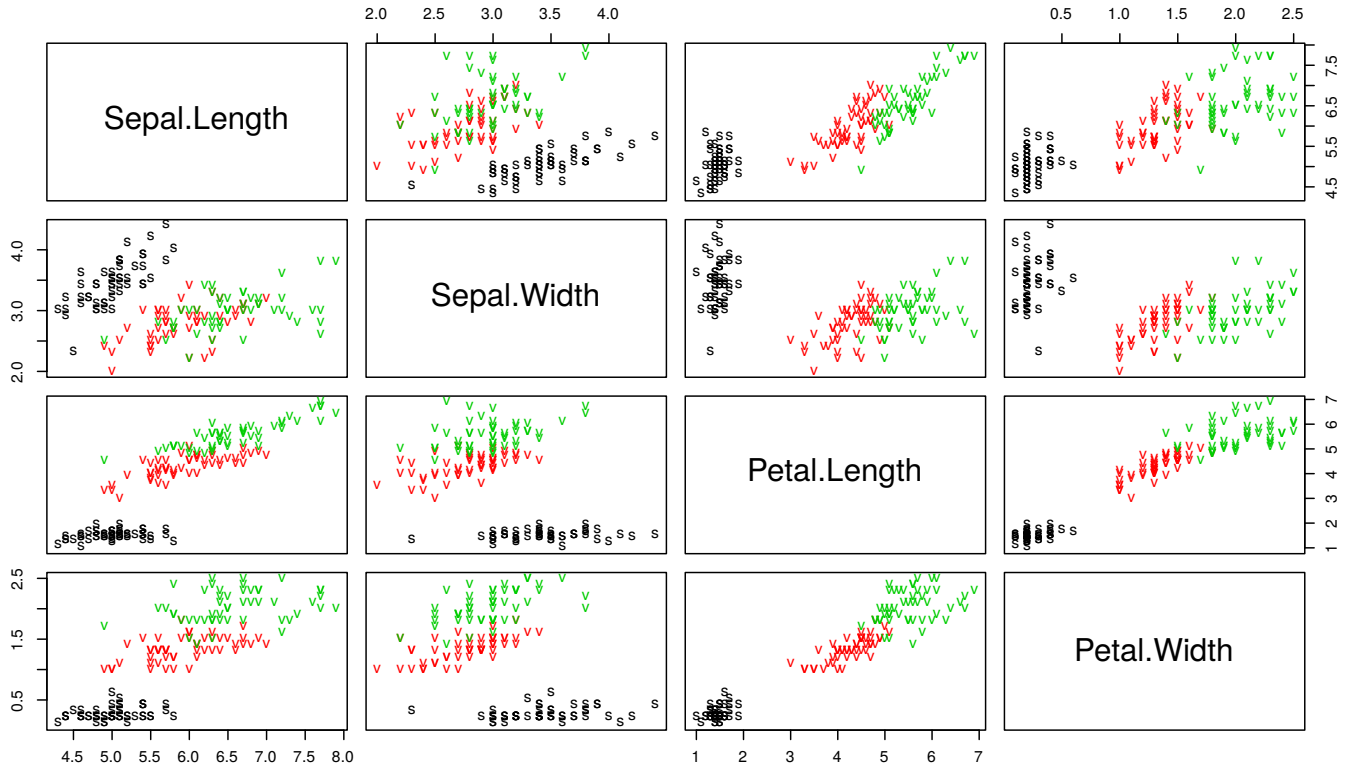
7.1 의사결정나무모형을 이용한 붓꽃자료의 분석

- 붓꽃자료

Variables	Description	변수역할
Sepal.Length	꽃받침의 길이	input
Sepal.Width	꽃받침의 폭	input
Petal.Length	꽃잎의 길이	input
Petal.Width	꽃잎의 폭	input
Species	붓꽃의 품종(setosa, versicolor, virginica)	target

- 자료의 탐색

```
plot(iris[,1:4], col=as.integer(iris$Species),
     pch=substring((iris$Species),1,1))
```



- tree 함수를 이용한 모형 성장

```
library(tree)
ir.tr <- tree(Species ~., iris)
ir.tr
```

- rpart 함수를 이용한 모형 성장

```r
library(rpart)
ir.tr2 <- rpart(Species ~., iris)
ir.tr2
```

```
n= 150

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

1) root 150 100 setosa (0.33333333 0.33333333 0.33333333)
  2) Petal.Length< 2.45 50   0 setosa (1.00000000 0.00000000 0.00000000) *
  3) Petal.Length>=2.45 100  50 versicolor (0.00000000 0.50000000 0.50000000)
    6) Petal.Width< 1.75 54   5 versicolor (0.00000000 0.90740741 0.09259259) *
    7) Petal.Width>=1.75 46   1 virginica (0.00000000 0.02173913 0.97826087) *
```

7.2 의사결정나무모형을 이용한 UniversalBank자료의 분석

 UniversalBank 자료의 변수 속성

Variables	Description
Age	Customer's age in completed years
Experience	years of professional experience
Income	Annual income of the customer ($000)
Family	Family size of the customer
CCAvg	Avg. spending on credit cards per month ($000)
Education	Education Level. 1: Undergrad; 2: Graduate; 3: Advanced/Professional
Mortgage	Value of house mortgage if any. ($000)
PersonalLoan	Did this customer accept the personal loan offered in the last campaign? target variable
SecuritiesAccount	Does the customer have a securities account with the bank?
CDAccount	Does the customer have a certificate of deposit(CD) account with the bank?
Online	Does the customer use internet-banking facilities?
CreditCard	Does the customer use a credit card issued by Universal Bank?

 자료불러오기

```
load(file="data/loan.RData")
dim(loan)
```

```
[1] 5000   12
```

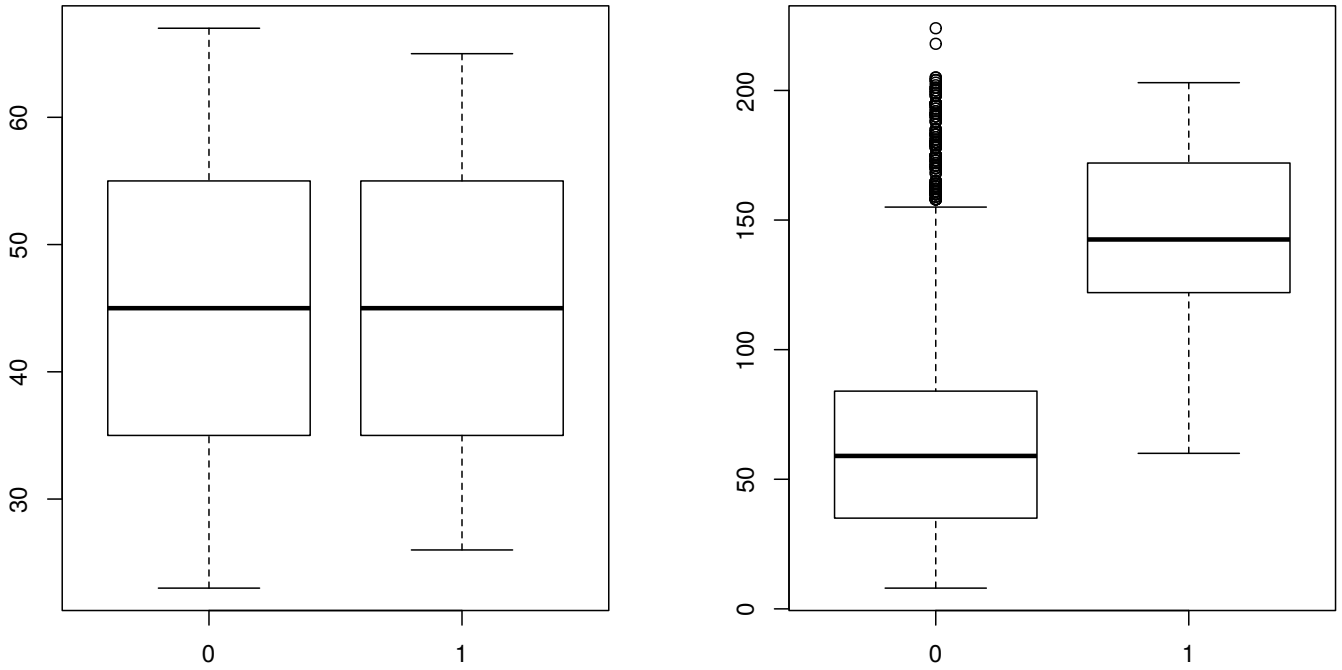
Age	Experience	Income	Family	CCAvg	Education
25	1	49	4	1.6	1
45	19	34	3	1.5	1
39	15	11	1	1.0	1
35	9	100	1	2.7	2
35	8	45	4	1.0	2
37	13	29	4	0.4	2

Mortgage	PersonalLoan	SecuritiesAccount	CDAccount	Online	CreditCard
0	0	1	0	0	0
0	0	1	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0

Mortgage	PersonalLoan	SecuritiesAccount	CDAccount	Online	CreditCard
0	0	0	0	0	1
155	0	0	0	1	0

탐색적자료분석 (EDA)

```
par(mfrow=c(1,2))
boxplot(Age~PersonalLoan, data=loan)
boxplot(Income~PersonalLoan, data=loan)
```



- Age는 PersonalLoan과 관련이 없어 보임
- Income이 높을수록 대출을 받을(PersonalLoan=1) 가능성이 높게 나타남

```r
x1 <- with(loan, table(CreditCard, PersonalLoan))
x2 <- with(loan, table(Education, PersonalLoan))
chisq.test(x1); chisq.test(x2)
```

```
    Pearson's Chi-squared test with Yates' continuity correction

data:  x1
X-squared = 0.021144, df = 1, p-value = 0.8844
```

```
    Pearson's Chi-squared test

data:  x2
X-squared = 111.24, df = 2, p-value < 2.2e-16
```

- CreditCard보유여부는 PersonalLoan과 관련이 없어 보임
- Education과 대출여부는(PersonalLoan) 연관성이 있음

 데이터의 분할

```
# 인덱스만 분할 -  훈련자료와 검증자료를 각 2500개씩
tr.idx <- sample(nrow(loan), 2500, replace=F)
```

 분류나무모형의 성장

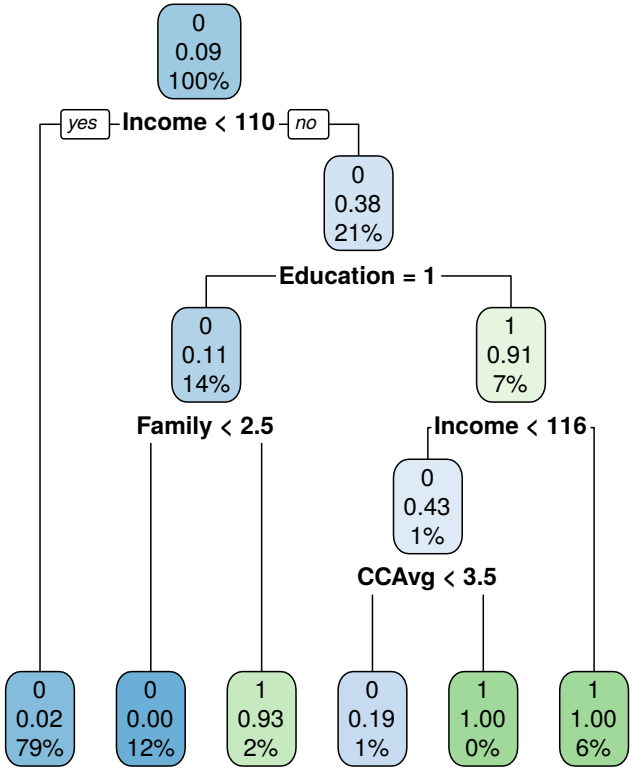
```
library(rpart)
ctfit <- rpart(PersonalLoan~.-CDAccount, data=loan, subset=tr.idx,
               method="class",#"anova" for regression tree
               control=rpart.control(minsplit = 20))
ctfit
```

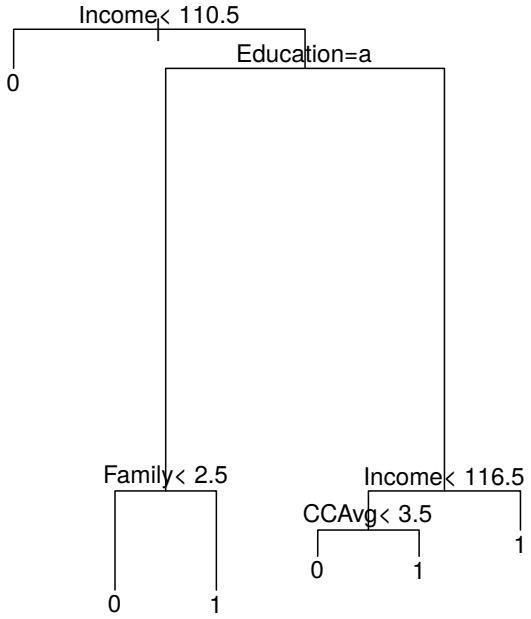
```
n= 2500

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

 1) root 2500 233 0 (0.90680000 0.09320000)
   2) Income< 110.5 1981  34 0 (0.98283695 0.01716305) *
   3) Income>=110.5 519 199 0 (0.61657033 0.38342967)
     6) Education=1 340  37 0 (0.89117647 0.10882353)
```

```
 12) Family< 2.5 300   0 0 (1.00000000 0.00000000) *
 13) Family>=2.5 40   3 1 (0.07500000 0.92500000) *
7) Education=2,3 179  17 1 (0.09497207 0.90502793)
 14) Income< 116.5 30  13 0 (0.56666667 0.43333333)
   28) CCAvg< 3.5 21   4 0 (0.80952381 0.19047619) *
   29) CCAvg>=3.5 9   0 1 (0.00000000 1.00000000) *
 15) Income>=116.5 149   0 1 (0.00000000 1.00000000) *
```

 rpart.plot를 이용한 모형 그림

```
library(rpart.plot); par(mfrow=c(1,2))
rpart.plot(ctfit); plot(ctfit); text(ctfit)
```





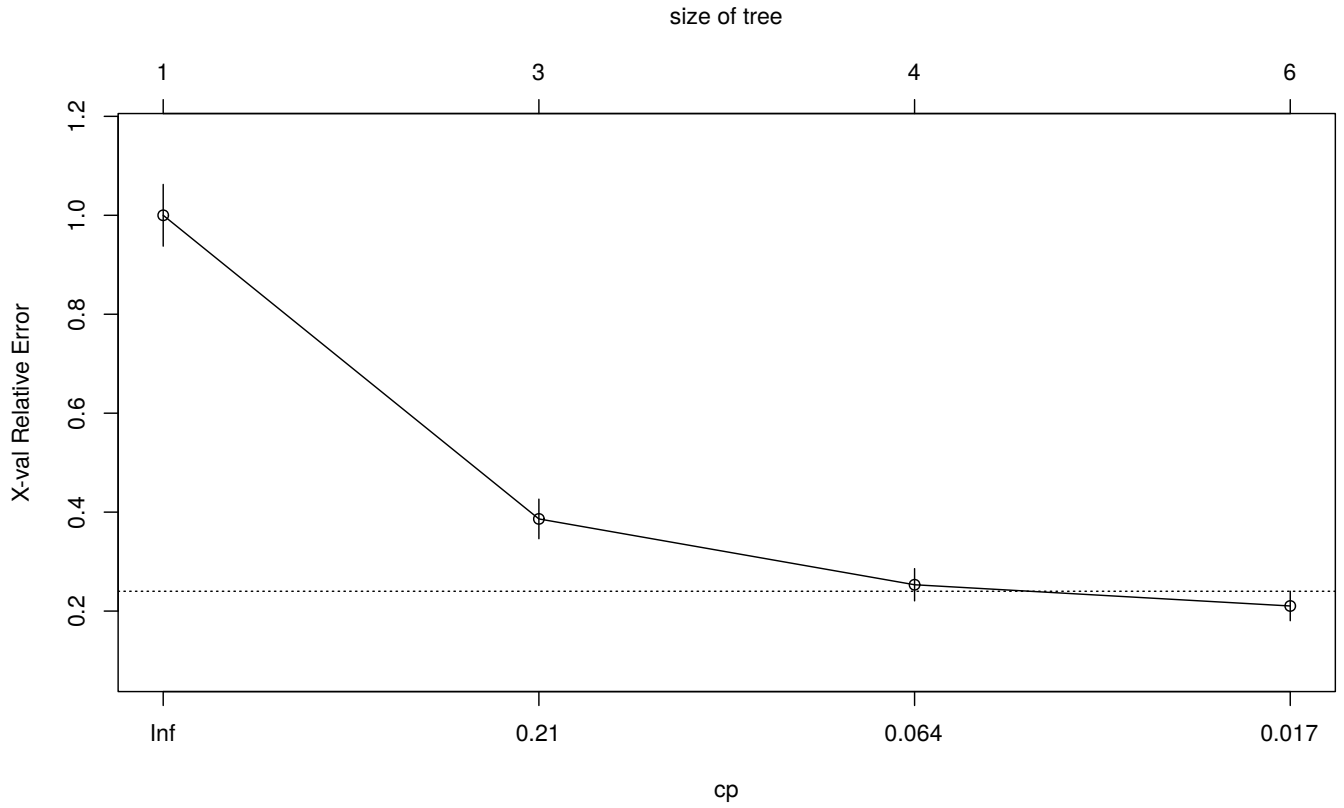
 가지치기 (pruning) 및 최적 분류나무모형의 선택

```
printcp(ctfit) # display the results
```

```
Classification tree:
rpart(formula = PersonalLoan ~ . - CDAccount, data = loan, subset = tr.idx,
    method = "class", control = rpart.control(minsplit = 20))

Variables actually used in tree construction:
[1] CCAvg     Education Family    Income

Root node error: 233/2500 = 0.0932

n= 2500

        CP nsplit rel error  xerror     xstd
1 0.311159      0   1.00000 1.00000 0.062385
2 0.145923      2   0.37768 0.38627 0.039976
3 0.027897      3   0.23176 0.25322 0.032575
4 0.010000      5   0.17597 0.21030 0.029747
```

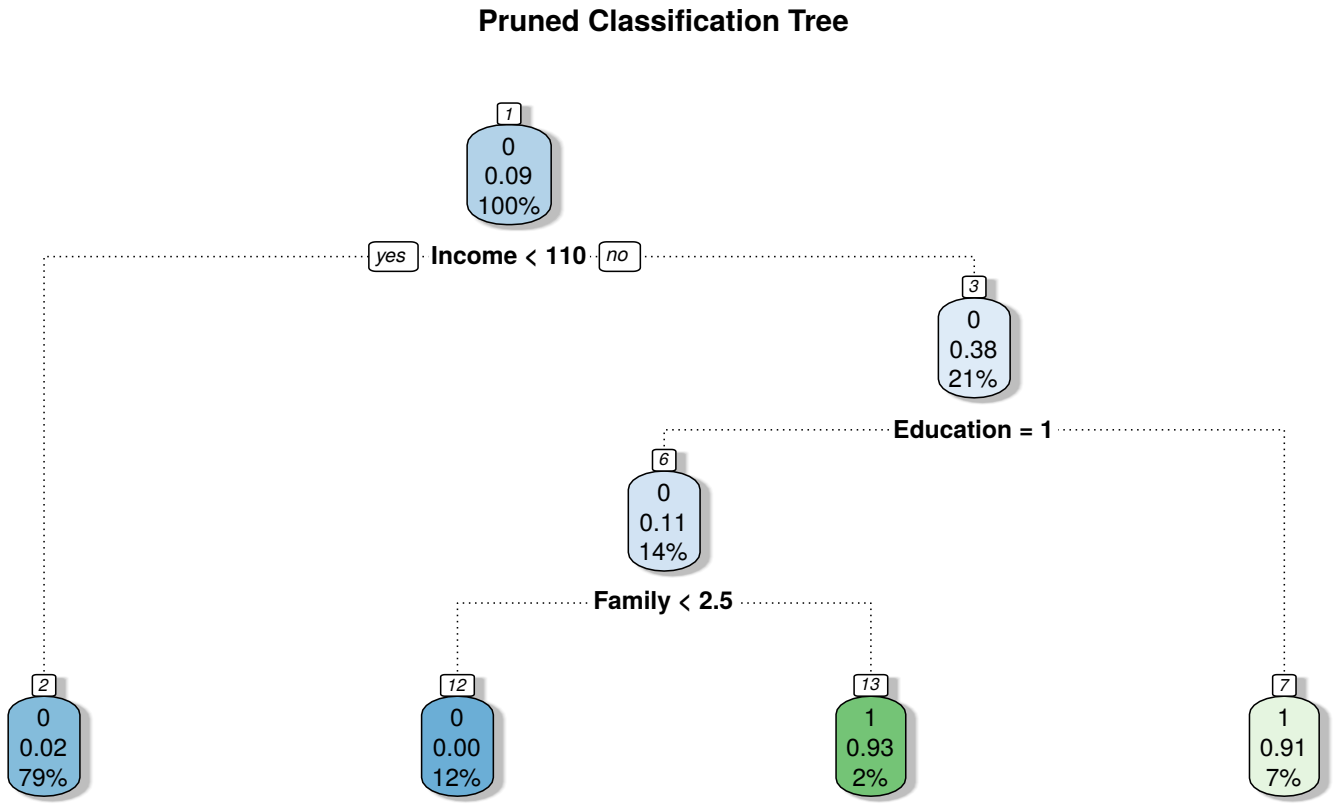
 cptable의 이웃한 두개의 CP값의 기하평균 vs CV-error

```r
plotcp(ctfit) # visualize cross-validation results
```




```r
pctfit <- prune(ctfit, cp=0.064)
# optimal cp value = 0.064
```

```
rpart.plot(pctfit, main="Pruned Classification Tree", branch.lty=3, shadow.col="gray",
           nn=TRUE)
```



 검증데이터를 이용 모형의 평가 및 예측

```
pred <- predict(pctfit, newdata=loan[-tr.idx,], type="prob")
(t1 <- table(y=loan$PersonalLoan[-tr.idx], pred=pred[,2]>0.5))
```

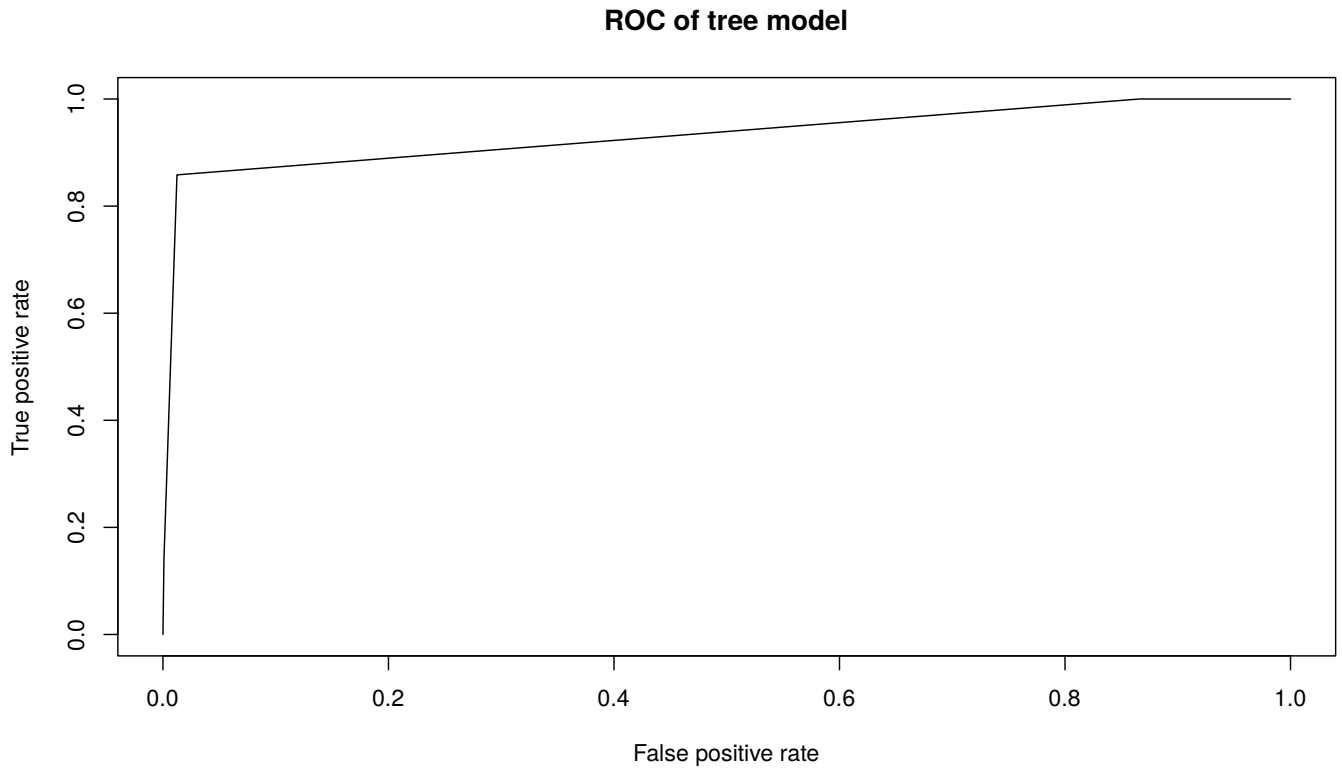
```
   pred
y    FALSE TRUE
  0  2225   28
  1    35  212
```

```
(err <- 1- sum(diag(t1))/sum(t1))
```

```
[1] 0.0252
```

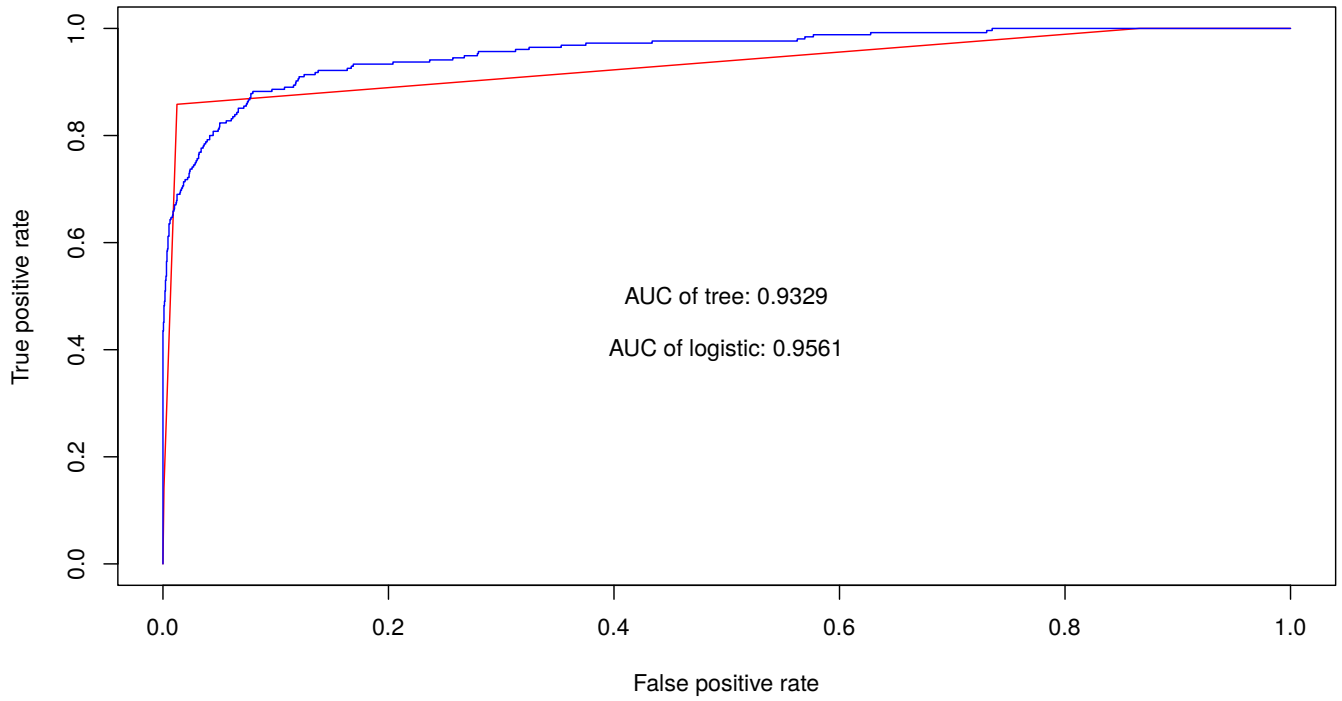
 Plot ROC

```
library(ROCR)
p <- prediction(pred[,2], loan$PersonalLoan[-tr.idx])
perf <- performance(p, measure = "tpr", x.measure = "fpr")
plot(perf, main="ROC of tree model")
```



- Plot ROC curve and add AUC (Area Under the Curve)

```
plot(perf, col="red")
plot(perf.log, add = TRUE, col="blue")
auc = as.numeric(performance(p, "auc")@y.values)
text(0.5, 0.5, paste("AUC of tree:",round(auc,4)))
text(0.5, 0.4, paste("AUC of logistic:",round(auc.log,4)))
```


True positive rate
0.0
0.2
0.4
0.6
0.8
1.0
AUC of tree: 0.9329
AUC of logistic: 0.9561
False positive rate

7.3 Automobile Data의 분석

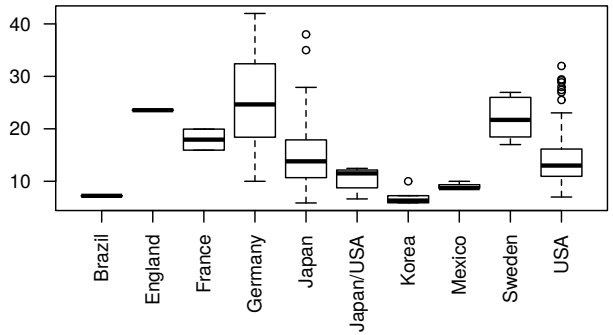
- cu.summary: from ‘Consumer Reports’ 1990
- 변수설명

Variables	Descriptions	type
Price	list price in US dollars of a standard model	target
Country	of origin, a factor with levels (Brazil, England, France, Germany, Japan, Japan/USA, Korea, Mexico, Sweden and USA)	
Reliability	an ordered factor with levels (Much worse $<$ worse $<$ average $<$ better $<$ Much better)	
Mileage	fuel consumption miles per US gallon, as tested.	
Type	a factor with levels (Compact, Large, Medium, Small, Sporty, Van)	

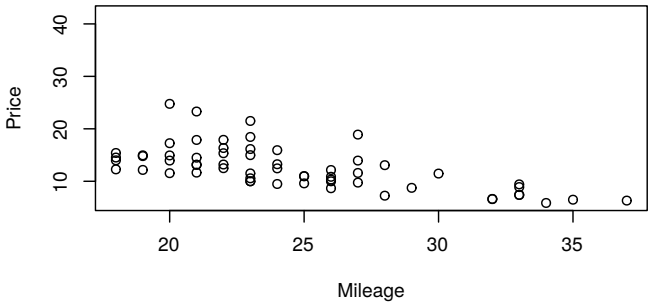
 탐색적 자료분석

```
cu.summary$Price <- cu.summary$Price/1000 # 단위조정
par(mfrow=c(2,2))
boxplot(Price~Country, data=cu.summary, las=2, main="Country vs. Price")
plot(Price~Mileage, data=cu.summary, main="Mileage vs. Price")
boxplot(Price~Reliability, data=cu.summary, las=2, main="Reliability vs. Price")
boxplot(Price~Type, data=cu.summary, las=2, main="Type vs. Price")
```

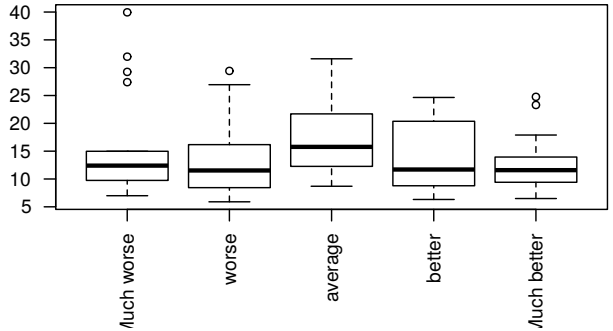
Country vs. Price



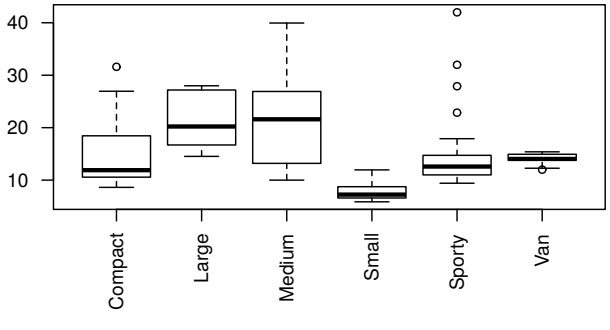
Mileage vs. Price



Reliability vs. Price



Type vs. Price



 회귀 나무모형의 성장

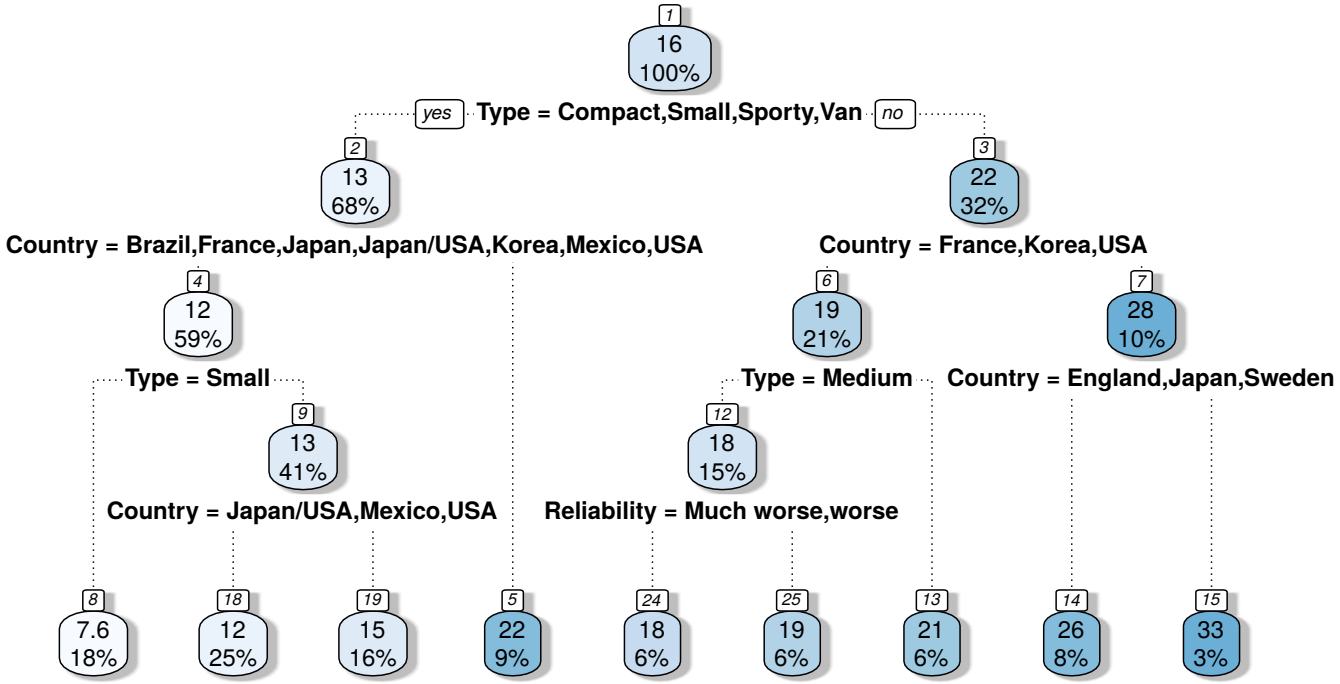
```
rtfit <- rpart(Price ~ Mileage + Country + Reliability + Type,
               method="anova", data=cu.summary,
               control=rpart.control(minsplit=10))
rtfit # detailed summary of splits
```

```
n= 117

node), split, n, deviance, yval
      * denotes terminal node

 1) root 117 7407.47300 15.743460
   2) Type=Compact,Small,Sporty,Van 80 3322.38900 13.035010
     4) Country=Brazil,France,Japan,Japan/USA,Korea,Mexico,USA 69 1426.42100 11.555160
       8) Type=Small 21   50.30983  7.629048 *
       9) Type=Compact,Sporty,Van 48  910.79000 13.272830
        18) Country=Japan/USA,Mexico,USA 29  482.34350 12.241550 *
        19) Country=France,Japan 19  350.52800 14.846890 *
     5) Country=Germany,Sweden 11  797.00420 22.317730 *
   3) Type=Large,Medium 37 2229.35100 21.599570
     6) Country=France,Korea,USA 25 1021.10200 18.697280
```

```
      12) Type=Medium 18  741.10160 17.607440
        24) Reliability=Much worse,worse 7  355.39950 18.183290 *
        25) Reliability=average,better,Much better 7  266.67240 19.010290 *
      13) Type=Large 7  203.64510 21.499710 *
    7) Country=England,Germany,Japan,Sweden 12  558.95500 27.646000
      14) Country=England,Japan,Sweden 9  343.62760 25.744670 *
      15) Country=Germany 3   85.18500 33.350000 *
```

```r
par(xpd = TRUE)
rpart.plot(rtfit, main="회귀나무모형", branch.lty=3, shadow.col="gray",
           nn=TRUE)
```

회귀나무모형




```
#plot(rtfit, compress = TRUE)
#text(rtfit, use.n = TRUE)
```

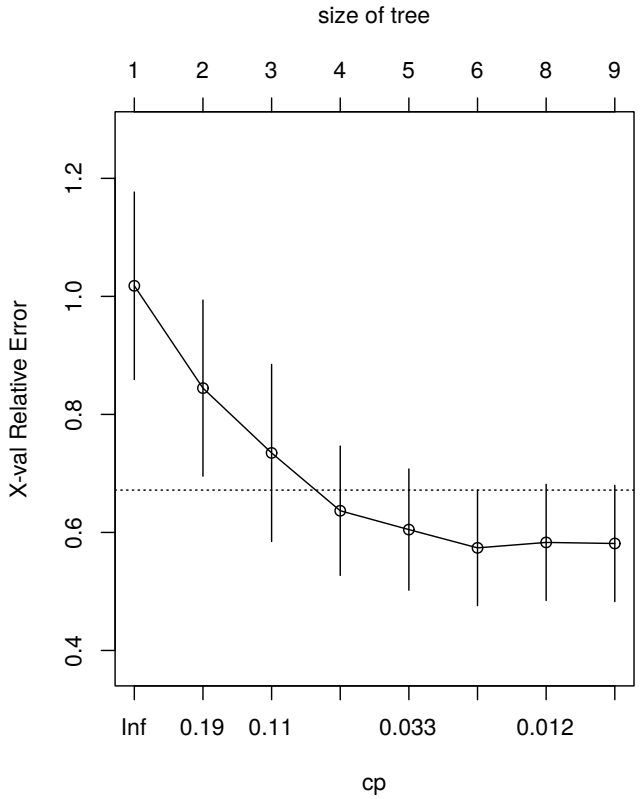
 가지치기를 통한 최종모형의 결정

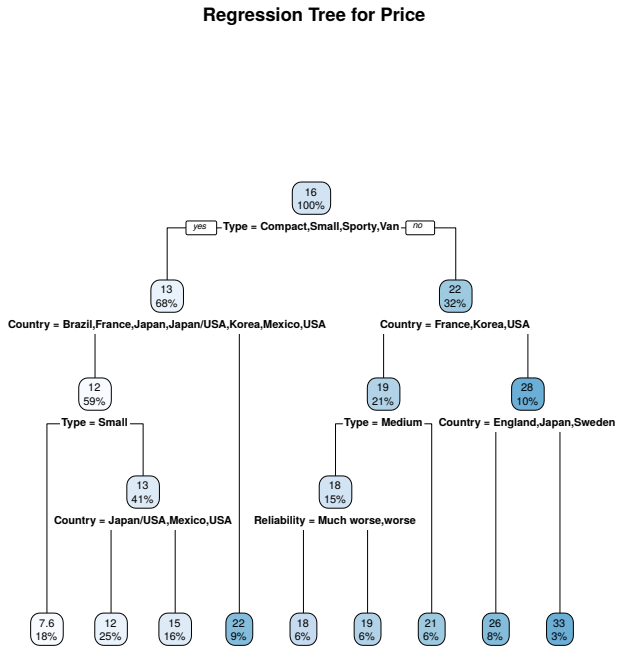
```
printcp(rtfit) # display the results
```

```
Regression tree:
rpart(formula = Price ~ Mileage + Country + Reliability + Type,
    data = cu.summary, method = "anova", control = rpart.control(minsplit = 10))

Variables actually used in tree construction:
[1] Country     Reliability Type

Root node error: 7407.5/117 = 63.312

n= 117

        CP nsplit rel error  xerror     xstd
1 0.250522      0   1.00000 1.01792 0.158890
2 0.148359      1   0.74948 0.84453 0.149189
3 0.087654      2   0.60112 0.73473 0.150014
4 0.062818      3   0.51347 0.63674 0.109571
5 0.017569      4   0.45065 0.60482 0.102699
```

```
6 0.013188      5   0.43308 0.57387 0.097879
7 0.010519      7   0.40670 0.58307 0.098481
8 0.010000      8   0.39618 0.58129 0.098495
```

```r
par(mfrow=c(1,2)) # two plots on one page
plotcp(rtfit)
rpart.plot(rtfit, main="Regression Tree for Price ")
```

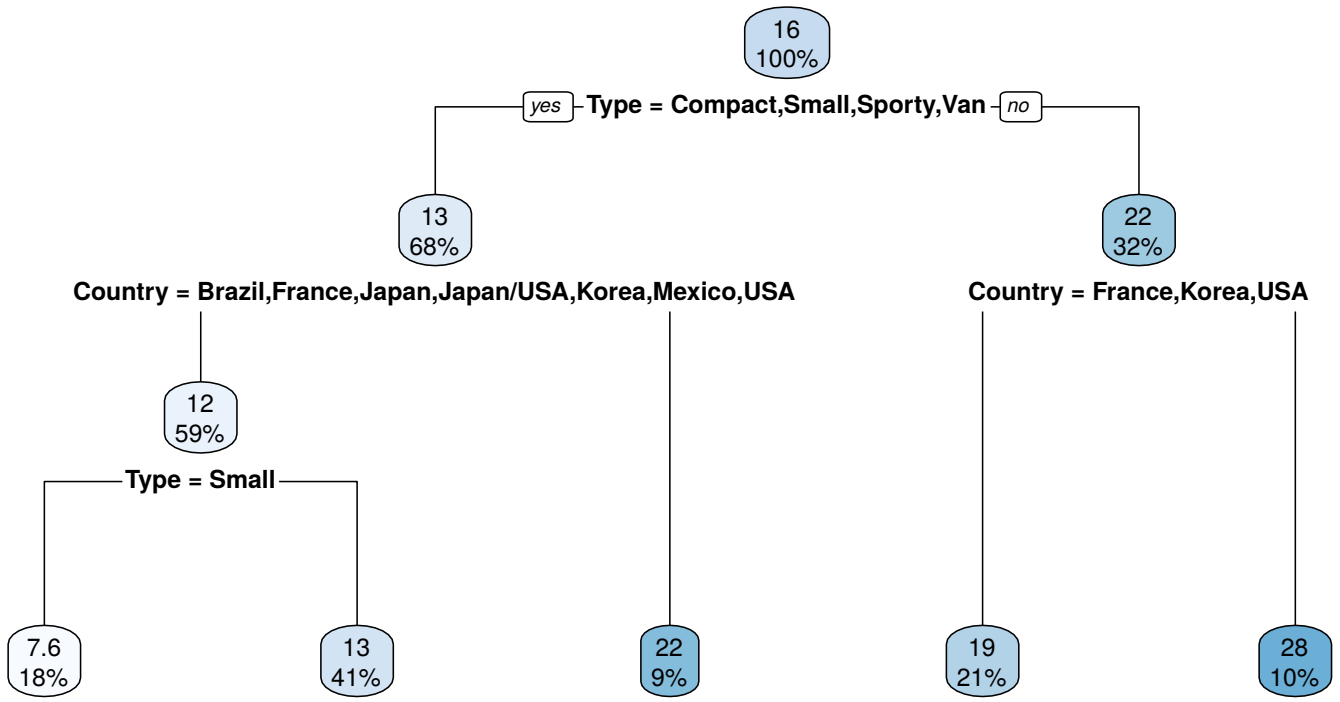




```
prtfit<- prune(rtfit, cp=0.033) # from cptable
prtfit
```

```
n= 117

node), split, n, deviance, yval
      * denotes terminal node

1) root 117 7407.47300 15.743460
  2) Type=Compact,Small,Sporty,Van 80 3322.38900 13.035010
    4) Country=Brazil,France,Japan,Japan/USA,Korea,Mexico,USA 69 1426.42100 11.555160
      8) Type=Small 21   50.30983  7.629048 *
      9) Type=Compact,Sporty,Van 48  910.79000 13.272830 *
    5) Country=Germany,Sweden 11  797.00420 22.317730 *
  3) Type=Large,Medium 37 2229.35100 21.599570
    6) Country=France,Korea,USA 25 1021.10200 18.697280 *
    7) Country=England,Germany,Japan,Sweden 12  558.95500 27.646000 *
```

```
rpart.plot(prtfit, uniform=TRUE,main="Pruned Regression Tree for Price")
```




```
#library(rpart.plot); par(mfrow=c(1,2))
#rpart.plot(prtfit);
```