Data Mining Practice - Decision Tree Part.2

1. German data

```
## url <-
## 'http://archive.ics.uci.edu/ml/machine-learning-databases/statlog/german/german.data'
## Data 가 있는 UCI URL 주소 download.file(url, destfile='./german.data') 현재
## working directory 로 데이터 download!
german <- read.table("./german.data")
# 내려받은 데이터 불러오기

names(german) <- c("check_a", "duration", "c_history", "purpose", "c_amount",
    "savings", "employment", "install_rate", "personal", "guarantee", "residence",
    "property", "age", "install_p", "housing", "n_exist", "job", "n_people",
    "tel", "foreign", "target")

##
german$target <- as.character(german$target)
german$target <- ifelse(german$target == 1, "good", "bad")
german$target <- as.factor(german$target)
```

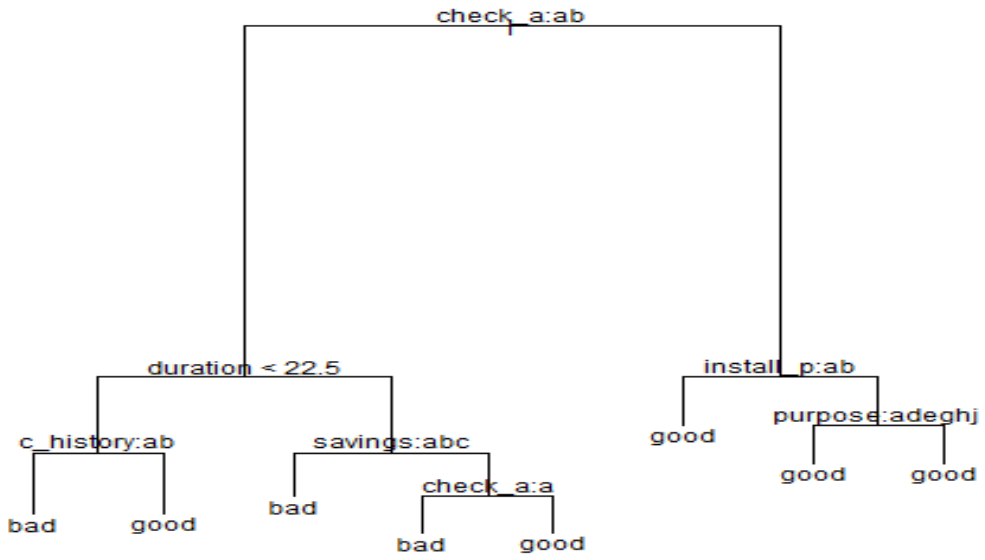
2. Growing Tree

```
library(tree)
library(rpart)
library(partykit)

## Loading required package: grid
```

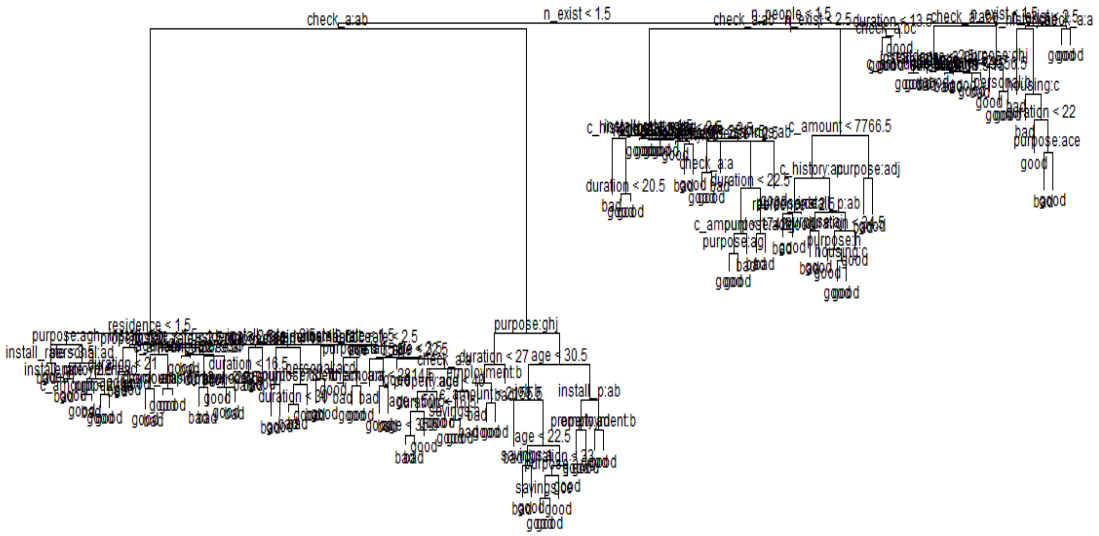
- **tree**
  - split : 사용할 분할 기준 (gini or deviance)
  - 나머지 argument 는 rpart 와 유사하다.

```
german.tr <- tree(target ~ ., data = german, method = "class")
plot(german.tr)
text(german.tr)
```



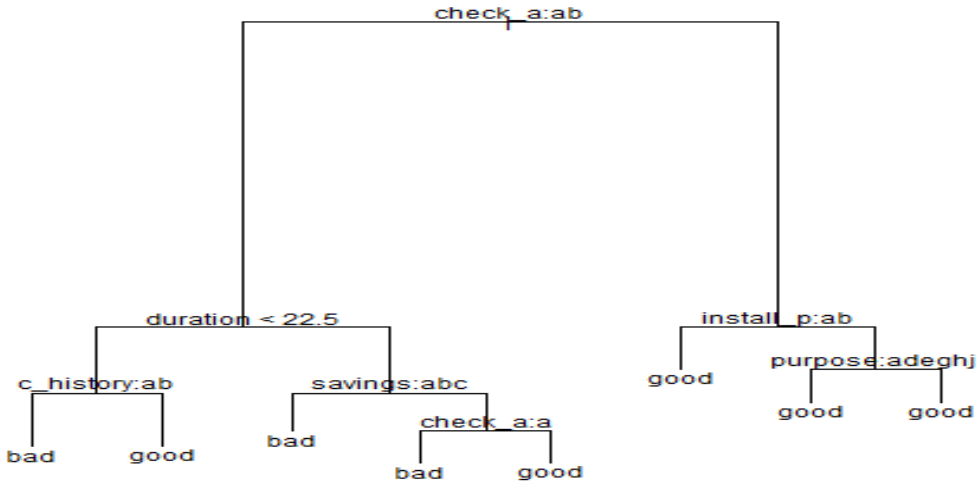
plot of chunk unnamed-chunk-3

```
# Split by Gini
german.tr.gini <- tree(target ~ ., data = german, method = "class", split
= "gini")
plot(german.tr.gini)
text(german.tr.gini)
```



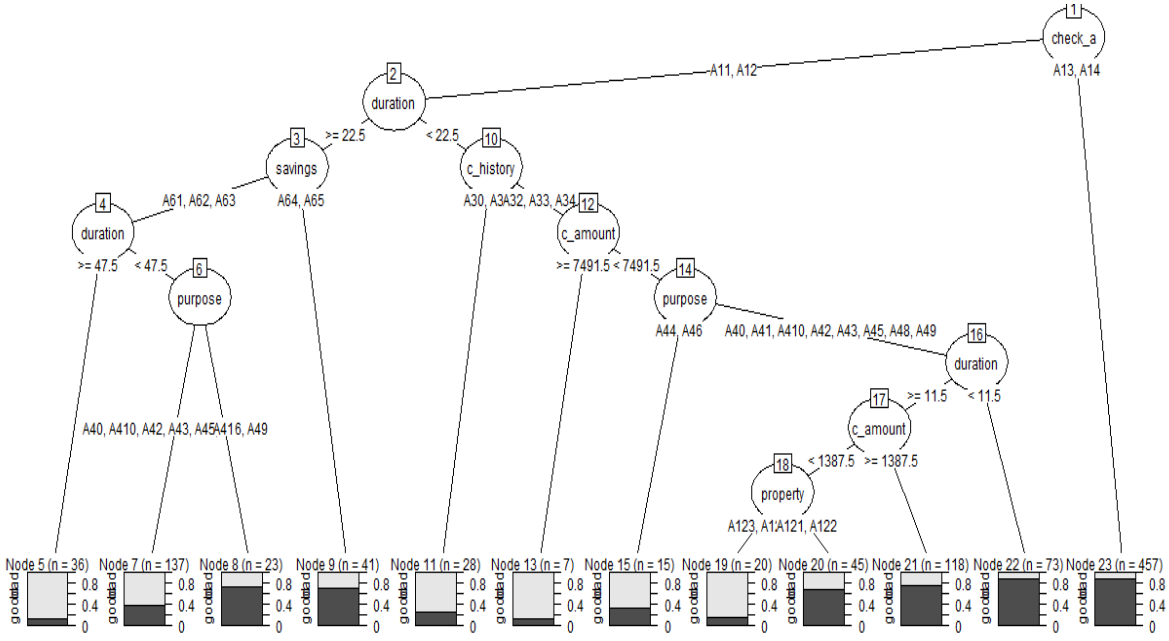
plot of chunk unnamed-chunk-4

```
# Split by Deviance
german.tr.dev <- tree(target ~ ., data = german, method = "class", split =
 "deviance")
plot(german.tr.dev)
text(german.tr.dev)
```



plot of chunk unnamed-chunk-5

```
# rpart package
german.dt <- rpart(target ~ ., data = german, method = "class", control =
rpart.control(xval = 10))
plot(as.party(german.dt))
```



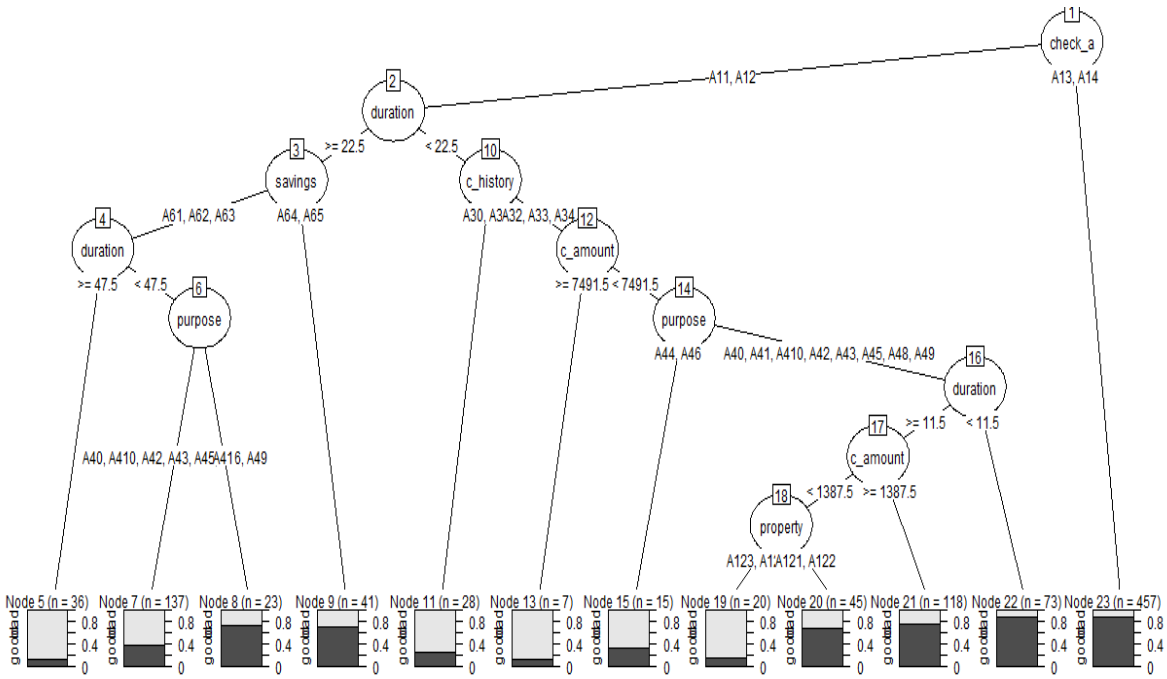
plot of chunk unnamed-chunk-6

3. Cross - Validation

- **control** : *rpart* algorithm 에 대한 상세 control.
  - **rpart.control** : 나무 모형 (rpart object) 에 대한 다양한 옵션을 줄 수 있다.
    - xval : # of cross-validation
- **printcp** : 적합된 나무 모형 (rpart obejct) 에 대한 cp table 을 보여준다.
- **plotcp** : Cross-Validation 결과를 그림으로 제공한다.

```
german.dt.cv <- rpart(target ~ ., data = german, method = "class", control
 = rpart.control(xval = 20))

plot(as.party(german.dt.cv))
```



plot of chunk unnamed-chunk-7

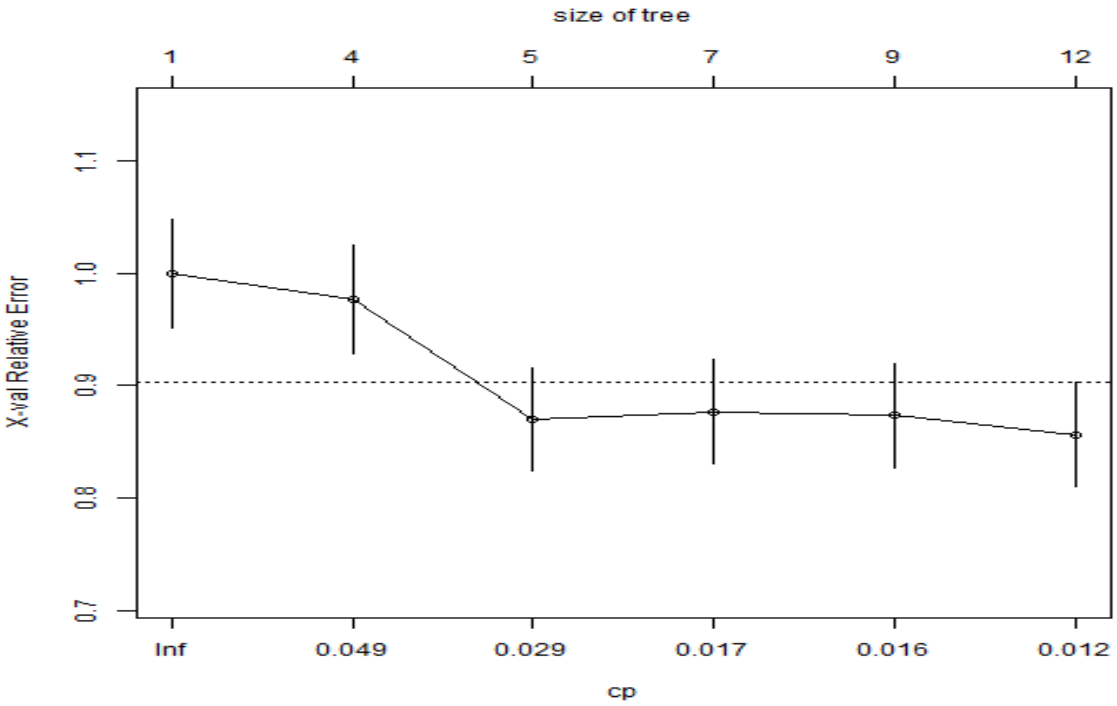
```
printcp(german.dt.cv)
```

```
##
## Classification tree:
## rpart(formula = target ~ ., data = german, method = "class",
##     control = rpart.control(xval = 20))
##
## Variables actually used in tree construction:
## [1] c_amount  c_history check_a   duration  property  purpose   savings

##
## Root node error: 300/1000 = 0.3
```

```
##
## n= 1000
##
##       CP nsplit rel error xerror  xstd
## 1 0.052      0      1.00   1.00 0.048
## 2 0.047      3      0.84   0.98 0.048
## 3 0.018      4      0.79   0.87 0.046
## 4 0.017      6      0.76   0.88 0.046
## 5 0.016      8      0.72   0.87 0.046
## 6 0.010     11      0.68   0.86 0.046
```

```
plotcp(german.dt.cv)
```



plot of chunk unnamed-chunk-8

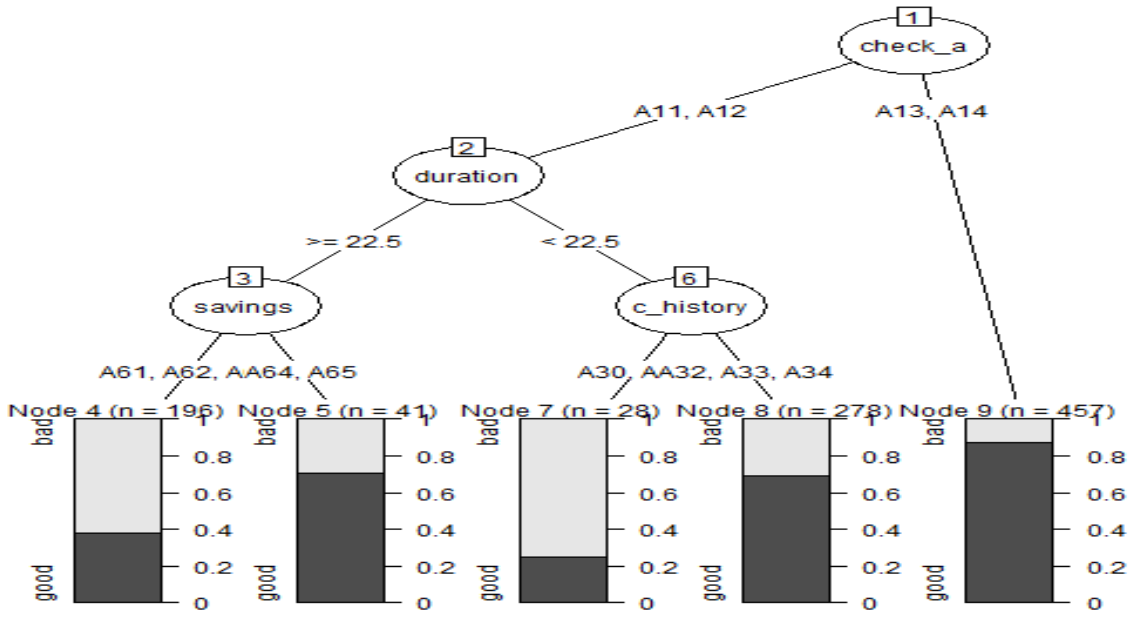
위의 Plot 을 보면 나무 모형의 크기가 5, 12 일 때 Cross-Validation Error 가 가장 낮다. 때문에 나무 모형의 크기가 5 일 때 cp 값으로 가지치기를 수행한다.

4. Pruning Tree

- **prune** : 가지치기 함수
  - tree : 적합된 모형의 오브젝트
  - cp : 가지치기에 적용할 Cost-Complexity parameter 값.

```
# pruning
german.dt.pr <- prune(german.dt.cv, cp = 0.029)
# german.dt.pr <- prune(german.dt.cv, cp = 0.012)
```

```
plot(as.party(german.dt.pr))
```



plot of chunk unnamed-chunk-9

5. Assignment

- UCI respository 에 있는 bank 데이터를 다운 받아 오늘 배운 내용을 수행하여 제출.
  - 압축을 풀고 난 뒤에 **bank-full.csv** 데이터 사용.
  - 데이터에 대한 설명은 **bank-names.txt** 에 있음.
- **제출기한** : 금주 금요일(18 일) 자정(12 시) 까지
- **제출방법**
  - 조교 메일로 제출. (yong.stat@gmail.com)
  - markdown 문서로 작성하여 html 파일만 첨부.
  - **파일명** : '학번_이름' (메일 제목도 동일하게)
- 수정해서 여러번 보내지 말것
  - **처음 받은 메일**만 과제 제출로 **인정**.
  - 그 **이후에 온 메일**은 **인정하지 않음**.