Big Data and R

2016.06.30

한남대학교 통계학과 권세혁교수

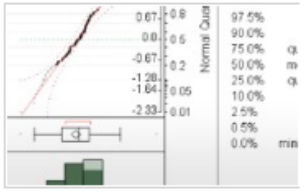
Why R in Big Data?

- Statistical Software
  - 통계학자에 의한 만들어지고 발전하는
  - 통계분석과 결과시각화를 위한
  - 통계학의 소프트웨어 : 대부분의 통계기법 분석 가능, 빅데이터분석 기법(Discriminant Analysis, Clustering Analysis, Decision Tree)

- Open Source, Big community
  - 전세계가 개발하고 : Google Analytic, Text Mining(Twitter)
  - 전세계로부터 도움을 받을 수 있음
  - 통계청 MDIS 서비스, 구글 맵 정보

- For Big Data Company
  - 빅데이터기업(구글, 페이스북) 분석 플랫폼
  - SAS/JMP, SPSS(버전18) R과 연동

JMP® and R Integration



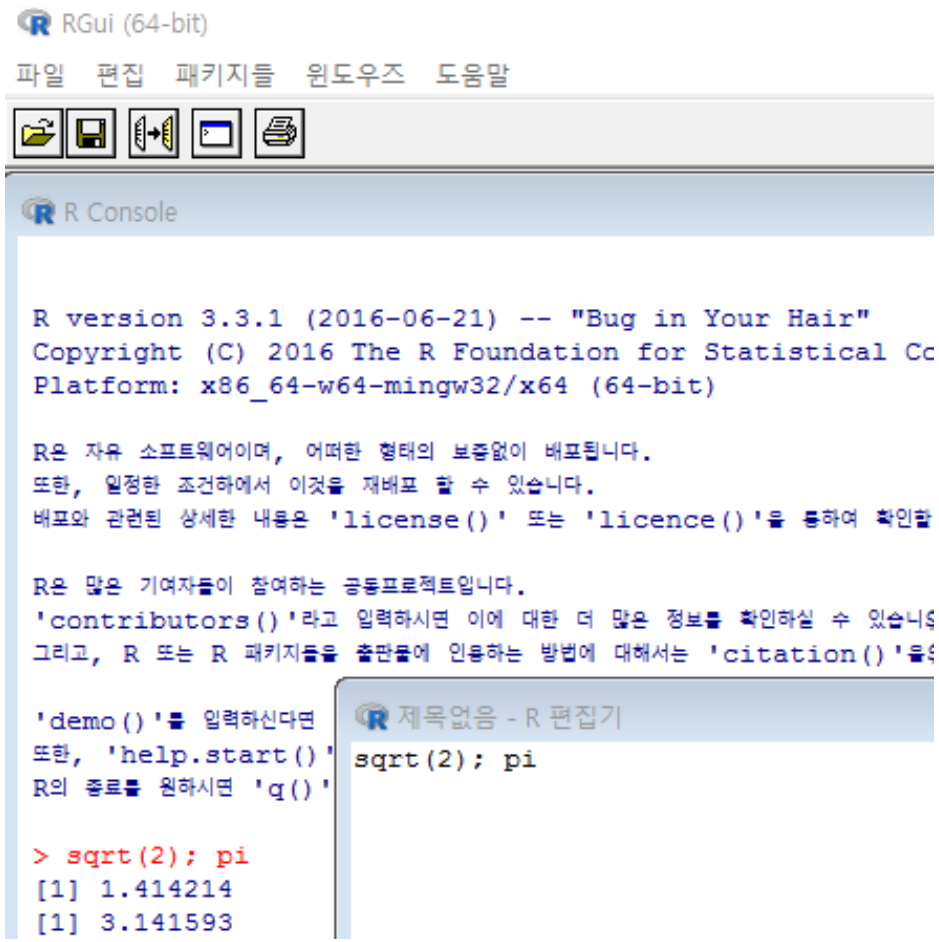
See how the advanced visualization in JMP can be integrated with the analytical capabilities of R.

Presenter: Kelci Miclaus

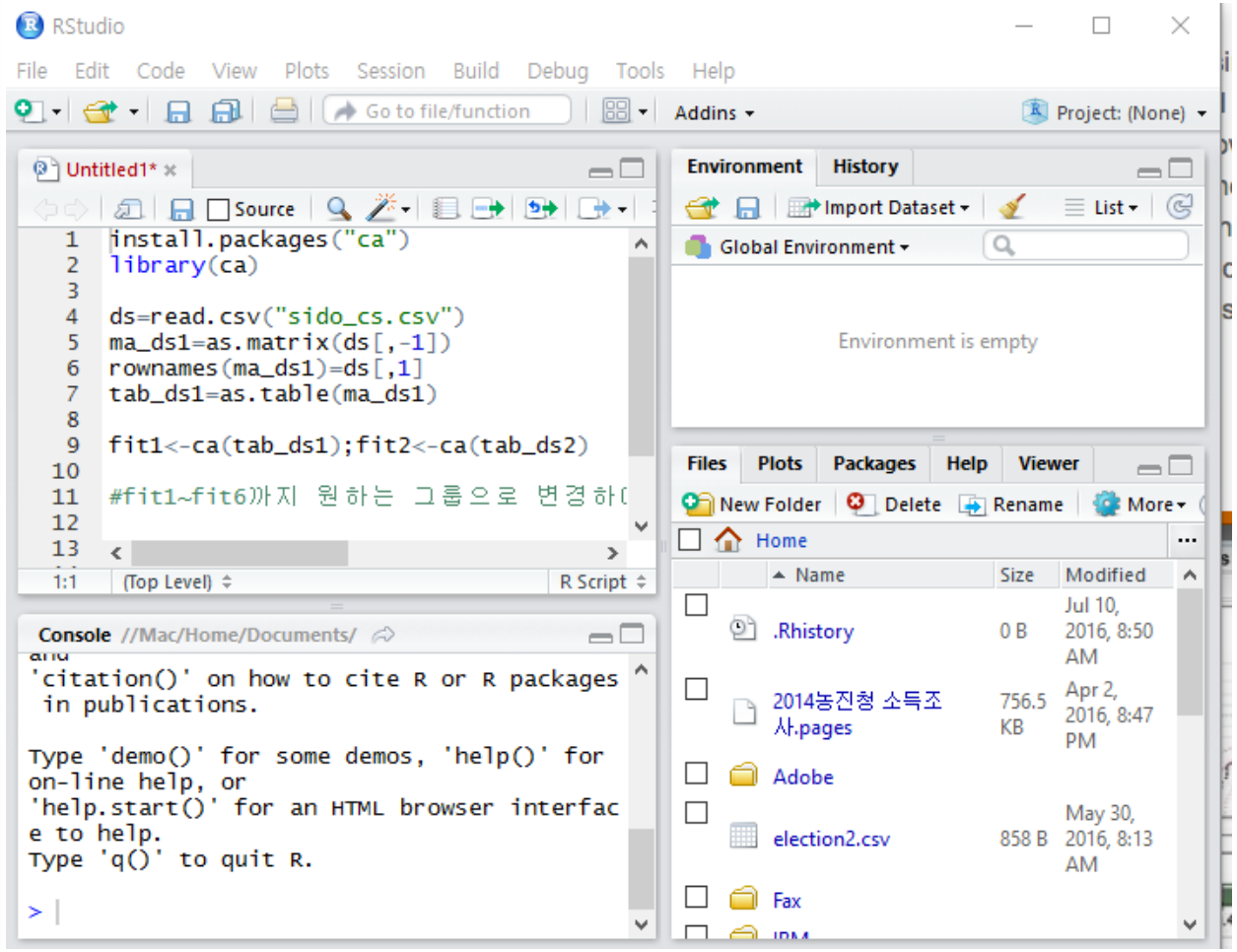
- Programming Language
  - 맞춤형 분석 툴을 만들 수 있음
  - 패키지 : Data Scientist

Installation of R and Rstudio

- http://www.r-project.org



- http://www.rstudio.com

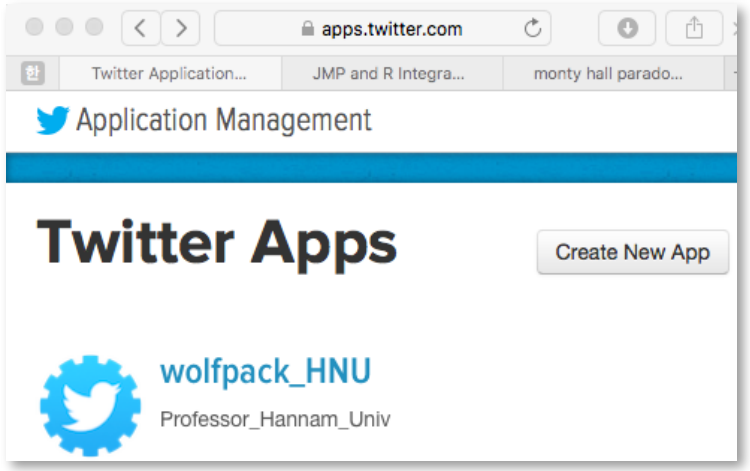


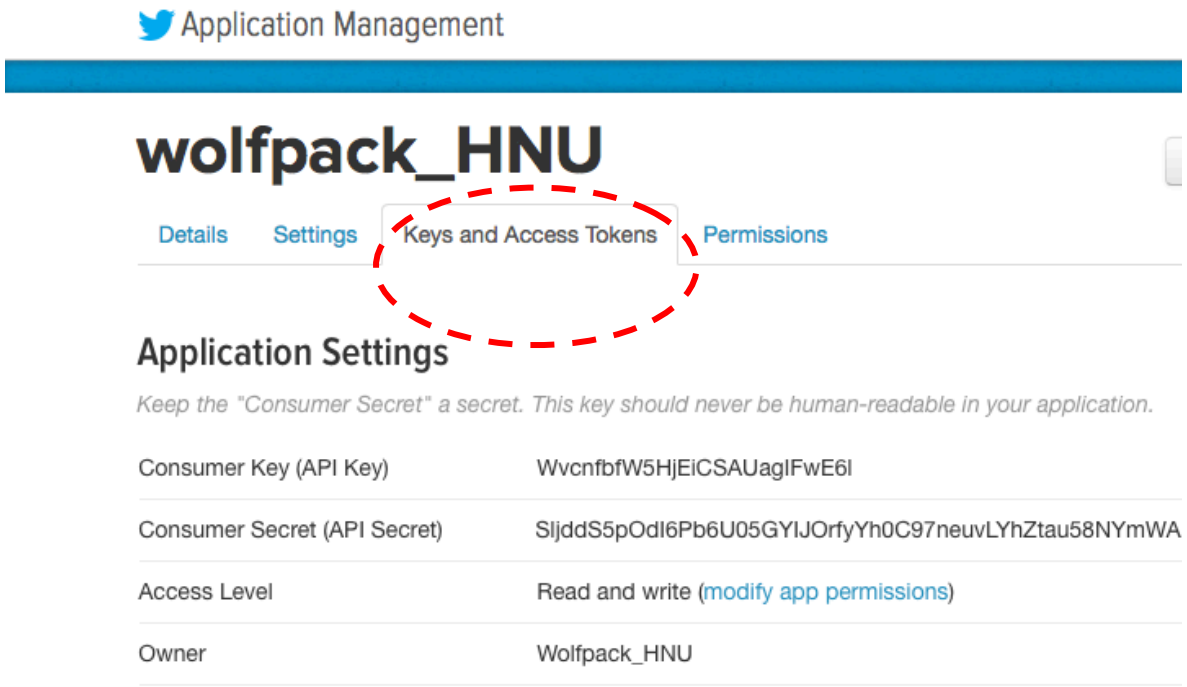
Text Mining with R (Twitter)

```
#R=Twitter mining with R
Sys.setlocale(category="LC_ALL",locale="English_United States.1252") #한글로된 문장을 찾음
1install.packages("twitteR"); require(twitteR)
install.packages("RCurl"); require(RCurl)
install.packages("ROAuth"); library(ROAuth)
install.packages("base64enc"); library(base64enc)

#Generating twitter Apps Key (https://apps.twitter.com ) with your Twitter account
consumerKey <- "WvcnfbfW5HjEiCSAUagIFwE6l"
consumerSecret <-"SljddS5pOdI6Pb6U05GYIJOrfyYh0C97neuvLYhZtau58NYmWA"
accesstoken <- "54046261-SDH8eu34JoW0JQFbXeamr90q9geIvatTq3diBrZ0F"
accesstokensecret <- "hpJdTZKH5g1Tykk6tv7cvCVjJvQ4OPxjkgKg4Oph5P12B"
setup_twitter_oauth(consumerKey, consumerSecret, accesstoken, accesstokensecret)
#Search keyword in Twitter
Search.tweets <- searchTwitter("빅데이터",n=100, lang="ko",since ="2016-06-13",until="2016-07-12")
Search.tweets
save(file="Search.tweets.csv", Search.tweets) #save data
```

http://apps.twitter.com






```
[[98]]
[1] "tomatoforever84: RT @wonsoonpark: 싱가포르 세계도시정상회의시장포럼 World Cities Summit Mayors Forum이 열리고 있습니다. 곧
저는 빅데이터를 활용한 대중교통 혁신에 관해 발표할 예정입니다 https://t.co/nuUunbPjyC"

[[99]]
[1] "limit0122: RT @SKSTORY_Blog: 많은 사람들의 휴가 계획이 궁금하거나, 사람에 치이는 휴가를 피하고 싶다면! 빅데이터로 살펴본 올 여름
휴가 키워드를 확인해보세요. https://t.co/OHMFyzlImT https://t.co/7nMt3a2uZY"

[[100]]
[1] "kisskenjo84: RT @wonsoonpark: 싱가포르 세계도시정상회의시장포럼 World Cities Summit Mayors Forum이 열리고 있습니다. 곧 저는
빅데이터를 활용한 대중교통 혁신에 관해 발표할 예정입니다 https://t.co/nuUunbPjyC"
```

GIS in R

#install.packages("ggmap"); #install.packages("ggplot2")

library(ggplot2); library(ggmap)

crop_korea <- read.csv("시도별_농작물_좌표_데이터.csv", header=T);names(crop_korea)

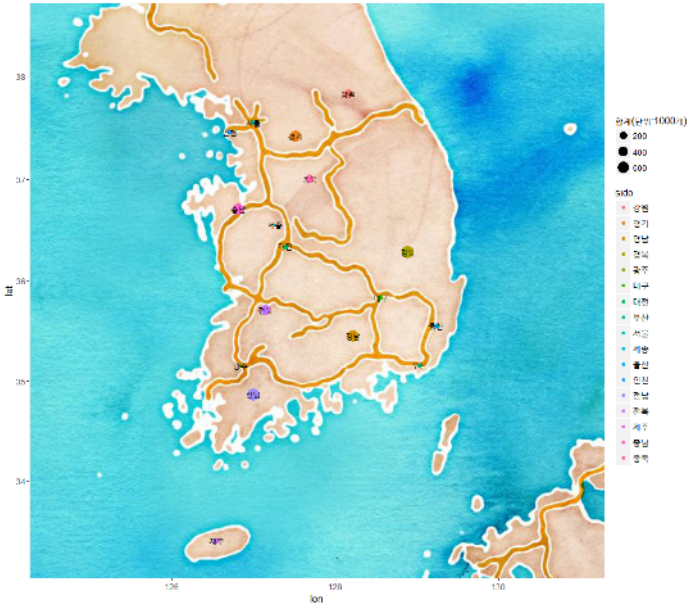
korea <- get_map("south-korea", zoom=7, maptype = "watercolor")

korea_map <- ggmap(korea)

korea_map <- korea_map + geom_jitter(data=crop_korea, aes(x=x, y=y, size = total/1000,color=sido)) + scale_size(name="Total")

korea_map + geom_text(data=crop_korea, aes(x = x, y = y, label=sido),size=3,col="black")+ geom_text(data=crop_korea, aes(x = y, y = y-0.1, label=total/1000),size=3,col="black")

```
#maptype=c("terrain", "terrain-background", "satellite",
#   "roadmap", "hybrid", "toner", "watercolor", "terrain-labels", "terrain-lines",
#   "toner-2010", "toner-2011", "toner-background", "toner-hybrid",
#   "toner-labels", "toner-lines", "toner-lite"),
```



- Monte-Hall show


```
data mh;
do i = 1 to 100000;
  prize = rand("TABLE",.333,.333); *당첨번호;
  initial_guess = rand("TABLE",.333,.333); *첫 콜;
  if initial_guess eq prize then do;
    new_guess = initial_guess;
    do until (new_guess ne initial_guess);
      new_guess = rand("TABLE",.333,.333);
   end;    end;
  if initial_guess ne prize then new_guess = prize;
  output;
  end;
run;
```

```
data mh2;
   set mh;
   win_by_keep = (initial_guess eq prize);
   win_by_switch = (new_guess eq prize);
run;

proc means data = mh2 mean;
   var win_by_keep win_by_switch;
run;
```

The MEANS Procedure

Variable	Mean
win_by_keep	0.3331200
win_by_switch	0.6668800

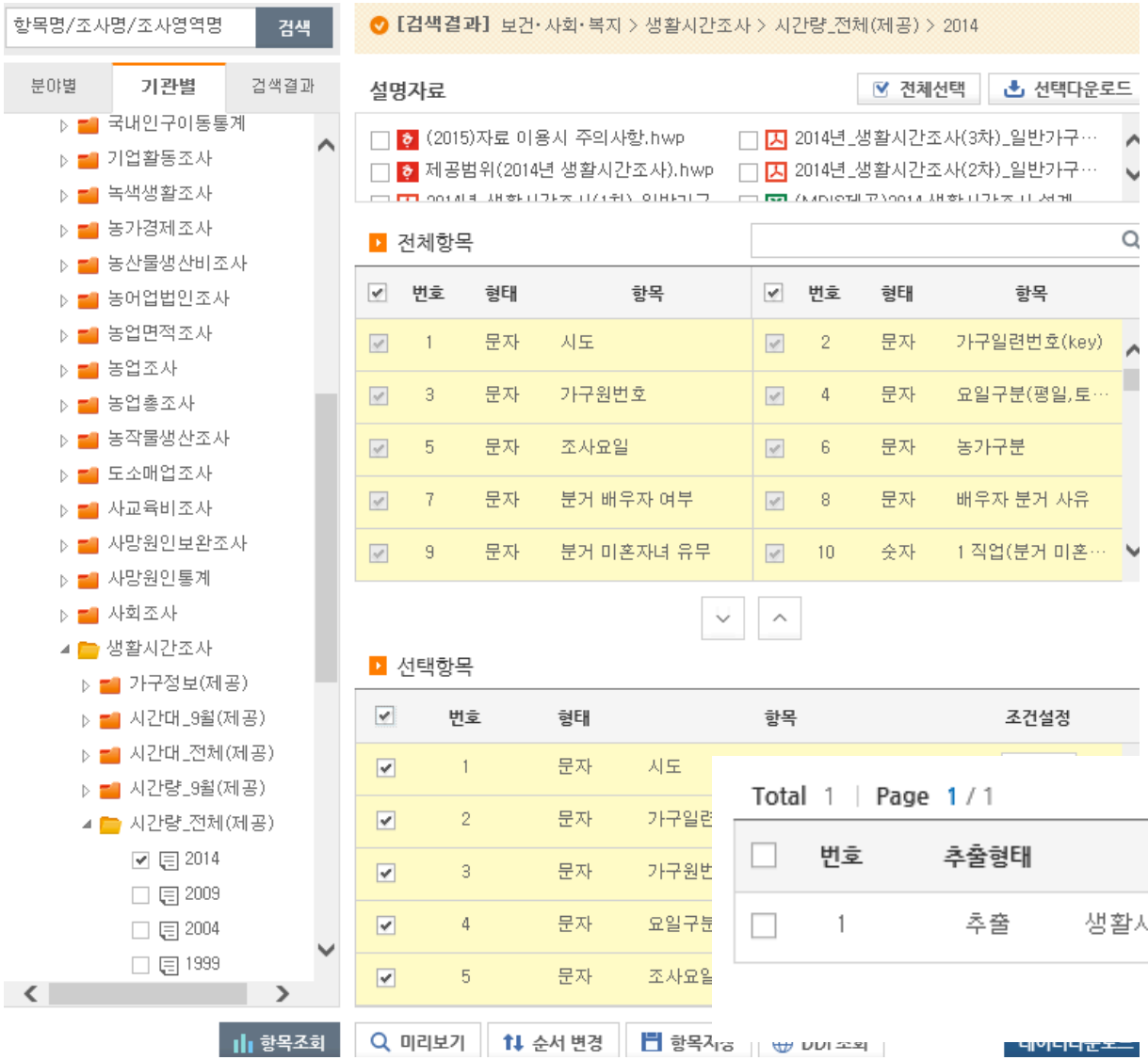
```
numsim = 100000
doors = 1:3
opendoor = function(x) {
  if (x[1]==x[2])
    return(sample(doors[-c(x[1])], 1))
  else return(doors[-c(x[1],x[2])])
}
swapdoor = function(x) { return(doors[-c(x[1], x[2])]) }
winner = sample(doors, numsim, replace=TRUE)
choice = sample(doors, numsim, replace=TRUE)
open = apply(cbind(winner, choice), 1, opendoor)
newchoice = apply(cbind(open, choice), 1, swapdoor)

sum(winner==choice)/numsim
sum(winner==newchoice)/numsim
```

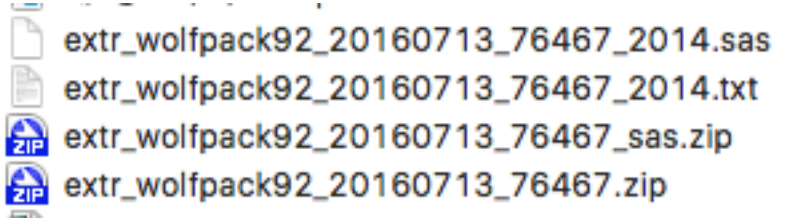
```
> sum(winner==choice)/numsim
[1] 0.3323
> sum(winner==newchoice)/numsim
[1] 0.6677
```

Micro Data Integrated Service https://mdis.kostat.go.kr

MDIS in 통계청



- SAS 프로그램 + Text 데이터 파일

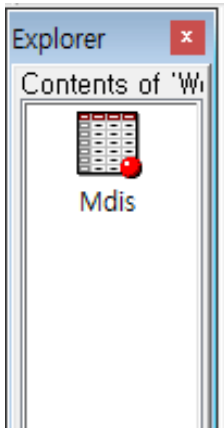


MDIS in SAS

extr_wolfpack92_20160713_76467_2014.sas

```
data WORK.MDIS                                    ;
%let _EFIERR_ = 0; /* set the ERROR detection macro variable */
infile 'Z:₩4Wolfpack₩extr_wolfpack92_20160713_76467_2014.txt
    format C1 $2. ;
    format C2 $5. ;
    format C3 $2. ;
    format C4 $1. ;
    format C5 $1. ;
    format C6 $1. ;
```

MDIS sas data



	시도	가구일련번호(key)	가구원번호	요일구분(평일,토요일,일요일)	조.
1	11	1	1	1	3
2	11	1	1	1	4
3	11	2	1	1	3
4	11	2	1	1	4
5	11	2	2	1	3
6	11	2	2	1	4
7	11	3	1	1	5

```
data mdis0;
set mdis;
 time1=sum(of c135-c137); time2=sum(of c140-c147);
time3=sum(of c148-c150);time4=sum(of c151-c155);
time5=sum(of c157-c164);time6=sum(of c165-c172);
run;

proc means data=mdis0 noprint nway;
            class c19 c17; var time1-time6;
            output out=out0 mean=;
run;

PROC EXPORT DATA=mdis0
  OUTFILE="Z:\4Wolfpack\mdis.csv" DBMS=csv
  REPLACE;
RUN;
```

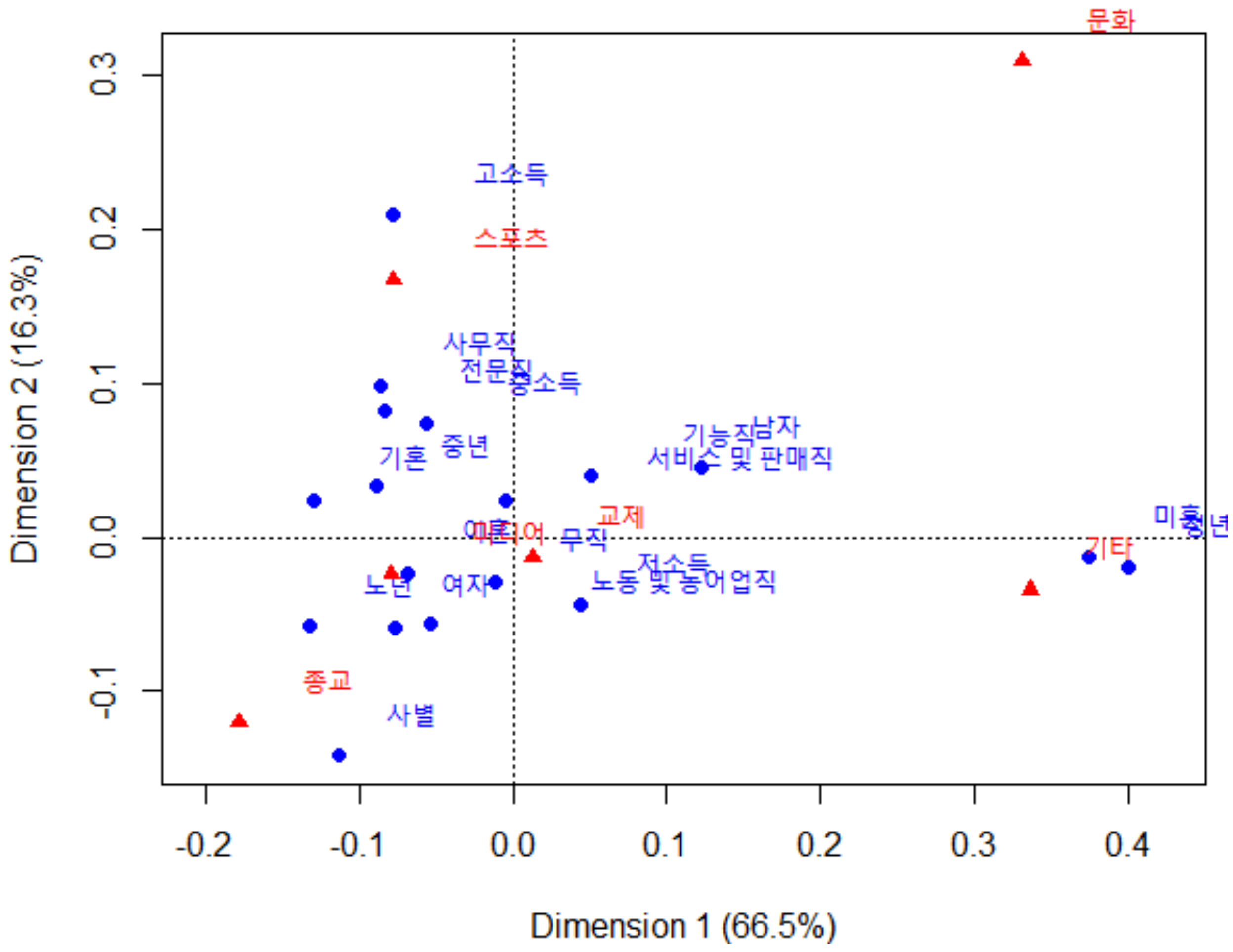
Corresponding Analysis in R

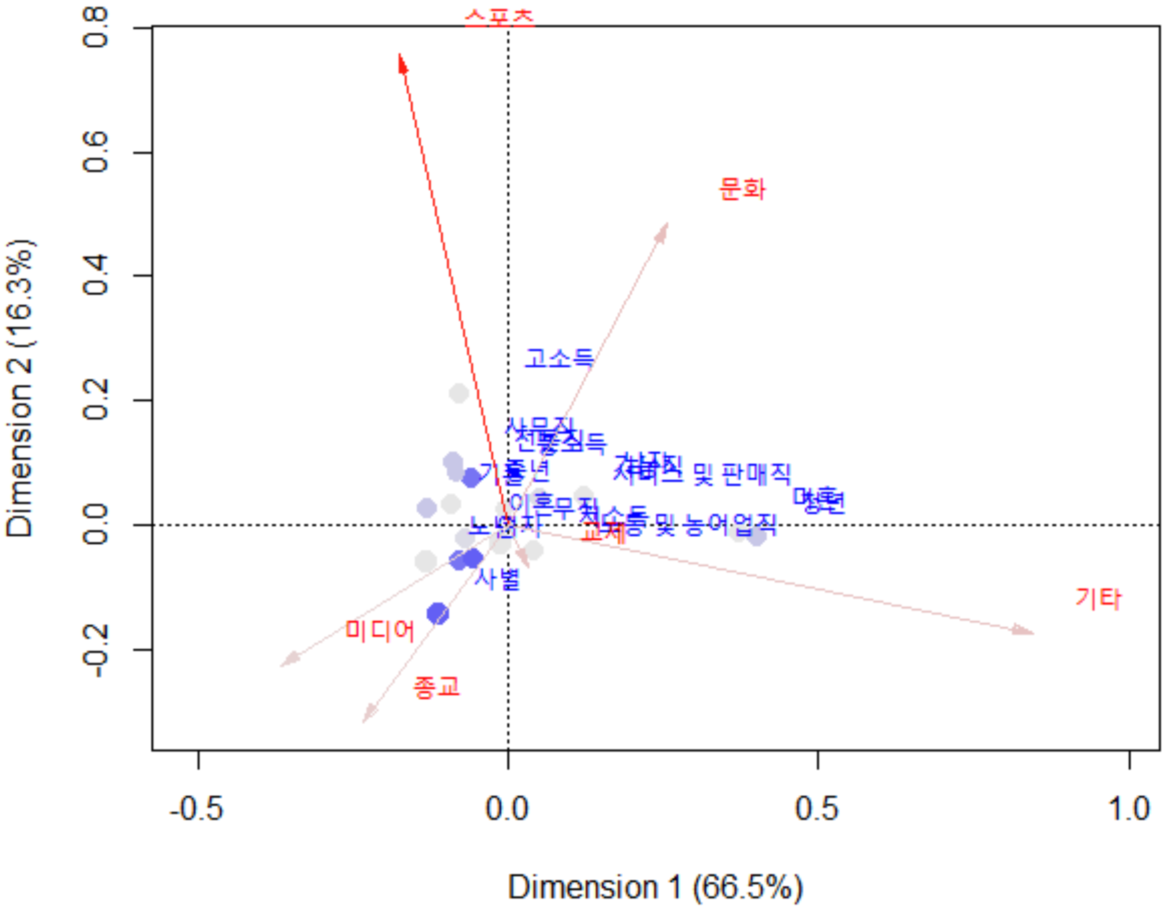
	교제	미디어	종교	문화	스포츠	기타
남자	38.90047	147.7393	7.039494	4.041074	40.53081	58.94629
여자	50.55372	143.4707	16.46469	4.212696	28.40341	33.09838
미혼	49.07827	119.1426	7.592688	7.649331	27.01596	78.37539
기혼	40.29193	145.7963	12.68571	3.016149	36.99503	28.95528
사별	65.57692	225.2592	23.15217	1.153846	36.65552	49.0301
이혼	43.45188	153.8703	10.64854	1.820084	34.93724	37.40586
청년	48.88204	114.868	7.661385	7.755282	25.80399	79.79754
중년	38.9226	125.9768	11.57641	3.702864	32.3873	28.14148
노년	54.99682	222.2695	17.99746	1.172918	46.68786	45.63255
저소득	48.9195	154.4704	13.38788	3.996747	33.09006	51.25839
중소득	35.07241	122.4909	9.757422	4.580014	33.31282	29.64881
고소득	35.74405	120.4067	5.684524	4.236111	45.36706	27.62897
전문직	41.68929	101.6742	14.66063	5.558069	31.51584	23.44646
노동 및 농	43.6674	136.8007	6.547357	1.261013	26.21145	32.60463
기능직	33.80518	115.9361	5.342466	3.576865	28.58447	37.01674
사무직	36.07764	121.6064	8.52075	5.461847	32.06827	24.58501
서비스 및	41.24856	101.8585	8.636364	4.447641	24.17146	27.65247
무직	53.46761	195.0121	18.10526	4.378543	44.62955	57.2834

```
#install.packages("ca")

sai=read.csv("2014년_7월_시간량평균_교차표.csv")
names(sai); attach(sai)
ma_sai=as.matrix(sai[,-1])
rownames(ma_sai)=sai[,1]
tab_sai=as.table(ma_sai)

library(ca)
fit<-ca(tab_sai)
print(fit)
summary(fit)
plot(fit)
plot(fit,mass=TRUE,contrib="absolute",map=
     "rowgreen",arrows=c(FALSE,TRUE))
```


Dimension 1 (66.5%)
Dimension 2 (16.3%)
문화
고소득
스포츠
사무직
전문직
중소득
기능직
남자
서비스 및 판매직
기혼
중년
교제
무직
미디어
저소득
노동 및 농어업직
기타
노년
여자
종교
사별

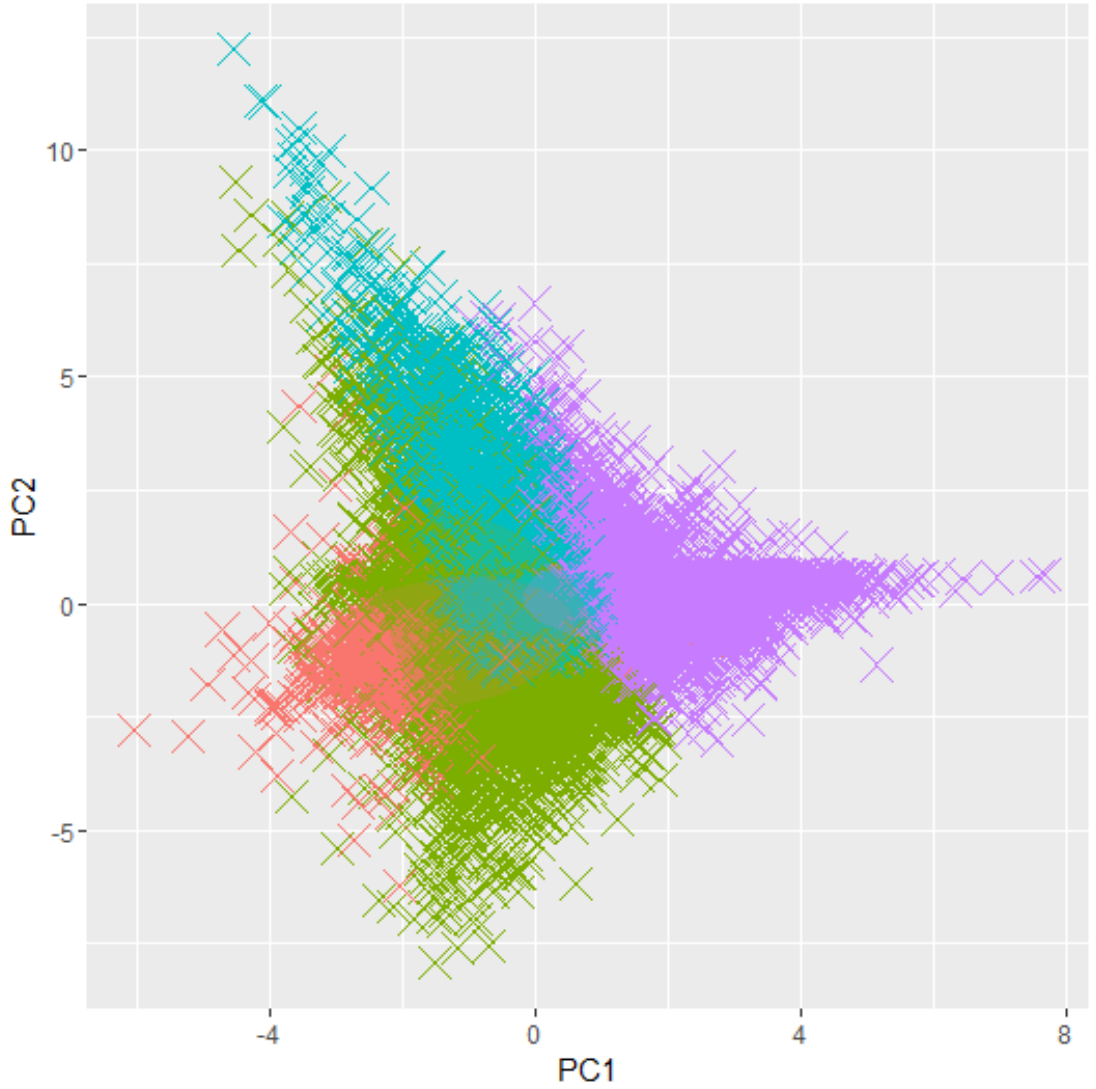

Dimension 1 (66.5%)
Dimension 2 (16.3%)
스포츠
문화
고소득
기타
미디어
종교
사별

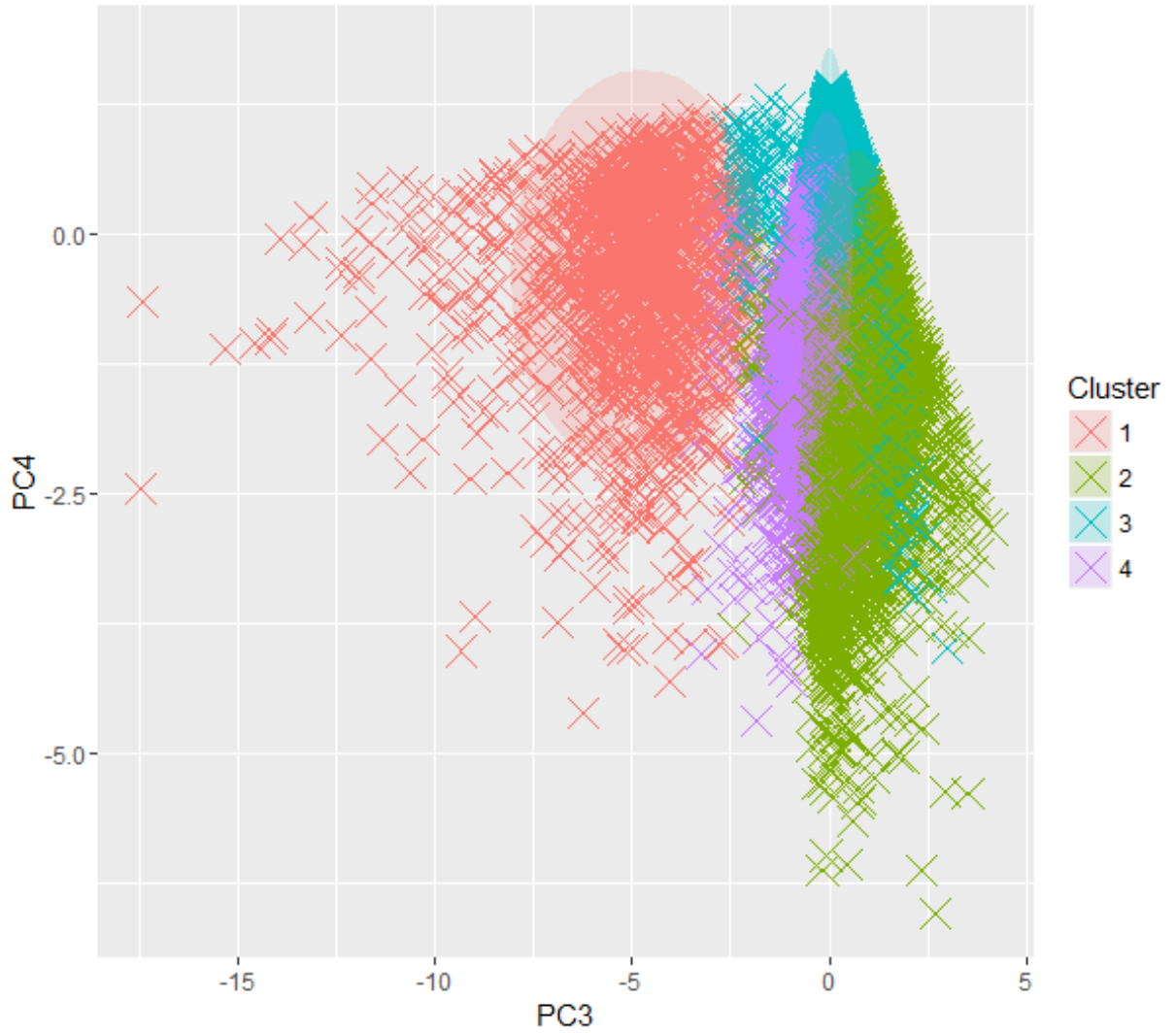
Clustering Analysis in R

```
install.packages("ggplot2")
ds=read.csv("2014년_시간량평균_데이터.csv",header=TRUE)
library(ggplot2)

pca    <- prcomp(ds[,8:13], retx=T, scale.=T) # scaled pca [exclude species col]
scores <- pca$x[,1:4]     # scores for first three PC's
print(pca)
# k-means clustering [assume 4 clusters]
km     <- kmeans(scores, centers=4, nstart=5)
ggdata <- data.frame(scores, Cluster=km$cluster)
ggplot(ggdata) +
  geom_point(aes(x=PC1, y=PC2, color=factor(Cluster)), size=5, shape=4) +
  stat_ellipse(aes(x=PC1,y=PC2,fill=factor(Cluster)),
               geom="polygon", level=0.95, alpha=0.2) +
  guides(color=guide_legend("Cluster"),fill=guide_legend("Cluster"))
```

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	요일구분	성별	혼인상태	연령대	소득	직업	조사차수	교제	미디어	종교	문화	스포츠	기타
2	1. 평일	2. 여자	1. 미혼	2. 중년	3. 고소득	4. 사무직	1. 2014년	50	90	0	0	90	30
3	1. 평일	2. 여자	1. 미혼	2. 중년	3. 고소득	4. 사무직	1. 2014년	70	30	0	0	100	10
4	1. 평일	2. 여자	2. 기혼	2. 중년	1. 저소득	6. 무직	1. 2014년	80	90	0	0	50	60
5	1. 평일	2. 여자	2. 기혼	2. 중년	1. 저소득	6. 무직	1. 2014년	90	200	0	0	0	30
6	1. 평일	1. 남자	2. 기혼	2. 중년	3. 고소득	1. 전문직	1. 2014년	30	0	0	0	20	40


10
5
0
-5
PC2
-4
0
4
8
PC1


0.0
-2.5
-5.0
PC4
-15
-10
-5
0
5
PC3
Cluster
1
2
3
4

Decision Tree

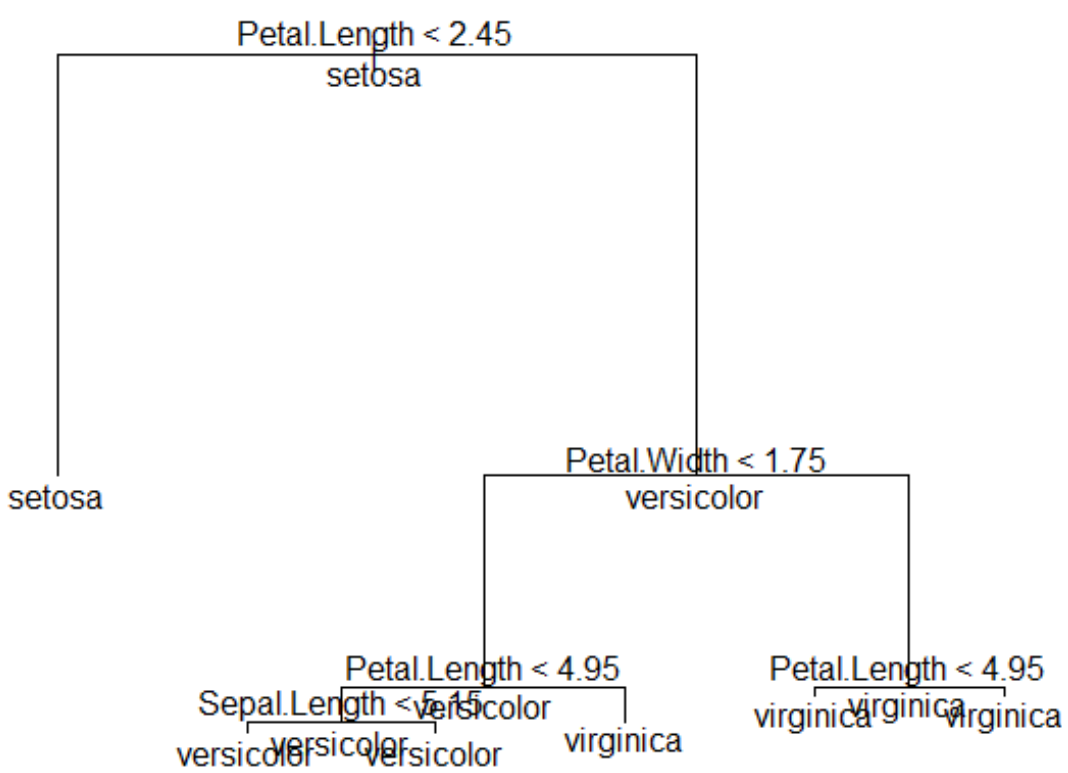
```
library(MASS) ; library(tree)
data(iris); table(iris$Species)
ir.fit=tree(Species ~., iris)
summary(ir.fit); ir.fit
plot(ir.fit); text(ir.fit, all = T)
ir.fit2 = prune.misclass(ir.fit,best=4)
plot(ir.fit2); text(ir.fit2, all = T)
```

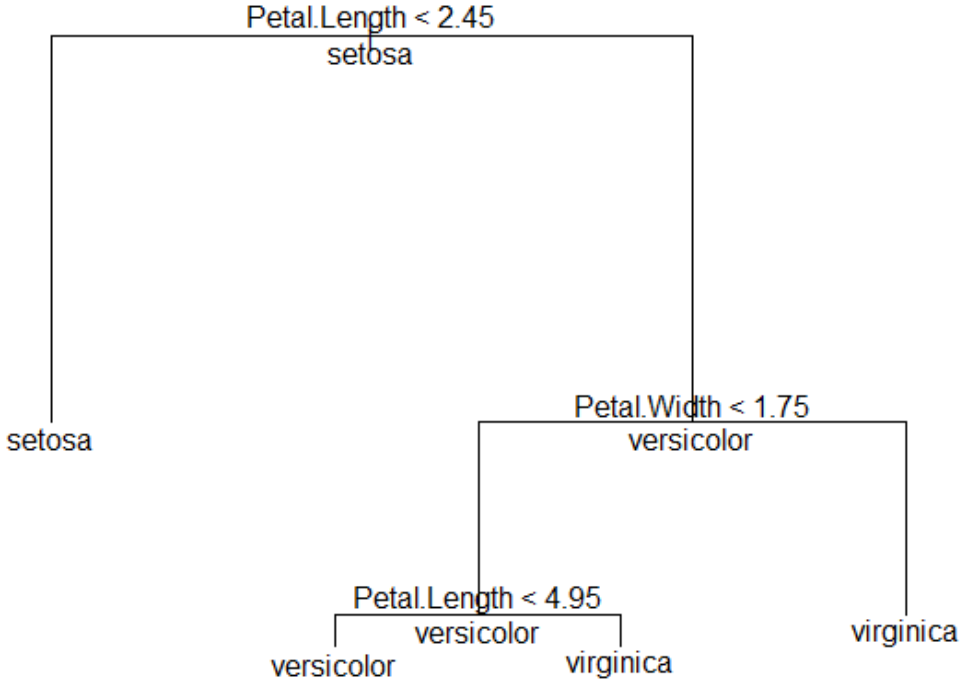
```
> summary(ir.fit)

Classification tree:
tree(formula = Species ~ ., data = iris)
Variables actually used in tree construction:
[1] "Petal.Length" "Petal.Width"  "Sepal.Length"
Number of terminal nodes:  6
Residual mean deviance:  0.1253 = 18.05 / 144
Misclassification error rate: 0.02667 = 4 / 150
```

```
> ir.fit
node), split, n, deviance, yval, (yprob)
      * denotes terminal node

 1) root 150 329.600 setosa ( 0.33333 0.33333 0.33333 )
   2) Petal.Length < 2.45 50   0.000 setosa ( 1.00000 0.00000 0.00000 ) *
   3) Petal.Length > 2.45 100 138.600 versicolor ( 0.00000 0.50000 0.50000 )
     6) Petal.Width < 1.75 54  33.320 versicolor ( 0.00000 0.90741 0.09259 )
      12) Petal.Length < 4.95 48   9.721 versicolor ( 0.00000 0.97917 0.02083 )
        24) Sepal.Length < 5.15 5   5.004 versicolor ( 0.00000 0.80000 0.20000 ) *
        25) Sepal.Length > 5.15 43   0.000 versicolor ( 0.00000 1.00000 0.00000 ) *
      13) Petal.Length > 4.95 6   7.638 virginica ( 0.00000 0.33333 0.66667 ) *
     7) Petal.Width > 1.75 46   9.635 virginica ( 0.00000 0.02174 0.97826 )
      14) Petal.Length < 4.95 6   5.407 virginica ( 0.00000 0.16667 0.83333 ) *
      15) Petal.Length > 4.95 40   0.000 virginica ( 0.00000 0.00000 1.00000 ) *
```


Petal.Length < 2.45
setosa
setosa
Petal.Width < 1.75
versicolor
Petal.Length < 4.95
Sepal.Length < 5.15
versicolor
versicolor
versicolor
virginica
Petal.Length < 4.95
virginica
virginica
virginica


Petal.Length < 2.45
setosa
setosa
Petal.Width < 1.75
versicolor
Petal.Length < 4.95
versicolor
versicolor
virginica
virginica

Sampling in R

```
library(sampling)
names(ds)
mytable=table(ds$요일구분,ds$소득)5
prop.table(mytable)
s=strata(ds,c("요일구분","소득"),size=c(43,12,5,14,5,5,14,5,5),method="srswor")
write.csv(s,"sampling.csv")
```

	A	B	C	D	E	F
1		요일구분	소득	ID_unit	Prob	Stratum
2	2238	1. 평일	3. 고소득	2238	0.017325	1
3	2615	1. 평일	3. 고소득	2615	0.017325	1
4	7051	1. 평일	3. 고소득	7051	0.017325	1
5	7099	1. 평일	3. 고소득	7099	0.017325	1
6	8003	1. 평일	3. 고소득	8003	0.017325	1
7	11159	1. 평일	3. 고소득	11159	0.017325	1
8	11491	1. 평일	3. 고소득	11491	0.017325	1
9	13733	1. 평일	3. 고소득	13733	0.017325	1