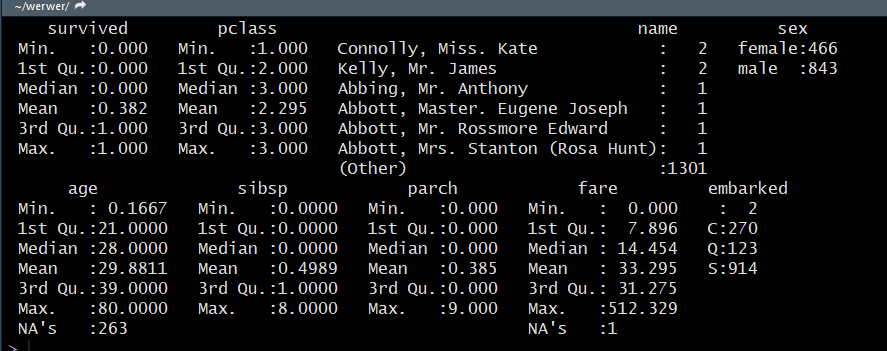
**R기반 통계학 보고서**

**2017204086 양지인**

1. 연관성 규칙

우선, titanic 파일을 List<-read.csv("./data/titanic.csv")을 통해 R로 불러오고 List라는 변수에 저장하였다. Summary 함수를 통해 자료를 요약해보면 여자는 모두 466명이며 남자는 843명인 것을 알 수 있다. 

<원본 자료를 요약한 것이다>

데이터를 정제화하기 위하여 18세가 넘는 사람들을 adult라 하고 나머지 사람들은 child로 계산하기로 하며, 나이가 측정이 안된 사람들은 생략하기로 하였다. 이름과 요금은 데이터를 분석하는데 쓸모가 없을 것 같으므로 삭제하기로 하였다. 또한 factor함수를 이용하여 보다 한눈에 데이터를 파악할 수 있도록 하였다.

데이터 정제화를 위하여 다음과 같은 소스코드를 이용하였다.

List$name<-NULL//이름을 삭제한다

List<-na.omit(List)//데이터가 NA인 것을 없앤다.

List$fare<-NULL//요금 열을 삭제한다

List$age<-ifelse(List$age>18,List$age<-"adult",List$age<-"child")

-18세를 넘으면 adult 아니면 child로 저장

List$survived<-ifelse(List$survived,List$survived<-"yes",List$survived<-"no")

-생존하였으면 yes 사망하였으면 no로 저장

List$age<-factor(List$age)

List$survived<-factor(List$survived)

List$pclass<-factor(List$pclass)

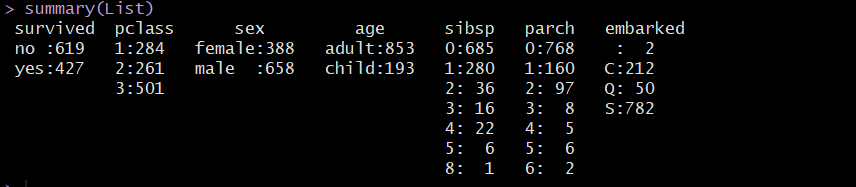
List$sibsp<-factor(List$sibsp)

List$parch<-factor(List$parch)

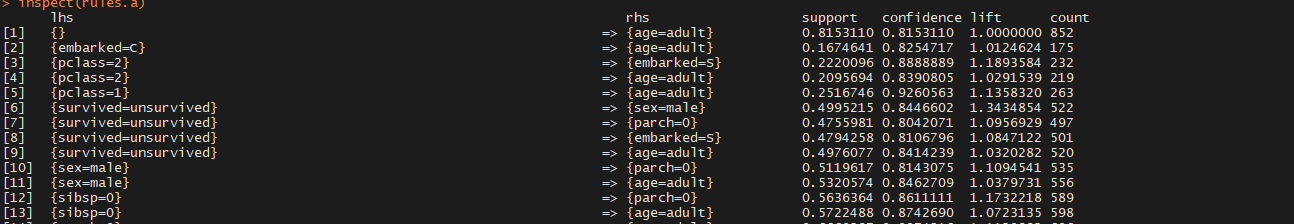
List$embarked<-factor(List$embarked)

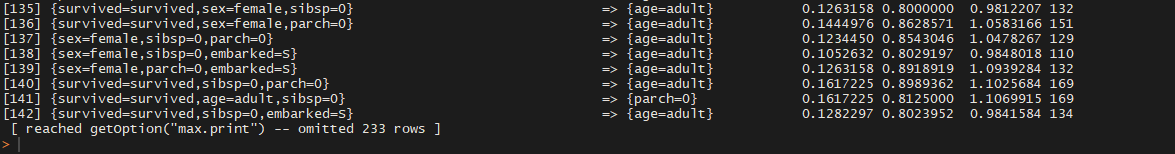
-자료들을 간단하게 하기위해 벡터형으로 변환한다.

정제된 데이터를 대략적으로 요약하면 총 427이 생존하였으며 618명이 사망하였다. 388명은 여자이고 657명은 남자이다. 어른의 수는 852명이며 아이들은 총 193명이다.



본격적으로 데이터를 연관성 규칙을 이용하여 분석하기로 하였다. 우선, List 전체 자료의 위 7가지의 분류를 이용하여 수도 많은 규칙들을 생성할 수 있었다. 우선 arules라는 패키지를 설치한 뒤 apriori라는 함수를 이용하여 규칙 및 연관성을 분석하여 rules.a라는 변수에 저장하기로 한 다음 inspect라는 함수를 이용하여 연관성 있는 규칙들을 직접 관찰해보기로 하였다.





너무 많이 나온 탓에 233열을 생략하였다고 화면에 출력이 된다. 결국, 아래의 소스코드를 이용하여 신뢰도, 지지도의 최솟값을 주거나, rhs와 lhs에 들어갈 수 있는 값을 제한하여 의미가 있는 규칙들을 생성하기로 하였다.

rules<-apriori(List,control=list(verbose=F),parameter=list(minlen=2,supp=0.4,conf=0.5),appearance=list(rhs=c("survived=no","survived=yes"),default="lhs"))

quality(rules)<-round(quality(rules),digits=3)

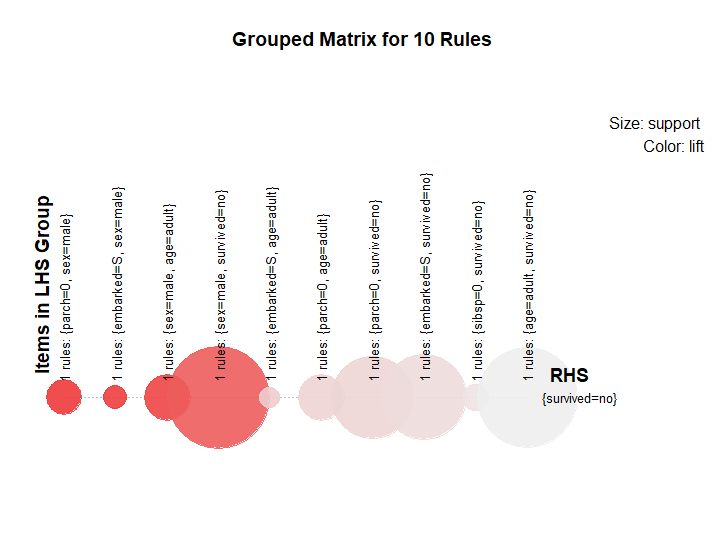
install.packages("arulesViz")

library(arulesViz)

plot(rules)

plot(rules,method="grouped")

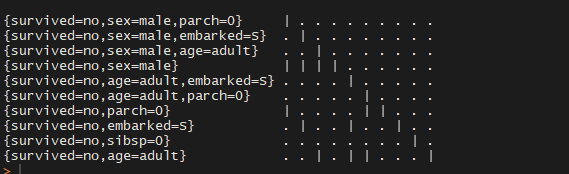
위의 소스코드를 이용하여 나름 정제된 rules의 데이터를 시각화 하였다.



rules.sorted<-sort(rules,by=”confidence”)

subset.matrix<-is.subset(rules.sorted,rules.sorted)

이 많은 데이터 중 연관성이 깊은 것들만 추출하기 위하여 지지도와 신뢰도가 최소0.4와 0.5를 넘고 rhs에 생존의 변수만 포함되게끔 하였다. 최소 2개 물품이 포함되게끔 하였으며 편하게 보기 위해 소수점은 3자리까지 출력하게 하였다. 이것을 rules라는 변수에 저장하였으며 그 결과 10개의 규칙들이 생성되었다. 마지막으로 규칙들의 신뢰도를 내림차순으로 정리한 뒤 rules.sorted라는 변수에 저장하였다. 중복된 규칙을 찾기 위하여 is.subset함수를 이용하였다.

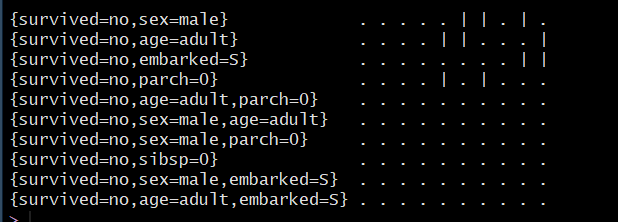


위 그림을 통해 중복되거나 쓸모 없는 규칙들을 찾아낼 수 있다. 예를 들면 규칙4, 7은 규칙1의 subset이 된다. Survived=no, sex=male, parch=0의 경우가 survived=no, sex=male이라는 규칙을 전제하고 있다는 의미이다. 그 외에도 규칙4는 3의 subset이라는 것 등 많은 것들을 알아낼 수 있다. 이러한 쓸모 없거나 중복된 규칙들을 아래 소스코드를 이용하여 제거하기로 하였다. 그리고 이 규칙들을 모두 rules.pruned라는 변수에 저장하였다.

subset.matrix[lower.tri(subset.matrix,diag=T)]<-0

redundant<-colSums(subset.matrix)>0

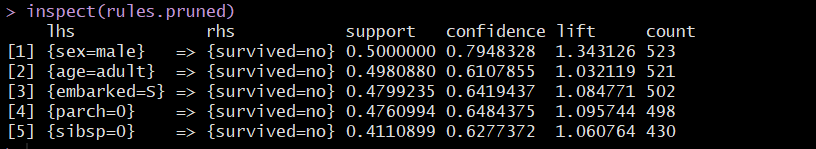
subset.matrix를 시각화 하게 되면 아래의 결과가 나온다.



which(redundant)

rules.pruned<-rules.sorted[!redundant]

rules.pruned를 시각화하게 되면 아래의 정제된 결과가 나온다.



아까와 다르게 쓸데 없는 데이터는 제거된 모습이 보인다.

1. 회귀 분석

우선, 회귀분석은 연관성 규칙보다는 적게 데이터를 정제화 하였다. 또한 회귀분석을 하는 함수는 직접적으로 벡터가 아닌 숫자를 인자로 전달받기 때문에 벡터 값으로 변환할 필요가 없었다.

List$name<-NULL

List<-na.omit(List)

List$survived<-ifelse(List$survived,List$survived<-"survived",List$survived<-"unsurvived")

이름을 없앤 뒤 na값을 모두 없애고 survived에 1과 0이 저장된 것을 survived와 ubnsurvived로 바꿨다. 데이터를 정제한 뒤 표본상관계수를 이용하여 두개의 변수 간의 연관성을 찾기로 하였다. 1이나 -`1에 가까울수록 두 변수 간의 관계성이 높다는 것을 의미한다.

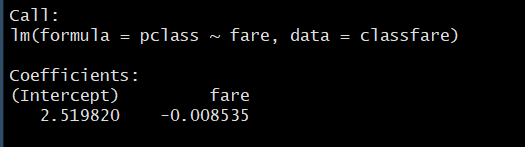
데이터에서 조건화 하여서 데이터를 추출한 뒤 공분산을 이용하여 최대한 연관성을 찾으려고 하였다. Pclass와 fare 즉 요금이 제일 관련성이 있을 듯하였다. 아래의 소스코드를 이용하여 변수 classfare에다 List의 pclass와 fare의 값만 추출하여 저장하였다. 그런 다음에 그 둘의 공분산을 구하였다. 그 결과 대략 -0.56이 나왔다. 음수이기 때문에 음의 상관관계를 갖는다는 것을 의미하였으며 절대값이 0.5를 넘는다는 것은 둘이 나름의 관계가 잇다는 것을 의미하였다.

classfare<-List[c("pclass","fare")]

cor(classfare$pclass,classfare$fare)

위 두 변수를 lm함수를 이용하여 회귀계수를 구한 값을 변수에 반환한다. 덧붙여 pclass를 y축으로fare을 x축으로 그래프로 시각화 한다. 그 위에 abline함수를 이용하여 회귀선을 시각화였다. 아래 시각화 된 그래프를 통해 분명하게 pclass와 fare의 음의 상관관계를 엿볼 수 있다. 즉 pclass의 번호 수가 높아질수록 요금은 낮아진다.

lm(pclass~fare,data=classfare)

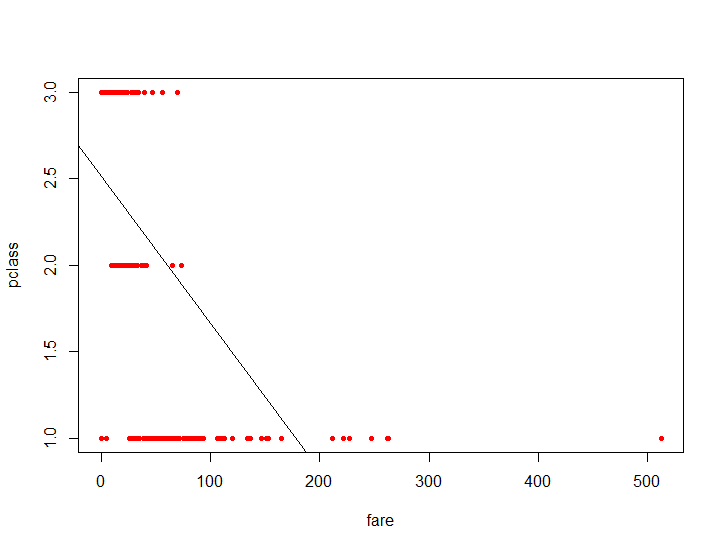


위 결과값을 통해 y=b0x+b1의 회귀계수를 구할 수 있다. B0의 값은 fare밑의 값이며 b1의 값은 intercept의 값이다.

cld <-lm(pclass~fare,data=classfare)

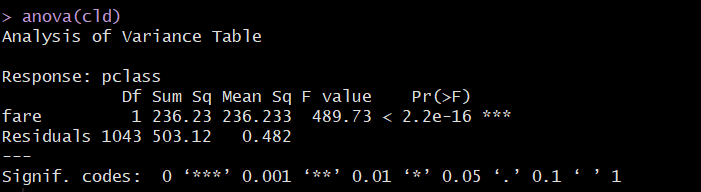
plot(pclass~fare,data=List,pch=20,col='red')

abline(cld)



Anova(cld)

위의 anova 함수를 통해 구축된 회귀모형으로부터 분산분석표를 볼 수 있다.



1. 의사결정 나무

타이타닉 데이터를 분석하기 위한 의사결정 나무를 만들기 위하여 아래의 소스코드를 이용하여 기본 데이터 정제화를 시행한 뒤 아래와 같은 패키지를 설치하였다. 즉, 데이털르 분석하기 위한 기본 환경설정을 하였다는 의미이다.

List<-read.csv("./data/titanic\_full.csv")

List$name<-NULL

List<-na.omit(List)

List$survived<-ifelse(List$survived,List$survived<-"yes",List$survived<-"no")

List$survived<-factor(List$survived)

List$pclass<-factor(List$pclass)

List$sibsp<-factor(List$sibsp)

List$parch<-factor(List$parch)

List$embarked<-factor(List$embarked)

List$sex<-factor(List$sex)

install.packages("caret")

install.packages("tree")

library(caret)

library(tree)

install.packages('C50')

library(C50)

set.seed(1000) 함수는 특정한 난수 생성 공식에서 처음 시작하는 값을 줘서 매번 같은 값이 랜덤하게 나오도록 한다. createDataPartition() 함수는 데이터를 훈련 데이터와 테스트 데이터로 분할하는 용도로 사용된다. Y 즉 List의 survived는 분류 (레이블)을 의미하고, p는 훈련 데이터에서 사용할 데이터의 비율을 의미한다. p = 0.7은 70%가 훈련 데이터라는 뜻이다. list는 결과를 리스트(TRUE)로 반환할지, 행렬(FALSE)로 반환할지를 선택한다. 아래의 소스코드를 이용하여 plot함수를 이용하여 의사결정나무를 만들었고 Text함수를 이용하여 그 위에 내용을 덧붙임으로써 데이터를 시각화였다. 아래의 시각화된 자료를 보았을 때, 분기가 복잡한 것으로 보이므로 가지치기를 해서, 분기를 합치는 과정이 필요하다.

set.seed(1000)

intrain<-createDataPartition(y=List$survived, p=0.7, list=FALSE)

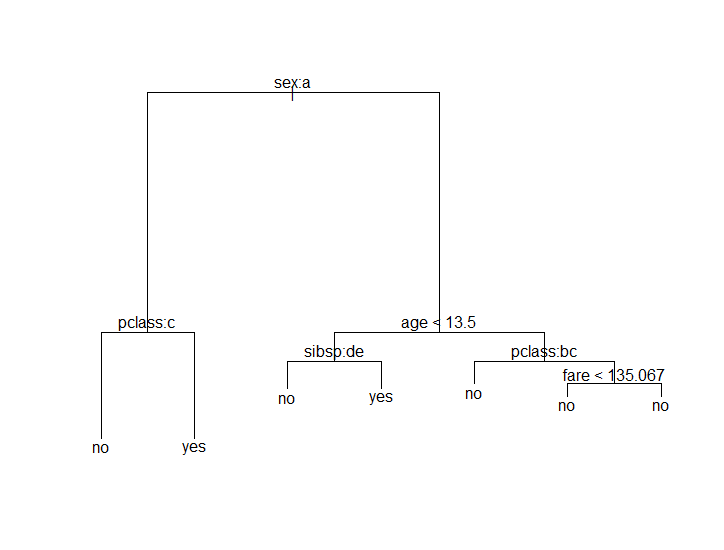
train<-List[intrain, ]

test<-List[-intrain, ]

treemod<-tree(survived~. , data=train)

plot(treemod)

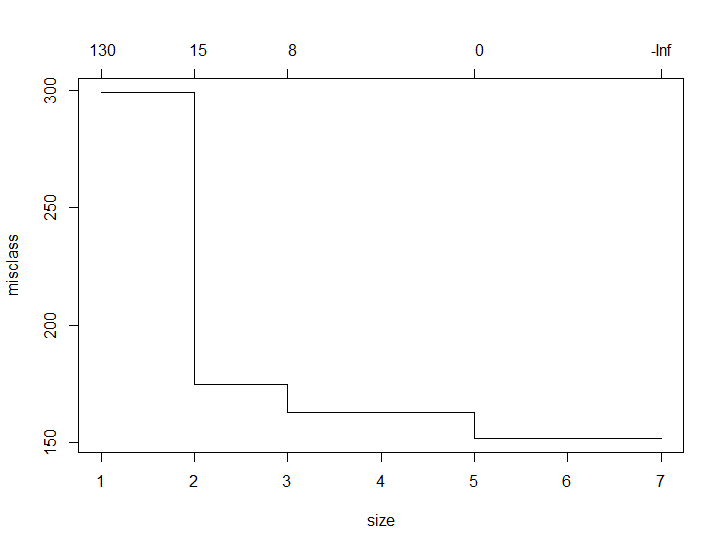
text(treemod)



cv.tree() 함수는 tree 객체를 대상으로 교차 확인을 실시할 때 사용한다. FUN은 가지치기 옵션으로 prune.tree 또는 prune.misclass로 설정할 수 있다.

cv.trees<-cv.tree(treemod, FUN=prune.misclass )

plot(cv.trees)

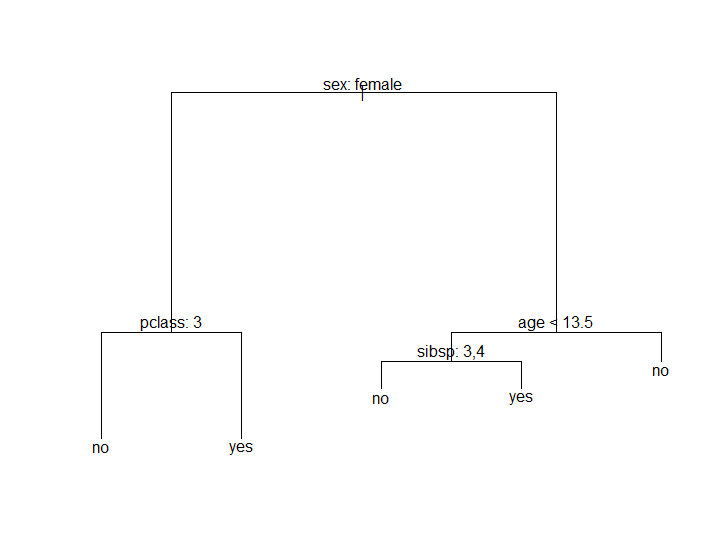


위의 그래프를 보면 5-6개의 가지가 있는 의사결정나무가 분산이 가장 낮은 것을 알 수 있다. 이것을 가만하여 의사결정나무를 가지치기 한다. pune.misclass() 함수는 가지치기를 한 모델을 생성하는 함수이다. best 옵션의 크기는 정수 값으로, 단말 노드의 수를 의미한다. 아래의 가지치기를 한 의사결정 나무는 처음에 만든 의사결정 나무보다 가지들이 정리가 된 것을 확인할 수 있다.

prune.trees <- prune.misclass(treemod, best=5,6)

plot(prune.trees)

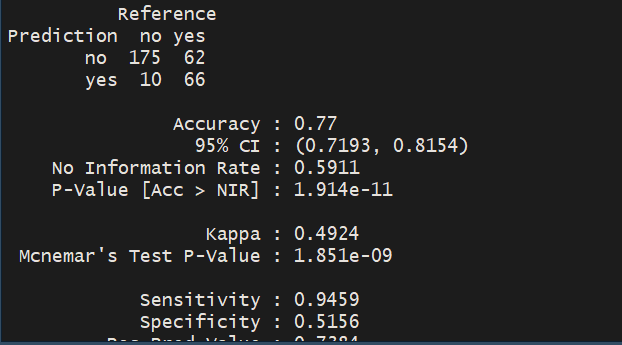
text(prune.trees, pretty=0)

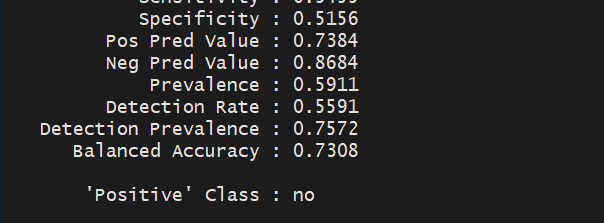


그 뒤에 마지막으로 predict함수를 이용하여 test셋의 survived를 예측한 다음, confusionMatrix()함수를 이용하여 모델의 정확성을 평가할 수 있다. Survived는 no와 yes로 분류를 할 수 있는데 분류의 예측값과 데이터의 실제 분류인 실제 값의 발생 빈도를 나열한 혼동 행렬로부터 평가 결과를 계산해낼 수 있다. 왼쪽 표에서는 실제값도 no인데 예측도 no였던 경우가 175이고 실제 값도 yes인데 예측도 yes였던 경우의 수가 66이었다. Accuracy가 0.77이라는 의미는 전체 예측에서 예측이 no이든 yes든 무관하게 옳은 예측의 비율을 의미한다. 따라서 위 자료는 나름 정확하게 생존 여부를 구분하고 있다고 할 수 있다.

treepred<-predict(prune.trees,test,type='class')

confusionMatrix(treepred,test$survived)





<각 분석방법의 장점 단점 및 비교>

1. 연관성 규칙

장점:

1. 유용성이 상당히 크고, 신축적인 특성도 가지고 있어 실무적으로 다양한 용도에 활용할 수 있다.
2. 탐색적인 기법으로서 분석 결과의 이해가 용이하다
3. 강력한 비목적성 분석기법이다.
4. 데이터의 형태가 간단하고 다루기 쉽다.
5. 거래와 품목이 많더라도 계산방법이 간단하다.

단점:

1. 개인적으로 요금이나 나이와 같은 세부적인 자료의 연관성을 찾기는 부적합하였다.

2. 상당한 수의 계산과정이 필요하다.

3. 품목의 수가 많을 때는 적절한 대상 품목을 선정하여 연관성 규칙을 찾아내야 한다.

4. 자료를 구성하고 있는 품목들의 빈도수가 비슷하지 않을 때는 빈도수가 적은 품목들의 경우 제외되기 쉽다.

1. 회귀분석

장점:

1. 한 개의 종속변수 *Y*와 한 개의 [독립변수](https://terms.naver.com/entry.nhn?docId=1923949&ref=y) *X*사이의 선형관계를 파악하거나, 한 개의 독립변수의 일정한 값에 대응되는 종속변수의 값을 예측할 수 있다.
2. 어떤 현상으로부터 관찰 가능한 변수들 간의 인과관계를 따져 모형화하고 분석할 수 있다.
3. 다른 모든 요인들 로부터 한 요인의 영향을 분리할 수 있다는 점에 있다
4. 다른 분석기법에 비해 상대적으로 적은 정보량을 요구한다

단점:  
1 회귀분석의 한계는 경로분석처럼 간접관계 혹은 매개효과를 모형화 하기 어렵고, 확인적 요인분석을 통한 측정오차를 반영하기 어렵다.

2 분석에 대한 오차항의 가정은 그대로 유지한다.

1. 의사결정 나무

장점

1. 자료를 보기 쉽게 분기점을 사용하여 시각화 할 수 있다.

2. 설명력이 높다

3. 결과에 대한 근거를 나뭇가지 형태로 추적할 수 있다.

4. 빠르고 변수 선택능력이 잇다.

5. 많은 변수들을 대상으로 종속변수에 영향이 높은 변수를 선택할 수 있다.

단점

1. 종속변수가 연속형일 때 쓸 수 없다.
2. 설명변수가 연속형일 때 낮은 예측능력을 보일 수 있다.
3. 자료의 추가에 의하여 나무구조가 바뀔 수 있다.
4. 비선형데이터에는 적합하지 못하다

<간단한 비교>

의사결정나무는 연관성 규칙 분석이나, 회귀분석에 비해 시각화 된 자료를 읽기가 너무나 쉬웠다. 즉, 회귀 분석이나 연관성 분석을 이용하여 시각화 된 자료들의 경우 처음 접하는 사람들의 경우 그 자료자체를 해석하기 어려웠을 것이다. 그러나 의사결정나무의 경우 단순하게 모든 사람들이 접해 봤을 법한 나뭇가지를 이용하여 분기점을 설정하고 이 마저도 쉽게 정리해 놓아 처음 보는 사람들일지라도 자료가 무엇을 의미하는지 쉽게 알아보거나 해석할 수 있을 것이다.

단순회귀분석의 경우 연관성 규칙 분석이나 의사결정 나무에 비해 독립변수 및 종속변수를 이용하여 식을 분명하게 표현해서 그런지 변수 간 인과관계를 분명하게 알 수 있었다. 식을 통해 독립변수가 정확히 종속변수에게 어떠한 영향을 미치는지 볼 수 있기 때문이다.

연관성 규칙 분석의 경우 다른 분석기법들에 비해 품목들을 서로 독특하게 관계를 맺고 있다는 것을 쉽게 명시하였다. 각 품목들은 서로 독립적으로 존재하는 것이 아닌 2개의 품목, 3개의 품목을 쉽게 묶을 수 있다는 의미이다. 일상 속에서는 묶어서 생각해보기도 어려운 품목들을 단 몇 개의 함수만으로도 다양한 묶음들을 한꺼번에 볼 수 있다는 것이 크나큰 장점이었다.

개인적으로 의사결정나무가 위 타이타닉 데이터를 분석하는데 있어 가장 유용한 분석기법이 아니었나 싶다. 우선 타이타닉 데이터의 경우 품목이 상당히 많았다. 그렇기 때문에 이 많은 품목들을 다른 분석 기법에 비해 나뭇가지만으로 간단하게 표현할 수 있었던 것은 나무의사결정 뿐이었 던 것 같다. 다른 분석기법의 경우 경우의 수가 너무 많아서 의도적으로 품목들을 제거해야 하는 경우가 많아 이 기법에 비해 상대적으로 불편했던 것 같다. 또한 이분화 된 자료 및 세분화된 자료 모두를 분기를 이용하여 나뭇가지로 나누는 것 또한 위 타이타닉의 다양한 유형의 자료를 표현하기에 상대적으로 적합했다고 본다.