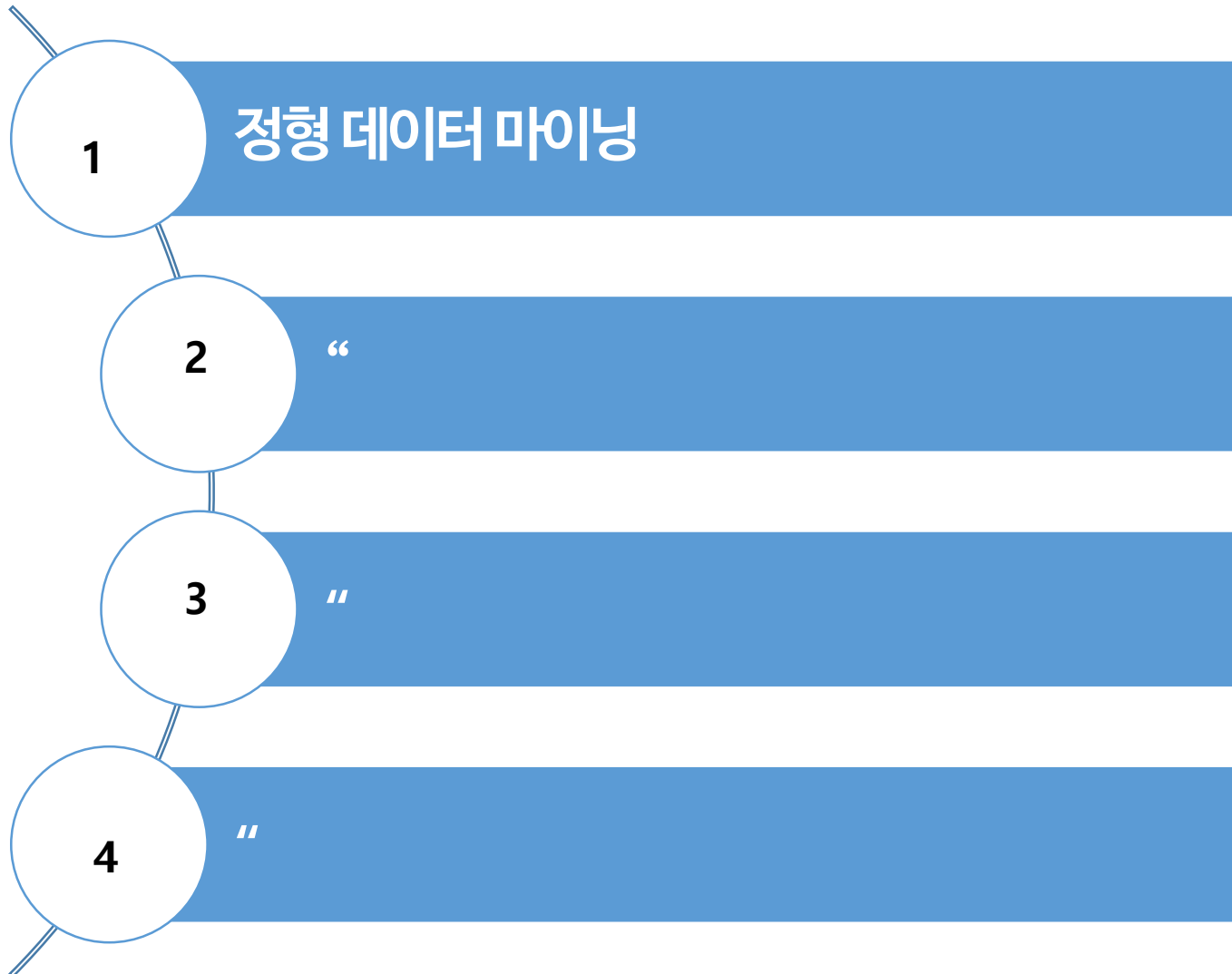


한국기술교육대학교 제2캠퍼스 실학관 902호

『2일차』 : 오후

◆ 훈련과정명 : 데이터 분석과 활용

◆ 훈 련 기 간 : 2023.12.19 - 2023.12.20



『3과목』 데이터 분석

제5장 정형 데이터 마이닝





제1절 데이터 마이닝의 개요

학습목표

- 데이터마이닝의 개념을 이해한다.
- 데이터마이닝 방법론의 종류를 이해한다.
- 데이터마이닝 절차를 이해한다.
- 데이터마이닝을 위한 데이터 분할과 모형 평가를 할 수 있다.

눈높이 체크

- 데이터마이닝을 들어 보신 적이 있나요?
- 데이터마이닝 방법론의 종류를 알고 계신가요?
- 데이터마이닝 절차를 알고 계신가요?



제1절 데이터 마이닝의 개요

데이터 마이닝

● 개요

- 데이터 마이닝은 대용량 데이터에서 의미 있는 패턴을 파악하거나 예측하여 의사결정에 활용하는 방법

● 데이터 마이닝과 통계분석과 차이점

- 통계분석: 가설이나 가정에 따른 분석이나 검증
- 데이터 마이닝: 다양한 수리 알고리즘을 통해 데이터베이스의 데이터로부터 의미 있는 정보를 찾아내는 방법

● 데이터 마이닝 종류

- 정보를 찾는 방법론에 따른 종류: 인공지능, 의사결정나무, K-평균 군집화, 연관분석, 회귀분석, 로짓분석, 최근접이웃
- 분석 분석대상, 활용 목적, 표현방법에 따른 분류: 시각화 분석, 분류, 군집화, 포케스팅



제1절 데이터 마이닝의 개요

데이터 마이닝

- 데이터 마이닝 사용분야
 - 병원에서 환자 데이터를 이용해서 해당 환자에게 발생 가능성이 높은 병을 예측
 - 기존 환자가 응급실에 왔을 때 어떤 조치를 먼저 해야 하는지 결정
 - 고객 데이터를 이용해 해당 고객의 우량/불량을 예측해 대출 적격 여부 판단
 - 세관 검사에서 입국자의 이력과 데이터를 이용해 관세 물품 반입 여부 예측



제1절 데이터 마이닝의 개요

데이터 마이닝

- 데이터 마이닝 최근 환경
- 데이터 마이닝 도구가 다양하고 체계화되어 환경에 적합한 제품을 선택하여 활용 가능하다.
- 알고리즘에 대한 깊은 이해가 없어도 분석에 큰 어려움이 없다.
- 분석 결과의 품질은 분석가의 경험과 역량에 따라 차이가 나기 때문에 분석 과제의 복잡성이나 중요도가 높으면 풍부한 경험을 가진 전문가에게 의뢰할 필요가 있다.
- 국내에서 데이터 마이닝이 적용된 시기는 1990년대 중반이다.
- 2000년대에 비즈니스 관점에서 데이터 마이닝이 CRM의 중요한 요소로 부각되었다. 대중화를 위해 많은 시도가 있었으나 통계학 전문가는 대기업 위주로 진행되었다.



제1절 데이터 마이닝의 개요

데이터 마이닝 분석방법

분석방법	내용
지도학습	의사결정나무, 인공신경망, 일반화 선형 모형, 회귀분석, 로지스틱 회귀분석, 사례기반 추론, 최근접 이웃법
비지도학습	OLAP, 연관성 규칙발견, 군집분석, SOM



제1절 데이터 마이닝의 개요

분석 목적에 따른 작업유형 기법

목적	작업유형	설명/사용기법
예측	분류규칙 Classification	가장 많이 사용되는 작업. 과거의 데이터로부터 고객특성을 찾아내어 분류모형을 만들어 이를 토대로 새로운 레코드의 결과값을 예측하는 것으로 목표 마케팅 및 고객 신용평가 모형에 활용 회귀분석, 판별분석, 신경망, 의사결정 나무
	연관규칙 Association	데이터 안에 존재하는 항목간의 종속관계를 찾아내는 작업. 제품이나 서비스의 교차판매, 매장진열, 첨부우편, 사기적발 등의 다양한 분야에 활용됨 동시발생 매트릭스
설명	연속규칙 Sequence	연관규칙이 시간관련 정보가 포함된 형태. 고객의 구매이력속성이 반드시 필요하며 목표마케팅이나 일대일마케팅에 활용 동시발생 매트릭스
	데이터군집화 Clustering	고객레코드들을 유사한 특성을 지닌 몇개의 소그룹으로 분할하는 작업. 작업의 특성이 분류 규칙과 유사하나 분석대상 데이터에 결과값이 없으며 판촉활동이나 이벤트 대상을 선정하는데 활용 K-Means Clustering



제1절 데이터 마이닝의 개요

데이터 마이닝 추진단계

- [1단계] 목적 설정
 - 데이터 마이닝을 통해 무엇을 왜 하는지 명확한 목적을 설정. 전문가가 참여해 목적에 따라 사용할 모델과 필요할 데이터를 정의
- [2단계] 데이터 준비
 - 고객정보, 거래정보, 상품정보, 마스터 정보, 웹로그 데이터, 소셜 네트워크 데이터 등 다양한 데이터를 활용
 - IT부서와 사전에 협의하고 일정을 조율하여 데이터 접근 부하에 유의하여야 하며, 필요시 다른 서버에 저장하여 운영에 지장이 없도록 데이터 준비
 - 데이터 정재를 통해 데이터 품질을 보장하고, 필요시 데이터 보강하여 충분한 데이터를 확보
- [3단계] 가공
 - 모델링 목적에 따라 목적 변수 정의. 필요한 데이터를 데이터 마이닝 소프트웨어에 적용할 수 있는 형식으로 가공
- [4단계] 기법 적용
 - 1단계에서 명확한 목적에 맞게 데이터 마이닝 기법을 적용하여 정보 추출
- [5단계] 검증
 - 데이터 마이닝으로 추출된 정보를 검증. 테스트 데이터와 과거 데이터를 활용하여 최적의 모델 선정
 - 검증이 완료되면 IT 부서와 협의해 상시 데이터마이닝 결과를 업무에 적용하고 보고서를 작성하여 추가 수익과 투자 대비 성과(ROI)등으로 기대효과를 전파



제1절 데이터 마이닝의 개요

데이터 마이닝을 위한 데이터 분할

● 개요

- 모델 평가용 테스트 데이터와 구축용 데이터로 분할하여 구축용 데이터로 모델을 생성하고 테스트 데이터로 모델이 얼마나 적합한지 판단

● 데이터 분할

- 구축용(50%): 추정용, 훈련용 데이터라고도 불리며 데이터 마이닝 모델을 만드는 데 활용
- 검정용(30%): 구축된 모델의 과대추정 또는 과소추정을 미세 조정하는데 활용
- 시험용(20%): 테스트 데이터나 과거 데이터를 활용하여 모델의 성능을 검증하는데 활용
- 데이터 양이 충분하지 않거나 입력 변수에 대한 설명이 충분한 경우
 - 1) 홀드 아웃 방법: 주어진 데이터를 랜덤 하게 두 개의 데이터를 주어진 데이터를 랜덤 하게 두 개의 데이터로 구분하여 사용하는 방법. 주로 학습용과 시험용으로 분리하여 사용
 - 2) 교차 확인방법: 주어진 데이터를 k개의 하부 집단으로 구분하여 k-1개의 집단을 학습용으로 나머지 하부 집단으로 검증용으로 설정하여 학습한다. k번 반복 측정한 결과를 평균 낸 값을 최종 값으로 사용. 주로 10-fold 교차 분석을 많이 사용



제1절 데이터 마이닝의 개요

성과분석

● 오분류에 대한 추정치

		Condition (실제)		
		Negative	Positive	
Prediction (예측)	Positive	TP (True Positive)	FP (False Positive)	Positive predictive Value =TP/(TP+FP)
	Negative	FN (False Negative)	TN (True Negative)	Negative predictive Value =TN/(TN+FN)
		민감도(sensitivity,TPR) =TP/(TP+FN)	특이도(Specificity,TNR) =TN/(TN+FP)	



제1절 데이터 마이닝의 개요

성과분석

Confusion matrix		예측값	
		TRUE	FALSE
실제값	TRUE	40	60
	FALSE	60	40

40 (TP)	60 (FN) Type II Error
60 (FP) Type I Error	40 (TN)

지표	설명
Precision	$TP / (TP + FP)$
Recall, Sensitivity	$TP / (TP + FN)$
F1	$2 * (Precision * Recall) / (Precision + Recall)$
Specificity	$TN / (TN + FP)$
FP Rate	$FP / (FP + TN), 1 - Specificity$
Error Rate	$(FP + FN) / (TP + FP + FN + TN)$
Accuracy	$(TP + TN) / (TP + FP + FN + TN)$

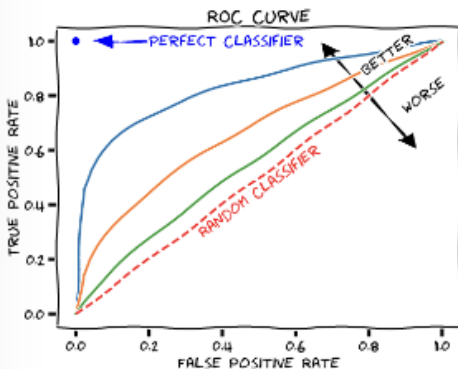


제1절 데이터 마이닝의 개요

성과분석

● ROC Curve(Receiver Operating Characteristic Curve)

- X축은 **FP Rate(1-Specificity)**, Y축은 민감도(**Sensitivity**)를 나타내 이 두 평가 값의 관계로 모델을 평가함.
- ROC 그래프의 밑부분의 면적(**AUC, Area Under the Curve**)이 넓을수록 좋은 모형으로 평가함.



● Recall, Sensitivity

- 실제 값이 True인 것에 대해 예측 값이 True로 된 비율
- $TP / (TP + FN)$

● FP Rate

- $FP / (FP + TN)$, $1 - \text{Specificity}$
- 실제가 False 인데 예측이 True로 된 비율 (1종 오류 비율)

• X축 : False Positive Rate ($1 - \text{Specificity}$)

• Y축 : True Positive Rate (Sensitivity)

제1절 데이터 마이닝의 개요

R을 이용한 ROC실습 코드

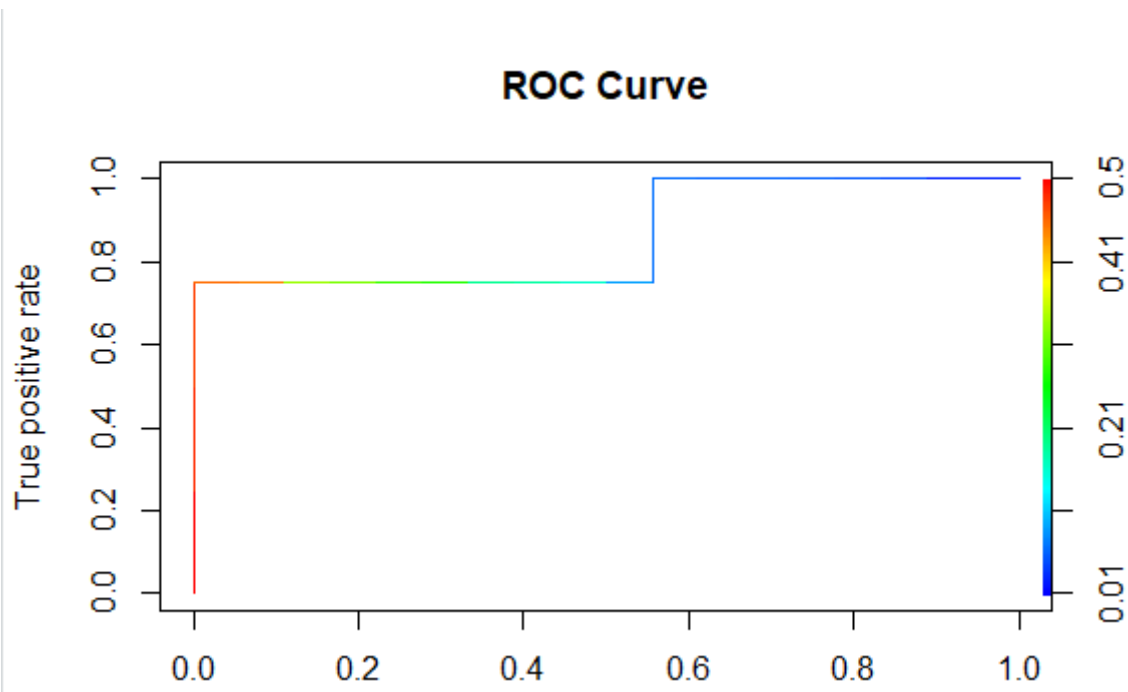
```
>> library(rpart)
>> library(party)
>> library(ROCR)
>> x <- kyphosis[sample(1:nrow(kyphosis), nrow(kyphosis), replace=F),]
>> x.train <- kyphosis[1:floor(nrow(x)*0.75),]
>> x <- kyphosis[sample(1:nrow(kyphosis), nrow(kyphosis), replace=F),]
>> x.train <- kyphosis[1:floor(nrow(x)*0.75),]
>> x.evaluate <- kyphosis[floor(nrow(x)*0.75):nrow(x),]
>> x.model <- cforest(Kyphosis~Age+Number+Start, data = x.train)
There were 50 or more warnings (use warnings() to see the first 50)
>> x.evaluate$prediction <- predict(x.model, newdata=x.evaluate)
>> x.evaluate$correct <- x.evaluate$prediction == x.evaluate$Kyphosis
>> print(paste("% of predicted classification corect",mean(x.evaluate$correct)))
[1] "% of predicted classification corect 0.818181818181818"
>> x.evaluate$probalilities <- 1- unlist(treeresponse(x.model, newdata=x.evaluate), use.names =
F)[seq(1,nrow(x.evaluate)*2,2)]
>> pred <- prediction(x.evaluate$probalilities, x.evaluate$Kyphosis)
>> perf <- performance(pred, "tpr", "fpr")
>> plot(perf, main="ROC Curve", colorize=T)
```



제1절 데이터 마이닝의 개요

R을 이용한 ROC실습 코드

>>

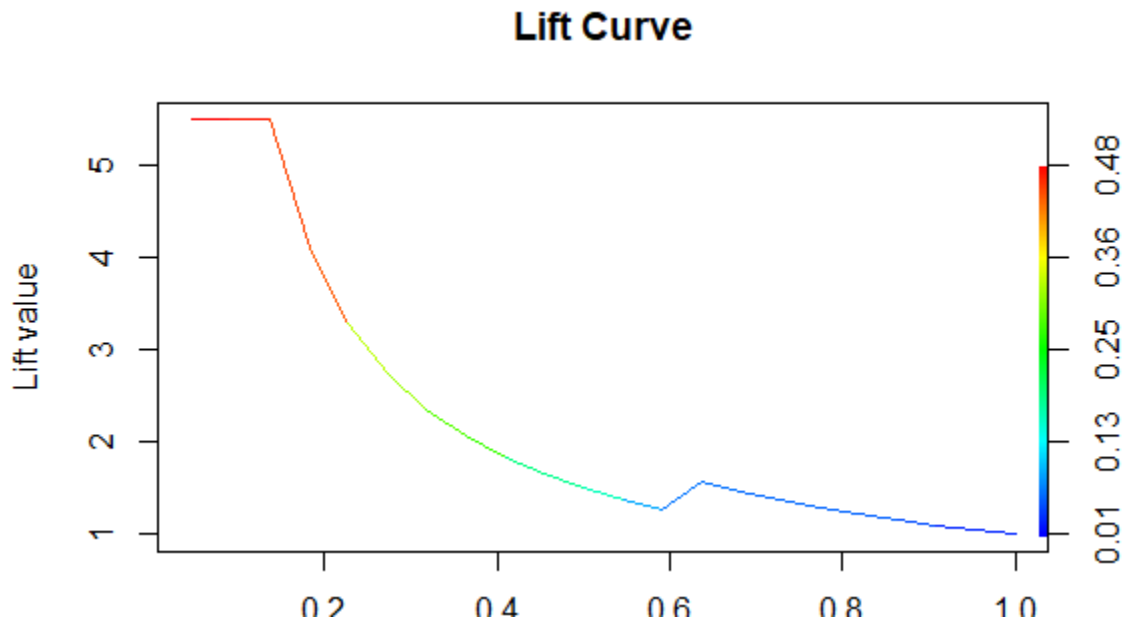




제1절 데이터 마이닝의 개요

R을 이용한 ROC실습 코드

```
>> perf <- performance(pred, "lift", "rpp")  
>> plot(perf, main="Lift Curve", colorize=T)
```



제1절 데이터 마이닝의 개요

성과분석

- 이익도표
 - 분류 모형의 성능을 평가하기 위한 척도로, 분류된 관측치에 대해 얼마나 예측이 잘 이루어졌는지 나타내기 위해 임의로 나눈 각 등급별로 반응 검출율, 반응률, 리프트 등의 정보를 산출하여 나타내는 도표
 - 이익 도표의 각 등급은 예측 확률에 따라 매겨진 순위이기 때문에 상위등급에서는 더 높은 반응률을 보이는 것이 좋은 모형이라고 평가할 수 있다.
 - 등급별 향상도가 급격하게 변동할수록 좋은 모형이라고 할 수 있고, 각 등급별로 향상도가 들쭉날쭉하면 좋은 모형이라고 볼 수 없다.
- $\text{Lift(향상도)} = \text{반응률} / \text{기본 향상도} \quad * \text{좋은 모델이면 Lift가 빠른 속도로 감소해야 한다}$



제1절 데이터 마이닝의 개요

과적합-과대 적합, 과소 적합

- 과적합-과대적합(Overfitting): 모형이 학습용 데이터를 과하게 학습하여 학습 데이터에 대해서는 높은 정확도를 나타내지만 테스트 데이터 혹은 다른 데이터에 적용할 때는 성능이 떨어지는 현상
- 과소 적합(Underfitting): 모형이 너무 단순하여 데이터 속 내재되어 있는 패턴이나 규칙을 제대로 학습하지 못하는 경우

학습목표

- 분류분석의 개요와 기법을 이해한다.
- 의사결정나무 방법론을 이해한다.
- 의사결정나무 방법론의 종류별 특성을 이해한다.
- R프로그램을 통해 의사결정나무 분석을 구현할 수 있다.

눈높이 체크

- 분류분석에 대해 들어 보신적이 있나요?
- 의사결정나무 분석 방법을 알고 계신가요?
- 의사결정나무의 종류를 알고 계신가요?

제2절 분류분석

분류 분석과 예측

	분류분석	예측분석
정의	데이터가 어떤 그룹에 속하는지 예측하는데 사용하는 기법. 클러스터링과 유사하지만 분류분석은 각 그룹이 정의되어 있음. 교사학습에 해당하는 예측기법	시계열분석처럼 시간에 따른 값 두 개만을 이용해 앞으로의 매출 또는 온도 등을 예측. 모델링을 하는 입력데이터가 어떤 것인지에 따라 특성이 다름. 여러 개의 다양한 설명변수(독립변수)가 아닌 한 개의 설명변수로 생각하면 된다.
공통점	레코드의 특정 속성의 값을 미리 알아맞힘	
차이점	레코드(튜플)의 범주형 속성 값을 알아맞힘	레코드(튜플)의 연속형 속성 값을 알아맞힘
예	학생들의 국,영,수 점수를 통해 내신을 맞힘 카드 회사에서 회원들의 가입 정보를 통해 1년 후 신용등급을 알아맞힘	학생들의 여러 가지 정보를 입력하여 수능점수를 맞힘 카드회사 회원들의 가입정보를 통해 연 매출액을 맞힘

분류 분석과 예측

- 분류 모델링
 - 신용평가모형, 사기 방지 모형, 이탈 모형, 고객 세분화
- 분류 기법
 - 회귀분석, 로지스틱 회귀분석 / 의사결정나무, CART, C5.0 / 베이
지안 분류, Naive Bayesian / 인공신경망 / 지지도벡터기계 / k 최
근접이웃 / 규칙 기반의 분류와 사례기반 추론

New York 대기 온도에 영향을 미치는 변수 알아보기

- ❖ `airquality`는 R에서 `datasets` 패키지에서 제공되는, New York의 대기에 관한 질을 측정한 데이터셋으로, 153개의 관측치와 6 개의 변수로 구성되어 있습니다.
- ❖ 주요 변수로는
 - `Ozone`(오존 수치)
 - `Solar.R`(태양광)
 - `Wind`(바람)
 - `Temp`(온도)
 - `Month`(측정 월)
 - `Day`(측정 날짜)가 있습니다.
- ❖ 수행 절차는 다음과 같습니다.
 - 단계 1: `party` 패키지 설치
 - 단계 2: `airquality` 데이터 셋 로딩
 - 단계 3: `formula` 생성
 - 단계 4: 분류모델 생성 - `formula`를 이용하여 분류모델 생성
 - 단계 5: 분류분석 결과

New York 대기 온도에 영향을 미치는 변수 알아보기

```
>> install.packages("party")
```

```
Installing package into '/home/k8s/R/x86_64-pc-linux-gnu-library/3.6'
```

```
(as 'lib' is unspecified)
```

```
also installing the dependencies 'TH.data', 'libcoin', 'matrixStats', 'multcomp', 'modeltools', 'strucchange', 'coin', 'zoo', 'sandwich'
```

```
URL 'https://cloud.r-project.org/src/contrib/TH.data_1.1-1.tar.gz'을 시도합니다
```

```
Content type 'application/x-gzip' length 8612708 bytes (8.2 MB)
```

```
=====
```

```
downloaded 8.2 MB
```

```
...
```

```
* DONE (party)
```

```
The downloaded source packages are in
```

```
'/tmp/Rtmp00BmxT/downloaded_packages'
```

```
>> library(party)
```

```
필요한 패키지를 로딩중입니다: grid
```

```
필요한 패키지를 로딩중입니다: mvtnorm
```

```
필요한 패키지를 로딩중입니다: modeltools
```

```
필요한 패키지를 로딩중입니다: stats4
```

```
필요한 패키지를 로딩중입니다: strucchange
```

```
필요한 패키지를 로딩중입니다: zoo
```

```
다음의 패키지를 부착합니다: 'zoo'
```

```
The following objects are masked from 'package:base':
```

```
as.Date, as.Date.numeric
```

```
필요한 패키지를 로딩중입니다: sandwich
```

```
>>>
```


New York 대기 온도에 영향을 미치는 변수 알아보기

```
>> #install.packages("datasets")  
>> library(datasets)  
>> str(airquality)
```

```
'data.frame': 153 obs. of 6 variables:  
 $ Ozone   : int  41 36 12 18 NA 28 23 19 8 NA ...  
 $ Solar.R: int  190 118 149 313 NA NA 299 99 19 194 ...  
 $ Wind    : num  7.4 8 12.6 11.5 14.3 14.9 8.6 13.8 20.1 8.6 ...  
 $ Temp    : int  67 72 74 62 56 66 65 59 61 69 ...  
 $ Month    : int  5 5 5 5 5 5 5 5 5 5 ...  
 $ Day      : int  1 2 3 4 5 6 7 8 9 10 ...
```

```
>>
```

153개의 관측치와 6 개의 변수

New York 대기 온도에 영향을 미치는 변수 알아보기

```
>> formula <- Temp ~ Solar.R + Wind + Ozone
```

온도에 영향을 미치는 변수를 알아보기 위해서 Temp(온도) 변수를 반응변수(종속변수)로 지정하고, Ozone(오존 수치), Solar.R(태양광), Wind(바람)는 설명변수(독립변수)로 지정하여 식을 완성합니다.

New York 대기 온도에 영향을 미치는 변수 알아보기

```
>> air_ctree <- ctree(formula, data = airquality)
>> air_ctree
```

Conditional inference tree with 5 terminal nodes

Response: Temp

Inputs: Solar.R, Wind, Ozone

Number of observations: 153

```
1) Ozone <= 37; criterion = 1, statistic = 56.086
  2) Wind <= 15.5; criterion = 0.993, statistic = 9.387
    3) Ozone <= 19; criterion = 0.964, statistic = 6.299
      4)* weights = 29
      3) Ozone >> 19
        5)* weights = 69
    2) Wind >> 15.5
      6)* weights = 7
  1) Ozone >> 37
    7) Ozone <= 65; criterion = 0.971, statistic = 6.691
      8)* weights = 22
    7) Ozone >> 65
      9)* weights = 7
```

>>

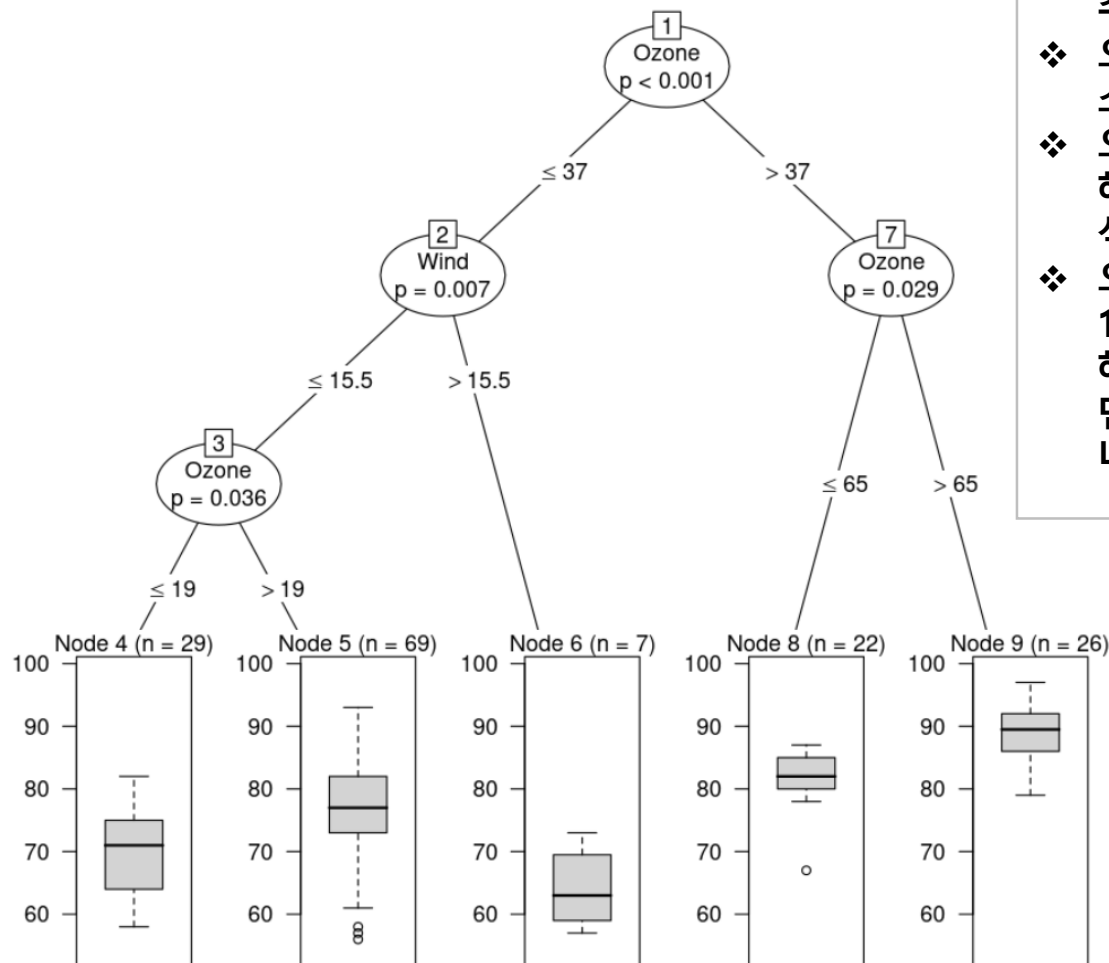
예 >> Ozone <= 65; criterion = 0.971, statistic = 6.691

- ① 7)은 반응변수에 대해서 설명변수가 영향을 미치는 중요 변수의 척도를 나타내는 수치입니다. 작을 수록 영향을 미치는 정도가 높으며, 순서는 분기되는 순서를 의미합니다.
- ② 의사결정 트리의 노드명입니다. 노드명 대신 *이 오면 해당 노드가 마지막 노드임을 의미하며, 노드명 뒤 해당 변수의 임계값이 조건식으로 옵니다.
- ③ 노드의 분기 기준이 되는 수치입니다. criterion = 0.971이면 유의확률 $p=0.029(1-0.971)$ 이 됩니다. 유의확률 p 는 의사결정 트리에서 확인이 가능합니다.
- ④ 반응변수의 통계량입니다. 마지막 노드이거나 또 다른 분기 기준이 있는 경우에는 세번째와 네번째 수치는 표시되지 않습니다.

제2절 분류분석

New York 대기 온도에 영향을 미치는 변수 알아보기

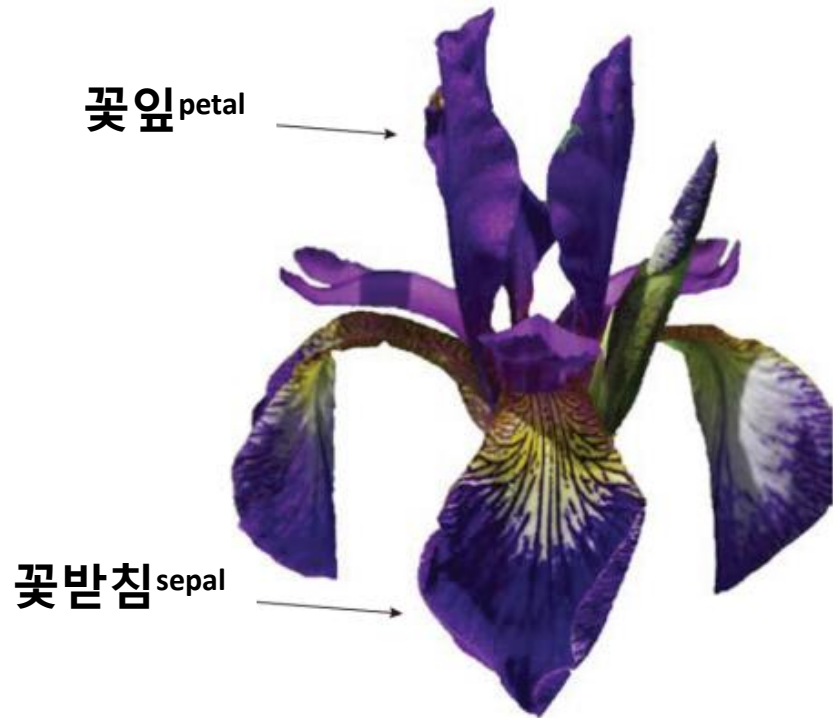
```
>> plot(air_ctree)
```



- ❖ 온도가 가장 큰 영향을 미치는 첫번째 변수는 오존 수치이고, 두번째 영향 변수는 바람으로 나타났습니다.
- ❖ 오존량이 감소하면 대체로 온도가 감소하는 경향을 보이고 있습니다.
- ❖ 오존량이 37이하이면 바람의 양에 의해서 온도에 영향을 미치는 것으로 분석되었습니다.
- ❖ 오존량이 37이하이면서 바람의 양이 15.5이상이면 평균 온도가 63 정도에 해당하지만, 바람의 양이 15.5이하이면 평균 온도가 70이상으로 나타났습니다.

Iris 변수 알아보기

- 붓꽃의 꽃잎^{petal}과 꽃받침^{sepal}의 폭과 길이를 센티미터 단위로 측정해 놓은 데이터입니다.
- '붓꽃(Iris)'은 프랑스의 국화



Iris 변수 알아보기

Iris plants dataset

****Data Set Characteristics:****

:Number of Instances: 150 (50 in each of three classes)
:Number of Attributes: 4 numeric, predictive attributes and the class
:Attribute Information:
- sepal length in cm
- sepal width in cm
- petal length in cm
- petal width in cm
- class:
- Iris-Setosa
- Iris-Versicolour
- Iris-Virginica

:Summary Statistics:

	Min	Max	Mean	SD	Class Correlation
sepal length:	4.3	7.9	5.84	0.83	0.7826
sepal width:	2.0	4.4	3.05	0.43	-0.4194
petal length:	1.0	6.9	3.76	1.76	0.9490 (high!)
petal width:	0.1	2.5	1.20	0.76	0.9565 (high!)

:Missing Attribute Values: None

:Class Distribution: 33.3% for each of 3 classes.

:Creator: R.A. Fisher

:Donor: Michael Marshall (MARSHALL@PLU@io.arc.nasa.gov)

:Date: July, 1988

...

그림 3-2 iris의 세 가지 품종(왼쪽부터 Setosa, Versicolor, Virginica)





제2절 분류분석

Iris 변수 알아보기

```
>> library(party)
>> #set.seed(1234)
>> idx <- sample(1:nrow(iris), nrow(iris) * 0.7)
>> train <- iris[idx, ]
>> test <- iris[-idx, ]

>> formula <- Species ~ Sepal.Length + Sepal.Width + Petal.Length + Petal.Width
```

Iris 변수 알아보기

```
> iris_ctree <- ctree(formula, data = train)
> iris_ctree
```

Conditional inference tree with 4 terminal nodes

Response: Species

Inputs: Sepal.Length, Sepal.Width, Petal.Length, Petal.Width

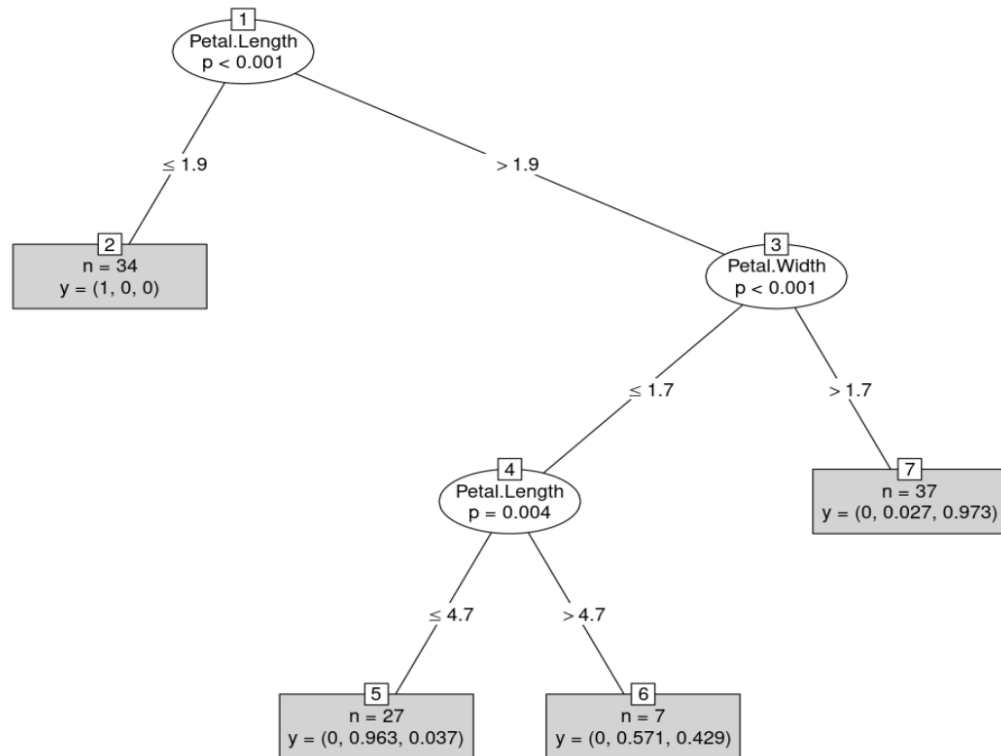
Number of observations: 105

```
1) Petal.Length <= 1.9; criterion = 1, statistic = 97.644
  2)* weights = 34
1) Petal.Length > 1.9
  3) Petal.Width <= 1.7; criterion = 1, statistic = 48.725
    4) Petal.Length <= 4.7; criterion = 0.996, statistic = 10.682
      5)* weights = 27
    4) Petal.Length > 4.7
      6)* weights = 7
  3) Petal.Width > 1.7
    7)* weights = 37
```

❖ 분류모델 결과로 가장 중요한 변수는 Petal.Length 와 Petal.Width 로 나타났습니다.

Iris 변수 알아보기

```
plot(iris_ctree, type = "simple")
```

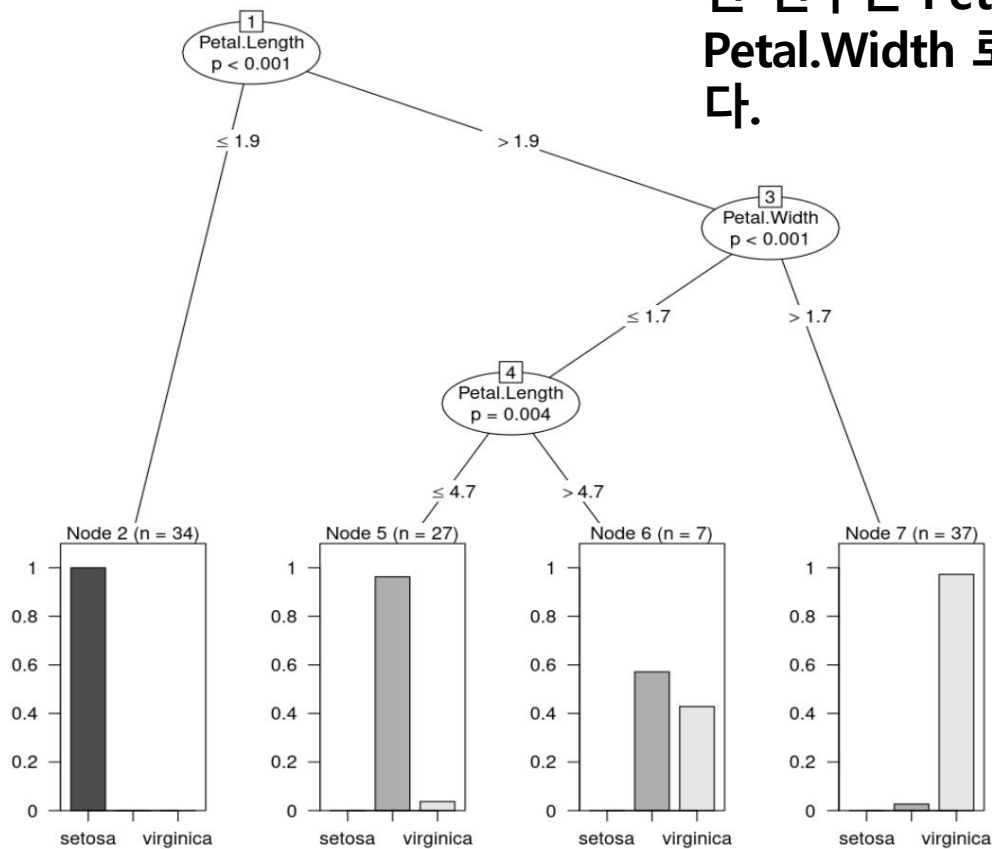


제2절 분류분석

Iris 변수 알아보기

`plot(iris_ctree)`

분류모델 결과로 가장 중요한 변수는 Petal.Length와 Petal.Width로 나타났습니다.



제2절 분류분석

Iris 변수 알아보기

분류모델 평가-모델의 예측치 생성과 혼돈 매트릭스 생성

```
> pred <- predict(iris_ctree, test)
> table(pred, test$Species)
```

```
pred      setosa versicolor virginica
setosa      12         0         0
versicolor  0         16         1
virginica   0          1         15
```

```
> (12 + 16 + 15) / nrow(test)
```

```
[1] 0.9555556
```

```
>
```

❖ 혼동행렬(Confusion Matrix)

예측값(Prediction)	실제값(Reference)	
	Y	N
Y	True Positive(TP)	False Positive(FP)
N	False Negative(FN)	True Negative(TN)

Accuracy

$$\frac{TP + TN}{TP + FP + FN + TN}$$

전체 예측에서(예측이 Y이든 N이든 무관하게) 옳은 예측의 비율

Iris 변수 알아보기

```
# caret 패키지 설치install.packages("caret")
```

```
> library(caret)
```

```
필요한 패키지를 로딩중입니다: ggplot2
```

```
필요한 패키지를 로딩중입니다: lattice
```

```
> confusionMatrix(pred, test$Species)
```

Confusion Matrix and Statistics

	Reference		
Prediction	setosa	versicolor	virginica
setosa	12	0	0
versicolor	0	16	1
virginica	0	1	15

Overall Statistics

Accuracy : 0.9556

95% CI : (0.8485, 0.9946)

No Information Rate : 0.3778

P-Value [Acc > NIR] : 2.61e-16

Kappa : 0.9326

McNemar's Test P-Value : NA

Statistics by Class:

	Class: setosa	Class: versicolor	Class: virginica
Sensitivity	1.0000	0.9412	0.9375
Specificity	1.0000	0.9643	0.9655
Pos Pred Value	1.0000	0.9412	0.9375
Neg Pred Value	1.0000	0.9643	0.9655
Prevalence	0.2667	0.3778	0.3556
Detection Rate	0.2667	0.3556	0.3333
Detection Prevalence	0.2667	0.3778	0.3556
Balanced Accuracy	1.0000	0.9527	0.9515

>

로지스틱 회귀 분석

● 개요

- 반응 변수가 범주형인 경우에 적용되는 회귀분석모형
- 새로운 설명변수(예측 변수)가 주어질 때 반응 변수의 각 범주에 속할 확률이 얼마인지를 추정(예측모형)
- 추정 확률을 기준치에 따라 분류하는 목적(분류 모형)으로 활용된다.
- 이때 모형의 적합을 통해 추정된 확률을 사후 확률이라 함
- 설명변수가 한 개인 경우 해당 회귀 계수의 부호에 따라 S자 모양($S > 0$) 또는 역 S자 모양($S < 0$) 모양을 가진다.
- 표준 로지스틱 분포의 누적 함수 분포로 성공의 확률을 추정
- glm() 함수 사용

● 로지스틱 회귀모형이 선호되는 이유

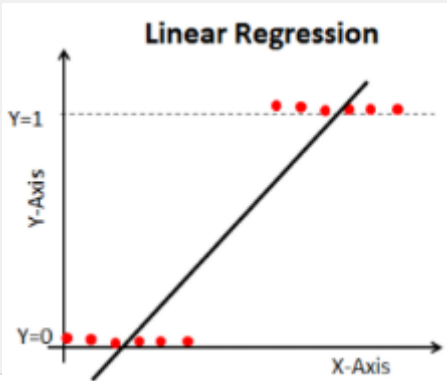
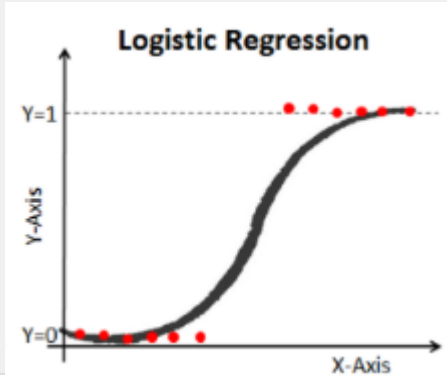
- 독립변수에 대해 어떤 가정도 필요하지 않음
- 독립변수가 연속형 및 이산형 모두 가능하기 때문에 판별분석보다 선호됨

● 오즈비(Odds Ratio)

- 오즈(odds): 성공할 확률이 실패할 확률의 몇 배인지를 나타내는 확률
- 오즈비(odds ratio) 오즈의 비율
- 성공 가능성이 높은 경우 1보다 크고 실패 가능성이 높은 경우 1보다 작다.

로지스틱 회귀 분석

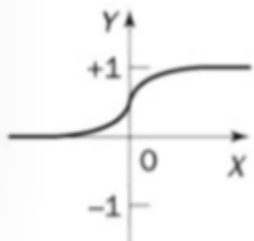
● 선형 회귀 분석 vs 로지스틱 회귀 분석

	선형회귀분석	로지스틱 회귀분석
종속변수	연속형 변수	(0,1)
계수 추정법	최소제곱법	최대우도 추정법
모형검정	F-검정, T-검정	카이제곱 검정
그래프	 <p>Linear Regression</p>	 <p>Logistic Regression</p>

제2절 분류분석

로지스틱 회귀 분석

- 승산(odds) = 성공률 / 실패율, $\frac{P_i}{1 - P_i}$, P_i = 성공률
 - 성공이 일어날 가능성이 높은 경우는 1.0 보다 큰 값
 - 실패가 발생할 가능성이 높은 경우는 1.0 보다 작은 값
 - 로지스틱의 회귀 계수, 확률에 대해 $0 \sim \infty$ 로 변환한 값
- log odds, logit transformation = log(odds)
 - 선형화(Linearization)의 하나로, odds 값에 log를 취하여 값의 범위를 전체 실수 범위로 확장함.
- sigmoid 함수
 - Logistic** 함수라 불리기도 하며, log_odds 값을 연속형 0~1 사이의 값으로 바꾸는 함수
 - 비선형 값을 얻기 위해 사용
 - $$y^{\text{sigmoid}} = \frac{1}{1 + e^{-x}}$$



sigmoid 함수

```
2 prob <- 0.8
3 odds <- prob / (1-prob)
4 log_odds <- log(odds)
5 r <- 1 / (1 + exp(-log_odds))
```

Values

log_odds	1.38629436111989
odds	4
prob	0.8
r	0.8

→ 성공이 일어날 가능성이 실패보다 4배

로지스틱 회귀 분석

```
8 prob_a <- 0.5
9 prob_b <- 0.2
10 odds_a <- prob_a / (1-prob_a)
11 odds_b <- prob_b / (1-prob_b)
12 odds_ratio <- odds_a / odds_b
```

Values	
odds_a	1
odds_b	0.25
odds_ratio	4

- 승산비(Odds Ratio) = 관심있는 사건이 발생할 상대 비율, $x = 1$ 일 때, $y = 1$ 이 되는 상대적 비율
 - $\text{odds}_a / \text{odds}_b = \exp(\text{coef}) = \exp(5.140336) = 170.7731385$
 - 로지스틱 회귀에서 $\exp(x_1)$ 의 의미 (단, x_1 : 회귀 계수)
 - 나머지 변수가 주어질 때 x_1 이 한 단위 증가할 때마다 성공($Y=1$)의 odds가 몇 배 증가하는지를 나타냄.

```
2 data(iris)
3 a <- subset(iris, Species=='setosa' | Species=='versicolor')
4 a$Species<-factor(a$Species)
5 b <- glm(Species~Sepal.Length, data=a, family = binomial)
```

```
> coef(b)
(Intercept) Sepal.Length
-27.831451    5.140336

> exp(coef(b))['Sepal.Length']
Sepal.Length
170.7732
```

- $Y=1$ 은 versicolor 일 경우, Sepal.Length 가 한 단위 증가하면 Versicolor 일 odds가 170배 증가를 의미함.

로지스틱 회귀 분석

- 최대 우도 추정법(MLE)
 - 모수의 미지의 θ 인 확률분포에서 뽑은 표본(관측치) x 들을 바탕으로 θ 를 추정하는 기법
 - 표본의 수가 클수록 최대 우도 추정법은 안정적
 - 우도: 이미 주어진 표본 x 들에 비추어봤을 때 모집단의 모수 θ 에 대한 추정이 그럴듯한 정도를 말한다
 - 우도 $L(x|\theta)$ 은 θ 가 존재되었을 때 표본 x 가 등장할 확률인 $P(x|\theta)$ 에 비례
- 카이제곱 검정
 - 분류 조합에 따라 특정값에 유효한 차이가 발생하는 지를 검정하는 것으로 명목 척도로 측정된 두 속성이 서로 관련되어 있는지 분석하고 싶을 때 사용하는 통계 분석방법
 - 독립변수와 종속변수가 모두 명목 척도일 경우 적합한 통계 기법

mtcars 데이터 단순회귀분석

- ❖ `lm()` 함수는 formula와 data를 넣어야 합니다. formula는 '종속변수 ~ 독립변수'의 형태를 갖습니다.
- ❖ 실린더의 수(cyl)와 마력(hp)의 선형관계를 알아보기 위해, 마력을 종속변수, 실린더의 수를 독립변수로 설정합니다.

```
> lm_fit <- lm(hp ~ cyl, data=mtcars)
> summary(lm_fit)
```

```
Call:
lm(formula = hp ~ cyl, data = mtcars)
```

```
Residuals:
    Min     1Q  Median     3Q    Max
-54.61 -25.99 -11.28  21.51 130.39
```

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  -51.054    24.982  -2.044  0.0499 *
cyl           31.958     3.884   8.229 3.48e-09 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Residual standard error: 38.62 on 30 degrees of freedom
```

```
Multiple R-squared:  0.693,
```

Adjusted R-squared: 0.6827

```
F-statistic: 67.71 on 1 and 30 DF, p-value: 3.478e-09
```

- Adjusted R-squared: 0.6827은 독립변수가 종속변수를 얼마나 설명해 줄 수 있는지를 의미 설명력을 말합니다.
- 즉, 실린더의 수가 마력을 68.2% 설명할 수 있다고 할 수 있습니다.



제2절 분류분석

mtcars 데이터 단순회귀분석

```
> lm_fit <- lm(hp ~ cyl, data=mtcars)
> summary(lm_fit)
```

Call:
lm(formula = hp ~ cyl, data = mtcars)

Residuals:
Min 1Q Median 3Q Max
-54.61 -25.99 -11.28 21.51 130.39

Coefficients:
Estimate Std. Error t value Pr(>|t|)
(Intercept) -51.054 24.982 -2.044 0.0499 *
cyl 31.958 3.884 8.229 3.48e-09 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 38.62 on 30 degrees of freedom
Multiple R-squared: 0.693, Adjusted R-squared: 0.6827

F-statistic: 67.71 on 1 and 30 DF, **p-value: 3.478e-09**

- p-value가 3.478e-09입니다. p-value가 0.05보다 작기 때문에 기울기가 0이 아니고 모형이 유의하며, 실린더의 수가 마력에 영향을 미친다고 할 수 있습니다.

>

mtcars 데이터 단순회귀분석

```
> lm_fit <- lm(hp ~ cyl, data=mtcars)
> summary(lm_fit)
```

Call:
lm(formula = hp ~ cyl, data = mtcars)

Residuals:
Min 1Q Median 3Q Max
-54.61 -25.99 -11.28 21.51 130.39

Coefficients:
Estimate Std. Error t value Pr(>|t|)
(Intercept) **-51.054** 24.982 -2.044 0.0499 *
cyl **31.958** 3.884 8.229 3.48e-09 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 38.62 on 30 degrees of freedom
Multiple R-squared: 0.693, Adjusted R-squared: 0.6827
F-statistic: 67.71 on 1 and 30 DF, p-value: 3.478e-09

- 회귀식은 아래와 같이 나타낼 수 있습니다.

$$hp = -51.054 + 31.958 \times cyl$$

- 실린더의 수가 하나 증가할수록 마력은 31.958 증가함을 의미합니다.

>

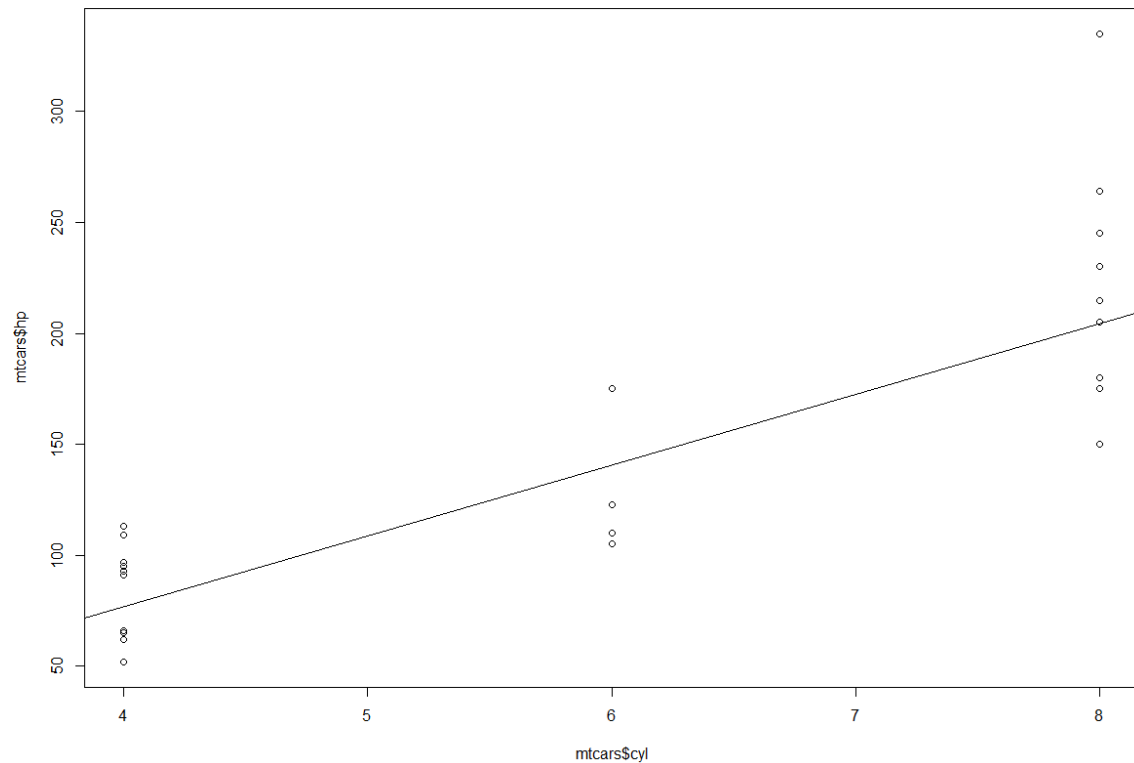


제2절 분류분석

mtcars 데이터 단순회귀분석

❖ 회귀직선

```
> plot(mtcars$cyl, mtcars$hp) # scatter plot  
> abline(lm_fit) # line
```



mtcars 데이터 단순회귀분석

❖ 회귀직선

- 실린더가 0개가 아닐뿐더러 마력이 음수, 해결하기 위해서 회귀선이 0점을 지나도록 추가 작업을 해줍니다.
- formula에 +0을 추가합니다.

```
> lm_fit_0 <- lm(hp ~ cyl+0, data=mtcars)
> summary(lm_fit_0)
```

Call:
lm(formula = hp ~ cyl + 0, data = mtcars)

Residuals:
Min 1Q Median 3Q Max
-45.29 -33.04 -14.59 12.71 140.41

Coefficients:
Estimate Std. Error t value Pr(>|t|)
cyl 24.323 1.114 21.83 <2e-16 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 40.55 on 31 degrees of freedom

Multiple R-squared: 0.9389, **Adjusted R-squared: 0.9369**

F-statistic: 476.4 on 1 and 31 DF, **p-value: < 2.2e-16**

- p-value는 <2.2e-16으로 여전히 모형이 유의하고, 설명력은 93.7%로 더 높아졌습니다.



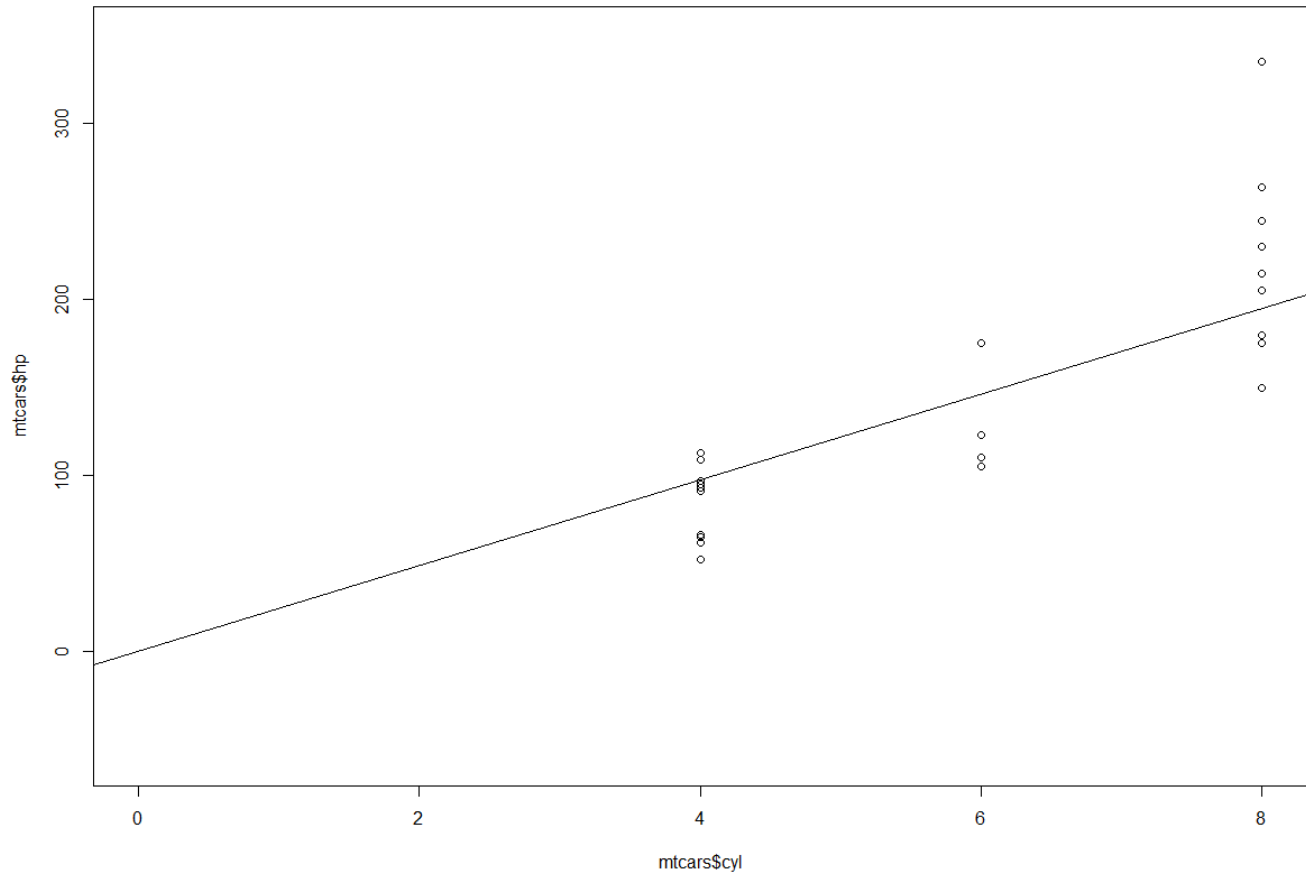
제2절 분류분석

mtcars 데이터 단순회귀분석

❖ 회귀직선

```
> plot(mtcars$cyl, mtcars$hp, xlim=c(0,8), ylim=c(-60, 350))
```

```
> abline(lm_fit_0)
```



R을 이용한 로지스틱 회귀분석 실습 코드

```
>> a <- iris[iris$Species=="setosa" | iris$Species=="versicolor",]  
>> b <- glm(Species ~ Sepal.Length, data=a, family = binomial)  
>> summary(b)
```

Call:

```
glm(formula = Species ~ Sepal.Length, family = binomial, data = a)
```

Deviance Residuals:

Min	1Q	Median	3Q	Max
-2.05501	-0.47395	-0.02829	0.39788	2.32915

Coefficients:

	Estimate	Std. Error	z value	Pr(> z)
(Intercept)	-27.831	5.434	-5.122	3.02e-07 ***
Sepal.Length	5.140	1.007	5.107	3.28e-07 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 138.629 on 99 degrees of freedom
Residual deviance: 64.211 on 98 degrees of freedom
AIC: 68.211

Number of Fisher Scoring iterations: 6

```
>>
```


의사결정 나무

● 개요

- 의사결정 나무는 분류 함수를 의사결정 규칙으로 이뤄진 나무 모양으로 그리는 기법
- 나무 구조는 연속적으로 발생하는 의사결정 문제를 시각화해 의사결정이 이뤄지는 시점&성과를 한눈에 볼 수 있다.
- 계산 결과가 의사결정 나무에 직접 나타나기 때문에 해석이 간편하다.
- 의사결정나무는 주어진 입력값에 대하여 출력 값을 예측하는 모형으로 분류 나무와 회귀 나무 모형이 있다.

● 의사결정 나무 구성요소

- 뿌리마디: 시작되는 마디로 전체 자료 포함
- 자식마디: 하나의 마디로부터 분리되어 나간 2개 이상의 마디들
- 부모마디: 주어진 마디의 상위 마디
- 끝마디: 자식이 없는 마디
- 중간마디: 부모와 자식마디가 모두 있는 마디
- 가지: 뿌리 마디로부터 끝마디까지 연결된 마디들
- 깊이: 뿌리 마디부터 끝마디까지 중간 마디의 수

의사결정 나무

● 예측력과 해석력

- 기대 집단의 사람들 중 가장 많은 반응을 보일 때 고객 유치방안을 예측하고자 하는 경우 예측력에 치중한다.
- 신용평가에서 심사 결과 부적격 판정이 나온 경우 고객에게 부적격 이유를 설명해야 하므로 해석력에 치중한다.

● 의사결정 활용

- 세분화: 데이터를 비슷한 특성을 갖는 몇 개의 그룹으로 분할해 그룹별 특성을 발견
- 분류: 여러 예측 변수들에 근거해 관측 개체의 목표 변수 범주를 몇 개의 등급으로 분류하고자 하는 경우에 사용
- 예측: 자료에서 규칙을 찾아내고 이를 이용해 미래의 사건을 예측하고자 하는 경우
- 차원 축소 및 변수 선택: 매우 많은 수의 예측변수 중에서 목표 변수에 큰 영향을 미치는 변수들을 골라내고자 하는 경우에 사용
- 교호 작용효과의 파악: 여러 개의 예측 변수들을 결합해 목표 변수에 작용하는 규칙을 파악하고자 하는 경우 범주형 변수의 범주를 소수의 몇 개로 병합하거나 연속형 목표 변수를 몇 개의 등급으로 이산화하고자 하는 경우

의사결정 나무

● 의사결정 나무 분석과정

- 1) 성장단계: 각 마디에서 적절한 최적의 분리 규칙을 찾아 나무를 성장시키는 과정 적절한 정지 규칙을 만족하면 중단
- 2) 가지치기 단계: 오차를 크게 할 위험이 높거나 부적절한 추론 규칙을 가지고 있는 가지 또는 불필요한 가지 제거
- 3) 타당성 및 평가 단계: 이익도표, 위험도표, 혹은 시험자료를 이용하여 의사결정나무를 평가
- 4) 해석 및 예측 단계: 구축된 나무모형 해석하고 예측 모형을 설정한 후 예측에 적용

의사결정 나무

● 나무의 성장

- 나무모형의 성장과정은 x 들로 이루어진 입력 공간을 재귀적으로 분할하는 과정

1) 분리 규칙

- 분리 변수가 연속형: $A=x \leq s$ / 분리 변수 범주형 $\{1, 2, 3, 4\}$: $A=1,2,4$ 와 A 의 여집합=3으로 나뉨
- 최적 분할의 결정은 불순도 감소량을 가장 크게 하는 분할
- 각 단계에서 최적 분리기준에 의한 분할을 찾은 다음 각 분할에 대하여도 동일한 과정 반복

2) 분리기준

목표변수	기준값	분리기준
이산형	카이제곱 통계량 p 값	p 값이 가장 작은 예측변수와 그때의 최적분리에 의해서 자식 마디 형성
	지니지수	지니 지수를 감소시켜주는 예측변수와 그 때의 최적분리에 의해서 자식마디 선택
	엔트로피지수	엔트로피 지수가 가장 작은 예측변수와 이 때의 최적분리에 의해 자식마디를 형성
연속형	분산분석에서 F 통계량	p 값이 가장 작은 예측변수와 그때의 최적분리에 의해서 자식 마디 형성
	분산의 감소량	분산의 감소량을 최대화하는 기준의 최적분리에의해서 자식 마디 형성

의사결정 나무

- 나무의 성장

- 3) 정지 규칙

- 더 이상 분리가 잃어 나지 않고, 현재의 마디가 끝마디가 되도록 하는 규칙
- 정지 기준: 의사결정 나무의 깊이를 지정, 끝마디의 레코드 수의 최소 개수를 지정

- 나무의 가지치기

- 너무 큰 나무모형은 자료를 과대 적합하고 너무 작은 나무모형은 과소 적합할 위험이 있다.
- 나무의 크기를 모형의 복잡도로 볼 수 있으며 최적의 나무 크기는 자료로부터 추정하게 된다.
- 일반적으로 사용되는 방법은 마디에 속하는 자료가 일정수(가령 5) 이하 때, 분할을 정지하고 비용-복잡도 가지치기를 이용하여 성장시킨 나무를 가지치기하게 된다.

불순도의 여러 가지 측도

- 목표 변수가 범주형 변수인 의사결정 나무의 분류 규칙을 선택하기 위해서는 카이제곱 통제량, 지니 지수, 엔트로피 지수를 활용한다.
- 카이제곱 통제량
 - 각 셀에 대한 ((실제도수-기대도수)의 제곱/기대도수)의 합으로 구할 수 있다.
 - 기대도수 = 열의 합계 * 합의 합계 / 전체 합계

$$\chi^2 = \sum_j \frac{(O_j - E_j)^2}{E_j}$$

- 지니지수
 - 노드의 불순도를 나타내는 방법. 지니지수의 값이 클수록 이질적, 순수도가 낮음

$$G = 1 - \sum_i p_i^2$$

- 엔트로피 지수
 - 열역학에 쓰는 개념으로 무질서 정도에 대한 측도. 엔트로피 지수의 값이 클수록 순수도가 낮음. 엔트로피 지수가 가장 낮은 예측변수와 이때의 최적분리 규칙에 의해 자식마디 형성

$$E = - \sum_i p_i \log_2 p_i ,$$

의사결정 나무 알고리즘

- CART(Classification and regression Tree)
 - 앞에서 설명한 방식의 가장 많이 활용되는 의사결정나무 알고리즘
 - 불순도의 측도로 출력 변수가 범주형일 경우 지니지수를 이용, 연속형인 경우 분산을 이용한 이진분리를 사용개별 입력변수 뿐만 아니라 입력변수들의 선형결합들 중 최적의 분리를 찾을 수 있다.
- C4.0와 C5.0
 - CART와 다르게 각 마디에서 다지분리가 가능하면 범주형 입력변수에 대해서는 범주의 수만큼 분리가 일어남
 - 불순도의 측도로는 엔트로피 지수 사용
- CHAID(Chi-squared Automatic Intergraction Detection)
 - 가지치기를 하지 않고 적당한 크기에서 나무모형의 성장을 중지하고 입력변수가 반드시 범주형변수이어야 함
 - 불순도의 측도로는 카이제곱 통계량 사용

R을 이용한 의사결정나무 실습 코드

- iris 데이터 셋

```
>> idx <- sample(2, nrow(iris), replace = TRUE, prob = c(0.7, 0.3))  
>> train.data <- iris[idx==1,]  
>> test.data <- iris[idx==2,]
```

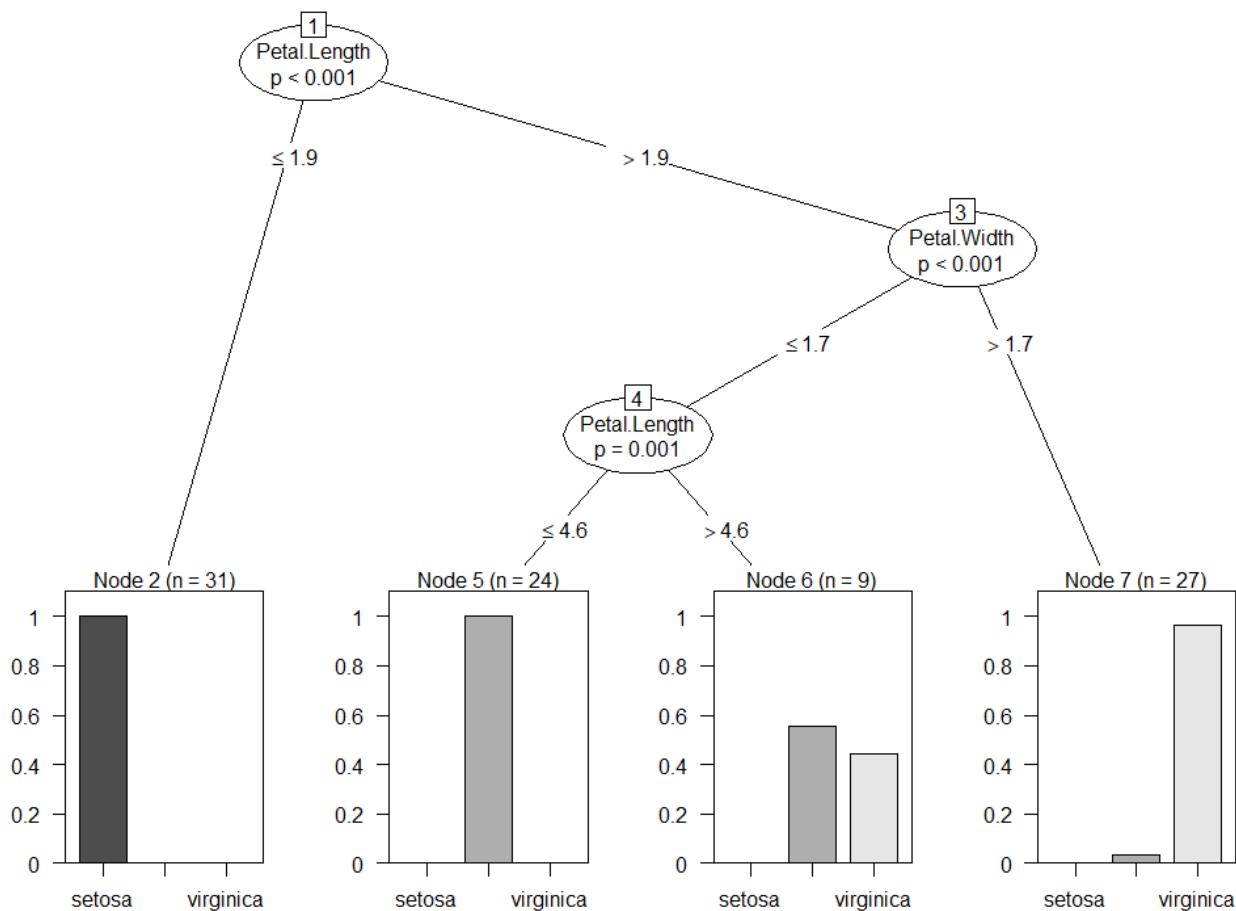
- train.data를 이용하여 모형생성

```
>> iris.tree <- ctree(Species~., data=train.data)  
>> plot(iris.tree)
```


제2절 분류분석

R을 이용한 의사결정나무 실습 코드

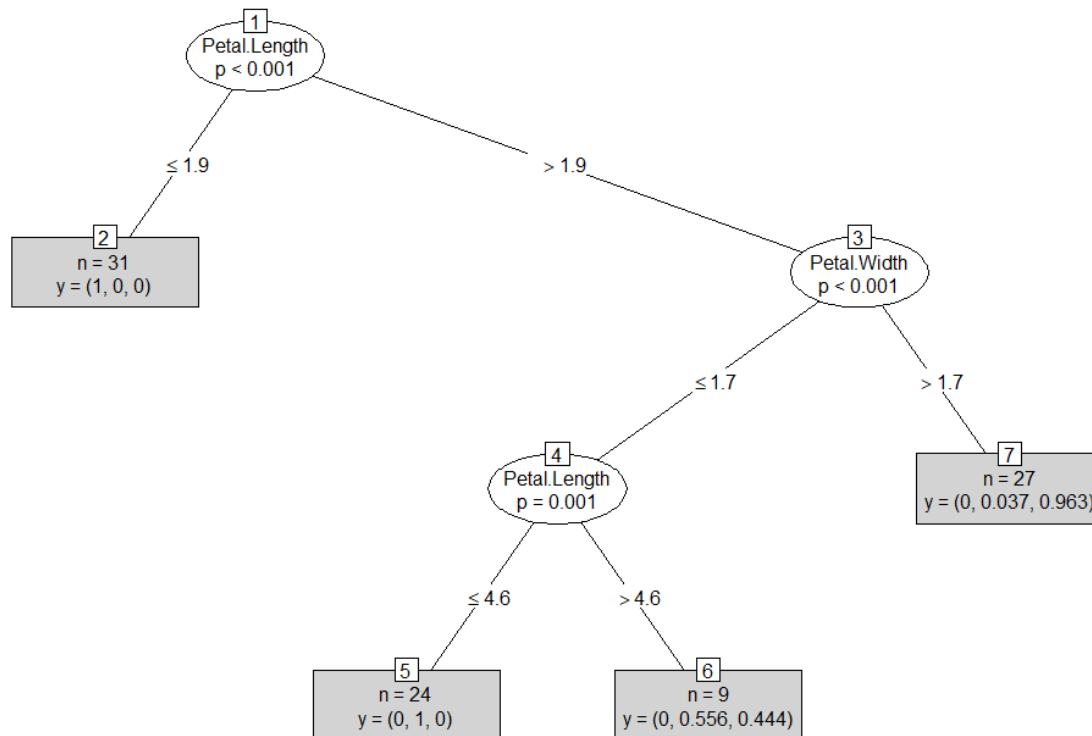
- iris 데이터 셋



R을 이용한 의사결정나무 실습 코드

- iris 데이터 셋

```
>> plot(iris.tree, type="simple")
```



Iris-rpart() 함수를 이용한 의사결정 트리 생성

```
> install.packages("rpart")
```

```
Installing package into '/home/k8s/R/x86_64-pc-linux-gnu-library/3.6'
```

```
(as 'lib' is unspecified)
```

```
URL 'https://cloud.r-project.org/src/contrib/rpart_4.1.16.tar.gz'을 시도합니다
```

```
Content type 'application/x-gzip' length 859107 bytes (838 KB)
```

```
=====
```

```
downloaded 838 KB
```

```
* installing *source* package 'rpart' ...
```

```
** 패키지 'rpart'는 성공적으로 압축해제되었고, MD5 sums 이 확인되었습니다
```

```
** using staged installation
```

```
** libs
```

```
gcc -std=gnu99 -I"/usr/share/R/include" -DNDEBUG -fpic -g -O2 -fdebug-prefix-map=/build/r-base-jbaK_j/r-base-3.6.3=. -fstack-protector-strong -Wformat -Werror=format-security -Wdate-time -D_FORTIFY_SOURCE=2 -g -c anova.c -o anova.o
```

```
...
```

```
* DONE (rpart)
```

```
The downloaded source packages are in
```

```
  '/tmp/RtmpbsjBkY/downloaded_packages'
```

```
> library(rpart)
```

```
> install.packages("rpart.plot")
```

```
Installing package into '/home/k8s/R/x86_64-pc-linux-gnu-library/3.6'
```

```
(as 'lib' is unspecified)
```

```
URL 'https://cloud.r-project.org/src/contrib/rpart.plot_3.1.0.tar.gz'을 시도합니다
```

```
Content type 'application/x-gzip' length 672013 bytes (656 KB)
```

```
=====
```

```
downloaded 656 KB
```

```
...
```

```
* DONE (rpart.plot)
```

```
The downloaded source packages are in
```

```
  '/tmp/RtmpbsjBkY/downloaded_packages'
```

```
> library(rpart.plot)
```

Iris-rpart() 함수를 이용한 의사결정 트리 생성

```
> data(iris)
```

- ❖ Rpart 패키지에서 rpart() 함수는 재귀분할의 의미가 있다. 기존 ctree() 함수에 비해 2 수준 요인으로 부산분석을 실행한 결과를 트리 형태로 제공하여 모델을 단순화해 주기 때문에 전체적인 분류기준을 쉽게 분석할 수 있는 장점이 있습니다.

Iris-rpart() 함수를 이용한 의사결정 트리 생성

```
> rpart_model <- rpart(Species ~ ., data = iris)
```

```
> rpart_model
```

```
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) *
```

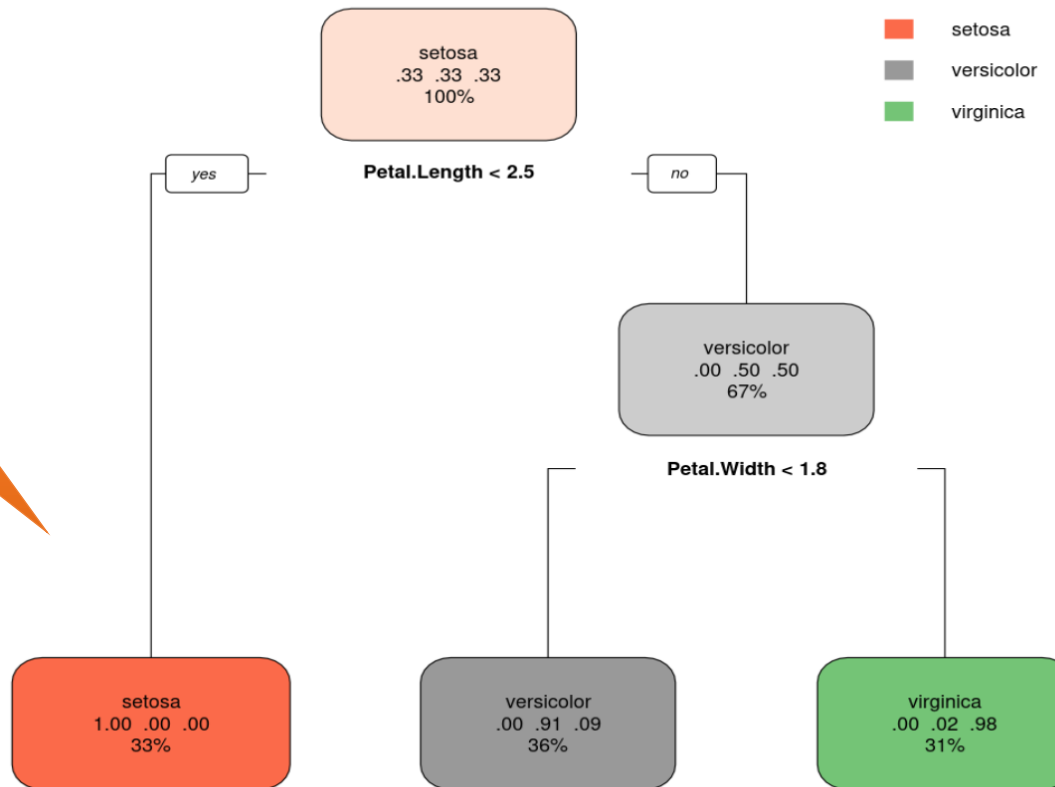
```
7) Petal.Width>=1.75 46 1 virginica (0.00000000 0.02173913 0.97826087) *
```

반응변수에 Species 변수를 지정하고 , 설명변수에 나머지 4개의 변수를 지정하기 위해 ~뒤에 .을 표시합니다.

Iris-rpart() 함수를 이용한 의사결정 트리 생성

> rpart.plot(rpart_model)

마지막 노드는
반응변수의 결
과값



R을 이용한 의사결정나무 실습 코드

- 예측된 데이터와 실제 데이터의 비교

```
>> table(predict(iris.tree), train.data$Species)
```

	setosa	versicolor	virginica
setosa	31	0	0
versicolor	0	29	4
virginica	0	1	26

```
>>
```

- 테스트 데이터를 적용하여 정확성 확인

```
>> test.pre <- predict(iris.tree, newdata=test.data)
```

```
>> table(test.pre, test.data$Species)
```

test.pre	setosa	versicolor	virginica
setosa	19	0	0
versicolor	0	20	1
virginica	0	0	19

```
>>
```

학습목표

- 앙상블 기법에 대해 이해한다.
- 배깅, 부스팅, 랜덤포레스트 기법의 차이점을 이해한다.

눈높이 체크

- 앙상블 기법에 대해 알고 계신가요?
- 앙상블 기법의 종류에는 어떤 것이 있을까요?

앙상블

- 앙상블 정의
 - 주어진 자료로부터 여러 개의 예측모형들을 조합하여 하나의 최종 예측 모형을 만드는 방법. 다중 모델 조합, 분류기 조합이 있다.
- 학습방법의 불안전성
 - 학습자료의 작은 변화에 의해 예측모형이 크게 변하는 경우 그 학습 방법은 불안정이다
 - 가장 안정적인 방법으로는 1-near neighbor(가장 가까운 자료만 변하지 않으면 예측 모형이 변하지 않음), 선형회귀모형(최소제곱법으로 추정해 모형 결정)이 존재한다
 - 가장 불안정한 방법으로는 의사결정나무가 있다.

앙상블

● 앙상블 기법의 종류

- 배깅
 - 주어진 자료에 여러개의 붓스트랩 자료를 생성하고 각 붓스트랩 자료에 예측모형을 만든 후 결합하여 최종 예측모형을 만드는 방법
 - 붓스트랩: 주어진 자료에서 동일한 크기의 표본을 랜덤 복원추출로 뽑은 자료를 의미
 - 보팅: 여러 개의 모형으로부터 산출된 결과를 다수결에 의해서 최종결과를 선정하는 과정
 - 최적의 의사결정나무를 구축할 때 가장 어려운 부분이 가지치기이지만 배깅에서는 가지치기 하지 않고 최대한 성장한 의사결정나무를 활용
 - 훈련자료의 모집단의 분포를 모르기 때문에 실제 문제에서는 평균 예측 모형을 구할 수 없다
 - 배깅은 이러한 문제를 해결하기 위해 훈련자료를 모집단으로 생각하고 평균예측 모형을 구하여 분산을 줄이고 예측력을 향상시킬 수 있다.

앙상블

- 앙상블 기법의 종류
 - 부스팅
 - 예측력이 약한 모형들을 결합하여 강한 예측모형을 만드는 방법
 - Adaboost: 부스팅 방법 중 하나. 이진분류 문제에서 랜덤분류기보다 조금 더 좋은 분류기 n 개에 각각 가중치를 설정하고 n 개의 분류기를 결합하여 최종 분류기를 만드는 방법(단, 가중치의 합은 1)
 - 훈련오차가 빨리 그리고 쉽게 줄일 수 있다
 - 배깅에 비해 많은 경우 예측오차가 향상 되어 Adaboost의 성능 배깅보다 뛰어난 경우가 많다.

앙상블

- 앙상블 기법의 종류
 - 랜덤 포레스트
 - 의사결정나무의 특징인 분산이 크다는 점을 고려하여 배깅과 부스팅보다 더 많은 무작위성을 주어 약한 학습기들을 생성한 후 이를 선형결합하여 최종 학습기를 만드는 방법
 - randomForest 패키지는 random input에 따른 forest of tree를 이용한 분류방법
 - 랜덤한 forest에 많은 트리들이 생성됨
 - 수천개의 변수를 통해 변수제거 없이 실행되므로 정확도 측면에서 좋은 성과를 보임
 - 이론적설명이나 최종 결과에 대한 해석이 어렵다는 단점이 있지만 예측력이 매우 높다. 특히 입력변수가 많은 경우 배깅과 부스팅과 비슷하거나 좋은 예측력을 보인다

랜덤 포레스트 기본 모델 생성

```
> install.packages("randomForest")
```

```
Installing package into '/home/k8s/R/x86_64-pc-linux-gnu-library/3.6'  
(as 'lib' is unspecified)
```

```
Warning in install.packages :
```

```
package 'randomForest' is not available (for R version 3.6.3)
```

```
> library(randomForest)
```

```
randomForest 4.6-14
```

```
Type rfNews() to see new features/changes/bug fixes.
```

다음의 패키지를 부착합니다: 'randomForest'

The following object is masked from 'package:ggplot2':

```
margin
```

```
> data(iris)
```

```
>
```

랜덤 포레스트 기본 모델 생성

❖ 기본 모델

```
> model1 <- randomForest(Species ~ ., data = iris)
> model1
```

Call:

```
randomForest(formula = Species ~ ., data = iris)
```

Type of random forest: classification

Number of trees: 500

No. of variables tried at each split: 2

OOB estimate of error rate: 4%

Confusion matrix:

	setosa	versicolor	virginica	class.error
setosa	50	0	0	0.00
versicolor	0	47	3	0.06
virginica	0	3	47	0.06

>

Number of trees: 500은 학습 데이터로 300개의 포레스트가 복원 추출방식으로 생성되었다는 의미입니다.

No. of variables tried at each split: 2은 2개의 변수를 이용하여 트리의 자식 노드가 분류되었다는 의미입니다.

랜덤 포레스트 기본 모델 생성

❖ 파라미터 조정 - 트리 개수 300개, 변수 개수 4개 지정

- ntree 트리의 개수 변수의 개수 mtry

❖ na.action = na.omit 속성은 결측치가 있는 경우 처리할 방법을 지정하는 속성입니다. na.omit() 함수를 이용하여 결측치를 제거했습니다.

```
> model2 <- randomForest(Species ~ ., data = iris,  
+                          ntree = 300, mtry = 4, na.action = na.omit)  
> model2
```

Call:
randomForest(formula = Species ~ ., data = iris, ntree = 300, mtry = 4, na.action = na.omit)

Type of random forest: classification

Number of trees: 300

No. of variables tried at each split: 4

OOB estimate of error rate: 4.67%

Confusion matrix:

	setosa	versicolor	virginica	class.error
setosa	50	0	0	0.00
versicolor	0	47	3	0.06
virginica	0	4	46	0.08

>

랜덤 포레스트 기본 모델 생성

- ❖ Importance 속성은 분류모델을 생성하는 과정에서 입력 변수 중 가장 중요한 변수가 어떤 변수인가를 알려주는 역할을 합니다.

```
model3 <- randomForest(Species ~ ., data = iris, importance = T, na.action = na.omit)
```


랜덤 포레스트 기본 모델 생성

❖ importance () 함수의 실행결과에서 MeanDecreaseAccuracy 는 분류정확도를 개선하는 데 기여한 변수를 수치로 제공하며, MeanDecreaseGini는 노드 불순도(불확실성)을 개선하는 데 기여한 변수를 수치로 제공합니다. 따라서, 꽃의 종류를 분류하는 데 있어 4개의 변수 중에서 가장 크게 기여하는 변수는 Petal.Length 로 나타났습니다.

> importance(model3)

	setosa	versicolor	virginica	MeanDecreaseAccuracy	MeanDecreaseGini
Sepal.Length	6.601863	7.910837	6.875874	11.022591	9.935650
Sepal.Width	4.947080	1.996566	4.472099	5.722676	2.297089
Petal.Length	23.334707	35.718754	28.878774	35.381956	
Petal.Width	21.311756	31.010528	30.062347	32.127687	42.956509

44.061533

$$\log_2\left(\frac{1}{1/2}\right) = 1$$

$$\log_2\left(\frac{1}{1/8}\right) = 3$$

$$\log_2\left(\frac{1}{1/4}\right) = 2$$

※ 엔트로피(Entropy): 불확실성 척도

$$H(x) = - \sum_{i=1}^n p(x_i) \log p(x_i)$$

랜덤 포레스트 기본 모델 생성

※ 엔트로피(Entropy): 불확실성 척도

- 동전의 앞면(x1)과 뒷면(x2)이 나올 확률이 동일한 경우,

```
> x1 <- 0.5; x2 <- 0.5  
> e1 <- -x1 * log2(x1) - x2 * log2(x2)  
> e1  
[1] 1
```

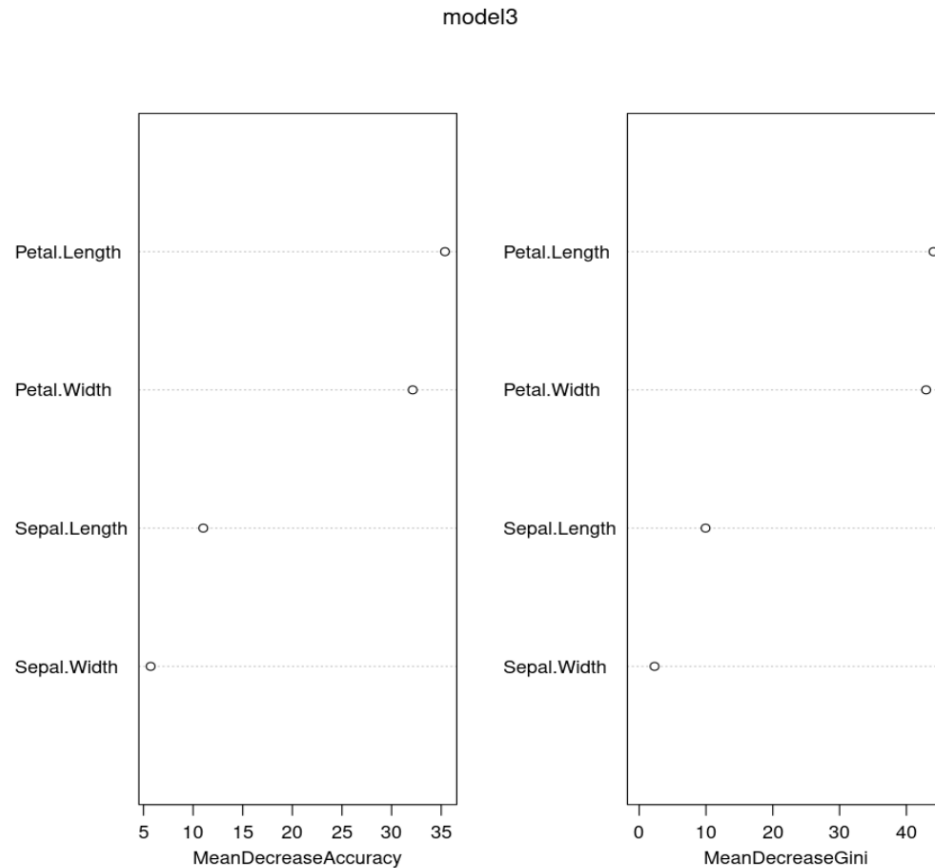
- 엔트로피가 1로 나옴.
- 앞면이 나올 확률이 더 높은 경우,

```
> x1 <- 0.7; x2 <- 0.3  
> e2 <- -x1 * log2(x1) - x2 * log2(x2)  
> e2  
[1] 0.8812909  
>
```

- 엔트로피가 0.881로 나옴. 이는 앞면이 나올 확률이 높기 때문에 그만큼 불확실성이 낮아진 것임.

랜덤 포레스트 기본 모델 생성

> varImpPlot(model3)



R을 이용한 랜덤포레스트 실습 코드

- 모형만들기

```
>> idx <- sample(2, nrow(iris), replace = TRUE, prob = c(0.7, 0.3))
>> train.data <- iris[idx==2,]
>> test.data <- iris[idx==1,]
>> r.f <- randomForest(Species~., data=train.data, ntree=100, proximity=TRUE)
```

- 오차율 계산하기

```
>> table(predict(r.f), train.data$Species)
```

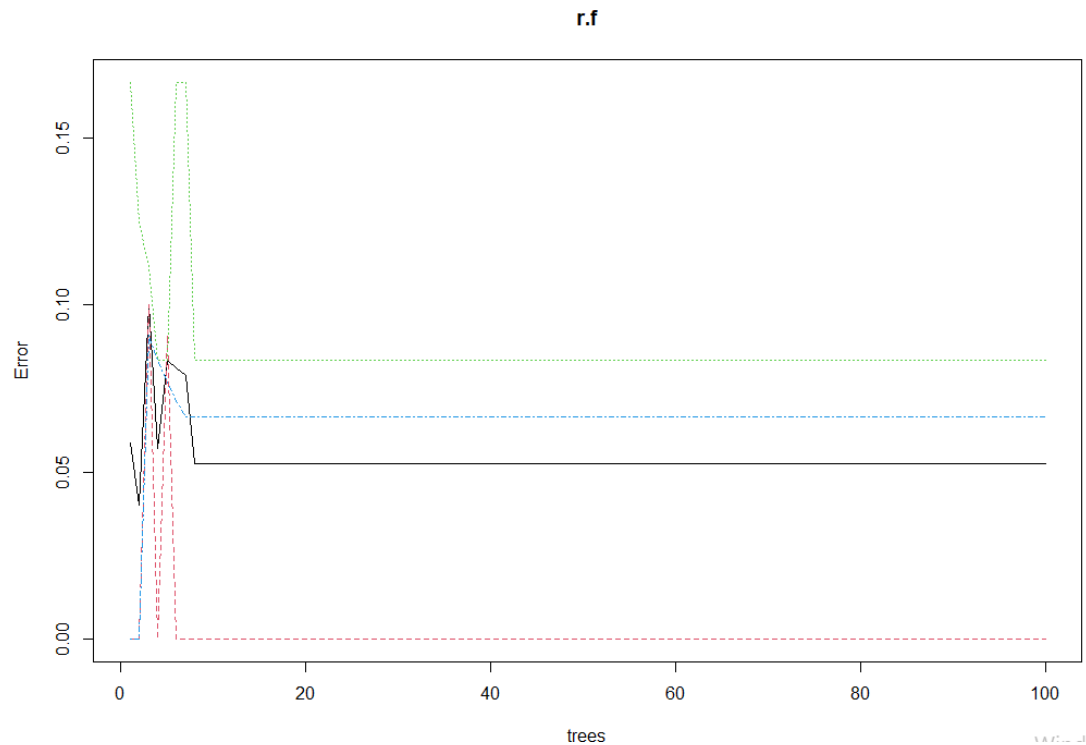
	setosa	versicolor	virginica
setosa	11	0	0
versicolor	0	11	1
virginica	0	1	14

제3절 앙상블분석

R을 이용한 랜덤포레스트 실습 코드

- 그래프 그리기1

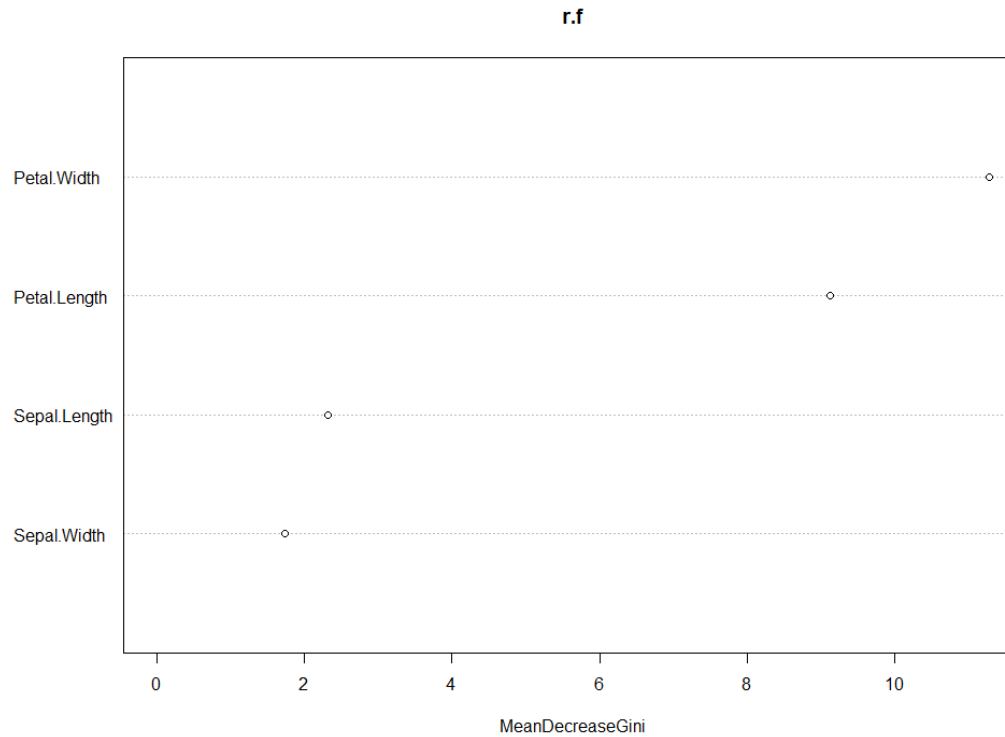
```
>> plot(r.f)
```



R을 이용한 랜덤포레스트 실습 코드

- 그래프 그리기2

```
>> varImpPlot(r.f)
```



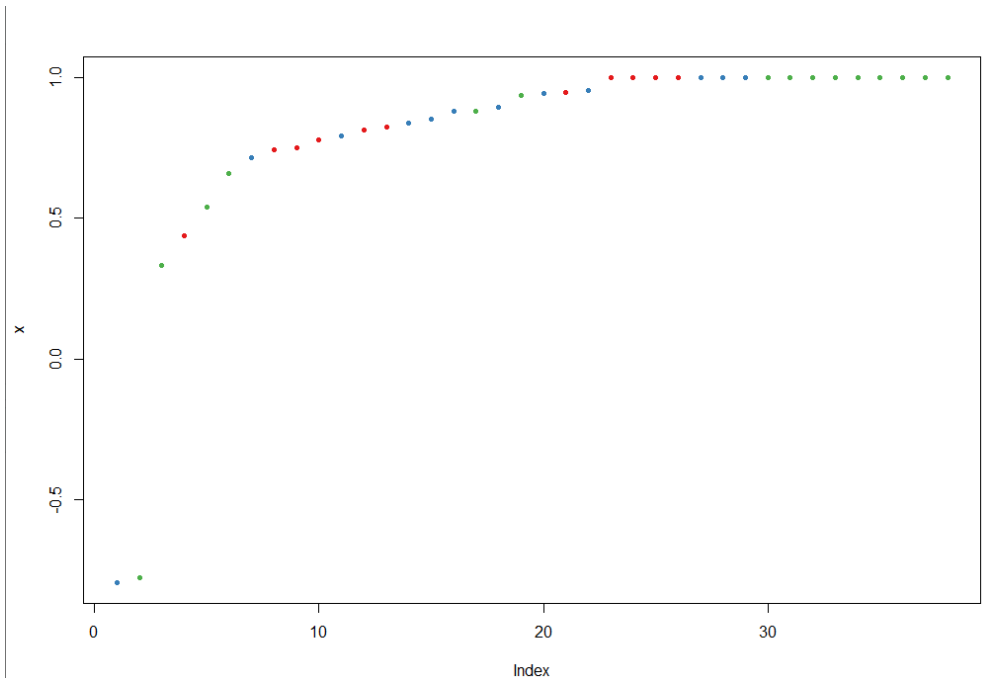
R을 이용한 랜덤포레스트 실습 코드

- 테스트 데이터 예측

```
>> table(pre.rf, test.data$Species)
```

pre.rf	setosa	versicolor	virginica
setosa	39	0	0
versicolor	0	37	4
virginica	0	1	31

```
>> plot(margin(r.f, test.data$Species))
```



다항 분류 xgboost 모델 생성

- ❖ Xgboost는 랜덤포레스트와 같은 앙상블 학습기법으로 모델을 생성하는 분류모델입니다. Xgboost는 부스팅 방식을 기반으로 만들어진 모델이기 때문에 분류하기 어려운 특정 영역에 초점을 두고 정확도를 높이는 알고리즘으로 구현되었습니다. 따라서 높은 정확도가 가장 큰 장점입니다.

```
> install.packages("xgboost")
```

```
Installing package into '/home/k8s/R/x86_64-pc-linux-gnu-library/3.6'  
(as 'lib' is unspecified)  
also installing the dependency 'data.table'
```

```
URL 'https://cloud.r-project.org/src/contrib/data.table_1.14.2.tar.gz'을 시도합니다  
Content type 'application/x-gzip' length 5301817 bytes (5.1 MB)  
=====  
downloaded 5.1 MB
```

```
URL 'https://cloud.r-project.org/src/contrib/xgboost_1.6.0.1.tar.gz'을 시도합니다  
Content type 'application/x-gzip' length 1062074 bytes (1.0 MB)  
=====  
downloaded 1.0 MB
```

```
installing *source* package 'data.table' ...
```

```
...
```

```
* DONE (xgboost)
```

```
The downloaded source packages are in  
  '/tmp/RtmpqplY0g/downloaded_packages'
```

```
> library(xgboost)
```


다항 분류 xgboost 모델 생성

```
> iris_label <- ifelse(iris$Species == 'setosa', 0,  
+                       ifelse(iris$Species == 'versicolor', 1, 2))  
> table(iris_label)  
iris_label  
0 1 2  
50 50 50  
> iris$label <- iris_label
```

- ❖ Xgboost에서 사용할 y 변수의 레이블은 숫자로 표기되어야 합니다. 앞서 원-핫 인코딩 방식을 사용할 수도 있지만, 각 범주를 0에서 2로 변경해 쓸 수도 있습니다.

다항 분류 xgboost 모델 생성

```
> idx <- sample(nrow(iris), 0.7 * nrow(iris))  
> train <- iris[idx, ]  
> test <- iris[-idx, ]
```

❖ 홀드 아웃 방식으로 훈련 셋과 검정 셋을 7:3 비율로 데이터 셋을 생성합니다.

다항 분류 xgboost 모델 생성

```
> train_mat <- as.matrix(train[-c(5:6)])
```

```
> dim(train_mat)
```

```
[1] 105  4
```

```
> train_lab <- train$label
```

```
> length(train_lab)
```

```
[1] 105
```

- ❖ Xgboost 모델을 생성하기 위해서 x변수는 matrix 객체로 변환하고, y변수는 label이용하여 준비합니다.
- ❖ train[-c(5:6)]은 iris\$Species와 iris\$label을 제외한 데이터셋을 말합니다.

다항 분류 xgboost 모델 생성

❖ xgb.DMatrix 객체 변환 xgb.DMatrix 객체 변환

```
> dtrain <- xgb.DMatrix(data = train_mat, label = train_lab)
```

❖ X,y 변수를 이용하여 xgboost 전용 xgb.Dmatrix() 함수를 이용하여 학습 데이터 셋을 생성합니다.

다항 분류 xgboost 모델 생성

```
> xgb_model <- xgboost(data = dtrain, max_depth = 2, eta = 1,
+                      nthread = 2, nrounds = 2,
+                      objective = "multi:softmax",
+                      num_class = 3,
+                      verbose = 0)
> xgb_model
##### xgb.Booster
raw: 8.7 Kb
call:
  xgb.train(params = params, data = dtrain, nrounds = nrounds,
    watchlist = watchlist, verbose = verbose, print_every_n = print_every_n,
    early_stopping_rounds = early_stopping_rounds, maximize = maximize,
    save_period = save_period, save_name = save_name, xgb_model = xgb_model,
    callbacks = callbacks, max_depth = 2, eta = 1, nthread = 2,
    objective = "multi:softmax", num_class = 3)
params (as set within xgb.train):
  max_depth = "2", eta = "1", nthread = "2", objective = "multi:softmax", num_class = "3", validate_parameters = "TRUE"
xgb.attributes:
  niter
callbacks:
  cb.evaluation.log()
# of features: 4
niter: 2
nfeatures : 4
evaluation_log:
  iter train_mlogloss
    1      0.2637150
    2      0.1149656
```

다항 분류 xgboost 모델 생성

```
> xgb_model <- xgboost(data = dtrain, max_depth = 2, eta = 1,
+                       nthread = 2, nrounds = 2,
+                       objective = "multi:softmax",
+                       num_class = 3,
+                       verbose = 0)
> xgb_model
##### xgb.Booster
raw: 8.7 Kb
call:
  xgb.train(params = params, data = dtrain, nrounds = nrounds,
    watchlist = watchlist, verbose = verbose, print_every_n = print_every_n,
    early_stopping_rounds = early_stopping_rounds, maximize = maximize,
    save_period = save_period, save_name = save_name, xgb_model = xgb_model,
    callbacks = callbacks, max_depth = 2, eta = 1, nthread = 2,
    objective = "multi:softmax", num_class = 3)
params (as set within xgb.train):
  max_depth = "2", eta = "1", nthread = "2", objective = "multi:softmax", num_class = "3", validate_parameters = "TRUE"
xgb.attributes:
  niter
callbacks:
  cb.evaluation.log()
# of features: 4
niter: 2
nfeatures : 4
evaluation_log:
  iter train_mlogloss
    1      0.2637150
    2      0.1149656
```

다항 분류 xgboost 모델 생성

❖ 주요 속성

- eta : 학습률 지정(기본 값 0.3-숫자가 낮을 수록 모델의 복잡도가 높아지고, 컴퓨팅 파워가 많아짐)
- nthread : CPU 사용 수 지정
- nrounds : 반복 학습 수 지정
- objective : y변수가 이항("binary:logistic") 또는 다항("multi:softmax") 지정
- verbose : 메시지 출력 여부(0: 메시지 출력 안 함, 1: 메시지 출력)

다항 분류 xgboost 모델 생성

```
> test_mat <- as.matrix(test[-c(5:6)])
```

```
> dim(test_mat)
```

```
[1] 45 4
```

```
> test_lab <- test$label
```

```
> length(test_lab)
```

```
[1] 45
```




```
> pred_iris <- predict(xgb_model, test_mat)
```

```
[1]00000000000000000000000011111212111111212222222222
```

```
> table(pred_iris, test_lab)
```

```

      test_lab
pred_iris 0  1  2
      0 19  0  0
      1  0 13  1
      2  0  2 12

```

다항 분류 xgboost 모델 생성

```
> (19 + 13 + 12) / length(test_lab)
[1] 0.9777778
```

```
> importance_matrix <- xgb.importance(colnames(train_mat),
+                                     model = xgb_model)
```

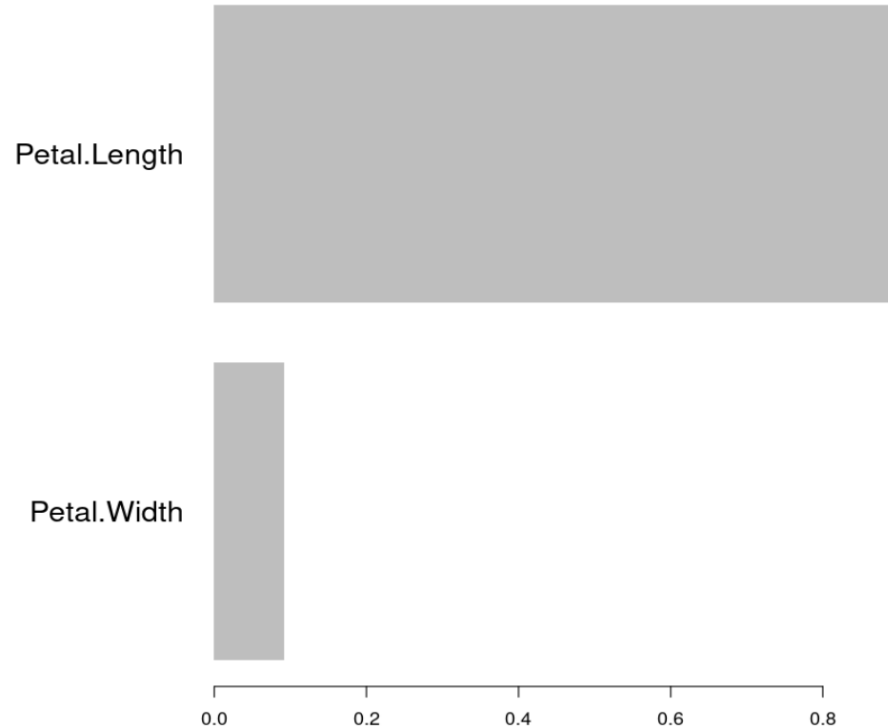
```
> importance_matrix
```

	Feature	Gain	Cover	Frequency
1:	Petal.Length	0.90825394	0.7810849	0.8
2:	Petal.Width	0.09174606	0.2189151	0.2

- ❖ 생성된 트리 모델과 함께 해서 y에 가장 영향을 미치는 주요 변수 x를 확인할 수 있습니다. 랜덤 포레스트에서 제공되는 결과와 함께 동일하게 Petal.Length와 Petal.Width 변수가 가장 중요한 변수로 나타나고 있습니다.

다항 분류 xgboost 모델 생성

```
> xgb.plot.importance(importance_matrix)
```





제4절 인공지능경망분석

학습목표

- 인공지능경망에 대해 이해한다.
- 인공지능경망 구축시 고려사항을 이해한다.
- R프로그램을 통해 인공지능경망 기법을 활용할 수 있다.
- R프로그램을 통해 예측분석을 활용할 수 있다.

눈높이 체크

- 인공지능경망에 대해 알고 계신가요?
- 인공지능경망을 개발할 때 고려해야 할 사항은 어떤 것이 있을까요?

인공신경망 분석(ANN)

- 인공신경망이란?
 - 인간 뇌를 기반으로 한 추론 모델. 뉴런은 기본적인 정보처리 단위이다
- 연구
 - 1943 매컬락과 피츠: 인간의 뇌를 수많은 신경세포가 연결된 하나의 디지털 네트워크 모형으로 간주하고 신경세포의 신호처리 과정을 모형화하여 단순 패턴 분류 모형을 개발
 - 헵: 신경세포 사이의 연결 강도를 조정하여 학습 규칙을 개발
 - 로젠블랫:퍼셉트론이라는 인공세포를 개발
 - 비선형성의 한계점 발생-XOR: 문제를 풀지 못하는 한계 발견
 - 홉필드, 러멜하트, 맥클랜드: 역전파 알고리즘을 활용하여 비선형성을 극복하여 다계층 퍼셉트론으로 새로운 인공신경망 모형이 등장

인공신경망 분석(ANN)

- 인간의 뇌를 형상화한 인공신경망
- 인간 뇌의 특징: 100억개의 뉴런과 6조 개의 시냅스의 결합체. 인간의 뇌는 현존하는 어떤 컴퓨터보다 빠르고 매우 복잡하고, 비선형적이며, 병력적인 정보처리 시스템과 같다. 적응성에 따라 잘못된 답에 대한 뉴런들 사이의 연결은 약화되고 올바른 답에 대한 연결이 강화된다.
- 인간의 뇌 모델링: 뉴런은 가중치가 있는 링크들로 연결되어 있다. 뉴런은 여러 입력 신호를 받지만 출력 신호는 오직 하나만 생성한다.
- 인공신경망 학습
- 신경망은 가중치를 반복적으로 저장하며 학습 뉴런은 링크로 연결되어 있고, 각 링크에는 수치적인 가중치가 있다.
- 인공신경망은 신경망의 가중치를 초기화하고 훈련 데이터를 통해 가중치를 갱신하여 신경망의 구조를 선택하고, 활용할 학습 알고리즘을 결정한 후 신경망을 훈련시킨다.

인공신경망의 특징

● 구조

- 입력링크에서 여러 신호를 받아서 새로운 활성화 수준을 계산하고, 출력 링크로 출력 신호를 보냄
입력 신호는 미가공 데이터 또는 다른 뉴런의 출력이 될 수 있다. 출력 신호는 문제의 최종적인 해가 되거나 다른 뉴런에 입력될 수 있다.

● 뉴런의 계산

- 뉴런은 전이함수 즉, 활성화 함수를 사용함
- 활성화 함수를 이용해 출력을 결정하며 입력 신호의 가중치 합을 계산하여 임계값과 비교
가중치 합이 임계값보다 작으면 뉴런은 -1, 같거나 크면 +1 출력

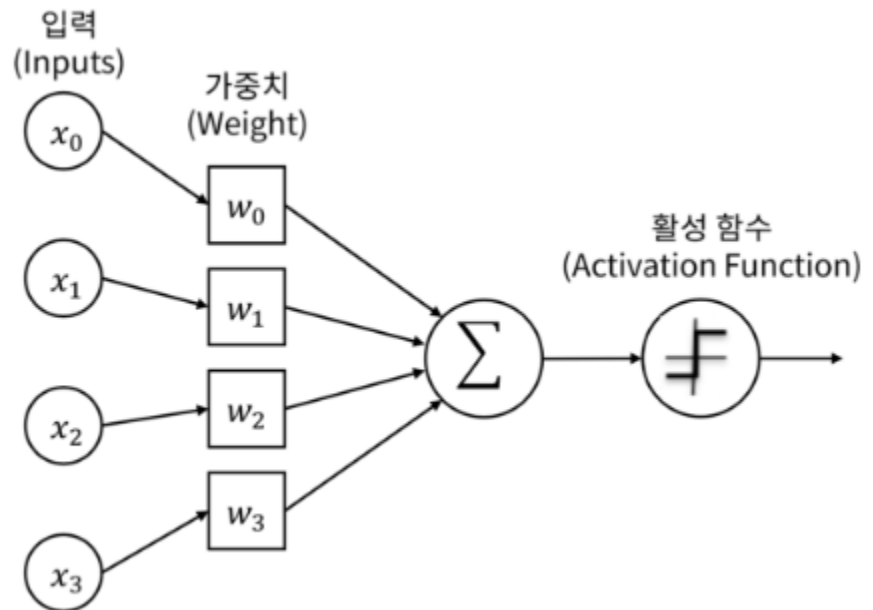
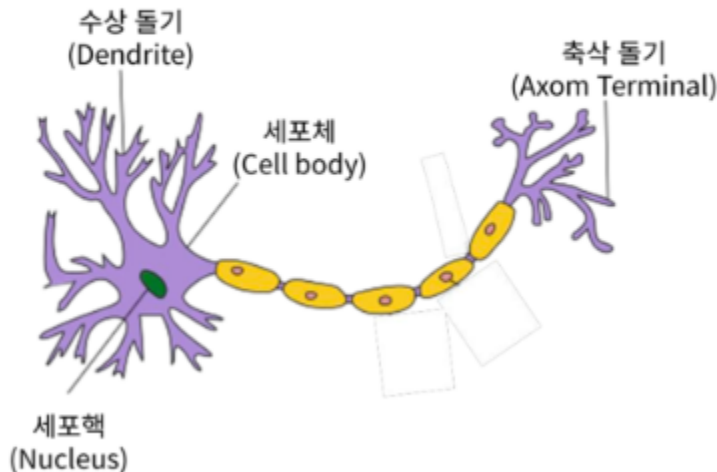
● 뉴런의 활성화함수

- 시그모이드 함수의 경우 로지스틱 회귀분석과 유사하며 0~1의 확률 값을 가진다
- softmax 함수: 표준화 지수 함수라고도 불리며, 출력 값이 여러 개로 주어지고 목표치가 다범주인 경우 각 범주에 속할 사후 확률을 제공하는 함수
- Relu함수: 입력값이 0 이하는 0, 0 이상은 x 값을 가진 함수이며 최근 딥러닝에서 많이 사용

제4절 인공신경망분석

인공신경망의 특징

- 단일 뉴런의 학습(단층 퍼셉트론)



- 퍼셉트론은 선형 결합기와 하드리 미터로 구성초평면은 n 차원 공간을 두 개의 영역으로 나눈다. 초평면을 선형 분리 함수로 정의한다.

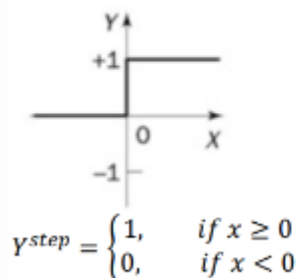


제4절 인공신경망분석

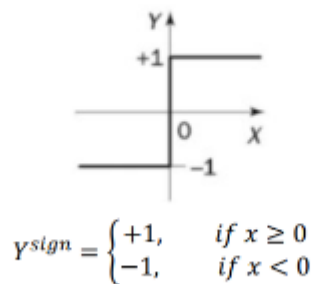
인공신경망의 특징

- 뉴런의 활성화 함수

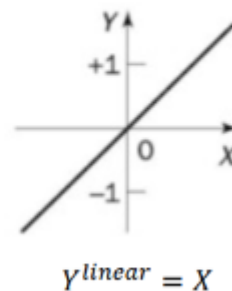
계단 함수



부호 함수



선형 함수





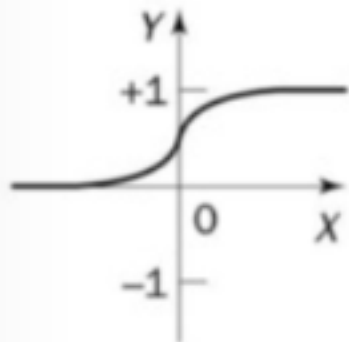
제4절 인공신경망분석

인공신경망의 특징

- 뉴런의 활성화 함수

sigmoid 함수

- 연속형 0~1, **Logistic** 함수라 불리기도 함.
- 선형적인 멀티-퍼셉트론에서 비선형 값을 얻기 위해 사용



$$y_{sigmoid} = \frac{1}{1 + e^{-x}}$$

인공신경망의 특징

● 뉴런의 활성화 함수

softmax 함수

- 모든 logits의 합이 1이 되도록 output을 정규화
- sigmoid 함수의 일반화된 형태로, 결과가 다 범주인 경우 각 범주에 속할 사후 확률(Posterior Probability)을 제공하는 활성화 함수
- 주로 3개 이상 분류 시 사용함.



$$\text{softmax}(x)_i = \frac{\exp(x_i)}{\sum_j \exp(x_j)}$$

신경망 모형 구축 시 고려사항

- 입력 변수
- 입력 변수가 범주형 또는 연속형일 때 아래의 조건이 신경망 모형에 적합하다.
 - 범주형 변수: 모든 범주에서 일정 빈도 이상의 값을 갖고 각 범주의 빈도가 일정할 때
 - 연속형 변수: 입력변수 값들의 범위가 변수 간의 큰 차이가 없을 때
- 연속형 변수의 경우 그 분포가 평균을 중심으로 대칭이 아니면 좋지 않은 결과를 도출하기 때문에 아래와 같은 방법을 활용
 - 변환: 고객의 소득(대부분 평균 미만이고, 특정 고객의 소득이 매우 큰): 로그 변환
 - 범주화: 각 범주의 빈도가 비슷하게 되도록 설정
- 범주형 변수의 경우 가변수화하여 적용(남녀:1,0 or 1,-1) 하고 가능하면 모든 범주형 변수는 같은 범위를 갖도록 가변수화 하는 것이 좋다.

신경망 모형 구축 시 고려사항

- 가중치의 초기값과 다중 최소값 문제
- 역전파 알고리즘은 초기값에 따라 결과가 많이 달라지므로 초기값의 선택은 매우 중요한 문제
- 가중치가 0이면 시그모이드 함수는 선형이 되고, 신경망 모형은 근사적으로 선형 모형이 된다.
- 일반적으로 초기값은 0 근처로 랜덤 하게 선택하므로 초기 모형은 선형 모형에 가깝고, 가중치값이 증가할수록 비선형 모형이 된다.(초기값이 0이면 반복하여도 값이 변하지 않고, 너무 크면 좋지 않은 해를 주는 문제점을 내포하고 있음)



제4절 인공신경망분석

신경망 모형 구축 시 고려사항

● 학습모드

학습모드	내용
온라인 학습모드	각 관측값을 순차적으로 하나씩 신경망에 투입하여 가중치 추정값이 매번 바뀐다. 일반적으로 속도가 빠르며 특히 훈련자료에 유사값이 많은 경우 그 차이가 두드러진다. 훈련자료가 비정상성과 같이 특이한 성질을 가진 경우가 좋다. 국소최솟값에서 벗어나기 더 쉽다.
확률적 학습모드	온라인 학습모드와 같으나 신경망에 통비되는 관측값의 순서가 랜덤하다.
배치 학습모드	전체훈련자료를 동시에 신경망에 투입한다. ※ 학습률: 처음에는 큰 값으로 정하고 반복 수행과정을 통해 해에 가까울수록 0에 수렴함



제4절 인공신경망분석

신경망 모형 구축 시 고려사항

- 은닉층과 은닉 노드의 수
- 신경망을 적용할 때 가장 중요한 부분은 모형의 선택이다.(은닉층의 수와 은닉 노드의 수 결정)
- 은닉층과 은닉 노드가 많으면 가중치가 많아져 과대적합 문제 발생 / 적으면 과소적합 발생
- 은닉층의 수가 하나인 신경망은 범용 근사자이므로 모든 매끄러운 함수를 근사적으로 표현할 수 있다. 그러므로 가능하면 은닉층은 하나로 선정
- 은닉 노드의 수는 적절히 큰 값으로 놓고 가중치를 감소시키며 적용하는 것이 좋음

신경망 모형 구축 시 고려사항

- 과대 적합
 - 알고리즘의 조기종료와 가중치 감소 기법으로 해결할 수 있다.
 - 모형이 적합한 과정에서 검증 오차가 증가하기 시작하면 반복을 중지하는 조기 종료를 시행
 - 선형 모형의 능형회귀와 유사한 가중치 감소라는 벌 점화 기법을 활용
- 딥러닝
 - 머신러닝의 한 분야로서 인공신경망의 한계를 극복하기 위해 제안된 심화 신경망을 활용한 방법 최근 음성과 이미지 인식, 자연어 처리, 헬스케어 등 전반적 인분야에 활용
 - 딥러닝 소프트웨어: Tensorflow, caffe, Theano, MXnet 등이 있다.

학습목표

- 군집분석에 대해 이해한다.
- 계층적 군집분석 알고리즘에 대해 이해한다.
- 비계층적 군집분석 알고리즘에 대해 이해한다.
- R 프로그램을 통해 군집분석을 활용할 수 있다.

눈높이 체크

- 군집분석에 대해 알고 계신가요?
- 군집분석의 종류는 어떤 것이 있을까요?
- k-means 군집분석에 대해 알고 계신가요?

군집분석

● 개요

- 각 객체의 유사성을 측정하여 유사성이 높은 대상 집단을 분류하고 군집에 속한 객체들의 유사성과 서로 다른 군집에 속한 객체 간의 상이성을 규명하는 분석방법
- 특성에 따라 고객을 여러 개의 배타적인 집단으로 나누는 것
- 결과는 구체적인 군집분석 방법에 따라 차이가 날 수 있다.
- 군집의 개수나 구조에 대한 가정 없이 데이터들 사이의 거리를 기준으로 군집화를 유도함
- 마케팅 조사에서 소비자들의 상품 구매행동이나 life style에 따른 소비자 군을 분류하여 시장 전략 수립 등에 활용

● 군집분석 특징

- 요인 분석과의 차이점: 요인 분석은 유사한 변수를 함께 묶어주는 것이 목적
- 판별분석과의 차이점: 판별분석은 사전에 집단이 나누어져 있는 자료를 통해 새로운 데이터를 기존의 집단에 할당하는 것이 목적

군집분석

● 군집 분석의 종류

군집	형태	군집간 거리 척도/연결법 (Linkage Method)
계층적 군집 (Hierarchical)	응집형(Agglomerative) Bottom-Up	<ul style="list-style-type: none"> - 단일(최단) 연결법 - 완전(최장) 연결법 - 평균 연결법 - 중심 연결법 - Ward 연결법
	분리형(Divisive) Top-Down	다이나나 방법(DIANA Method)
분할적 군집 (Partitional)	프로토타입 기반 (Prototype-Based)	<ul style="list-style-type: none"> - k-중심 군집 : k-평균 군집, k-중앙값 군집, k-메도이드 군집 - 퍼지(Fuzzy) 군집
	분포 기반 (Distribution-Based)	혼합 분포 군집
	밀도 기반 (Density-Based)	<ul style="list-style-type: none"> - 중심 밀도 군집 - 격자 기반 군집

거리

- 군집분석에서는 관측 데이터 간 유사성이나 근접성을 측정해 어느 군집으로 묶을 수 있는지 판단해야 한다.
- 연속형 변수인 경우
- 유클리디안 거리 / 표준화 거리 / 마할라노비스 거리 / 체비셰프 거리 / 맨하탄 거리 / 캔버라 거리 / 민코우스키 거리

종류	내용
유클리디안 거리	<p>데이터간의 유사성을 측정할 때 많이 사용되는 거리. 통계적 개념이 내포되어 있지 않아 변수들의 산포정도가 전혀 감안되어 있지 않다.</p> $d(x,y) = \sqrt{(x_1 - y_1)^2 + \dots + (x_p - y_p)^2} = \sqrt{(x-y)'(x-y)}$
표준화거리	<p>해당 변수의 표준편차로 척도 변환 후 유클리드안 거리를 계산하는 방법 표준화하게 되면 척도의 차이, 분산의 차이로 왜곡을 피할 수 있음</p> $d(x,y) = \sqrt{(x-y)'D^{-1}(x-y)}$

거리

● 연속형 변수인 경우

마할라노비스 거리	<p>통계적 개념이 포함된 거리. 변수들의 산포를 고려하여 이를 표준화한 거리 두 벡터 사이의 거리를 산포를 의미하는 표준공분산으로 나눠주어야 하며 그룹에 대한 사전지식없이 표본공분산을 계산할 수 없으므로 사용하기 곤란</p> $d(x,y) = \sqrt{(x-y)'S^{-1}(x-y)}$
체비셰프 거리	$d(x,y) = \max_i x_i - y_i $
맨하탄 거리	<p>유클리디안 거리와 함께 가장 많이 사용되는 거리로 맨하탄 도시에서 건물에서 건물을 가기 위한 최단 거리를 구하기 위해 고안된 거리</p> $d(x,y) = \sum_{i=1}^p x_i - y_i $
캔버라 거리	$d(x,y) = \sum_{i=1}^p \frac{ x_i - y_i }{(x_i - y_i)}$
민코우스키 거리	<p>맨하탄 거리와 유클리디안 거리를 한번에 표현한 공식. L1거리, L2거리라 불리고 있음</p> $d(x,y) = [\sum_{i=1}^p x_i - y_i ^m]^{1/m}$



제5절 군집분석

거리

● 범주형 변수의 경우

종류	내용
자카드 거리	$1 - J(A,B) = \frac{ A \cup B - A \cap B }{ A \cup B }$
다카드 계수	$J(A,B) = \frac{ A \cap B }{ A \cup B }$
코사인 거리	문서를 유사도를 기준으로 분류 혹은 그룹핑할 때 유용하게 사용 $d_{\cos}(A,B) = 1 - \frac{A \cdot B}{\ A\ _2 \cdot \ B\ _2}$
코사인 유사도	두 개체 벡터 내적 코사인 값을 이용하여 측정 된 벡터간의 유사한 정도. 두 벡터 A, B에 대해 코사인 유사도는 아래와 같이 정의된다. $\text{cosine similarity} = \frac{A \cdot B}{\ A\ _2 \cdot \ B\ _2}$

계층적 군집분석

- 계층적 군집방법은 n 개의 군집으로 시작해 점차 군집의 개수를 줄여 나가는 방법
- 계층적 군집을 형성하는 방법은 합병형 방법과 분리형 방법이 있다.
- 계층적군집 형성 방법

종류	내용
최단연결법	$n \times n$ 거리행렬에서 거리가 가장 가까운 데이터를 묶어서 군집을 생성 군집과 군집 or 데이터와의 거리를 계산 시 최단거리를 거리로 계산하여 거리행렬 수정을 진행 수정된 거리행렬에서 거리가 가까운 데이터 또는 군집을 새로운 군집으로 생성
최장연결법	군집과 군집 or 데이터와의 거리를 계산할 때 최장거리를 거리로 계산하여 거리행렬을 수정
중심연결법	두 군집 중심간 거리를 군집간 거리로 한다. 군집이 결합될 때, 새로운 군집의 평균은 가중평균을 통해 구해짐
평균연결법	군집과 군집 또는 데이터와의 거리를 계산할 때 평균을 거리로 계산하여 거리행렬을 수정
와드연결법	군집 내 편차들의 제곱합을 고려하는 방법 군집 간 정보의 손실을 최소화하기 위해 군집화를 진행

계층적 군집분석

- 군집화
 - 거리행렬을 통해 가장 가까운 거리의 객체들 간의 관계를 규명하고 덴드로그램을 그린다. 덴드로그램을 보고 군집의 개수를 변화해가면서 적절한 군집을 선정한다. 군집의 수는 분석 목적에 따라 선정할 수 있지만 대부분 5개 이상의 군집은 잘 활용하지 않는다.
- 군집화 단계
 - 1) 거리행렬을 기준으로 덴드로그램을 그린다.
 - 2) 덴드로그램의 최상단으로부터 세로축의 개수에 따라 가로선을 그려 군집의 개수를 선택
 - 3) 각 객체들의 구성을 고려하여 적절한 군집수를 선정

비계층적 군집분석

- n 개의 개체를 g 개의 군집으로 나눌 수 있는 모든 가능한 방법을 점검해 최적화한 군집을 형성하는 것(K-평균 군집분석)
- K-평균 군집분석 개념
 - 주어진 데이터를 k 개의 클러스터로 묶는 알고리즘. 각 클러스터와 거리 차이의 분산을 최소화하는 방식으로 동작
- K-평균 군집분석 과정
 - 1) 원하는 군집의 개수와 초기값들을 정해 seed 중심으로 군집을 형성
 - 2) 각 데이터를 거리가 가장 가까운 seed가 있는 군집으로 분류
 - 3) 각 군집의 seed 값을 다시 계산모든 개체가 군집으로 할당될 때까지
 - 4) 1)~3) 반복

비계층적 군집분석

- K-평균 군집분석 특징
 - 거리 계산을 통해 군집화가 이루어지므로 연속형 변수에 활용이 가능
 - K개의 초기 중심값은 임의로 선택이 가능. 가급적이면 멀리 떨어지는 것이 바람직
 - 초기 중심값을 임의로 선정할 때 일렬로 선택하면은 군집 혼합이 되자 않고 층으로 나누어질 수 있어 주의하여야 한다. 초기 중심값의 선정에 따라 결과가 달라질 수 있다.
 - 초기 중심으로부터 오차 제곱합을 최소화하는 방향으로 군집이 형성되는 그리디 알고리즘으로 안정된 군집을 보장하나 최적이라는 보장이 없다.

비계층적 군집분석

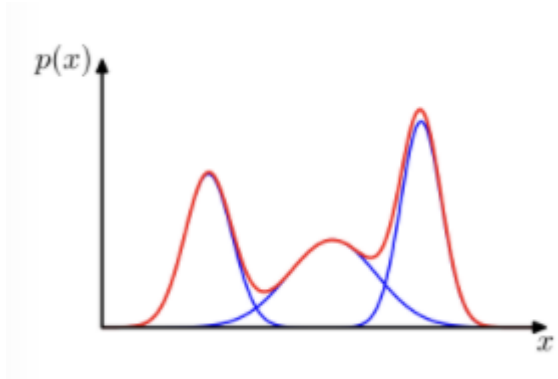
- K-평균 군집분석 장점
 - 알고리즘이 단순하며, 빠르게 수행되어 분석방법 적용이 용이
 - 계층적 군집분석에 비해 많은 양의 데이터를 다룰 수 있음
 - 내부 구조에 대한 사전 정보가 없어도 의미 있는 자료구조를 찾을 수 있다
 - 다양한 형태의 데이터에 적용이 가능
- K-평균 군집분석 단점
 - 군집의 수, 가중치와 거리 정의가 어렵다.
 - 사전에 주어진 목적이 없으므로 결과 해석이 어렵다.
 - 잡음이나 이상 값의 영향을 많이 받는다.
 - 불룩한 형태가 아닌 군집이 존재할 경우 성능이 떨어진다.
 - 초기 군집수 결정에 어려움이 있다.

혼합 분포 군집

- 개요
 - 모형기법의 군집 방법. 데이터가 k 개의 모수적 모형(흔히 정규분포 또는 다변량 정규분포를 가정)의 가중합으로 표현되는 모집단 모형으로부터 나왔다는 가정하에서 모수와 함께 가중치를 자료로부터 추정하는 방법
 - K 개의 각 모형은 군집을 의미, 각 데이터는 추정된 k 개의 모형 중 어느 모형으로부터 나왔을 확률이 높은지 따라 군집의 분류가 이루어짐
 - 흔히 혼합 모형에서는 모수와 가중치의 추정에는 EM 알고리즘에 사용

혼합 분포 군집

- 혼합 분포 모형으로 설명할 수 있는 데이터의 형태



(a)

- (a)는 자료의 분포형태가 다봉형의 형태를 띠므로 단일 분포로의 적절하지 않으며, 대략 3개 정통의 정규분포 결합을 통해 설명될 수 있을 것으로 생각할 수 있다.
- (b)의 경우에도 여러 개의 이변량 정규분포의 결합을 통해 설명될 수 있을 것이다. 두 경우 모두 반드시 정규분포로 제한할 필요가 없다.

혼합 분포 군집

- EM 알고리즘
 - 각 자료에 대한 Z 의 조건부 분포(어느 집단에 속할지에 대한 기댓값을 구할 수 있다.
 - (E-단계) 관측 변수 X 와 잠재 변수 Z 를 포함하는 (X, Z) 에 대한 로그-가능도 함수에 Z 값 대신 상수값이 Z 의 조건부 기댓값을 대입하면, 로그-가능도함수를 최대화 하는 모수를 쉽게 찾을 수 있다. (M-단계) 갱신된 모수 추정치에 대해 위 과정을 반복한다면 수렴하는 값을 얻게 되고, 이는 최대 가능도 추정치로 사용될 수 있다.
 - E-단계: 잠재변수 Z 의 기대치 계산
 - M-단계: 잠재변수 Z 의 기대치를 이용하여 파라미터를 추정

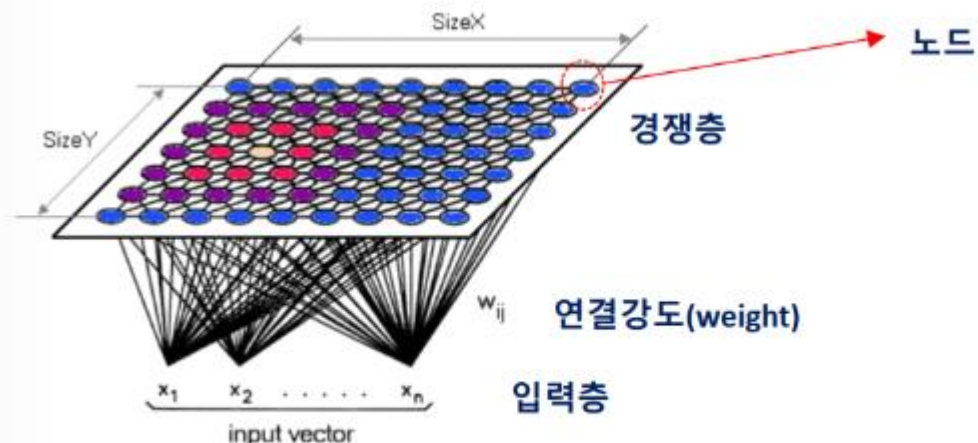
혼합 분포 군집

- 혼합 분포 군집 모형의 특징
 - K평균 군집의 절차와 유사하지만 확률 분포를 도입하여 군집을 수행
 - 군집을 몇 개의 모수로 표현할 수 있으며, 서로 다른 크기나 모형의 군집을 찾을 수 있다.
 - EM알고리즘을 이용한 모수 추정에서 데이터가 커지면 수렴에 시간이 걸릴 수 있다.
 - 군집의 크기가 너무 작으면 추정의 정도가 떨어지거나 어려울 수 있다.
 - K평균 군집과 같이 이상치 자료에 민감하므로 사전 조치가 필요하다.

SOM(Self Organizing Map)

● 개요

- 자기 조직화 지도(SOM) 알고리즘은 코호넨에 의해 제시, 개발되었으며 코호넨 맵이라고도 알려져 있다.
- SOM은 비지도 신경망으로 고차원 데이터를 이해하기 쉬운 저차원의 뉴런으로 정렬하여 지도의 형태로 형상화한다. 이러한 형상화는 입력 변수의 위치 관계를 그대로 보존한다는 특징이 있다. 다시 말해 실제 공간의 입력 변수가 가까이 있으면, 지도상에서도 가까운 위치에 있게 된다.



SOM(Self Organizing Map)

- 구성
- SOM모델은 두 개의 인공신경망층으로 구성되어 있다.

구성	내용
입력층	<p>입력벡터를 받은 층. 입력변수의 개수와 동일하게 뉴런의 수가 존재한다 입력층의 자료는 학습을 통하여 경쟁층에 정렬되는데, 이를 지도라 부름입력층에 있는 뉴런들은 경쟁층에 있는 각각의 뉴런들과 연결되어 있으며, 이때 완전연결 되어 있다.</p>
경쟁층	<p>2차원 격차로 구성된 층. 입력벡터의 특성에 따라 벡터의 한 점으로 클러스tring 되는 층SOM은 경쟁학습으로 각각의 뉴런이 입력벡터와 얼마나 가까운가를 계산하여 연결강도를반복적으로 재정하여 학습. 이 과정을 거치면서 연결강도는 입력패턴과 가장 유사한 경쟁층 뉴런이 승자가 됨 입력층의 표본벡터에 가장가까운 프로토타입 벡터를 선택해 BMU(Best-Matching-Unit)이라고 하며, 코호넨의 승자 독점의 학습규칙에 따라 위상학적 이웃에 대한 연결강도를 조정승자독식 구조로 인해 경쟁층에는 승자 뉴런만이 나타나며, 승자와 유사한 연결강도를 갖는 입력패턴이 동일한 경쟁뉴런으로 배열됨</p>

SOM(Self Organizing Map)

- 특징
 - 고차원의 데이터를 저 차원의지도 형태로 형상화하기 때문에 시각적으로 이해가 쉽다.
 - 입력 변수의 위치 관계를 그대로 보존하기 때문에 실제 데이터가 유사하면 지도상에서 가깝게 표현된다. 이런 특징 때문에 패턴 발견, 이미지 분석 등에서 뛰어난 성능이 보인다.
 - 역전파 알고리즘 등을 이용하는 인공신경망과 달리 단 하나의 전방 패스를 사용함으로써 매우 빠르다. 따라서 실시간 학습처리를 할 수 있다

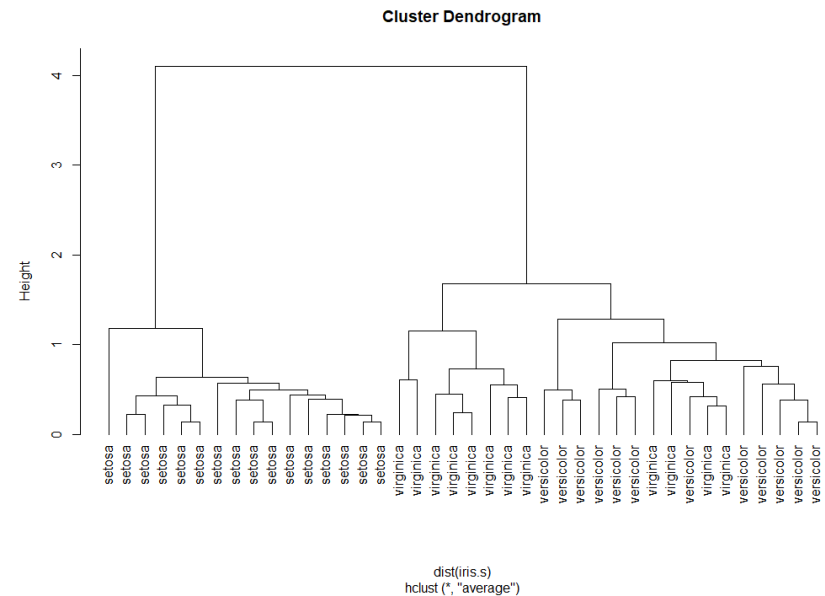
SOM과 신경망 모형의 차이점

구분	신경망모형	SOM
학습방법	오차역전파법	경쟁학습방법
구성	입력층, 은닉층, 출력층	입력층, 2차 격자 형태의 경쟁층
기계학습방법의 분류	지도학습	비지도학습

R을 이용한 군집분석 실습 코드

- Hierarchical Clustering

```
>> idx <- sample(1:dim(iris)[1], 40)
>> iris.s <- iris[idx,]
>> iris.s$Species <- NULL
>> hc <- hclust(dist(iris.s), method = "ave")
>> plot(hc, hang = -1, labels = iris$Species[idx])
```

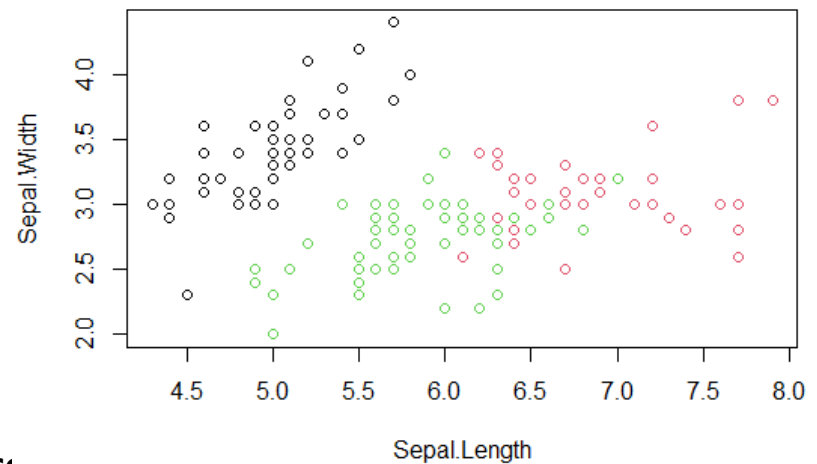


R을 이용한 군집분석 실습 코드

- K-means Clustering

```
>> data(iris)
>> newiris <- iris
>> newiris$Species <- NULL
>> kc <- kmeans(newiris, 3)
>> table(iris$Species, kc$cluster)
```

```
      1  2  3
setosa  50  0  0
versicolor  0  2 48
virginica  0 36 14
>> plot(newiris[c("Sepal.Length", "Sepal.Width")], cl
```



학습목표

- 연관분석에 대해 이해한다.
- 연관분석의 척도를 이해한다.
- 연관규칙의 장점과 단점을 이해한다.
- R프로그램을 통해 연관성분석을 적용할 수 있다.

눈높이 체크

- 연관성분석에 대해 알고 계신가요?
- 연관성분석의 척도를 이해하고 계신가요?
- 연관성분석의 장단점을 알고 계신가요?

연관 규칙

- 연관 규칙 분석 개념
 - 기업의 데이터베이스에서 상품의 구매, 서비스 등 일련의 거래 또는 사건들 간의 규칙을 발견하기 위해 적용한다.
 - 장바구니 분석: 장바구니에 무엇이 같이 들어 있는지에 대한 분석
 - 서열분석: A를 산 다음 B를 산다.
- 연관 규칙의 형태
 - 조건과 반응의 형태로 이루어져 있다.

연관 규칙

- 연관 규칙의 측도
- 산업의 특성에 따라 지지도, 신뢰도, 향상도 값을 잘 보고 규칙을 선택해야 한다.

종류	내용
지지도	전체 거래 중 항목 A와 B를 동시에 포함하는 거래의 비율 $P(A \cap B) / P(A)$: A와 B가 동시에 포함된 거래 수 / A가 포함된 거래 수
신뢰도	항목 A를 포함하는 거래 중 항목 A와 항목 B가 같이 포함될 확률. 연관성 정도를 파악 $P(A \cap B) / P(A)$: A와 B가 동시에 포함된 거래 수 / A가 포함된 거래 수
향상도	A가 구매되지 않았을 때 품목 B의 구매확률에 비해 A가 구매됐을 때 품목 B의 구매확률의 증가비 연관규칙 A->> B는 품목 A와 품목 B의 구매가 서로 관련이 없는 경우에 향상도는 1이 됨 향상도가 1보다 높을 수록 연관성이 높다. 향상도가 1보다 크면 B를 구매할 확률보다 A를 구매한 B를 구매할 확률이 더 높다는 의미 $P(B A) / P(B) = P(A \cap B) / (P(A) * P(B))$ = A와 B 동시 포함된 거래 수 / (A가 포함된 거래수 * B가 포함된 거래수)

연관성 분석 주요 측도

- 연관성 분석에서 가장 핵심적인 개념은 각 아이템 간의 연관성을 파악하는 주요 3개 측도인 지지도(Support), 신뢰도(Confidence), 향상도(Lift)이다.
- 지지도(Support)
 - 지지도는 전체 데이터 세트에서 해당 아이템 집합이 포함된 비율을 말하며 아래의 $s(X)$ 혹은 $s(X,Y)$ 와 같이 표현된다.

$$S(X) = \frac{\text{count}(X)}{N} \quad (4.11)$$

$$S(X, Y) = \frac{\text{count}(X, Y)}{N} = P(X \cap Y)$$

- 즉, 지지도는 빈도적 관점에서 확률을 정의할 때, 전체 데이터 세트 중 아이템 집합 $\{X, Y\}$ 가 발생할 확률과 같다.

연관성 분석 주요 척도

- 신뢰도(Confidence)
- 신뢰도는 연관규칙 $\{X\} \rightarrow \{Y\}$ 에서 '조건' X 를 포함한 아이템 세트 중에서 X, Y 둘 다 포함된 아이템 세트가 발생한 비율을 말하는데, 규칙의 왼쪽에 있는 '조건 X '가 발생했다는 조건하에 규칙의 오른쪽에 있는 '결과 Y '가 발생할 확률을 의미한다. 신뢰도는 특정 연관 규칙의 예측력이나 정확도에 대한 측정이다. 사실 이 신뢰도는 조건부 확률 $P(Y|X)$ 와 동일한 의미이다.

$$\begin{aligned} Conf(X \Rightarrow Y) &= \frac{S(X, Y)}{S(X)} = \frac{\frac{count(X, Y)}{N}}{\frac{count(X)}{N}} = \frac{count(X, Y)}{count(X)} \quad (4.12) \\ &= \frac{P(X \cap Y)}{P(X)} = P(Y|X) \end{aligned}$$

- 신뢰도 $(X \rightarrow Y)$ 와 신뢰도 $(Y \rightarrow X)$ 는 서로 같지 않다. 즉, 두 신뢰도 모두 X, Y 가 모두 포함된 지지도(X, Y)가 분자에 포함되어 있지만 신뢰도 $(X \rightarrow Y)$ 는 지지도(X)가 분모에 들어가게 되고, 신뢰도 $(Y \rightarrow X)$ 는 지지도(Y)가 분모에 들어가게 되는 점이 다르다. 결국 이는 조건부 확률 $P(Y|X)$ 와 $P(X|Y)$ 가 서로 다른 결과를 가져오는 것과 동일한 의미라고 할 수 있다.

연관성 분석 주요 척도

- 향상도(Lift)
 - 지지도와 신뢰도는 연관규칙 생성과 탐사에 있어서 매우 중요한 핵심개념임에는 틀림없지만, 지지도(X, Y)와 신뢰도($X \rightarrow Y$)가 높았다는 것만으로 유의미한 규칙이라고 결론 내리기는 어렵다. 그 이유는 만일 전체 데이터 세트에서 원래부터 아이템 세트 $\{Y\}$ 가 포함된 경우의 수가 많았다면, 아이템 세트 $\{Y\}$ 와 함께 아이템 세트 $\{X\}$ 가 포함되어 있을 가능성도 그만큼 커지고, 지지도(X, Y)와 신뢰도($X \rightarrow Y$)는 높게 나타날 수밖에 없기 때문이다.
 - 즉, 연관규칙 $\{X\} \rightarrow \{Y\}$ 가 탐색 되었을 때 정말로 조건 $\{X\}$ 가 발생했을 때 결과 $\{Y\}$ 가 함께 나타나는 경우의 수가 많아서 그런 것인지, 아니면 원래부터 $\{Y\}$ 가 많이 포함되어 있어 연관규칙 $\{X\} \rightarrow \{Y\}$ 가 탐색 된 것인지에 대한 보다 명확한 측정 지표가 필요하게 된다. 이럴 때 이를 측정하는 지표가 바로 향상도(Lift)이다.
 - 향상도(Lift)가 의미하는 바는 조건 $\{X\}$ 가 주어지지 않았을 때의 결과 $\{Y\}$ 가 발생할 확률 대비, 조건 $\{X\}$ 가 주어졌을 때의 결과 $\{Y\}$ 의 발생 확률의 증가 비율을 의미한다. 즉, 아이템 집합 $\{Y\}$ 가 원래 발생된 경우의 수보다 연관규칙 $\{X\} \rightarrow \{Y\}$ 가 탐색 되었을 때 조건에 해당하는 아이템 집합 $\{X\}$ 가 주어졌다는 “정보”가 결과 아이템 집합 $\{Y\}$ 가 발생하게 되는 경우의 수를 예상하는 데 얼마나 유용하느냐를 나타낸다고 할 수 있다.



제6절 연관분석

연관성 분석 주요 척도

- 향상도(Lift)
- 향상도(Lift)는 다음과 같이 측정된다.

$$Lift(X \Rightarrow Y) = \frac{Conf(X \Rightarrow Y)}{S(Y)} \quad (4.13)$$

$$= \frac{\frac{S(X, Y)}{S(X)}}{S(Y)} = \frac{S(X, Y)}{S(X)S(Y)} = \frac{\frac{count(X, Y)}{N}}{\frac{count(X)}{N} \frac{count(Y)}{N}}$$

$$= \frac{P(Y|X)}{P(Y)} = \frac{\frac{P(X \cap Y)}{P(X)}}{P(Y)} = \frac{P(X \cap Y)}{P(X)P(Y)}$$

- 따라서 향상도가 1을 넘어가면 조건과 결과 아이템 집합 간에 서로 양의 상관관계가 있으며, 1보다 작으면 서로 음의 상관관계 만일 향상도가 1로 나타나면 조건과 결과 아이템 집합은 서로 독립적인 관계라고 할 수 있다.

제6절 연관분석

연관성 분석 주요 척도

• 예

지지도(Support), 신뢰도(Confidence), 향상도(Lift) 예시

Customer ID	Transaction ID	Items
1131	1번	계란, 우유
2094	2번	계란, 기저귀, 맥주, 사과
4122	3번	우유, 기저귀, 맥주, 콜라
4811	4번	계란, 우유, 맥주, 기저귀
8091	5번	계란, 우유, 맥주, 콜라

↓
N = 5 (전체 transaction 개 수)

$$s(Y) = n(Y) / N \\ = n(2번, 3번, 4번) / N = 3 / 5 = 0.6$$

연관규칙 {계란, 맥주} → {기저귀} 에 대해
X Y

▪ 지지도(Support)

$$s(X \rightarrow Y) = n(X \cup Y) / N \\ = n(2번, 4번) / N \\ = 2 / 5 = 0.4$$

▪ 신뢰도(Confidence)

$$c(X \rightarrow Y) = n(X \cup Y) / n(X) \\ = n(2번, 4번) / n(2번, 4번, 5번) \\ = 2 / 3 = 0.667$$

▪ 향상도(Lift)

$$Lift(X \rightarrow Y) = c(X \rightarrow Y) / s(Y) \\ = 0.667 / 0.6 = 1.111$$

연관 규칙

- 연관 규칙의 장점
 - 탐색적인 기법으로 조건반응으로 표현되는 연관성 분석 결과를 쉽게 이해할 수 있다.
 - 강력한 비목적성 분석기법으로 분석 방향이나 목적이 특별히 없는 경우 목적 변수가 없으므로 유용하게 활용 사용이 편리한 분석 데이터의 형태로 거래 내용에 대한 데이터를 변호나 없이 그 자체로 이용할 수 있는 간단한 자료구조를 갖는다. 분석을 위한 계산이 간단

연관 규칙

- 연관 규칙의 단점
 - 품목의 수가 증가하면 분석에 필요한 계산은 기하급수적으로 늘어난다.
 - 이를 개선하기 위해 유사한 품목을 한 범주로 일반화
 - 연관규칙의 신뢰도 하한을 새롭게 정의해 실제 드물게 관찰되는 의미 적은 연관 규칙은 제외함
 - 너무 세분화한 품목을 갖고 연관성 규칙을 찾으면 의미 없는 분석이 될 수도 있다.
 - 적절히 구분되는 큰 범주로 구분해 전체 분석에 포함시킨 후 결과 중에서 세부적으로 연관 규칙을 찾는 작업 수행
 - 거래량이 적은 품목은 당연히 포함된 거래수가 적을 것이고, 규칙 발견 시 제외하기 쉬움
 - 이런 경우, 그 품목이 관련성을 살펴보고자 하는 중요한 품목이라면 유사한 품목들과 함께 범주로 구성하는 방법 등을 통해 연관성 규칙의 과정에 포함시킬 수 있다.

연관 규칙

- 순차 패턴
 - 동시에 구매될 가능성이 큰 상품군을 찾아내는 연관성 분석에 시간이라는 개념을 포함시켜 순차적으로 구매 가능성이 큰 상품군을 찾아내는 것이다.
 - 연관성 분석에서의 데이터 형태에서 각각의 고객으로부터 발생한 구매 시점에 대한 정보가 포함된다.

기존 연관성 분석의 이슈

- 대용량 데이터에 대한 연관성 분석은 불가능하다.
- 시간이 많이 걸리거나 기존 시스템에서 실행 시 시스템 다운 현상이 발생한다.

최근 연관성 분석 동향

- 1세대 알고리즘: Apriori / 2세대 알고리즘: FP-Growth / 3세대 알고리즘: FPV를 이용해 메모리를 효율적으로 사용함으로써 SKU 레벨의 연관성 분석을 성공적으로 적용
- 거래내역에 포함되는 모든 품목의 개수가 n 개일 때, 품목들의 전체 집합에서 추출할 수 있는 품목의 부분집합의 개수는 $2^n - 1$ 이다. 그리고 가능한 모든 연관 규칙의 개수는 $3^n - 2^{n+1} + 1$ 개이다.

최근 연관성 분석 동향

● Apriori 알고리즘(1세대)

- 모든 가능한 품목 부분집합의 개수를 줄이는 방식으로 작동하는 것이 Apriori 알고리즘
- 최소 지지도보다 큰 지지도 값을 갖는 품목의 집합을 빈발 항목 집단이라고 한다. Apriori 알고리즘은 품목 집합에 대한 지지도를 전부 계산하는 것이 아니라, 최소 지지도 이상의 빈발 항목 집합을 찾은 후 그것들에 대해서만 연관 규칙을 계산하는 것이다.
- 구현과 이해가 쉽다는 장점이 있으나, 지지도가 낮은 후보 집합 생성 시 아이템의 개수가 많아지면 계산 복잡도가 증가한다는 문제를 가짐

● FP-Growth 알고리즘(2세대)

- 거래내역 안에 포함된 품목의 개수를 줄여 비교하는 횟수를 줄이는 방식으로 작동하는 것
- 후보 빈발 항목 집합을 생성하지 않고 FP-Tree를 만든 후 분할정복방식을 통해 Apriori 알고리즘보다 더 빠르게 빈발 항목 집합을 추출하는 방법
- Apriori 알고리즘의 약점을 보완하기 위해 고안된 것. 데이터베이스를 스캔하는 횟수가 작도 빠른 속도로 분석 가능

제6절 연관분석

연관성 분석 주요 척도

● 연관성 분석 알고리즘

Association Rule : strategy and algorithm



- 1 모든 가능한 항목집합 개수(M)를 줄이는 방식 ➡ Apriori algorithm
- 2 Transaction 개수(N)를 줄이는 방식 ➡ DHP Algorithm
- 3 비교하는 수(W)를 줄이는 방식 ➡ FP-growth Algorithm

Apriori algorithm

- 최소지지도 이상을 갖는 항목집합을 빈발항목집합(frequent item set)이라고 합니다. 모든 항목집합에 대한 지지도를 계산하는 대신에 최소 지지도 이상의 빈발항목집합만을 찾아내서 연관규칙을 계산하는 것이 Apriori algorithm의 주요 내용입니다.
- 빈발항목집합 추출의 Apriori Principle
 1. 한 항목집합이 빈발(frequent)하다면 이 항목집합의 모든 부분집합은 역시 빈발항목집합이다.(frequent item sets -> next step)
 2. 한 항목집합이 비빈발(infrequent)하다면 이 항목집합을 포함하는 모든 집합은 비빈발항목집합이다. (superset -> pruning)
- Apriori Pruning Principle

If there is any itemset which is infrequent, its superset should not be generated/tested!

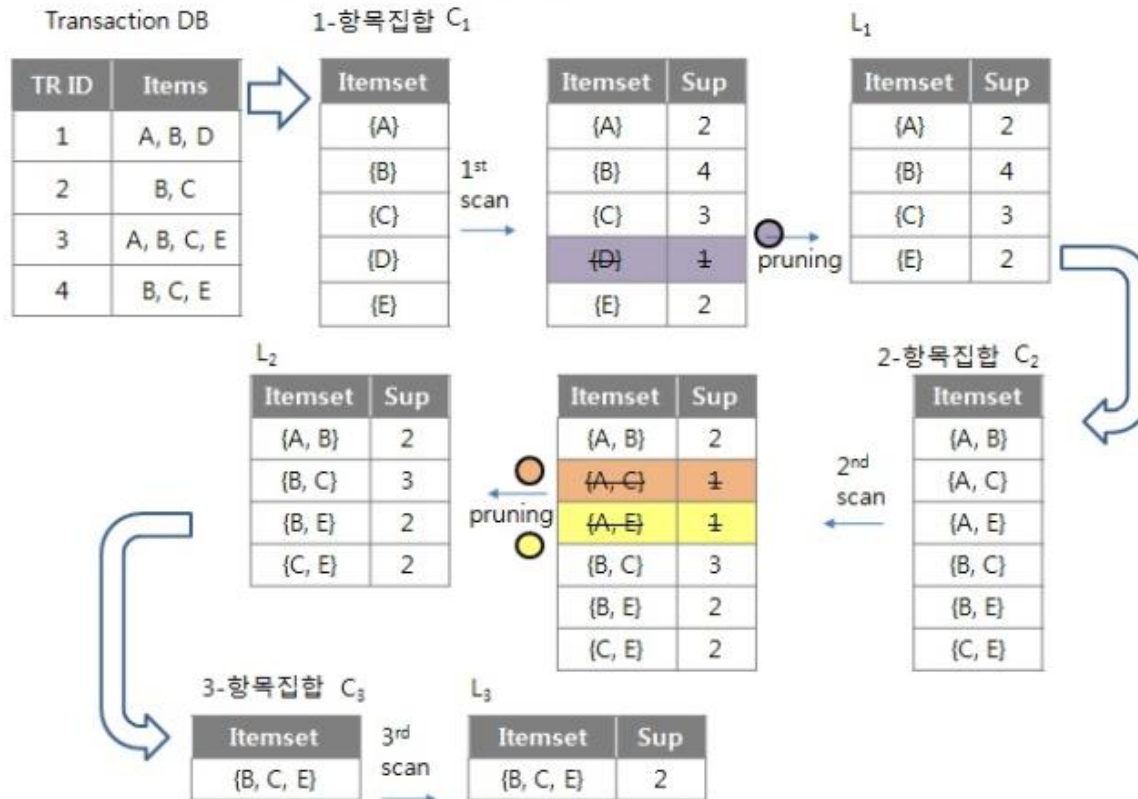
(Agrawal & Srikant @VLDB'94, Mannila, et al. @ KDD' 94)

제6절 연관분석

Apriori algorithm

- {A, B, C, D, E}의 5개 원소 항목을 가지는 4건의 transaction에서 minimum support 2건 (=2건/총4건=0.5) 기준으로 pruning 하는 예

▪ Pruning criteria (minimum support) : $\text{Sup}_{\min} = 2$



Apriori algorithm

● 빈발항목집합 추출 Pseudo Aprior algorithm

```
1)  $L_1 = \{\text{large 1-itemsets}\};$ 
2) for ( $k = 2; L_{k-1} \neq 0; k++$ ) do begin
3)    $C_k = \text{apriori-gen}(L_{k-1});$  // New candidates
4)   for all transactions  $t \in T$  do begin
5)      $C_t = \text{subset}(C_k, t);$  // Candidates contained in  $t$ 
6)     for all candidates  $c \in C_t$  do
7)        $c.\text{count}++;$ 
8)   end
9)    $L_k = \{c \in C_k \mid c.\text{count} \geq \text{min sup}\}$ 
10) end
11) Answer =  $\bigcup_k L_k;$ 
```

k-itemset	[Notation] An itemset having k items
L_k	Set of large k-itemsets (those with minimum support) Each member of this set has two fields: i) itemset and ii) support count.
C_k	Set of candidate k-itemsets (potentially large itemsets) Each member of this set has two fields: i) itemset and ii) support count
\overline{C}_k	Set of candidate k-itemsets when the TIDs of the generating transactions are kept associated with the candidates

연관성 분석 활용방안

- 장바구니 분석의 경우는 실시간 상품 추천을 통해 교차판매에 응용
- 순차 패턴 분석은 A를 구매한 사람인데 B를 구매하지 않은 경우, B를 추천하는 교차판매 캠페인에 사용

R을 이용한 연관분석 실습 코드

- **Groceries** : 식료품 판매점의 1달 동안의 POS 데이터이며, 총 169개의 제품과 9835건의 거래건수를 포함하고 있다.

```
>> install.packages("arulesViz")
```

```
...
```

다운로드된 바이너리 패키지들은 다음의 위치에 있습니다

C:\Users\Wk8s\AppData\Local\Temp\WRtmpEbR3rn\downloaded_packages

```
>> library(arulesViz)
```

```
...
```

```
>> data(Groceries)
```

```
>> inspect(Groceries[1:3])
```

items

[1] {citrus fruit, semi-finished bread, margarine, ready soups}

[2] {tropical fruit, yogurt, coffee}

[3] {whole milk}

```
>> rules <- apriori(Groceries, parameter = list(support=0.01, confidence = 0.3))
```

Apriori

Parameter specification:

confidence	minval	smax	arem	aval	originalSupport	maxtime	support	minlen	maxlen	target	ext
0.3	0.1	1	none	FALSE	TRUE	5	0.01	1	10	rules	TRUE

```
...
```

creating S4 object ... done [0.00s].

```
>>
```

R을 이용한 연관분석 실습 코드

- **Groceries** : 식료품 판매점의 1달 동안의 POS 데이터이며, 총 169개의 제품과 9835건의 거래건수를 포함하고 있다.

```
>> inspect(sort(rules, by=c("lift"), decreasing = TRUE)[1:20])
```

lhs	rhs	support	confidence	coverage
[1] {citrus fruit, other vegetables}	=>> {root vegetables}	0.01037112	0.3591549	0.02887646
[2] {tropical fruit, other vegetables}	=>> {root vegetables}	0.01230300	0.3427762	0.03589222
[3] {beef}	=>> {root vegetables}	0.01738688	0.3313953	0.05246568
[4] {citrus fruit, root vegetables}	=>> {other vegetables}	0.01037112	0.5862069	0.01769192
[5] {tropical fruit, root vegetables}	=>> {other vegetables}	0.01230300	0.5845411	0.02104728
[6] {other vegetables, whole milk}	=>> {root vegetables}	0.02318251	0.3097826	0.07483477
[7] {whole milk, curd}	=>> {yogurt}	0.01006609	0.3852140	0.02613116
[8] {root vegetables, rolls/buns}	=>> {other vegetables}	0.01220132	0.5020921	0.02430097
[9] {root vegetables, yogurt}	=>> {other vegetables}	0.01291307	0.5000000	0.02582613
[10] {tropical fruit, whole milk}	=>> {yogurt}	0.01514997	0.3581731	0.04229792
[11] {yogurt, whipped/sour cream}	=>> {other vegetables}	0.01016777	0.4901961	0.02074225
[12] {other vegetables, whipped/sour cream}	=>> {yogurt}	0.01016777	0.3521127	0.02887646
[13] {tropical fruit, other vegetables}	=>> {yogurt}	0.01230300	0.3427762	0.03589222
[14] {root vegetables, whole milk}	=>> {other vegetables}	0.02318251	0.4740125	0.04890696
[15] {whole milk, whipped/sour cream}	=>> {yogurt}	0.01087951	0.3375394	0.03223183
[16] {citrus fruit, whole milk}	=>> {yogurt}	0.01026945	0.3366667	0.03050330
[17] {onions}	=>> {other vegetables}	0.01423488	0.4590164	0.03101169
[18] {pork, whole milk}	=>> {other vegetables}	0.01016777	0.4587156	0.02216573
[19] {whole milk, whipped/sour cream}	=>> {other vegetables}	0.01464159	0.4542587	0.03223183
[20] {curd}	=>> {yogurt}	0.01728521	0.3244275	0.05327911

lift	count
[1] 3.295045	102
[2] 3.144780	121
...	
[19] 2.347679	144
[20] 2.325615	170

```
>>
```

『3과목』 데이터 분석

제5장 정형 데이터 마이닝-QnA





1

01. 다음 중 대용량 데이터 속에서 숨겨진 지식 또는 새로운 규칙을 추출해 내는 과정을 일컫는 것은?

- ① 지식경영
- ② 의사결정지원시스템
- ③ 데이터웨어하우징
- ④ 데이터마이닝

데이터 마이닝은 대용량 데이터에서 의미 있는 패턴을 파악하거나 예측하여 의사결정에 활용하는 방법이다.



2

02. 다음 중 기법의 활용 분야가 나머지와 다른 하나를 고르시오.

- ① 로지스틱 회귀 분석
- ② 인공신경망
- ③ 의사결정나무
- ④ **SOM**

SOM은 비지도 학습에 해당한다.



3

03. 아래 집단에 대해 지니지수는 얼마인가?

● ◆ ◆ ● ●

- ① 1
- ② 2
- ③ $\frac{1}{2}$
- ④ 12/25

$$G = 1 - \sum_i^c p_i^2$$

$$1 - (2/5)^2 - (3/5)^2$$



4

04. 아래는 오분류표를 나타낸 것이다. 다음 중 특이도는 얼마인가?

실제값 \ 예측치	True	False	합계
True	30	70	100
False	60	40	100
합계	90	110	200

- ① 3/10
- ② **4/10**
- ③ 13/20
- ④ 7/11

$$TN / TN + FP = 40 / 40 + 60$$

05. 다음 중 아래 (㉠)에서 설명하는 활성화함수로 가장 적절한 것은?

입력층이 직접 출력층에 연결되는 단층신경망에서 활성화함수를 (㉠)로 사용하면 로지스틱 회귀 모형과 작동원리가 유사해진다.

- ① 계단함수
- ② tanh 함수
- ③ ReLU 함수
- ④ 시그모이드 함수

6

06. 아래 데이터 셋 A,B 간의 유사성을 맨하튼거리로 계산하면 얼마인가?

	키	몸무게
A	165	65
B	170	70

- ① 25
- ② 20
- ③ 15
- ④ 10

$$d(x,y) = \sum_{i=1}^p |x_i - y_i|$$

$$|165 - 170| + |65 - 70|$$



7

07. 아래는 k-평균군집을 수행하는 절차를 단계별로 기술한 것이다. 다음 중 k-평균군집 수행 절차로 가장 올바른 것은?

- 가. 각 자료를 가장 가까운 군집 중심에 할당한다.
- 나. 군집 중심의 변화가 거의 없을 때 (또는 최대 반복 수)까지 단계2와 단계3을 반복한다.
- 다. 초기(군집의) 중심으로 k개의 객체를 임의로 선택한다.
- 라. 각 군집 내의 자료들의 평균을 계산하여 군집의 중심을 업데이트한다.

- ① 다 >> 라 >> 가 >> 나
- ② 가 >> 다 >> 라 >> 나
- ③ 가 >> 라 >> 다 >> 나
- ④ 다 >> 가 >> 라 >> 나

08. 아래는 쇼핑몰의 거래내역이다. 연관 규칙 "우유 >> 커피"에 대한 지지도는 얼마인가?

품목	거래건수
우유	10
커피	20
{우유, 커피}	30
{커피, 초코렛}	40
전체 거래 수	100

- ① 0.1
- ② 0.2
- ③ **0.3**
- ④ 0.4