06. R 정형데이터 분석 02
분류와 예측

성현곤

충북대학교 도시공학과
Dept.of URBAN ENGINEERING

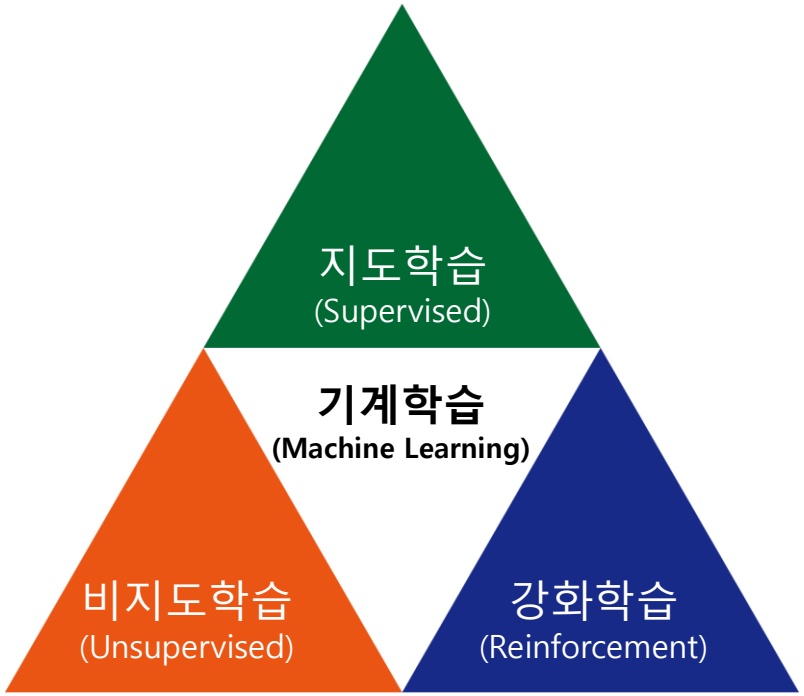
목차

분류와 기계학습의 개념

지도학습과 비지도학습

- 기계학습(Machine Learning)
  - 데이터를 이용해서 컴퓨터를 학습시키는 방법론
- 기계학습 알고리즘의 종류
  - 지도 학습(Supervised Learning)
    - 데이터에 대한 **레이블(Label)-명시적인 정답-이 주어진 상태에서 컴퓨터를 학습시키는 방법**
    - 학습된 알고리즘으로 얼마나 예측(Prediction)하는 데 사용
      - 예측하는 결과값이 **discrete value(이산값)면 classification(분류) 문제**
      - 결과값이 **continuous value(연속값)면 regression(회귀) 문제**
  - 비지도 학습(Unsupervised Learning)
    - 데이터에 대한 **레이블(Label)-명시적인 정답**-이 주어지지 상태에서 컴퓨터를 학습시키는 방법론
      - 예: **클러스터링(Clustering)**
    - **데이터의 숨겨진(Hidden) 특징(Feature)이나 구조를 발견**하는데 사용
  - 강화 학습(Reinforcement Learning)
    - **에이전트가 주어진 환경(state)에 대해 어떤 행동(action)을 취하고 이로부터 어떤 보상(reward)을 얻으면서 학습**을 진행하는 방법
    - 에이전트는 보상(reward)을 최대화(maximize)하도록 학습
    - 일종의 동적인 상태(dynamic environment)에서 데이터를 수집하는 과정까지 포함되어 있는 알고리즘
      - 예: Q-Learning, Deep-Q-Network(DQN) , 알파고

- Labeled data
- Direct feedback
- Predict outcome/future



- No labels
- No feedback
- "Find hidden structure

- Decision process
- Reward system
- Learn series of actions

출처: http://solarisailab.com/archives/1785

분류 알고리즘의 종류

- 분류 알고리즘(Classification Algorithms)
  - 훈련 데이터셋을 분석하여 목표 집단(target class)을 분류하고, 예측하는 것
    - 각 목표 집단을 결정하기 위하여 사용하여 더 적합한 분류의 경계 조건을 판단하기 위하여 훈련 데이터셋을 활용
    - 분류의 경계조건을 가지고 목표집단까지 예측하고 평가하는 과정을 모두 분류라고 함
  - 목표집단(target class)의 예
    - 고객 데이터(성별, 나이, 소득, 취미 등등)로 컴퓨터 부속품을 구매할 것인가?(목표집단: Yes/No)
    - 색상, 맛, 크기, 무게 등의 특징을 활용하여 어떤 과일인지 알 수 있을까?(목표집단: 사과, 배, 포도, 수박, 멜론 등)
    - 머리카락 길이로 남자인지 여자인지를 알 수 있을까?(목표집단: 남자, 여자)
  - 분류와 군집 알고리즘의 비교
    - 군집 알고리즘(clustering algorithms)
      - 분류에서 처럼 목표집단을 예측하지 않음
      - 가장 적합한 조건을 고려하여 유사한 종류(특징)을 집단화 함
      - 특징들에서 동일 집단에서는 유사하여야 하며, 집단간은 유사하지 않아야 함을 충족하게 하는 알고리즘

분류 알고리즘의 종류

- 분류 알고리즘 용어
  - **Classifier:** An algorithm that maps the input data to a specific category.
  - **Classification model:** 실험 데이터로 부터 어떤 결론들을 도출할 수 있는 분류 모델
    - 새로운 데이터에 대한 분류 수준 또는 항목을 예측하고, 평가하는 데 활용
  - **Feature:** A feature is an individual measurable property of a phenomenon being observed.
  - **Binary Classification:** 두 가지 가능한 수준의 분류
    - 예: 성별 분류
  - **Multi class classification:** 3개 이상의 분류로 각 샘플은 오로지 한 개의 집단(수준)에 할당되어짐
    - 예: 과일의 종류, 교통수단의 종류, 주택유형의 종류
  - **Multi label classification:** 각 표본이 한 개 이상의 목표 수준의 세트에 할당되어지는 분류
    - 뉴스 기사는 스포츠, 인사, 입지로 동시에 분류될 수 있음

출처: https://analyticsindiamag.com/7-types-classification-algorithms/

분류 알고리즘의 종류

- 분류 알고리즘의 응용분야
  - **스팸 이메일 분류**
  - **은행고객의 대출 반환 예측**
  - **암 또는 종양의 확인**
  - **감성(기분) 분석**
  - **개의 종류 분류**
  - **안면(홍채) 특징 인식 및 판단**
  - **자율주행차량에서의 보행자 인식 등**

분류 알고리즘의 종류

- 분류 알고리즘의 유형(종류)
  - 선형 분류 알고리즘
    - Logistic regression
    - Naïve Bayes classifier
    - Fisher's linear discriminant
  - 서포트 백터 머신(support vector machines)
    - Least squares support vector machines
- 다항 분류 알고리즘
  - Quadratic classifiers
- 커널 추정 알고리즘
  - K-nearest neighbor
- 결정나무 알고리즘
  - Decision tree
  - Random forests
- 신경망(Neural networks)
- .....

분류 모델의 평가(1)

- 모델 통계량: 회귀모델 정확성(accuracy) 진단
  - **Mean Absolute Error** (MAE)
    - 오차의 평균 절대값 차이로, 이상치(outliers)에 덜 민감하여 RMSE의 대안적 통계량
    - 낮을수록 더 적합한 모델로 평가
  - **MSE** (Mean Squared Error)
    - 평균 차이의 제곱
  - **Root Mean Squared Error** (RMSE)
    - 오차(실측값과 예측값의 차이)의 평균 제곱
    - 낮을수록 더 적합한 모델로 평가
  - R-squared, Adj. R-squared, McFadden's pseudo r-squared, Maximum likelihood pseudo r-squared 등
    - 높을 수록 더 적합한 모델로 평가
  - AIC, BIC 등
    - 더 낮을 수록 더 적합한 모델로 평가

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}|$$

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y})^2$$

$$RMSE = \sqrt{MSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y})^2}$$

$$R^2 = 1 - \frac{\sum(y_i - \hat{y})^2}{\sum(y_i - \bar{y})^2}$$

Where,

$\hat{y}$ – *predicted value of* y

$\bar{y}$ – *mean value of* y

출처: https://www.datatechnotes.com/2019/02/regression-model-accuracy-mae-mse-rmse.html

분류와 머신러닝의 개념

분류 모델의 평가(2)

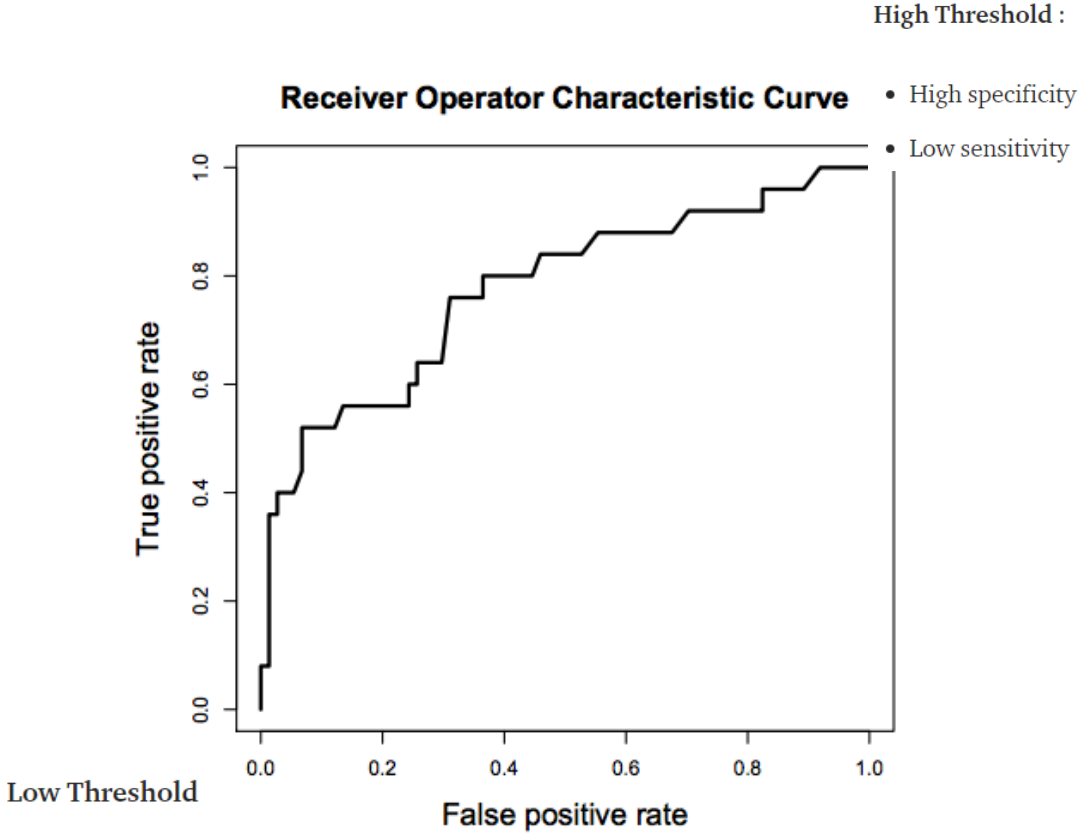
- 혼동행렬(confusion matrix)
  - 용어
    - **True positives (TP)**
    - **True negatives (TN)**
    - **False positives (FP)**
    - **False negatives (FN)**

n=165	**Predicted: NO**	**Predicted: YES**
Actual: NO	TN 50	FP 10
Actual: YES	FN 5	TP 100

메트릭	계산식	의미
Precison	$\frac{TP}{TP+FP}$	Y로 예측된 것 중 실제로도 Y인 비율
Accuracy	$\frac{TP+TN}{TP+FP+FN+TN}$	전체 예측에서(예측이 Y든 N이든 무관하게) 옳은 예측의 비율
Recall (Sensitivity, TP Rate, Hit Rate)	$\frac{TP}{TP+FN}$	실제로 Y인 것들 중 예측이 Y로 된 경우의 비율
Specificty	$\frac{TN}{FP+TN}$	실제로 N인 것들 중 예측이 N으로 된 경우의 비율
FP Rate (False Alarm Rate)	$\frac{FP}{FP+TN}$	Y가 아닌데 Y로 예측된 비율, 1-Specificity와 같은 값
F1 점수	$2 \times \frac{precision \cdot recall}{precision + recall}$	Precision와 Recall의 조화평균, 시스템의 성능을 하나로 수치로 표현하기 위해 사용하는 점수로, 0~1 사이의 값을 갖는다. Precision과 Recall 중 한 쪽만 클 때보다 두 값이 골고루 클 때 큰 값을 가진다.
Kappa	$k = \frac{accuracy - P(e)}{1 - P(e)}$	코헨의 카파는 두 평가자의 평가가 얼마나 일치하는지 평가하는 값으로 0~1 사이의 값을 가진다. P(e)는 두 평가자의 평가가 우연히 일치할 확률을 뜻하며, 코헨의 카파는 두 평가자의 평가가 우연히 일치할 확률을 제외한 뒤의 점수다.

분류 모델의 평가(2)

- ROC(Receiver Operator Characteristic) curve
  - 최선의 경계값(threshold value)을 결정 지원
  - False Positive Rate (x-axis)와 True Positive Rate (y-axis) 를 기반으로 그래프
    - 분류가 항상 0과 1사이에 존재하도록 하는 그래프
- **Area Under the Curve(AUC)**
  - 분류 성공률(ROC곡선 하단 면적



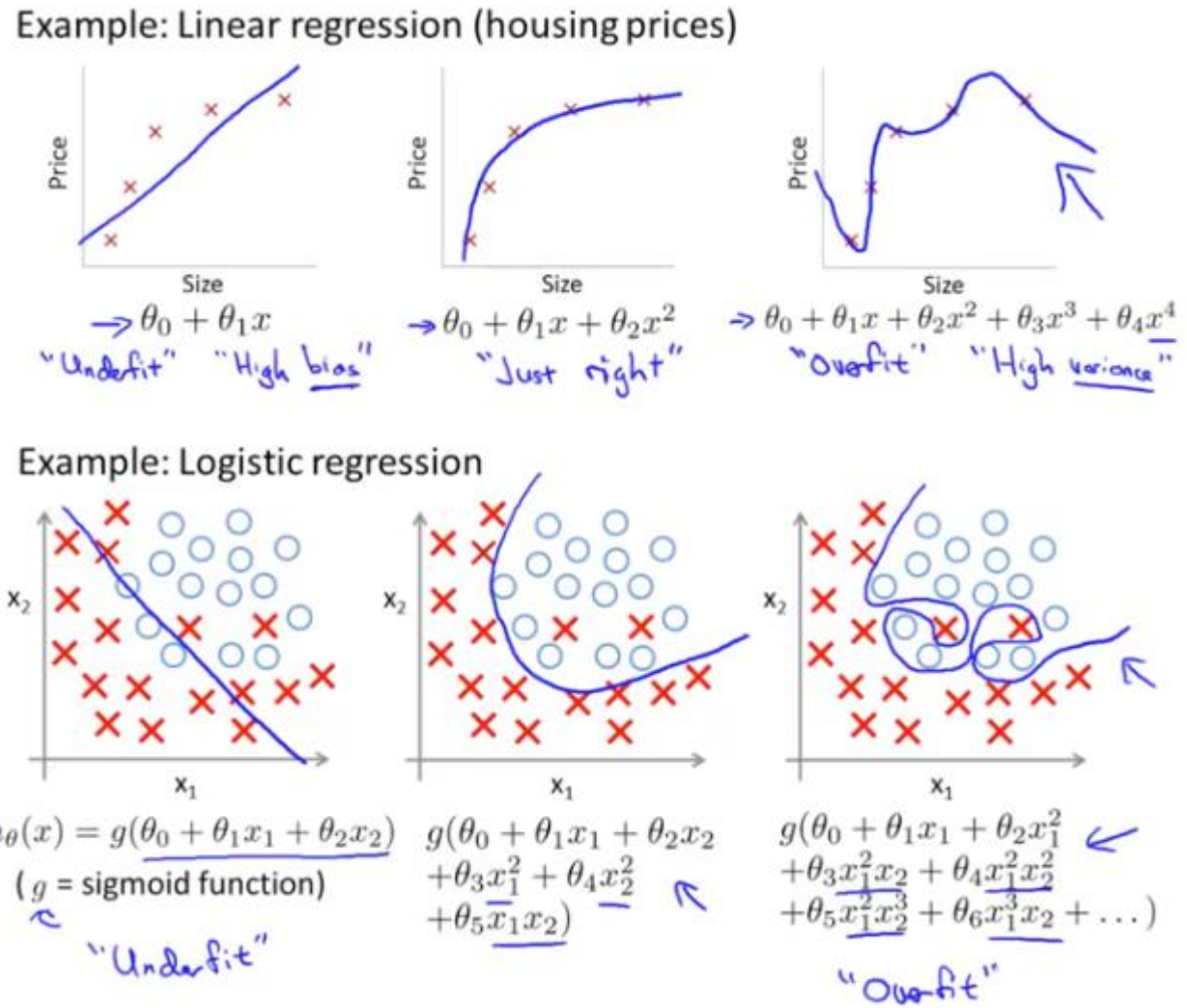
Area under curve (AUC) 값과 모델평가

- excellent = 0.9~1
- good = 0.8~0.9
- fair = 0.7~0.8
- poor = 0.6~0.7
- fail = 0.5~0.6

분류와 머신러닝의 개념

분류 모델의 평가 (3)

- 과대적합(over-fitting)과 과소적합(under-fitting)
  - 과대적합(over-fitting)
    - 샘플데이터로만 학습하여 새로운 데이터로 모델링할 경우 정확도가 낮아지는 문제
      - 범용성(robustness): 어떤 데이터를 넣어줘도 높은 정확도를 가지는 모델이 기계학습에서 매우 중요
  - 과소적합(under-fitting)
    - 데이터가 너무 적거나 학습이 제대로 일우저지 않아 Decision Boundary에 근접하지 못한 상태
- 과대적합 처리방법
  - 정규화(Regularization): 데이터를 일정한 규칙에 따라 변형하여 이용
    - 예측하고자 하는 특성과 연관성이 높은 특성의 경우 높은 값을 반대의 경우 낮은 값(θ)을 줌
    - θ 페널티를 주어 0에 가깝게 만듦으로써 예측에 불필요한 변수의 영향을 최소화하는 과정
  - 교차타당성(Cross Validation): 주어진 데이터의 일부를 학습시키고, 나머지 데이터로 모델을 검증
  - 더 많은 데이터 추가

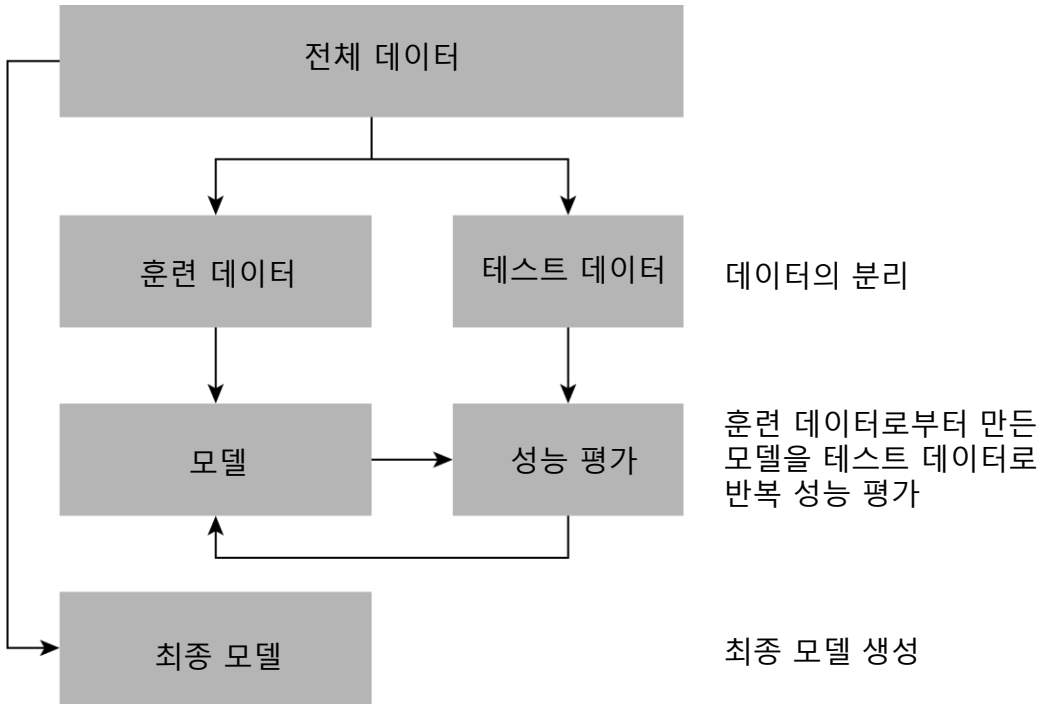


출처: https://daeson.tistory.com/184

교차타당성

- 교차타당성(Cross-validation test)
  - 훈련 데이터와 테스트 데이터를 분리하여 모델을 만드는 방법 중 가장 자주 사용하는 기법으로, 데이터를 다수의 조각으로 나누어 훈련과 테스트를 반복하는 기법
    - 데이터 중 일부는 모델을 만드는 훈련 데이터로 사용하고, 나머지 일부는 테스트 데이터로 사용해 모델을 평가
  - 과적합(over-fitting) 해소
    - 데이터 전체를 사용해 모델을 만들 경우, 해당 데이터에는 잘 동작하지만 새로운 데이터에는 좋지 않은 성능을 보이는 모델을 만들 가능성

- 교차타당성과 모델링 과정(절차)

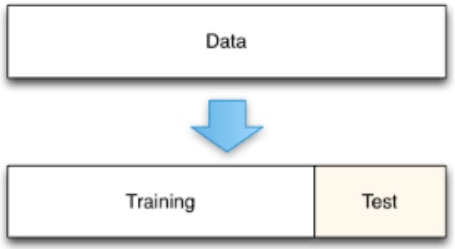


출처: https://medium.com/@hslee09/r-%EB%B6%84%EB%A5%98-%EC%95%8C%EA%B3%A0%EB%A6%AC%EC%A6%98-%EB%AA%A8%EB%8D%B8-%ED%8F%89%EA%B0%80-%EB%B0%A9%EB%B2%95-1a8f3c7913a3

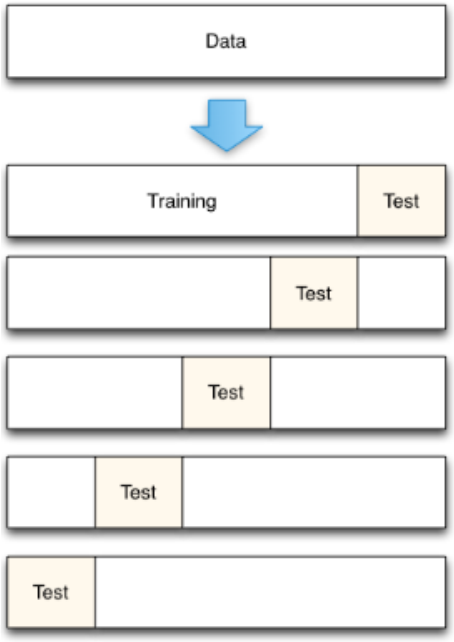
분류와 머신러닝의 개념

교차타당성

- 교차타당성(Cross-validation test)
  - Holdout Method
    - Validation set approach (or data split)
    - 일정 비율로 전체 데이터셋에서 훈련(training)시킬 데이터셋과 이 모델로 예측하여 성능을 평가할 시험(test) 데이터 셋으로 대별하는 방법
    - 20~30%를 시험 데이터셋으로 설정
    - 일부 데이터로 학습하여 bias 가능성 높음
  - Leave-One-Out Cross Validation(LPOCV) 또는 Leave-P-Out
    - 하나의 모델에 성능검증 데이터 포인트로 한 개 또는 모든 가능한 조합을 사용하는 방법
    - 실행시간이 길어짐
    - 다양한 결과로 인하여 분산이 높아질 수 있음
  - K-fold 교차타당성
    - 데이터를 훈련 데이터와 검증 데이터로 나누어 모델링 및 평가하는 작업을 K회 반복
    - Repeated k-fold Cross Validation



<Holdout method>



<k-fold method>

< LPOCV>

< Leave-P-Out >

출처: https://machine-learning-course.readthedocs.io/en/latest/content/overview/crossvalidation.html

교차타당성

- 교차타당성 검사(Cross-validation test)
  - K-fold 교차타당성의 장점
    - often gives more accurate estimates of the test error rate than does LOOCV (James et al. 2014).
  - K-fold 교차타당성 절차
    ① 무작위로 k-하위 데이터셋(예: 5개)을 분류
    ② 한 개의 하위 데이터셋을 제외하고 나머지로 모델 훈련
    ③ 남겨 둔 한 개의 데이터 셋으로 모델을 평가하고 예측 오차를 기록
    ④ K-개 하위데이터셋들의 각 시험 데이터셋으로 남겨질 때까지 ③의 과정을 반복
    ⑤ K-개의 기록된 오차의 평균을 계산
  - 하위데이터셋의 K-개 결정(how to choose right value of k?)
    - 작은 K값일수록 더 편의(bias)되고, 큰 k값일수록 덜 편의되지만 큰 변동성(variability)으로 바람직하지 않음
    - 실제로는 k-fold cross-validation에서 k = 5 또는 k = 10을 사용함
      - 이 값들은 과도한 편의 또는 변동성으로 문제되지 않을 정도의 오차율 추정치를 가지는 것으로 실증됨

출처: http://www.sthda.com/english/articles/38-regression-model-validation/157-cross-validation-essentials-in-r/

분류와 머신러닝의 개념

교차타당성

- 실습: “caret” 패키지

```
createDataPartition(y, times = 1, p = 0.5, list = TRUE,
  groups = min(5, length(y)))

createFolds(y, k = 10, list = TRUE, returnTrain = FALSE)

createMultiFolds(y, k = 10, times = 5)

createTimeSlices(y, initialWindow, horizon = 1, fixedWindow = TRUE,
  skip = 0)

groupKFold(group, k = length(unique(group)))

createResample(y, times = 10, list = TRUE)
```

`y`	a vector of outcomes. For `createTimeSlices`, these should be in chronological order.
`times`	the number of partitions to create
`p`	the percentage of data that goes to training
`list`	logical - should the results be in a list (`TRUE`) or a matrix with the number of rows equal to `floor(p * length(y))` and `times` columns.
`groups`	for numeric `y`, the number of breaks in the quantiles (see below)
`k`	an integer for the number of folds.
`returnTrain`	a logical. When true, the values returned are the sample positions corresponding to the data used during training. This argument only works in conjunction with `list = TRUE`
`initialWindow`	The initial number of consecutive values in each training set sample
`horizon`	the number of consecutive values in test set sample
`fixedWindow`	logical, if `FALSE`, all training samples start at 1
`skip`	integer, how many (if any) resamples to skip to thin the total amount
`group`	a vector of groups whose length matches the number of rows in the overall data set.

교차타당성

- 실습: “df.seoul.worker.csv”
  - 통근시간(분): commuting.time
  - 성별: gender
    - [1=남성, 2=여성]
  - 나이(만): age
  - 통근지역: commuting.area
    - [1. '현재 살고 있는 동 내', 2. '현재 살고 있는 구 내 다른 동', 3. '다른 구', 4. '다른 시도']

```
> ### 사용하게 될 패키지와 작업 폴더 파일 확인
> library(dplyr) # dplyr 패키지
> library(ggplot2) # 데이터 시각화 ggplot2 패키지
> library(caret)
> setwd("K:\\기타\\2019년2학기\\수치해석\\실습데이터") # 실습데이터가 있는 폴더로 작업폴더 변경
> getwd() # 현재 작업 중인 폴더 (변경) 확인
[1] "K:/기타/2019년2학기/수치해석/실습데이터"
> ?read.csv
> df <- read.table("df.seoul.worker.csv", header = TRUE, sep=",") # csv 파일 불러오기
> str(df) # 데이터 구조 확인
'data.frame':   26107 obs. of  51 variables:
 $ X                    : int  1 2 3 4 5 6 7 8 9 10 ...
 $ hh_id                : int  529 530 531 531 531 532 532 533 534 534 ...
 $ hhm.id               : int  1 1 2 3 1 2 1 1 1 2 ...
 $ com.mode             : int  2 2 2 2 2 2 2 1 1 2 ...
 $ com.2modes           : int  0 0 0 0 0 0 0 0 0 0 ...
 $ commuting.time       : int  30 20 20 20 20 40 30 20 20 30 ...
 $ commuting.area       : int  3 2 2 2 2 3 3 2 1 3 ...
 $ age                  : int  71 69 39 35 67 60 65 56 50 54 ...
 $ gender               : int  1 1 1 2 2 2 1 1 1 2 ...
```

```
> df2 <- df %>%  # 파이프 연산자
+   select(commuting.time, gender, age, commuting.area) # 4개 변수(열)만 추출
```

분류와 머신러닝의 개념

교차타당성

- 실습: “df.seoul.worker.csv”
  - 선형회귀모델: 통근시간의 결정요인 예측모형
    - 통근시간(분): commuting.time
    - 성별: gender
    - 나이(만): age
    - 통근지역: commuting.area
- Holdout Method
  - Validation set approach (or data split)

```
> ### Split the data into training and test set
> ?createDataPartition() # 비율 생성 함수
> ?set.seed() # 난수생성
> set.seed(123) # 123 코드로 난수생성
> training.samples <- df2$commuting.time %>% # df2의 통근시간
+   createDataPartition(p = 0.8, list = FALSE) # 비율 0.8로 데이터 분할 생성 명령
> train.data  <- df[training.samples, ] # 80% 훈련 데이터 생성
> test.data <- df[-training.samples, ] # 20% 시험 데이터 생성(-x)
> nrow(df2); nrow(train.data); nrow(test.data)
[1] 26107
[1] 20887
[1] 5220
```

```
### 2. Build the model
> model <- lm(commuting.time ~ # 선형회귀모델: 종속변수=통근시간
+               as.factor(gender) + # 설명변수: gender를 요인으로
+               age + # 설명변수: age 연속변수
+               as.factor(commuting.area) # 설명변수: 요인으로
+               , data = train.data) # 훈련 데이터셋
> summary(model)

Call:
lm(formula = commuting.time ~ as.factor(gender) + age + as.factor(comm
    data = train.data)

Residuals:
    Min      1Q  Median      3Q     Max
-53.898  -7.581  -3.014   5.866 179.863

Coefficients:
                             Estimate Std. Error t value Pr(>|t|)
(Intercept)                 20.971909   0.521318  40.229  < 2e-16 ***
as.factor(gender)2          -0.958500   0.209525  -4.575  4.8e-06 ***
age                         -0.104472   0.008106 -12.888  < 2e-16 ***
as.factor(commuting.area)2   9.952378   0.292045  34.078  < 2e-16 ***
as.factor(commuting.area)3  28.577074   0.290743  98.290  < 2e-16 ***
as.factor(commuting.area)4  45.015554   0.471589  95.455  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 14.17 on 20881 degrees of freedom
Multiple R-squared:  0.4775,    Adjusted R-squared:  0.4774
F-statistic:  3817 on 5 and 20881 DF,  p-value: < 2.2e-16

> ### 3.  Make predictions and compute the R2, RMSE and MAE
> predictions <- model %>% # 훈련데이터 모델 결과
+   predict(test.data) # 시험 데이터 예측값 생성
> data.frame( R2 = R2(predictions, test.data$commuting.time),
+             RMSE = RMSE(predictions, test.data$commuting.time),
+             MAE = MAE(predictions, test.data$commuting.time))
         R2     RMSE      MAE
1 0.4897574 13.64557 10.33002
```

교차타당성

- 실습: “df.seoul.worker.csv”
  - 선형회귀모델: 통근시간의 결정요인 예측모형
    - 통근시간(분): commuting.time
    - 성별: gender
    - 나이(만): age
    - 통근지역: commuting.area
- Leave one out cross validation(LOOCV)
  - ?trainControl() # Control parameters for train

Arguments

`method`	The resampling method: `"boot"`, `"boot632"`, `"optimism_boot"`, `"boot_all"`, `"cv"`, `"repeatedcv"`, `"LOOCV"`, `"LGOCV"` (for repeated training/test splits), `"none"` (only fits one model to the entire training set), `"oob"` (only for random forest, bagged trees, bagged earth, bagged flexible discriminant analysis, or conditional tree forest models), `timeslice`, `"adaptive_cv"`, `"adaptive_boot"` or `"adaptive_LGOCV"`
`number`	Either the number of folds or number of resampling iterations
`repeats`	For repeated k-fold cross-validation only: the number of complete sets of folds to compute
`p`	For leave-group out cross-validation: the training percentage

분류와 머신러닝의 개념

교차타당성

- 실습: “df.seoul.worker.csv”
  - 선형회귀모델: 통근시간의 결정요인 예측모형
    - 통근시간(분): commuting.time
    - 성별: gender
    - 나이(만): age
    - 통근지역: commuting.area
- Leave one out cross validation(LOOCV)
  - ?trainControl() # Control parameters for train

```
> ## Leave one out cross validation - LOOCV
> ### Define training control
> ?trainControl()
> ### Train the model
> ?train()
> model <- train(commuting.time ~ # 선형회귀모델: 종속변수=통근시간
+                  as.factor(gender) + # 설명변수: gender를 요인으로
+                  age + # 설명변수: age 연속변수
+                  as.factor(commuting.area), # 설명변수: 요인으로,
+                data = df2, # 사용할 데이터셋
+                method = "lm", # 선형회귀모델
+                trControl = train.control) # 훈련데이터 통제방법
> ### Summarize the results
> print(model)
Linear Regression

26107 samples
    3 predictor

No pre-processing
Resampling: Leave-One-Out Cross-Validation
Summary of sample sizes: 26106, 26106, 26106, 26106, 26106, 26106, ...
Resampling results:

  RMSE      Rsquared  MAE
  14.06854  0.479595  10.47255

Tuning parameter 'intercept' was held constant at a value of TRUE
```

시간이 오래 걸림

분류와 머신러닝의 개념

교차타당성

- 실습: “df.seoul.worker.csv”
  - 선형회귀모델: 통근시간의 결정요인 예측모형
    - 통근시간(분): commuting.time
    - 성별: gender
    - 나이(만): age
    - 통근지역: commuting.area
- K-fold cross-validation
  - ?trainControl() # Control parameters for train

```
> ## K-fold cross-validation
> ### 1. Define training control
> set.seed(123) # 난수 생성
> train.control <- trainControl(method = "cv", # cross validation
+                               number = 5) # 5-fold 설정
> ### 2. Train the model
> model <- train(commuting.time ~ # 선형회귀모델: 종속변수=통근시간
+                  as.factor(gender) + # 설명변수: gender를 요인으로
+                  age + # 설명변수: age 연속변수
+                  as.factor(commuting.area), # 설명변수: 요인으로,
+                data = df2, # 사용할 데이터셋
+                method = "lm", # 선형회귀모델
+                trControl = train.control) # 훈련데이터 통제방법
> ### 3. Summarize the results
> print(model)
Linear Regression

26107 samples
    3 predictor

No pre-processing
Resampling: Cross-Validated (5 fold)
Summary of sample sizes: 20885, 20885, 20886, 20885, 20887
Resampling results:

  RMSE      Rsquared   MAE
  14.06594  0.4800299  10.472

Tuning parameter 'intercept' was held constant at a value of TRUE
```

교차타당성

- 실습: “df.seoul.worker.csv”
  - 선형회귀모델: 통근시간의 결정요인 예측모형
    - 통근시간(분): commuting.time
    - 성별: gender
    - 나이(만): age
    - 통근지역: commuting.area
- Repeated K-fold cross-validation
  - ?trainControl() # Control parameters for train

```
> ## Repeated K-fold cross-validation
> ### 1. Define training control
> set.seed(123)
> train.control <- trainControl(method = "repeatedcv", # 훈련데이터 통제 = "repeatedcv"
+                               number = 5, # 5-fold 설정
+                               repeats = 3) # 반복횟수 설정
> ### 2. Train the model
> model <- train(commuting.time ~ # 선형회귀모델: 종속변수=통근시간
+                  as.factor(gender) + # 설명변수: gender를 요인으로
+                  age + # 설명변수: age 연속변수
+                  as.factor(commuting.area), # 설명변수: 요인으로,
+                data = df2, # 사용할 데이터셋
+                method = "lm", # 선형회귀모델
+                trControl = train.control) # 훈련데이터 통제방법
> ### 3. Summarize the results
> print(model)
Linear Regression

26107 samples
    3 predictor

No pre-processing
Resampling: Cross-Validated (5 fold, repeated 3 times)
Summary of sample sizes: 20885, 20885, 20886, 20885, 20887, 20885, ...
Resampling results:

  RMSE     Rsquared   MAE
  14.0657  0.4799255  10.47177

Tuning parameter 'intercept' was held constant at a value of TRUE

> print(model$results)
  intercept    RMSE  Rsquared      MAE    RMSESD  RsquaredSD     MAESD
1      TRUE 14.0657 0.4799255 10.47177 0.2021135 0.007752879 0.1054807
```

연습문제 01

- 분류 모델의 결과로 다음과 같은 혼동행렬의 값을 얻었다. 아래의 질문에 답하시오.

		예측	
		No	Yes
실측	No	1000 TN	150 FP
	Yes	50 FN	10000 TP

- Precision =?
- Accuracy =?
- Recall =?
- Specificity =?
- FP Rate =?
- F1 score =?

메트릭	계산식
Precision	$\frac{TP}{TP+FP}$
Accuracy	$\frac{TP+TN}{TP+FP+FN+TN}$
Recall (Sensitivity, TP Rate, Hit Rate로도 부름)	$\frac{TP}{TP+FN}$
Specificity	$\frac{TN}{FP+TN}$
FP Rate (False Alarm Rate로도 부름)	$\frac{FP}{FP+TN}$
F1 점수[14]	$2\times\frac{1}{\frac{1}{precision}+\frac{1}{recall}}=2\times\frac{precision\cdot recall}{precision+recall}$

연습문제 02

- caret패키지의 createDataPartition()함수로 "df.seoul.worker.csv" 데이터를 불러들여 df 객체에 할당 한 후 com.2modes 변수를 기준으로 훈련데이터셋(train.data)을 70%로, 시험데이터(test.data)으로 30%를 무작위로 추출하고자 한다.
  - 이렇게 하였을 경우에 전체 데이터의 관측치수, 훈련데이터의 관측치수, 시험데이터의 관측치의 갯수는 얼마인가?

이항 로지스틱 모델

- 로지스틱 회귀(logistic regression)
  - 독립 변수의 선형 결합을 이용하여 사건의 발생 가능성을 예측하는데 사용되는 통계 기법
  - 일반적인 회귀 분석의 목표와 동일하게 종속 변수와 독립 변수간의 관계를 구체적인 함수로 나타내어 향후 예측 모델에 사용하는 것
  - 종속 변수가 범주형 데이터를 대상으로 하며 입력 데이터가 주어졌을 때 해당 데이터의 결과가 특정 분류로 나뉘기 때문에 일종의 분류 (classification) 기법으로도 볼 수 있음
- 로지스틱 회귀모델의 종류
  - 이항 로지스틱 모델(binomial logistic model):
    - 이항형 문제(즉, 유효한 범주의 개수가 두개인 경우)
  - 다항 또는 분화 로지스틱 모델(Multinomial or polytomous logistic model):
    - 두 개 이상의 범주를 가지는 문제가 대상인 경우
  - 순서 로지스틱 모델(ordinal logistic model):
    - 복수의 범주이면서 순서가 존재

이항 로지스틱 모델

- 이항로지스틱 회귀모델의 응용분야
  - (보건의료) 어떤 사람들이 어떤 특정 암이나 종양 질환을 앓게 되는가?
  - (도시 및 교통계획) 어떤 직장인들이 승용차를 이용하는가?
  - (자동차 산업) 자율주행차량의 운행 중 보행자를 어떻게 식별하여 교통사고를 회피할 것인가?
  - (마케팅) 어떤 사람들이 주로 최신 스마트폰을 구매하는가?
  - (금융) 어떤 사람들이 대출 반환을 하지 않을 것인가?
  - ……

이항 로지스틱 모델

- 이항 로지스틱 함수
  - 로지스틱 모형
    - 이항형 로지스틱의 회귀 분석에서 2개의 카테고리는 0과 1이며, 각각의 카테고리로 분류될 확률(p)의 합은 1이다
    - 연속이고 증가함수이며 [0,1]에서 값을 갖는 연결 함수 y(x)
    - 로지스틱 회귀의 계수 추정은 최대 가능도(maximum likelihood) 방법을 이용 $\frac{e^x}{1+e^x}$
  - 이항 로지스틱 함수
    - 오즈(odds):
      - 성공 확률이 실패 확률에 비하여 몇 배 더 높은가? $\text{odds} = \frac{p(y=1|x)}{1-p(y=1|x)}$
    - 로짓 변환:
      - 오즈에 로그를 취한 함수로서 입력 값의 범위가 [0,1] 일 때 출력 값의 범위를 무한대의 범위로 조정 $\text{logit}(p) = \log \frac{p}{1-p}$
    - 로지스틱 함수 (logistic function):
      - 독립 변수 x가 주어졌을 때 종속 변수가 1의 범주에 속할 확률 $\text{logistic function} = \frac{e^{\beta \cdot X_i}}{1+e^{\beta \cdot X_i}}$

$$\text{logit}(\mathbb{E}[Y_i|x_{1,i},\cdots,x_{m,i}]) = \text{logit}(p_i) = \ln\frac{p_i}{1-p_i}$$

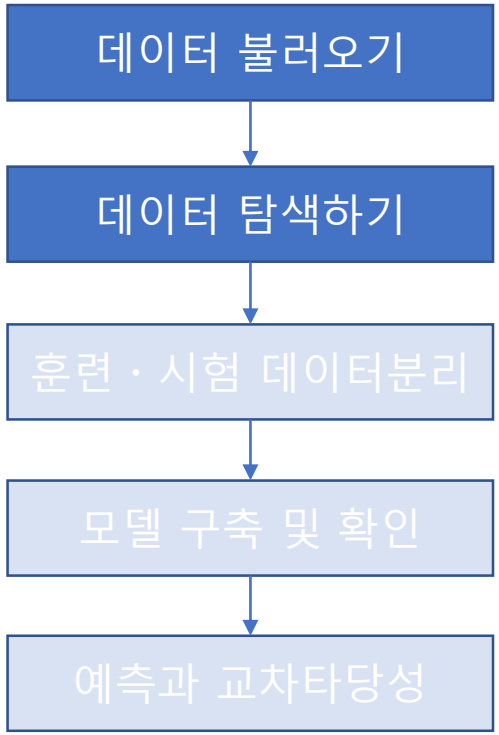
$$\text{logit}(p_i) = \beta_0 + \beta_1 x_{1,i} + \cdots + \beta_m x_{m,i} = \beta \cdot X_i$$

$$\ln\frac{p_i}{1-p_i} = \beta \cdot X_i$$

$$p_i = \text{logit}^{-1}(\beta \cdot \mathbf{X}_i) = \frac{1}{1+e^{-\beta \cdot \mathbf{X}_i}}$$

이항 로지스틱 모델

- 실습 1: 통근시간의 특성에 따라 통근수단(승용차=1, 기타 수단 =0) 중 승용차를 선택할 확률 결정




```
> ## 1. 데이터 불러오기와 탐색
> ?read.csv
> df <- read.csv("df.seoul.worker.csv")
> str(df)
'data.frame':   26107 obs. of  51 variables:
 $ X                 : int  1 2 3 4 5 6 7 8 9 10 ...
 $ hh_id             : int  529 530 531 531 531 532 532 533 534 534 .
 $ hhm.id            : int  1 1 2 3 1 2 1 1 1 2 ...
 $ com.mode          : int  2 2 2 2 2 2 2 1 1 2 ...
 $ com.2modes        : int  0 0 0 0 0 0 0 0 0 0 ...
 $ commuting.time    : int  30 20 20 20 20 40 30 20 20 30 ...
 $ commuting.area    : int  3 2 2 2 2 3 3 2 1 3 ...
 $ age               : int  71 69 39 35 67 60 65 56 50 54 ...
 $ gender            : int  1 1 1 2 2 2 1 1 1 2 ...

> attach(df) # 데이터 객체 바로 접근하기

> table(com.2modes) # 통근수단 1=승용차, 0= 기타
com.2modes
    0     1
19021  7086
> df %>%
+   group_by(com.2modes) %>%  # 통근수단별 집단화
+   summarize(m_time = mean(commuting.time), # 평균 통근시간
+             sd_time = sd(commuting.time), # 표준편차
+             max_time = max(commuting.time), # 최대값
+             min_time = min(commuting.time),) # 최소값
# A tibble: 2 x 5
  com.2modes m_time sd_time max_time min_time
       <int>  <dbl>   <dbl>    <int>    <int>
1          0   32.5    19.0      180        1
2          1   37.8    20.3      240        5
```

분류와 예측

이항 로지스틱 모델

- 실습 1: 통근시간의 특성에 따라 통근수단(승용차=1, 기타 수단 =0) 중 승용차를 선택할 확률 결정

데이터 불러오기

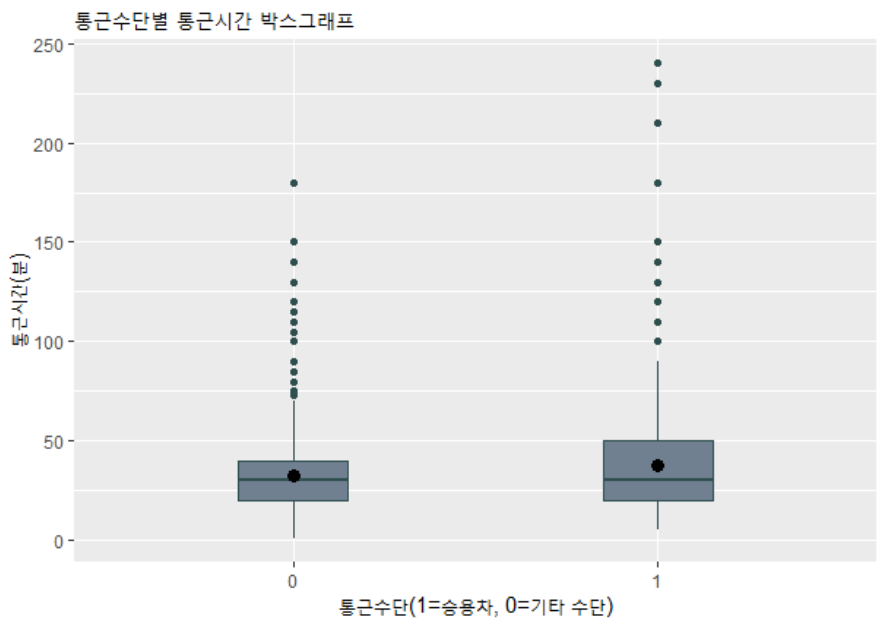
데이터 탐색하기

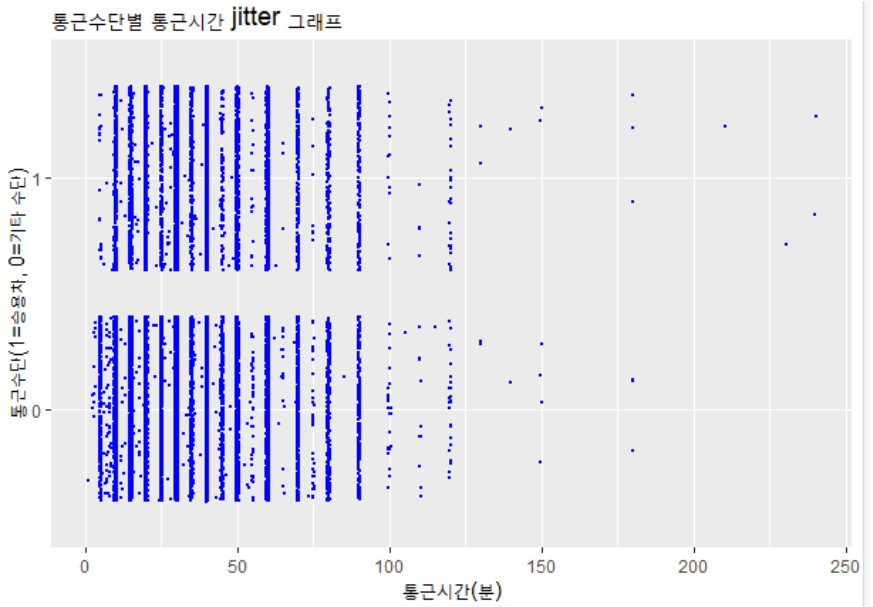
훈련 · 시험 데이터분리

모델 구축 및 확인

예측과 교차타당성

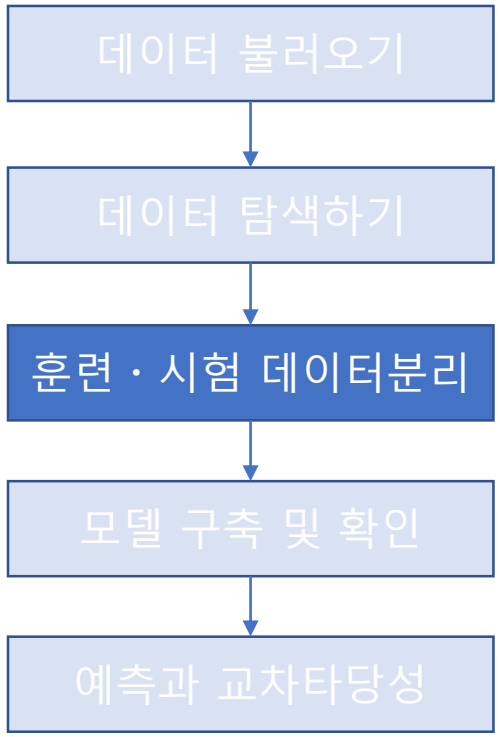
```
> ggplot(data=df, aes(x=as.factor(com.2modes), y= commuting.time,group=com.2modes))+
+   geom_boxplot(fill='slategrey',color='darkslategrey',width=0.3) +
+   stat_summary(fun.y="mean", geom="point", size=3, fill="blue") +
+   labs(title = "통근수단별 통근시간 박스그래프",
+        x = "통근수단(1=승용차, 0=기타 수단)",
+        y = "통근시간(분)")
> ggplot(data=df, aes(y=as.factor(com.2modes), x= commuting.time))+
+   geom_jitter(cex=0.3, col="blue") +
+   labs(title = "통근수단별 통근시간 박스그래프",
+        x = "통근시간(분)",
+        y = "통근수단(1=승용차, 0=기타 수단)")
```





이항 로지스틱 모델

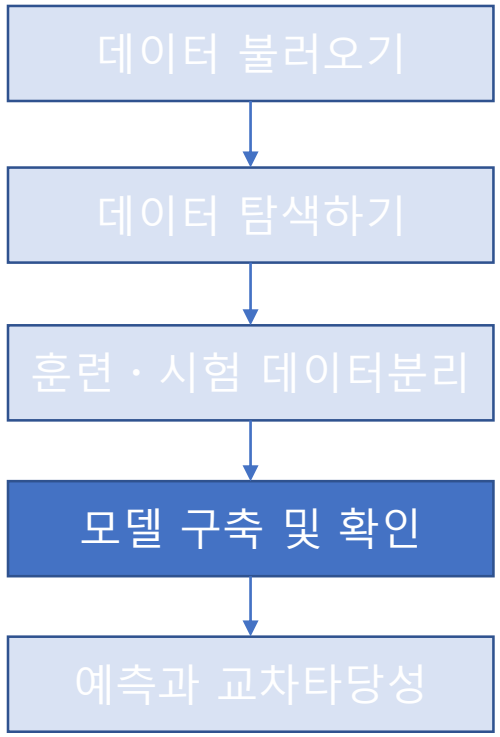
- 실습 1: 통근시간의 특성에 따라 통근수단(승용차=1, 기타 수단 =0) 중 승용차를 선택할 확률 결정




```
> ## 2.데이터 분리(구분)
> ### Split the data into training and test set
> ?floor()
> sample.size = floor(0.80*nrow(df)) # largest integers not greater than x
> set.seed(123) # 난수생성
> ?sample() # Random Samples, x 객체, size = 추출할 갯수
> train.id <- sample(seq_len(nrow(df)), # 1, 2, ...., n까지의 번호 생성, 무작위 표본 추출
+                    size = sample.size)  # 무작위 표본 추출, size = 추출할 갯수(p=0.8)
> train <- df[train.id,] # 훈련 데이터
> test <- df[-train.id,] # 시험 데이터
> nrow(train); nrow(test) # 추출된 객체 갯수 확인
[1] 20885
[1] 5222
```

이항 로지스틱 모델

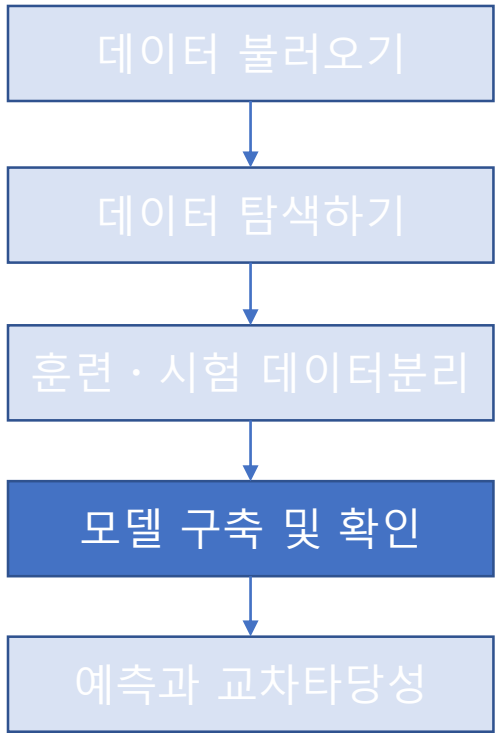
- 실습 1: 통근시간의 특성에 따라 통근수단(승용차=1, 기타 수단 =0) 중 승용차를 선택할 확률 결정




```
> ## 3. Logistic Regression Model
> ?glm() # Fitting Generalized Linear Models
> m.slogit <- glm(com.2modes ~ # 통근수단: 1=승용차, 0=기타수단
+                   commuting.time, # 통근시간 설명변수
+                   family = binomial, # 이항로짓모형 설정
+                   data = train) # 데이터셋
> summary(m.slogit) # 결과 요약

Call:
glm(formula = com.2modes ~ commuting.time, family = binomial,
    data = train)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.6053  -0.8178  -0.7278   1.4041   1.7984

Coefficients:
                Estimate Std. Error z value Pr(>|z|)
(Intercept)    -1.4633140  0.0323052  -45.30   <2e-16 ***
commuting.time  0.0134957  0.0007862   17.16   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 24387  on 20884  degrees of freedom
Residual deviance: 24092  on 20883  degrees of freedom
AIC: 24096
                    > round(summary(m.slogit)$coef, 4) # 회귀계수 소숫점 4자리로 반올림
                                  Estimate Std. Error  z value Pr(>|z|)
                    (Intercept)     -1.4633     0.0323 -45.2965        0
                    commuting.time   0.0135     0.0008  17.1649        0
                    > round(exp(m.slogit$coef), 4) # 오즈비(odds ratio) 계산
                       (Intercept) commuting.time
                            0.2315         1.0136
```

이항 로지스틱 모델

- 실습 1: 통근시간의 특성에 따라 통근수단(승용차=1, 기타 수단 =0) 중 승용차를 선택할 확률 결정



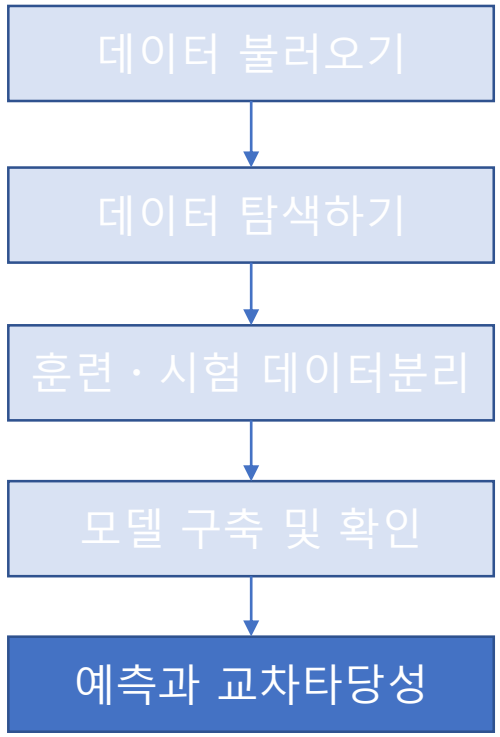

```
> anova(m.slogit, test="Chisq") # 귀무모형과 모형 비교 진단
Analysis of Deviance Table

Model: binomial, link: logit

Response: com.2modes

Terms added sequentially (first to last)


              Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
NULL                          20884      24387
commuting.time  1   294.22     20883      24092 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> library(pscl)
> ?pR2() # compute various pseudo-R2 measures
> round(pR2(m.slogit), 4) # 모형 통계량 산출
       llh     llhNull          G2    McFadden        r2ML        r2CU
-12046.2207 -12193.3284    294.2155      0.0121      0.0140      0.0203
```

모형통계량

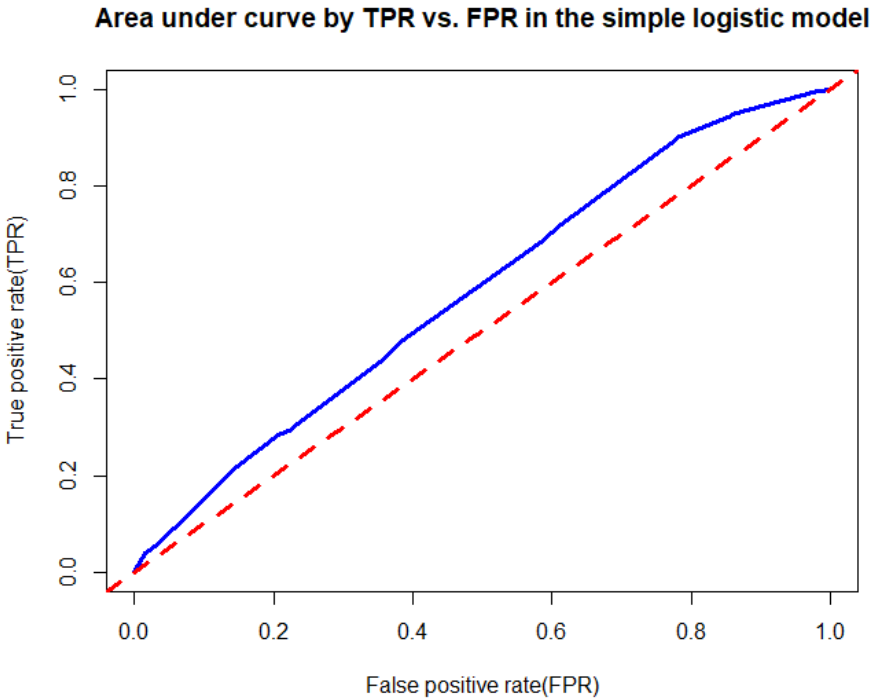
- Llh = The log-likelihood from the fitted model,
- llhNull = The log-likelihood from the intercept-only restricted model
- G2 = Minus two times the difference in the log-likelihoods,
- McFadden = McFadden's pseudo r-squared
- r2ML = Maximum likelihood pseudo r-squared,
- r2CU = Cragg and Uhler's pseudo r-squared

이항 로지스틱 모델

- 실습 1: 통근시간의 특성에 따라 통근수단(승용차=1, 기타 수단 =0) 중 승용차를 선택할 확률 결정




```
> ## 4. Making predictions on Training set
> library(ROCR)
> p <- predict(m.slogit, newdata=test, type="response")
> ?prediction()
> ?performance() # Function to create performance objects
> ## 4. Making predictions on Training set
> library(ROCR) # Function to create prediction objects
> p <- predict(m.slogit, # 훈련모델
+              newdata=test, # 시험 데이터셋
+              type="response")  # 종속변수 예측
> ?prediction() # Function to create prediction objects
> pr <- prediction(p, # 시험데이터로 부터 예측된 예측값
+                  test$com.2modes) # 시험 데이터의 실측값
> ?performance() # 인수 내용 확인 필요: Function to create perfor
> prf <- performance(pr, measure = "tpr", x.measure = "fpr")
> plot(prf,
+      main = "Area under curve by TPR vs. FPR in the simple l
+      xlab = "False positive rate(FPR)",
+      ylab = "True positive rate(TPR)",
+      lwd=3,
+      col="blue")
> abline(a=0, b=1, col="red", lwd=3, lty=2)
> auc <- performance(pr, measure = "auc") # Area under the ROC
> # This is equal to the value of the Wilcoxon-Mann-Whitney te
> # and also the probability that the classifier will score ar
> auc <- auc@y.values[[1]]
> auc
[1] 0.5808932
```



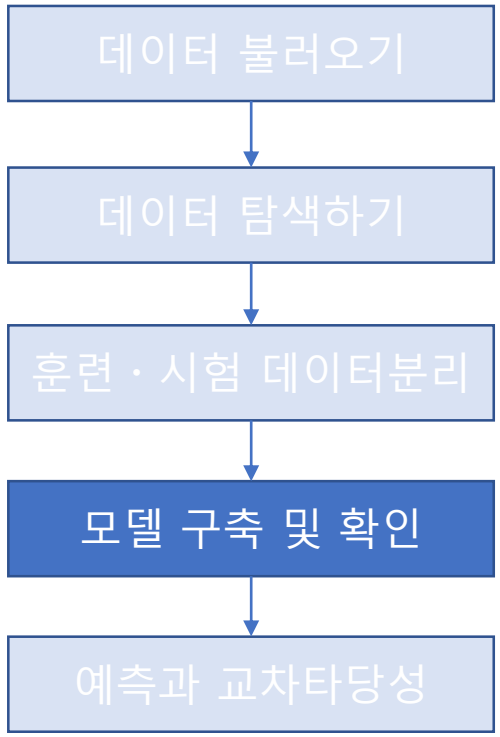
Area under curve (AUC) 값과 모델평가

- excellent = 0.9~1
- good = 0.8~0.9
- fair = 0.7~0.8
- poor = 0.6~0.7
- fail = 0.5~0.6

- 모델의 성능 향상을 위한 작업
  - ① 적당한 설명 변수를 입력 하는 것
  - ② 적절한 모델/알고리즘을 선택하는 것
  - ③ 위 1~2번을 반복

이항 로지스틱 모델

- 실습 2: 다양한 특성들에 의하여 직장인들이 통근수단 중 승용차를 선택할 확률 결정




```
> ## 실습 2: 다양한 특성들에 의하여 직장인들이 통근수단(승용차=1, 기타 수단 =0) 중 승용차를 선택할 확률 결정
> ### 3. 훈련 모델(데이터 탐색과 데이터 분리는 앞서와 동일)
> m.mlogit <- glm(com.2modes ~ # 통근수단 종속변수
+                   commuting.time + # 통근시간(분)
+                   as.factor(commuting.area) + # 통근지역(1.'현재 살고 있는 동 내', 2. '현재 살고 있는 구
+                   as.factor(gender) + # 성별(1=남성, 2=여성)
+                   age + # 만 나이
+                   edu.level + # 교육수준( 1 '중학교이하', 2 '고졸이하', 3 '대졸이하', 4 '대학원이상')
+                   as.factor(job.type) + # 직업유형(1'관리전문직', 2 '화이트칼라', 3 '블루칼라', 4 '기타)
+                   hhm.no + # 가구원 수
+                   as.factor(h.type) +  # 주택유형(1.단독, 2.아파트, 3.다세대, 4.연립/빌라, 5.기타)
+                   hh.income + # 가구총소득
+                   as.factor(area.living) + # 서울거주지역(1.도심, 2.동북, 3.서북, 4.서남, 5.동남)
+                   feel.walk.reside  +  # 거주지 걷기와 달리기 적합도
+                   feel.bus + # 버스이용환경 만족도
+                   feel.subway, # 지하철 이용환경 만족도
+                 family = binomial,
+                 data = train)
> summary(m.mlogit) # 모형 결과 요약
```

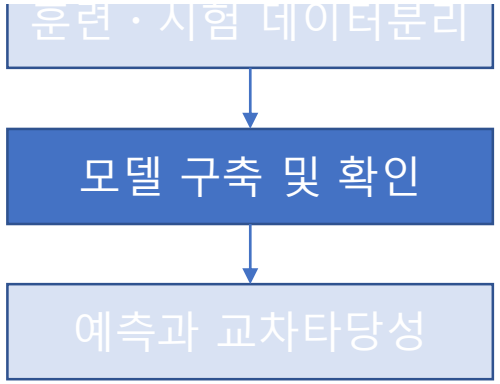
결과는 뒤에.....

분류와 예측

이항 로지스틱 모델

- 실습 2: 다양한 특성들에 의하여 직장인들이 [illegible]

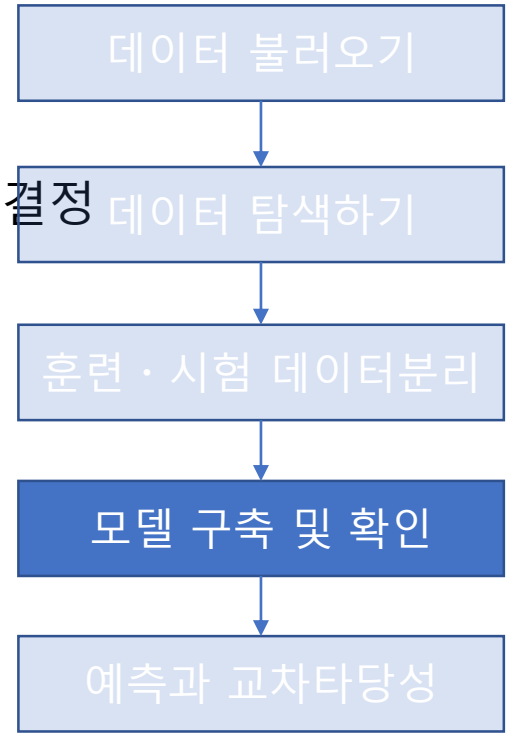
```
m.2modes ~ # 통근수단 종속변수
commuting.time + # 통근시간(분)
as.factor(commuting.area) + # 통근지역(1.'현재 살고 있는 동 내', 2. '현재 살고 있는 구
as.factor(gender) + # 성별(1=남성, 2=여성)
age + # 만 나이
edu.level + # 교육수준( 1 '중학교이하', 2 '고졸이하', 3 '대졸이하', 4 '대학원이상')
as.factor(job.type) + # 직업유형(1'관리전문직', 2 '화이트칼라', 3 '블루칼라', 4 '기타)
hhm.no + # 가구원 수
as.factor(h.type) +  # 주택유형(1.단독, 2.아파트, 3.다세대, 4.연립/빌라, 5.기타)
hh.income + # 가구총소득
as.factor(area.living) + # 서울거주지역(1.도심, 2.동북, 3.서북, 4.서남, 5.동남)
feel.walk.reside  +  # 거주지 걷기와 달리기 적합도
feel.bus + # 버스이용환경 만족도
feel.subway, # 지하철 이용환경 만족도
```




```
> summary(m.mlogit) # 모형 결과 요약

Call:
glm(formula = com.2modes ~ commuting.time + as.factor(commuting.area) +
    as.factor(gender) + age + edu.level + as.factor(job.type) +
    hhm.no + as.factor(h.type) + hh.income + as.factor(area.living) +
    feel.walk.reside + feel.bus + feel.subway, family = binomial,
    data = train)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.4106  -0.7806  -0.4749   0.7657   2.9482

Coefficients:
                          Estimate Std. Error z value Pr(>|z|)
(Intercept)              -6.332889   0.231494 -27.357  < 2e-16 ***
commuting.time           -0.014336   0.001221 -11.740  < 2e-16 ***
as.factor(commuting.area)2  1.675121   0.075181  22.281  < 2e-16 ***
as.factor(commuting.area)3  2.099770   0.080904  25.954  < 2e-16 ***
as.factor(commuting.area)4  3.676638   0.109183  33.674  < 2e-16 ***
as.factor(gender)2       -1.189631   0.042108 -28.252  < 2e-16 ***
age                       0.033923   0.001732  19.582  < 2e-16 ***
edu.level                 0.595739   0.039588  15.049  < 2e-16 ***
as.factor(job.type)2     -0.314276   0.057382  -5.477 4.33e-08 ***
as.factor(job.type)3     -0.525176   0.063059  -8.328  < 2e-16 ***
as.factor(job.type)4     -0.335699   0.339241  -0.990   0.3224
hhm.no                   -0.031265   0.019799  -1.579   0.1143
as.factor(h.type)2        0.189232   0.041063   4.608 4.06e-06 ***
as.factor(h.type)3        0.071971   0.060004   1.199   0.2304
as.factor(h.type)4        0.015089   0.062302   0.242   0.8086
as.factor(h.type)5        0.185892   0.251991   0.738   0.4607
hh.income                 0.057037   0.005536  10.302  < 2e-16 ***
as.factor(area.living)2   0.085575   0.069334   1.234   0.2171
as.factor(area.living)3   0.137566   0.078902   1.744   0.0812 .
as.factor(area.living)4  -0.031845   0.071054  -0.448   0.6540
as.factor(area.living)5   0.158526   0.073422   2.159   0.0308 *
feel.walk.reside          0.030721   0.022593   1.360   0.1739
feel.bus                  0.188393   0.020565   9.161  < 2e-16 ***
feel.subway               0.004670   0.020882   0.224   0.8230
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 24387  on 20884  degrees of freedom
Residual deviance: 20097  on 20861  degrees of freedom
AIC: 20145
```

이항 로지스틱 모델

- 실습 2: 다양한 특성들에 의하여 직장인들이 통근수단 중 승용차를 선택할 확률 결정




```
m.2modes ~ # 통근수단 종속변수
commuting.time + # 통근시간(분)
as.factor(commuting.area) + # 통근지역(1.'현재 살고 있는 동 내', 2. '현재 살고 있는 구
as.factor(gender) + # 성별(1=남성, 2=여성)
age + # 만 나이
edu.level + # 교육수준( 1 '중학교이하', 2 '고졸이하', 3 '대졸이하', 4 '대학원이상')
as.factor(job.type) + # 직업유형(1'관리전문직', 2 '화이트칼라', 3 '블루칼라', 4 '기타)
hhm.no + # 가구원 수
as.factor(h.type) +  # 주택유형(1.단독, 2.아파트, 3.다세대, 4.연립/빌라, 5.기타)
hh.income + # 가구총소득
as.factor(area.living) + # 서울거주지역(1.도심, 2.동북, 3.서북, 4.서남, 5.동남)
feel.walk.reside  +  # 거주지 걷기와 달리기 적합도
feel.bus + # 버스이용환경 만족도
feel.subway, # 지하철 이용환경 만족도
```

```
> round(exp(m.mlogit$coef), 3) # odds ratio
                (Intercept)             commuting.time as.factor(commuting.area)2 as.factor(commuting.area)3 as.factor(commuting.area)4
                      0.002                      0.986                      5.339                      8.164                     39.513
      as.factor(gender)2                        age                  edu.level      as.factor(job.type)2      as.factor(job.type)3
                      0.304                      1.035                      1.814                      0.730                      0.591
    as.factor(job.type)4                        hhm.no       as.factor(h.type)2       as.factor(h.type)3       as.factor(h.type)4
                      0.715                      0.969                      1.208                      1.075                      1.015
      as.factor(h.type)5                   hh.income    as.factor(area.living)2    as.factor(area.living)3    as.factor(area.living)4
                      1.204                      1.059                      1.089                      1.147                      0.969
  as.factor(area.living)5             feel.walk.reside                  feel.bus                feel.subway
                      1.172                      1.031                      1.207                      1.005
```

이항 로지스틱 모델

- 실습 2: 다양한 특성들에 의하여 직장인들이 통근수단 중 승용차를 선택할 확률 결정

```
> library(car) # 패키지 다중공선성(Multicollinearity) 진단: vif
> ?vif
> round(vif(m.mlogit), 3) # 분산팽창계수(vif)
                        GVIF Df GVIF^(1/(2*Df))
commuting.time         1.788  1           1.337
as.factor(commuting.area) 1.840  3           1.107
as.factor(gender)      1.071  1           1.035
age                    1.508  1           1.228
edu.level              1.545  1           1.243
as.factor(job.type)    1.224  3           1.034
hhm.no                 1.297  1           1.139
as.factor(h.type)      1.064  4           1.008
hh.income              1.369  1           1.170
as.factor(area.living) 1.093  4           1.011
feel.walk.reside       1.038  1           1.019
feel.bus               1.554  1           1.247
feel.subway            1.542  1           1.242
```

GVIF = generalized vifs are invariant with respect to the coding of the terms in the model

및 확인

예측과 교차타당성

```
> anova(m.mlogit, test="Chisq") # 귀무모형과의 모형 비교
Analysis of Deviance Table

Model: binomial, link: logit

Response: com.2modes

Terms added sequentially (first to last)


                        Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
NULL                                    20884      24387
commuting.time           1   294.22     20883      24092 < 2.2e-16 ***
as.factor(commuting.area) 3  1734.63     20880      22358 < 2.2e-16 ***
as.factor(gender)        1  1203.37     20879      21154 < 2.2e-16 ***
age                      1   145.95     20878      21009 < 2.2e-16 ***
edu.level                1   470.74     20877      20538 < 2.2e-16 ***
as.factor(job.type)      3    95.11     20874      20443 < 2.2e-16 ***
hhm.no                   1    15.32     20873      20427 9.088e-05 ***
as.factor(h.type)        4    36.95     20869      20390 1.845e-07 ***
hh.income                1   133.33     20868      20257 < 2.2e-16 ***
as.factor(area.living)   4    17.06     20864      20240 0.0018822 **
feel.walk.reside         1    10.87     20863      20229 0.0009772 ***
feel.bus                 1   132.40     20862      20097 < 2.2e-16 ***
feel.subway              1     0.05     20861      20097 0.8230188
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

이전 통근시간 모형 결과와 비교

```
> round(pR2(m.slogit), 4) # 모형 통계량 산출
       llh     llhNull          G2    McFadden        r2ML        r2CU
-12046.2207 -12193.3284   294.2155      0.0121      0.0140      0.0203
```

```
> library(pscl)
> round(pR2(m.mlogit), 4) # 모형 통계량
       llh     llhNull          G2    McFadden        r2ML        r2CU
-10048.3309 -12193.3284  4289.9950      0.1759      0.1857      0.2695
```

이항 로지스틱 모델

- 실습 2: 다양한 특성들에 의하여 직장인들이 통근수단 중 승용차를 선택할 확률 결정

데이터 불러오기

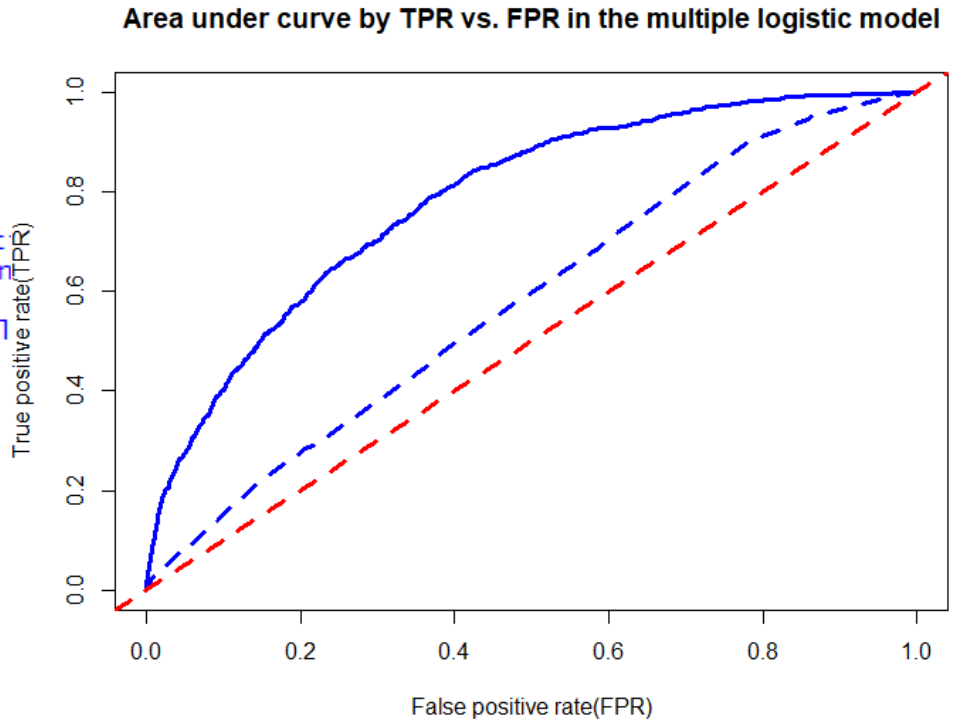
데이터 탐색하기

훈련·시험 데이터분리

모델 구축 및 확인

예측과 교차타당성

```
> ### 4. 예측과 교차검증
> library(ROCR)
> p <- predict(m.mlogit, # 훈련모델
+              newdata=test,# 시험 데이터셋 사용
+              type="response")
> pr <- prediction(p, # 예측값
+                  test$com.2modes) # 실측값
> ?performance() # Function to create performance objects
> prf2 <- performance(pr, # 성능평가할 객체
+                     measure = "tpr", # 평가지표 = True posit
+                     x.measure = "fpr") # A second performan
> plot(prf2,# TPR과 FPR로 그래프 작성
+      main = "Area under curve by TPR vs. FPR in the multipl
+      xlab = "False positive rate(FPR)",
+      ylab = "True positive rate(TPR)",
+      lwd=3, # 선의 두께 = 3
+      col="blue") # 색 = 파란색
> plot(prf, # prf 객체 활용 그래프: 이전 이항로지스틱 모델 결과
+      add=TRUE, # 앞의 그래프에 추가
+      col="blue", lwd=3, lty=2) # 선유형 =2
> abline(a=0, b=1, # a=절편, b=기울기
+        col="red", lwd=3, lty=2)
> auc <- performance(pr, measure = "auc") # auc::: excellent
> auc <- auc@y.values[[1]]
> auc
[1] 0.7850789
```



분류와 예측

이항 로지스틱 모델

- 실습3: 통근수단 중 승용차를 선택할 확률과 이에 영향을 주는 설명변수들은 선형의 관계일까?

```
> library(gam)
> gam.logit <-gam(com.2modes ~
+                   s(commuting.time, df=6) + # 통근시간 knots=6
+                   s(commuting.area) + # 통근지역과 주거지 일치
+                   as.factor(gender) + # 성별(여성=2)
+                   s(age, df=6) + # 만나이
+                   s(hhm.no) + # 가구원수
+                   s(hh.income, df=6), # 가구총소득
+                 family = binomial,
+                  data = train)
```

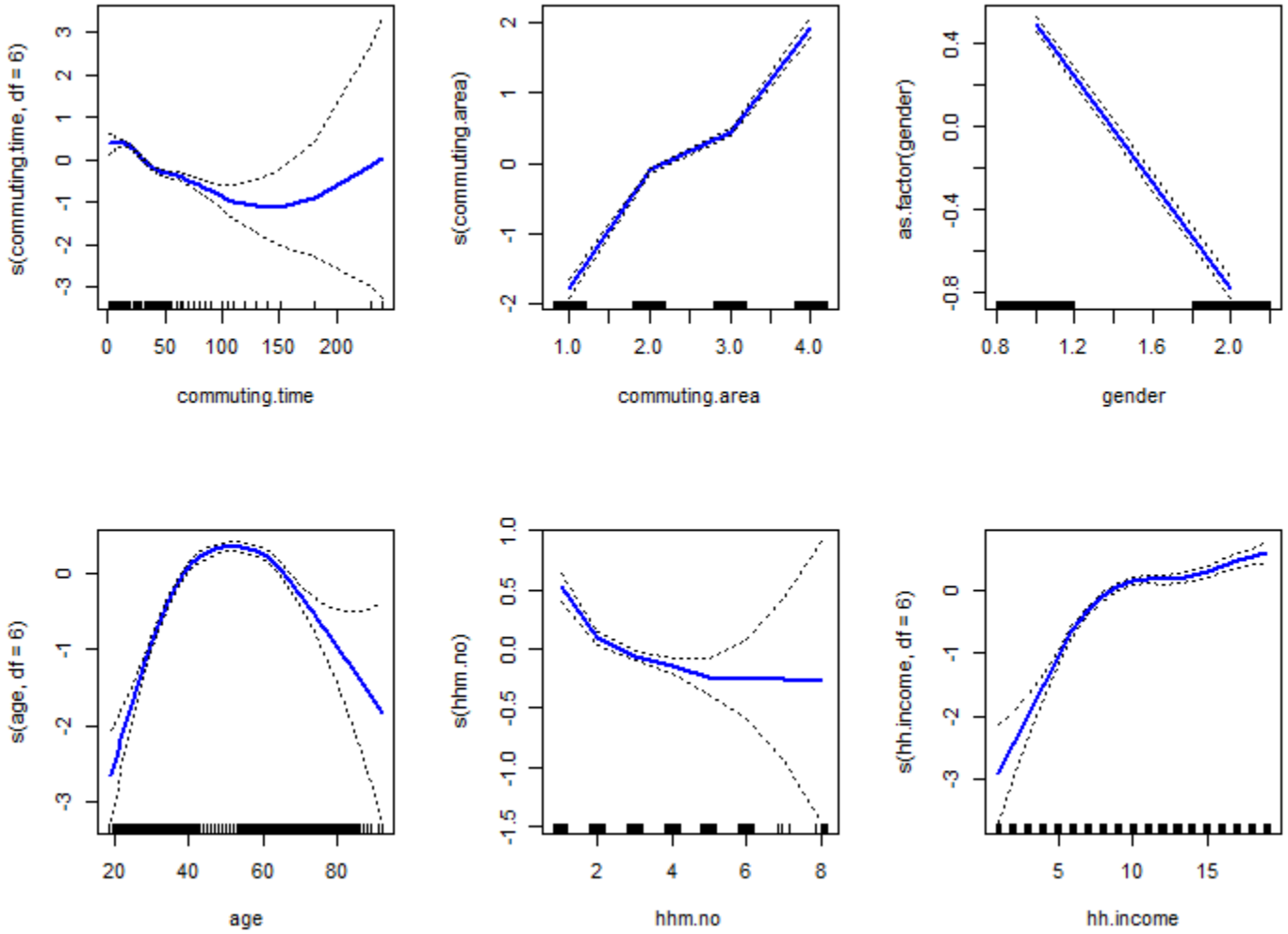
```
> summary(gam.logit) # 결과 요약

Call: gam(formula = com.2modes ~ s(commuting.time, df = 6) + s(commu
    as.factor(gender) + s(age, df = 6) + s(hhm.no) + s(hh.income,
    df = 6), family = binomial, data = train)
Deviance Residuals:
    Min      1Q  Median      3Q     Max
-2.0394 -0.7669 -0.4538  0.7820  3.2979

(Dispersion Parameter for binomial family taken to be 1)

    Null Deviance: 24386.66 on 20884 degrees of freedom
Residual Deviance: 19960.04 on 20858 degrees of freedom
AIC: 20014.04

Number of Local Scoring Iterations: 7

Anova for Parametric Effects
                           Df  Sum Sq Mean Sq  F value Pr(>F)
s(commuting.time, df = 6)   1    99.1   99.11  96.7695 <2e-16 ***
s(commuting.area)           1   946.1  946.13 923.7728 <2e-16 ***
as.factor(gender)           1   921.4  921.38 899.6126 <2e-16 ***
s(age, df = 6)              1   156.9  156.95 153.2396 <2e-16 ***
s(hhm.no)                   1     1.1    1.06   1.0353 0.3089
s(hh.income, df = 6)        1   270.9  270.93 264.5331 <2e-16 ***
Residuals               20858 21362.7    1.02
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Anova for Nonparametric Effects
                         Npar Df Npar Chisq    P(Chi)
(Intercept)
s(commuting.time, df = 6)       5      35.15 1.407e-06 ***
s(commuting.area)               2     201.68 < 2.2e-16 ***
as.factor(gender)
s(age, df = 6)                  5     404.33 < 2.2e-16 ***
s(hhm.no)                       3      22.55 5.020e-05 ***
s(hh.income, df = 6)            5     137.03 < 2.2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

분류와 예측

이항 로지스틱 모델

- 실습3: 통근수단 중 승용차를 선택할 확률과 이에 영향을 주는 설명변수들은 선형의 관계일까?

```
> par(mfrow=c(2,3)) # 한 화면에 2*3 그래픽 창 환경 설정
> plot(gam.logit, # gam 결과 그래프
+      se=T, # 표준오차 그래프
+      col="blue", # 색은 블루
+      lwd=2) # 선두께는 2
> par(mfrow=c(1,1)) # 복귀
```



연습문제 03

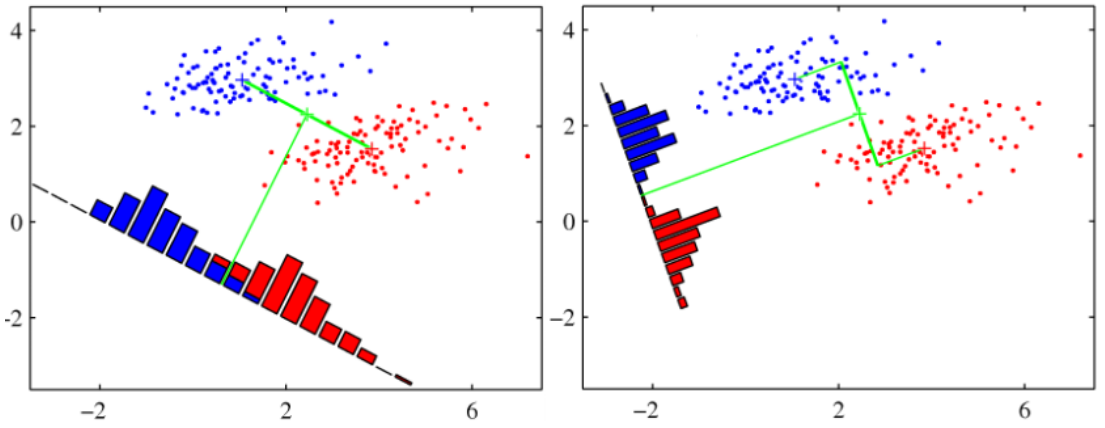
- "df.seoul.worker.csv"를 불러들여, df 객체에 할당한 다음, 종속변수로 com.2modes, 독립변수로 as.factor(commuting.area)와 hh.income으로 이항 로지스틱 모델링을 수행하였다. 실행 결과 분석을 통하여 다음의 질문에 답하시오.
  - hh.income의 회귀계수 추정값은 얼마인가요?
  - commmuting.area 변수의 4수준 요인에 대한 오즈비(odds ratio)는 얼마인가요?
  - hh.income의 일반화 분산팽창계수(generalized VIF)는 얼마인가요?

판별분석

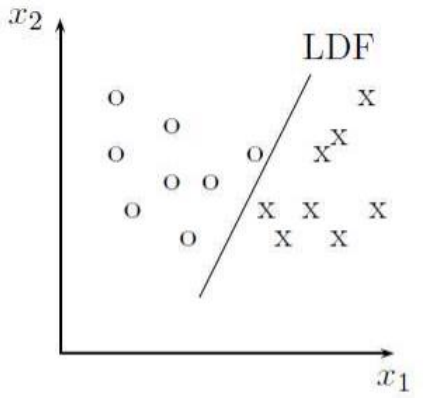
- 판별분석(discriminant analysis)
  - 주어진 자료를 분류하고, 예측하는 것
    - 주어진 각 개체를 미리 정하여진 집단으로 분류하기 위하여 훈련자료를 통하여 판별함수를 구하고, 새로운 개체를 판별함수를 통하여 소속집단을 예측하는 분류과정으로 구성
  - 반응(종속)변수가 범주형 변수
    - 판별분석
    - 종속변수가 없는 경우는 군집분석을 실행
- 판별분석의 목적
  - 집단분류에서 의미 있는 독립변수들을 알 수 있음
  - 집단들간의 유의미한 통계적 차이가 있을 알려줌
  - 판별함수로 새로운 대상을 어느 집단으로 분류할 것인가를 예측
- 판별분석과 로지스틱 모두 선형의 관계로 가정하는 유사한 특징
  - 왜, 판별분석을 사용할까?
    - 2 개 이상의 범주가 있을 시 성능이 더욱 뛰어남
    - 명확하게 구분되지 않은 반응변수일 경우 로지스틱 모형은 계수 등 변동으로 불안정함
    - 자료의 개수가 적을 때에도 로지스틱 모델보다 안정적인 성능 유지
- 판별분석의 가정
  - 변수들은 다변량 정규분포(multivariate normal distribution)를 이루고 있다
    - 다수의 독립변수들 값의 조합이 정규분포를 이룸
  - 각 집단들간의 변수간 분산-공분산 행렬은 동일하다

분류와 예측

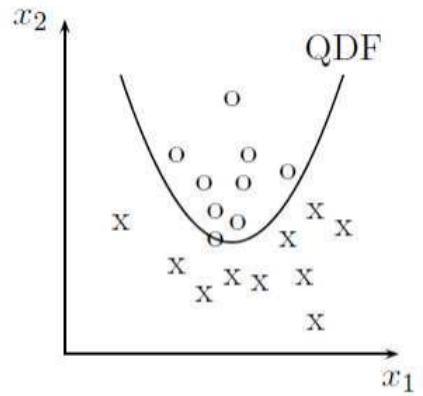
판별분석



- 판별분석의 종류
  - 다변량 정규분포인 경우,
    - 선형판별분석(linear discriminant analysis)
      - 판별함수가 선형함수, 등분산 가정(공분산행렬의 동질성 검정 필요)
    - 이차판별분석(quadratic discriminant analysis)
      - 공분산 행렬이 동질하지 않은 경우
  - 다변량 정규분포가 아닌 경우,
    - Fisher의 판별분석
- 판별분석(discriminant analysis) 모델 비교
  - 공분산 행렬의 동질성 여부가 LDA와 QDA에 영향을 미칠까?
  - 일반적으로 QDA 보다 LDA가 성능이 우수하여 선호
  - 비교
    - $\Sigma_1 \approx \Sigma_2$ 인 경우 LDA가 만족스러운 결과
    - Σ_1, Σ_2 의 차이가 작고, p가 크며 (p>6), 표본의 수가 작으면 ($n_1, n_2 < 25$) LDA가 QDA보다 효율적이다.
    - Σ_1, Σ_2 의 차이가 크면 QDA가 효율적



(a) 선형판별분석

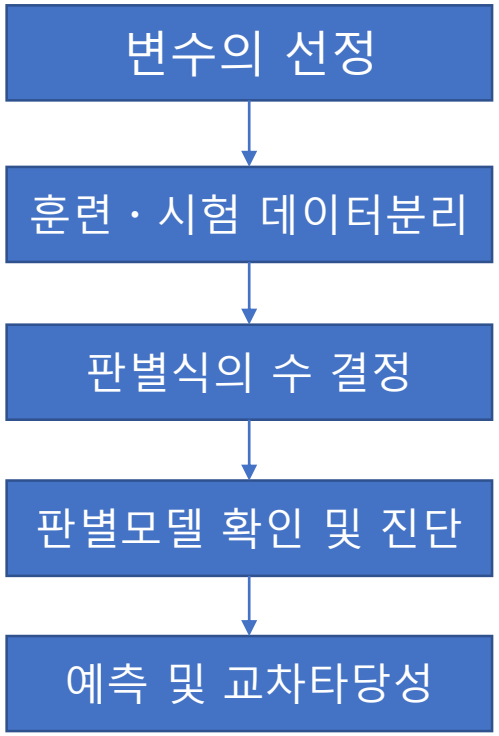


(b) 이차판별분석

출처: http://contents2.kocw.or.kr/KOCW/document/2016/kunsan/jungkangmo/10.pdf

판별분석

- 판별분석의 절차



- 종속변수의 집단 분류
- 기준: 상호배타적(mutually exclusive)이면서 어느 한 집단에 소속(collectively exhaustive)하여야 함

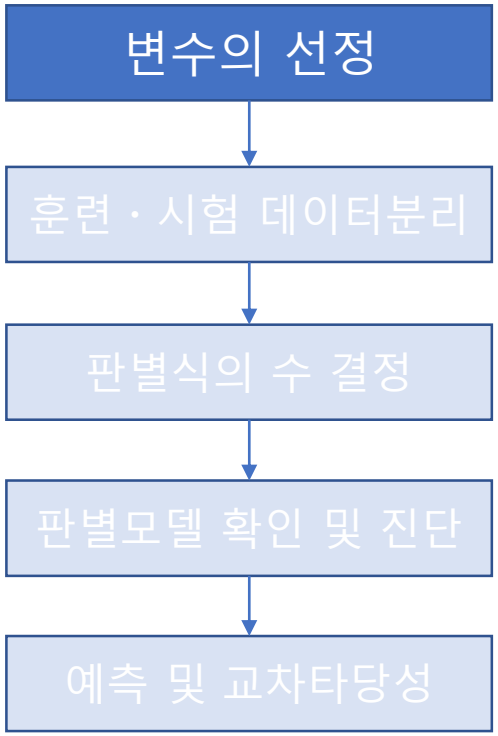
- 표본의 선정

판별식의 수 = min{g-1, p}
여기에서 g: 집단의 수
p: 독립변수의 수

분류와 예측

판별분석

- 선형 판별분석



- 결혼상태(기혼, 비혼, 사별/별거)의 선형판별
  - gender, # 성별
  - age, # 만나이
  - edu.level, # 교육수준
  - hh.income, # 가구 총소득
  - feel.stress, # 스트레스 정도
  - s.reside.year, # 서울 거주년수
  - h.reside.year, # 현 주택 거주년수
  - commuting.area # 통근 목적지 분류

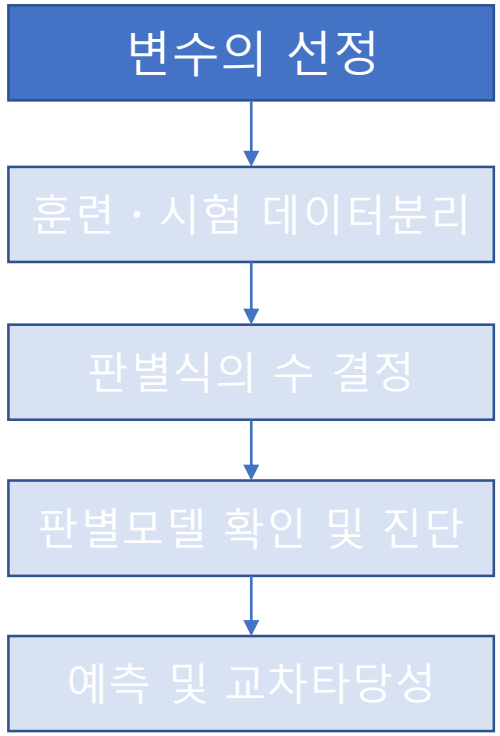
"dplyr" 패키지의 select() 오류 수정
- Error in select() : unused arguments()
- 에러가 발생하는 이유는 ' MASS ' 패키지의 select()함수와 ' dplyr ' 패키지의 select() 함수가 충돌하기 때문
- (방법 1) dplyr::select() <= select는 dplyr 패키지의 select 임을 명시적으로 지정
- (방법 2) select <- dplyr::select <= select는 dplyr 패키지의 select 임을 명시적으로 지정
- (방법 3) MASS 패키지를 먼저 로딩하고, dplyr 패키지를 나중에 로딩하기

```
> ## 데이터 불러오기 및 확인
> df <- read.csv("df.seoul.worker.csv")
> table(df$marriage.status) # 결혼상태 일원빈도분포표

    1     2     3     5     6
18550  5238  1357   952    10
> df$marriage[df$marriage.status == 1 ] <- "기혼" # 기혼
> df$marriage[df$marriage.status == 2 | df$marriage.status == 6] <- "비혼" # 비혼(미혼+등거)
> df$marriage[df$marriage.status == 3 | df$marriage.status == 5] <- "사별/별거" # 사별/별거
> table(df$marriage.status, df$marriage) # 이원빈도분포표

     기혼  비혼 사별/별거
  1 18550     0         0
  2     0  5238         0
  3     0     0      1357
  5     0     0       952
  6     0    10         0
> str(df$marriage) # 데이터 구조: 문자
 chr [1:26107] "기혼" "기혼" "기혼" "기혼" "사별/별거" "기혼" "기혼" "사별/별거" "기혼" "기혼" "사별
> df$marriage <- as.factor(df$marriage) # 요인으로 전환
> attach(df) # 데이터 객체 바로 접근하기
```

분류와 예측

판별분석

- 선형 판별분석



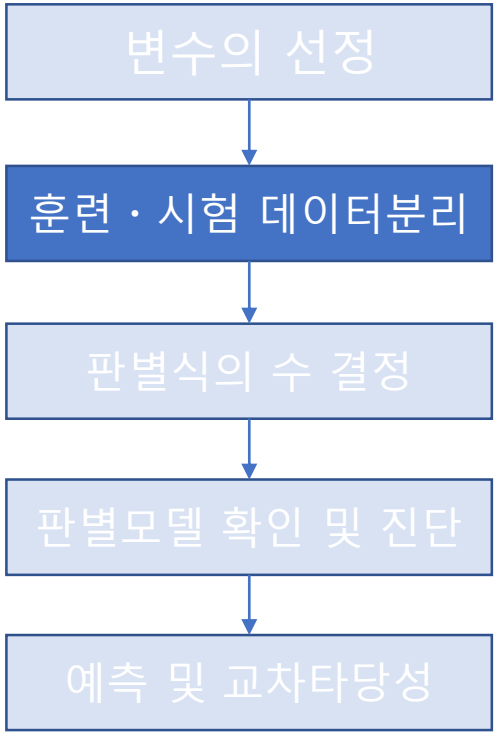
- 결혼상태(기혼, 비혼, 사별/별거)의 선형판별
  - gender, # 성별, age, # 만나이, edu.level, # 교육수준, hh.income, # 가구 총소득
  - feel.stress, # 스트레스 정도, s.reside.year, # 서울 거주년수,
  - h.reside.year, # 현 주택 거주년수, commuting.area # 통근 목적지 분류

```
> str(lda.df) # 데이터 구조 확인하기
'data.frame':   26107 obs. of  9 variables:
 $ marriage      : Factor w/ 3 levels "기혼","비혼",..: 1 1 1 1 3 1 1 3 1 1 ...
 $ gender        : int  1 1 1 2 2 2 1 1 1 2 ...
 $ age           : int  71 69 39 35 67 60 65 56 50 54 ...
 $ edu.level     : int  2 2 3 3 2 1 2 2 2 2 ...
 $ hh.income     : int  11 9 12 12 12 15 15 11 11 11 ...
 $ feel.stress   : int  2 4 2 4 4 2 4 2 4 2 ...
 $ s.reside.year : int  50 50 25 34 25 38 45 20 10 10 ...
 $ h.reside.year : int  5 3 3 3 3 3 3 2 3 3 ...
 $ commuting.area: int  3 2 2 2 2 3 3 2 1 3 ...
> summary(lda.df) # 요약 통계량
      marriage          gender           age          edu.level       hh.income      feel.stress     s.reside.
 기혼      :18550   Min.   :1.000   Min.   :19.00   Min.   :1.000   Min.   : 1.00   Min.   :1.000   Min.   : 1
 비혼      : 5248   1st Qu.:1.000   1st Qu.:36.00   1st Qu.:2.000   1st Qu.: 8.00   1st Qu.:3.000   1st Qu.:24
 사별/별거: 2309   Median :1.000   Median :47.00   Median :3.000   Median :10.00   Median :4.000   Median :32.
                   Mean   :1.389   Mean   :47.14   Mean   :2.612   Mean   :10.48   Mean   :3.434   Mean   :3
                   3rd Qu.:2.000   3rd Qu.:57.00   3rd Qu.:3.000   3rd Qu.:13.00   3rd Qu.:4.000   3rd Qu.:4
                   Max.   :2.000   Max.   :94.00   Max.   :4.000   Max.   :19.00   Max.   :5.000   Max.   :8
 h.reside.year    commuting.area
 Min.   : 0.000   Min.   :1.000
 1st Qu.: 3.000   1st Qu.:2.000
 Median : 6.000   Median :2.000
 Mean   : 7.998   Mean   :2.369
 3rd Qu.:10.000   3rd Qu.:3.000
 Max.   :78.000   Max.   :4.000
```

분류와 예측

판별분석

- 선형 판별분석



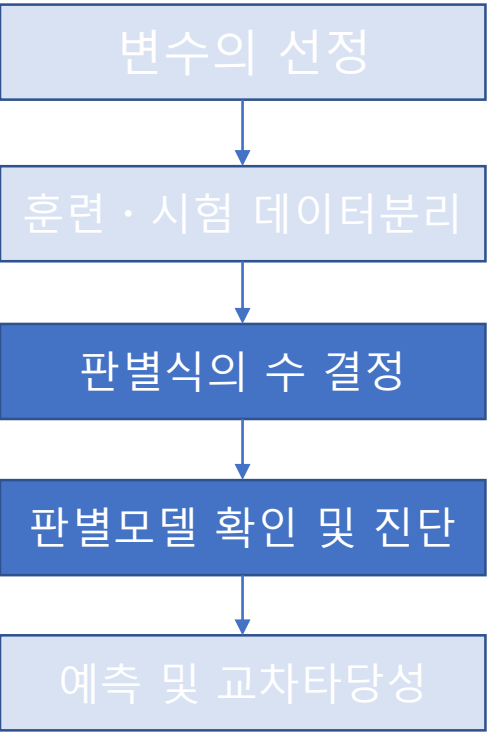
- 결혼상태(기혼, 비혼, 사별/별거)의 선형판별
  - gender, # 성별, age, # 만나이, edu.level, # 교육수준, hh.income, # 가구 총소득
  - feel.stress, # 스트레스 정도, s.reside.year, # 서울 거주년수,
  - h.reside.year, # 현 주택 거주년수, commuting.area # 통근 목적지 분류

```
> ## 데이터 분리하기: train-test split
> library(caTools)
> set.seed(1234) # 난수 생성
> ?sample.split() # Split Data into Test and Train Set
> split <- sample.split(marriage, # 결혼요인 기준
+                       SplitRatio = 0.7) # 비율 0.7:03
> split %>% head(15) # 15줄 내용확인
 [1]  TRUE  TRUE  TRUE  TRUE FALSE FALSE  TRUE FALSE  TRUE  TRUE  TRUE FALSE  TRUE FALSE  TRUE
> train <- subset(df, # 하위데이터셋 추출
+                 split == TRUE) # TRUE이면 추출
> test <- subset(df, split == FALSE) # FALSE이면 추출
> test.y <- test[ ,"marriage"] # test 데이터 중 marriage 변수만 추출
> nrow(train) ; nrow(test)
[1] 18275
[1] 7832
```

분류와 예측

판별분석

- 선형 판별분석



- 결혼상태(기혼, 비혼, 사별/별거)의 선형판별
  - gender, # 성별, age, # 만나이, edu.level, # 교육수준, hh.income, # 가구 총소득
  - feel.stress, # 스트레스 정도, s.reside.year, # 서울 거주년수,
  - h.reside.year, # 현 주택 거주년수, commuting.area # 통근 목적지 분류

판별식의 수 = min{g-1, p}
여기에서 g: 집단의 수
p: 독립변수의 수

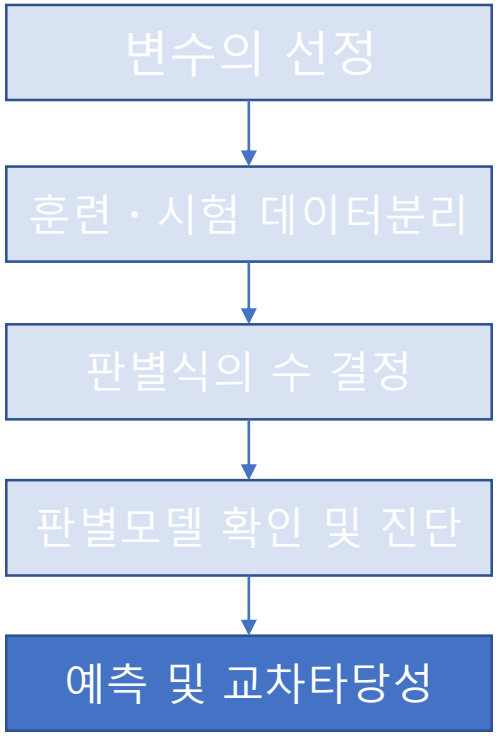
```
> ## 3. 선형판별분석: lda modeling
> library(MASS)
> df.lda <- lda(marriage ~ # 결혼상태
+                 as.factor(gender) + # 성별
+                 age + # 만나이
+                 edu.level + # 교육수준
+                 hh.income + # 가구 총소득
+                 feel.stress + # 스트레스 정도
+                 s.reside.year + # 서울 거주년수
+                 h.reside.year + # 현 주택 거주년수
+                 as.factor(commuting.area), # 통근지 유형
+               data = train)
> df.lda # 결과 확인
```

```
> df.lda # 결과 확인
Call:
lda(marriage ~ as.factor(gender) + age + edu.level + hh.income +
    feel.stress + s.reside.year + h.reside.year + as.factor(commuting.area),
    data = train)

Prior probabilities of groups:
      기혼       비혼  사별/별거
0.71053352 0.20103967 0.08842681

Group means:
         as.factor(gender)2      age edu.level hh.income feel.stress s.reside.year h.reside.year as.factor(commuting.area)2
기혼              0.3276088 49.62495  2.592222 10.845129     3.425645      33.53955      7.820331                  0.3376973
비혼              0.4855743 33.67202  2.863364 10.477681     3.439848      25.78443      8.262112                  0.2887861
사별/별거         0.6707921 57.34282  2.199876  7.630569     3.461015      35.89604      8.719678                  0.3644802
         as.factor(commuting.area)3 as.factor(commuting.area)4
기혼                      0.4094725                  0.06669234
비혼                      0.5538922                  0.05498095
사별/별거                 0.2704208                  0.03898515
```

분류와 예측

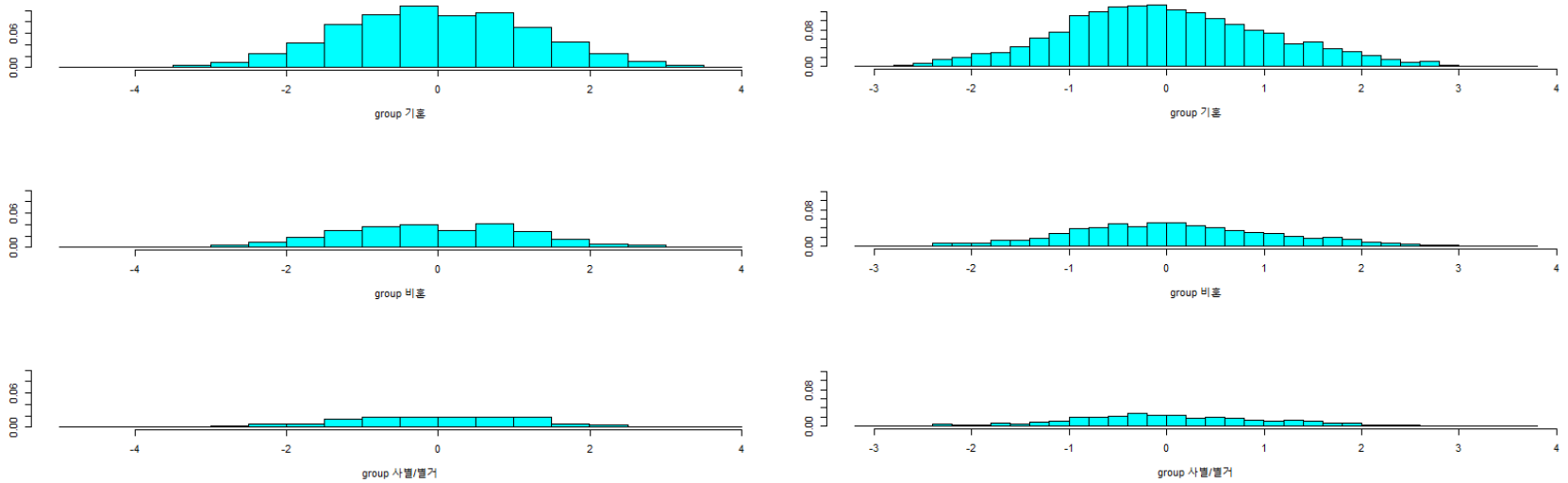
판별분석

- 선형 판별분석



- 결혼상태(기혼, 비혼, 사별/별거)의 선형판별
  - gender, # 성별, age, # 만나이, edu.level, # 교육수준, hh.income, # 가구 총소득
  - feel.stress, # 스트레스 정도, s.reside.year, # 서울 거주년수,
  - h.reside.year, # 현 주택 거주년수, commuting.area # 통근 목적지 분류

```
> ### 4. Prediction Accuracy
> lda.pred <- predict(df.lda, data = test) # 모델 예측
> class(lda.pred)
[1] "list"
> #*** Stacked Histogram of the LDA Values
> ldahist(lda.pred$x[,1], g = test$marriage) # first discriminant function
> ldahist(lda.pred$x[,2], g = test$marriage)
```



분류와 예측

판별분석

- 선형 판별분석

변수의 선정

↓

훈련 · 시험 데이터분리

↓

판별식의 수 결정

↓

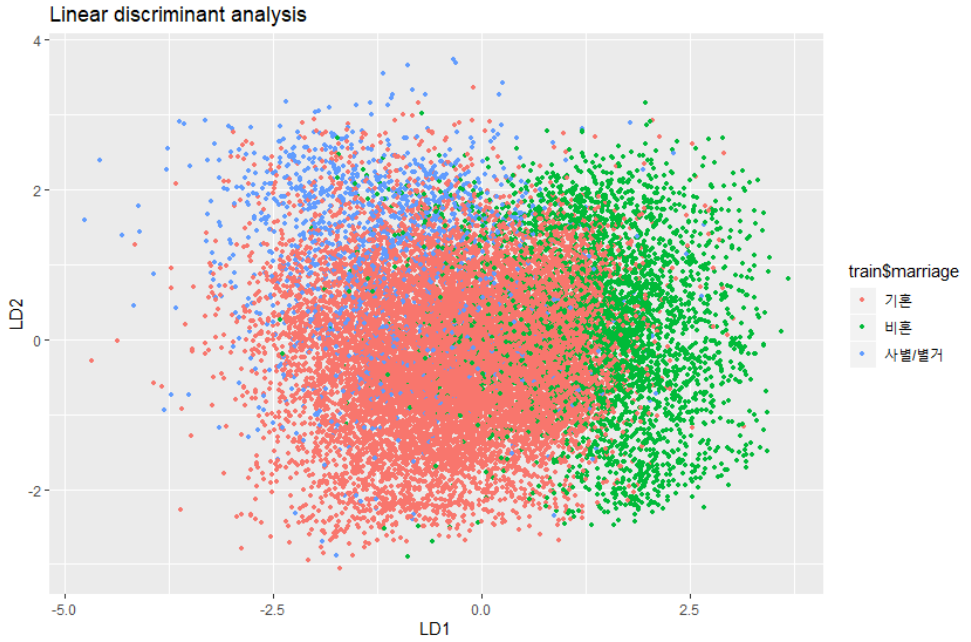
판별모델 확인 및 진단

↓

예측 및 교차타당성

- 결혼상태(기혼, 비혼, 사별/별거)의 선형판별
  - gender, # 성별, age, # 만나이, edu.level, # 교육수준, hh.income, # 가구 총소득
  - feel.stress, # 스트레스 정도, s.reside.year, # 서울 거주년수,
  - h.reside.year, # 현 주택 거주년수, commuting.area # 통근 목적지 분류

```
> #convert to data frame
> newdata <- data.frame(type = train[,"marriage"], lda = lda.pred$x)
> head(newdata)
  type    lda.LD1    lda.LD2
1 기혼 -2.4209848 -0.8274021
2 기혼 -2.5933126 -0.5276929
3 기혼  0.6111468 -1.1796679
4 기혼  0.6688061  0.3443925
7 기혼 -2.0091991 -1.4958851
9 기혼 -0.4605385 -0.4801534
> library(ggplot2)
> ggplot(newdata) +
+   geom_point(aes(lda.LD1, lda.LD2,
+                  colour = train$marriage),
+              size = 1.2) +
+   labs(title = "Linear discriminant analysis",
+        x = "LD1",
+        y = "LD2")
```

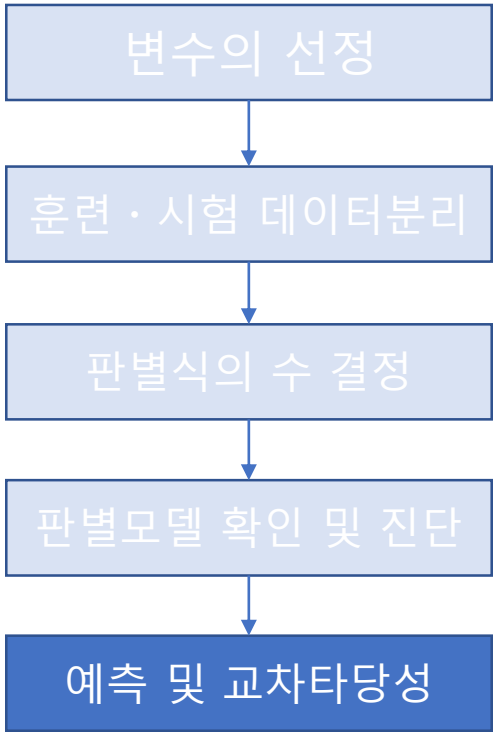



```
> table(train$marriage, lda.pred$class) # 실측치와 예측치 이원빈도분포표

           기혼  비혼 사별/별거
기혼      12062   650      273
비혼       1445  2210       19
사별/별거  1148    39      429
```

분류와 예측

판별분석

- 선형 판별분석



- 결혼상태(기혼, 비혼, 사별/별거)의 선형판별
  - gender, # 성별, age, # 만나이, edu.level, # 교육수준, hh.income, # 가구 총소득
  - feel.stress, # 스트레스 정도, s.reside.year, # 서울 거주년수,

muting.area # 통근 목적지 분류

```
> library(caret)
> confusionMatrix(train$marriage, lda.pred$class)
Confusion Matrix and Statistics

          Reference
Prediction   기혼  비혼 사별/별거
  기혼      12062   650       273
  비혼       1445  2210        19
  사별/별거  1148    39       429

Overall Statistics

               Accuracy : 0.8044
                 95% CI : (0.7986, 0.8102)
    No Information Rate : 0.8019
    P-Value [Acc > NIR] : 0.1993

                  Kappa : 0.5047

 Mcnemar's Test P-Value : <2e-16
Statistics by Class:

                     Class: 기혼 Class: 비혼 Class: 사별/별거
Sensitivity              0.8231      0.7623          0.59501
Specificity              0.7450      0.9048          0.93238
Pos Pred Value           0.9289      0.6015          0.26547
Neg Pred Value           0.5098      0.9528          0.98247
Prevalence               0.8019      0.1586          0.03945
Detection Rate           0.6600      0.1209          0.02347
Detection Prevalence     0.7105      0.2010          0.08843
Balanced Accuracy        0.7840      0.8336          0.76369
```

- No Information Rate: 가장 많은 값이 발견된 분류의 비율
- 맥니마 검정: **짝지은 명목형 데이터**에서 Column과 Row의 **marginal probability**가 같은지를 검정

분류와 예측

판별분석

- 선형 판별분석

- 결혼상태(기혼, 비혼, 사별/별거)의 선형판별
  - gender, # 성별, age, # 만나이, edu.level, # 교육수준, hh.income, # 가구 총소득
  - feel.stress, # 스트레스 정도, s.reside.year, # 서울 거주년수,
  - h.reside.year, # 현 주택 거주년수, commuting.area # 통근 목적지 분류

```
> ## 분산-공분산 행렬이 동일한지 체크
> library(biotools)
> boxM(train[ , c("gender", "age", "edu.level", "hh.income",
+                 "feel.stress", "s.reside.year", "h.reside.year", "commuting.area")], train$marriage)

        Box's M-test for Homogeneity of Covariance Matrices

data:  train[, c("gender", "age", "edu.level", "hh.income", "feel.stress",     "s.reside.year", "h.reside.year", "commuting.are
a")]
Chi-Sq (approx.) = 4429.4, df = 72, p-value < 2.2e-16
```

훈련

판별식의 수 결정

판별모델 확인 및 진단

- **Box's M test** is a multivariate statistical test used to check the equality of multiple variance-covariance matrices.
- The test is commonly used to test the assumption of homogeneity of variances and covariances in MANOVA and linear discriminant analysis.
- It is named after George E. P. Box.

- 가정 : 각 그룹에서 공분산행렬이 동일
- 요인 A의 각 수준의 공분산행렬이 동일해야한다.
- 요인 B의 각 수준의 공분산행렬이 동일해야한다.
- 교호요인 A:B 의 각 수준의 공분산행렬이 동일해야한다.

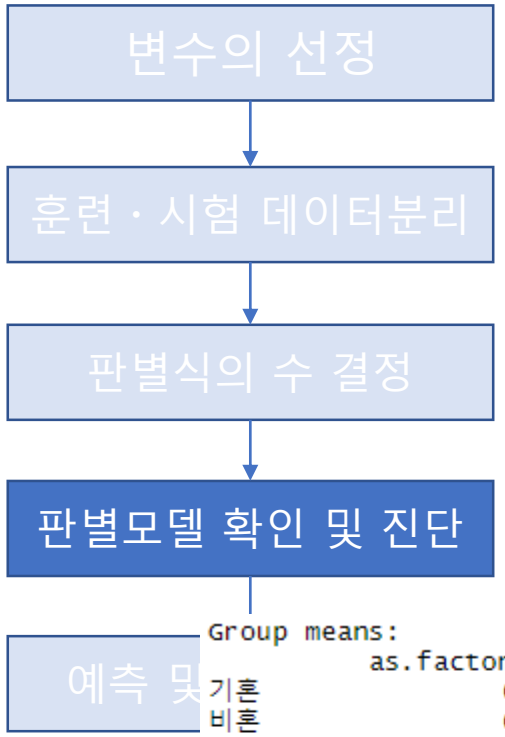
- 공분산행렬의 동질성 검정
- $H_0 : \Sigma_1 = \Sigma_2 = \cdots = \Sigma_k$
- 검정통계량 : Box's M test
- 합동 공분산행렬 S_p , $M = \gamma \sum_{i=1}^{k} (n_i - 1) \log |S_i^{-1} S_p| \sim \chi^2((p+1)(k-1)/2)$
- 행렬식에 로그를 취한 꼴이므로 M이 0에 가까울수록 H_0 채택

출처: http://contents2.kocw.or.kr/KOCW/document/2016/kunsan/jungkangmo/6.pdf

분류와 예측

판별분석

- 다항 판별분석

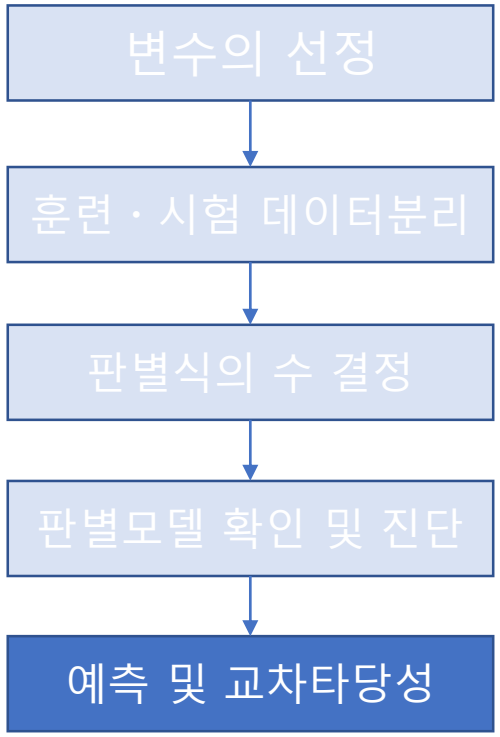


- 결혼상태(기혼, 비혼, 사별/별거)의 **다항판별**
  - gender, # 성별, age, # 만나이, edu.level, # 교육수준, hh.income, # 가구 총소득
  - feel.stress, # 스트레스 정도, s.reside.year, # 서울 거주년수,
  - h.reside.year, # 현 주택 거주년수, commuting.area # 통근 목적지 분류

```
> ## qda modeling
> df.qda <- qda(marriage ~ # 결혼상태
+                 as.factor(gender) + # 성별
+                 age + # 만나이
+                 edu.level + # 교육수준
+                 hh.income + # 가구 총소득
+                 feel.stress + # 스트레스 정도
+                 s.reside.year + # 서울 거주년수
+                 h.reside.year + # 현 주택 거주년수
+                 as.factor(commuting.area), # 통근지 유형,
+                 data = train)
> df.qda
Call:
qda(marriage ~ as.factor(gender) + age + edu.level + hh.income +
    feel.stress + s.reside.year + h.reside.year + as.factor(commuting.area),
    data = train)

Prior probabilities of groups:
      기혼       비혼  사별/별거
0.71053352 0.20103967 0.08842681

Group means:
       as.factor(gender)2      age edu.level hh.income feel.stress s.reside.year h.reside.year as.factor(commuting.area)2
기혼            0.3276088 49.62495  2.592222 10.845129    3.425645      33.53955      7.820331                 0.3376973
비혼            0.4855743 33.67202  2.863364 10.477681    3.439848      25.78443      8.262112                 0.2887861
사별/별거       0.6707921 57.34282  2.199876  7.630569    3.461015      35.89604      8.719678                 0.3644802
       as.factor(commuting.area)3 as.factor(commuting.area)4
기혼                    0.4094725                  0.06669234
비혼                    0.5538922                  0.05498095
사별/별거               0.2704208                  0.03898515
```

판별분석

- 다항 판별분석



- 결혼상태(기혼, 비혼, 사별/별거)의 **다항판별**
  - gender, # 성별, age, # 만나이, edu.level, # 교육수준, hh.income, # 가구 총소득
  - feel.stress, # 스트레스 정도, s.reside.year, # 서울 거주년수,
  - h.reside.year, # 현 주택 거주년수, commuting.area # 통근 목적지 분류

```
> # Prediction Accuracy
> qda.pred <- predict(df.qda, data = train) # 모델 예측
> class(qda.pred)
[1] "list"
> table(train$marriage, qda.pred$class) # 실측치와 예측치 이원빈도분포표

             기혼  비혼 사별/별거
  기혼      11423   983       579
  비혼       1070  2548        56
  사별/별거   878    88       650

> library(caret)
> confusionMatrix(train$marriage, qda.pred$class)
Confusion Matrix and Statistics

          Reference
Prediction   기혼  비혼 사별/별거
  기혼      11423   983       579
  비혼       1070  2548        56
  사별/별거   878    88       650

Overall Statistics

               Accuracy : 0.8001
                 95% CI : (0.7942, 0.8058)
    No Information Rate : 0.7317
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.5394

 Mcnemar's Test P-Value : 1.473e-15
```

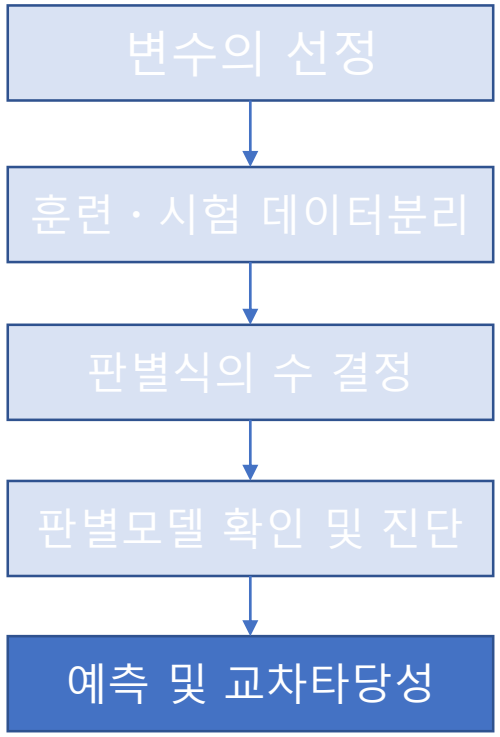
LDA모델 결과와 비교

```
Overall Statistics

               Accuracy : 0.8044
                 95% CI : (0.7986, 0.8102)
    No Information Rate : 0.8019
    P-Value [Acc > NIR] : 0.1993

                  Kappa : 0.5047

 Mcnemar's Test P-Value : <2e-16
```

분류와 예측

판별분석

- 다항 판별분석



- 결혼상태(기혼, 비혼, 사별/별거)의 **다항판별**
  - gender, # 성별, age, # 만나이, edu.level, # 교육수준, hh.income, # 가구 총소득
  - feel.stress, # 스트레스 정도, s.reside.year, # 서울 거주년수,
  - h.reside.year, # 현 주택 거주년수, commuting.area # 통근 목적지 분류

```
> # Prediction Accuracy
> qda.pred <- predict(df.qda, data = train) # 모델 예측
> class(qda.pred)
[1] "list"
> table(train$marriage, qda.pred$class) # 실측치와 예측치 이원빈도분포표

             기혼  비혼 사별/별거
  기혼      11423   983       579
  비혼       1070  2548        56
  사별/별거   878    88       650

> library(caret)
> confusionMatrix(train$marriage, qda.pred$class)
Confusion Matrix and Statistics

          Reference
Prediction   기혼  비혼 사별/별거
  기혼      11423   983       579
  비혼       1070  2548        56
  사별/별거   878    88       650

Overall Statistics

               Accuracy : 0.8001
                 95% CI : (0.7942, 0.8058)
    No Information Rate : 0.7317
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.5394

 Mcnemar's Test P-Value : 1.473e-15
```

LDA모델 결과와 비교

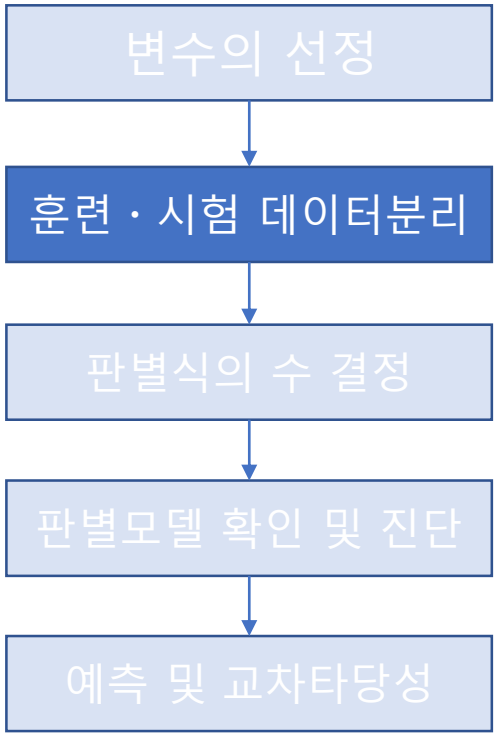
```
Overall Statistics

               Accuracy : 0.8044
                 95% CI : (0.7986, 0.8102)
    No Information Rate : 0.8019
    P-Value [Acc > NIR] : 0.1993

                  Kappa : 0.5047

 Mcnemar's Test P-Value : <2e-16
```

분류와 예측

판별분석

추가 실습

- 선형 판별분석

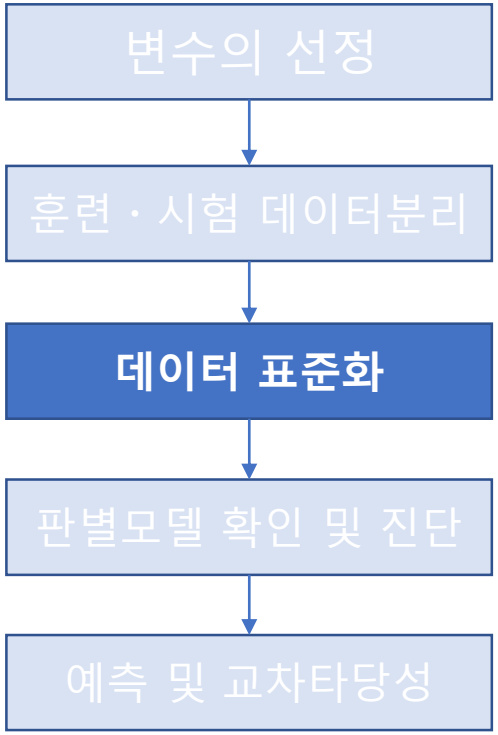


- 결혼상태(기혼, 비혼, 사별/별거)의 선형판별
  - gender, # 성별, age, # 만나이, edu.level, # 교육수준, hh.income, # 가구 총소득
  - feel.stress, # 스트레스 정도, s.reside.year, # 서울 거주년수,
  - h.reside.year, # 현 주택 거주년수, commuting.area # 통근 목적지 분류

```
> # 1. Split the data into training and test set:
> # Load the data
> lda.df <- df %>% dplyr::select(marriage,
+                                gender, # 성별
+                                age, # 만나이
+                                edu.level, # 교육수준
+                                hh.income, # 가구 총소득
+                                feel.stress, # 스트레스 정도
+                                s.reside.year, # 서울 거주년수
+                                h.reside.year, # 현 주택 거주년수
+                                commuting.area) # 통근 목적지 분류
> # Split the data into
> library(caTools)
> set.seed(1234)
> split <- sample.split(lda.df$marriage, SplitRatio = 0.7)
> split %>% head(15)
 [1]  TRUE  TRUE  TRUE  TRUE FALSE FALSE  TRUE FALSE  TRUE  TRUE  TRUE FALSE
> train.data <- subset(lda.df, split == T)
> test.data <- subset(lda.df, split == F)
> test.y <- test.data[,"marriage"]
> str(train.data)
'data.frame':   18275 obs. of  9 variables:
 $ marriage      : Factor w/ 3 levels "기혼","비혼",..: 1 1 1 1 1 1 1 3 2 3 ...
 $ gender        : int  1 1 1 2 1 1 2 1 2 2 ...
 $ age           : int  71 69 39 35 65 50 54 56 44 64 ...
 $ edu.level     : int  2 2 3 3 2 2 2 2 3 1 ...
 $ hh.income     : int  11 9 12 12 15 11 11 9 11 11 ...
 $ feel.stress   : int  2 4 2 4 4 4 2 4 4 4 ...
 $ s.reside.year : int  50 50 25 34 45 10 10 40 43 46 ...
 $ h.reside.year : int  5 3 3 3 3 3 3 1 7 7 ...
 $ commuting.area: int  3 2 2 2 3 1 3 2 2 2 ...
```

판별분석

- 선형 판별분석



- 결혼상태(기혼, 비혼, 사별/별거)의 선형판별
  - gender, # 성별, age, # 만나이, edu.level, # 교육수준, hh.income, # 가구 총소득
  - feel.stress, # 스트레스 정도, s.reside.year, # 서울 거주년수,
  - h.reside.year, # 현 주택 거주년수, commuting.area # 통근 목적지 분류

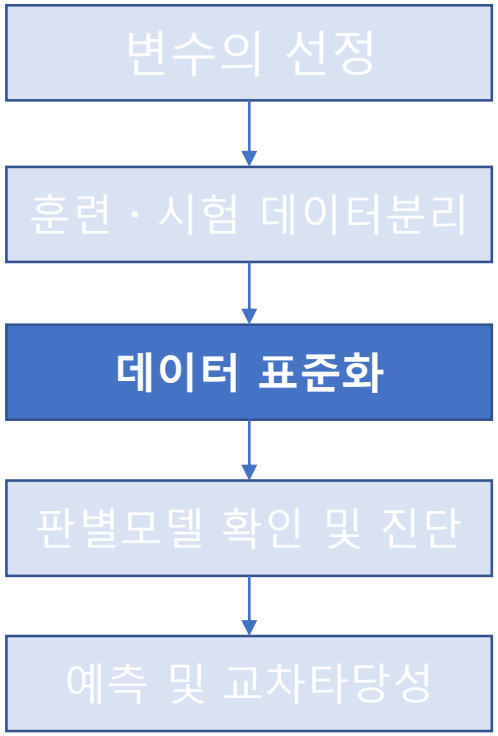
?preprocess()

- Normalize the data.
- Pre-processing transformation (centering, scaling etc.) can be estimated from the training data and applied to any data set with the same variables.

```
preProcess(x, ...)

## Default S3 method:
preProcess(x, method = c("center", "scale"),
  thresh = 0.95, pcaComp = NULL, na.remove = TRUE, k = 5,
  knnSummary = mean, outcome = NULL, fudge = 0.2, numUnique = 3,
  verbose = FALSE, freqCut = 95/5, uniqueCut = 10, cutoff = 0.9,
  rangeBounds = c(0, 1), ...)
```

판별분석

- 선형 판별분석



- 결혼상태(기혼, 비혼, 사별/별거)의 선형판별
  - gender, # 성별, age, # 만나이, edu.level, # 교육수준, hh.income, # 가구 총소득
  - feel.stress, # 스트레스 정도, s.reside.year, # 서울 거주년수,
  - h.reside.year, # 현 주택 거주년수, commuting.area # 통근 목적지 분류

```
> # 2. Normalize the data. Categorical variables are automatically ignored.
> # Estimate preprocessing parameters
> ?preProcess() # Pre-Processing of Predictors
> preproc.param <- train.data %>%
+   preProcess(method = c("center", "scale"))
> # Transform the data using the estimated parameters
> train.transformed <- preproc.param %>% predict(train.data)
> test.transformed <- preproc.param %>% predict(test.data)
> summary(train.transformed); summary(train.data)
```

```
      marriage          gender              age             edu.level         hh.income         feel.stress
 기혼      :12985   Min.   :-0.7991   Min.   :-2.200235   Min.   :-2.7318   Min.   :-2.5390   Min.   :-2.7914
 비혼      : 3674   1st Qu.:-0.7991   1st Qu.:-0.869144   1st Qu.:-1.0372   1st Qu.:-0.6656   1st Qu.:-0.4955
 사별/별거: 1616   Median :-0.7991   Median :-0.007849   Median : 0.6574   Median :-0.1303   Median : 0.6525
                   Mean   : 0.0000   Mean   : 0.000000   Mean   : 0.0000   Mean   : 0.0000   Mean   : 0.0000
                   3rd Qu.: 1.2514   3rd Qu.: 0.775146   3rd Qu.: 0.6574   3rd Qu.: 0.6726   3rd Qu.: 0.6525
                   Max.   : 1.2514   Max.   : 3.672227   Max.   : 2.3521   Max.   : 2.2784   Max.   : 1.8004
```

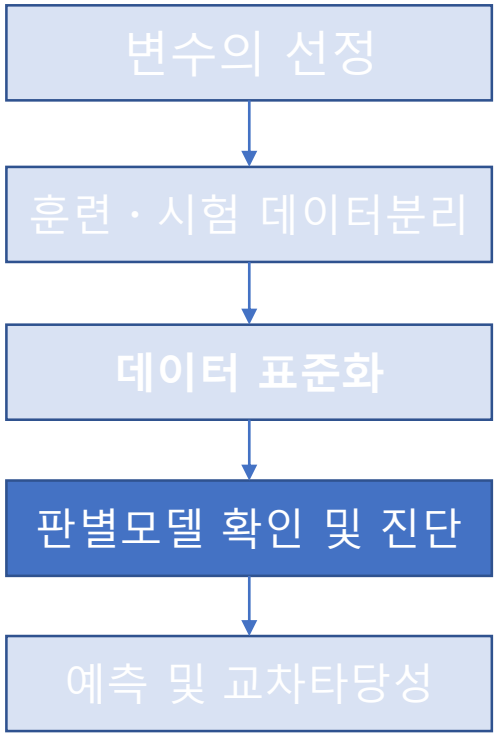
<표준화 이후 데이터>

```
      marriage          gender            age           edu.level       hh.income       feel.stress     s.reside.year
 기혼      :12985   Min.   :1.00   Min.   :19.0   Min.   :1.000   Min.   : 1.00   Min.   :1.000   Min.   : 1.00
 비혼      : 3674   1st Qu.:1.00   1st Qu.:36.0   1st Qu.:2.000   1st Qu.: 8.00   1st Qu.:3.000   1st Qu.:24.00
 사별/별거: 1616   Median :1.00   Median :47.0   Median :3.000   Median :10.00   Median :4.000   Median :32.00
                   Mean   :1.39   Mean   :47.1   Mean   :2.612   Mean   :10.49   Mean   :3.432   Mean   :32.19
                   3rd Qu.:2.00   3rd Qu.:57.0   3rd Qu.:3.000   3rd Qu.:13.00   3rd Qu.:4.000   3rd Qu.:40.00
                   Max.   :2.00   Max.   :94.0   Max.   :4.000   Max.   :19.00   Max.   :5.000   Max.   :82.00
```

<표준화 이전 데이터>

판별분석

- 선형 판별분석



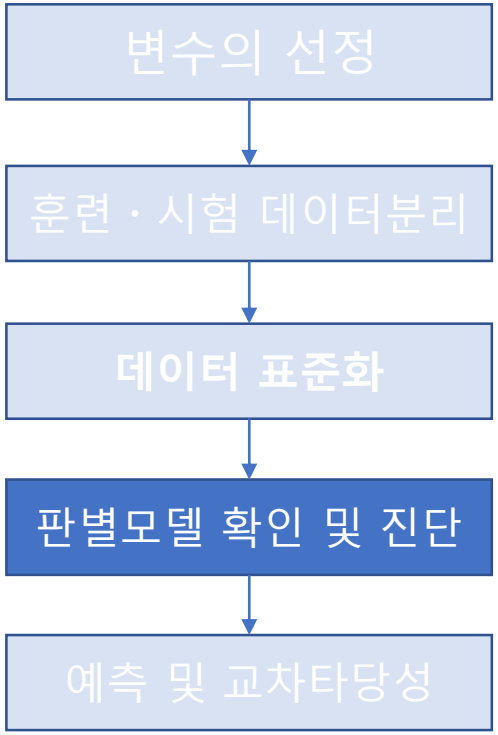
- 결혼상태(기혼, 비혼, 사별/별거)의 선형판별
  - gender, # 성별, age, # 만나이, edu.level, # 교육수준, hh.income, # 가구 총소득
  - feel.stress, # 스트레스 정도, s.reside.year, # 서울 거주년수,
  - h.reside.year, # 현 주택 거주년수, commuting.area # 통근 목적지 분류

```
> # 3. Linear discriminant analysis - LDA
> library(MASS)
> # Fit the model
> model <- lda(marriage ~ # 결혼상태
+                  gender + # 성별
+                  age + # 만나이
+                  edu.level + # 교육수준
+                  hh.income + # 가구 총소득
+                  feel.stress + # 스트레스 정도
+                  s.reside.year + # 서울 거주년수
+                  h.reside.year + # 현 주택 거주년수
+                  commuting.area, # 통근지 유형
+                data = train.transformed)
> model
Call:
lda(marriage ~ gender + age + edu.level + hh.income + feel.stress +
    s.reside.year + h.reside.year + commuting.area, data = train.transformed)

Prior probabilities of groups:
      기혼       비혼  사별/별거
0.71053352 0.20103967 0.08842681
```

판별분석

추가 실습

- 선형 판별분석



- 결혼상태(기혼, 비혼, 사별/별거)의 선형판별
  - gender, # 성별, age, # 만나이, edu.level, # 교육수준, hh.income, # 가구 총소득
  - feel.stress, # 스트레스 정도, s.reside.year, # 서울 거주년수,
  - h.reside.year, # 현 주택 거주년수, commuting.area # 통근 목적지 분류

```
Group means:
              gender        age   edu.level   hh.income  feel.stress s.reside.year h.reside.year commuting.area
기혼       -0.1273409  0.1976832 -0.03358157  0.09584606 -0.006868069     0.1006722   -0.02388987    -0.01367025
비혼        0.1965592 -1.0514234  0.42590335 -0.00249517  0.009435718    -0.4773390    0.03880451     0.22771326
사별/별거  0.5763385  0.8019885 -0.69846053 -0.76447635  0.033734555     0.2763086    0.10373899    -0.40786530

Coefficients of linear discriminants:
                     LD1         LD2
gender         -0.10588539  0.70677767
age            -1.35305894 -0.09091837
edu.level      -0.09820972 -0.18262762
hh.income       0.01255595 -0.73791690
feel.stress    -0.05272273  0.04485399
s.reside.year  -0.04050020 -0.07991995
h.reside.year   0.45757352  0.19371857
commuting.area  0.03855591  0.03963325

Proportion of trace:
   LD1    LD2
0.8155 0.1845
```

분류와 예측

판별분석

추가 실습

- 선형 판별분석

변수의 선정

훈련 · 시험 데이터분리

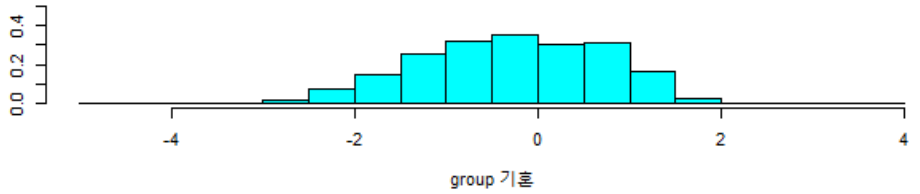
데이터 표준화

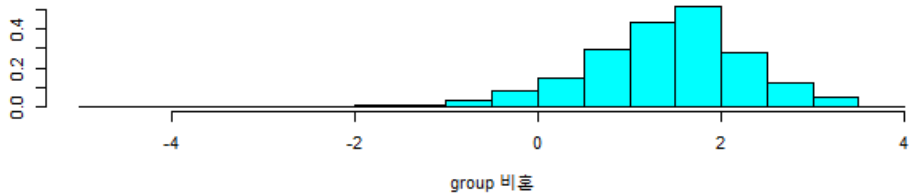
판별모델 확인 및 진단

예측 및 교차타당성

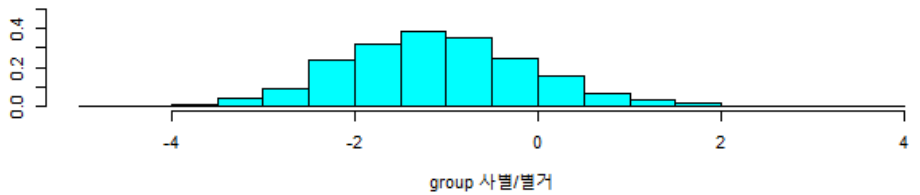
- 결혼상태(기혼, 비혼, 사별/별거)의 선형판별
  - gender, # 성별, age, # 만나이, edu.level, # 교육수준, hh.income, # 가구 총소득
  - feel.stress, # 스트레스 정도, s.reside.year, # 서울 거주년수,
  - h.reside.year, # 현 주택 거주년수, commuting.area # 통근 목적지 분류

```
> ### 4. 예측 및 교차 검증
> # Make predictions
> predictions <- model %>% predict(test.transformed)
> # Model accuracy
> mean(predictions$class==test.transformed$marriage)
[1] 0.8130746
> # Compute LDA:
> plot(model, dimen = 1)
```





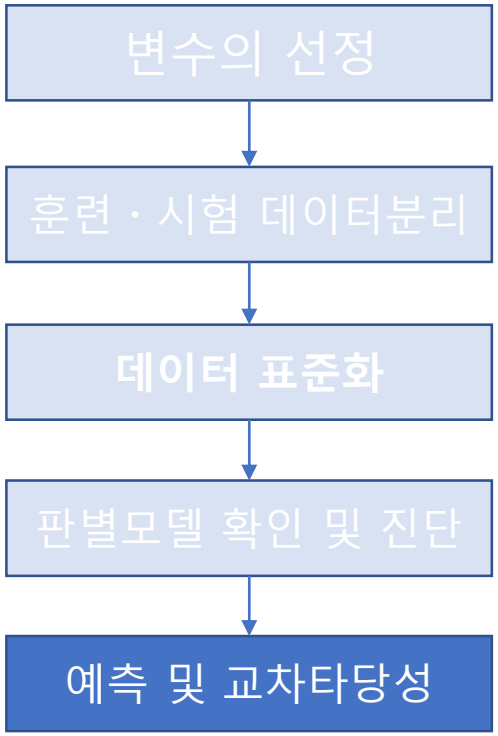

```
> ## 만들어진 lda 함수를 가지고 시험 데이터로 predict
> testpred <- predict(model, test.transformed)
> ## misclass error 확인
> misclass.error <- mean(test.y != testpred$class)
> misclass.error
[1] 0.1869254
```



판별분석

추가 실습

- 선형 판별분석



- 결혼상태(
  - gende
  - feel.str
  - h.resid

```
> library(gmodels)
> CrossTable(x=test.y, y=testpred$class, prop.chisq = TRUE)
```

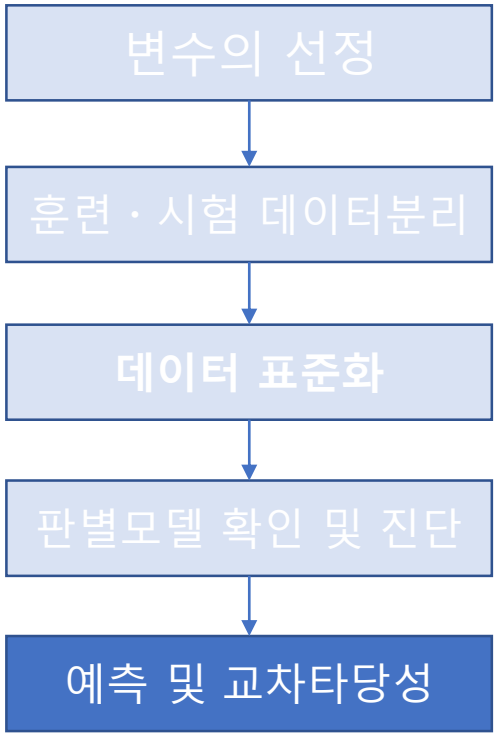
e, # 가구 총소득

지 분류

```
  Cell Contents
|-------------------------|
|                       N |
| Chi-square contribution |
|           N / Row Total |
|           N / Col Total |
|         N / Table Total |
|-------------------------|

Total Observations in Table:  7832

             | testpred$class
      test.y |       기혼 |       비혼 |  사별/별거 |  Row Total |
-------------|------------|------------|------------|------------|
        기혼 |       5184 |        273 |        108 |       5565 |
             |    124.854 |    427.382 |     69.346 |            |
             |      0.932 |      0.049 |      0.019 |      0.711 |
             |      0.830 |      0.218 |      0.325 |            |
             |      0.662 |      0.035 |      0.014 |            |
-------------|------------|------------|------------|------------|
        비혼 |        598 |        968 |          8 |       1574 |
             |    344.456 |   2039.655 |     51.681 |            |
             |      0.380 |      0.615 |      0.005 |      0.201 |
             |      0.096 |      0.773 |      0.024 |            |
             |      0.076 |      0.124 |      0.001 |            |
-------------|------------|------------|------------|------------|
   사별/별거 |        466 |         11 |        216 |        693 |
             |     13.642 |     89.873 |   1185.590 |            |
             |      0.672 |      0.016 |      0.312 |      0.088 |
             |      0.075 |      0.009 |      0.651 |            |
             |      0.059 |      0.001 |      0.028 |            |
-------------|------------|------------|------------|------------|
Column Total |       6248 |       1252 |        332 |       7832 |
             |      0.798 |      0.160 |      0.042 |            |
-------------|------------|------------|------------|------------|
```

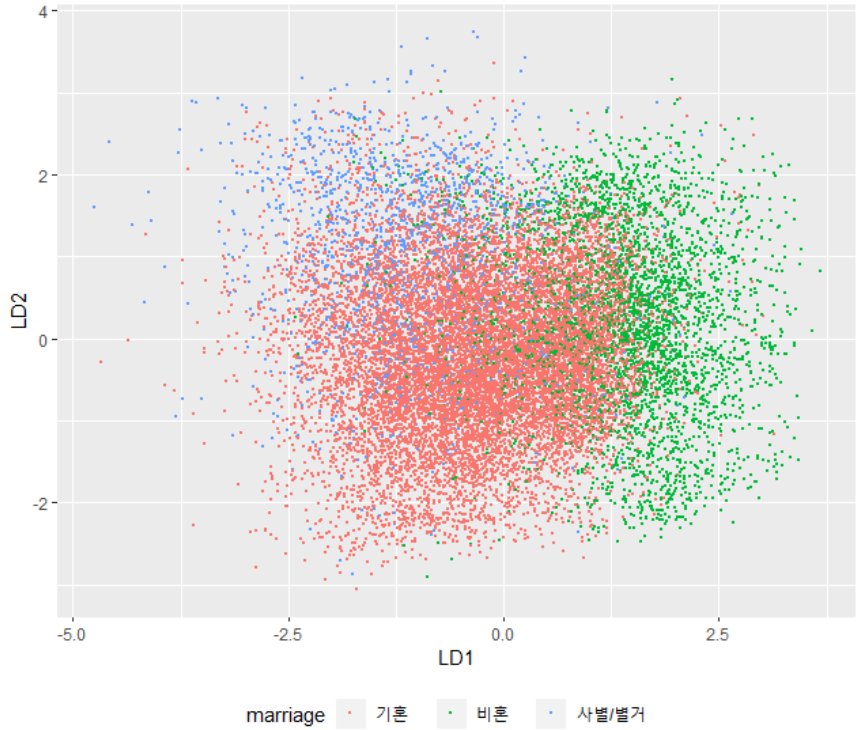
판별분석

- 선형 판별분석



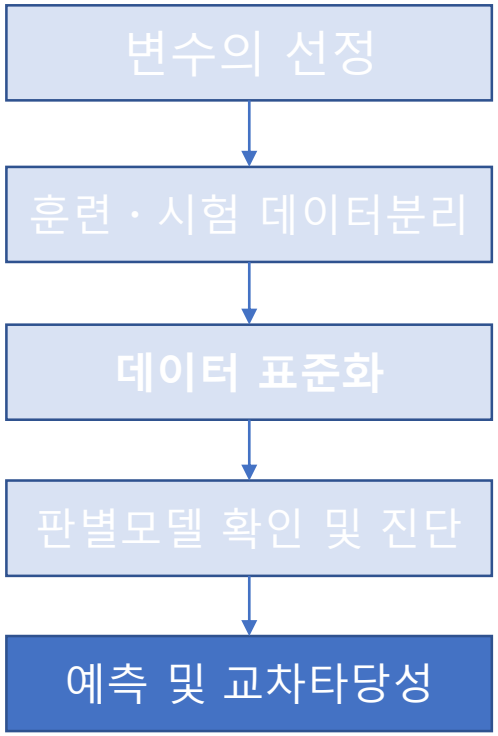
- 결혼상태(기혼, 비혼, 사별/별거)의 선형판별
  - gender, # 성별, age, # 만나이, edu.level, # 교육수준, hh.income, # 가구 총소득
  - feel.stress, # 스트레스 정도, s.reside.year, # 서울 거주년수,
  - h.reside.year, # 현 주택 거주년수, commuting.area # 통근 목적지 분류

```
> plot(model, cex = 0.5)
> # 실측치와 예측치 자료 합치기
> lda.data <- cbind(train.transformed, predict(model)$x)
> ggplot(lda.data, # ggplot 그래픽 틀 설정
+        aes(LD1, LD2)) + # x, y 축 설정
+   geom_point(aes(color = marriage), # marriage 별 색상
+              cex=0.5) + # 점의 크기
+   theme(legend.position = "bottom") # 범례 위치
```



판별분석

- 선형 판별분석



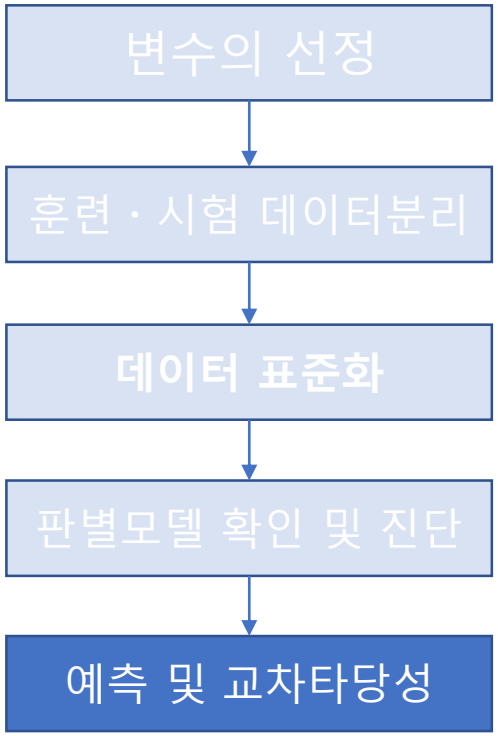
- 결혼상태(기혼, 비혼, 사별/별거)의 선형판별
  - gender, # 성별, age, # 만나이, edu.level, # 교육수준, hh.income, # 가구 총소득
  - feel.stress, # 스트레스 정도, s.reside.year, # 서울 거주년수,
  - h.reside.year, # 현 주택 거주년수, commuting.area # 통근 목적지 분류

```
> predictions <- model %>% predict(test.transformed) # 훈련 모델로 시험 데이터 예측
> class(predictions)
[1] "list"
> names(predictions) # 리스트 이름 확인
[1] "class"     "posterior" "x"
> # Predicted classes
> head(predictions$class, 6)
[1] 기혼      기혼      기혼      비혼      기혼      사별/별거
Levels: 기혼 비혼 사별/별거
> # Predicted probabilities of class memebership.
> head(predictions$posterior, 6)
        기혼         비혼    사별/별거
5  0.6107539 0.001298120 0.387948003
6  0.8422188 0.008858350 0.148922897
8  0.9455768 0.013608850 0.040814390
12 0.3537495 0.642280512 0.003969997
14 0.7917807 0.199863496 0.008355814
17 0.3938375 0.003194763 0.602967734
> # Linear discriminants
> head(predictions$x, 3)
        LD1        LD2
5 -2.487192  0.5511220
6 -1.441827  0.1842647
8 -1.036854 -0.7229776
> mean(predictions$class==test.transformed$marriage) # 일치 확률 계산
[1] 0.8130746
```

판별분석

추가 실습

- 선형 판별분석



- 결혼상태(기혼, 비혼, 사별/별거)의 선형판별
  - gender, # 성별, age, # 만나이, edu.level, # 교육수준, hh.income, # 가구 총소득
  - feel.stress, # 스트레스 정도, s.reside.year, # 서울 거주년수,
  - h.reside.year, # 현 주택 거주년수, commuting.area # 통근 목적지 분류

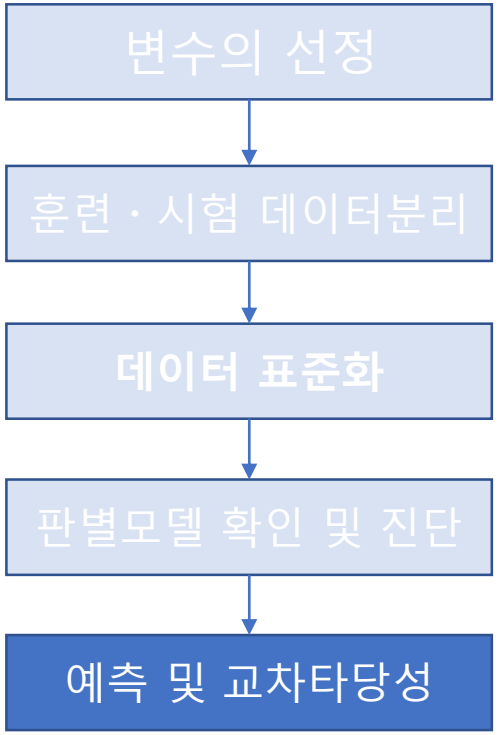
```
> library(caret)
> confusionMatrix(test.transformed$marriage, predictions$class)
Confusion Matrix and Statistics

          Reference
Prediction  기혼 비혼 사별/별거
  기혼      5184  273       108
  비혼       598  968         8
  사별/별거  466   11       216

Overall Statistics

               Accuracy : 0.8131
                 95% CI : (0.8043, 0.8217)
    No Information Rate : 0.7978
    P-Value [Acc > NIR] : 0.0003475

                  Kappa : 0.5295

 Mcnemar's Test P-Value : < 2.2e-16

Statistics by Class:

                     Class: 기혼 Class: 비혼 Class: 사별/별거
Sensitivity              0.8297      0.7732          0.65060
Specificity              0.7595      0.9079          0.93640
Pos Pred Value           0.9315      0.6150          0.31169
Neg Pred Value           0.5307      0.9546          0.98375
Prevalence               0.7978      0.1599          0.04239
Detection Rate           0.6619      0.1236          0.02758
Detection Prevalence     0.7105      0.2010          0.08848
Balanced Accuracy        0.7946      0.8405          0.79350
```

비표준화 LDA모델 결과와 비교

```
Overall Statistics

               Accuracy : 0.8044
                 95% CI : (0.7986, 0.8102)
    No Information Rate : 0.8019
    P-Value [Acc > NIR] : 0.1993

                  Kappa : 0.5047

 Mcnemar's Test P-Value : <2e-16
```

판별분석

- 선형 판별분석

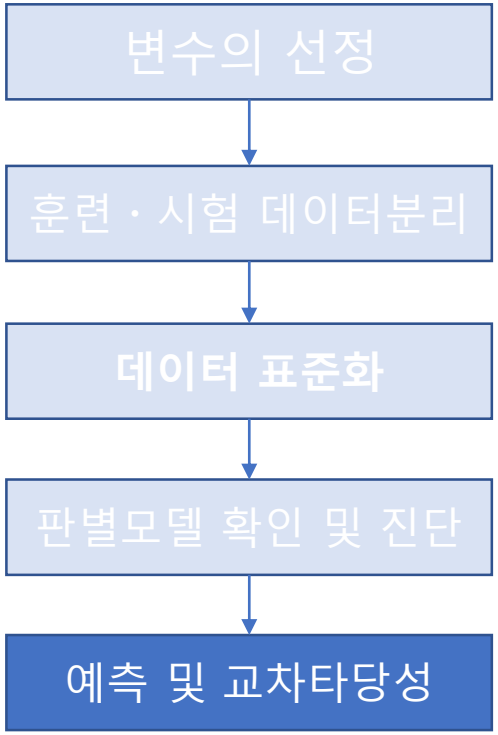


- 결혼상태(기혼, 비혼, 사별/별거)의 선형판별
  - gender, # 성별, age, # 만나이, edu.level, # 교육수준, hh.income, # 가구 총소득
  - feel.stress, # 스트레스 정도, s.reside.year, # 서울 거주년수,
  - h.reside.year, # 현 주택 거주년수, commuting.area # 통근 목적지 분류

```
> ## 분산-공분산 행렬이 동일한지 체크
> library(biotools)
> xx <- test.transformed %>% dplyr::select(-marriage)
> boxM(xx, test.transformed$marriage)

        Box's M-test for Homogeneity of Covariance Matrices

data:  xx
Chi-Sq (approx.) = 2066.9, df = 72, p-value < 2.2e-16
```

판별분석

- 선형 판별분석

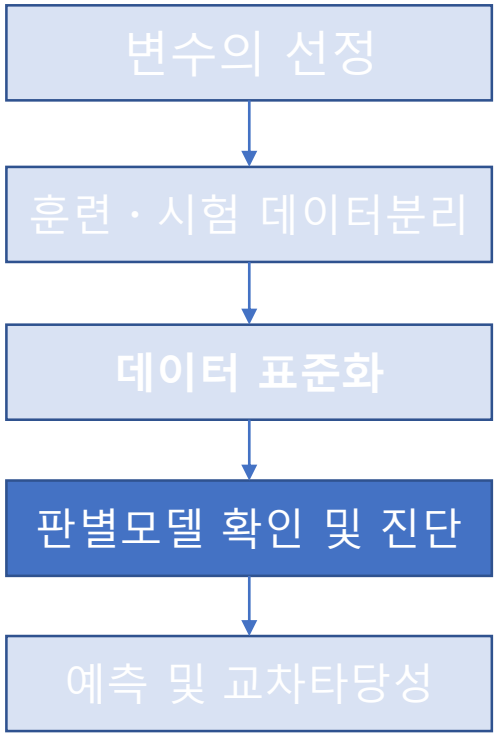


- 결혼상태(기혼, 비혼, 사별/별거)의 선형판별
  - gender, # 성별, age, # 만나이, edu.level, # 교육수준, hh.income, # 가구 총소득
  - feel.stress, # 스트레스 정도, s.reside.year, # 서울 거주년수,
  - h.reside.year, # 현 주택 거주년수, commuting.area # 통근 목적지 분류

```
> ## 분산-공분산 행렬이 동일한지 체크
> library(biotools)
> xx <- test.transformed %>% dplyr::select(-marriage)
> boxM(xx, test.transformed$marriage)

        Box's M-test for Homogeneity of Covariance Matrices

data:  xx
Chi-Sq (approx.) = 2066.9, df = 72, p-value < 2.2e-16
```

판별분석

- 다항 판별분석

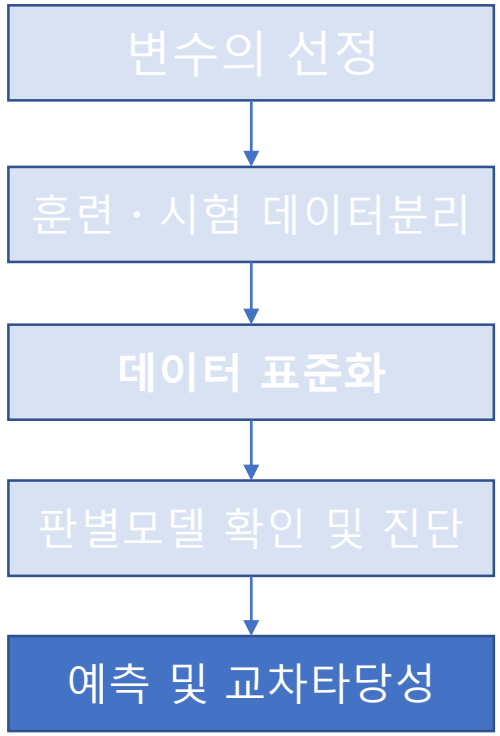


- 결혼상태(기혼, 비혼, 사별/별거)의 다항판별
  - gender, # 성별, age, # 만나이, edu.level, # 교육수준, hh.income, # 가구 총소득
  - feel.stress, # 스트레스 정도, s.reside.year, # 서울 거주년수,
  - h.reside.year, # 현 주택 거주년수, commuting.area # 통근 목적지 분류

```
> ### Quadratic discriminant analysis - QDA
> library(MASS)
> # Fit the model
> model <- qda(marriage ~  # 결혼상태
+                gender + # 성별
+                age + # 만나이
+                edu.level + # 교육수준
+                hh.income + # 가구 총소득
+                feel.stress + # 스트레스 정도
+                s.reside.year + # 서울 거주년수
+                h.reside.year + # 현 주택 거주년수
+                commuting.area, # 통근지 유형
+              data = train.transformed)
> model
Call:
qda(marriage ~ gender + age + edu.level + hh.income + feel.stress +
    s.reside.year + h.reside.year + commuting.area, data = train.transformed)

Prior probabilities of groups:
      기혼       비혼  사별/별거
0.71053352 0.20103967 0.08842681

Group means:
              gender        age   edu.level   hh.income  feel.stress s.reside.year h.reside.year commuting.ar
기혼      -0.1273409  0.1976832 -0.03358157  0.09584606 -0.006868069     0.1006722   -0.02388987    -0.013670
비혼       0.1965592 -1.0514234  0.42590335 -0.00249517  0.009435718    -0.4773390    0.03880451     0.227713
사별/별거  0.5763385  0.8019885 -0.69846053 -0.76447635  0.033734555     0.2763086    0.10373899    -0.407865
```

판별분석

- 다항 판별분석



- 결혼상태(기혼, 비혼, 사별/별거)의 다항판별
  - gender, # 성별, age, # 만나이, edu.level, # 교육수준, hh.income, # 가구 총소득
  - feel.stress, # 스트레스 정도, s.reside.year, # 서울 거주년수,
  - g.area # 통근 목적지 분류

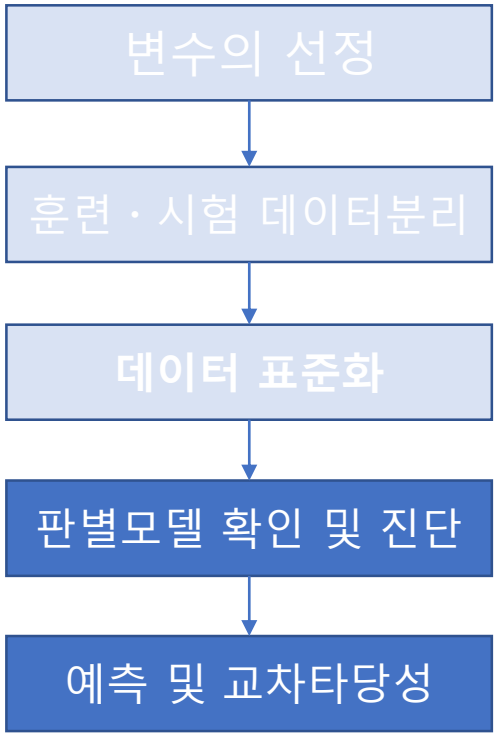
```
> # Make predictions
> predictions <- model %>% predict(test.transformed)
> # Model accuracy
> mean(predictions$class == test.transformed$marriage)
[1] 0.8262257
> library(caret)
> confusionMatrix(test.transformed$marriage, predictions$class)
Confusion Matrix and Statistics

          Reference
Prediction  기혼 비혼 사별/별거
  기혼      5096  303       166
  비혼       448 1106        20
  사별/별거  398   26       269

Overall Statistics

               Accuracy : 0.8262
                 95% CI : (0.8176, 0.8346)
    No Information Rate : 0.7587
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.5852

 Mcnemar's Test P-Value : < 2.2e-16

Statistics by Class:

                     Class: 기혼 Class: 비혼 Class: 사별/별거
Sensitivity              0.8576      0.7707          0.59121
Specificity              0.7519      0.9268          0.94252
Pos Pred Value           0.9157      0.7027          0.38817
Neg Pred Value           0.6268      0.9474          0.97395
Prevalence               0.7587      0.1832          0.05809
Detection Rate           0.6507      0.1412          0.03435
Detection Prevalence     0.7105      0.2010          0.08848
Balanced Accuracy        0.8047      0.8488          0.76687
```

표준화 선형판별 함수 결과와 비교

```
Overall Statistics

               Accuracy : 0.8126
                 95% CI : (0.8037, 0.8212)
    No Information Rate : 0.7976
    P-Value [Acc > NIR] : 0.0004754

                  Kappa : 0.5283

 Mcnemar's Test P-Value : < 2.2e-16
```

판별분석

- Mixture discriminant analysis(MDA)



- 결혼상태(기혼, 비혼, 사별/별거)의 Mixture discriminant analysis(MDA)

```
> library(mda)
> ?mda() # Mixture Discriminant Analysis
> # Fit the model
> model <- mda(marriage ~   # 결혼상태
+                gender + # 성별
+                age + # 만나이
+                edu.level + # 교육수준
+                hh.income + # 가구 총소득
+                feel.stress + # 스트레스 정도
+                s.reside.year + # 서울 거주년수
+                h.reside.year + # 현 주택 거주년수
+                commuting.area, # 통근지 유형
+              data = train.transformed)
> model
Call:
mda(formula = marriage ~ gender + age + edu.level + hh.income +
    feel.stress + s.reside.year + h.reside.year + commuting.area,
    data = train.transformed)

Dimension: 8

Percent Between-Group Variance Explained:
    v1     v2     v3     v4     v5     v6     v7     v8
 74.40  90.20  96.68  99.73  99.89  99.97  99.99 100.00

Degrees of Freedom (per dimension): 9

Training Misclassification Error: 0.21663 ( N = 18275 )

Deviance: 21626.76
> # Make predictions
> predicted.classes <- model %>% predict(test.transformed)
> # Model accuracy
> mean(predicted.classes == test.transformed$marriage)
[1] 0.7925179
```

연습문제 04

- 표준화 한 훈련 데이터(train.transformed)와 시험 데이터(test.transformed)를 활용하여, marriage에 대하여 age, edu.level, hh.income의 세 변수로 선형 판별분석을 시행하였다.
  - 혼동행렬을 활용하여 분류에서의 정확도(accuracy)가 얼마인지 적으시오.

요약

- 분류와 머신러닝
  - 지도학습과 비지도 학습
  - 분류 알고리즘의 종류
  - 분류 모델의 평가
  - 교차타당성(cross validation)
- 분류와 예측
  - 로지스틱 회귀모델
    - 이항 로지스틱 모델
    - 일반화 가법 모델(gam)
  - 판별분석
    - 선형 판별모델
    - 비선형 판별모델
- 분류와 기계학습
  - K-근접 이웃 모델
  - 트리 모델
  - 배깅과 랜덤 포레스트
  - 부스팅
- 비지도학습과 기계학습
  - 주성분과 요인 분석
    - 주성분 분석
    - 요인분석
  - 군집(클러스터링)
    - K-평균 클러스터링
    - 계층적 클러스터링
    - 모델 기반 클러스터링

끝

- 질의와 토의(Question & Discussion)
  - 이번 강의 내용을 시청하고, 실행하면서 궁금한 점이나 어려운 점에 대하여 토의해봅시다.

- 다음 차 강의주제
  - **분류와 기계학습**