.권용재 15일 전

read.csv 로 데이터 프레임을 읽어들일때 컬럼 이름 깨짐 현상.

read.csv 로 데이터를 불러들일때 첫 컬럼 이름이 깨지는 현상을 혹시 colnames 같이 재지정하는 방법 말고 다른방법 아시는 분 계신가요..?

1.PNG (1.073 Kb)

댓글

.한상혁 15일 전

댓글: read.csv 로 데이터 프레임을 읽어들일때 컬럼 이름 깨짐 현상.

인코딩 문제라면 read.csv에서 인코딩 관련 인자들을 조정하는 방법이 있긴 한데 정확히 무슨 문제인지, 인코딩이 무엇인지 파악하려면 저 데이터가 있어야 할 거 같아요.

.최민기 11시간 전

댓글: read.csv 로 데이터 프레임을 읽어들일때 컬럼 이름 깨짐 현상.

read.csv (..., fileEncoding = "UTF-8-BOM")를 시도해셔요.

735

The UTF-8 BOM is a sequence of Bytes at the start of a text-stream (EF BB BF) that allows the reader to more reliably guess a file as being encoded in UTF-8.

Normally, the BOM is used to signal the endianness of an encoding, but since endianness is irrelevant to UTF-8, the BOM is unnecessary.

According to the Unicode standard, the **BOM for UTF-8 files is not recommended**:

> ## 2.6 Encoding Schemes
>
> ... Use of a BOM is neither required nor recommended for UTF-8, but may be encountered in contexts where UTF-8 data is converted from other encoding forms that use a BOM or where the BOM is used as a UTF-8 signature. See the “Byte Order Mark” subsection in *Section 16.8, Specials*, for more information.

share edit

edited Oct 30 '18 at 15:39

Deduplicator

38.2k ● 6 ● 52 ● 92

answered Feb 8 '10 at 18:33

Martin Cote

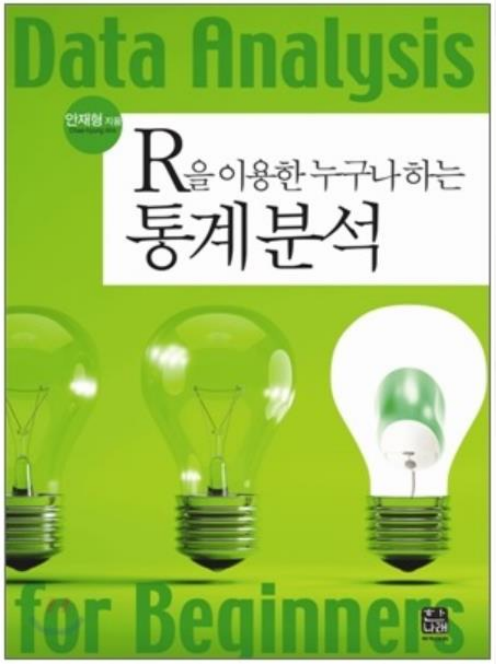
24.8k ● 13 ● 68 ● 97

105 It might not be recommended but from my experience in Hebrew conversions the BOM is sometimes crucial for UTF-8 recognition in Excel, and may make the difference between Jibrish and Hebrew – Matanya Dec 7 '12 at 8:13

22 It might not be recommended but it did wonders to my powershell script when trying to output "æøå" – Marius Nov 12 '13 at 9:22

Classification

분류: 로지스틱 회귀분석과 판별분석



1. 로지스틱 회귀분석 소개
2. 로지스틱 회귀분석 방법
3. 선형판별분석 소개
4. 선형판별분석 방법

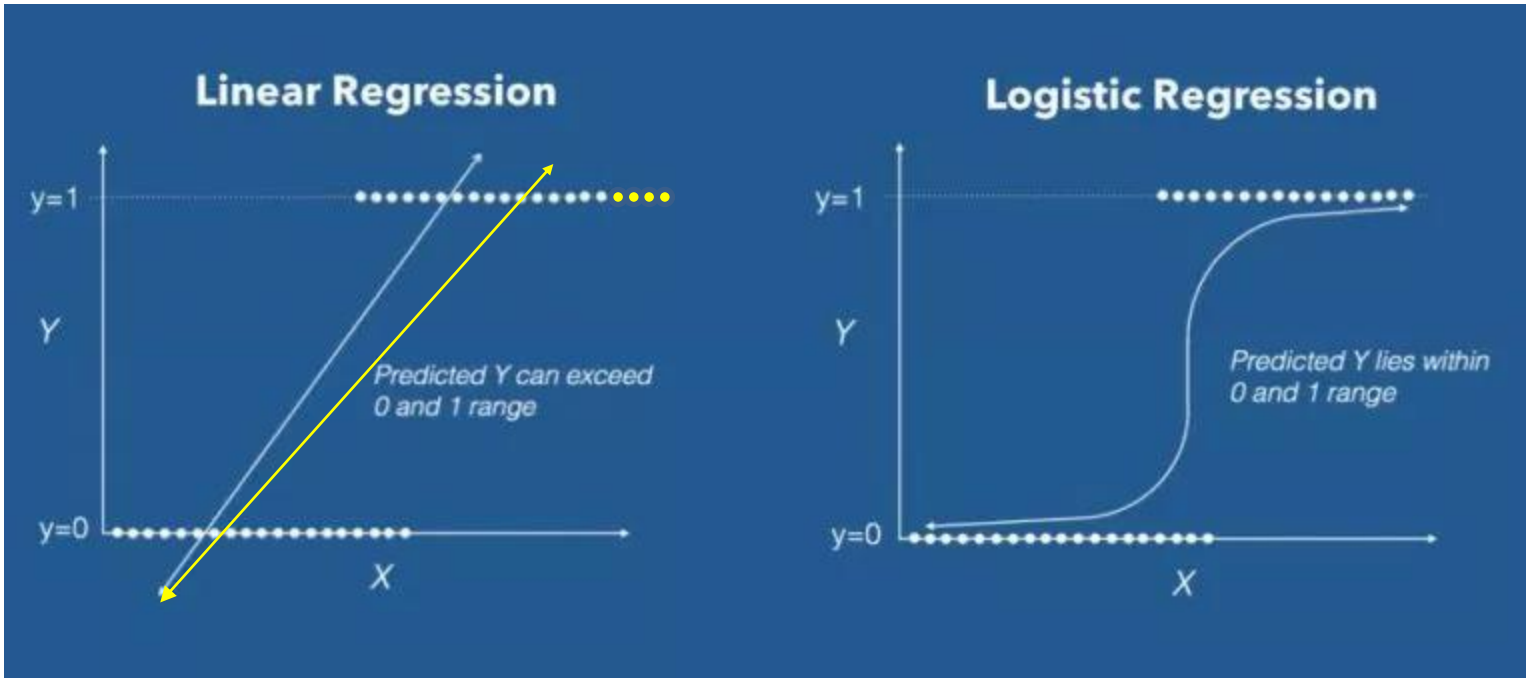
1. 로지스틱 회귀분석

1. 소개

- 독립변수와 **(0/1로 구분되는) 종속변수** 간의 인과관계를 검증하기 위한 방법
  - ✓ 독립변수는 명목/등간/비율척도로 측정된 변수
  - ✓ 종속변수는 **0/1의 이진(binary) 명목척도**로 측정된 변수
  - ✓ **Odds(승산)의 로그변환값(logit)**을 실제 종속변수로 사용
- 예제
  - ✓ 스팸메일 여부 예측 (스팸/정상)
  - ✓ 신용카드 부정사용 여부 예측 (부정/정상)
  - ✓ 조직검사로 악성종양 여부 예측 (악성/정상)
  - ✓ 고객의 구매 가능성 예측 (구매/비구매)
  - ✓ 대출고객사의 부도 예측 (부도/정상)

1. 로지스틱 회귀분석

2. 로지스틱 회귀분석의 필요성

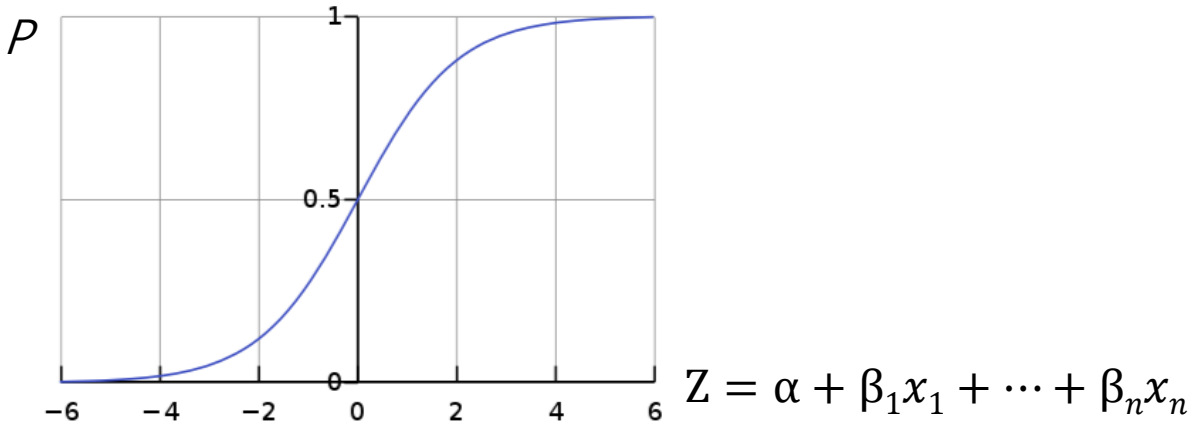


(https://www.machinelearningplus.com/logistic-regression-tutorial-examples-r/)

- 선형 회귀분석은 Y값의 예측치가 **0과 1사이를 벗어나며, X값의 범위에 따라 기울기가 변동**
- 로지스틱 회귀분석은 **Y값이 1이 될 확률**을 예측

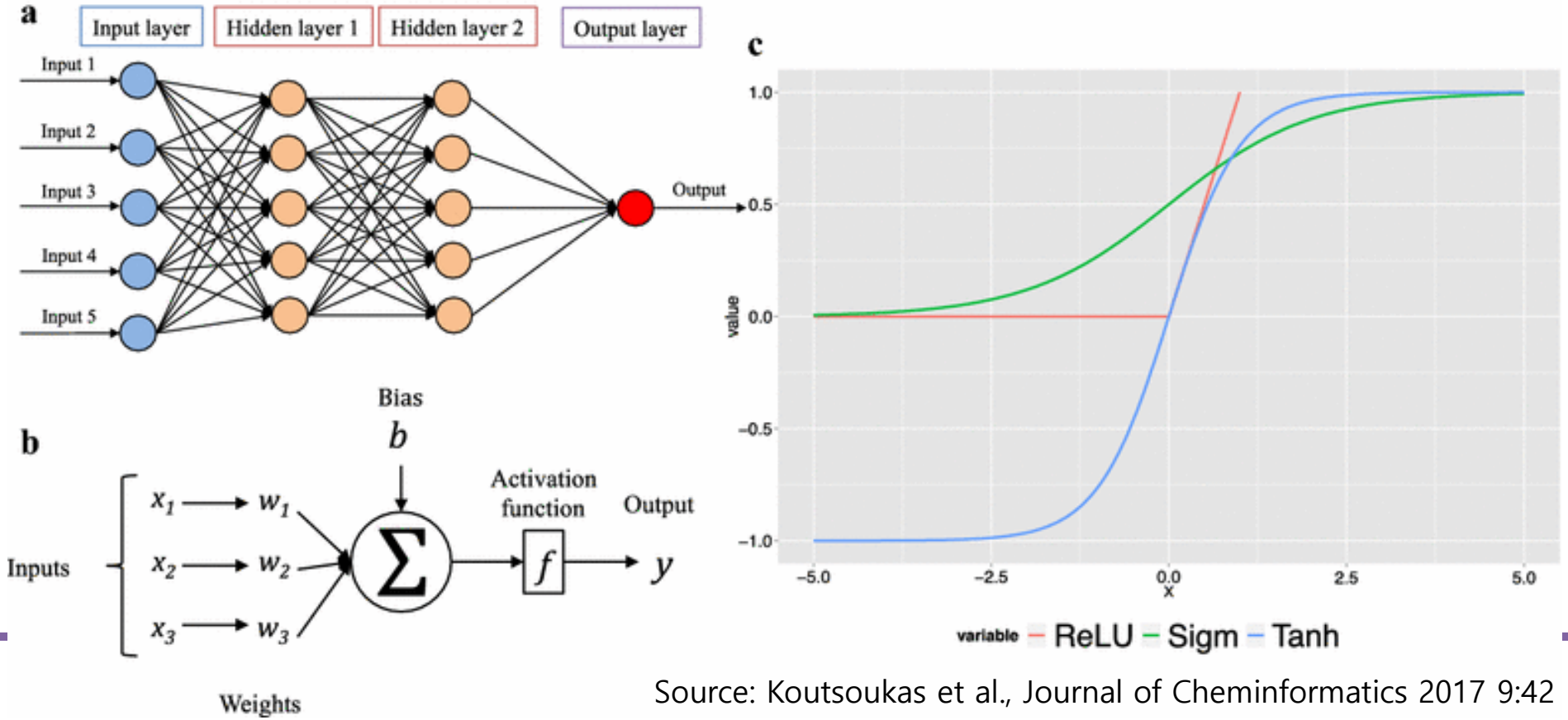
3. 로지스틱 모델

- Odds(승산)의 로그변환값(log-odds or logit)을 종속변수로 활용
  - ✓ $\text{Odds} = \frac{P}{1-P}$
  - ✓ $\text{Z} = ln(\frac{P}{1-P}) = \alpha + \beta_1 x_1 + \cdots + \beta_n x_n$
  - ✓ $P = E(y = 1|X) = \frac{1}{1+e^{-z}} = \frac{e^Z}{1+e^Z}$ (Sigmoid(or logistic) function)



1. 로지스틱 회귀분석

- ✓ P가 logit 함수 $\left(ln\left(\frac{P}{1-P}\right)\right)$ 를 매개로 x_i들의 선형식으로 표현되므로 로지스틱 회귀분석을 logit-link를 가지는 **generalized linear model**이라고도 함
- ✓ 신경망의 활성화(activation) 함수로 사용되는 대표적 함수의 하나가 logistic regression에 활용되는 Sigmoid 함수



Source: Koutsoukas et al., Journal of Cheminformatics 2017 9:42

*Generalized linear model(일반화선형모델)

- 종속변수가 독립변수들의 선형식으로 표현된다는 가정과 종속변수가 정규분포를 따른다는 가정을 완화시킨 것이 일반화 선형 모형(generalized linear model)

Distribution	Name	Link Function	Mean Function
Normal	Identity	$\mathbf{X}\boldsymbol{\beta} = \mu$	$\mu = \mathbf{X}\boldsymbol{\beta}$
Exponential Gamma	Inverse	$\mathbf{X}\boldsymbol{\beta} = \mu^{-1}$	$\mu = (\mathbf{X}\boldsymbol{\beta})^{-1}$
Inverse Gaussian	Inverse squared	$\mathbf{X}\boldsymbol{\beta} = \mu^{-2}$	$\mu = (\mathbf{X}\boldsymbol{\beta})^{-1/2}$
Poisson	Log	$\mathbf{X}\boldsymbol{\beta} = \ln(\mu)$	$\mu = \exp(\mathbf{X}\boldsymbol{\beta})$
Binomial	Logit	$\mathbf{X}\boldsymbol{\beta} = \ln\left(\frac{\mu}{1-\mu}\right)$	$\mu = \frac{\exp(\mathbf{X}\boldsymbol{\beta})}{1+\exp(\mathbf{X}\boldsymbol{\beta})}$

*최대우도법 (maximum likelihood method)

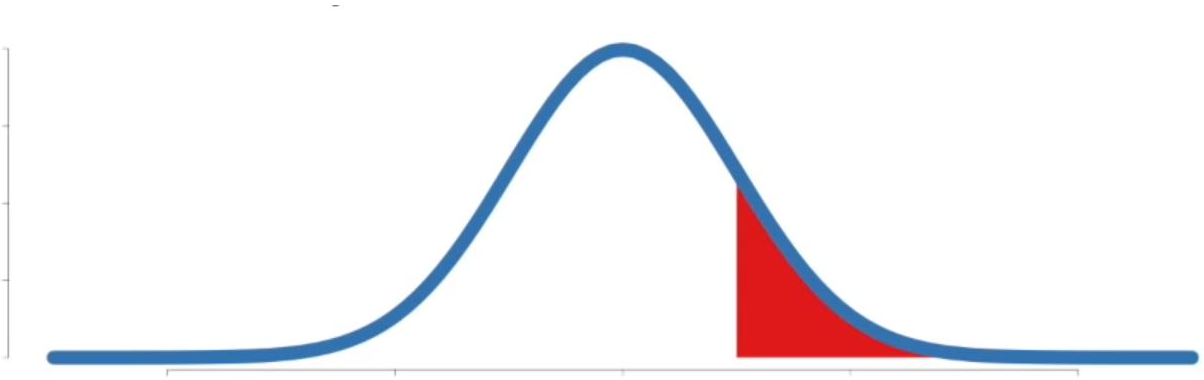
1. 선형회귀분석에서는 최소자승법에 의해 x와 y간의 관계를 추정하는 반면, **일반화선형모형에서는 최대우도법을** 사용해 회귀식을 추정

2. 관측치가 고정되고, 그러한 관측치가 관찰될 가능성이 가장 높은 모수값을 추정하는데, **"해당 관측치가 관찰될 가능성"이 우도**

*Probabilities vs. Likelihoods (by StatQuest)

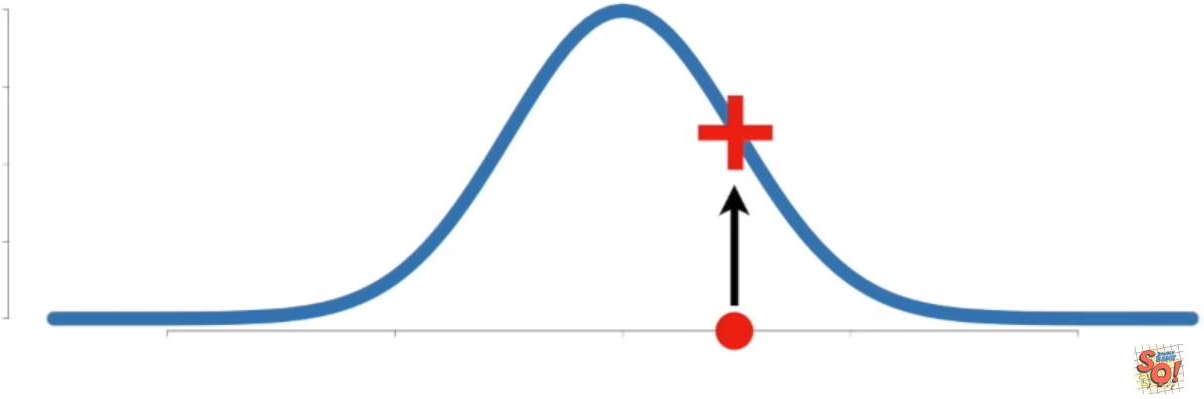
Probabilities are the areas under a fixed distribution…

pr(data | distribution)

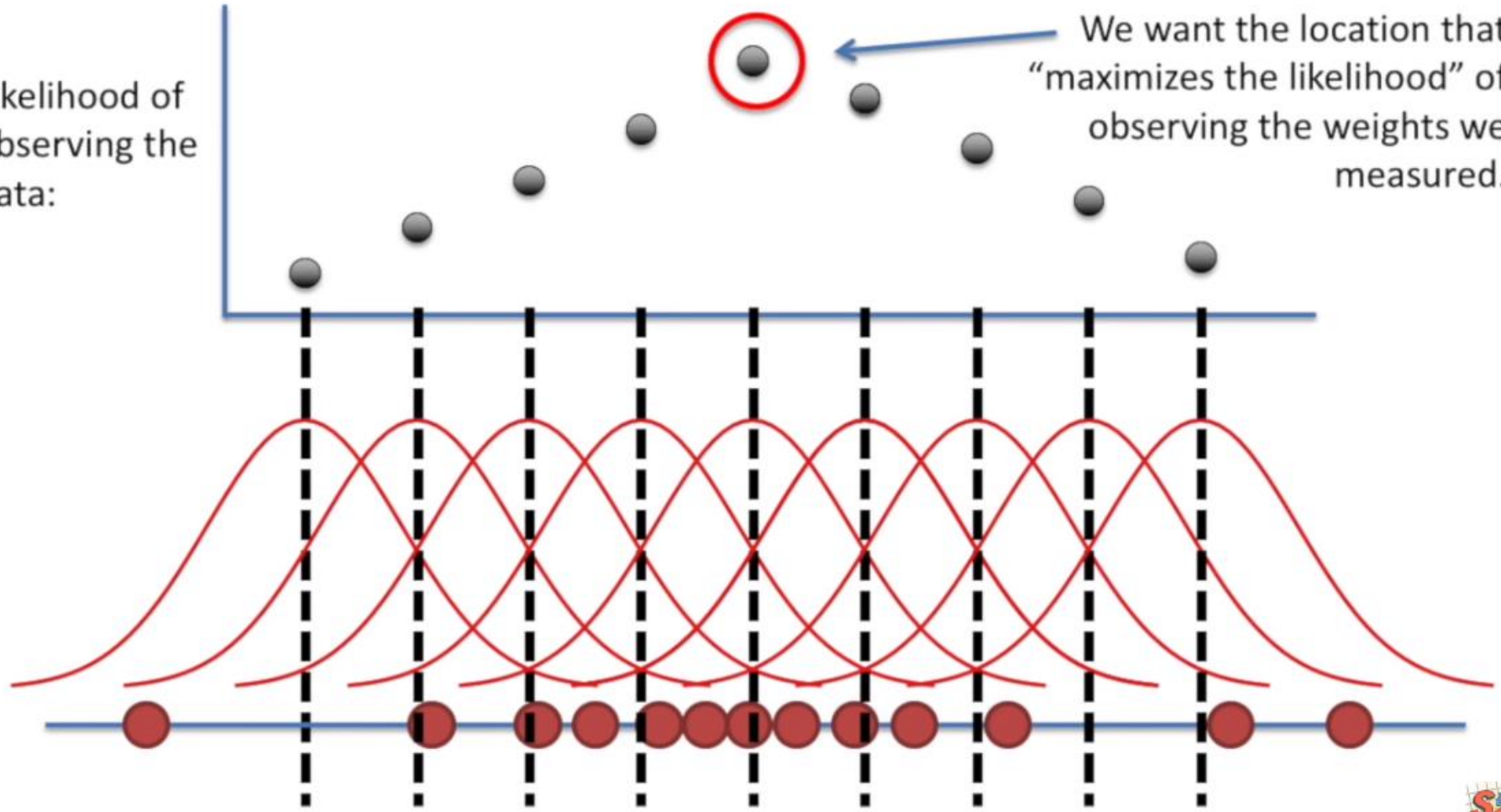


Likelihoods are the y-axis values for fixed data points with distributions that can be moved…

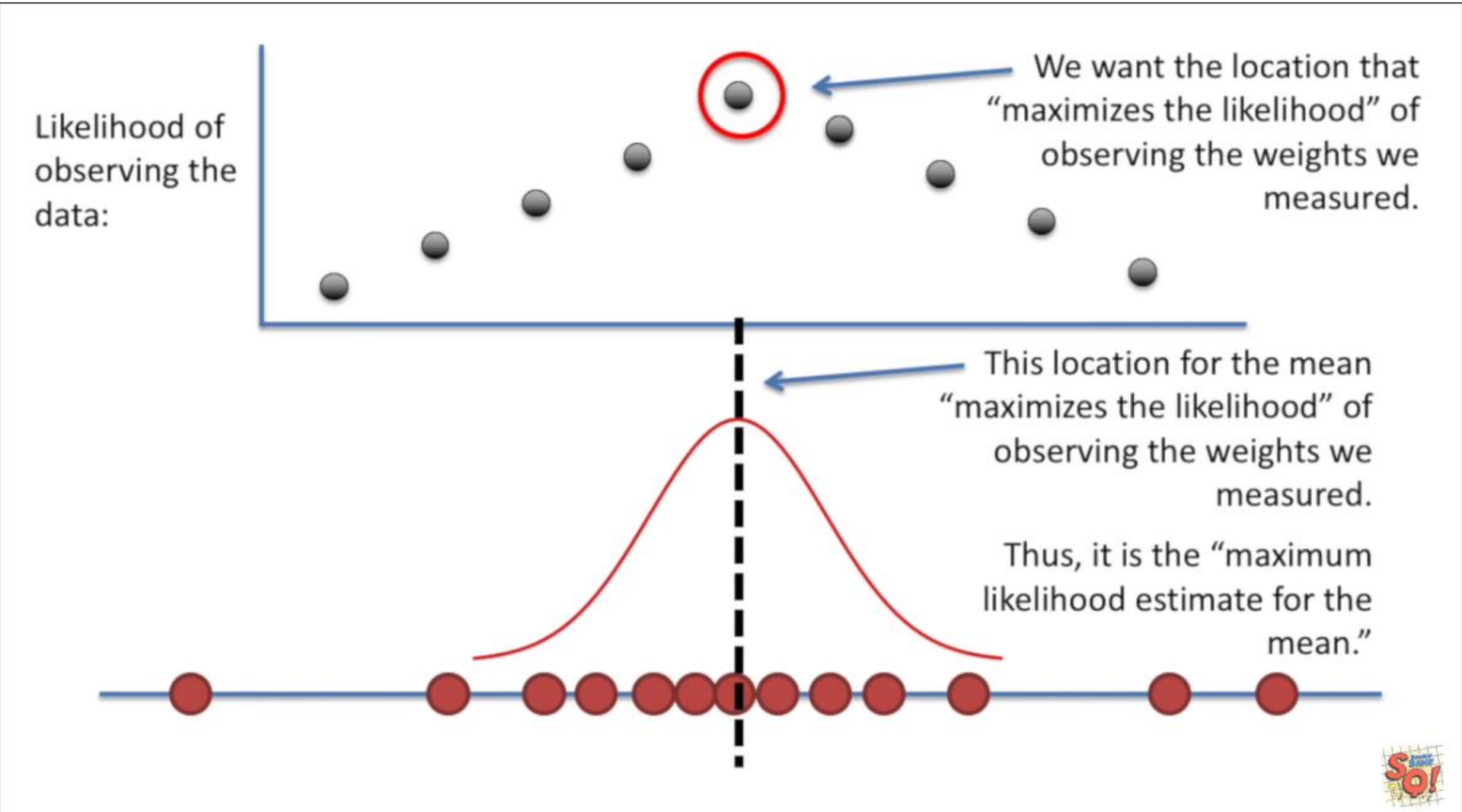
L(distribution | data)



*Maximum Likelihood Est. (by StatQuest)



*Maximum Likelihood Est. (by StatQuest)



2. 로지스틱 회귀분석 방법

1. Dose-Response 실험

- 쥐 30마리를 10마리 세 그룹으로 나눈 후 약물의 양(dose)을 0, 1, 2를 주입해 10마리의 쥐 중 몇 마리가 죽는지 실험
- Dose의 증가가 쥐의 사망률을 높이는지 확인

```
# ① 자료 로딩(toxic.csv)
toxic <- read.csv ("toxic.csv")
toxic #30 records
# ② 로지스틱 회귀분석 (예측모델)
logreg1 <- glm(response ~ dose, family=binomial, data=toxic)
summary(logreg1)
# ③ 로지스틱 회귀분석 (Null 모델)
logreg1Null <- glm(response ~ 1, family = binomial, data = toxic)
summary(logreg1Null)
# ④ 모델 평가
anova(logreg1Null, logreg1, test = "LRT")
```

2. 로지스틱 회귀분석 방법

```
> toxic <- read.csv("toxic.csv")
> toxic
   dose response
1     0        0     #7개
8     0        1     #3개
11    1        0     #5개
16    1        1     #5개
21    2        0     #2개
23    2        1     #8개
```

- ✓ 로지스틱 회귀분석을 시행하기 위해 glm 함수를 이용
  - glm 함수에는 종속변수와 독립변수를 차례대로 입력하고, 종속변수와 독립변수 사이에는 '~' 표시
  - 종속변수는 response이고, 독립변수가 dose이므로 response ~ dose라고 입력
  - family = **binomial**로 지정하여 로지스틱 회귀분석을 실행
  - 종속변수와 독립변수가 있는 데이터를 지정(data=toxic)
  - summary 함수에 이 객체를 지정하면 분석 결과를 확인

2. 로지스틱 회귀분석 방법

```
> logreg1 <- glm(response ~ dose, family = binomial, data = toxic)
> summary(logreg1)

Call:
glm(formula = response ~ dose, family = binomial, data = toxic)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.7401  -0.8105   0.7050   1.0088   1.5956

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  -0.9446     0.6360  -1.485   0.1375
dose          1.1051     0.5186   2.131   0.0331 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 41.455  on 29  degrees of freedom
Residual deviance: 36.196  on 28  degrees of freedom
AIC: 40.196

Number of Fisher Scoring iterations: 4
```

2. 로지스틱 회귀분석 방법

- 분석결과
  - ✓ **Deviance**는 Generalized Linear Model의 적합도(goodness of fit)를 보여주는 수치 (값이 낮을수록 적합도는 높음)
  - ✓ **Null** deviance는 절편만 있는 null 모델이 종속변수를 얼마나 잘 예측하는지를 제시 (41.455)
    - 2(~~LL(Saturated Model)~~ - LL(Null Model)) on df = (30 – 1)
  - ✓ **Residual** deviance는 독립변수들을 포함한 모델이 종속변수를 얼마나 잘 예측하는지를 제시 (36.195)
    - 2(~~LL(Saturated Model)~~ - LL(Proposed Model)) on df = (30 – 2)
  - ✓ 본 모델에서는 1개의 독립변수를 Null 모델에 추가하여 Deviance를 5.26만큼 감소시킴 → **2(LL(Proposed Model) - LL(Null Model)), df = 1(df_Proposed – df_Null)인 chi-square 분포를 기준으로** 유의성 확인 가능
    - Likelihood-ratio test (LR test): 두 모델(Null model과 Proposed model) 간의 적합도를 비교하기 위해 Proposed model이 Null model보다 몇 배나 발생가능성이 높은 지(LR)를 찾고, 이를 통해 p 값을 계산하여 모델의 유의성을 판단

2. 로지스틱 회귀분석 방법

```
> summary(logreg1Null)

Call:
glm(formula = response ~ 1, family = binomial, data = toxic)

Deviance Residuals:
   Min      1Q  Median      3Q     Max
-1.235  -1.235   1.121   1.121   1.121

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)   0.1335     0.3660   0.365    0.715

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 41.455  on 29  degrees of freedom
Residual deviance: 41.455  on 29  degrees of freedom
AIC: 43.455

Number of Fisher Scoring iterations: 3
```

2. 로지스틱 회귀분석 방법

```
> anova(logreg1Null, logreg1, test = "LRT")
Analysis of Deviance Table

Model 1: response ~ 1
Model 2: response ~ dose
  Resid. Df Resid. Dev Df Deviance Pr(>Chi)
1        29     41.455
2        28     36.196  1   5.2593  0.02183 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

> anova(logreg1Null, logreg1, test = "Chisq")
Analysis of Deviance Table

Model 1: response ~ 1
Model 2: response ~ dose
  Resid. Df Resid. Dev Df Deviance Pr(>Chi)
1        29     41.455
2        28     36.196  1   5.2593  0.02183 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

*Deviance, AIC, BIC

1. $Deviance = 2(\ln(\widehat{L_S}) - \ln(\hat{L})) = -2\ln(\hat{L})$
   - $\widehat{L_S}$: 포화 모델의 Likelihood function의 최대값은 1, 그러므로 $\ln(\widehat{L_S}) = 0$
   - $\hat{L}$: 모델의 Likelihood function의 최대값
2. $AIC = 2k - 2\ln(\hat{L})$
   - Akaike information criterion
   - k : 모델에서 추정해야 하는 모수(절편 포함)의 개수
3. $BIC = \ln(n) * k - 2\ln(\hat{L})$
   - Bayesian information criterion
   - n: 샘플 데이터 개수(크기)

2. 로지스틱 회귀분석 방법

- 분석결과
  - ✓ 로지스틱 회귀분석에서 계수 값은 해당 독립변수가 1unit 증가할 때, 관심 사건의 log-odds $\left(ln\left(\frac{P}{1-P}\right)\right)$ 가 그만큼 증가한다는 의미
  - ✓ 해석이 어려우므로, 계수를 지수 변환을 하면 odds ratio가 되어, 관심 사건이 발생할 odds $\left(\frac{P}{1-P}\right)$가 e^{coeff}배 증가한다고 해석 가능
    - **경로계수보다 odds ratio(e^{coeff})를 보고하는 경우가 다수**
    - e^{coeff} 가 1보다 크면 odds가 증가, 1보다 작으면 odds가 감소
    - **dose가 1unit 증가하면 "odds"가 3(=3.0193924)배 증가**

```
> coef(logreg1)
(Intercept)        dose
 -0.9445574   1.1050556
> exp(coef(logreg1))
(Intercept)        dose
  0.3888516   3.0193924
```

```
> confint(logreg1, parm = "dose")
2.5 %    97.5 %
0.1535415 2.2341929
> exp(confint(logreg1, parm =
"dose"))
2.5 %   97.5 %
1.165956 9.338941
```


124

.한상혁

1일 전

R에서 사용할 수 있는 간단한 팁들

1. 현재 만든 그래프 및 그 설정을 초기화 시켜줄 때 Plots 창에서 빗자루 아이콘을 클릭하셔서 그래프를 지우는 방법도 있지만 dev.off() 코드를 사용해서 더 쉽게 제거하실 수도 있습니다.

```
par(mfrow=c(1, 2))

plot(c(1, 2, 3), c(3, 4, 5))

plot(c(1, 3), c(2, 4))

dev.off()
```

2. %in% 기능을 통하여 특정 원소의 포함 여부를 확인하실 수 있습니다.

```
> "a" %in% c("a", "b", "c")

[1] TRUE
```

3. grepl 함수를 통하여 특정 패턴이 문자열에 포함되어 있는지를 확인하실 수 있습니다.

.권용재 12시간 전

merge 한 데이터를 가지고 변수를 바꾸려고 할때

GDP_w_reg$country[GDP_w_reg$country == "Bosnia and Herzegovina"] <- "Bosnia"

이런 코드를 쓰면 Bosnia로 바뀌어야 하는데 자꾸 NA값으로 나오네요.. 혹시 이유를 아시는분 있을까요? 해결책이나..

GDP_w_reg 데이터는 이미 한번 merge 한 데이터입니다

댓글

.권용재 12시간 전

댓글: merge 한 데이터를 가지고 변수를 바꾸려고 할때

아 해결했습니다.... factor 값으로 되어있는 데이터라서 NA가 생성되는 것이였네요 as.character로 바꿔서 해결했습니다.

.최민기

5시간 전

회귀분석 관련 함수 predict()

회귀분석 관련 함수 중 predict()를 소개해 드릴까합니다.

predict()함수는 lm( )을 통해 모델을 만들고 나면 새로운 데이터에 대한 예측값을 구할 수 있습니다. predict( )는 인자로 주어진 모델에 따라 내부적으로 predict.glm( ), predict.lm( ), predict.nls( ) 등의 함수로 부를 수 있는데, 저희가 배운 선형 회귀모형의 경우 predict.lm을 사용합니다.

함수식은

predict.lm(model, newdata = A, interval = c("confidence", "prediction"). ...) 이며

#model : 예측에 사용할 회귀분석 결과식

#newdata : 예측에 사용할 x값

#interval : 지정된 x값에 대한 y의 confidence interval 또는 prediction interval을 출력한다.

ex)

grade0 <- lm(q18a1w1~q18a2w1, data=spssdata) - 회귀분석(국어,영어성적)

predict(grade0, newdata=data.frame(q18a2w1=4),interval="confidence") - 평균신뢰구간 구하기

predict(grade0, newdata=data.frame(q18a2w1=4),interval="prediction") - 예측구간 구하기

fit은 예측값의 점 추정치, lwr과 upr은 각각 해당 구간의 하한과 상한 값을 의미합니다.

```
> predict(grade0, newdata=data.frame(q18a2w1=4),interval="confidence")
       fit      lwr      upr
1 3.597483 3.519535 3.675432
```

```
> predict(grade0, newdata=data.frame(q18a2w1=4),interval=c("prediction"))
       fit      lwr      upr
1 3.597483 2.139716 5.055251
```

국어점수가 4라면 영어점수는 3.59라고 예측한다.

댓글 인용 수정 삭제 이메일

2. 로지스틱 회귀분석 방법

2. 인종차별과 사형판결 성향

✓ 미국 플로리다 주에서 수집된 인종차별에 대한 데이터로 **희생자의 인종(victim)과 피고의 인종(defendant)**이 백인인지 여부에 따라 사형 선고(death)에 영향을 주는지 조사한 자료

```
# ① 자료 로딩(death_penalty.csv)
death.penalty <- read.csv ("death_penalty.csv")
head(death.penalty)
table(death.penalty)
# ② 로지스틱 회귀분석 (예측모델)
logreg2 <- glm(death ~ victim + defendant, family = binomial, data
= death.penalty)
summary(logreg2)
# ③ 로지스틱 회귀분석 (Null 모델)
logreg2Null <- glm(death ~ 1, family = binomial, data =
death.penalty)
summary(logreg2Null)
# ④ 모델 평가
anova(logreg2, logreg2Null, test = "LRT")
```

2. 로지스틱 회귀분석 방법

```
> head(death.penalty)
  victim defendant death
1  White     White   yes
2  White     White   yes
3  White     White   yes
4  White     White   yes
5  White     White   yes
6  White     White   yes

> table(death.penalty)
, , death = no

       defendant
victim  Black White
  Black    97     9
  White    52   132

, , death = yes

       defendant
victim  Black White
  Black     6     0
  White    11    19
```

2. 로지스틱 회귀분석 방법

```
> logreg2 <- glm(death ~ victim + defendant, family = binomial, data =
death.penalty)
> summary(logreg2)

Call:
glm(formula = death ~ victim + defendant, family = binomial,
    data = death.penalty)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-0.6296  -0.5138  -0.5138  -0.3367   2.4078

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)   -2.8421     0.4203  -6.762 1.36e-11 ***
victimWhite    1.3242     0.5193   2.550   0.0108 *
defendantWhite -0.4402     0.4009  -1.098   0.2722
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 226.51  on 325  degrees of freedom
Residual deviance: 219.08  on 323  degrees of freedom
AIC: 225.08

Number of Fisher Scoring iterations: 5
```

2. 로지스틱 회귀분석 방법

```
> logreg2Null <- glm(death ~ 1, family = binomial, data = death.penalty)
> summary(logreg2Null)

Call:
glm(formula = death ~ 1, family = binomial, data = death.penalty)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-0.4838  -0.4838  -0.4838  -0.4838   2.0992

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  -2.0864     0.1767  -11.81   <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: 226.51  on 325  degrees of freedom
Residual deviance: 226.51  on 325  degrees of freedom
AIC: 228.51

Number of Fisher Scoring iterations: 4
```

2. 로지스틱 회귀분석 방법

```
> anova(logreg2Null, logreg2, test = "LRT")
Analysis of Deviance Table

Model 1: death ~ 1
Model 2: death ~ victim + defendant
  Resid. Df Resid. Dev Df Deviance Pr(>Chi)
1       325     226.51
2       323     219.08  2   7.4309  0.02434 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

> anova(logreg2Null, logreg2, test = "Chisq")
Analysis of Deviance Table

Model 1: death ~ 1
Model 2: death ~ victim + defendant
  Resid. Df Resid. Dev Df Deviance Pr(>Chi)
1       325     226.51
2       323     219.08  2   7.4309  0.02434 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

2. 로지스틱 회귀분석 방법

- 분석결과
  - ✓ victim과 defendant의 효과를 고려한 모델(logreg2)에서, victim만 유의한 영향력을 지니는 것으로 나타남
    - 피해자가 백인일 경우 사형 판결이 날 odds가 $e^{1.3242}$배, 즉, 3.759225배 늘어남
  - ✓ victim과 defendant의 효과를 고려한 모델(logreg2)이 Null 모델(logreg2Null)과 유의한 차이를 보이므로 예측 모델이 타당

2. 로지스틱 회귀분석 방법

3. 대학원 입학결정 요인

- GRE 성적, 학부 GPA, 학부 학교 Rank를 가지고 대학원 입학 허가 여부를 예측

```
# ① 자료 로딩(binary.csv from stats.idre.ucla.edu)
admission <- read.csv("binary.csv")
str(admission)
# ② 로지스틱 회귀분석 (rank as a continuous variable)
logreg3_1 <- glm(admit ~ gre + gpa + rank, data = admission,
family = binomial)
summary(logreg3_1)
coef(logreg3_1)
exp(coef(logreg3_1))
# ③ 로지스틱 회귀분석 (rank as a nominal variable)
logreg3_2 <- glm(admit ~ gre + gpa + as.factor(rank), data =
admission, family = binomial)
summary(logreg3_2)
coef(logreg3_2)
exp(coef(logreg3_2))
# ④ 모형 비교
anova(logreg3_1, logreg3_2, test = "LRT")
```

2. 로지스틱 회귀분석 방법

```
> logreg3_1 <- glm(admit ~ gre + gpa + rank, data = admission,
family = binomial)
> summary(logreg3_1)

Call:
glm(formula = admit ~ gre + gpa + rank, family = binomial, data =
admission)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.5802  -0.8848  -0.6382   1.1575   2.1732

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -3.449548   1.132846  -3.045  0.00233 **
gre          0.002294   0.001092   2.101  0.03564 *
gpa          0.777014   0.327484   2.373  0.01766 *
rank        -0.560031   0.127137  -4.405 1.06e-05 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 499.98  on 399  degrees of freedom
Residual deviance: 459.44  on 396  degrees of freedom
AIC: 467.44

Number of Fisher Scoring iterations: 4
```

2. 로지스틱 회귀분석 방법

```
> #check coefficients
> coef(logreg3_1)
(Intercept)         gre         gpa        rank
-3.44954840  0.00229396  0.77701357 -0.56003139
> exp(coef(logreg3_1))  #if the coefficient is larger than zero,
odds will increase (odds = e^coefficient)
(Intercept)         gre         gpa        rank
 0.03175998  1.00229659  2.17496718  0.57119114
```

- 분석결과
  - ✓ GRE, GPA, Rank가 모두 유의한 영향력을 가짐
    - 지원자 GRE 점수가 1점 올라갈 경우 합격할 odds가 $e^{0.0023}$배, 즉, 1.0023배로 늘어남
    - 지원자 GPA 점수가 1점 올라갈 경우 합격할 odds가 $e^{0.7770}$배, 즉, 2.1750배로 늘어남
    - 지원자 학부 Rank가 1단계 내려갈 경우 합격할 odds가 $e^{-0.5600}$배, 즉, 0.5711배로 줄어듦

```
# can we use linear regression?
library(sjPlot)
plot_model(lm(admit ~ gre + gpa + rank, data = admission),
type = "diag")
```

2. 로지스틱 회귀분석 방법

```
> logreg3_2 <- glm(admit ~ gre + gpa + as.factor(rank), data =
admission, family = binomial)
> summary(logreg3_2)

Call:
glm(formula = admit ~ gre + gpa + as.factor(rank), family =
binomial,
    data = admission)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.6268  -0.8662  -0.6388   1.1490   2.0790

Coefficients:
                   Estimate Std. Error z value Pr(>|z|)
(Intercept)      -3.989979   1.139951  -3.500 0.000465 ***
gre               0.002264   0.001094   2.070 0.038465 *
gpa               0.804038   0.331819   2.423 0.015388 *
as.factor(rank)2 -0.675443   0.316490  -2.134 0.032829 *
as.factor(rank)3 -1.340204   0.345306  -3.881 0.000104 ***
as.factor(rank)4 -1.551464   0.417832  -3.713 0.000205 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 499.98  on 399  degrees of freedom
Residual deviance: 458.52  on 394  degrees of freedom
AIC: 470.52

Number of Fisher Scoring iterations: 4
```

2. 로지스틱 회귀분석 방법

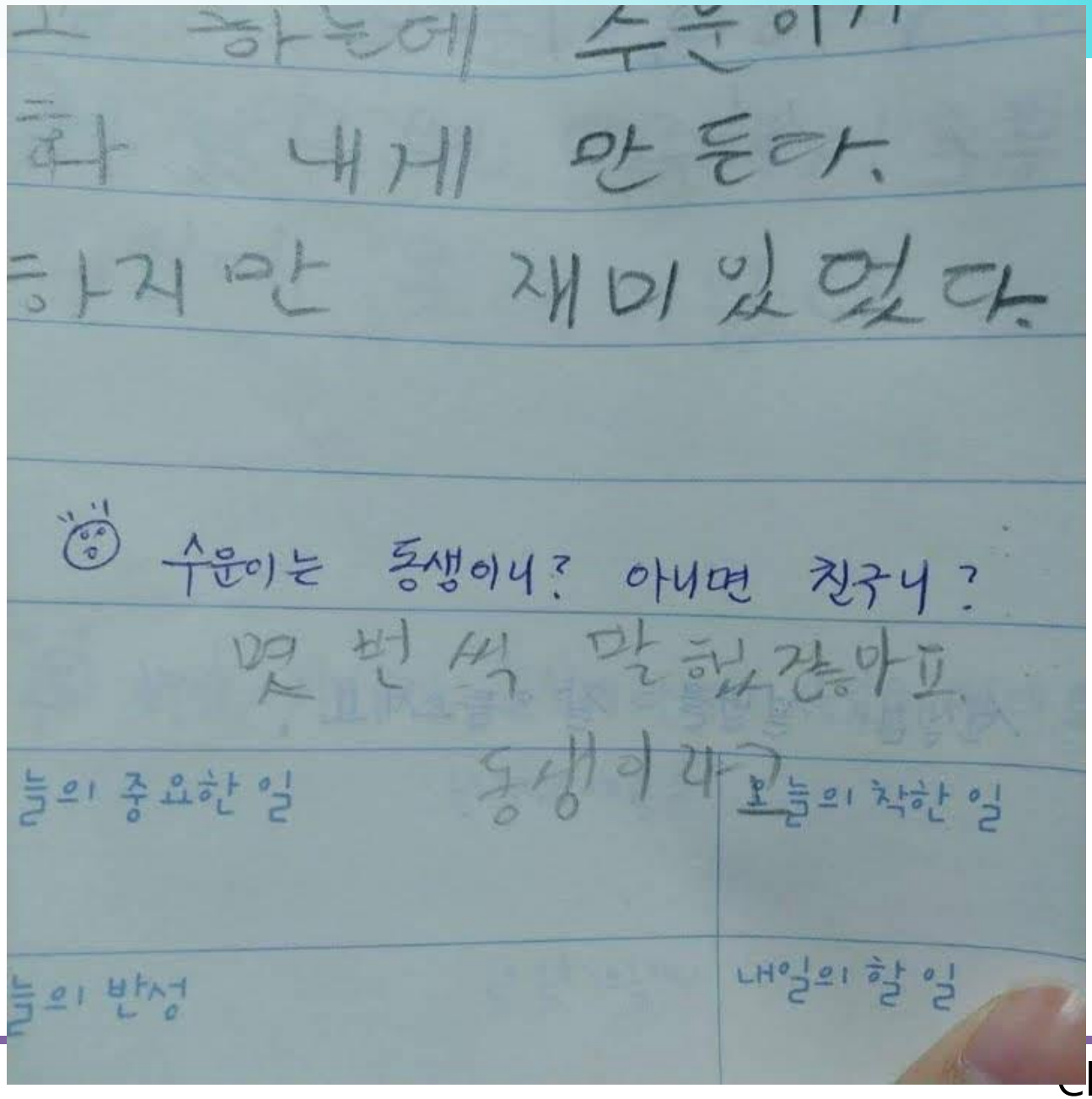
```
> #check coefficients
> coef(logreg3_2)
     (Intercept)                gre                gpa as.factor(rank)2
    -3.989979073       0.002264426       0.804037549      -0.675442928
as.factor(rank)3 as.factor(rank)4
    -1.340203916      -1.551463677
> exp(coef(logreg3_2))
     (Intercept)                gre                gpa as.factor(rank)2
       0.0185001         1.0022670         2.2345448         0.5089310
as.factor(rank)3 as.factor(rank)4
       0.2617923         0.2119375
```

- 분석결과
  - ✓ GRE, GPA, Rank(2, 3, 4)가 모두 유의한 영향력을 가짐
    - 지원자 GRE 점수가 1점 올라갈 경우 합격할 odds가 $e^{0.0023}$배, 즉, 1.0023배 늘어남
    - 지원자 GPA 점수가 1점 올라갈 경우 합격할 odds가 $e^{0.8040}$배, 즉, 2.2345배 늘어남
    - 지원자 학부가 Rank 2에 속할 경우, 합격할 odds가 Rank 1 학부 출신 지원자에 비해 $e^{-0.6754}$배, 즉, 0.5089배로 줄어듦

2. 로지스틱 회귀분석 방법

```
> #model comparison
> anova(logreg3_1, logreg3_2, test = "LRT")
Analysis of Deviance Table

Model 1: admit ~ gre + gpa + rank
Model 2: admit ~ gre + gpa + as.factor(rank)
  Resid. Df Resid. Dev Df Deviance Pr(>Chi)
1       396     459.44
2       394     458.52  2  0.92427   0.6299
```

- 분석결과
  - ✓ 모델 비교결과, 두 모델 간의 차이는 없는 것으로 나타남
    - – Continuous variable로 간주해도 문제 없음


하는데 수운이가
화 내게 만든다.
하지만 재미있었다.
수운이는 동생이니? 아니면 친구니?
몇번씩 말해봐라.
동생이다
늘의 중요한 일
오늘의 착한 일
늘의 반성
내일의 할 일

*ROC (Receiver Operating Characteristic) curve

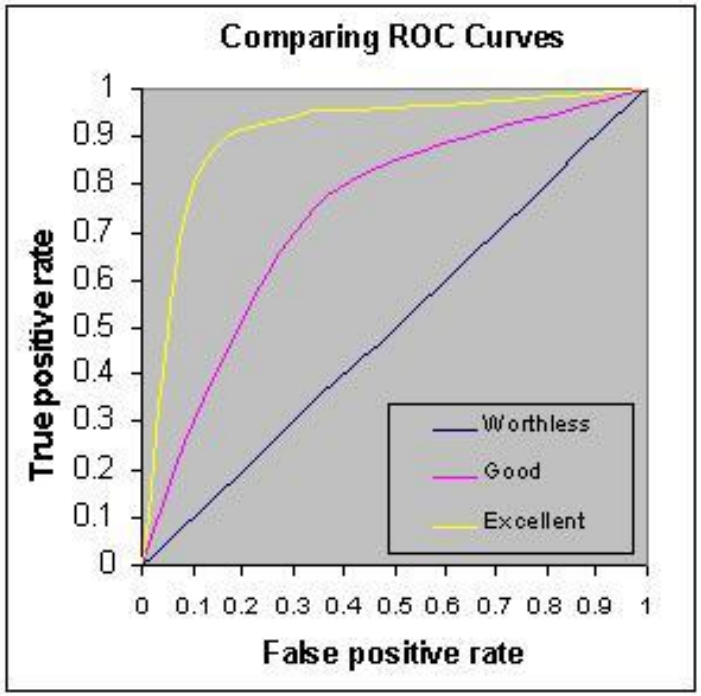
- 이진 분류기(Binary Classifier)의 유용성을 검증하는 방식 중에 널리 사용되는 것은 **ROC 커브**와 **ROC 커브 아래의 면적(the area under the ROC curve; AUC)**
- ROC 커브는 **민감도(Sensitivity; 실제 값이 1일 때 검사 값이 1인 경우)**와 **특이도(Specificity; 실제 값이 0일 때 검사 값이 0인 경우)** 간의 관계를 2차원 상에 표현
- 민감도를 높이기 위해서는 선별 기준을 민감하게 하면 되는 데(조금만 이상해도 검사 값이 1로 나오도록; 예를 들어, 고혈압 기준을 확장기혈압 기준 60mmHg로 설정), 이럴 경우, 특이도가 낮아짐(실제 값이 0인데 검사 값은 1로 나올 가능성 증가)

*ROC (Receiver Operating Characteristic) curve

- 반대로 선별 기준을 까다롭게 하면 (예를 들어, 고혈압 기준을 확장기혈압 기준 100mmHg로 설정) 특이도가 높아지지만, 민감도가 낮아짐(실제 값이 1인데 검사 값은 0으로 나올 가능성 증가)
- ROC 커브의 **x축은 False Positive Rate(음성을 양성으로 착각; 1- 특이도), y축은 True Positive Rate(양성을 양성으로 검출; 민감도)**로 구성
- 커브가 좌상단 코너에 근접할수록 분류기 성능이 좋다고 평가. 분류기가 완벽한 경우 TPR, 즉, 민감도 = 1이고, FPR, 즉, (1-특이도) = 0
- 선별 기준을 변화시켜가면서 FPR, TPR 값을 계산한 것이 ROC 커브. 완전한 랜덤 예측일 경우 (0, 0), (1, 1)을 잇는 대각선이 됨(base line)

*ROC (Receiver Operating Characteristic) curve

- 분류기 정확도는 **ROC 커브 아래 면적(the area under the ROC curve; AUC)**로 측정.
- 면적=1은 완벽한 분류기; 면적 0.5는 랜덤분류와 동일한 분류기(가치 없음).
- 학계에서 전통적으로 사용되는 판단 기준은 아래와 같음
  - ✓ **.90-1 = excellent**
  - ✓ **.80-.90 = good**
  - ✓ **.70-.80 = fair**
  - ✓ .60-.70 = poor
  - ✓ .50-.60 = fail



*Sensitivity, Specificity, Positive Predictive Value, Negative Predictive Value

		CONDITION determined by "Gold Standard"			
	TOTAL POPULATION	CONDITION POS	CONDITION NEG	PREVALENCE = CONDITION POS / TOTAL POPULATION	
TEST OUT-COME	TEST POS	True Pos TP	*Type I Error* False Pos FP	*Precision* Pos Predictive Value PPV = TP / TEST P	False Discovery Rate FDR = FP / TEST P
	TEST NEG	*Type II Error* False Neg FN	True Neg TN	False Omission Rate FOR = FN / TEST N	Neg Predictive Value NPV = TN / TEST N
	ACCURACY ACC ACC = (TP + TN) / TOT POP	Sensitivity (SN), Recall True Pos Rate TPR TPR = TP / CONDITION POS	Fall-Out False Pos Rate FPR FPR = FP / CONDITION NEG	Pos Likelihood Ratio LR + LR + = TPR / FPR	Diagnostic Odds Ratio DOR DOR = LR + / LR -
		Miss Rate False Neg Rate FNR FNR = FN / CONDITION POS	Specificity (SPC) True Neg Rate TNR TNR = TN / CONDITION NEG	Neg Likelihood Ratio LR - LR - = TNR / FNR	

From Wikimedia Commons

***Accuracy : (TP + TN) / (TP + TN + FP + FN)**
****F1-score : 2 * (Precision * Recall) / (Precision + Recall)**

*Example case

Suppose the fecal occult blood (FOB) screen test is used in 2030 people to look for bowel cancer:

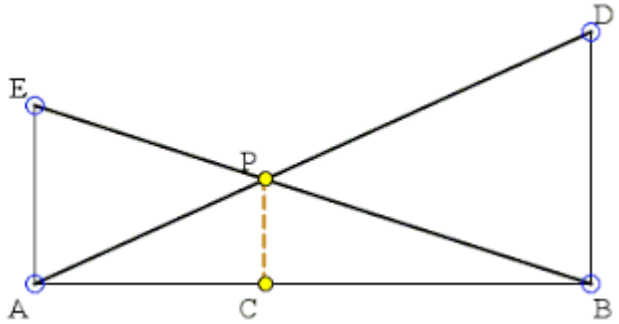
		Patients with bowel cancer (as confirmed on endoscopy)		
		Condition positive	Condition negative	
Fecal occult blood screen test outcome	Test outcome positive	**True positive** (TP) = 20	**False positive** (FP) = 180	(Precision) Positive predictive value = TP / (TP + FP) = 20 / (20 + 180) = **10%**
	Test outcome negative	**False negative** (FN) = 10	**True negative** (TN) = 1820	Negative predictive value = TN / (FN + TN) = 1820 / (10 + 1820) ≈ **99.5%**
		(Recall) Sensitivity = TP / (TP + FN) = 20 / (20 + 10) ≈ **67%**	Specificity = TN / (FP + TN) = 1820 / (180 + 1820) = **91%**	

$$\text{Accuracy} = \frac{(20+1820)}{(20+10+180+1820)} = 90.6\%$$

*F-(beta) score (or F-measure)

1. A variant of accuracy not affected by negatives
2. **Harmonic mean** of Precision and Recall

- $F_\beta = \frac{(\beta^2+1)\cdot P\cdot R}{\beta^2\cdot P+R}$
- β defines relative importance of P and R
- β > 1 means R is more important
- β = 1 is common (F1 score)
- $F_1 = \frac{2PR}{P+R} = 2(\frac{1}{\frac{1}{P}+\frac{1}{R}})$



- Heavily penalizes small values of P and R
- In the previous example case, $F_1 = \frac{2*0.10*0.67}{(0.10+0.67)} = 17.4\%$

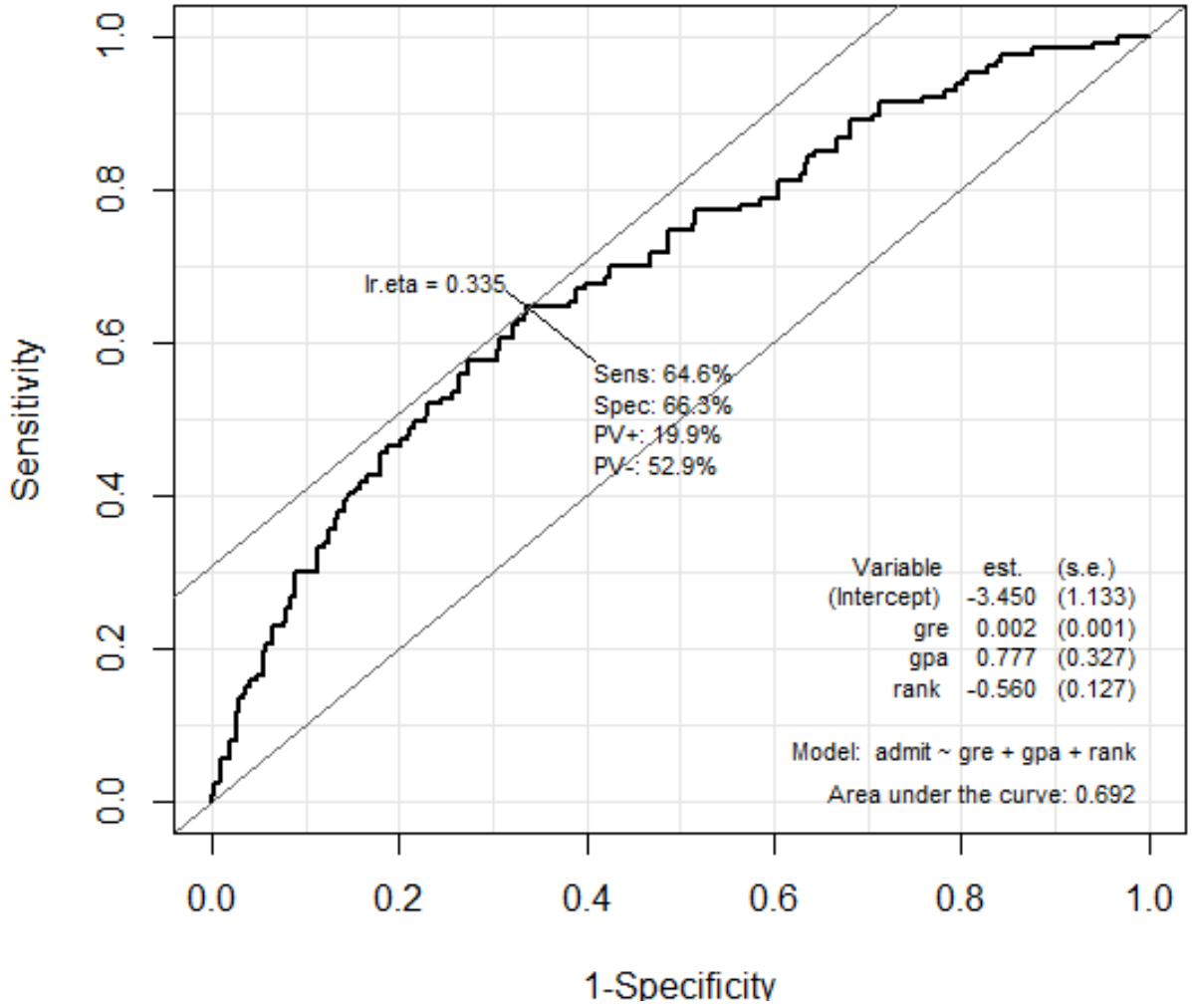
2. 로지스틱 회귀분석 방법

- ROC 커브를 통해 로지스틱 회귀모델의 성능을 평가

```
#ROC 커브로 로지스틱 회귀모델 성능 평가
#install.packages("Epi")
library(Epi)
?ROC
graph1 <- ROC(form = admit ~ gre + gpa + rank, data = admission,
plot = "ROC")
graph1$res # dataframe with variables sens, spec, pvp, pvn, and
lr.eta(probability)
head(graph1$res)
tail(graph1$res)
graph1$res[round(graph1$res$lr.eta,3) == 0.335,]
graph1$AUC # 0.5 ~ 1.0: The bigger, the better
graph1$lr # Logistic regression output
```

- ✓ ROC 커브를 그리기 위해서는 Epi 패키지를 설치/로딩
- ✓ ROC 함수를 통해 ROC 커브를 그리고, graph1에 결과를 저장
- ✓ graph1에 저장된 ROC 커브 개별 data point의 값(res), ROC 커브 아래 면적(AUC), 로지스틱 회귀분석 결과(lr)을 각각 확인 가능

2. 로지스틱 회귀분석 방법



2. 로지스틱 회귀분석 방법

- 로지스틱 회귀분석을 통해 예측한 값(admit 예측치)과 실제 관찰값(admit)을 비교하여 다른 정확도 지표 계산 가능

```
logreg3_1$data  #observed inputs and output (admit gre  gpa rank)
logreg3_1$y #observed output
logreg3_1$linear.predictors  #linear combinations of x (logit or z)
cbind(ifelse(1/(1+exp(-logreg3_1$linear.predictors))>0.335,1,0),
logreg3_1$y)  #compare estimated value with observed value
table(ifelse(1/(1+exp(-logreg3_1$linear.predictors))>0.335,1,0),
logreg3_1$y)  #cross-tabulation to calculate accuracy measures
```

✓ logreg3_1$linear.predictors 값(z)에서 p를 계산

✓ Cut-off value(0.335)를 기준으로 1과 0으로 구분한 뒤, 실제 관찰값 logreg3_1$y와 비교

2. 로지스틱 회귀분석 방법

```
> table(ifelse(1/(1+exp(-logreg3_1$linear.predictors))>0.335,1,0),
logreg3_1$y)

      0   1
  0 181  45
  1  92  82
> sum(logreg3_1$y)
[1] 127
> 82/127  #sensitivity(recall)
[1] 0.6456693
> sum(ifelse(logreg3_1$y==0, 1, 0))
[1] 273
> 181/273  #specificity
[1] 0.6630037
> 82/(92+82)  #precision(positive predictive value)
[1] 0.4712644
> 181/(181+45)   #negative predictive value
[1] 0.800885
> 2*82/174*82/127/(82/174+82/127) #F1 score
[1] 0.5448505
```

.허정민

3일 전

국가의 이름을 변환해주는 countrycode

국가의 코드를 표준화 해주는 countrycode를 소개합니다.

제가 생각하기에 이 라이브러리의 장점은 국가의 다양한 이름을 모두 인식해서 표준화 해준다는 것입니다.

사용방법은

library(countrycode)

countrycode(data, origin = "현재국가명상태",destination = "바꾸고싶은국가명형태")

로 사용하시면 될 것 같습니다.

예를들어 국제 표준화 기구(ISO)에서 발행한 3문자 국가 코드로 변환해 본다면

```
> test<-(c("Korea","korea (Republic of)","South Korea","Republic of Korea","USA","United States","U.S.A"))
> test[1:7]
[1] "Korea"              [2] "korea (Republic of)"[3] "South Korea"        [4] "Republic of Korea"  [5] "USA"                [6] "United States"
  [7] "U.S.A"
> countrycode(test[1:7], origin = "country.name",destination = "iso3c")
[1] "KOR" "KOR" "KOR" "KOR" "USA" "USA" "USA"
```

또한 destination을 continent로 지정한다면

```
> countrycode(test[1:7], origin = "country.name",destination = "continent")
[1] "Asia"     "Asia"     "Asia"     "Asia"     "Americas" "Americas" "Americas"
```

로 출력되는 것을 볼 수 있습니다.

다음링크를 참고하시면 더 다양한 변환방식을 참고 할 수 있습니다

https://cran.r-project.org/web/packages/countrycode/countrycode.pdf

.최민기 16시간 전

용어정리

맑은날을 0, 흐린 날을 1이라고 가정합니다. 하늘이 흐리다고 예측한다면 h(x)=1이 되는 것이고, 맑다고 예측한다면 h(x)=0이 되는 것입니다.

1) TP(True Positive) - 민감도(recall)

TP는 실제로 정답이 1인데, 우리가 1로 예측한 경우입니다. 하늘이 흐리다고 예측 하였고 실제로 하늘이 흐린 경우 입니다.

2) TN(True Negative) - 특이도(specificity)

TN는 실제로 정답이 0인데, 우리가 0으로 잘 예측한 경우입니다. 하늘이 맑다고 예측하고 실제로도 하늘이 맑을 경우입니다. (TN이 높다면 흐린날도 아니고 비오는 날도 아닌 맑은 날을 잘 골라냅니다)

3) FP(False Positive) - Type1 error

FP는 정답이 아닌 경우(0)를 우리가 정답(1)으로 예측한 경우입니다. 에러에 해당합니다. 예를 들어, 하늘이 맑은지(0) / 흐린지(1)를 판단하는 모델을 생각해봅시다. FP는 실제로 하늘이 맑은데 흐리다고 판단했다면 증가하는 에러입니다.

4) FN(False Negative) - Type2 error

FN는 정답인 경우(1)을 우리가 정답이 아닌 것으로(0)로 예측한 경우입니다. 이것도 또한 에러에 해당합니다. 위와 같은 날씨가 어떤지 판단하는 모델에서 실제로 하늘이 흐린데 맑다고 예측한 경우입니다.

Precision(정밀도) 정밀도란 모델이 1이라고 예측한 것 중에서 실제 1인 것의 비율입니다.

실제로 1인경우 / 1이라고 예측한 경우

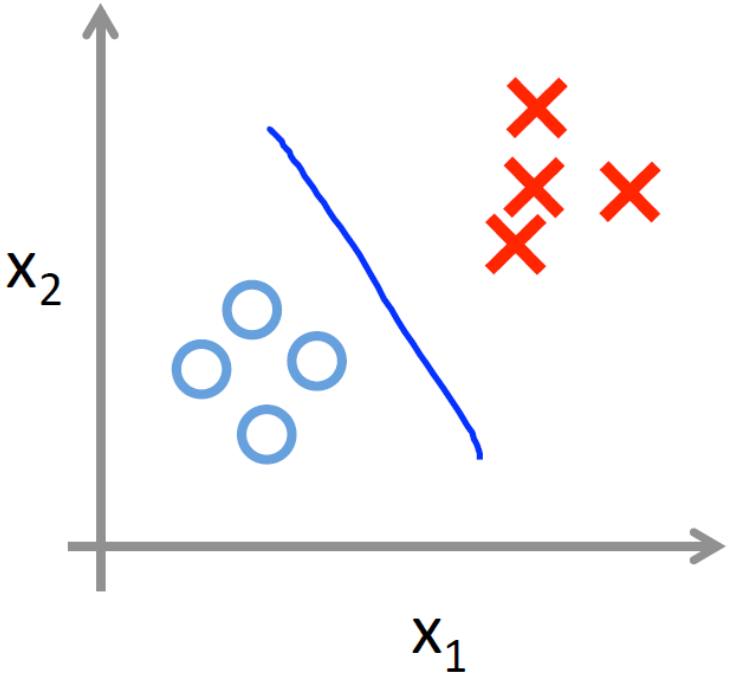
Recall 실제 1인 것 중에서 모델이 1이라고 예측한 것의 비율입니다.

1이라고 예측한 경우 / 실제로 1인경우

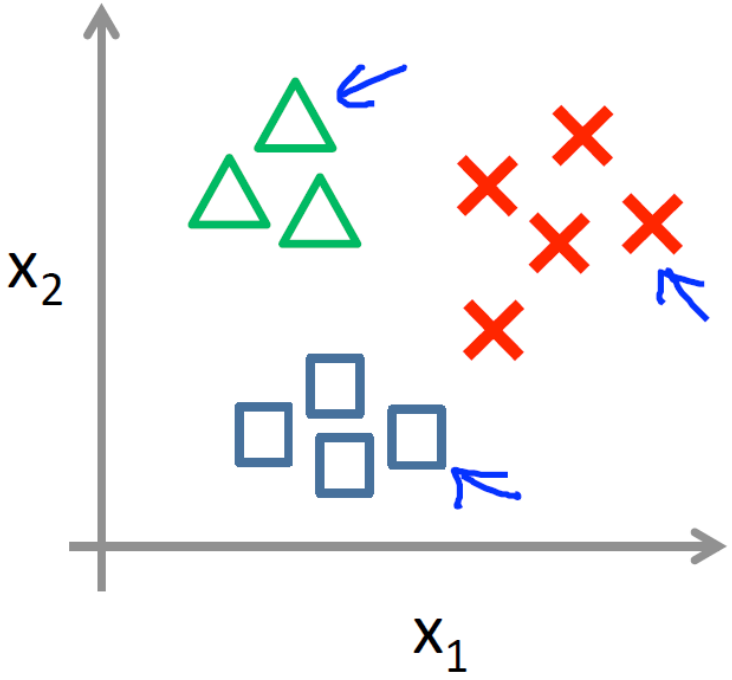
댓글 인용 수정 삭제 이메일

*Multiclass-classification(다중분류)

Binary classification:



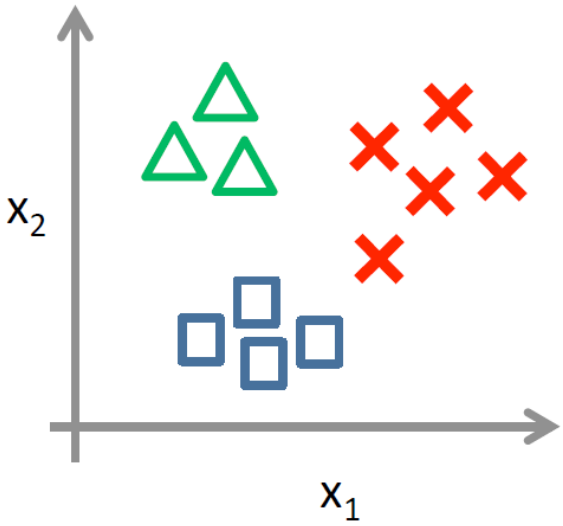
Multi-class classification:



Andrew Ng

*One-vs-all (one-vs-rest)

One-vs-all (one-vs-rest):

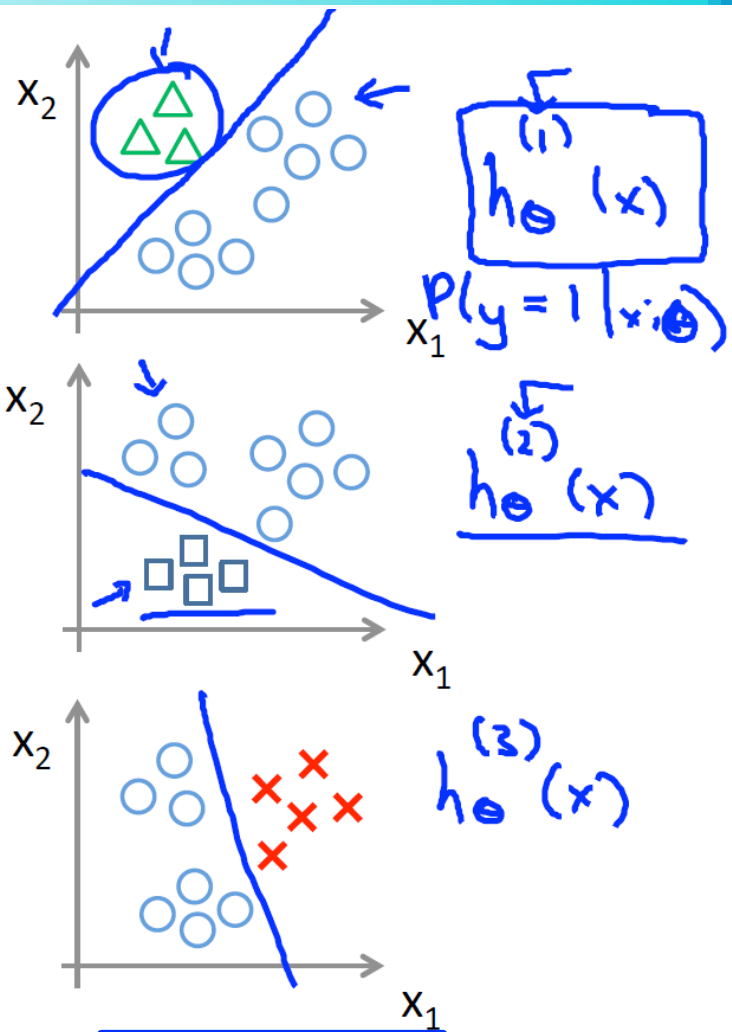


Class 1: △ ←

Class 2: □ ←

Class 3: ✕ ←

$$h_\theta^{(i)}(x) = P(y = i|x;\theta) \qquad (i = 1, 2, 3)$$



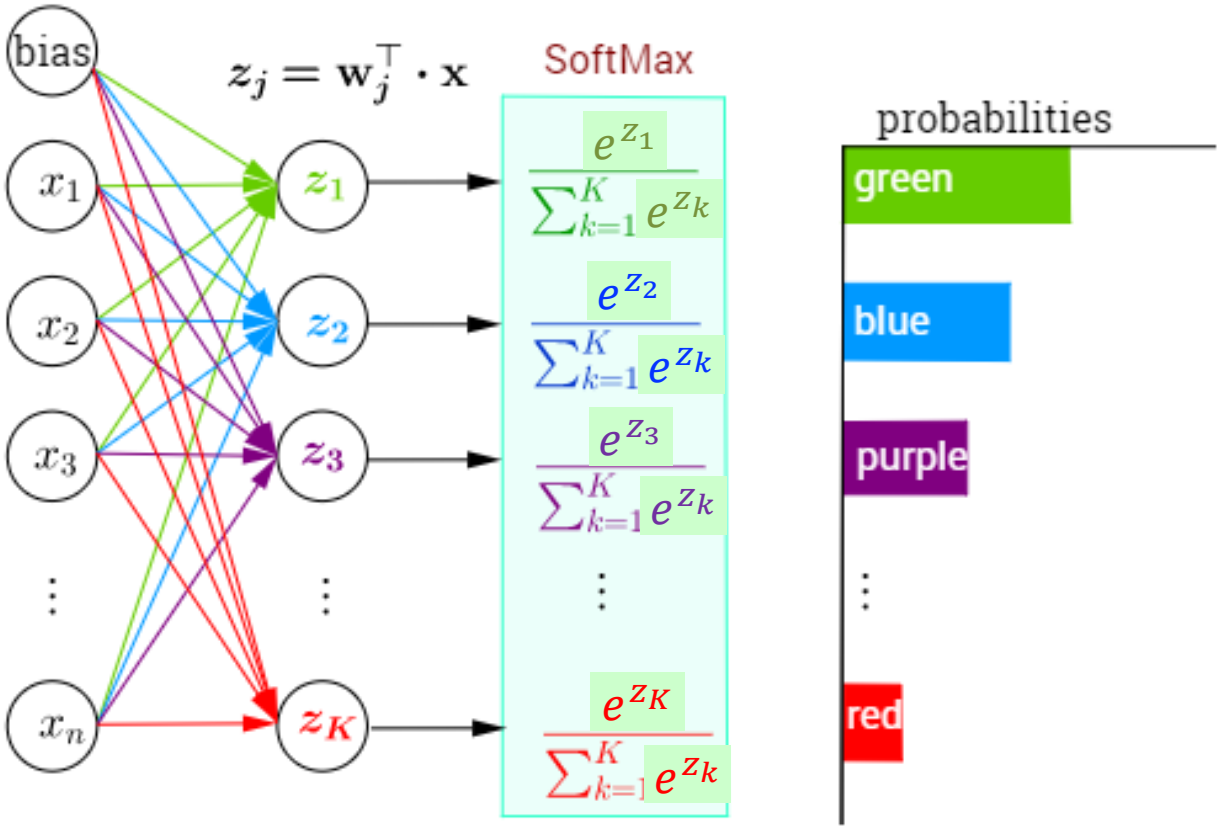
$$\max_i h_\theta^{(i)}(x)$$

Andrew Ng

*SoftMax function

Multi-Class Classification with NN and SoftMax Function

$$\mathbf{z} = \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ \vdots \\ z_K \end{bmatrix} = \begin{bmatrix} \mathbf{w}_1^\top \\ \mathbf{w}_2^\top \\ \mathbf{w}_3^\top \\ \vdots \\ \mathbf{w}_K^\top \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{bmatrix}$$



Hotdog identifying app ☺

3. 선형판별분석

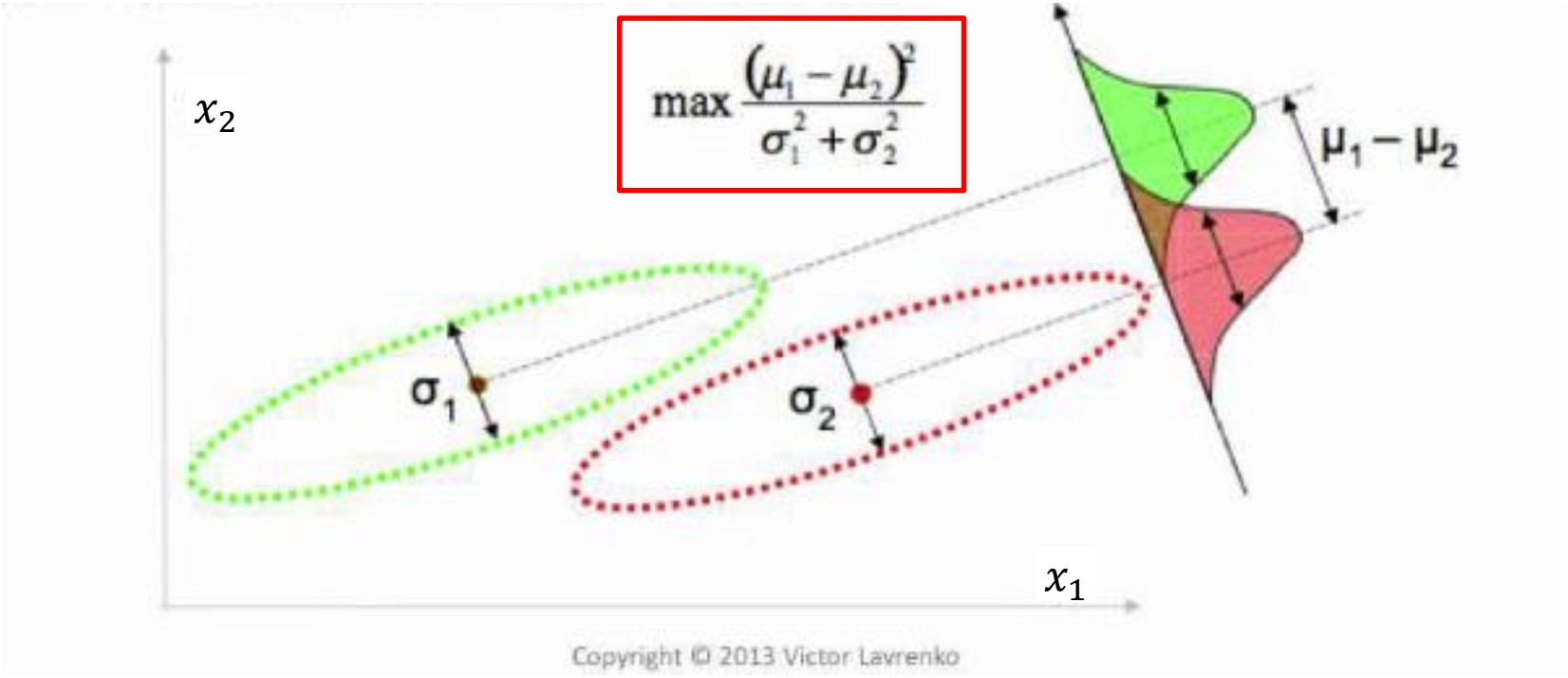
1. 소개

- 주어진 독립변수를 이용해 (명목척도)**종속변수**를 예측하는 방법 (효과적 분류를 위한 차원 축소 방법으로도 사용됨)
- 로지스틱 회귀분석과 달리,
  - ✓ 각 집단 내에서 **독립변수들이 다변량 정규분포를** 이루고
  - ✓ 각 집단별로 **독립변수들의 분산/공분산이 동일하다는** 가정이 존재
- 선형판별분석이 선호되는 경우
  (An Introduction to Statistical Learning, Ch. 4.4)
  - ✓ 집단의 경계가 명확할 때(클래스들이 잘 분리될 때)
  - ✓ 샘플 수가 작고, 각 집단에서 설명변수의 분포가 정규분포를 따를 때
  - ✓ 집단의 수가 3 이상일 때

3. 선형판별분석

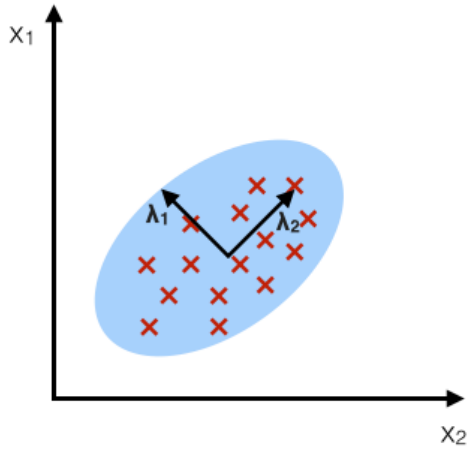
2. 기본 원리 (차원 발굴/축소 관점)

- 집단 간 평균 차이를 극대화 하고,
- 집단 내 분산을 최소화하는 새로운 차원을 발굴

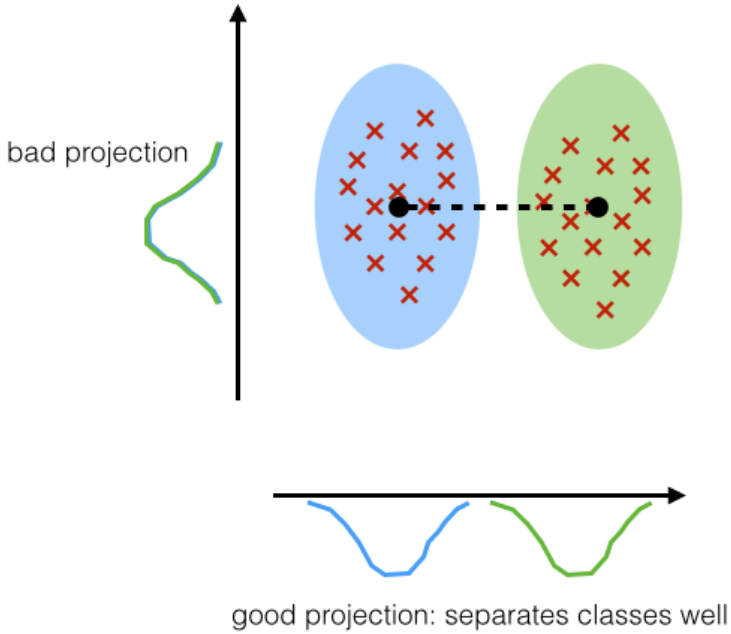


*PCA vs. LDA

PCA:
component axes that maximize the variance

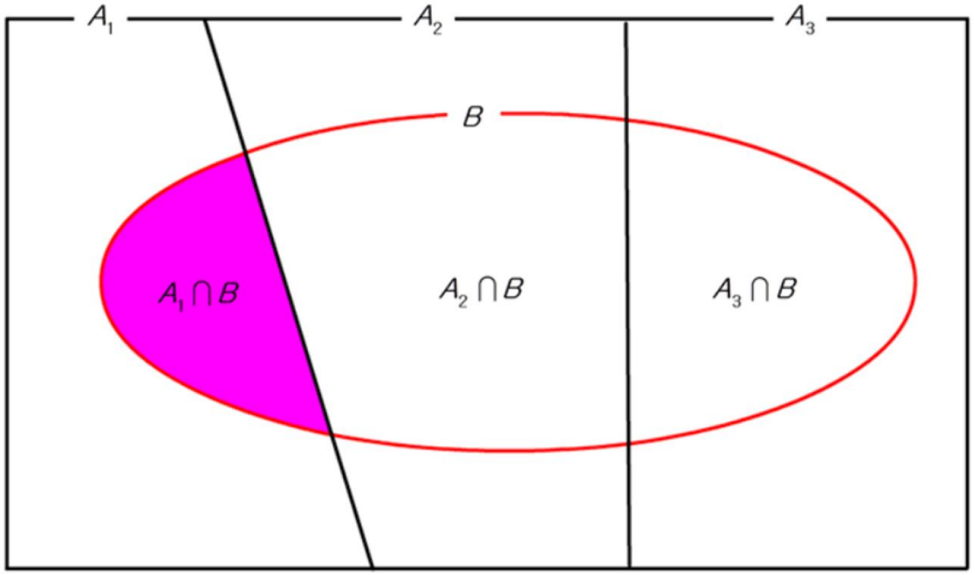


LDA:
maximizing the component axes for class-separation



https://sebastianraschka.com/Articles/2014_python_lda.html

*베이즈 정리(Bayes Theorem)



$$P(A_k|B)$$

$$= \frac{P(A_k \cap B)}{P(B)}$$

$$= \frac{P(A_k \cap B)}{P(A_1 \cap B) + P(A_2 \cap B) + \cdots + P(A_n \cap B)}$$

$$= \frac{P(B|A_k)P(A_k)}{P(B|A_1)P(A_1) + P(B|A_2)P(A_2) + \cdots + P(B|A_n)P(A_n)}$$

(단 k가 $k = 1,2,3, \cdots, n$)

3. 기본 원리 (베이즈 정리를 활용)

사전확률 확률밀도함수(k 집단의)

① $$\Pr(Y = k|X = x) = \frac{\pi_k f_k(x)}{\sum_{l=1}^{K} \pi_l f_l(x)}$$

사후확률

각 집단의 사전확률*확률밀도함수의 합

각 집단 내 x의 분포를 정규분포로 가정하면,

② $$f_k(x) = \frac{1}{\sqrt{2\pi}\sigma_k} e^{-\frac{1}{2}\left(\frac{x-\mu_k}{\sigma_k}\right)^2}$$

집단 별 분산이 동일하다고 가정하고 ①과 ②를 결합하면,

③ $$p_k(x) = \Pr(Y = k|X = x) = \frac{\pi_k \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2}\left(\frac{x-\mu_k}{\sigma}\right)^2}}{\sum_{l=1}^{K} \pi_l \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2}\left(\frac{x-\mu_l}{\sigma}\right)^2}}$$

3. 선형판별분석

주어진 x값을 기반으로 집단을 분류하기 위해서는,
p_k(x)가 최대값을 갖는 k를 찾아야 함

③에 로그를 취하고 k에 관련 없는 항들을 제거하면,
이 값을 최대로 하는 k가 결국 p_k(x)를 최대로 만듦

④ $$\delta_k(x) = x \cdot \frac{\mu_k}{\sigma^2} - \frac{\mu_k^2}{2\sigma^2} + \log(\pi_k)$$

판별함수

*Naïve Bayes Classifier

1) 입력 데이터의 특성

$X = \langle X_1, X_2, \cdots, X_n \rangle$, X_i : 명목 혹은 연속 척도 , Y: 명목 척도

2) Naive Bayes classifier

$$P(Y = y_k | X_1 \ldots X_n) \overset{\text{Bayes}}{=} \frac{P(Y = y_k) P(X_1 \ldots X_n | Y = y_k)}{\sum_j P(Y = y_j) P(X_1 \ldots X_n | Y = y_j)}$$

$$\overset{\text{Naive Bayes}}{=} \frac{P(Y = y_k) \prod_i P(X_i | Y = y_k)}{\sum_j P(Y = y_j) \prod_i P(X_i | Y = y_j)}$$

조건부 독립, 즉 서로 다른 X_i와 X_j는 Y가 주어졌을 때 서로 독립

따라서, 위의 확률을 최대로 만드는 y_k가 Y값이라고 추정하는 것이 논리적

$$Y \leftarrow \arg\max_{y_k} \frac{P(Y = y_k) \prod_i P(X_i | Y = y_k)}{\sum_j P(Y = y_j) \prod_i P(X_i | Y = y_j)}$$

분모값은 y_k에 상관없이 일정하기 때문에 고려할 필요가 없음

$$Y \leftarrow \arg\max_{y_k} P(Y = y_k) \prod_i P(X_i | Y = y_k)$$

*설명변수 많을 때 사용.
독립변수 간 독립 가정은 PCA로 구현 가능.

4. 선형판별분석 방법

1. 당뇨환자 예측 by 로지스틱 회귀분석

- National Institute of Diabetes and Digestive and Kidney Diseases에서 수집된 21세 이상의 Pima 인디언 여성 768명의 데이터
- 임신횟수, 혈당, 혈압, 피부두께, 인슐린수치, BMI, 당뇨 가족력, 나이, 환자여부(268명이 환자)
- 자료 확인

```
data <- read.csv("diabetes.csv")
names(data)
data$Outcome <- factor(data$Outcome, levels = c(0, 1), labels =
c("False", "True"))  #factor type으로 재정의
str(data)
head(data)
library(skimr)
skim(data)
```

4. 선형판별분석 방법

```
> skim(data)
Skim summary statistics
 n obs: 768
 n variables: 9

-- Variable type:factor ---------------------------------------------------
 variable missing complete   n n_unique                top_counts ordered
  Outcome       0      768 768        2 Fal: 500, Tru: 268, NA: 0   FALSE

-- Variable type:integer --------------------------------------------------
      variable missing complete   n   mean     sd p0 p25   p50    p75 p100     hist
           Age       0      768 768  33.24  11.76 21  24  29     41      81 ▇▃▁▁▁▁▁▁
 BloodPressure       0      768 768  69.11  19.36  0  62  72     80     122 ▁▁▁▂▇▃▁▁
       Glucose       0      768 768 120.89  31.97  0  99 117 140.25     199 ▁▁▁▃▇▅▂▁
       Insulin       0      768 768  79.8  115.24  0   0  30.5 127.25  846 ▇▂▁▁▁▁▁▁
   Pregnancies       0      768 768   3.85   3.37  0   1   3      6      17 ▇▃▂▁▁▁▁▁
 SkinThickness       0      768 768  20.54  15.95  0   0  23     32      99 ▇▆▇▁▁▁▁▁

-- Variable type:numeric --------------------------------------------------
                 variable missing complete   n  mean   sd    p0   p25   p50   p75  p
                      BMI       0      768 768 31.99 7.88 0     27.3  32    36.6  67
 DiabetesPedigreeFunction       0      768 768  0.47 0.33 0.078  0.24  0.37  0.63  2
```

- ✓ Outcome은 factor type 변수이며, False 500명, True 268명 존재
- ✓ 최소값이 0인 변수가 6개 존재 → 확인 필요

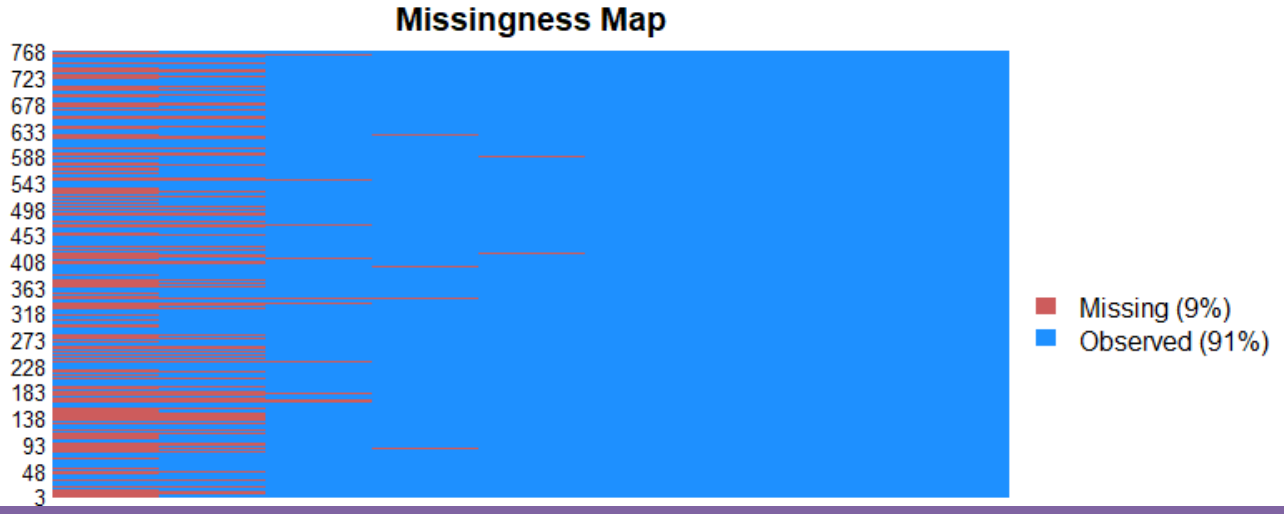
4. 선형판별분석 방법

- 자료 정제(NA 확인)

```
names(data)
data[, 2:6][data[, 2:6] == 0] <- NA
#check how many missing values we have now
install.packages('Amelia')
library(Amelia)
missmap(data)
```

✓ 혈당, 혈압, 피부두께, 인슐린수치, BMI 값이 0인 것은 NA로 해석하는 것이 타당

✓ 'Amelia' 패키지의 missmap을 활용하여 결측치 분포를 확인



4. 선형판별분석 방법

- 결측값 대체

```
#install.packages('mice')
library(mice)
mice_mod <- mice(data[, c("Glucose","BloodPressure",
                          "SkinThickness","Insulin","BMI")], method='rf')
mice_mod
mice_complete <- complete(mice_mod)
#Transfer the predicted missing values into the main data set
data$Glucose <- mice_complete$Glucose
data$BloodPressure <- mice_complete$BloodPressure
data$SkinThickness <- mice_complete$SkinThickness
data$Insulin<- mice_complete$Insulin
data$BMI <- mice_complete$BMI
missmap(data)
```

✓ 결측값을 대체하기 위해 'mice' 패키지의 mice 함수를 사용
  – 방법은 random forest 외에도 다양 (F1눌러 확인 가능)

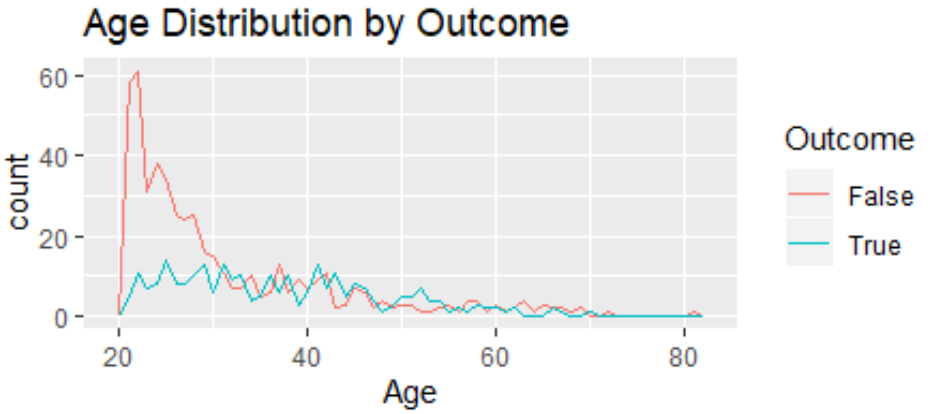
✓ complete 함수로 결측값 대체된 데이터셋 추출

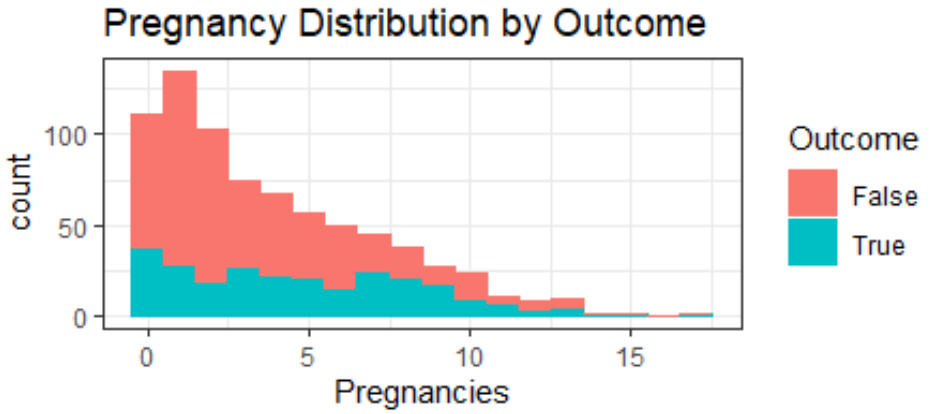
✓ missmap으로 결측값 제거된 것을 확인

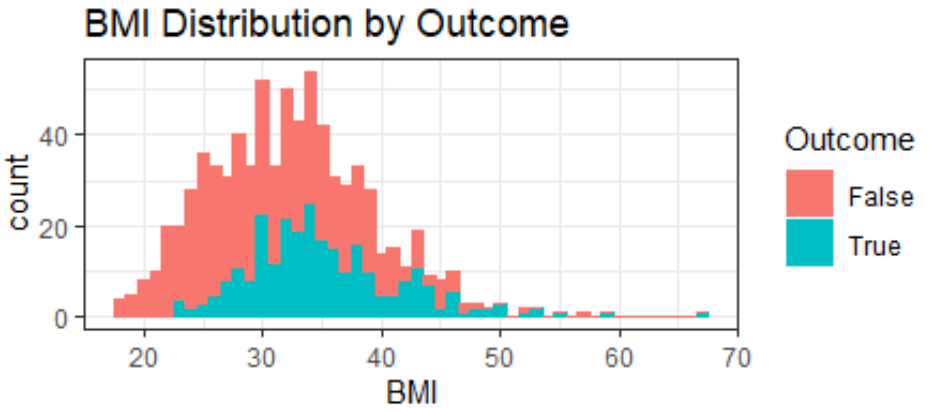
4. 선형판별분석 방법

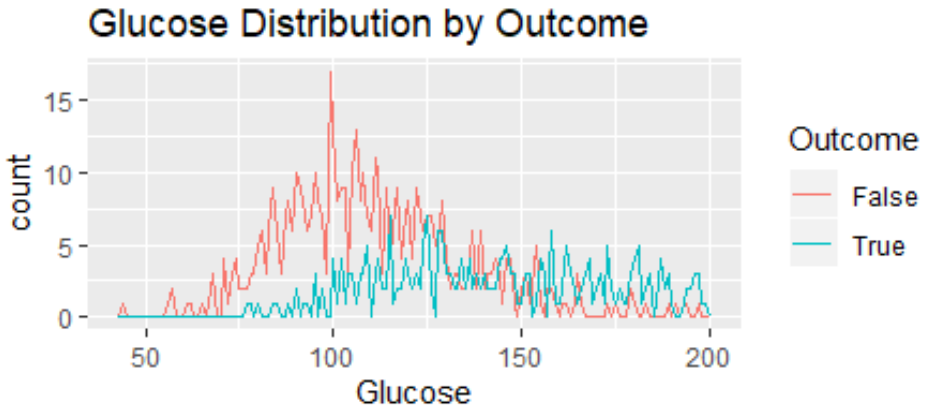
- 시각화를 통해 탐색적 데이터 분석
  - ✓ 당뇨환자 여부에 따른 변수 분포 확인 (관련성 확인)
  - ✓ 변수 간 상관관계 확인

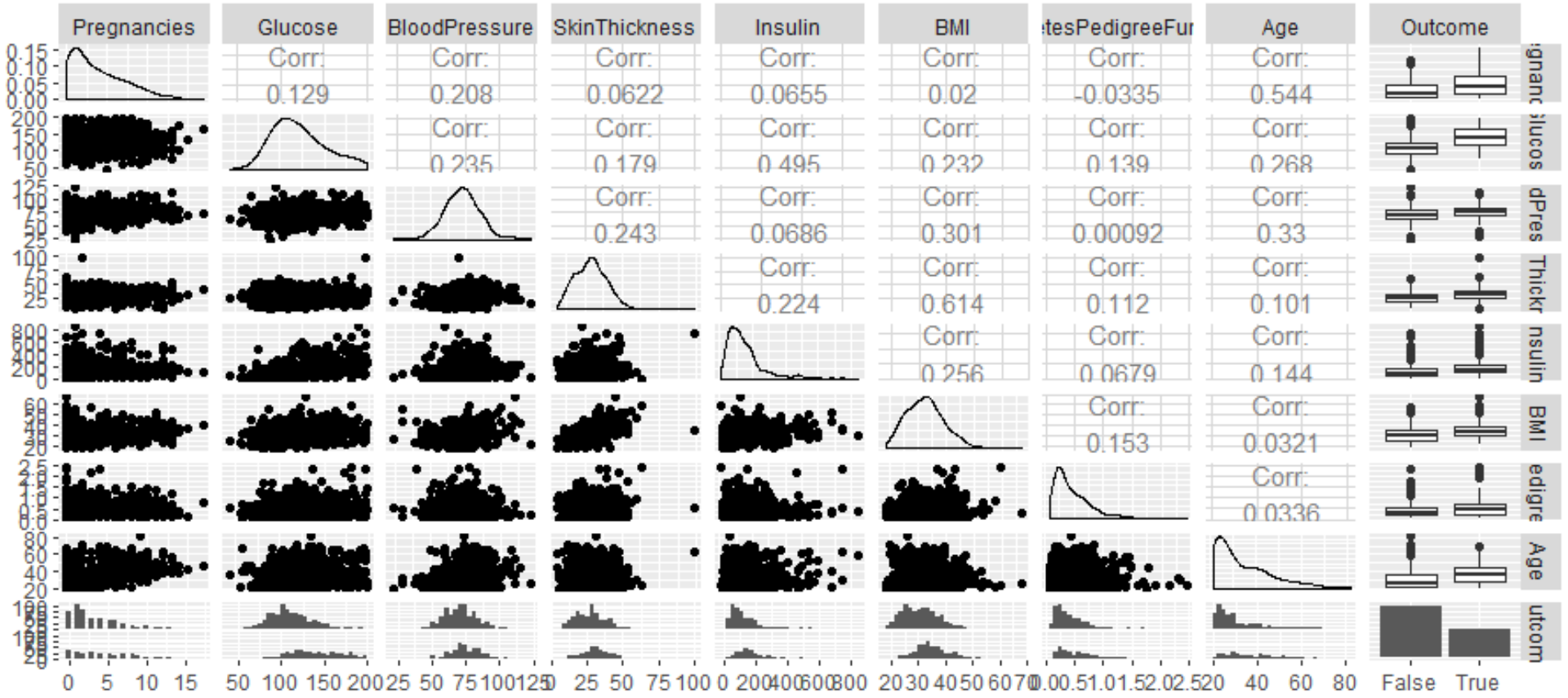
```
library(ggplot2)
ggplot(data, aes(Age, colour = Outcome)) +
  geom_freqpoly(binwidth = 1) + labs(title="Age Distribution by
Outcome")
ggplot(data, aes(x=Pregnancies, fill=Outcome, color=Outcome)) +
  geom_histogram(binwidth = 1) + labs(title="Pregnancy
Distribution by Outcome") + theme_bw()
ggplot(data, aes(x=BMI, fill=Outcome, color=Outcome)) +
  geom_histogram(binwidth = 1) + labs(title="BMI Distribution by
Outcome") + theme_bw()
ggplot(data, aes(Glucose, colour = Outcome)) +
  geom_freqpoly(binwidth = 1) + labs(title="Glucose Distribution
by Outcome")
library(GGally)
ggpairs(data)
```


Age Distribution by Outcome
count
Age
Outcome
False
True


Pregnancy Distribution by Outcome
count
Pregnancies
Outcome
False
True


BMI Distribution by Outcome
count
BMI
Outcome
False
True


Glucose Distribution by Outcome
count
Glucose
Outcome
False
True


Pregnancies
Glucose
BloodPressure
SkinThickness
Insulin
BMI
tesPedigreeFur
Age
Outcome
Corr:
0.129
Corr:
0.208
Corr:
0.0622
Corr:
0.0655
Corr:
0.02
Corr:
-0.0335
Corr:
0.544
Corr:
0.235
Corr:
0.179
Corr:
0.495
Corr:
0.232
Corr:
0.139
Corr:
0.268
Corr:
0.243
Corr:
0.0686
Corr:
0.301
Corr:
0.00092
Corr:
0.33
Corr:
0.224
Corr:
0.614
Corr:
0.112
Corr:
0.101
Corr:
0.256
Corr:
0.0679
Corr:
0.144
Corr:
0.153
Corr:
0.0321
Corr:
0.0336
False
True

4. 선형판별분석 방법

- 훈련용 데이터셋과 검증용 데이터셋 구성
  - ✓ 훈련용 : 검증용 = 3 : 1 사용

```
set.seed(123)
indxTrain <- sample(1:nrow(data), 0.75*nrow(data))
str(indxTrain)
training <- data[indxTrain,]
testing <- data[-indxTrain,]
x = training[,-9]
y = training$Outcome
str(x)
str(y)
```

```
> str(x)
'data.frame':    576 obs. of  8 variables:
 $ Pregnancies             : int  2 7 5 1 6 3 11 1 3 7 ...
 $ Glucose                 : int  109 102 130 128 119 158 143 89 122 114 ...
 $ BloodPressure           : int  92 74 82 48 50 70 94 24 78 76 ...
 $ SkinThickness           : int  33 40 26 45 22 30 33 19 13 17 ...
 $ Insulin                 : int  170 105 478 194 176 328 146 25 126 110 ...
 $ BMI                     : num  42.7 37.2 39.1 40.5 27.1 35.5 36.6 27.8 ...
 $ DiabetesPedigreeFunction: num  0.845 0.204 0.956 0.613 1.318 ...
 $ Age                     : int  54 45 37 24 33 35 51 21 40 31 ...
> str(y)
 Factor w/ 2 levels "False","True": 1 1 2 2 2 2 2 1 1 1 ...
```

.한상혁

2일 전

댓글: 용어정리

참고로 caret 패키지의 confusionMatrix 함수를 이용하여 로지스틱 회귀분석으로 만든 모델의 Sensitivity, Specificity, Precision, Recall, Accuracy 등의 정보를 구할 수 있습니다.

```
library(caret)
library(magrittr)

iris2 <- iris
iris2$Species %<>% as.character
iris2[iris2$Species == "versicolor" | iris2$Species == "virginica", "Species"] <- "Not setosa"
iris2$Species %<>% as.factor

m <- glm(Species ~ ., family="binomial", data=iris2)
p <- predict(m, iris2)
p <- ifelse(p<0.5, "Not setosa", "setosa")
p <- as.factor(p)
cm <- confusionMatrix(p, iris2$Species, positive="setosa")

cm$byClass["Sensitivity"] # Sensitivity
cm$byClass["Specificity"] # Specificity
cm$byClass["Precision"] # Precision
cm$byClass["Recall"]    #    Recall
(cm$table[1,1] + cm$table[2,2]) / sum(cm$table) # Accuracy
```

.최민기

댓글: 용어정리

Accuracy는 overall에서도 구할 수 있습니다~~~

```
cm$overall["Accuracy"]
```

4. 선형판별분석 방법

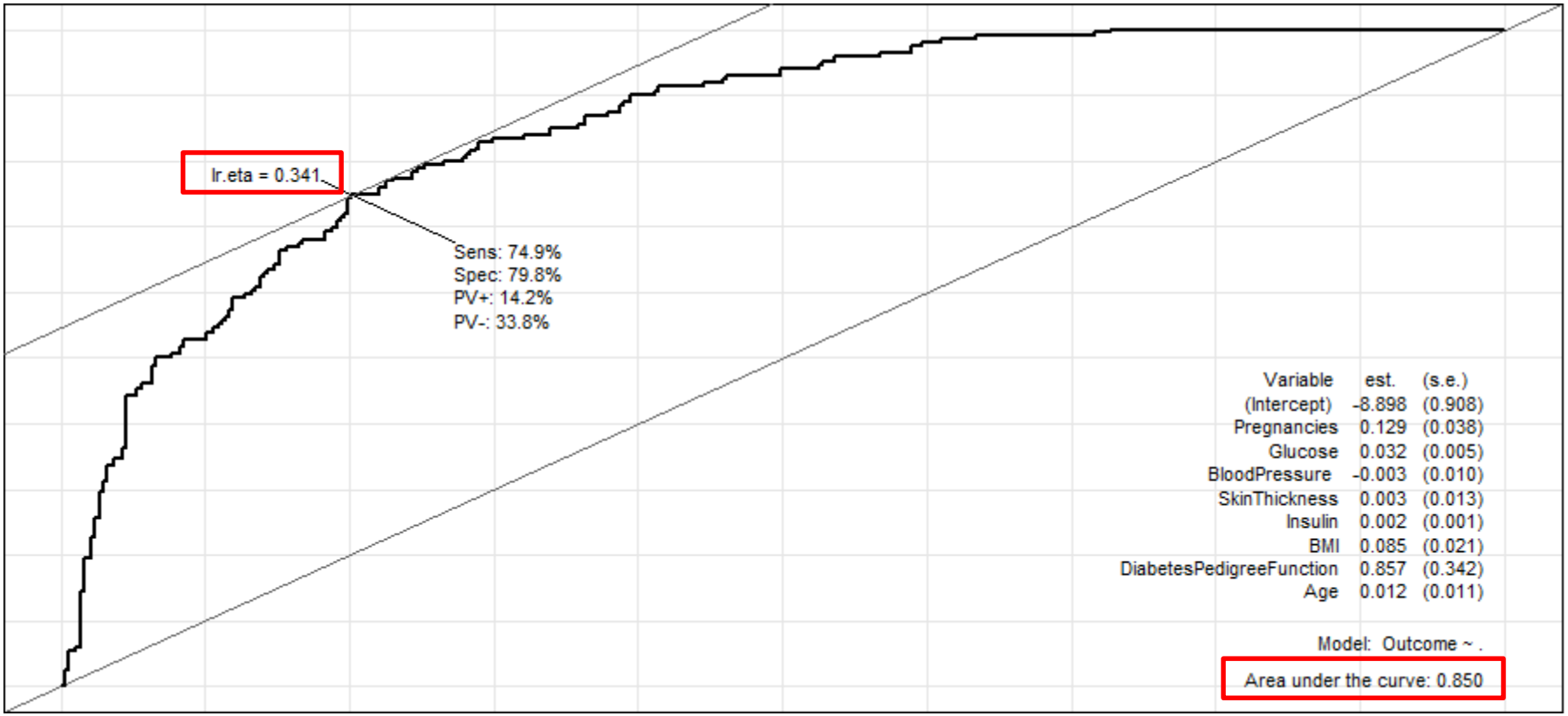
- 로지스틱 회귀분석
  - ✓ 훈련용 데이터셋(training)으로 학습하고, 검증용 데이터셋(testing)으로 예측
  - ✓ 검증 성능평가는 Accuracy와 ROC 커브 활용

```
model_logreg <- glm(Outcome ~ ., data = training, family = binomial)
summary(model_logreg)
pred_logreg <- predict(model_logreg, testing[,-9], type = "response" )
pred_logreg_class <- ifelse(pred_logreg > 0.5, 1, 0)
#you can use lr.eta from ROC curve
pred_logreg_class
tab <- table(pred_logreg_class, testing$Outcome)
tab
sum(tab[row(tab)==col(tab)])/sum(tab)

library(Epi)
ROC_logreg <- ROC(form = Outcome ~ ., data = training, plot = "ROC")
ROC_logreg$res[round(ROC_logreg$res$lr.eta,3) == 0.341,]
pred_logreg_class2 <- ifelse(pred_logreg > 0.341, 1, 0)
tab2 <- table(pred_logreg_class2, testing$Outcome)
tab2
sum(tab2[row(tab2)==col(tab2)])/sum(tab2)
```

4. 선형판별분석 방법

```
> summary(model_logreg)

Call:
glm(formula = Outcome ~ ., family = binomial, data = training)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-3.1252  -0.7041  -0.4027   0.6798   2.2176

Coefficients:
                          Estimate Std. Error z value Pr(>|z|)
(Intercept)              -8.898493   0.908087  -9.799  < 2e-16 ***
Pregnancies               0.129245   0.037685   3.430 0.000605 ***
Glucose                   0.031678   0.004540   6.977 3.01e-12 ***
BloodPressure            -0.003016   0.009973  -0.302 0.762309
SkinThickness             0.003094   0.013154   0.235 0.814043
Insulin                   0.002198   0.001056   2.082 0.037349 *
BMI                       0.084711   0.020996   4.035 5.47e-05 ***
DiabetesPedigreeFunction  0.857452   0.341938   2.508 0.012155 *
Age                       0.011819   0.011063   1.068 0.285377
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 742.59  on 575  degrees of freedom
Residual deviance: 530.96  on 567  degrees of freedom
AIC: 548.96

Number of Fisher Scoring iterations: 5
```

4. 선형판별분석 방법


lr.eta = 0.341
Sens: 74.9%
Spec: 79.8%
PV+: 14.2%
PV-: 33.8%
Variable est. (s.e.)
(Intercept) -8.898 (0.908)
Pregnancies 0.129 (0.038)
Glucose 0.032 (0.005)
BloodPressure -0.003 (0.010)
SkinThickness 0.003 (0.013)
Insulin 0.002 (0.001)
BMI 0.085 (0.021)
DiabetesPedigreeFunction 0.857 (0.342)
Age 0.012 (0.011)
Model: Outcome ~ .
Area under the curve: 0.850

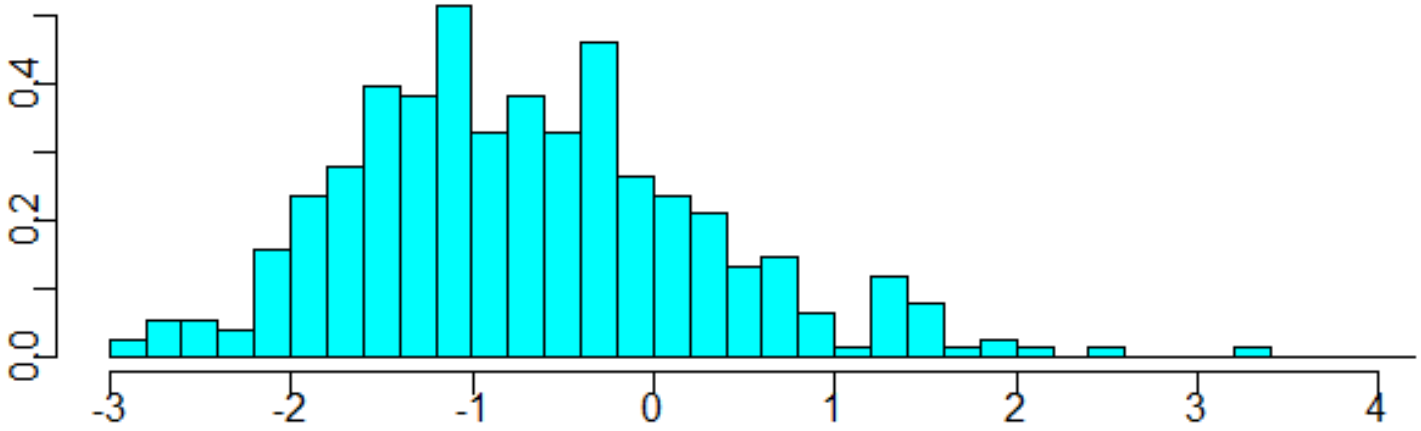
4. 선형판별분석 방법

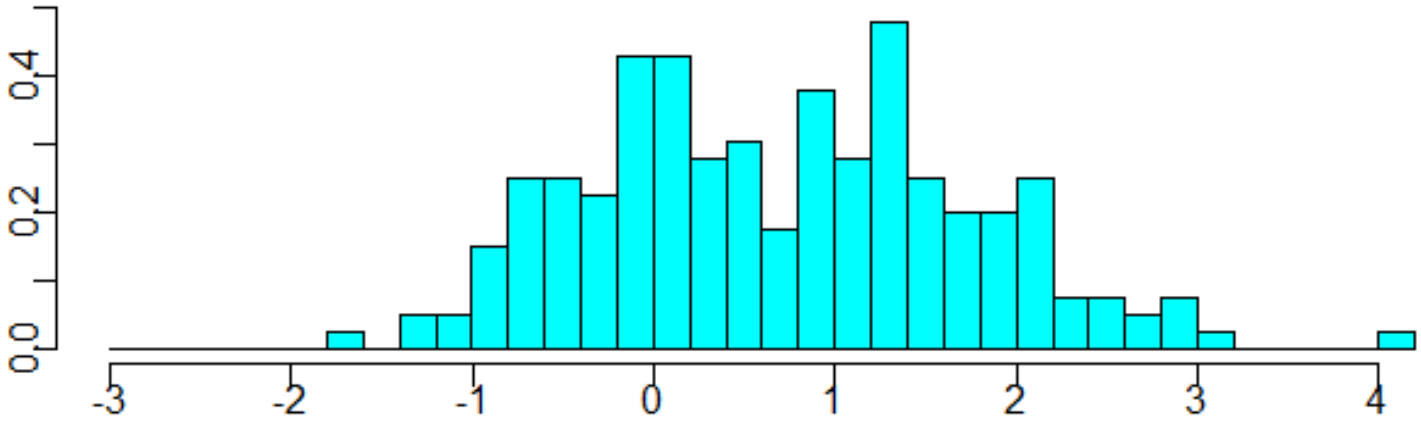
- 선형판별분석
  - ✓ 훈련용 데이터셋(training)으로 학습하고, 검증용 데이터셋(testing)으로 예측
  - ✓ 집단구분, 사후확률, 판별점수 등 확인
  - ✓ Accuracy로 예측 성능 평가

```
library(MASS)
model_lda <- lda(Outcome ~ ., data = training)
model_lda
plot(model_lda)
pred_lda <- predict(model_lda, testing[,-9])
# names(pred_lda)
tail(pred_lda$class)      #집단구분
tail(pred_lda$posterior)  #사후확률
tail(pred_lda$x)          #판별점수
par(mfrow = c(2,1))
hist(pred_lda$x[as.integer(pred_lda$class)==2], breaks = 20, col =
'cyan', xlim = range(pred_lda$x))
hist(pred_lda$x[as.integer(pred_lda$class)==1], breaks = 20, col =
'cyan', xlim = range(pred_lda$x))
tab <- table(pred_lda$class, testing$Outcome)
tab
sum(tab[row(tab)==col(tab)])/sum(tab)
```

4. 선형판별분석 방법

```
> model_lda
Call:
lda(Outcome ~ ., data = training)

Prior probabilities of groups:
    False      True
0.6545139 0.3454861

Group means:
      Pregnancies  Glucose BloodPressure SkinThickness  Insulin      BMI
False    3.238727 109.1432      70.51989      26.81698 119.1141 30.96127
True     4.894472 139.8794      75.63819      32.75879 201.6432 35.58543

Coefficients of linear discriminants:
                                  LD1
Pregnancies                0.0967432539
Glucose                    0.0253458484
BloodPressure             -0.0005714212
SkinThickness              0.0003177088
Insulin                    0.0015852295
BMI                        0.0598701847
DiabetesPedigreeFunction   0.5671007261
Age                        0.0080124872
```


0,0
0,2
0,4
-3
-2
-1
0
1
2
3
4

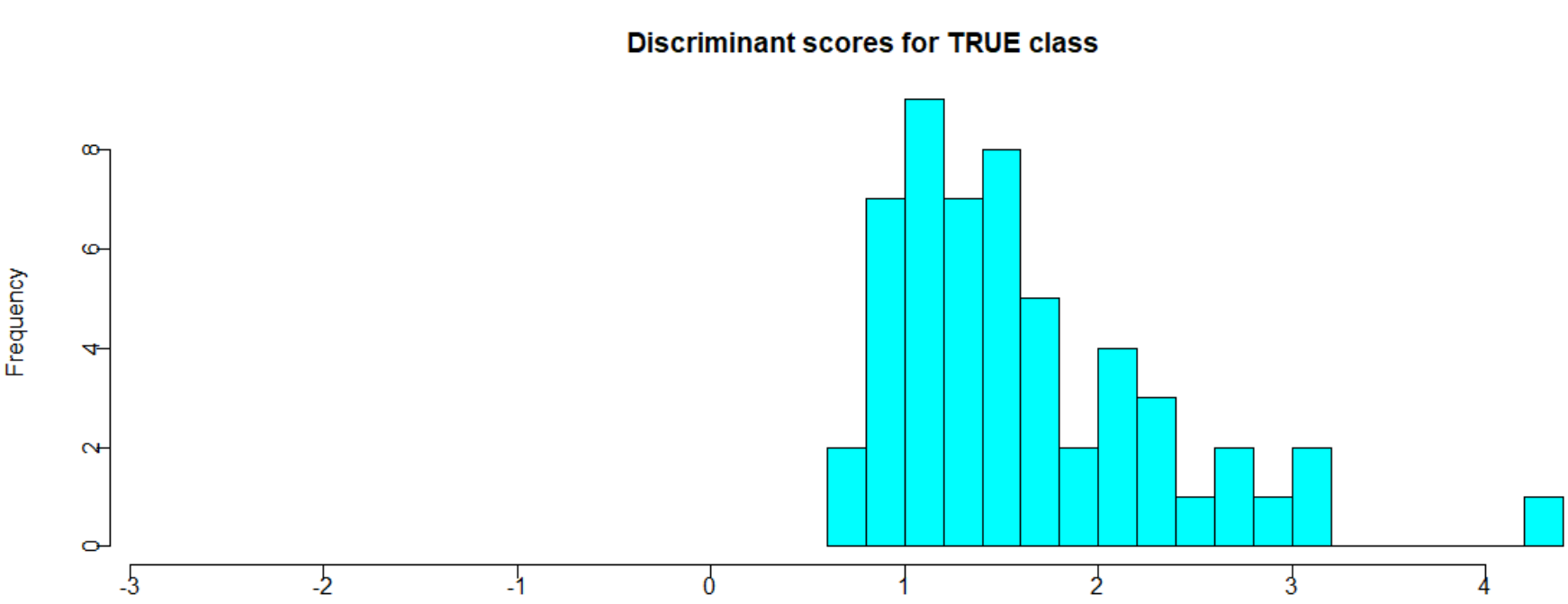

0,0
0,2
0,4
-3
-2
-1
0
1
2
3
4

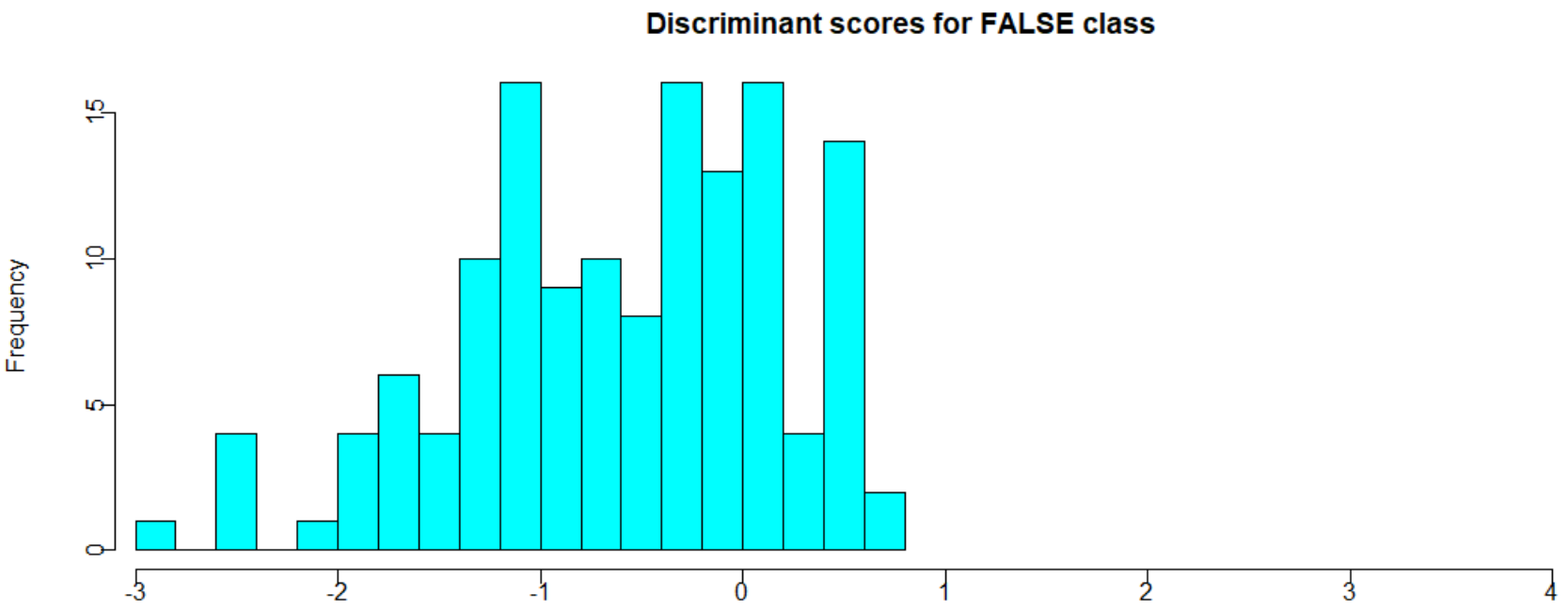
4. 선형판별분석 방법

```
> tail(pred_lda$class)       #집단구분
[1] True  False True  True  True  True
Levels: False True
> tail(pred_lda$posterior)   #사후확률
         False      True
729 0.46893993 0.5310601
746 0.67021268 0.3297873
747 0.28672127 0.7132787
750 0.49468297 0.5053170
755 0.26534289 0.7346571
760 0.04048226 0.9595177
> tail(pred_lda$x)           #판별점수
          LD1
729 0.7481552
746 0.1777536
747 1.2866774
750 0.6775816
755 1.3599113
760 2.8292434
```

4. 선형판별분석 방법

```
#manually calculate discriminant score using discriminant
rowNum <- 156
model_lda$scaling          #판별함수 계수
sum(testing[rowNum, -9]*model_lda$scaling)-pred_lda$x[rowNum]
par(mfrow = c(2,1))
hist(pred_lda$x[as.integer(pred_lda$class)==2], breaks = 20, col =
'cyan', xlim = range(pred_lda$x), main = "Discriminant scores for
TRUE class")
hist(pred_lda$x[as.integer(pred_lda$class)==1], breaks = 20, col =
'cyan', xlim = range(pred_lda$x), main = "Discriminant scores for
FALSE class")
tab <- table(pred_lda$class, testing$Outcome)
tab
sum(tab[row(tab)==col(tab)])/sum(tab)
```


Discriminant scores for TRUE class
Frequency
0
2
4
6
8
-3
-2
-1
0
1
2
3
4


Discriminant scores for FALSE class
Frequency
0
5
10
15
-3
-2
-1
0
1
2
3
4

4. 선형판별분석 방법

- 나이브 베이즈 분류
  - ✓ 훈련용 데이터셋(training)으로 학습하고, 검증용 데이터셋(testing)으로 예측
  - ✓ 종속변수 사전확률, 집단 별 설명변수의 분포 or 평균/표준편차 확인
  - ✓ Accuracy로 예측 성능 평가

```
library(e1071)
model_nb <- naiveBayes(Outcome ~ ., data = training, laplace = 1)
#laplace smoothing is enabled
#Assumes Gaussian distribution of features given class
model_nb
model_nb$apriori
model_nb$tables
mean(training$Age[as.integer(training$Outcome)==2])
sd(training$Age[as.integer(training$Outcome)==1])
pred_nb <- predict(model_nb, testing[,-9])
tab <- table(pred_nb, testing$Outcome)
tab
sum(tab[row(tab)==col(tab)])/sum(tab)
```

4. 선형판별분석 방법

```
> model_nb

Naive Bayes Classifier for Discrete Predictors

Call:
naiveBayes.default(x = X, y = Y, laplace = laplace)

A-priori probabilities:
Y
  False    True
0.65625 0.34375

Conditional probabilities:
       Pregnancies
Y           [,1]     [,2]
  False 3.412698 3.037179
  True  4.686869 3.692897

       Glucose
Y           [,1]     [,2]
  False 109.6931 25.29164
  True  141.0051 29.58023

       BloodPressure
Y           [,1]     [,2]
  False 70.14021 11.78516
  True  75.16667 12.47851
```

Weekly Assignment #8

- 통신회사 고객이탈 예측
  - Logistic regression/LDA/Naïve Bayes Classifier 사용
  - 종속변수: Churn (지난 달 이탈 여부)
  - 독립변수:
    - ✓ 고객이 이용중인 서비스: phone, multiple lines, internet, online security, online backup, device protection, tech support, and streaming TV and movies
    - ✓ 고객계정 관련 정보: how long they've been a customer(tenure), contract, payment method, paperless billing, monthly charges, and total charges
    - ✓ 고객의 인구통계학 정보 – gender, age range, and if they have partners and dependents
  - 참고: https://www.kaggle.com/blastchar/telco-customer-churn/data#