데이터마이닝 / 정보디자인

# Linear and Logistic Regression

광운대학교 정보융합학부

2018204036 윤서안

# 1. 일상생활에서 볼 수 있는 Odd 사례

Odds = p / (1-p)

P를 성공 범주에 속할 확률로 정의할 때 Odds는 성공 대비 실패의 확률 비율로 정의

주식 시장에서 배당금의 위치를 추측하기 위해 계산하는 일을 예시로 들 수 있다. 배당 권리가 있는 주식에 투자했을 때 해당 기업이 영업이익을 냈다면 이익잉여금의 일부를 주식 소유자에게 분배해 주는데 이것을 배당금이라고 한다. 자신이 어느 정도의 이익을 볼 수 있을지 그 승산을 계산하여 투자를 결정할 때 Odd가 사용된다고 볼 수 있다.

# 2. 선형회귀분석과 로지스틱회귀분석

## 1) 데이터 수집 및 전처리

선형회귀분석을 위한 데이터 (컬럼이 많기 때문에 사진을 나눠서 첨부)

| | A | B | C | D | E | F | G | H | I | J | K | L |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | school | sex | age | address | famsize | Pstatus | Medu | Fedu | Mjob | Fjob | reason | guardian |
| 2 | GP | F | 18 | U | GT3 | A | 4 | 4 | at_home | teacher | course | mother |
| 3 | GP | F | 17 | U | GT3 | T | 1 | 1 | at_home | other | course | father |
| 4 | GP | F | 15 | U | LE3 | T | 1 | 1 | at_home | other | other | mother |
| 5 | GP | F | 15 | U | GT3 | T | 4 | 2 | health | services | home | mother |
| 6 | GP | F | 16 | U | GT3 | T | 3 | 3 | other | other | home | father |
| 7 | GP | M | 16 | U | LE3 | T | 4 | 3 | services | other | reputation | mother |
| 8 | GP | M | 16 | U | LE3 | T | 2 | 2 | other | other | home | mother |
| 9 | GP | F | 17 | U | GT3 | A | 4 | 4 | other | teacher | home | mother |
| 10 | GP | M | 15 | U | LE3 | A | 3 | 2 | services | other | home | mother |

| M | N | O | P | Q | R | S | T | U | V | W | X |
|---|---|---|---|---|---|---|---|---|---|---|---|
| traveltime | studytime | failures | schoolsup | famsup | paid | activities | nursery | higher | internet | romantic | famrel |
| 2 | 2 | 0 | yes | no | no | no | yes | yes | no | no | 4 |
| 1 | 2 | 0 | no | yes | no | no | no | yes | yes | no | 5 |
| 1 | 2 | 0 | yes | no | no | no | yes | yes | yes | no | 4 |
| 1 | 3 | 0 | no | yes | no | yes | yes | yes | yes | yes | 3 |
| 1 | 2 | 0 | no | yes | no | no | yes | yes | no | no | 4 |
| 1 | 2 | 0 | no | yes | no | yes | yes | yes | yes | no | 5 |
| 1 | 2 | 0 | no | no | no | no | yes | yes | yes | no | 4 |
| 2 | 2 | 0 | yes | yes | no | no | yes | yes | no | no | 4 |
| 1 | 2 | 0 | no | yes | no | no | yes | yes | yes | no | 4 |

| Y | Z | AA | AB | AC | AD | AE | AF | AG |
|---|---|---|---|---|---|---|---|---|
| freetime | goout | Dalc | Walc | health | absences | G1 | G2 | G3 |
| 3 | 4 | 1 | 1 | 3 | 4 | 0 | 11 | 11 |
| 3 | 3 | 1 | 1 | 3 | 2 | 9 | 11 | 11 |
| 3 | 2 | 2 | 3 | 3 | 6 | 12 | 13 | 12 |
| 2 | 2 | 1 | 1 | 5 | 0 | 14 | 14 | 14 |
| 3 | 2 | 1 | 2 | 5 | 0 | 11 | 13 | 13 |
| 4 | 2 | 1 | 2 | 5 | 6 | 12 | 12 | 13 |
| 4 | 4 | 1 | 1 | 3 | 0 | 13 | 12 | 13 |
| 1 | 4 | 1 | 1 | 1 | 2 | 10 | 13 | 13 |
| 2 | 2 | 1 | 1 | 1 | 0 | 15 | 16 | 17 |

UC Irvine 머신 러닝 저장소에서 제공하는포르투갈의 두 학교 학생들의 데이터이다. 총 649명의 학생들의 다양한 정보와 세 학기의 국어(포르투갈어)성적을 포함하여 33개의 컬럼이 있다. 종속 변수를 G3(마지막 학기의 성적)으로 두고 상관성을 분석할 예정이다.

데이터 전처리 R 코드

```
### 데이터 불러오기 ###
student <- read.csv("student.csv")

### 데이터 전처리 ###
student <- student[,-c(31, 32)] # 상관성이 너무 큰 변수 제거
student <- student[,-c(1)] # 의미 없는 변수(학교) 제거

### 범주형 변수 변환 ###
install.packages("dummies")
library(dummies)
student <- dummy.data.frame(student, names = c("school", "sex", "address", "famsize", "Pstatus",
                                               "Mjob", "Fjob", "reason", "guardian", "schoolsup",
                                               "famsup", "paid", "activities", "nursery", "higher",
                                               "internet", "romantic"))
student <- student[,-c(2, 5, 7, 9, 32, 34, 36, 38, 40, 42, 44, 46)] # 더미 변수로 인해 중복되는 컬럼 제거

### 데이터 스케일링 ###
student <- transform(student, failures = scale(student$failures),
                     absences = scale(student$absences)) # 표준화
```

우선 G3과 상관성이 너무 높은 G1, G2 변수를 제거하고의미 없는 학교 변수를 제거했다. 그리고 범주형 변수들을 모두 더미 변수로 변환했다. 값이 yes or no인 변수들은 더미 변수로 변환하면 중복되는 컬럼이 생기기 때문에 no 컬럼을 제거했다.

그 후 데이터 스케일링을 진행하려고 했는데 숫자형 변수들은 대부분 조사 당시에 5점 척도를 사용하여 나타낸 값이었기 때문에 적절하지 않다고 생각했다. 그래서 범위가 다른 변수인 failures와 absences에 표준화를 적용시켰다.

## 로지스틱회귀분석을 위한 데이터

| | A | B | C | D | E | F | G | H | I | J | K |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | age | job | marital | education | default | housing | loan | contact | month | day_of_we | duration |
| 2 | 56 | housemaid | married | basic.4y | no | no | no | telephone | may | mon | 261 |
| 3 | 57 | services | married | high.school | unknown | no | no | telephone | may | mon | 149 |
| 4 | 37 | services | married | high.school | no | yes | no | telephone | may | mon | 226 |
| 5 | 40 | admin. | married | basic.6y | no | no | no | telephone | may | mon | 151 |
| 6 | 56 | services | married | high.school | no | no | yes | telephone | may | mon | 307 |
| 7 | 45 | services | married | basic.9y | unknown | no | no | telephone | may | mon | 198 |
| 8 | 59 | admin. | married | professional.course | no | no | no | telephone | may | mon | 139 |
| 9 | 41 | blue-collar | married | unknown | unknown | no | no | telephone | may | mon | 217 |
| 10 | 24 | technician | single | professional.course | no | yes | no | telephone | may | mon | 380 |

| L | M | N | O | P | Q | R | S | T | U |
|---|---|---|---|---|---|---|---|---|---|
| campaign | pdays | previous | poutcome | emp.var.rate | cons.price.idx | cons.conf.idx | euribor3m | nr.employed | y |
| 1 | 999 | 0 | nonexistent | 1.1 | 93.994 | -36.4 | 4.857 | 5191 | no |
| 1 | 999 | 0 | nonexistent | 1.1 | 93.994 | -36.4 | 4.857 | 5191 | no |
| 1 | 999 | 0 | nonexistent | 1.1 | 93.994 | -36.4 | 4.857 | 5191 | no |
| 1 | 999 | 0 | nonexistent | 1.1 | 93.994 | -36.4 | 4.857 | 5191 | no |
| 1 | 999 | 0 | nonexistent | 1.1 | 93.994 | -36.4 | 4.857 | 5191 | no |
| 1 | 999 | 0 | nonexistent | 1.1 | 93.994 | -36.4 | 4.857 | 5191 | no |
| 1 | 999 | 0 | nonexistent | 1.1 | 93.994 | -36.4 | 4.857 | 5191 | no |
| 1 | 999 | 0 | nonexistent | 1.1 | 93.994 | -36.4 | 4.857 | 5191 | no |
| 1 | 999 | 0 | nonexistent | 1.1 | 93.994 | -36.4 | 4.857 | 5191 | no |

UC Irvine 머신 러닝 저장소에서 제공하는포르투갈 금융 기관의 마케팅 데이터이다. 총 41189개의 행(고객의 수)과 21개의 컬럼이 있다. 종속 변수를 y(고객의 정기 예금 신청 여부)로 두고 분석을 진행하려고 한다.

## 데이터 전처리 R 코드

```r
### 데이터 불러오기 ###
bank <- read.csv("bank.csv")

### 데이터 전처리 ###
bank <- bank[,-c(11)] # 상관성이 너무 큰 변수 제거
bank$pdays <- replace(bank$pdays, bank$pdays == 999, 50) # 결측치 처리

### 범주형 변수 변환 ###
install.packages("dummies")
library(dummies)
bank <- dummy.data.frame(bank, names = c("job", "marital", "education", "default", "housing",
                                         "loan", "Fjob", "contact", "month", "day_of_week",
                                         "poutcome", "y"))
bank
bank <- bank[,-c(26, 29, 32, 63)]

### 데이터 스케일링 ###
bank <- transform(bank, emp.var.rate = scale(bank$emp.var.rate),
                  cons.price.idx = scale(bank$cons.price.idx),
                  cons.conf.idx = scale(bank$cons.conf.idx),
                  euribor3m = scale(bank$euribor3m),
                  nr.employed = scale(bank$nr.employed)) # 표준화

normalize <- function(n){
  return((n - min(n)) / (max(n) - min(n))) # 정규화
}
bank <- transform(bank, campaign = normalize(bank$campaign),
                  previous = normalize(bank$previous))
```

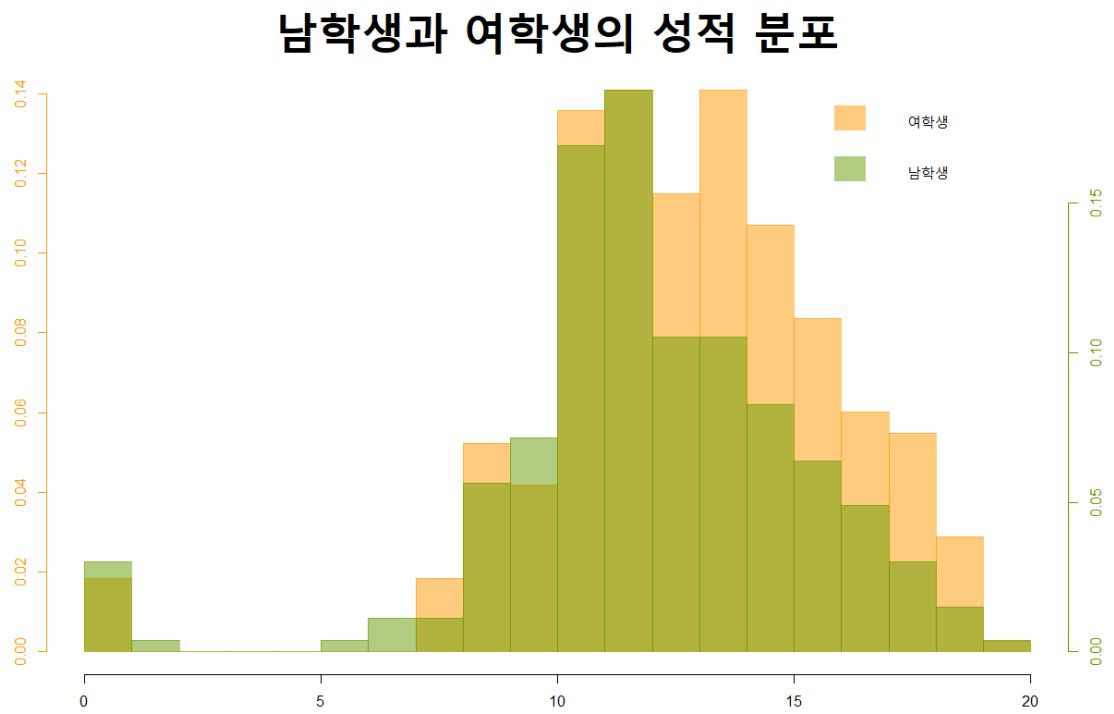
우선 변수 중 duration은 마지막으로 고객과 연락했을 때의 연락 지속 시간을 의미한다. 이 시간이 길면 고객이 정기 예금을 신청하고 시간이 짧으면 신청하지 않을 확률이 높아 종속 변수와의 상관성이 너무 높기 때문에 이 변수를 제거하였다. 그리고 변수 pdays는 이번 마케팅이 아닌 직전 마케팅에서 고객과 연락한 후 지난 일 수이며 만약 직전 마케팅에서 고객과 연락을 하지 않았다면 999 즉, 결측치라고 볼 수 있는 값으로 처리되었다. 이 숫자를 그대로 사용하여 분석하면 분석 결과에 안 좋은 영향을 미칠 수 있으므로 최대 일 수인 28과 크게 차이 나지 않는 50으로 값을 바꿔주었다.

다음으로는 범주형 변수 변환을 했는데 이 때 값이 yes or no인 변수들은 더미 변수로 변환하면 중복되는 컬럼이 생기기 때문에 no 컬럼을 제거했다. 하지만 값이 unknown 컬럼은 어떤 값인지 모르는 값이므로 그대로 두었다.

마지막으로 데이터 스케일링을 하였다. 고용 변동률을 나타내는 변수인 emp.var.rate, 소비자 물가지수를 나타내는 변수인 cons.price.idx, 소비자 신뢰지수를 나타내는 변수인 cons.conf.idx, 3개월 동안의 유로 금리를 나타내는 변수인 euribor3m, 고용자의 수를 나타내는 변수인 nr.employed에 표준화를 적용시켰다. 그리고 이번 마케팅을 진행하는 동안 고객에게 한 연락 수인 campaign과 직전 마케팅을 진행하는 동안 고객에게 한 연락 수인 previous에 정규화를 적용시켰다.

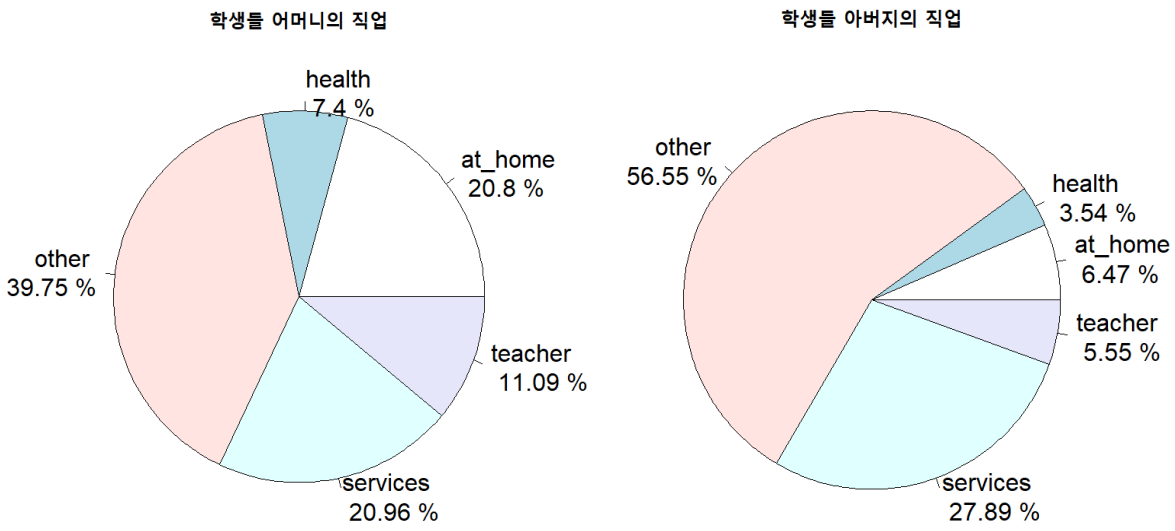
2) 탐색적 데이터 분석

Student 데이터의 남학생과 여학생의 성적 분포 히스토그램



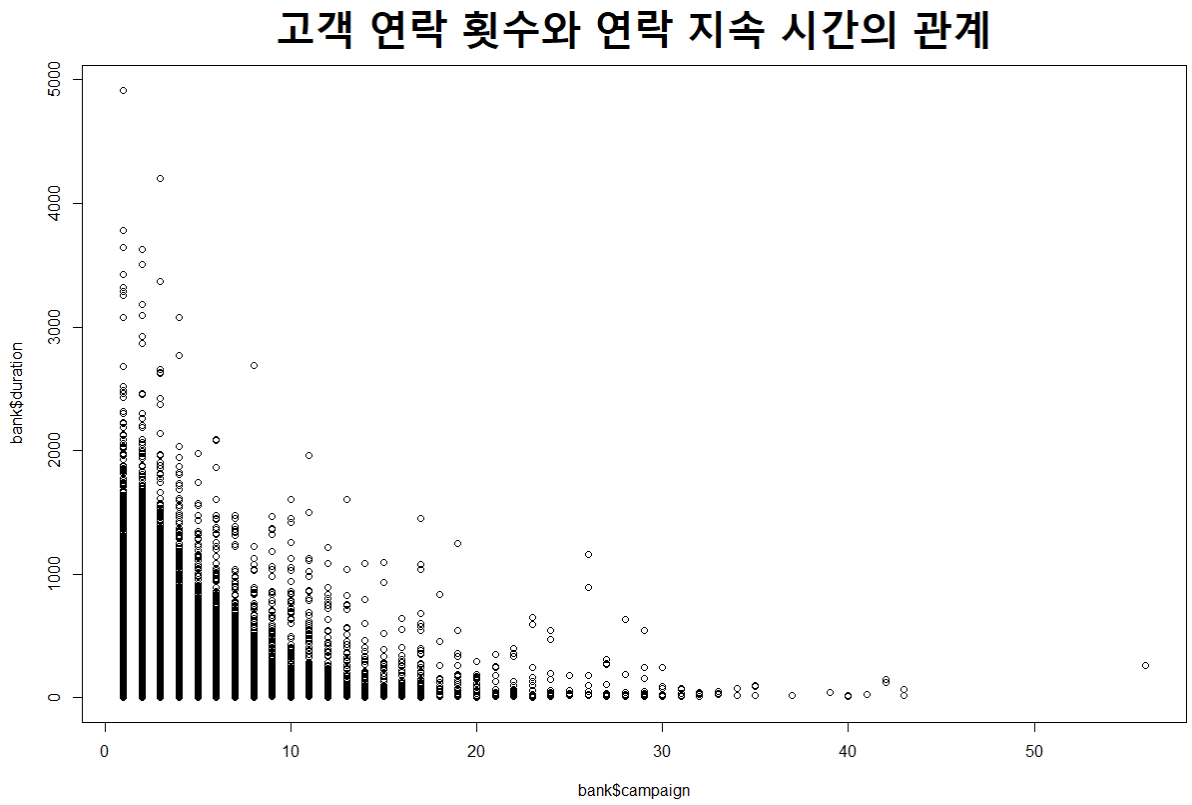

두 가지의 히스토그램을 3D가 아닌 평면에 겹쳐 놓아서 두 그래프가 잘 보이고 비교가 쉽도록 그린 점이 잘 된 점이라고 생각했다. 아쉬운 점은 그래프의 y축의 범위가 달라서 한 눈에 비교하기가 어렵다는 점이다. R 코드에서 ylim을 각각 같은 값으로 설정해주면 해결할 수 있을 것이라고 생각했다.

Student 데이터의 학생들의 어머니와 아버지 직업 분포 파이 차트



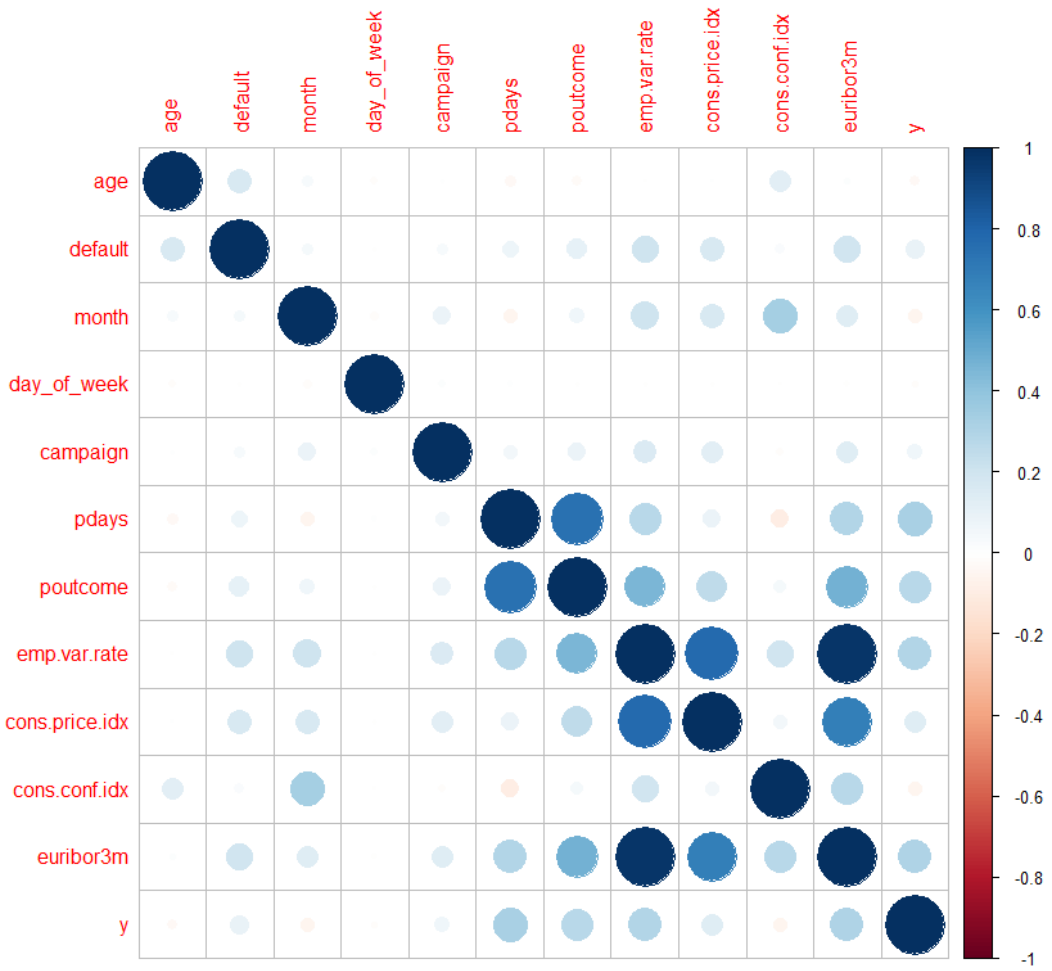
잘 된 점은 실제 값이 아닌 비율을 계산하여 퍼센트로 표현했기 때문에 비교하기 쉬운 점이라고 생각했다. 아쉬운 점은 각 값이 크기 순으로 정렬되어 있지 않다는 점이다.

Bank 데이터의 고객 연락 횟수와 연락 지속 시간 관계 산점도




잘 된 점은 점으로 표현돼서 위쪽 데이터의 값을 알아보기가 편하다는 점이라고 생각했다. 아쉬운 점은 색깔 정보를 더 활용했다면 좋았을 것 같다.

Bank 데이터의 상관계수 행렬




색깔과 크기가 적절하게 사용되어 복잡할 수도 있는 그래프를 보기 쉽게 표현했다는 점이 잘 된 점이라고 생각했다. 하지만 색이 뚜렷하지 않을수록 구분하기 힘든 것이 아쉬웠기 때문에 R 코드에서 order을 사용하여 같은 색끼리 뭉쳐서 볼 수 있게 해주면 더 좋을 것 같다고 생각했다.

3) 다중선형회귀 모델 구축

다중선형회귀 모델 R 코드

```
### 다중선형회귀 모형 구축 ###
multi_model <- lm(G3 ~., data = student)
summary(multi_model)
```

다중선형회귀 분석 결과 해석

```
Call:
lm(formula = G3 ~ ., data = student)

Residuals:
    Min      1Q  Median      3Q     Max
-13.518  -1.312   0.071   1.487   6.881
```

**잔차(residuals)**는 학습 모델의 오차(e)로 실제 y값과 추정된 y값의 차를 quintile 형식으로 보여준다.

```
Coefficients: (4 not defined because of singularities)
                 Estimate Std. Error t value Pr(>|t|)
(Intercept)       9.961133   2.269465   4.389 1.34e-05 ***
sexF              0.535748   0.252961   2.118 0.034587 *
age               0.134365   0.103666   1.296 0.195417
addressR         -0.632626   0.256488  -2.466 0.013918 *
famsizeGT3       -0.254428   0.248606  -1.023 0.306516
PstatusA         -0.225042   0.351922  -0.639 0.522760
Medu              0.058215   0.153611   0.379 0.704837
Fedu              0.184968   0.139849   1.323 0.186457
Mjobat_home      -0.539120   0.509700  -1.058 0.290601
Mjobhealth        0.458169   0.537813   0.852 0.394598
Mjobother        -0.481103   0.447891  -1.074 0.283179
Mjobservices     -0.035641   0.437155  -0.082 0.935048
Mjobteacher             NA         NA      NA       NA
Fjobat_home      -0.926800   0.678184  -1.367 0.172257
Fjobhealth       -1.331314   0.756556  -1.760 0.078959 .
Fjobother        -0.819350   0.535838  -1.529 0.126758
Fjobservices     -1.390548   0.546091  -2.546 0.011130 *
Fjobteacher             NA         NA      NA       NA
reasoncourse     -0.335490   0.301470  -1.113 0.266213
reasonhome       -0.219195   0.334244  -0.656 0.512203
reasonother      -0.983978   0.419003  -2.348 0.019174 *
reasonreputation        NA         NA      NA       NA
guardianfather   -0.313239   0.537912  -0.582 0.560564
guardianmother   -0.566848   0.494147  -1.147 0.251780
guardianother           NA         NA      NA       NA
```

**계수(Coefficient)**는 선형 회귀의 베타 값을 의미하며 Intercept는 B0를 의미한다. Intercept의 **p-value**가 낮을수록 유의성이 높다고 볼 수 있는데 위 결과에서는 1.34e-05로 매우 낮으므로 모델의 회귀 계수가 모두 유의하다고 볼 수 있다.

```
traveltime        -0.004981   0.160910  -0.031 0.975314
studytime          0.472961   0.141329   3.347 0.000869 ***
failures          -0.864336   0.123063  -7.024 5.79e-12 ***
schoolsupyes      -1.107323   0.366841  -3.019 0.002646 **
famsupyes         -0.019677   0.231846  -0.085 0.932390
paidyes           -0.398780   0.468574  -0.851 0.395076
activitiesyes      0.271796   0.226577   1.200 0.230771
nurseryyes        -0.237719   0.275580  -0.863 0.388688
higheryes          1.769031   0.388619   4.552 6.41e-06 ***
internetyes        0.387091   0.278973   1.388 0.165780
romanticyes       -0.481137   0.232527  -2.069 0.038951 *
famrel             0.181031   0.117844   1.536 0.125010
freetime          -0.152300   0.114040  -1.335 0.182214
goout             -0.095639   0.108952  -0.878 0.380397
Dalc              -0.222987   0.155394  -1.435 0.151805
Walc              -0.069706   0.120280  -0.580 0.562443
health            -0.171288   0.078317  -2.187 0.029114 *
absences          -0.085317   0.115355  -0.740 0.459827
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.707 on 610 degrees of freedom
Multiple R-squared:  0.3392,    Adjusted R-squared:  0.298
F-statistic: 8.239 on 38 and 610 DF,  p-value: < 2.2e-16
```

**결정계수(R-squared)**는 0.3392, 변수 증가를 방지하는 **수정 결정계수(Adjusted R-squared)**는 0.298로 결정계수 값에 따르면 이 모델은 33.92%의 설명력을 가진다고 할 수 있다.

학생들의 마지막 학기 성적은 평균적으로 9.96점 정도이며 성별(sex), 거주 지역(address), 아버지의 직업(Fjob), 재학중인 학교에 지원한 이유(reason), 공부 시간(studytime), 수업에서 F를 받은 횟수(failures), 학교 외의 교육 여부(schoolsup), 더 높은 수준의 교육을 받기를 원하는지의 여부(higher), 연애 중인지의 여부(romantic), 건강 수치(health)가 마지막 학기 성적에 통계적으로 유의한 영향을 준다고 할 수 있다.

모델 성능 평가 R 코드

```r
### 모델 성능 평가 ###
d <- deviance(multi_model) # 잔차제곱합
mse <- d / 610 # MSE
mse

install.packages("forecast")
library(forecast)
accuracy(multi_model) # ME, RMSE, MAE, MPE, MAPE, MASE
```

먼저 **MSE**는 Mean Squared Error로 실제 데이터와 모델이 떨어진 정도를 의미하는데 모든 잔차제곱의 합을 자유도(degrees of freedom)으로 나누어 구할 수 있다. 앞서 summary 결과에서 자유도가 610인 것을 확인했고 R의 내장 함수인 deviance()를 사용하여 잔차제곱합을 구했다.

다음으로 **MAE**는 Mean of Absolute Error로 MSE에 제곱근 시킨 RMSE라는 지표에 오차의 크기만 고려하기 위해 오차에 절대값을 씌우고 데이터 수로 나눈 것이며 forecast 패키지의 accuracy() 함수를 사용하여 구했다.

결과 해석

```
> mse
[1] 7.326884

> accuracy(multi_model) # ME, RMSE, MAE, MPE, MAPE, MASE
                      ME     RMSE      MAE  MPE MAPE      MASE
Training set -4.954545e-17 2.624232 1.88859 -Inf  Inf 0.7849884
```

MSE와 MAE는 값이 작을수록 회귀 성능이 좋다. 위 결과에서 MSE의 값은 약 7.33, MAE의 값은 약 1.89 정도가 나왔다. 이것은 예측 변수에 비해 크게 작은 값은 아니기 때문에 추정된 회귀식은 아주 타당하다고는 볼 수 없다.

변수선택법 적용 R 코드

```
### 변수 선택법 적용 ###
model_fwd <- step(lm(G3 ~ 1, student, family = binomial()),
                  direction = "forward", trace = 0, scope = formula(multi_model))
summary(model_fwd) # forward selection

model_bwd <- step(lm(G3 ~ ., student, family = binomial()),
                  direction = "backward", trace = 0,
                  scope = list(lower = G3 ~ 1, upper = formula(multi_model)))
summary(model_bwd) # backward selection

model_step <- step(lm(G3 ~ ., student, family = binomial()),
                   direction = "both", trace = 0,
                   scope = list(lower = G3 ~ 1, upper = formula(multi_model)))
summary(model_step) # stepwise selection
```

변수선택법 적용 후 분석 결과

```
Residual standard error: 2.693 on 626 degrees of freedom
Multiple R-squared:  0.3288,    Adjusted R-squared:  0.3052
F-statistic: 13.94 on 22 and 626 DF,  p-value: < 2.2e-16
```

Backward Selection과 Stepwise Selection을 적용했을 때 수정 결정계수의 값이 0.3025로 증가했다. 따라서 모델의 설명력이 조금 높아졌다고 볼 수 있다.

4) 로지스틱회귀 모델 구축

로지스틱회귀 모델 R 코드

```r
### 성능 평가 함수 정의 ###
perf_eval <- function(cm){
  TPR = Recall = cm[2,2] / sum(cm[2,]) # true positive rate
  Precision = cm[2,2] / sum(cm[,2]) # precision
  TNR = cm[1,1] / sum(cm[1,]) # true negative rate
  ACC = sum(diag(cm)) / sum(cm) # accuracy
  BCR = sqrt(TPR * TNR) # balance corrected accuracy (geometric mean)
  F1 = 2 * Recall * Precision / (Recall + Precision) # f1 measure

  re <- data.frame(TPR = TPR,
                   Precision = Precision,
                   TNR = TNR,
                   ACC = ACC,
                   BCR = BCR,
                   F1 = F1)
  return(re)
}

### training 데이터와 test 데이터 나누기 ###
set.seed(2020)
test_id <- sample(1:nrow(bank), round(nrow(bank) * 0.7))
bank_train <- bank[test_id, ]
bank_test <- bank[-test_id, ]

### 로지스틱회귀 모형 구축 ###
logistic_model <- glm(yyes ~ ., bank_train, family = binomial())
summary(logistic_model)

### 예측 수행 ###
pred_prob <- predict(logistic_model, bank_test, type = "response")
pred_class <- rep(0, nrow(bank_test))
pred_class[pred_prob > 0.5] <- 1
cm <- table(pred = pred_class, actual = bank_test$yyes)
perf_eval(cm)
```

로지스틱회귀 분석 결과 해석

```
Call:
glm(formula = yyes ~ ., family = binomial(), data = bank_train)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.1245  -0.3900  -0.3151  -0.2577   2.9233
```

```
Call:
lm(formula = G3 ~ ., data = student)

Residuals:
    Min      1Q  Median      3Q     Max
-13.518  -1.312   0.071   1.487   6.881
```

**잔차이탈도(Deviance Residuals)**는 독립 변수를 포함한 모형의 적합도로 낮을수록 좋은 모형이라고 볼 수 있다. 위 결과에서는 잔차이탈도와 잔차를 quintile 형식으로 보여주고 있다.

```
Coefficients: (8 not defined because of singularities)
                              Estimate Std. Error z value Pr(>|z|)
(Intercept)                  -8.851e-01  6.123e-01  -1.446 0.148272
age                           6.311e-04  2.558e-03   0.247 0.805128
jobadmin.                     2.092e-01  2.605e-01   0.803 0.421818
jobblue.collar                2.176e-03  2.626e-01   0.008 0.993389
jobentrepreneur               2.434e-01  2.804e-01   0.868 0.385510
jobhousemaid                  1.608e-01  2.910e-01   0.553 0.580603
jobmanagement                 8.154e-02  2.695e-01   0.303 0.762260
jobretired                    4.033e-01  2.713e-01   1.486 0.137166
jobself.employed              1.687e-01  2.807e-01   0.601 0.547852
jobservices                  -3.965e-02  2.691e-01  -0.147 0.882870
jobstudent                    4.249e-01  2.795e-01   1.521 0.128371
jobtechnician                 2.295e-01  2.630e-01   0.873 0.382900
jobunemployed                 2.331e-01  2.847e-01   0.819 0.412935
jobunknown                           NA         NA      NA       NA
maritaldivorced              -5.113e-02  5.070e-01  -0.101 0.919659
maritalmarried               -6.231e-02  5.034e-01  -0.124 0.901488
maritalsingle                -4.951e-03  5.041e-01  -0.010 0.992163
maritalunknown                       NA         NA      NA       NA
educationbasic.4y            -1.546e-01  1.263e-01  -1.224 0.220845
educationbasic.6y             3.904e-02  1.432e-01   0.273 0.785068
educationbasic.9y            -3.728e-02  1.192e-01  -0.313 0.754544
educationhigh.school         -1.182e-01  1.116e-01  -1.060 0.289288
educationilliterate           8.007e-01  7.176e-01   1.116 0.264534
educationprofessional.course -8.415e-02  1.209e-01  -0.696 0.486508
educationuniversity.degree   -4.665e-02  1.104e-01  -0.423 0.672655
educationunknown                     NA         NA      NA       NA
defaultunknown               -3.356e-01  7.049e-02  -4.760 1.93e-06 ***
defaultyes                   -8.678e+00  1.393e+02  -0.062 0.950314
housingunknown               -7.080e-02  1.425e-01  -0.497 0.619212
housingyes                   -4.152e-02  4.342e-02  -0.956 0.338943
loanunknown                          NA         NA      NA       NA
loanyes                      -2.496e-02  5.943e-02  -0.420 0.674536
contactcellular               6.952e-01  7.976e-02   8.716  < 2e-16 ***
contacttelephone                     NA         NA      NA       NA

monthapr                     -1.392e-01  1.893e-01  -0.735 0.462263
monthaug                      3.119e-01  1.515e-01   2.059 0.039523 *
monthdec                      5.668e-01  2.305e-01   2.459 0.013934 *
monthjul                     -3.741e-02  1.846e-01  -0.203 0.839422
monthjun                     -7.216e-01  2.518e-01  -2.866 0.004156 **
monthmar                      1.339e+00  1.690e-01   7.925 2.28e-15 ***
monthmay                     -5.393e-01  1.608e-01  -3.354 0.000796 ***
monthnov                     -6.157e-01  1.628e-01  -3.781 0.000156 ***
monthoct                     -1.987e-01  1.556e-01  -1.277 0.201596
monthsep                             NA         NA      NA       NA
day_of_weekfri               -1.221e-01  6.790e-02  -1.798 0.072211 .
day_of_weekmon               -3.995e-01  6.844e-02  -5.837 5.33e-09 ***
day_of_weekthu               -1.190e-01  6.582e-02  -1.809 0.070525 .
day_of_weektue               -1.533e-01  6.734e-02  -2.276 0.022825 *
day_of_weekwed                       NA         NA      NA       NA
campaign                     -2.333e+00  6.053e-01  -3.855 0.000116 ***
pdays                        -2.570e-02  5.924e-03  -4.339 1.43e-05 ***
previous                     -1.769e-01  4.780e-01  -0.370 0.711295
poutcomefailure              -7.556e-01  2.582e-01  -2.926 0.003433 **
poutcomenonexistent          -2.159e-01  2.565e-01  -0.842 0.400067
poutcomesuccess                      NA         NA      NA       NA
```

**계수(Coefficient)**는 선형 회귀의 베타 값을 의미하며 Intercept는 B0를 의미한다. Intercept의 **p-value**가 낮을수록 유의성이 높다고 볼 수 있는데 위 결과에서는 약 0.15로 크게 유의하지는 않다는 것을 알 수 있다.

```
emp.var.rate                -2.361e+00  2.351e-01 -10.043  < 2e-16 ***
cons.price.idx               1.150e+00  1.533e-01   7.502 6.26e-14 ***
cons.conf.idx                1.055e-01  3.895e-02   2.710 0.006729 **
euribor3m                    5.576e-01  2.387e-01   2.336 0.019510 *
nr.employed                  3.108e-01  2.363e-01   1.315 0.188432
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 20340  on 28831  degrees of freedom
Residual deviance: 15750  on 28780  degrees of freedom
AIC: 15854

Number of Fisher Scoring iterations: 10
```

고객이 신용불량자인지 여부(default), 연락한 방식(contact), 마지막으로 연락한 날의 달(month)과 요일(day_of_week), 현재 마케팅 기간 중 고객과 연락한 횟수(campaign), 직전 마케팅에서 고객과 연락한 후 지난 일 수(pdays), 직전 마케팅이 성공했는지 실패했는지의 결과(poutcome), 고용 변동률(emp.var.rate), 소비자 물가지수(cons.price.idx), 소비자 신뢰지수(cons.conf.idx), 3개월 동안의 유로 금리(euribor3m)가 고객의 정기 예금 신청 여부에 통계적으로 유의한 영향을 준다고 볼 수 있다.

```
> perf_eval(cm)
        TPR Precision       TNR       ACC       BCR        F1
1 0.6388889 0.2329957 0.9105636 0.899482 0.7627247 0.3414634
```

실제 정기 예금을 신청한 고객 중 예측한 신청자의 비율을 나타내는 **TPR(Recall)** 값은 비교적 높지만 정기 예금을 신청했다고 예측한 고객 중 실제 신청자의 비율을 나타내는 **Precision**의 값은 낮고 둘의 조화 평균 척도인 **F1 measure**도 낮게 나왔다. 하지만 전체에서 올바르게 예측한 것이 몇 개인지를 나타내는 **ACC(Accuracy)**는 1에 가까운 높은 수치로 나온 것을 볼 수 있다.

모델 성능 평가 R 코드

```r
### 모델 성능 평가 ###
d <- deviance(logistic_model) # 잔차제곱합
mse <- d / 28780 # MSE
mse

install.packages("forecast")
library(forecast)
accuracy(logistic_model) # ME, RMSE, MAE, MPE, MAPE, MASE
```

결과 해석

```
> mse
[1] 0.5472519

> accuracy(logistic_model) # ME, RMSE, MAE, MPE, MAPE, MASE
                        ME      RMSE       MAE  MPE MAPE      MASE
Training set -6.578187e-10 0.2779624 0.1544462 -Inf  Inf 0.7704546
```

MSE와 MAE는 값이 작을수록 회귀 성능이 좋다. 위 결과에서 MSE의 값은 약 0.55, MAE의 값은 약 0.15 정도가 나왔다. 이것은 예측 변수에 비해 크게 작은 값은 아니기 때문에 추정된 회귀식은 아주 타당하다고는 볼 수 없다.

변수선택법 적용 R 코드

```
### 변수 선택법 적용 ###
model_fwd <- step(lm(yyes ~ 1, bank_train, family = binomial()),
                  direction = "forward", trace = 0, scope = formula(logistic_model))
pred_prob <- predict(model_fwd, bank_test, type="response")
pred_class <- rep(0, nrow(bank_test))
pred_class[pred_prob > 0.5] <- 1
cm <- table(pred=pred_class, actual=bank_test$yyes)
perf_eval(cm) # forward selection

model_bwd <- step(lm(yyes ~ ., bank_train, family = binomial()),
                  direction = "backward", trace = 0,
                  scope = list(lower = yyes ~ 1, upper = formula(logistic_model)))
pred_prob <- predict(model_bwd, bank_test, type="response")
pred_class <- rep(0, nrow(bank_test))
pred_class[pred_prob > 0.5] <- 1
cm <- table(pred=pred_class, actual=bank_test$yyes)
perf_eval(cm) # backward selection

model_step <- step(lm(yyes ~ ., bank_train, family = binomial()),
                   direction = "both", trace = 0,
                   scope = list(lower = yyes ~ 1, upper = formula(logistic_model)))
pred_prob <- predict(model_step, bank_test, type="response")
pred_class <- rep(0, nrow(bank_test))
pred_class[pred_prob > 0.5] <- 1
cm <- table(pred=pred_class, actual=bank_test$yyes)
perf_eval(cm) # stepwise selection
```

변수선택법 적용 후 분석 결과

```
        TPR Precision       TNR       ACC       BCR        F1
1 0.6416309 0.2163531 0.9089151 0.8988346 0.7636675 0.3235931
```

세 가지의 변수 선택법 모두 TPR(Recall) 값은 높아졌지만 Precision과 F1 measure 값이 낮아졌다. 이에 따라 ACC(Accuracy)는 약간 낮아졌고 이는 모델의 정확도가 조금 떨어졌다는 것을 의미한다.