# R_ML_STAT_02_모델만들기

## 02 미국에 사는 인디언들의 당뇨병 예측

### 학습 내용

- 로지스틱 회귀분석을 이용하여 당뇨병 여부 예측해 보기

### 라이브러리 불러오기

- 패키지가 없다고 뜨면 install.packages()를이용하여 설치를 진행
- install.packages(“faraway”)
- install.packages(“pscl”)

```
library(faraway)
library(pscl)
```

```
## Classes and Methods for R developed in the
## Political Science Computational Laboratory
## Department of Political Science
## Stanford University
## Simon Jackman
## hurdle and zeroinfl functions by Achim Zeileis
```

```
library(caret)
```

```
## Loading required package: lattice
```

```
##
## Attaching package: 'lattice'
```

```
## The following object is masked from 'package:faraway':
##
##     melanoma
```

```
## Loading required package: ggplot2
```

```
library(ROCR)
```

```
## Loading required package: gplots
```

```
##
## Attaching package: 'gplots'
## The following object is masked from 'package:stats':
##
##     lowess
```

```
search()
```

```
##  [1] ".GlobalEnv"        "package:ROCR"      "package:gplots"
##  [4] "package:caret"     "package:ggplot2"   "package:lattice"
##  [7] "package:pscl"      "package:faraway"   "package:stats"
## [10] "package:graphics"  "package:grDevices" "package:utils"
## [13] "package:datasets"  "package:methods"   "Autoloads"
## [16] "package:base"
```

# 01 데이터 준비 및 나누기

```
data(pima, package="faraway")
pima$test <- factor(pima$test)
dim(pima)
```

```
## [1] 768   9
```

```
head(pima)
```

```
##   pregnant glucose diastolic triceps insulin  bmi diabetes age test
## 1        6     148        72      35       0 33.6    0.627  50    1
## 2        1      85        66      29       0 26.6    0.351  31    0
## 3        8     183        64       0       0 23.3    0.672  32    1
## 4        1      89        66      23      94 28.1    0.167  21    0
## 5        0     137        40      35     168 43.1    2.288  33    1
## 6        5     116        74       0       0 25.6    0.201  30    0
```

```
str(pima)
```

```
## 'data.frame':    768 obs. of  9 variables:
##  $ pregnant : int  6 1 8 1 0 5 3 10 2 8 ...
##  $ glucose  : int  148 85 183 89 137 116 78 115 197 125 ...
##  $ diastolic: int  72 66 64 66 40 74 50 0 70 96 ...
##  $ triceps  : int  35 29 0 23 35 0 32 0 45 0 ...
##  $ insulin  : int  0 0 0 94 168 0 88 0 543 0 ...
##  $ bmi      : num  33.6 26.6 23.3 28.1 43.1 25.6 31 35.3 30.5 0 ...
##  $ diabetes : num  0.627 0.351 0.672 0.167 2.288 ...
##  $ age      : int  50 31 32 21 33 30 26 29 53 54 ...
##  $ test     : Factor w/ 2 levels "0","1": 2 1 2 1 2 1 2 1 2 2 ...
```

- (직접 해보기) 데이터의 확인 내용을 간단하게 Notepad에 적어보자.

# pima

- pregnant : Number of times pregnant
- glucose : Plasma glucose concentration at 2 hours in an oral glucose tolerance test
- diastolic : Diastolic blood pressure (mm Hg)

- triceps : Triceps skin fold thickness (mm)
- insulin : 2-Hour serum insulin (mu U/ml)
- bmi : Body mass index (weight in kg/(height in metres squared))
- diabetes : Diabetes pedigree function
- age : Age (years)
- test : test whether the patient shows signs of diabetes (coded 0 if negative, 1 if positive)
- The data may be obtained from UCI Repository of machine learning databases at http://archive.ics.uci.edu/ml/ (http://archive.ics.uci.edu/ml/)

# 02 데이터 나누기

- 학습용 데이터 50%
- 테스트 용 데이터 50%

```
# 샘플 5:5
idx <- sample(NROW(pima)/2)

# 데이터 셋 나누기
train <- pima[idx, ]
test <- pima[-idx, ]
```

# 03 로지스틱 회귀(Logistic regression) 모델 구하기

- 지도학습(Supervised Learning)의 한 종류
- 종속변수가 범주형인 데이터에 사용되는 기법.
-

```
m <- glm(test ~ pregnant + glucose + bmi, family=binomial, data=train)
m
```

```
##
## Call:  glm(formula = test ~ pregnant + glucose + bmi, family = binomial,
##     data = train)
##
## Coefficients:
## (Intercept)     pregnant      glucose          bmi
##    -7.32445      0.11668      0.02978      0.07993
##
## Degrees of Freedom: 383 Total (i.e. Null);  380 Residual
## Null Deviance:       509.1
## Residual Deviance: 395.6     AIC: 403.6
```

```
summary(m)
```

```
## 
## Call:
## glm(formula = test ~ pregnant + glucose + bmi, family = binomial, 
##     data = train)
## 
## Deviance Residuals: 
##     Min       1Q   Median       3Q      Max  
## -2.0534  -0.8275  -0.4409   0.8306   2.6437  
## 
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)    
## (Intercept) -7.324455   0.842333  -8.695  < 2e-16 ***
## pregnant     0.116679   0.036213   3.222  0.00127 ** 
## glucose      0.029781   0.004437   6.712 1.92e-11 ***
## bmi          0.079932   0.018065   4.425 9.66e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## (Dispersion parameter for binomial family taken to be 1)
## 
##     Null deviance: 509.09  on 383  degrees of freedom
## Residual deviance: 395.60  on 380  degrees of freedom
## AIC: 403.6
## 
## Number of Fisher Scoring iterations: 4
```

- pregnant, glucose, bmi의 p-value중에 가장 낮은 것이 glucose이므로 예측력이 좀 더 강해 보인다.

# 04 모델을 이용하여 예측을 수행하기

- predict(모델, newdata=데이터, type=[])

```
pred <- predict(m , newdata = test , type = "response")
pred[0:10]  # 10개만 보기
```

```
##       385       386       387       388       389       390       391 
## 0.1761319 0.1322363 0.3307852 0.5491063 0.5262903 0.1868749 0.1581803 
##       392       393       394 
## 0.8647732 0.1959110 0.1629253
```

```
# 0 또는 1로 해야 하므로 0.5를 기준으로 TRUE(1), FALSE(0)로 나눈다.
pred <- as.integer(pred > 0.5)
pred[0:10]  # 10개만 보기
```

```
##  [1] 0 0 0 1 1 0 0 1 0 0
```

# 05 모델 평가

# (1) 분할표 확인

```
actual <- test[ , "test"]
xt = xtabs( ~ pred + actual)
xt
```

```
##      actual
## pred   0   1
##    0 234  49
##    1  27  74
```

```
# 확률로 분할표 보기
prop.table(xt)
```

```
##      actual
## pred         0         1
##    0 0.6093750 0.1276042
##    1 0.0703125 0.1927083
```

## (2) confusionMatrix 확인

```
# caret 패키지를 이용한 정확도 및 기타 확인
# library(caret)
pred <- as.factor(pred)
confusionMatrix(pred, actual)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0 234  49
##          1  27  74
##
##                Accuracy : 0.8021
##                  95% CI : (0.7587, 0.8408)
##     No Information Rate : 0.6797
##     P-Value [Acc > NIR] : 5.797e-08
##
##                   Kappa : 0.5229
##
##  Mcnemar's Test P-Value : 0.016
##
##             Sensitivity : 0.8966
##             Specificity : 0.6016
##          Pos Pred Value : 0.8269
##          Neg Pred Value : 0.7327
##              Prevalence : 0.6797
##          Detection Rate : 0.6094
##    Detection Prevalence : 0.7370
##       Balanced Accuracy : 0.7491
##
##        'Positive' Class : 0
##
```

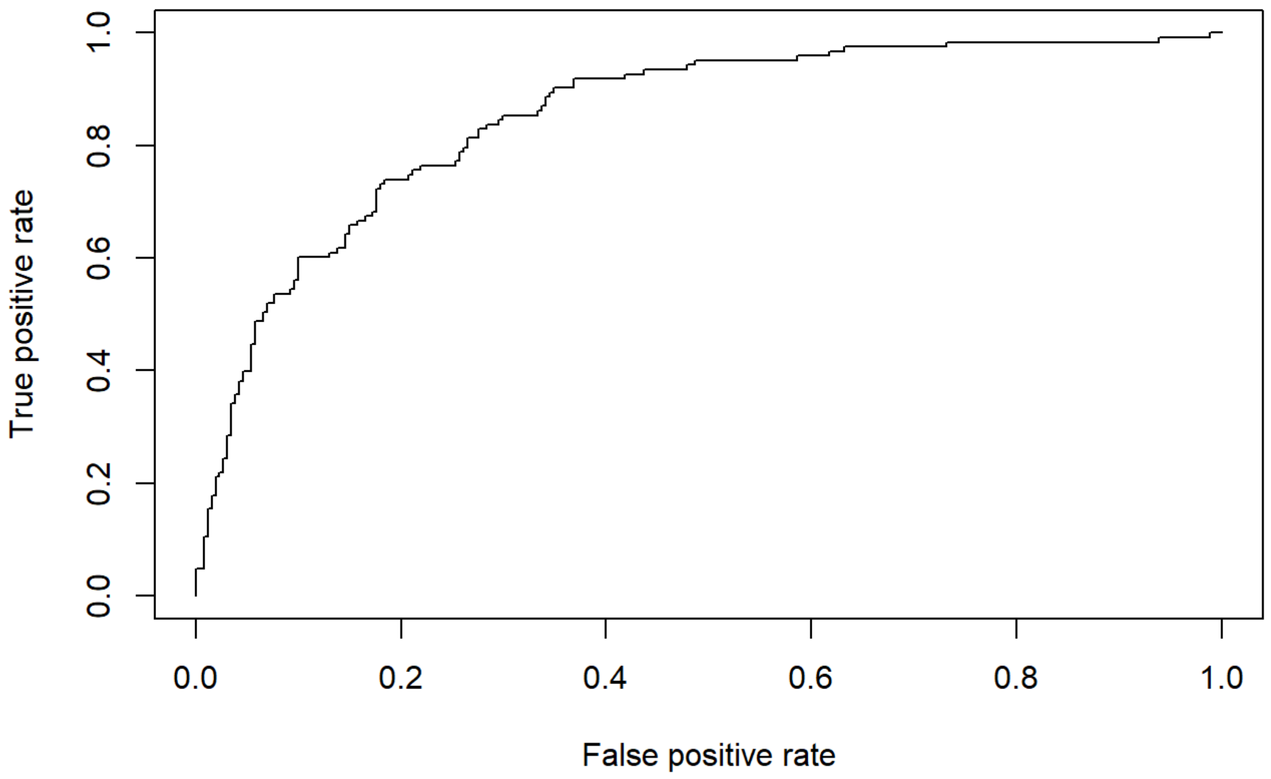
- 약 80%의 정확도

## (3) ROC 커브 그리기

```
library(ROCR)
pred_prob <- predict(m , newdata = test , type = "response")  # 확률 값
str(test)
```

```
## 'data.frame':    384 obs. of  9 variables:
##  $ pregnant : int  1 1 5 8 5 3 1 5 1 4 ...
##  $ glucose  : int  125 119 116 105 144 100 100 166 131 116 ...
##  $ diastolic: int  70 54 74 100 82 68 66 76 64 72 ...
##  $ triceps  : int  24 13 29 36 26 23 29 0 14 12 ...
##  $ insulin  : int  110 50 0 0 285 81 196 0 415 87 ...
##  $ bmi      : num  24.3 22.3 32.3 43.3 32 31.6 32 45.7 23.7 22.1 ...
##  $ diabetes : num  0.221 0.205 0.66 0.239 0.452 0.949 0.444 0.34 0.389 0.463 ...
##  $ age      : int  25 24 35 45 58 28 42 27 21 37 ...
##  $ test     : Factor w/ 2 levels "0","1": 1 1 2 2 2 1 1 2 1 1 ...
```

```
# ROC 커브를 위한 pima의 test 변수을 labels로 지정
labels <- test[  ,"test"]
pred3 <- prediction(pred_prob , labels)
plot(performance(pred3 , "tpr" , "fpr"))
```



```
# AUC 값 확인(1의 값에 가까울 수록 좋다.)
performance(pred3, "auc")
```

```
## An object of class "performance"
## Slot "x.name":
## [1] "None"
##
## Slot "y.name":
## [1] "Area under the ROC curve"
##
## Slot "alpha.name":
## [1] "none"
##
## Slot "x.values":
## list()
##
## Slot "y.values":
## [[1]]
## [1] 0.8511354
##
##
## Slot "alpha.values":
## list()
```

# 더 알아보기

# 로지스틱 회귀에서의 R^2유사한 개념

- Mcfadden R^2
- r2CU를 확인해 보면 약 35%임을 알 수 있음.

```
library(pscl)
pR2(m)
```

```
##          llh      llhNull           G2     McFadden         r2ML
## -197.7986250 -254.5455586  113.4938672    0.2229343    0.2558830
##         r2CU
##    0.3484252
```