LR

KWONHYUNJIN

2018년 9월 20일

```
bc <- read.xlsx("New_version_breast_cancer.xlsx",1)
head(bc)
```

```
##   NA. age tumor.size inv.nodes deg.malig irradiat   Class lt40 ge40
## 1   1   4          4         1         3        0   recur    0    0
## 2   2   5          4         1         1        0 norecur    0    1
## 3   3   5          8         1         2        0   recur    0    1
## 4   4   4          8         1         3        1 norecur    0    0
## 5   5   4          7         2         2        0   recur    0    0
## 6   6   5          6         2         2        1 norecur    0    0
##   premeno node.capse no.node.capse breast.left breast.right quad.cen
## 1       1          1             0           0            1        0
## 2       0          0             1           0            1        1
## 3       0          0             1           1            0        0
## 4       1          1             0           0            1        0
## 5       1          1             0           1            0        0
## 6       1          0             1           0            1        0
##   quad.Ll quad.Lu quad.Rl quad.Ru
## 1       0       1       0       0
## 2       0       0       0       0
## 3       1       0       0       0
## 4       1       0       0       0
## 5       0       0       0       1
## 6       0       1       0       0
```

#set.seed란 랜덤한 값을 시작하기 전에 사용하면 이후에도 같은 값으로 랜덤값을 갖는다. #동일한 랜덤값을 계속해서 받기위해

```
set.seed(123)
bc_shuffle <- bc[sample(nrow(bc)), ]
head(bc_shuffle)
```

```
##     NA. age tumor.size inv.nodes deg.malig irradiat   Class lt40 ge40
## 80   80   3          1         1         2        0 norecur    0    0
## 218 218   3          5         1         3        1   recur    0    0
## 113 113   4          8         4         2        1 norecur    0    0
## 242 242   5          6         1         1        0 norecur    0    0
## 257 257   5          1         1         2        0 norecur    0    1
## 13   13   5          7         1         1        0 norecur    0    1
##     premeno node.capse no.node.capse breast.left breast.right quad.cen
## 80        1          0             1           0            1        1
## 218       1          0             1           1            0        0
## 113       1          1             0           0            1        0
## 242       1          0             1           1            0        0
## 257       0          0             1           1            0        1
## 13        0          0             1           0            1        1
##     quad.Ll quad.Lu quad.Rl quad.Ru
## 80        0       0       0       0
## 218       0       1       0       0
## 113       0       1       0       0
## 242       1       0       0       0
## 257       0       0       0       0
## 13        0       0       0       0
```

```
bc2 <-bc_shuffle[-1]
str(bc2)
```

```
## 'data.frame':    277 obs. of  18 variables:
##  $ age          : num  3 3 4 5 5 5 4 5 6 4 ...
##  $ tumor.size   : num  1 5 8 6 1 7 11 3 3 4 ...
##  $ inv.nodes    : num  1 1 4 1 1 1 1 1 1 1 ...
##  $ deg.malig    : num  2 3 2 1 2 1 2 1 1 2 ...
##  $ irradiat     : num  0 1 1 0 0 0 1 0 0 0 ...
##  $ Class        : Factor w/ 2 levels "norecur","recur": 1 2 1 1 1 1 2 1 1 2 ...
##  $ lt40         : num  0 0 0 0 0 0 0 0 1 0 ...
##  $ ge40         : num  0 0 0 0 1 1 0 1 0 0 ...
##  $ premeno      : num  1 1 1 1 0 0 1 0 0 1 ...
##  $ node.capse   : num  0 0 1 0 0 0 0 0 0 0 ...
##  $ no.node.capse: num  1 1 0 1 1 1 1 1 1 1 ...
##  $ breast.left  : num  0 1 0 1 1 0 0 1 1 1 ...
##  $ breast.right : num  1 0 1 0 0 1 1 0 0 0 ...
##  $ quad.cen     : num  1 0 0 0 1 1 0 0 0 0 ...
##  $ quad.Ll      : num  0 0 0 1 0 0 1 0 0 0 ...
##  $ quad.Lu      : num  0 1 1 0 0 0 0 0 1 0 1 ...
##  $ quad.Rl      : num  0 0 0 0 0 0 0 0 0 0 0 ...
##  $ quad.Ru      : num  0 0 0 0 0 0 0 0 0 1 0 ...
```

```
train_num<-round(0.9*nrow(bc2),0)
bc_train<-bc2[1:train_num,]
bc_test<-bc2[(train_num+1):nrow(bc2),]
```

#일반화 선형 모형은 종속변수가 정규분포하지 않는 경우를 포함하는 선형모형의 확장이며 glm()함수를 사용 #family는 종속변수의 분포에 따라 사용 #종속변수의 분포가 정규분포인 경우 gaussian, 이항분포인 경우 binomial, 포아송분포인 경우 poisson, 역정규분포인 경우 inverse.gaussian, 감마분포인 경우 gamma, 응답분포가 확실하지 않은 때를 위한 유사가능도 모형인 경우 quasi를 사용 #p값이 0.05 보다 작은 deg.malig 만 의미있어 보임

```
model <- glm(Class ~ ., data = bc_train, family = binomial)
summary(model)
```

```
##
## Call:
## glm(formula = Class ~ ., family = binomial, data = bc_train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.6241  -0.7673  -0.5386   0.8604   2.4256
##
## Coefficients: (4 not defined because of singularities)
##                 Estimate Std. Error z value Pr(>|z|)
## (Intercept)     -2.35800    1.26332  -1.867   0.0620 .
## age             -0.20057    0.22806  -0.879   0.3791
## tumor.size       0.11470    0.08009   1.432   0.1521
## inv.nodes        0.21295    0.17496   1.217   0.2236
## deg.malig        0.72484    0.25323   2.862   0.0042 **
## irradiat         0.33458    0.37081   0.902   0.3669
## lt40           -15.24082 1162.88078  -0.013   0.9895
## ge40            -0.13369    0.45947  -0.291   0.7711
## premeno               NA         NA      NA       NA
## node.capse       0.29157    0.45916   0.635   0.5254
## no.node.capse         NA         NA      NA       NA
## breast.left      0.28890    0.33724   0.857   0.3916
## breast.right          NA         NA      NA       NA
## quad.cen        -0.44993    0.74767  -0.602   0.5473
## quad.Ll         -0.46319    0.51156  -0.905   0.3652
## quad.Lu         -0.70376    0.49716  -1.416   0.1569
## quad.Rl         -0.77811    0.73979  -1.052   0.2929
## quad.Ru               NA         NA      NA       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 291.98  on 248  degrees of freedom
## Residual deviance: 252.15  on 235  degrees of freedom
## AIC: 280.15
##
## Number of Fisher Scoring iterations: 15
```

#anova 분석에 따르면 tumorsize, invnodes,degmalig가 의미있는 변수

```
anova(model, test="Chisq")
```

```
## Analysis of Deviance Table
##
## Model: binomial, link: logit
##
## Response: Class
##
## Terms added sequentially (first to last)
##
##
##               Df Deviance Resid. Df Resid. Dev Pr(>Chi)
## NULL                            248     291.98
## age            1   4.1366       247     287.85 0.041965 *
## tumor.size     1   8.9390       246     278.91 0.002791 **
## inv.nodes      1  10.5761       245     268.33 0.001146 **
## deg.malig      1   9.5247       244     258.81 0.002027 **
## irradiat       1   1.0219       243     257.78 0.312079
## lt40           1   2.1915       242     255.59 0.138778
## ge40           1   0.1009       241     255.49 0.750778
## premeno        0   0.0000       241     255.49
## node.capse     1   0.4470       240     255.04 0.503786
## no.node.capse  0   0.0000       240     255.04
## breast.left    1   0.6717       239     254.37 0.412446
## breast.right   0   0.0000       239     254.37
## quad.cen       1   0.0146       238     254.36 0.903787
## quad.Ll        1   0.0949       237     254.26 0.758016
## quad.Lu        1   0.9746       236     253.29 0.323540
## quad.Rl        1   1.1430       235     252.15 0.285022
## quad.Ru        0   0.0000       235     252.15
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

type을 response로 지정하고 예측을 수행하면 0에서 1사이의 결과값을 구하

```
pred <- predict(model,bc_test,type="response")
```

```
## Warning in predict.lm(object, newdata, se.fit, scale = 1, type =
## ifelse(type == : prediction from a rank-deficient fit may be misleading
```

```
pred
```

```
##          248          117           76          261          120
## 3.980026e-01 2.291462e-01 5.848324e-08 8.642351e-02 9.823035e-02
##           93           18          223           55          115
## 2.944372e-01 9.300814e-02 2.500325e-01 2.149975e-01 5.797272e-01
##           16           40          140           67          200
## 4.601410e-01 6.648374e-01 3.092631e-01 5.937831e-01 2.111514e-01
##          129          198          175          141          204
## 5.140885e-01 1.167572e-01 4.191263e-01 4.441002e-01 6.836112e-01
##            5          114          201          199            8
## 5.246458e-01 1.079183e-01 3.682641e-01 4.417615e-01 1.724426e-01
##          186          135          206
## 5.382089e-01 7.301522e-01 2.500325e-01
```

```
head(table(pred,bc_test$Class))
```

```
##
## pred                  norecur recur
##   5.84832416469955e-08       1     0
##   0.0864235113697127         1     0
##   0.0930081397179329         1     0
##   0.0982303494045051         1     0
##   0.107918269381323          1     0
##   0.116757163716209          1     0
```

```
glm.pred=ifelse(pred>0.5,"recur","norecur")
table(glm.pred,bc_test$Class)
```

```
##
## glm.pred  norecur recur
##   norecur      13     7
##   recur         2     6
```

#Sensitivity: 0일때 0으로 예측할 확률 #Specificity: 1일때 1로 예측할 확률 #Pos Pred Value : 0으로 예측했는데 0일 확률 #Neg Pred Value : 1로 예측했는데 1로 나올 확률 #Accuracy : 0.75

```
confusionMatrix(table(glm.pred, bc_test$Class))
```

```
## Confusion Matrix and Statistics
##
##
## glm.pred  norecur recur
##   norecur      13     7
##   recur         2     6
##
##                Accuracy : 0.6786
##                  95% CI : (0.4765, 0.8412)
##     No Information Rate : 0.5357
##     P-Value [Acc > NIR] : 0.0913
##
##                   Kappa : 0.3368
##  Mcnemar's Test P-Value : 0.1824
##
##             Sensitivity : 0.8667
##             Specificity : 0.4615
##          Pos Pred Value : 0.6500
##          Neg Pred Value : 0.7500
##              Prevalence : 0.5357
##          Detection Rate : 0.4643
##    Detection Prevalence : 0.7143
##       Balanced Accuracy : 0.6641
##
##        'Positive' Class : norecur
##
```