# 16주차 복습과제
# Solution

1. $H0 : p = 0.5$ $H1 : p \neq 0.5$

Wald test $z = \frac{\left(\frac{450}{1000}\right) - 0.5}{\sqrt{\left(\frac{450}{1000}\right)*\left(\frac{550}{1000}\right)/1000}} = -3.1782$ $z^2 = 10.10$

Likelihood test

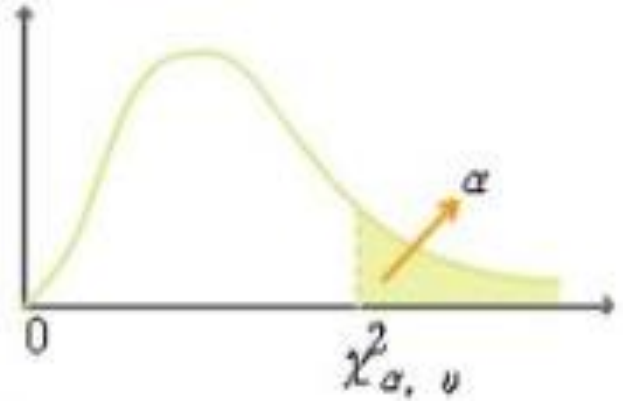
$$l_0 = 1000C450 * \left(\frac{500}{1000}\right)^{450} * \left(\frac{500}{1000}\right)^{550}$$

$$l_1 = 1000C450 * \left(\frac{450}{1000}\right)^{450} * \left(\frac{550}{1000}\right)^{550}$$

$$-2\log\left(\frac{l_0}{l_1}\right) = 4.350$$

**모든 검정에서 귀무가설 기각!**

분포표




| α=.05 | α=.025 | α=.01 | α=.005 | ν |
|---|---|---|---|---|
| 3.841 | 5.024 | 6.635 | 7.879 | 1 |
| 5.991 | 7.378 | 9.210 | 10.597 | 2 |
| 7.815 | 9.348 | 11.345 | 12.838 | 3 |
| 9.488 | 11.143 | 13.277 | 14.860 | 4 |
| 11.070 | 12.832 | 15.086 | 16.750 | 5 |

```
# Wald test
> z=(0.45-0.5)/sqrt((0.45*0.55)/1000)
> z^2
[1] 10.10101
> qchisq(0.05,1,lower.tail = F)
[1] 3.841459
> z^2
>qchisq(0.05,1,lower.tail = F)
[1] TRUE
```

```
# Likelihood test
> l0=(0.5^450)*(0.5^550)
> l1=(0.45^450)*(0.55^550)
> -2*log(l0/l1)
[1] 10.01673
> -2*log(l0/l1)
> qchisq(0.05,1,lower.tail = F)
[1] TRUE
```

2.

```
test=read.csv("test.csv",header=T)
train=read.csv("train.csv",header = T)
train=train[,c(2:10)] ; test=test[,c(2:9)]
train$past_login_total=ifelse(is.na(train$past_login_total),as.integer(median(train$past_login_total,na.rm = T)),train$past_login_total)
train$past_1_month_login=ifelse(is.na(train$past_1_month_login),as.integer(median(train$past_1_month_login,na.rm = T)),train$past_1_month_login)
train$past_1_week_login=ifelse(is.na(train$past_1_week_login),as.integer(median(train$past_1_week_login,na.rm = T)),train$past_1_week_login)


train$Sex=as.factor(train$Sex)
train$login=as.factor(train$login)
train$email_type=as.factor(train$email_type)


fit.full<-glm(login~.,data=train,family = "binomial")
fit.full
```

```
> fit.full
Call: glm(formula = login ~ ., family = "binomial", data = train)

Coefficients:
(Intercept) -3.069135 Sexmale 0.515228 past_login_total 0.008276

past_1_month_login 0.470408 past_1_week_login 0.511578 sub_size 0.012988

email_typehanmail 0.501045 email_typenate -0.736645 email_typenaver -0.018413

email_typeother -0.432805 phone_rat 0.033886 apple_rat -0.016121

Degrees of Freedom: 1499 Total (i.e. Null); 1488 Residual

Null Deviance: 970.8

Residual Deviance: 733.1 AIC: 757.1
```

# 2-1.

```
fit.full<-glm(login~.,data=train,family = "binomial")
fit.null<-glm(login~1,data=train,family = "binomial")
```

```
fit.step<-
step(fit.null,scope=list(upper=fit.full,lower=fit.null),direction="bo
th")
```

# 2-1.

```
> fit

Call: glm(formula = login ~ past_1_month_login + sub_size + Sex +
past_1_week_login + past_login_total, family = "binomial", data = train)

Coefficients:
(Intercept) -3.120739 past_1_month_login 0.475335 sub_size 0.012396
Sexmale 0.520271 past_1_week_login 0.514290 past_login_total 0.007972

Degrees of Freedom: 1499 Total (i.e. Null); 1494 Residual
Null Deviance: 970.8
Residual Deviance: 736.3 AIC: 748.3
```

2-2.

```
test$Sex=as.factor(test$Sex)
test$login=as.factor(test$login)
test$email_type=as.factor(test$email_type)


test$past_login_total = ifelse(is.na(test$past_login_total),as.integer(median(test$past_login_total,na.rm = T)),test$past_login_total)
test$past_1_month_login = ifelse(is.na(test$past_1_month_login),as.integer(median(test$past_1_month_login,na.rm = T)),test$past_1_month_login)
test$past_1_week_login = ifelse(is.na(test$past_1_week_login),as.integer(median(test$past_1_week_login,na.rm = T)),test$past_1_week_login)

predict_test=predict(fit,newdata=test,type="response")
predict_test

test$login=ifelse(predict_test>=0.5,1,0)
test$login
```

```
> head(predict_test)
1 0.04809804
2 0.04290986
3 0.06962288
4 0.04290986
5 0.04665898
6 0.04258366
```

```
> head(test$login)
[1] 0 0 0 0 0 0
```

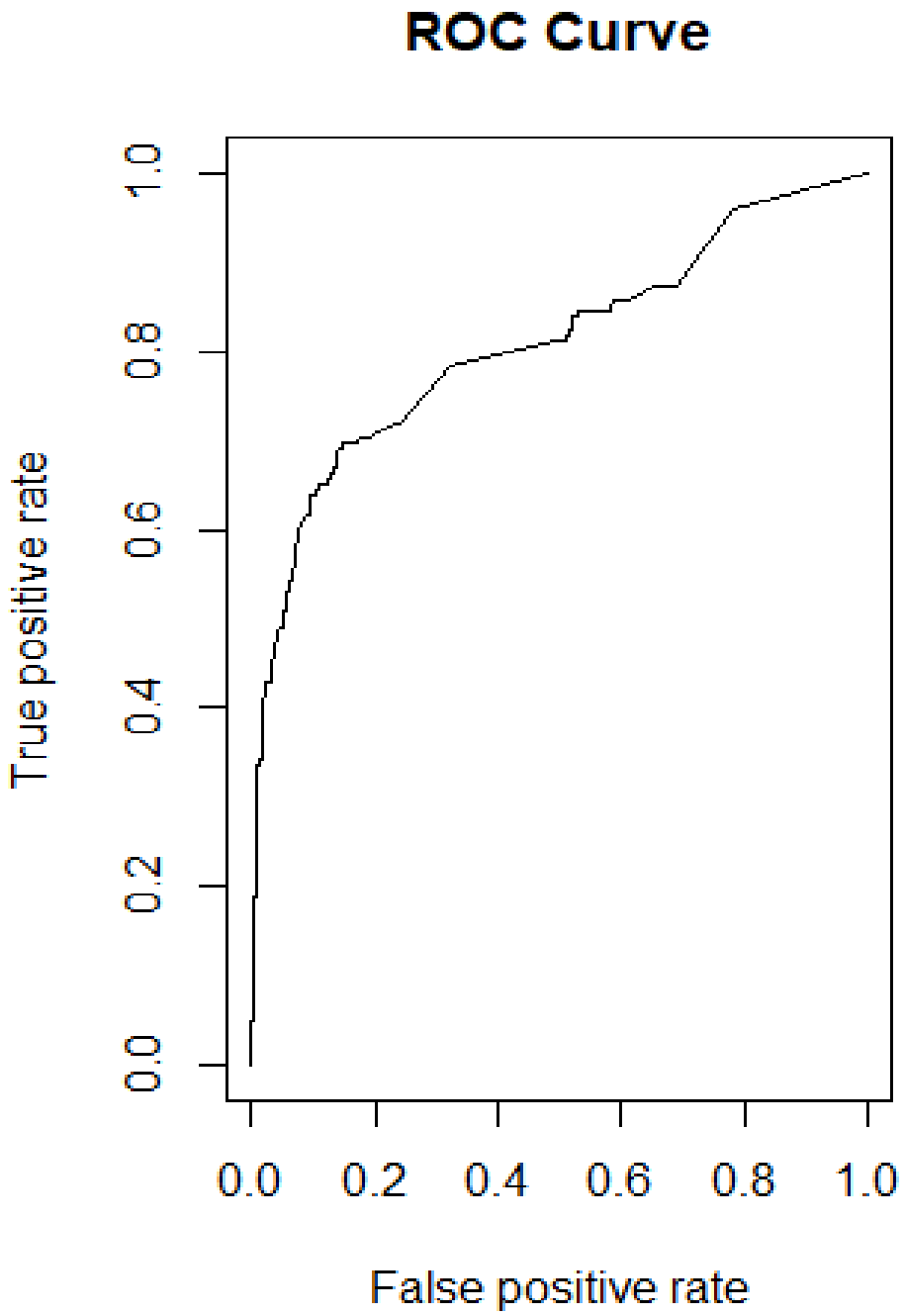
2-3.

```
library("ROCR")

predict_train=predict(fit,newdata=train,type="response")
predictroc=prediction(predict_train , train$login)

perf =
performance(predictroc,measure="tpr", x.measure="fpr")

plot(perf, main="ROC Curve")

auc=performance(predictroc , measure="auc")
auc
```

prediction ( 확률 값, 실제 분류 데이터 )

performance

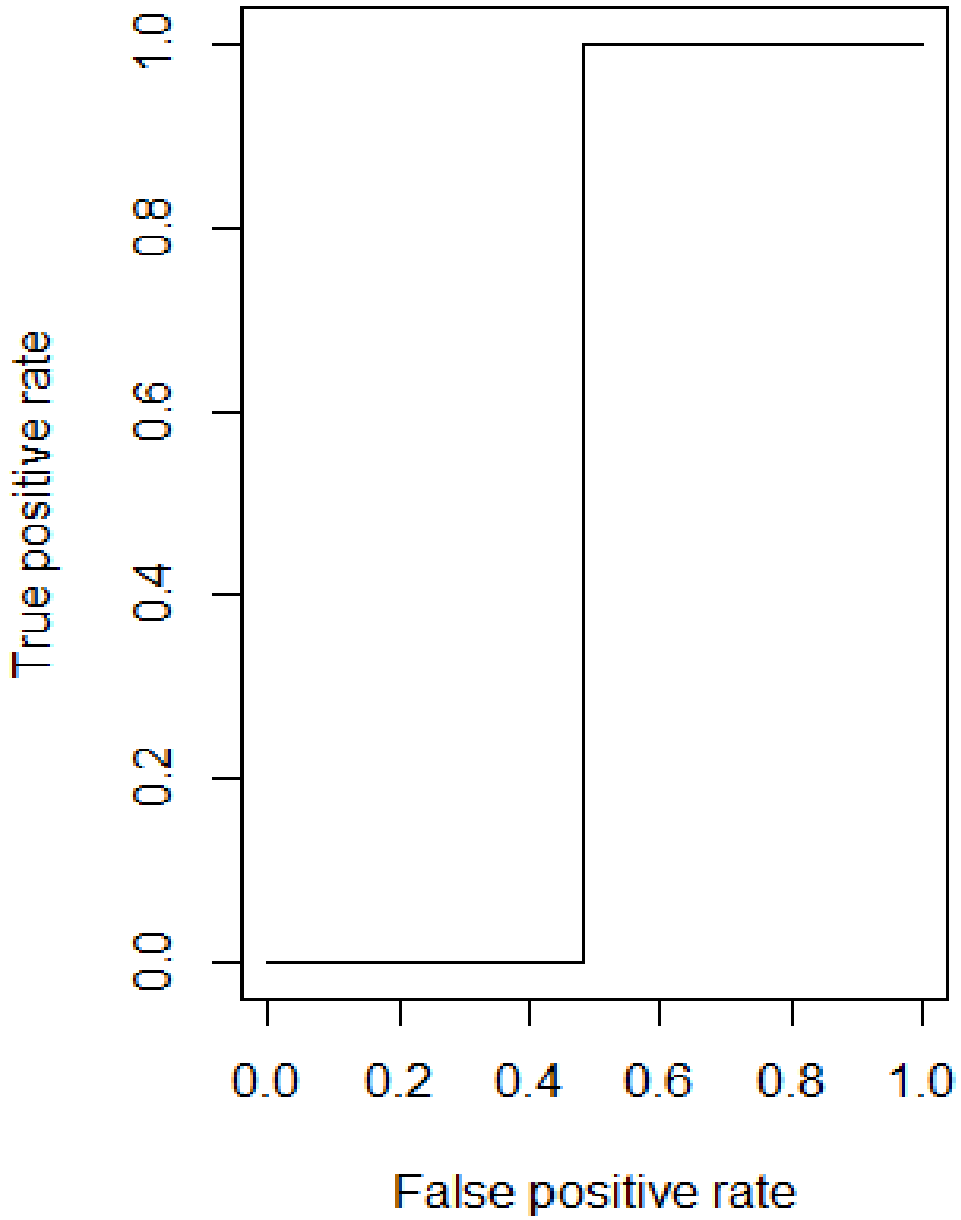
( prediction 데이터 , measure =( "auc" , "tpr"), x.measure = "fpr" ROC CURVE 적합 시 )




```
> 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] 0.8092042

Slot "alpha.values": list()
```

3.

```
library(ROCR)

set.seed(5372)
probs <- runif(100)

label <- as.factor(ifelse(probs <= 0.5  &  probs >= 0.4 , "B", "A"))

predict <- prediction ( probs , label )

plot(performance(predict, "tpr", "fpr"))

auc1 <- performance(predict, "auc")
auc1
```




```
auc1
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] 0.5164835

Slot "alpha.values":
list()
```