**Homework** (8 points, please pay attention to all the words in this orange box)

**Ex 23-1:** If the probability of getting an admission is  for the ith subject, which might be related to his/her gpa and gre scores. Considering the model

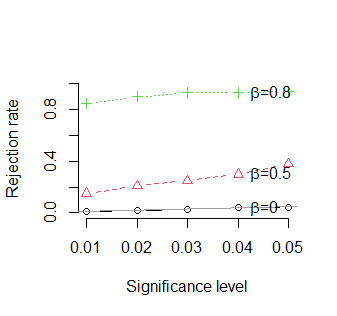


 and . Suppose the sample size is 1000. Number of replications = 100, and seed numbers from 1 to 100, respectively. Please use simulations to evaluate the type I error rate and power of the Wald’s test for , where , respectively. Please make a figure to describe both type I error rate and power.

Hint:

1. Generate gpa and gre scores for n subjects.
2. Given gpa and gre, claculate .
3. sample(c(0,1),1,c(1-,),replace=F)
4. Use the Newton-Raphson method to find MLEs, variance-covariance matrix, and the Wald’s test statistic.

**Please note: Using the R built-in function “**glm(admit~gpa+gre,family=binomial)**” to answer this homework will be scored as 0, although you may use it to check your own answers.**



> rej.rate

[,1] [,2] [,3] [,4] [,5]

[1,] 0.01 0.02 0.03 0.04 0.04

[2,] 0.15 0.21 0.25 0.30 0.38

[3,] 0.85 0.90 0.93 0.93 0.94

Code:

### Ex 23-1:

betagpa <- c(0,0.5,0.8)

sig <- seq(0.01,0.05,0.01)

n <- 1000

Y <- c()

no.rep <- 100

tol = 1e-9

n.max = 100

rej.rate <- matrix(NA,length(betagpa),length(sig))

for(betaloop in 1:length(betagpa)){

pvalue <- c()

for(i in 1:no.rep){

set.seed(i)

gpa <- rnorm(n,3.1,0.3)

gre <- rnorm(n,580,80)

x.beta <- -06+betagpa[betaloop]\*gpa+0.005\*gre

pi.admit <- exp(x.beta)/(1+exp(x.beta))

for(j in 1:n){

Y[j] <- sample(c(0,1),1,c(1-pi.admit[j],pi.admit[j]),replace=F)

}

X <- cbind(rep(1,length(Y)),gpa,gre)

betacoef <- c(0,0,0)

pi1 <- exp(X %\*% betacoef)/(1+exp(X %\*% betacoef))

gradient <- t(X) %\*% (Y-pi1)

hessian <- -t(X) %\*% diag(c(pi1\*(1-pi1)),length(Y)) %\*% X

n.iter <- 0

while ((max(abs(gradient)) > tol) & (n.iter < n.max)) {

betacoef <- betacoef - solve(hessian) %\*% gradient

pi1 <- exp(X %\*% betacoef)/(1+exp(X %\*% betacoef))

gradient <- t(X) %\*% (Y-pi1)

hessian <- -t(X) %\*% diag(c(pi1\*(1-pi1)),length(Y)) %\*% X

n.iter <- n.iter + 1

}

if (n.iter == n.max) {

cat('newton failed to converge\n')

} else {

MLE <- betacoef

}

seMLE <- sqrt(diag(solve(-hessian)))

pvalue[i] <- ((1-pnorm(abs(MLE/seMLE),0,1))\*2)[2]

pvalueglm <- summary(glm(Y~gpa+gre,family=binomial))$coef[2,4]

if(abs(pvalue[i]-pvalueglm) > 1e-4){

cat('Error\n')

break

}

}

for(k in 1:length(sig)){

rej.rate[betaloop,k] <- sum(pvalue<sig[k])/no.rep

}

}

rej.rate

matplot(sig,t(rej.rate),col=c(1:length(betagpa)),pch=c(1:length(betagpa)),lty=c(1:length(betagpa))

,type="b",frame=F,xlab="Significance level",ylab="Rejection rate",ylim=c(0,1))

abline(a=0,b=1,col=8)

legend(0.04,rej.rate[1,4]+0.05,expression(paste(beta,'=0')),bty="n")

legend(0.04,rej.rate[2,4]+0.05,expression(paste(beta,'=0.5')),bty="n")

legend(0.04,rej.rate[3,4]+0.05,expression(paste(beta,'=0.8')),bty="n")