

# Negative Binomial Regression

Jieun Shin

2022-08-26

## 1. 음이항 회귀모형

$y$ 를 음이항 분포를 따르는 계수형 (count) 값이라 하자. 포아송 분포에서는 평균과 분산이 같지만, 음이항 분포에서는 평균이 분산보다 작다고 가정한다. 음이항 모형은 (1)반응변수  $y$ 가 어떤 사건이나 현상에 대한 계수값을 가지고 (2) 음이항 분포를 구성하는 모수를 가지는 일반화 선형모형 (GLM; generalized linear model)이다.  $y$ 의 평균  $\mu$ 와 산포모수  $\alpha > 0$ 를 갖는 음이항 분포는 다음과 같이 정의된다.

$$f(y) = \mathbb{P}(Y = y) = \frac{\Gamma(y + \frac{1}{\alpha})}{\Gamma(y + 1)\Gamma(\frac{1}{\alpha})} \left( \frac{1}{1 + \alpha\mu} \right)^{\frac{1}{\alpha}} \left( \frac{\alpha\mu}{1 + \alpha\mu} \right)^y$$

연결함수 (link function)에 의해  $\ln \mu = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_p X_p$ 로 표현되고 여기서  $X_1, X_2, \cdots, X_p$ 는 독립변수,  $\beta_1, \beta_2, \cdots, \beta_p$ 는 회귀계수이다. 각 변수는  $n$ 개의 관측값  $(x_{1j}, x_{2j}, \cdots, x_{nj})^T$ 을 가지고,  $\boldsymbol{\beta} = (\beta_0, \beta_1, \cdots, \beta_p)^T$ 는 모수벡터라 하자. 그러면 설계행렬 (design matrix)  $\mathbf{X}$ 는

$$\mathbf{X} = \begin{bmatrix} 1 & x_{11} & x_{12} & \cdots & x_{1p} \\ 1 & x_{21} & x_{22} & \cdots & x_{2p} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & x_{n1} & x_{n2} & \cdots & x_{np} \end{bmatrix}$$

와 같이 나타내어진다. 음이항 분포를  $i = 1, 2, \dots, n$ 번째 관측치에 대하여

$$\begin{aligned} f(y_i) = \mathbb{P}(Y = y_i) &= \frac{\Gamma(y_i + \frac{1}{\alpha})}{\Gamma(y_i + 1)\Gamma(\frac{1}{\alpha})} \left( \frac{1}{1 + \alpha\mu_i} \right)^{\frac{1}{\alpha}} \left( \frac{\alpha\mu_i}{1 + \alpha\mu_i} \right)^{y_i} \\ &= \frac{\Gamma(y_i + \frac{1}{\alpha})}{\Gamma(y_i + 1)\Gamma(\frac{1}{\alpha})} \left( \frac{1}{1 + \alpha \exp(X_i \boldsymbol{\beta})} \right)^{\frac{1}{\alpha}} \left( \frac{\alpha \exp(X_i \boldsymbol{\beta})}{1 + \alpha \exp(X_i \boldsymbol{\beta})} \right)^{y_i} \end{aligned}$$

와 같이 정리할 수 있다.

## 2. 로그가능도 (log-likelihood) 함수

음이항 분포의 로그 가능도함수는 다음과 같다.

$$\log L(\alpha, \boldsymbol{\beta}) = \sum_{i=1}^n \left\{ y_i \log \alpha + \left( y_i + \frac{1}{\alpha} \right) \log(1 + \alpha \exp(X_i \boldsymbol{\beta})) + \log \Gamma\left( y_i + \frac{1}{\alpha} \right) - \log(y_i + 1) - \log \Gamma\left( \frac{1}{\alpha} \right) \right\}$$

### 3. 음이항분포의 유도

$\lambda$ 와  $u$ 가 주어졌을 때,  $y$ 의 분포를  $f(y_i; \lambda, u) = \frac{e^{-\lambda_i u_i} (\lambda_i u_i)^{y_i}}{y_i!}$ 라 하자.

감마분포에 의해  $g(u)$ 를 정의하는 방법으로부터  $y$ 의 분포는  $u = \exp(\epsilon)$ 이 된다. 여기서  $\log \mu_i = x_i \beta + \epsilon_i$ 이고 평균은 감마분포의 평균과 같다. 감마분포와  $u$ 가 주어졌을 때의 평균을 갖는 포아송분포의 혼합분포의 평균은  $u$ 가 주어졌을 때  $y$ 의 평균이며 분산은  $u$ 가 주어졌을 때  $y$ 의 분산이다. 포아송-감마 혼합분포는 다음과 같이 정의된다.

$$\begin{aligned} f(y_i; \lambda, u) &= \int_0^\infty \frac{e^{-\lambda_i u_i} (\lambda_i u_i)^{y_i}}{y_i!} \frac{v^v}{\Gamma(v)} u_i^{v-1} e^{-v u_i} du_i \\ &= \frac{\lambda_i^{y_i}}{\Gamma(y_i + 1)} \frac{v^v}{\Gamma(v)} \int_0^\infty e^{-\lambda_i u_i} \cdot u_i^{(y_i+v)-1} du_i \\ &= \frac{\Gamma(y_i + v)}{\Gamma(y_i + 1)\Gamma(v)} \cdot \left( \frac{v}{\lambda_i + v} \right)^v \cdot \left( \frac{\lambda_i}{\lambda_i + v} \right)^{y_i} \\ &= \frac{\Gamma(y_i + v)}{\Gamma(y_i + 1)\Gamma(v)} \cdot \left( \frac{1}{1 + \frac{\lambda_i}{v}} \right)^v \cdot \left( 1 - \frac{1}{1 + \frac{\lambda_i}{v}} \right)^{y_i} \end{aligned}$$

감마분포의 척도모수 (scale parameter)  $\alpha$ 가  $v$ 의 역수 꼴이며, 음이항 분포에서 과산포 (overdispersion)모수 혹은 이질성 (heterogeneity)모수라 부른다. 최종적으로 음이항 분포가 다음과 같이 정의된다.

$$f(y_i; \mu, \alpha) = \frac{\Gamma(y_i + \frac{1}{\alpha})}{\Gamma(y_i + 1)\Gamma(\frac{1}{\alpha})} \cdot \left( \frac{1}{1 + \lambda_i \alpha} \right)^{\frac{1}{\alpha}} \cdot \left( 1 - \frac{1}{1 + \lambda_i \alpha} \right)^{y_i}$$

### 4. 추정방법

$\alpha$ 와  $\beta$ 는 IRLS (iteratively reweighted least square)로 추정한다. 이 방법은 피셔 스코어 함수 (Fisher score function)를 이용하는 방법이며, 이는 1차 미분한 행렬로 선형모형을 추정하기 위해 사용되는 최대 우도 (ML; maximum likelihood) 추정의 일부이다.

지수족 분포는  $f(y_i; \theta, \phi) = \exp \left\{ \frac{y_i \theta_i - b(\theta_i)}{\alpha_i(\phi)} + c(y_i; \phi) \right\}$ 와 같이 나타내어지는데, 여기서  $\theta_i$ 는 정준 모수 (canonical parameter) 또는 연결함수이고,  $b(\theta_i)$ 는 누적량 (cumulant),  $\alpha(\phi)$ 는 척도모수,  $c(y_i; \phi)$ 는 정규화 상수 (normalization term)이다. 지수족 분포의 장점은 특정 분포의  $\theta$ 에 대한 1차, 2차미분으로부터 유일한 (unique)한 평균과 분산을 알 수 있다는 것이다:

$$b'(\theta_i) = \text{mean}, \quad b''(\theta_i) = \text{variance}$$

일반화 선형모형의 pdf는  $f(y_i; \theta, \phi)$ 이고 여기서  $y_i$ 는 반응변수 (response variable),  $\theta_i$ 는 위치모수 (location parameter),  $\phi$ 는 척도모수이다. 로그 가능도함수를  $L(\theta_i, \phi; y_i)$ 라 하면 IRLS는 테일러 전개 (Taylor expansion)를 기반으로 유도된다:

$$0 = f(y_0) + f(y_1 - y_0)f'(y_0) + \frac{(y_1 - y_0)^2}{2!}f''(y_0) + \dots$$

처음 두 번째 항까지만 고려하면,

$$\begin{aligned} 0 &= f(y_0) + f(y_1 - y_0)f'(y_0) \\ \Rightarrow y_1 &= y_0 - \frac{f(y_0)}{f'(y_0)} \end{aligned}$$

의 관계식을 얻는다.

로그 가능도 함수는 최대점 (peak)이 존재한다. ML추정은 그라디언트 (gradient) 혹은 피셔 스코어 (로그 가능도함수의  $\beta$ 에 대한 1차 도함수)를 0으로 놓고 푸는 것이다. 로그 가능도 함수의 2차 도함수 행렬을 정보행렬 (information matrix) 또는 헤시안 행렬 (Hessian matrix)라고 부르며 분산-공분산 행렬이 된다. 분산-공분산 행렬의 대각원소로부터 추정량의 표준오차를 알 수 있다.

IRLS 알고리즘의 설명을 위해 로그 가능도 함수의 1차 도함수 행렬을  $U$ 로, 2차 도함수 행렬을  $H$ 로 표기하자:

$$U = \partial L, \quad H = \partial^2 L$$

뉴턴-랩슨 (Newton-Raphson) 알고리즘을 이용한 모수 추정값은

$$\beta^+ \rightarrow \beta - H^{-1}U$$

를 반복함으로써 얻는다. 로그 가능도 함수는 지수족의 형태로  $L(\theta_i; y_i, \phi) = \sum_{i=1}^n \frac{y_i \theta_i - b(\theta_i)}{a_i(\phi)} + c(y_i; \phi)$ 이고,  $\beta_j$ 에 대하여  $L$ 을 풀기위해 연쇄법칙 (chain rule)을 이용하면

$$\frac{\partial L}{\partial \beta_j} = \sum_{i=1}^n \frac{\partial L}{\partial \theta_i} \frac{\partial \theta_i}{\partial \mu_i} \frac{\partial \mu_i}{\partial \eta_i} \frac{\partial \eta_i}{\partial \beta_j}$$

와 같이 전개할 수 있다. 각 term 을 차례대로 풀면,

$$\frac{\partial L}{\partial \beta_j} = \sum_{i=1}^n \frac{y_i \theta_i - b'(\theta_i)}{a_i(\phi)} + \sum_{i=1}^n \frac{y_i - \mu_i}{a_i(\phi)}$$

이고,  $b'(\theta_i) = \mu_i$ 임을 이용하여 두 번째 term

$$\begin{aligned} \frac{\partial \mu_i}{\partial \theta_i} &= \frac{\partial b'(\theta_i)}{\partial \theta_i} = b''(\theta_i) = V(\mu_i) \\ \Rightarrow \frac{\partial \theta_i}{\partial \mu_i} &= \frac{1}{V(\mu_i)} \end{aligned}$$

을 얻을 수 있다.

그리고  $\eta_i = X_i^T \beta_j$ 이므로 네 번째 term,

$$\frac{\partial \eta_i}{\partial \beta_j} = \frac{\partial (X_i^T \beta_j)}{\partial \beta_j} = X_{ij}$$

을 얻을 수 있다. 그리고 세 번째 term,

$$\frac{\partial \mu_i}{\partial \eta_i} = [g^{-1}(\eta_i)]' = \frac{1}{\frac{\partial \eta_i}{\partial \mu_i}} = \frac{1}{g'(\mu_i)}$$

을 얻는다. 또한  $\mu_i$ 의  $\eta_i$ 에 대한 미분은 연결함수의 역수임을 이용한다:

$$\sum_{i=1}^n \frac{(y_i - \mu_i)x_i}{a_i(\phi)V(\mu_i)g'(\mu_i)} = \sum_{i=1}^n \frac{(y_i - \mu_i)x_i}{a_i(\phi)V(\mu_i)} \left( \frac{\partial \mu_i}{\partial \eta_i} \right) = 0$$

여기서  $y_i$ 는 반응변수,  $\mu_i$ 는 적합된 변수 (fitted variable)이다. 정보행렬을  $I = E \left[ \frac{\partial^2 L}{\partial \beta_j \partial \beta_k} \right] = E \left[ \frac{\partial L}{\partial \beta_j} \frac{\partial L}{\partial \beta_k} \right]$ 라 놓으면 다음과 같이 전개할 수 있다:

$$\begin{aligned}
I &= \frac{\partial}{\partial \beta_j} \left[ \frac{(y_i - \mu_i)x_j}{a_i(\phi)V(\mu_i)} \left( \frac{\partial \mu_i}{\partial \eta_i} \right) \right] \cdot \frac{\partial}{\partial \beta_k} \left[ \frac{(y_i - \mu_i)x_k}{a_i(\phi)V(\mu_i)} \left( \frac{\partial \mu_i}{\partial \eta_i} \right) \right] \\
&= \frac{(y_i - \mu_i)^2 x_j x_k}{\{a_i(\phi)V(\mu_i)\}^2} \left( \frac{\partial \mu_i}{\partial \eta_i} \right)^2
\end{aligned}$$

이 때,  $(y_i - \mu_i)^2 = a_i(\phi)V(\mu_i)$ 이므로  $V(y_i) = a_i(\phi)V(\mu_i) = (y_i - \mu_i)^2$ 라 하면

$$I = \frac{x_j x_k}{V(y_i)} \left( \frac{\partial \mu_i}{\partial \eta_i} \right)^2 = \frac{x_j x_k}{V(y_i)g'^2}$$

이 성립한다. 따라서 뉴턴-랩슨 알고리즘은

$$\beta^+ \leftarrow \beta - \left[ \frac{x_j x_k}{V(y_i)} \left( \frac{\partial \mu_i}{\partial \eta_i} \right)^2 \right]^{-1} \cdot \left[ \frac{(y_i - \mu_i)x_k}{V(y_i)} \left( \frac{\partial \mu_i}{\partial \eta_i} \right) \right]$$

이 된다.

계속해서 역함수 term을 양 변에 곱해보자:

$$\left[ \frac{x_j x_k}{V(y_i)} \left( \frac{\partial \mu_i}{\partial \eta_i} \right)^2 \right] \beta^+ = \left[ \frac{x_j x_k}{V(y_i)} \left( \frac{\partial \mu_i}{\partial \eta_i} \right)^2 \right] \beta + \left[ \frac{(y_i - \mu_i)x_k}{V(y_i)} \left( \frac{\partial \mu_i}{\partial \eta_i} \right) \right]$$

여기서  $W = \frac{1}{V(y_i)} \left( \frac{\partial \mu_i}{\partial \eta_i} \right)^2$ 라 놓으면 선형 예측자 (linear predictor)  $\eta_i = X_i \beta$ 에 대하여 왼쪽 항은

$$\left[ \frac{x_j x_k}{V(y_i)} \left( \frac{\partial \mu_i}{\partial \eta_i} \right)^2 \right] \beta^+ = (X^T W X) \beta^+$$

로 표현되며, 오른 쪽 첫번째 항은

$$\left[ \frac{x_j x_k}{V(y_i)} \left( \frac{\partial \mu_i}{\partial \eta_i} \right)^2 \right] \beta = X^T W \eta_i$$

이 되고,  $W$ 와  $V(y_i)$ 의 정의에 의하여 오른쪽 두 번째 항은

$$\frac{(y_i - \mu_i)x_k}{V(y_i)} \left( \frac{\partial \mu_i}{\partial \eta_i} \right) = \frac{(y_i - \mu_i)x_k}{\frac{1}{W} \left( \frac{\partial \mu_i}{\partial \eta_i} \right)^2} \left( \frac{\partial \mu_i}{\partial \eta_i} \right) = x_k W (y_i - \mu_i) \left( \frac{\partial \mu_i}{\partial \eta_i} \right)$$

으로 표현할 수 있다.

정리하면

$$\begin{aligned}
(X^T W X) \beta^+ &= X^T W \eta_i + x_k W (y_i - \mu_i) \left( \frac{\partial \mu_i}{\partial \eta_i} \right) \\
&= X^T W \eta_i + \frac{(y_i - \mu_i)x_k}{\frac{1}{W} \left( \frac{\partial \mu_i}{\partial \eta_i} \right)^2} \left( \frac{\partial \mu_i}{\partial \eta_i} \right)
\end{aligned}$$

이고,  $z_i = \eta_i + (y_i - \mu_i) \left( \frac{\partial \mu_i}{\partial \eta_i} \right)$ 라 놓으면 최종적으로

$$(X^T W X) \beta^+ = X^T W Z$$

$$\Rightarrow \beta^+ = (X^T W X)^{-1} W Z$$

를 얻는다.

## 5. 예시

### 5-1. 포아송 회귀

포아송 확률변수  $Y_i \sim P(\mu_i)$ 는 평균과 분산이 각각  $\mathbb{E}(Y_i) = \mu_i$ ,  $Var(Y_i) = \mu_i$ 이다. 포아송 회귀를 위한 연결함수  $\eta_i$ , 조정된 반응변수  $z_i$ , 가중치는  $w_i$  다음과 같이 정리된다.

$$\eta_i = \log \mu_i,$$

$$z_i = \eta_i + (y_i - \mu_i) \frac{\partial \mu_i}{\partial \eta_i} = \eta_i + \frac{y_i - \mu_i}{\mu_i},$$

$$w_i = \frac{1}{V(y_i)} \left( \frac{\partial \mu_i}{\partial \eta_i} \right)^2 = \frac{1}{\mu_i} \mu_i^2 = \mu_i$$

### 5-2. 음이항 회귀

음이항 확률변수  $Y_i \sim NB(\mu_i, \phi)$ 는 평균과 분산이 각각  $\mathbb{E}(Y_i) = \mu_i$ ,  $Var(Y_i) = \mu_i + \frac{\mu_i^2}{\phi}$ 이다. 음이항 회귀를 위한 연결함수  $\eta_i$ , 조정된 반응변수  $z_i$ , 가중치는  $w_i$  다음과 같이 정리된다.

$$\eta_i = \log \mu_i,$$

$$z_i = \eta_i + (y_i - \mu_i) \frac{\partial \mu_i}{\partial \eta_i} = \eta_i + \frac{y_i - \mu_i}{\mu_i},$$

$$w_i = \frac{1}{V(y_i)} \left( \frac{\partial \mu_i}{\partial \eta_i} \right)^2 = \frac{1}{\mu_i + \frac{\mu_i^2}{\phi}} \frac{1}{\mu_i^2} = \frac{1}{1 + \mu_i \phi^{-1}}$$

## 6. 시뮬레이션

먼저 계수형 데이터를 다음과 같이 생성한다. 이어서 데이터의 생김새를 파악하자.

```
# 시뮬레이션 데이터 생성
```

```
require(foreign)
```

```
## 필요한 패키지를 로딩중입니다: foreign
```

```
require(ggplot2)
```

```
## 필요한 패키지를 로딩중입니다: ggplot2
```

```
require(MASS)
```

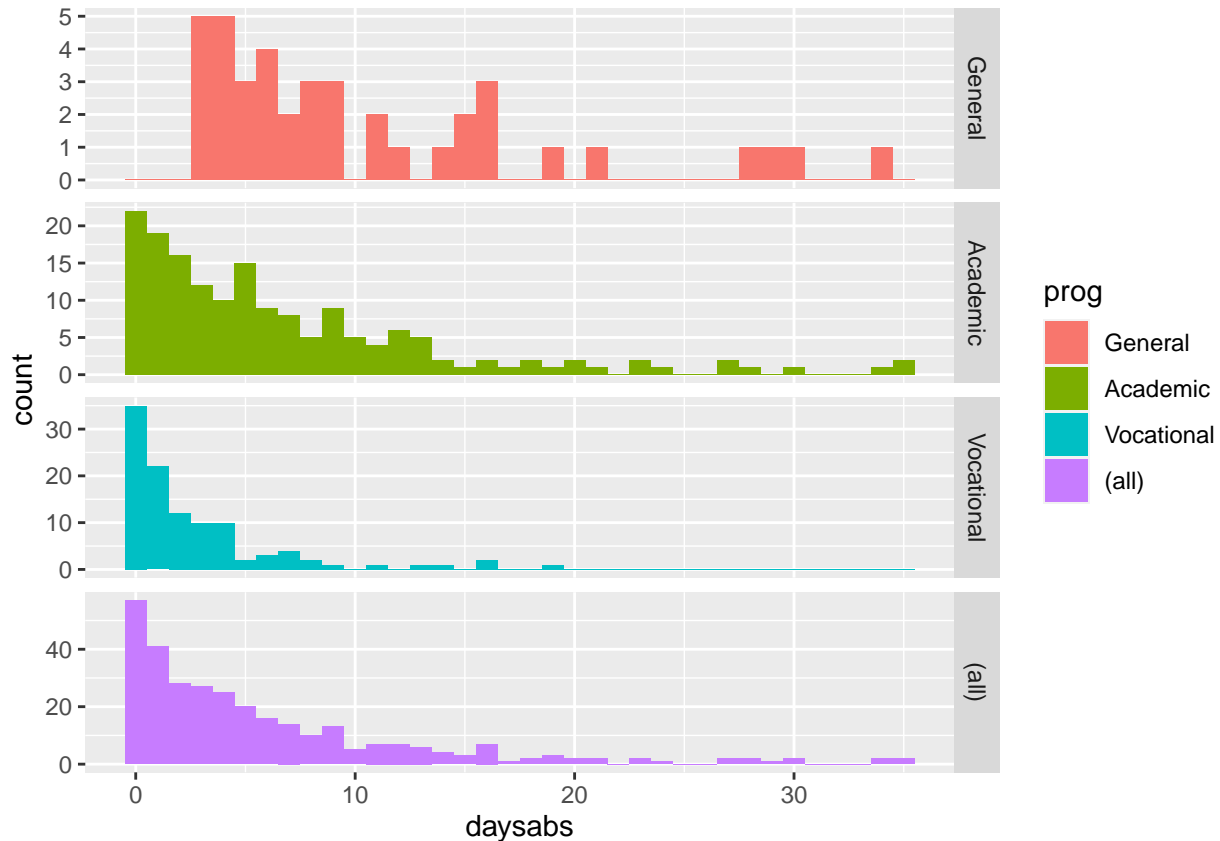
```
dat <- read.dta("https://stats.idre.ucla.edu/stat/stata/dae/nb_data.dta")
```

```
dat <- within(dat, {
  prog <- factor(prog, levels = 1:3, labels = c("General", "Academic", "Vocational"))
  id <- factor(id)
})
```

```
summary(dat)
```

```
##          id          gender          math          daysabs          prog
## 1001      : 1    female:160    Min.   : 1.00    Min.   : 0.000    General   : 40
## 1002      : 1    male :154    1st Qu.:28.00    1st Qu.: 1.000    Academic  :167
## 1003      : 1                                Median :48.00    Median : 4.000    Vocational:107
## 1004      : 1                                Mean   :48.27    Mean   : 5.955
## 1005      : 1                                3rd Qu.:70.00    3rd Qu.: 8.000
## 1006      : 1                                Max.   :99.00    Max.   :35.000
## (Other):308
```

```
ggplot(dat, aes(daysabs, fill = prog)) + geom_histogram(binwidth = 1) + facet_grid(prog ~
., margins = TRUE, scales = "free")
```



```
with(dat, tapply(daysabs, prog, function(x) {
  sprintf("M (SD) = %1.2f (%1.2f)", mean(x), sd(x))
}))
```

```
##          General          Academic          Vocational
## "M (SD) = 10.65 (8.20)" "M (SD) = 6.93 (7.45)" "M (SD) = 2.67 (3.73)"
```

데이터 탐색을 했으면 glm 함수를 통해 음이항 회귀분석의 계수를 뽑는다.

```
# 음이항 회귀분석
```

```
summary(m1 <- glm.nb(daysabs ~ math + prog, data = dat))
```

```
##
```

```
## Call:
```

```
## glm.nb(formula = daysabs ~ math + prog, data = dat, init.theta = 1.032713156,
##       link = log)
```

```
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -2.1547  -1.0192  -0.3694   0.2285   2.5273
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    2.615265   0.197460  13.245 < 2e-16 ***
## math          -0.005993   0.002505  -2.392  0.0167 *
## progAcademic  -0.440760   0.182610  -2.414  0.0158 *
## progVocational -1.278651   0.200720  -6.370 1.89e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for Negative Binomial(1.0327) family taken to be 1)
##
##      Null deviance: 427.54  on 313  degrees of freedom
## Residual deviance: 358.52  on 310  degrees of freedom
## AIC: 1741.3
##
## Number of Fisher Scoring iterations: 1
##
##
##              Theta:  1.033
##              Std. Err.:  0.106
##
## 2 x log-likelihood:  -1731.258
```

다음은 직접 알고리즘을 짜서 그 결과를 비교하고자 한다.

```
# # 데이터 정리
# dat = read.dta("https://stats.idre.ucla.edu/stat/stata/dae/nb_data.dta")
# dat$id = NULL
# dat$progAcademic = ifelse(dat$prog == 1, 1, 0)
# dat$progVocational = ifelse(dat$prog == 2, 1, 0)
# dat$prog = NULL
# dat$gender = NULL
#
# n = nrow(dat)
# p = ncol(dat)
# dat = as.matrix(dat, n, p)
#
# X = cbind(1, dat[, -2])
# y = dat[, 2]
#
# # 함수 지정
# D = function(X, y, phi, mu){
#   n = nrow(X)
#   p = ncol(X) - 1
#   #
#   Dvec = matrix(0, nrow = p+2, ncol = 1)
#   #
#   v1 = rep(0, n)
#   for(i in 1:n){
#     yy = y[i]
```

```

#
#   if(yy == 0){
#       v1[i] = v1[i] + 0
#   }else if (yy > 0){
#       for(v in 0:(yy-1)) v1[i] = v1[i] + 1/(phi + v)
#   }
# }
#
# #
#
# for(i in 1:length(Dvec)){
#
#   if(i == 1) Dvec[i] = sum(phi*(y-mu) / (mu + phi))
#
#   if(i != 1 && i != length(Dvec)) Dvec[i] = sum(phi*X[, i-1]*(y-mu) / (mu + phi))
#
#   if(i == length(Dvec)) Dvec[i] = sum(v1) + sum( log(phi / (phi + mu)) + (y-mu) / (phi + mu) )
# }
#
# return(Dvec)
# }
#
# H = function(X, y, phi, mu){
#   n = nrow(X)
#   p = ncol(X) - 1
#
#   Hmat = matrix(0, nrow = p+2, ncol = p+2)
#
#   #
#   v1 = rep(0, n)
#   for(i in 1:n){
#       yy = y[i]
#
#       if(yy == 0){
#           v1[i] = v1[i] + 0
#       }else if (yy > 0){
#           for(v in 0:(yy-1)) v1[i] = v1[i] - 1/(phi + v)^2
#       }
#   }
#
#   #
#
#   for(i in 2:(nrow(Hmat) - 1)){
#       for(j in 1:ncol(Hmat)){
#           if(i < j && i != 1 && i != nrow(Hmat) && j != nrow(Hmat))
#               Hmat[i, j] = sum( (-phi*mu*X[, i-1]*X[, (j-1)])/(mu + phi) - (-phi*(y-mu)*X[, (i-1)]*X[, (j-1)])
#           )
#       }
#   }
#
#   for(i in 2:(p+1)){
#       Hmat[1, i] = sum( -phi*mu*X[, i]/(mu + phi) - phi*X[, i]*(y-mu)/(mu+phi)^2 )
#   }
#
#

```



```

# Hmat[1, ncol(Hmat)] = sum( (y-mu)/(mu + phi) - phi*(y-mu)/(mu+phi)^2 )
#
# for(j in 2:(p+1)){
#   Hmat[j, ncol(Hmat)] = sum( X[,j]*(y-mu)/(mu + phi) - phi*X[,j]*(y-mu)/(mu+phi)^2 )
# }
#
# for(i in 1:nrow(Hmat)){
#   if(i != nrow(Hmat)) diag(Hmat)[i] = sum( -phi*X[,i]^2*mu/(mu+phi) - phi*X[,i]^2*(y-mu)/(mu+phi)^2 )
#   if(i == nrow(Hmat)) diag(Hmat)[i] = sum(v1) + sum( mu/phi/(phi+mu)^2 - (y-mu)/(phi+mu)^2 )
# }
#
# for(i in 1:nrow(Hmat)){
#   for(j in 1:ncol(Hmat)){
#     if(i > j) Hmat[i,j] = Hmat[j,i]
#   }
# }
#
# return(Hmat)
# }
#
# # 뉴튼랩스 방법
# eps = 1e-3
# max_iter = 1e+2
#
# # initialize
# beta = runif(p)
# mu = exp(X %*% beta)
# phi = runif(1)
# beta_phi = c(beta, phi)
#
# t = 0
# while(t < max_iter && abs(norm(beta_phi - beta_phi_new)) > eps){
#   t = t + 1
#
#   # update
#   if( t != 1){
#     beta_phi = beta_phi_new
#
#     mu = exp(X %*% beta_phi[-5])
#     phi = beta_phi[5]
#   }
#
#   Dvec = D(X, y, phi, mu)
#   Hmat = H(X, y, phi, mu)
#   beta_phi_new = beta_phi - solve(Hmat) %*% Dvec
#
#   mu_new = exp(X %*% beta_phi_new[-5])
#   phi_new = beta_phi_new[5]
#
# }
#
# beta_phi_new
#

```

```

#
# # 음이항 회귀분석
# eps = 1e-3
# max_iter = 1e+2
#
# # initialize
# beta = runif(p)
# mu = exp(X %*% beta)
# phi = runif(1)
# beta_phi = c(beta, phi)
#
# t = 0
# while(t < max_iter && abs(norm(beta_phi - beta_phi_new)) > eps){
#   t = t + 1
#   #
#   # update
#   if( t != 1){
#     beta_phi = beta_phi_new
#   #
#     mu = exp(X %*% beta_phi[-5])
#     phi = beta_phi[5]
#   }
#   #
#   Dvec = D(X, y, phi, mu)
#   Hmat = H(X, y, phi, mu)
#   beta_phi_new = beta_phi - solve(Hmat) %*% Dvec
#   #
#   mu_new = exp(X %*% beta_phi_new[-5])
#   phi_new = beta_phi_new[5]
#   #
# }
#
# beta_phi_new
#
#
#
#

```