R_HW5_Count Regression

Eom SangJun

2020 10 14

이번에는 Response 가 Unbounded Count 인 경우에 대해서 살펴볼 것이다.

Unbounded 라는 조건이 붙은 이유는 bounded 인 경우 앞서 살펴보았던 binomial-type response regression 등이 더 적합할 수도 있기 때문이다.

추가적으로, count 가 충분히 큰 몇몇 경우, normal approximation 이 가능해서 normal linear model 이 사용될 수도 있음을 참고하자.

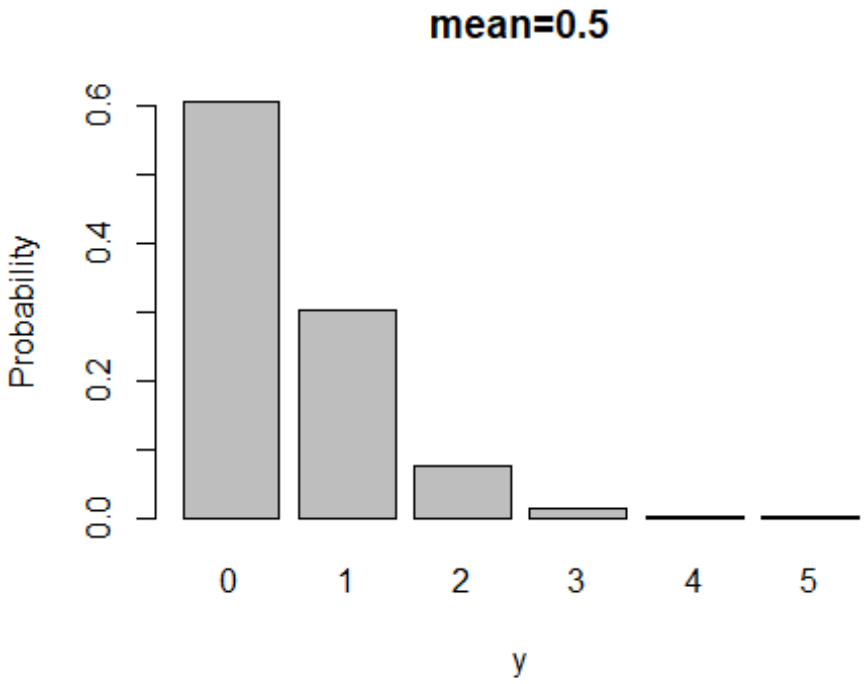
Unbounded Response 인 경우, 주로 사용되는 분포는 Poisson 과 Negative Binomial(덜 일반적)이다.

#1. Poisson Regression

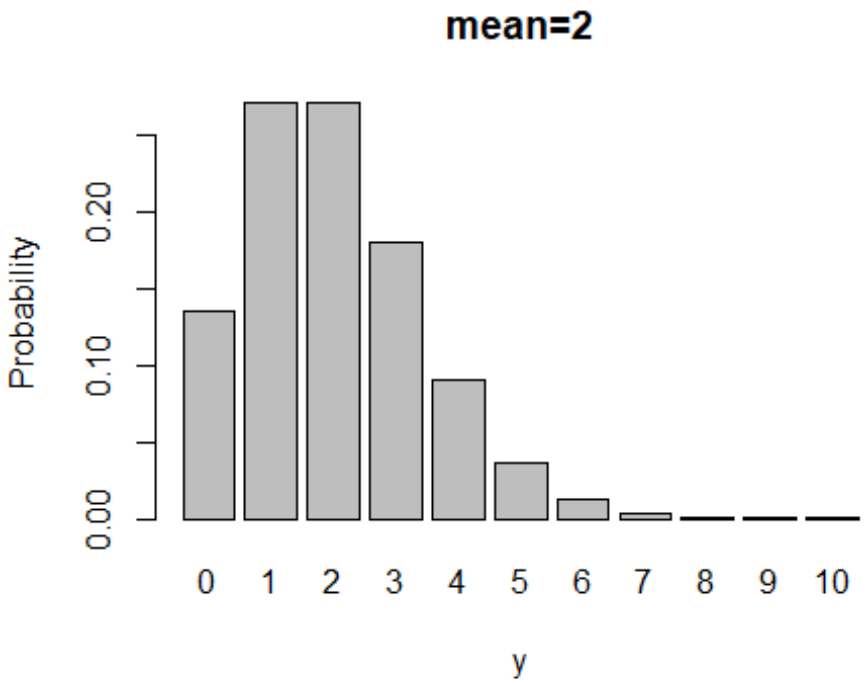
Y 가 Poisson 분포를 따를 때의 pdf 는 다음과 같다.

$$P(Y = y) = \frac{e^{-\mu}\mu^{y}}{y!}, \quad y = 0,1,2, \ldots$$

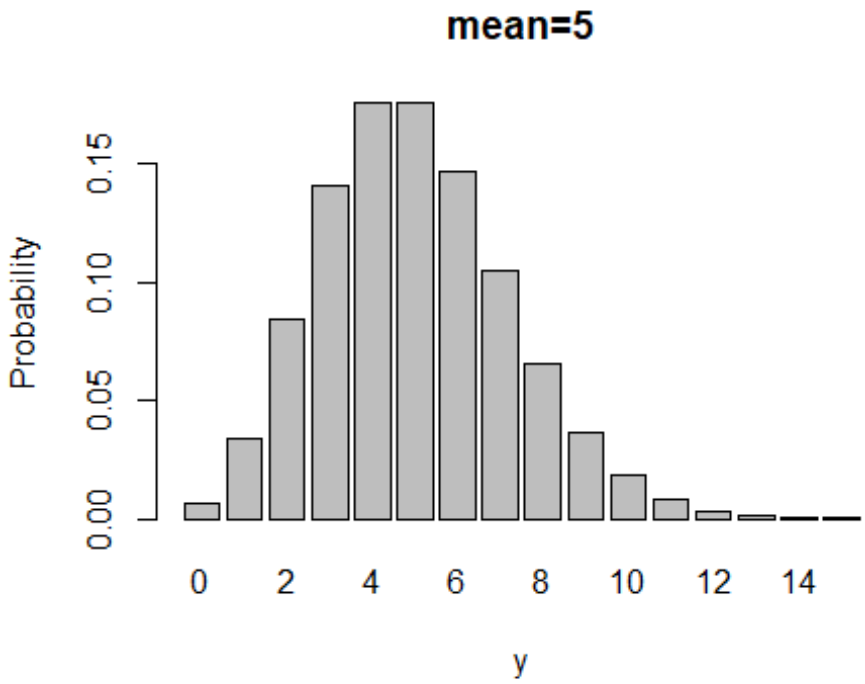
```
barplot(dpois(0:5, 0.5), xlab='y', ylab='Probability', names=0:5,
        main ='mean=0.5')
```




```r
barplot(dpois(0:10, 2), xlab='y', ylab='Probability', names=0:10,
        main ='mean=2')
```



```r
barplot(dpois(0:15, 5), xlab='y', ylab='Probability', names=0:15,
        main ='mean=5')
```



➔ Poission 분포는 Mean 값이 커질수록 정규분포 모양에 가까워짐을 알 수 있다.

Poisson 분포가 자연스럽게 일어나는 경우는 다음과 같다.

1. Large Totals & small success probabilities

만약 total number 도 일정하고 count 도 적당하면, binomial 이 모델로서 더 적합할 수 있다. 그러나 Total 은 엄청 많은데 success probabilities 는 작으면 Poisson 이 더 적당하다.

2. 특정하게 주어진 시간대에서 어떤 사건이 발생할 확률이 시간대의 길이에 비례하며, 다른 사건들의 발생과 독립적이라고 가정하자. 그렇다면 특정한 시간대에서 사건의 발생 수는 포아송 분포를 따른다.

3. 사건 발생 간의 시간이 exponential 분포를 iid 하게 따른다면, 사건의 발생 수는 포아송 분포를 따른다.

만약 사건의 수가 특정 카테고리로 분류된다면(예를 들어, 특정 혈액형인 사람의 수 등) multinomial response model 또는 categorical data analysis 를 사용해야 한다.

Poisson 분포의 중요한 특징 중 하나는 Poisson Random Variable 들을 더해도 Poisson 분포가 나온다는 것이다.

$If,\ \ Y_i \sim Pois(\mu_i)\ \ \ for\ i = 1,2, \ldots$

$\sum_i Y_i \sim Pois(\sum_i \mu_i)$

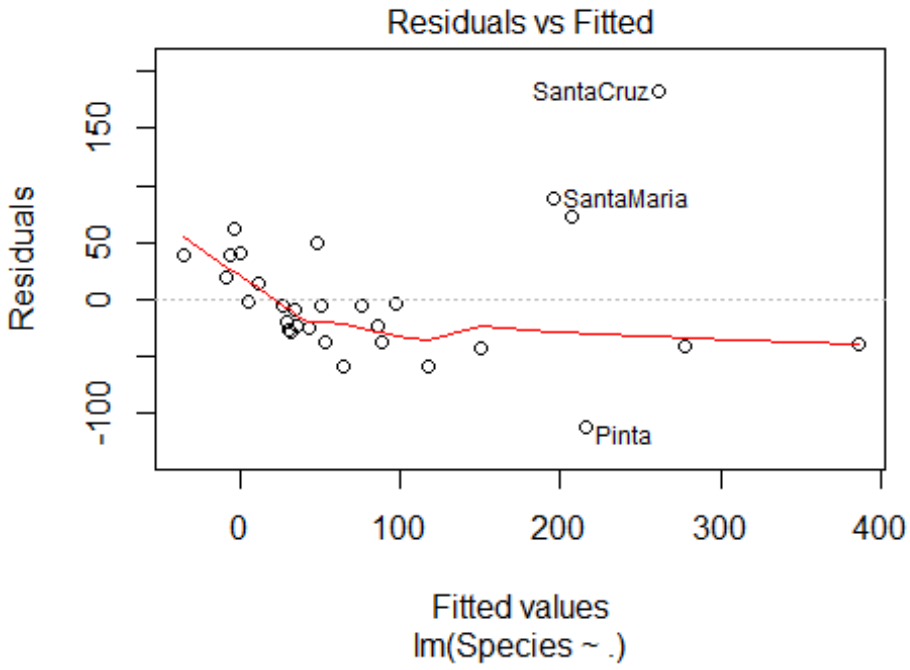
이것이 중요한 이유는 때때로 우리는 aggregated data 만을 data 로 받을 때가 있기 때문이다. 따라서 만약 individual-level data 가 Poisson 을 따른다고 가정하면 우리는 Summed Data 와 Poisson model 을 사용할 수가 있다.

갈라파고스 데이터를 사용하여 이를 살펴보자.

갈라파고스 데이터에서는 갈라파고스의 30 개 섬들을 조사하여 각 섬들의 고유한 식물 종들의 수를 알아내었다.

```
data(gala, package='faraway')
gala <- gala[,-2]
```
→ 2 번 째 변수인 endemic 은 지금은 필요 없으니 빼자.

```
modl <- lm(Species ~ . , gala)
plot(modl, 1)
```



➔ Constant Variance 로 보이지 않는다. 즉, 단순히 종속변수 변환이 필요하다.

➔ Square Root 변환을 해보자.

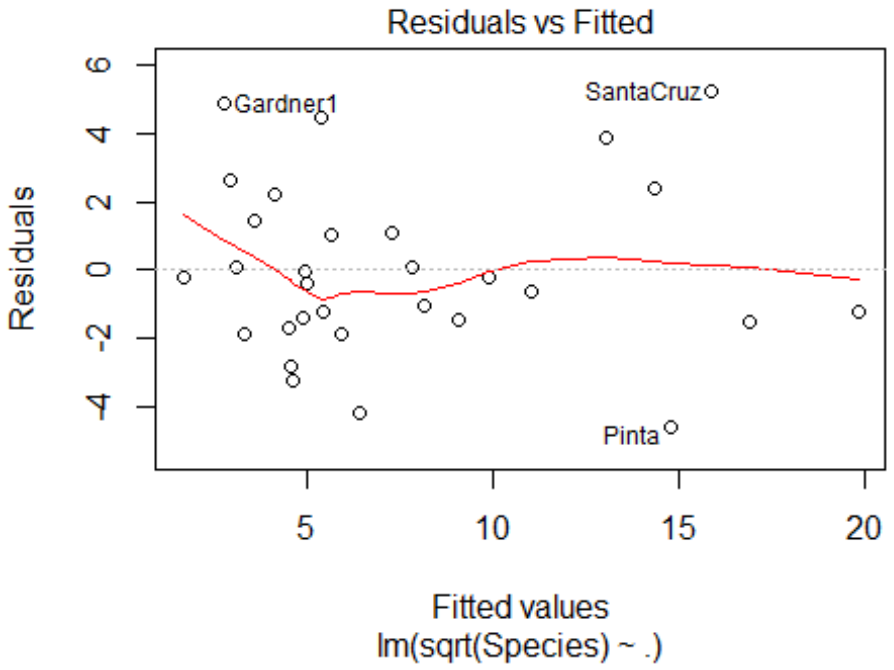
```
modt <- lm(sqrt(Species) ~ ., gala)
plot(modt, 1)

library(faraway)

## Warning: package 'faraway' was built under R version 3.6.3

##
## Attaching package: 'faraway'

## The following object is masked _by_ '.GlobalEnv':
##
##     gala
```



➔ Nonconstant 문제가 해결되었다.

```
sumary(modt)

##                Estimate  Std. Error t value  Pr(>|t|)
## (Intercept)  3.39192432  0.87126781  3.8931 0.0006900
## Area        -0.00197182  0.00101993 -1.9333 0.0650799
## Elevation    0.01647844  0.00244096  6.7508 5.546e-07
```

```
## Nearest      0.02493256  0.04794953  0.5200 0.6078444
## Scruz       -0.01348264  0.00979801 -1.3761 0.1815090
## Adjacent    -0.00336689  0.00080513 -4.1818 0.0003325
##
## n = 30, p = 6, Residual SE = 2.77358, R-Squared = 0.78
```

→ Model 의 R-square 값만 보면 나쁘지 않은 것 같다.

그러나 response 를 변환해서 얻은 값으로 해석이 어렵다는 문제점이 있다.

또한 몇몇 response value 는 너무 작아서 Normal Approximation 의 타당성에 의문이 있다. 따라서 물론 model 이 나쁘지는 않지만 우리는 포아송 모델을 사용하면 더 잘 할 수 있다.

Poisson Model 에서도 Binomial 과 같이 Link Function 을 사용하는데 여기서 Link Function 은 주로 Log 를 사용한다.

즉 Linear Predictor 를 $\eta_i = x_i^T\beta$ 라고 했을 때

$log\mu_i = \eta_i = x_i^T\beta$

따라서 log-likelihood 는

$$l(\beta) = \sum_{i=1}^{n}(y_i x_i^T\beta - \exp(x_i^T\beta) - \log(y_i!))$$

β_j로 미분해서 풀면

$$\sum_{i=1}^{n}(y_i - \exp(x_i^T\beta))x_{ij} = 0 \quad for\ all\ j$$

이를 다시 쓰면

$X^T y = X^T\hat{\mu}$

→ Gaussian Linear Model 과 Binomial Regression with logit link 일 때와 form 이 동일하다.

→ 이런 성질을 가지고 있는 link function 을 canonical link 라고 부른다.

하지만 Poisson 과 Binomial 의 경우 explicit formula for $\hat{\beta}$가 존재하지 않는다. 따라서 solution 을 찾기 위해 numerical 한 방법을 사용해야 한다.

이제 Poisson Model 을 fit 해보자.

```
modp <- glm(Species ~., family=poisson, gala)
sumary(modp)
```

```
##                Estimate  Std. Error  z value  Pr(>|z|)
## (Intercept)  3.1548e+00  5.1750e-02  60.9630 < 2.2e-16
## Area        -5.7994e-04  2.6273e-05 -22.0737 < 2.2e-16
## Elevation    3.5406e-03  8.7407e-05  40.5070 < 2.2e-16
## Nearest      8.8256e-03  1.8213e-03   4.8459 1.261e-06
## Scruz       -5.7094e-03  6.2562e-04  -9.1260 < 2.2e-16
## Adjacent    -6.6303e-04  2.9328e-05 -22.6078 < 2.2e-16
##
## n = 30 p = 6
## Deviance = 716.84577 Null Deviance = 3510.72862 (Difference = 2793.88284)
```

Binomial 때와 마찬가지로 Deviance 가 나온다.

여기서 Deviance 의 값은

$$D = 2\sum_{i=1}^{n}(y_i \log(y_i/\hat{\mu}_i) - (y_i - \hat{\mu}_i))$$

Poisson Deviance 는 G-statistic 이라고도 알려져 있다.

Asymptotic inference 는 binomial 때와 동일하다.

만약 goodness of fit measure 로 다른 방법을 사용하고 싶다면 Pearson's X^2 statistic 도 있다.

$$X^2 = \sum_{i=1}^{n}\frac{(y_i - \hat{\mu}_i)^2}{\hat{\mu}_i}$$

앞의 예시에서 Deviance 는 717 on 24 degrees of freedom 이다. 이는 만약 response 에 대해 Poisson model 이 정말 맞다면 굉장히 안 좋게 fit 되었다는 뜻이다.

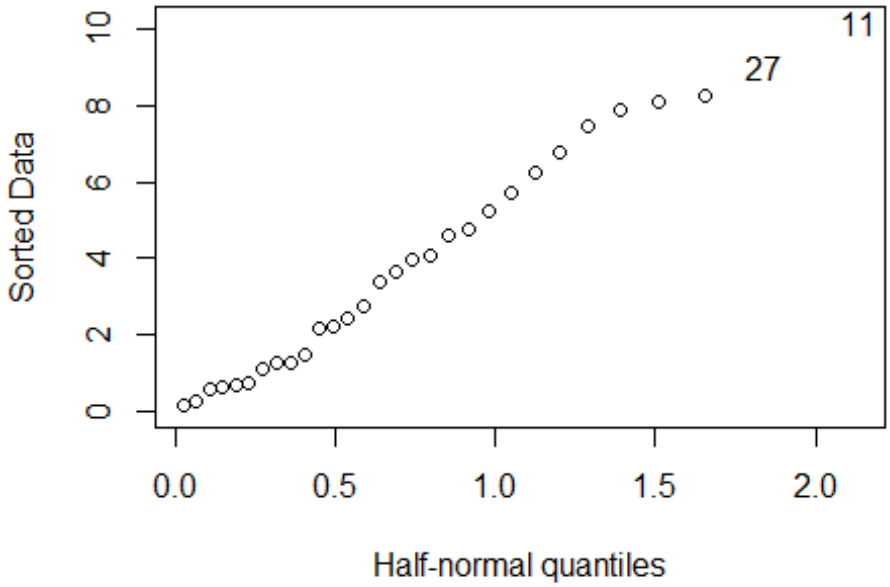
Standard Poisson Model 의 문제점은 Poisson Model 이 암시하는 것보다 종종 response 가 변동이 더 크다는 점이다. Standard linear model 이 mean 과 독립적인 variance parameter 를 가지고 있어서 더 유연한 반면, Poisson 분포는 mean 과 variance 가 동일한 parameter 를 가지고 있어서 덜 유연하다는 단점이 있다.

#2. Dispersed Poisson Model

바로 앞서 언급했던 문제점을 보완하기 위해 우리는 standard Poisson model 을 수정할 수 있다. 다만 그 전에 우리는 Deviance 가 큰 이유가 다른 곳에 있는 것은 아닌지 확인해야 한다.

우선 outlier 를 확인해본다.

```
halfnorm(residuals(modp))
```



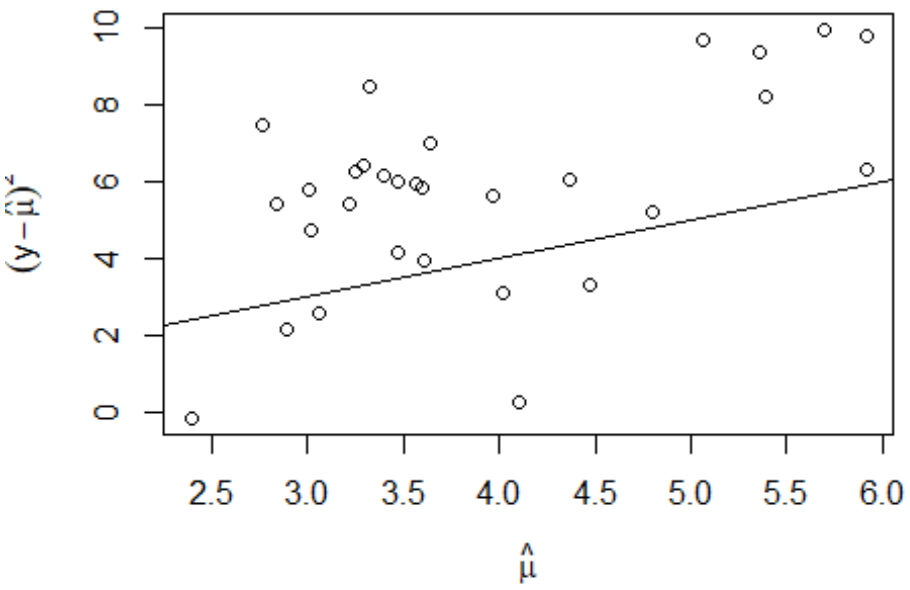
➔ 문제없는 것으로 보인다.

Model 의 구조적 형태가 문제라고 생각할 수 있지만 predictor 들의 form 을 변형했을 때 나타나는 improvement 는 거의 없다. 또한 이 모델을 통해 설명되는 deviance 의 비율은 1-717/3510 = 0.796 으로 linear model 의 것과 거의 동일하다.

원래 Poisson model 에서는 mean 과 variance 가 같아야 한다. 이 모델에서도 실제로 그러한 지를 보자.

우선 우리는 variance 를 $(y-\hat{\mu})^2$를 통해 대략적으로 추정할 수 있다.

```
plot(log(fitted(modp)), log((gala$Species-fitted(modp))^2),
     xlab=expression(hat(mu)), ylab=expression((y-hat(mu))^2))
abline(0,1)
```



➔ 대체적으로 Variance 가 mean 보다 더 크다는 것을 알 수 있다.

만약 link function 과 predictor 의 선택은 옳지만 Poisson regression model 의 variance assumption 은 틀린 경우, β의 추정치는 consistent 하지만, standard error 는 틀릴 것이다. 따라서 어떤 predictor 가 통계적으로 유의한 것인지 우리가 가진 ouput 을 이용해 만든 위의 model 을 통해서는 알 수 없다.

우리는 우리 스스로 dispersion parameter 를 도입함으로써 overdispersion 문제를 해결할 수 있다.

Poisson Model 에서 over 또는 underdispersion 문제는 다양한 방식으로 나타난다. 예를 들어 rate 인 λ가 constant 가 아니라 random variable 일 수 있다. 이 때 우리는 λ가 평균이 μ이고 분산은 μ/ϕ인 gamma 분포를 따른다고 가정할 수 있다.

Y 를 평균이 μ이고 분산은 $\mu(1+\phi)/\phi$인 Negative Binomial 이라고 할 수도 있다.

만약 위의 예시처럼 우리가 특정한 메커니즘을 알고 있다면 response 가 negative binomial 또는 다른 유연한 분포를 따르는 model 로 바꿀 수 있다. 만약 그렇지 않은 경우, 우리는 Poisson model 에 dispersion parameter ϕ를 도입할 수 있다.

$Var(Y) = \phi EY = \phi\mu$

Regular Poisson regression 의 경우에는 $\phi = 1$인 것이다. $\phi > 1$이면 overdispersion, $\phi < 1$이면 underdispersion 이다.

그리고 ϕ는 다음과 같이 추정될 수 있다.

$$\hat{\phi} = \frac{X^2}{n-p} = \sum_i \frac{(y_i - \hat{\mu}_i)^2/\hat{\mu}_i}{n-p}$$

우리의 예시에서 dispersion parameter 를 추정해보자.

```
(dp <- sum(residuals(modp, type='pearson')^2/modp$df.residual))
```

```
## [1] 31.74914
```

이를 통해 우리는 standard error 를 조정해줄 수 있다.

```
sumary(modp, dispersion=dp)
```

```
##                Estimate  Std. Error z value  Pr(>|z|)
## (Intercept)  3.15480788  0.29158975 10.8193 < 2.2e-16
## Area        -0.00057994  0.00014804 -3.9175 8.947e-05
## Elevation    0.00354059  0.00049251  7.1889 6.530e-13
## Nearest      0.00882557  0.01026214  0.8600    0.3898
## Scruz       -0.00570942  0.00352514 -1.6196    0.1053
## Adjacent    -0.00066303  0.00016525 -4.0123 6.013e-05
##
## Dispersion parameter = 31.74914
## n = 30 p = 6
## Deviance = 716.84577 Null Deviance = 3510.72862 (Difference = 2793.88284)
```

→ dispersion 의 추정과 regression parameter estimation 은 독립이다. 따라서 regression parameter 에는 영향이 없는 것을 알 수 있다.

→ Linear Regression Model 과 변수를 선택하는 측면에서 비슷한 점이 있다는 것을 알 수 있다.

애초에 modeling 을 할 때 quasi-Poisson 을 이용하면 dispersion parameter 를 model 에 포함시킬 수 있다.

```
modd <- glm(Species ~ ., family=quasipoisson, gala)
```

Poisson model 들을 비교할 때는 카이제곱 test 가 아니라 F-test 를 사용한다. Normal linear model 에서 분산을 추정했던 것처럼 여기서는 dispersion parameter 를 추정하는데, 이는 F-test 의 사용을 요구한다.

Full model 과 비교했을 때 각 predictor 의 중요성을 test 해보자.

```
drop1(modd, test='F')
```

```
## Single term deletions
##
## Model:
## Species ~ Area + Elevation + Nearest + Scruz + Adjacent
##           Df Deviance F value    Pr(>F)
## <none>         716.85
## Area       1  1204.35 16.3217 0.0004762 ***
## Elevation  1  2389.57 56.0028 1.007e-07 ***
## Nearest    1   739.41  0.7555 0.3933572
## Scruz      1   813.62  3.2400 0.0844448 .
## Adjacent   1  1341.45 20.9119 0.0001230 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

→ 일반적으로 sumary()의 z-statistic 은 F-test 보다 신뢰도가 떨어지니 F-test 를 이용하는 것이 낫다.

#3. Rate Models

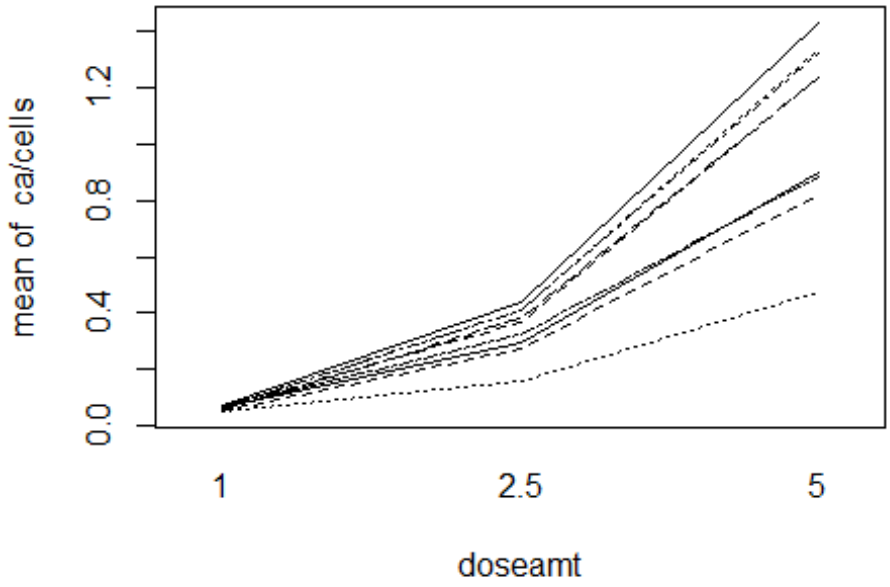
관측되는 사건의 수는 사건이 발생하는 기회를 결정하는 size variable 에 종속적이다. 예를 들어 지역별 도난 범죄의 수를 조사할 때 그 수는 가구의 수에 종속적일 것이다. 대표적인 size variable 로는 시간도 있다. 이러한 경우에는 Rate Model 을 사용하면 된다.

Gamma Radiation 이 chromosomal abnormalities(ca)에 미치는 영향에 관한 데이터를 살펴보자. Ca 는 gamma radiation 에 노출된 cell 의 수가 많을수록 높아질 것이다. 따라서 size variable 이 여기서는 cell 인 것이다. 그렇다면 ca/cells 를 response 로 하고 predictor 인 dosemat 과 doserate 가 interaction 효과가 있는지 살펴보자.

```
round(xtabs(ca/cells ~ doseamt + doserate, dicentric),2)
```

```
##         doserate
## doseamt  0.1 0.25  0.5    1  1.5    2  2.5    3    4
##     1   0.05 0.05 0.07 0.07 0.06 0.07 0.07 0.07 0.07
##     2.5 0.16 0.28 0.29 0.32 0.38 0.41 0.41 0.37 0.44
##     5   0.48 0.82 0.90 0.88 1.23 1.32 1.34 1.24 1.43
```

```
with(dicentric, interaction.plot(doseamt, doserate, ca/cells,
                                 legend=FALSE))
```



➔ Dose rate 의 효과가 multiplicative 일 수 있다.

➔ doserate Variable 에 log 를 취해준다.

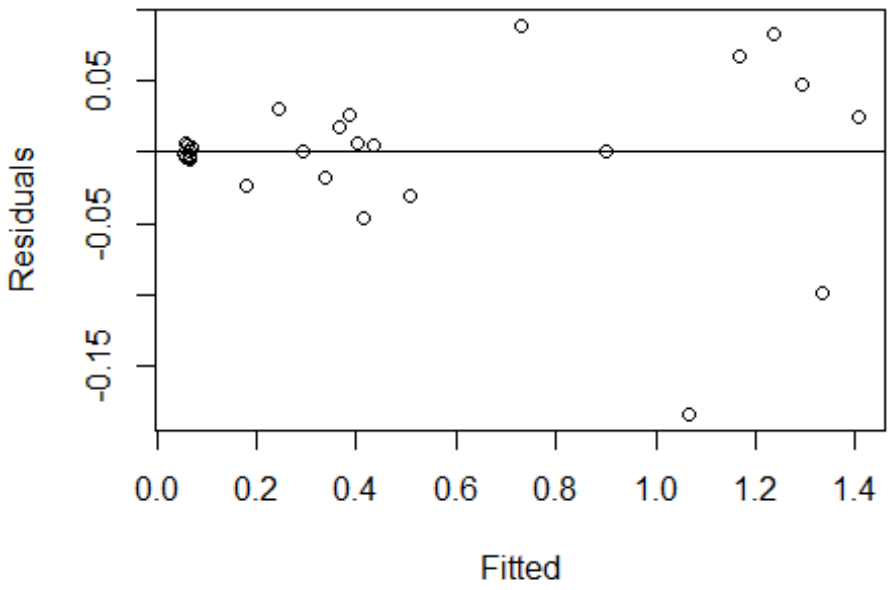
Rate 를 바로 modeling 해보자.

```
lmod <- lm(ca/cells ~ log(doserate)*factor(doseamt), dicentric)
summary(lmod)$adj
```

```
## [1] 0.9844421
```

➔ adjusted R square 가 굉장히 높게 나온 것을 알 수 있다. 그러나 diagnostic 을 보면 문제가 드러난다.

```
plot(residuals(lmod) ~ fitted(lmod), xlab='Fitted', ylab='Residuals')
abline(h=0)
```



➔ 점들이 한 곳(0)에 모여 있다.

따라서 ratio 가 아닌 Count response 를 바로 modeling 하는 것을 생각해보자. Cell predictor 는 Response 에 multiplicative effect 를 줄 것으로 기대되므로 log 를 취해서 predictor 에 포함시키자. 앞서 ratio 모델의 형태를 생각해보면 이는 자연스럽다.

$\log(ca/cells) = X\beta$

$\log(ca) = \log cells + X\beta$

```
dicentric$dosef <- factor(dicentric$doseamt)
pmod <- glm(ca ~ log(cells)+log(doserate)*dosef, family=poisson, dicentric)
sumary(pmod)
```

```
##                          Estimate Std. Error z value  Pr(>|z|)
## (Intercept)            -2.765342   0.381164 -7.2550 4.017e-13
## log(cells)              1.002521   0.051365 19.5175 < 2.2e-16
## log(doserate)           0.071998   0.035475  2.0295 0.0424031
## dosef2.5                1.629839   0.102728 15.8655 < 2.2e-16
## dosef5                  2.766728   0.122872 22.5171 < 2.2e-16
## log(doserate):dosef2.5  0.161108   0.048368  3.3309 0.0008658
## log(doserate):dosef5    0.193163   0.042995  4.4927 7.033e-06
```

```
##
## n = 27 p = 7
## Deviance = 21.74755 Null Deviance = 916.12679 (Difference = 894.37924)
```

→ log cells 의 coefficient 의 값이 1 에 가까운 것을 볼 수 있다. 이는 그냥 coefficient 를 1 로 고정해서 model 을 fit 하는 것과 거의 동일하다. 이러한 방식으로 우리는 Count Response 를 가지는 Poisson Model 을 유지하면서 ca 의 비율을 modeling 할 수 있다. 이를 rate model 이라고 한다.

우리는 offset command 를 이용해서 coefficient 값을 1 로 고정할 수 있다. 이렇게 offset 으로 고정한 predictor 쪽에 있는 term 에는 parameter 가 부여되지 않는다.

```
rmod <- glm(ca ~ offset(log(cells))+log(doserate)*dosef, family=poisson, dice
ntric)
sumary(rmod)
```

```
##                          Estimate Std. Error  z value  Pr(>|z|)
## (Intercept)             -2.746711   0.034263 -80.1649 < 2.2e-16
## log(doserate)            0.071778   0.035176   2.0405 0.0412992
## dosef2.5                 1.625420   0.049460  32.8631 < 2.2e-16
## dosef5                   2.761087   0.043488  63.4905 < 2.2e-16
## log(doserate):dosef2.5   0.161222   0.048302   3.3378 0.0008445
## log(doserate):dosef5     0.193502   0.042427   4.5608 5.096e-06
##
## n = 27 p = 6
## Deviance = 21.74996 Null Deviance = 4753.00404 (Difference = 4731.25408)
```

→ Residual Deviance 를 봤을 때 model 이 잘 fit 되었다는 것을 알 수 있다.

#4. Negative Binomial

일반적인 Binomial 은 trial 의 횟수가 고정되어 있고 성공의 횟수를 센다. 그런데 Negative Binomial 에서는 성공의 횟수가 정해지고 그 때까지의 trial 의 횟수가 response 의 값이다. Z 를 성공확률이 p 일 때 k 번째 성공할 때까지 시행한 횟수라고 하자.

$$P(Z = z) = \binom{z-1}{k-1} p^k (1-p)^{z-k}, \quad for\ z = k, k+1, \dots$$

만약 Y 를 Z 번까지의 실패횟수라고 한다면 오히려 parameterization 이 쉬워진다.

$Y = Z - k, \quad p = (1+\alpha)^{-1}$

이라 하자.

$$P(Y = y) = \binom{y+k-1}{k-1} \frac{\alpha^y}{(1+\alpha)^{y+k}} \quad for\ y = 0,1,2, \dots$$

이 경우 $EY = \mu = k\alpha$ 이고 $Var\, Y = k\alpha + k\alpha^2 = \mu + \mu^2/k$

Log-likelihood 는 다음과 같다.

$$\sum_{i=1}^{n}(y_i \log\frac{\alpha}{1+\alpha} - k\log(1+\alpha) + \sum_{j=0}^{y_i-1}\log(j+k) - \log(y_i!)\ )$$

그리고 가장 mean response 를 linear combination of predictor X 와 link 하는 가장 간편한 방법은 다음과 같다.

$$\eta = x^T\beta = \log\frac{\alpha}{1+\alpha} = \log\frac{\mu}{\mu+k}$$

이 때 k 는 고정되어 있다고 간주할 수도 있고 아니면 추정되어야 할 추가적인 parameter 라고 볼 수도 있다.

이제 예시를 통해 살펴보자.

우리가 사용할 데이터는 납땜에 관련한 데이터로, Response 는 육안검사때까지 납땜을 얼마나 skip 하였는지에 관한 데이터이다.

우선 Poisson Regression 을 해보자.

```
modp <- glm(skips ~ . , family=poisson, data=solder)
c(deviance(modp), df.residual(modp))
```

```
## [1] 1829.002  882.000
```

→ Full model 의 deviance 가 1829 on 882 degrees of freedom 인 것을 알 수 있다. 이는 model 이 잘 fit 되지 않았다는 의미이다.

Interaction 을 반영하지 않은 문제일 수도 있으니 interaction term 을 넣어보자.

```
modp2 <- glm(skips ~ (Opening + Solder + Mask + PadType + Panel)^2,
             family=poisson, data=solder)
deviance(modp2)
```

```
## [1] 1068.817
```

```
pchisq(deviance(modp2), df.residual(modp2), lower=FALSE)
```

```
## [1] 1.130696e-10
```

→ 조금 나아지기는 했지만 그래도 여전히 부족하다.

물론 더 많은 interaction term 을 넣을 수 있지만 해석이 너무 어려워질 위험이 있다.

그럼 이제 Negative Binomial Model 을 사용해보자. 우리는 negative binomial 하고 괄호가운데에 link parameter k 를 지정해줄 수 있다. 우리는 k=1 이라고 가정할 것.

참고로 k=1 인 경우 response 가 geometric distribution 을 따른다는 가정에 해당된다.

```
library(MASS)
modn <- glm(skips ~ . , negative.binomial(1), solder)
modn
```

```
##
## Call:  glm(formula = skips ~ ., family = negative.binomial(1), data = sold
er)
##
## Coefficients:
## (Intercept)     OpeningM     OpeningS   SolderThin       MaskA3       Mask
A6
##    -1.69933      0.50854      1.99966      1.04894      0.65710      2.526
49
##      MaskB3       MaskB6    PadTypeD6    PadTypeD7    PadTypeL4    PadType
L6
##     1.27261      2.08026     -0.46118      0.01608      0.46883     -0.471
15
##   PadTypeL7    PadTypeL8    PadTypeL9    PadTypeW4    PadTypeW9        Pan
el
##    -0.29494     -0.08493     -0.52125     -0.14250     -1.48361      0.169
32
##
## Degrees of Freedom: 899 Total (i.e. Null);  882 Residual
## Null Deviance:       1743
## Residual Deviance: 558.7     AIC: 3884
```

k 값을 일일이 지정하지 않고 그냥 estimated 되게 할 수도 있다. glm.nb 는 negative binomial model 을 사용하는 function 이며 k 를 지정해주지 않는 경우에는 maximum likelihood 에서 자동으로 추정된다.

```
modn <- glm.nb(skips ~ . , solder)
summary(modn)
```

```
##
## Call:
## glm.nb(formula = skips ~ ., data = solder, init.theta = 4.397157245,
##     link = log)
##
## Deviance Residuals:
```

```
##     Min       1Q   Median       3Q      Max
## -2.7376  -1.0068  -0.3834   0.4460   2.7829
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.42245    0.14274  -9.965  < 2e-16 ***
## OpeningM     0.50294    0.07976   6.306 2.87e-10 ***
## OpeningS     1.91317    0.07152  26.750  < 2e-16 ***
## SolderThin   0.93932    0.05362  17.517  < 2e-16 ***
## MaskA3       0.58981    0.09651   6.112 9.87e-10 ***
## MaskA6       2.26734    0.10182  22.269  < 2e-16 ***
## MaskB3       1.21101    0.09637  12.566  < 2e-16 ***
## MaskB6       1.99037    0.09223  21.580  < 2e-16 ***
## PadTypeD6   -0.46592    0.11238  -4.146 3.38e-05 ***
## PadTypeD7   -0.03315    0.10673  -0.311 0.756114
## PadTypeL4    0.38268    0.10265   3.728 0.000193 ***
## PadTypeL6   -0.57844    0.11413  -5.068 4.01e-07 ***
## PadTypeL7   -0.36656    0.11094  -3.304 0.000953 ***
## PadTypeL8   -0.15890    0.10821  -1.468 0.141986
## PadTypeL9   -0.56600    0.11393  -4.968 6.77e-07 ***
## PadTypeW4   -0.20044    0.10873  -1.844 0.065255 .
## PadTypeW9   -1.56460    0.13621 -11.486  < 2e-16 ***
## Panel        0.16369    0.03139   5.214 1.85e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for Negative Binomial(4.3972) family taken to be 1)
##
##     Null deviance: 4043.3  on 899  degrees of freedom
## Residual deviance: 1008.3  on 882  degrees of freedom
## AIC: 3683.3
##
## Number of Fisher Scoring iterations: 1
##
##
##               Theta:  4.397
##           Std. Err.:  0.495
##
##  2 x log-likelihood:  -3645.309
```

→ Theta 에 해당하는 것이 k 의 추정치이다.

Negative Binomial Model 들을 비교하는 방법은 이전에 소개했던 방법들과 동일하다.

#5. Zero Inflated Count Models

일반적으로 Poisson 이나 Negative Binomial 이 예측하는 것보다 사건 발생 수가 0 인 것이 훨씬 많은 경우가 있다. 이는 dispersion parameter 를 추가하는 것으로도 해결하지 못한다.

박사 학위 중인 학생들의 지난 3 년간의 article 발표 수에 관한 데이터로 이를 살펴보자.

```
library(pscl)
```

```
## Warning: package 'pscl' was built under R version 3.6.3
```

```
## 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
```

우선 Poisson 모델을 적용해보자.

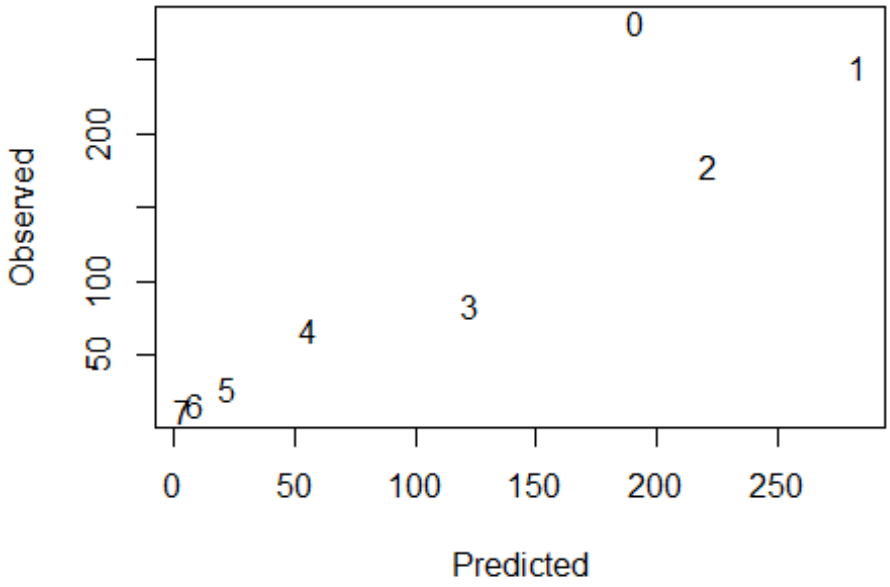
```
modp <- glm(art ~ . , data=bioChemists, family=poisson)
sumary(modp)
```

```
##               Estimate Std. Error z value  Pr(>|z|)
## (Intercept)  0.3046168  0.1029814  2.9580  0.003097
## femWomen    -0.2245942  0.0546135 -4.1124 3.915e-05
## marMarried   0.1552434  0.0613744  2.5294  0.011424
## kid5        -0.1848827  0.0401269 -4.6075 4.076e-06
## phd          0.0128226  0.0263970  0.4858  0.627139
## ment         0.0255427  0.0020061 12.7327 < 2.2e-16
##
## n = 915 p = 6
## Deviance = 1634.37098 Null Deviance = 1817.40530 (Difference = 183.03432)
```

→ Degrees of freedom 에 비해 deviance 가 지나치게 높다. 따라서 문제가 있는 것. 이전에 살펴본 해결방법으로는 이 문제를 해결할 수 없었다.

우리의 모델로 predict 한 값과 observed 값을 비교하는 그래프를 그려보자.

```
ocount <- table(bioChemists$art)[1:8]
pcount <- colSums(predprob(modp)[,1:8])
plot(pcount, as.numeric(ocount), type='n', xlab='Predicted', ylab='Observed')
text(pcount, ocount, 0:7)
```



➔ 다른 값들은 얼추 y=x line 에 있지만 0 이 유독 predict 값보다 observed 값에 많다는 것을 알 수 있다.

이렇게 excess of zero counts 를 modeling 하는 대표적인 방법은 두 가지가 있다.

우선 hurdle model 이다. 이는 latent variable 의 관점에서 생각할 수 있다. 우리가 미처 고려하거나 찾아내지 못한 잠재적인 변수가 있다고 생각해보자. 만약 그 변수의 수치가 어떤 일정한 hurdle 을 넘는다면 Response 가 생성되고(적어도 사건이 하나 발생) 만약 그렇지 못하면 Response 의 값이 0 이 나온다고 생각해보자. 이 때 model 을 다시 세워보면,

$P(Y = 0) = f_1(0)$

$P(Y = j) = \frac{1 - f_1(0)}{1 - f_2(0)} f_2(j), \quad j > 0$

모델에서 첫 번째 부분은 zero 가 관측될 확률이다.

우리는 이러한 확률을 predictor 에 link 시키기 위해 binary response 모델을 사용할 것이다.

두 번째 부분은 outcome 이 zero 보다 클 확률을 의미한다.

이 때 f_2를 위해서는 Poisson 분포를 사용할 건데, 그 중에서도 0 이 허용되지 않으므로 truncated Poisson 을 사용할 것이고 이에 따라 distribution 을 rescale 해주어야 한다. 여기서는 zero 를 hurdle 로 사용하였는데 사실 반드시 0 일 필요는 없다.

이 hurdle model 을 이용해서 fit 을 해보자.

```
modh <- hurdle(art ~ . , data=bioChemists)
summary(modh)
```

```
##
## Call:
## hurdle(formula = art ~ ., data = bioChemists)
##
## Pearson residuals:
##     Min      1Q  Median      3Q     Max
## -2.4105 -0.8913 -0.2817  0.5530  7.0324
##
## Count model coefficients (truncated poisson with log link):
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.67114    0.12246   5.481 4.24e-08 ***
## femWomen    -0.22858    0.06522  -3.505 0.000457 ***
## marMarried   0.09649    0.07283   1.325 0.185209
## kid5        -0.14219    0.04845  -2.934 0.003341 **
## phd         -0.01273    0.03130  -0.407 0.684343
## ment         0.01875    0.00228   8.222  < 2e-16 ***
## Zero hurdle model coefficients (binomial with logit link):
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.23680    0.29552   0.801   0.4230
## femWomen    -0.25115    0.15911  -1.579   0.1144
## marMarried   0.32623    0.18082   1.804   0.0712 .
## kid5        -0.28525    0.11113  -2.567   0.0103 *
## phd          0.02222    0.07956   0.279   0.7800
## ment         0.08012    0.01302   6.155 7.52e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Number of iterations in BFGS optimization: 12
## Log-likelihood: -1605 on 12 Df
```

→ 두 파트로 나뉜 것을 확인할 수 있다.

두 번째 방법은 다음과 같은 아이디어에서 비롯된다. 만약 어떤 사람들에게 지난 한달간 체스를 둔 적이 있냐고 그리고 두었다면 몇 번 두었냐고 질문했다고 해보자. 그럼 이 때 0 이라고 답한 응답자들 중 어떤 사람은 원래 아예 체스를 두지 않는 사람도 있을 것이고 어떤 사람은 체스를 두지만 지난 한 달 동안에만 체스를 두지 않았다고 가정해보자. 우리는 이러한 케이스를 분류하고자 한다.

그리고 이러한 케이스를 분류한 것을 mixture model 이라고 부른다.

Parameter ϕ를 언제나 0 으로 답하는 사람의 비율이라고 해보자. 그럼 이 때 mixture model 은

$P(Y = 0) = \phi + (1 - \phi)f(0)$

$P(Y = j) = (1 - \phi)f(j), \qquad j > 0$

우리는 이러한 비율을 binary response model 을 이용해서 modeling 할 수 있다.

그리고 f 분포는 positive response 를 할 수도 있는 개인들의 응답 수를 modeling 한다.

이 때 f 분포로는 Poisson 분포를 사용하는데, 이 경우에는 zero-inflated Poisson 또는 ZIP model 이라고 불린다.

R 로 살펴보자.

```
modz <- zeroinfl(art ~ . , data=bioChemists)
summary(modz)
```

```
##
## Call:
## zeroinfl(formula = art ~ ., data = bioChemists)
##
## Pearson residuals:
##     Min      1Q  Median      3Q     Max
## -2.3253 -0.8652 -0.2826  0.5404  7.2976
##
## Count model coefficients (poisson with log link):
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.640839   0.121307   5.283 1.27e-07 ***
## femWomen    -0.209144   0.063405  -3.299 0.000972 ***
## marMarried   0.103750   0.071111   1.459 0.144567
## kid5        -0.143320   0.047429  -3.022 0.002513 **
## phd         -0.006166   0.031008  -0.199 0.842376
## ment         0.018098   0.002294   7.888 3.07e-15 ***
##
## Zero-inflation model coefficients (binomial with logit link):
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.577060   0.509386  -1.133  0.25728
## femWomen     0.109752   0.280082   0.392  0.69517
## marMarried  -0.354018   0.317611  -1.115  0.26501
## kid5         0.217095   0.196483   1.105  0.26920
## phd          0.001275   0.145263   0.009  0.99300
## ment        -0.134114   0.045243  -2.964  0.00303 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Number of iterations in BFGS optimization: 19
## Log-likelihood: -1605 on 12 Df
```

→ hurdle model 과 유사한 form 의 결과를 보여준다.

→ Zero Part 를 비교해보자. 두 가지 approach 모두 ment variable 이 통계적으로 유의하다고 나오는데, sign 은 반대라는 것을 알 수 있다.

이는 hurdle model 은 positive count 가 나올 확률을 계산하는 반면,

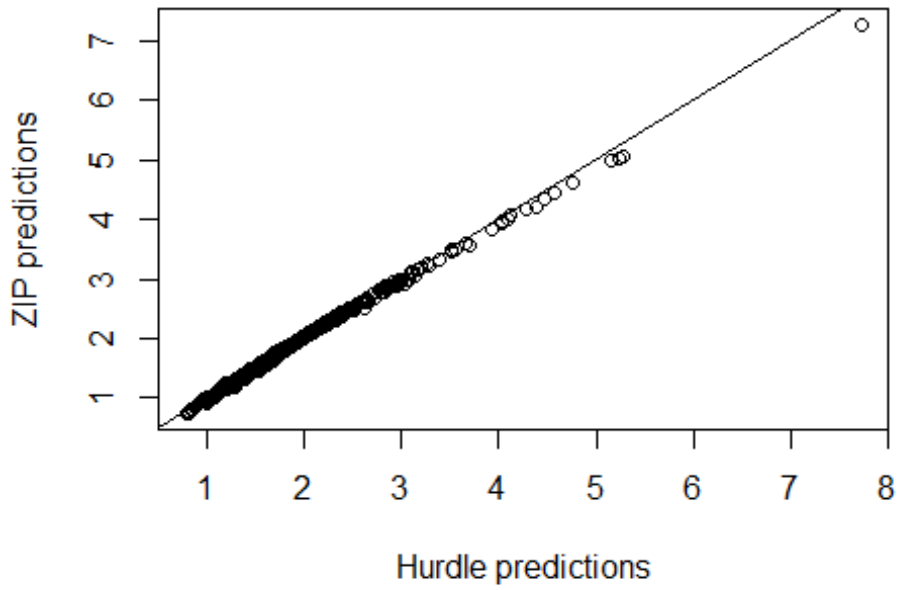
Zero-inflated approach model 은 zero count 의 확률을 계산하기 때문이다.

여기서는 ment 의 수치가 높을수록 0 이 나올 확률이 줄어든다는 것을 알 수 있는 것.

그럼 우리는 어떤 approach 를 써야할까?

우선 fitted value 를 비교해보자.

```
plot(fitted(modh), fitted(modz), xlab='Hurdle predictions', ylab='ZIP predict
ions')
abline(0,1)
```



➔ 거의 동일하다.

➔ 선택을 위해 우리의 사전 지식을 이용할 수도 있다.

nested model 을 비교하기 위해 우리는 standard likelihood testing theory 를 이용할 수 있다. 예를 들어 ZIP model 에서 count part 와 zero part 의 predictor 가 다른 경우를 고려해보자.

R 에서 equation 에서 |를 전후로 앞에는 count part, 뒤에는 zero part 를 지정해줄 수 있다.

```
modz2 <- zeroinfl(art ~ fem+kid5+ment | ment, data=bioChemists)
summary(modz2)
```

```
##
## Call:
## zeroinfl(formula = art ~ fem + kid5 + ment | ment, data = bioChemists)
##
## Pearson residuals:
##     Min      1Q  Median      3Q     Max
## -2.2802 -0.8807 -0.2718  0.5131  7.4788
##
## Count model coefficients (poisson with log link):
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.694517   0.053025  13.098  < 2e-16 ***
## femWomen    -0.233857   0.058400  -4.004 6.22e-05 ***
## kid5        -0.126516   0.039668  -3.189  0.00143 **
## ment         0.018004   0.002224   8.096 5.67e-16 ***
##
## Zero-inflation model coefficients (binomial with logit link):
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.68488    0.20529  -3.336 0.000849 ***
## ment        -0.12680    0.03981  -3.185 0.001448 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Number of iterations in BFGS optimization: 22
## Log-likelihood: -1608 on 6 Df
```

loglikelihood 값의 차이에 두 배를 해주면 그 값은 근사적으로 카이제곱 분포를 따르고 이 때 자유도는 두 모델의 parameter 수의 차이이다. 이전 모델은 12 개, 간결화 된 모델은 6 개 따라서 6 이다.

```
(lrt <- 2*(modz$loglik-modz2$loglik))
```

```
## [1] 6.172789
```

```
1-pchisq(6.1728,6)
```

```
## [1] 0.4041141
```

→ p-value 가 0.4 이상이므로 간결화 된 모델이 타당하다고 볼 수 있다.

predictor 해석을 위해서 coefficient 값에 exponential 을 취해주자.

```
exp(coef(modz2))
```

```
## count_(Intercept)    count_femWomen        count_kid5        count_ment
##         2.0027411         0.7914748         0.8811604         1.0181669
##  zero_(Intercept)         zero_ment
##         0.5041522         0.8809081
```

→ 여자인경우 남자인 경우보다 0.79 배로 article 을 쓰며 mentor production 이 하나 올라가면 1.8%만큼 additional article 이 product 된다.

Zero 측면에서 보면 each extra article from mentor 가 nonproductive student 의 odds 를 0.88 만큼 감소시킨다.

이제 예측을 해보자.

```
newman <- data.frame(fem='Men', mar='Single', kid5=0, ment=6)
predict(modz2, newdata = newman, type='prob')
```

```
##           0         1       2         3          4          5          6
## 1 0.2775879 0.1939403 0.21636 0.1609142 0.08975799 0.04005363 0.01489462
##             7           8            9           10           11
 12
## 1 0.004747556 0.001324094 0.0003282578 7.324092e-05 1.485593e-05 2.762214e
-06
##             13           14           15           16          17
 18
## 1 4.740812e-07 7.555503e-08 1.123857e-08 1.567219e-09 2.05693e-10 2.54968e
-11
##             19
## 1 2.994131e-12
```

→ article 을 하나도 쓰지 않을 확률이 제일 높다.

Zero part 의 관점에서 no production 의 확률을 계산해보자.

```
predict(modz2, newdata = newman, type='zero')
```

```
##        1
## 0.190666
```

→ 아까 전체 part 로 보았을 때는 0.278 이었다. 따라서 0.279-0.191 만큼은 Poisson count part 에서 왔다는 것을 알 수 있다. 이러한 차이는 학생이 article 을 원래는 쓰는데 이번에만 쓰지 않았을 가능성에서 비롯된 수치라는 것을 알 수 있다.

추가숙제

Binomial Distribution

- Likelihood & Notation 정리.

Y가 이항분포를 따른다고 가정했을 때 (성공확률을 θ라 하자)

$f_Y(y) = \binom{n}{y} \theta^y (1-\theta)^{n-y}$ 이다.

∴ log-likelihood의 형태는

$l = \sum_{i=1}^{n} \{ y_i \log\left(\frac{\theta_i}{1-\theta_i}\right) - n_i \log\left(\frac{1}{1-\theta_i}\right) \} + \text{Constant}$ 이다.

∴ 앞선 Notation에 따르면

$r_i = \log\left(\frac{\theta_1}{1-\theta_i}\right)$, $\theta_i = \frac{\exp(r_i)}{1+\exp(r_i)}$

$b(r_i) = n_i \log(1+\exp(r_i))$

$\tau^2 = 1$

이라고 볼 수 있다.

- $E[y_i]$, $V(\mu_i)$, $g_\mu(\mu_i)$

$b'(r_i) = \frac{n_i}{1+\exp(r_i)} \times (\exp(r_i)) = n_i \theta_i = \mu_i \overset{let}{\equiv} E[y_i]$

$V(\mu_i) = b''(r_i) = \frac{n_i \exp(r_i)}{(1+\exp(r_i))^2} = n_i \theta_i (1-\theta_i)$

$g(\theta_i) = \log \frac{\theta_i}{1-\theta_i}$ 로 두면 $g'(\theta_i) = \frac{1}{\theta_i(1-\theta_i)} = \frac{n_i}{V(\mu_i)} \overset{let}{=} g_\mu(\mu_i)$

- W , Δ

y의 분포가 exponential family 일 때 $\frac{\partial l}{\partial \beta} = \frac{1}{\tau^2} \sum_{i=1}^{n} (y_i - \mu_i) w_i g_\mu(\mu_i) x_i$ 라고 알려져 있다.

이 때 $w_i = \frac{1}{V(\mu_i) g_\mu^2(\mu_i)}$ 앞서 $g_\mu(\mu_i) = \frac{n_i}{V(\mu_i)}$ ∴ 이항분포 때 $w_i = \frac{\theta_i(1-\theta_i)}{n_i}$

또한 $\frac{\partial \ell}{\partial \beta} = \frac{1}{\tau^2} X^T W \Delta (y-\mu)$

$$X = \begin{pmatrix} x_1^T \\ \vdots \\ x_N^T \end{pmatrix} \quad W = \begin{pmatrix} w_1 & & 0 \\ & w_2 & \\ 0 & & \ddots \\ & & & w_N \end{pmatrix} \quad \Delta = \begin{pmatrix} g_\mu(\mu_1) & & 0 \\ & \ddots & \\ 0 & & g_\mu(\mu_N) \end{pmatrix}$$

x_i^T : i-th row vector

앞서 w_i와 $g_\mu(\mu_i)$를 구했다.

$w_i = \frac{\theta_i(1-\theta_i)}{n_i}$

$g_\mu(\mu_i) = \frac{1}{\theta_i(1-\theta_i)}$

$\therefore \quad W\Delta = \mathrm{Diag}\left(\frac{1}{n_i}\right)$

$\therefore \quad \frac{\partial \ell}{\partial \beta} = \frac{1}{\tau^2} \times \frac{1}{n_i} X^T(y-\mu) = \frac{1}{n_i} X^T(y-\mu) \quad (\because \tau^2 = 1)$

- Maximum Likelihood Equation

이제 MLE를 구하기 위해 $\frac{\partial \ell}{\partial \beta} = 0$ 으로 두면

$X^T(y-\mu) = 0$

$X^T y = X^T \mu$

- Fisher Information Matrix

$-E\left[\frac{\partial^2 \ell(\beta)}{\partial \beta^2}\right] = \frac{1}{\tau^2} X^T W X = X^T W X$

where $W = \begin{pmatrix} w_1 & & 0 \\ & \ddots & \\ 0 & & w_N \end{pmatrix} = \begin{pmatrix} V(\mu_1) & & 0 \\ & \ddots & \\ 0 & & V(\mu_N) \end{pmatrix} = \begin{pmatrix} n_1\theta_1(1-\theta_1) & & 0 \\ & \ddots & \\ 0 & & n_N\theta_N(1-\theta_N) \end{pmatrix}$

$\therefore \quad X^T W X = \sum_{i=1}^{n} n_i \theta_i (1-\theta_i) x_i x_i^T$

Poisson

- Likelihood & Notation

$Y \sim \text{Poisson}(\theta)$ 라고 하면

$f_Y(y) = \frac{\theta^y e^{-\theta}}{y!}$ $\quad \exp\{y\log\theta - \theta - \log y!\}$

$\therefore$ log-likelihood 의 형태는

$\ell = \sum_{i=1}^{n}\{y_i\log\theta_i - \theta_i - \log y_i!\}$

$\therefore \gamma_i = \log\theta_i$

$b(\gamma_i) = \theta_i = \exp(\gamma_i)$

$\tau^2 = 1$

- $E[y]$, $V(\mu_i)$, $g_m(\mu_i)$

$b'(\gamma_i) = \exp(\gamma_i) = \theta_i = \mu_i \overset{\text{def}}{=} E[y_i]$

$b''(\gamma_i) = \exp(\gamma_i) = \theta_i = V(\mu_i)$

$g(\mu_i) = \gamma_i = \log\theta_i = \log\mu_i$

$\therefore$ link function = log

$\frac{\partial g(\mu_i)}{\partial \mu_i} = \frac{\partial \log\theta_i}{\partial \theta_i} = \frac{1}{\theta_i} = \frac{1}{V(\mu_i)} = g_m(\mu_i)$

- W, Δ

$w_i = \frac{1}{V(\mu_i)g_m^2(\mu_i)}$ 앞서 $g_m(\mu_i) = \frac{1}{V(\mu_i)}$ $\therefore$ $w_i = \frac{\{V(\mu_i)\}^2}{V(\mu_i)} = V(\mu_i)$

$$\therefore W = \begin{pmatrix} V(\mu_1) & & 0 \\ & \ddots & \\ 0 & & V(\mu_N) \end{pmatrix} = \begin{pmatrix} \theta_1 & & 0 \\ & \ddots & \\ 0 & & \theta_N \end{pmatrix}$$

$$\Delta = \begin{pmatrix} g_N(\mu_1) & & 0 \\ & \ddots & \\ 0 & & g_N(\mu_N) \end{pmatrix} = \begin{pmatrix} \frac{1}{V(\mu_1)} & & 0 \\ & \ddots & \\ 0 & & \frac{1}{V(\mu_N)} \end{pmatrix} = \begin{pmatrix} \frac{1}{\theta_1} & & 0 \\ & \ddots & \\ 0 & & \frac{1}{\theta_N} \end{pmatrix}$$

$\therefore W\Delta = I_N$

$\therefore \frac{\partial l}{\partial \beta} = X^T(y-\mu)$

- Maximum Likelihood Equation

let $\frac{\partial l}{\partial \beta} = X^T(y-\mu) = 0$

$X^T y = X^T \mu$

where $y = \begin{pmatrix} y_1 \\ \vdots \\ y_N \end{pmatrix}$ $\mu = \begin{pmatrix} \mu_1 \\ \vdots \\ \mu_N \end{pmatrix} = \begin{pmatrix} \theta_1 \\ \vdots \\ \theta_N \end{pmatrix}$

- Fisher Information Matrix

$$-E\left[\frac{\partial^2 l(\beta)}{\partial \beta^2}\right] = \frac{1}{\tau^2} X^T W X = X^T W X$$

where $W = \begin{pmatrix} \theta_1 & & 0 \\ & \ddots & \\ 0 & & \theta_N \end{pmatrix}$

$\therefore X^T W X = \sum_{i=1}^{n} \theta_i x_i x_i^T$