Biostat 200C Homework 2

Due Apr 26 @ 11:59PM

Q1. Beta-Binomial

Let Y_i be the number of successes in n_i trials with

$$Y_i \sim \text{Bin}(n_i, \pi_i),$$

where the probabilities π_i have a Beta distribution

$$\pi \sim \text{Be}(\alpha, \beta)$$

with density function

$$f(x; \alpha, \beta) = \frac{\Gamma(\alpha + \beta)}{\Gamma(\alpha)\Gamma(\beta)} x^{\alpha-1}(1-x)^{\beta-1}, \quad x \in [0,1], \alpha > 0, \beta > 0.$$

1.1

Find the mean and variance of π.

1.2

Find the mean and variance of Y_i and show that the variance of Y_i is always larger than or equal to that of a Binomial random variable with the same batch size and mean.

Q2. Poisson regression log-likelihood

Let $Y_1, \ldots, Y_n$ be independent random variables with $Y_i \sim \text{Poisson}(\mu_i)$ and $\log \mu_i = \mathbf{x}_i^T \boldsymbol{\beta}$, $i = 1, \ldots, n$.

2.1

Write down the log-likelihood function.

2.2

Derive the gradient vector of the log-likelihood function with respect to the regression coefficients $\boldsymbol{\beta}$, i.e. taking derivative with respect to each β_j.

2.3

Show that for the fitted values $\widehat{\mu}_i$ from maximum likelihood estimates

$$\sum_i \widehat{\mu}_i = \sum_i y_i.$$

Therefore the deviance reduces to

$$D = 2 \sum_i y_i \log \frac{y_i}{\widehat{\mu}_i}.$$

Q1. Beta-Binomial

Let Y_i be the number of successes in n_i trials with

$$Y_i \sim \text{Bin}(n_i, \pi_i),$$

where the probabilities π_i have a Beta distribution

$$\pi \sim \text{Be}(\alpha, \beta)$$

with density function

$$f(x; \alpha, \beta) = \frac{\Gamma(\alpha + \beta)}{\Gamma(\alpha)\Gamma(\beta)} x^{\alpha-1}(1-x)^{\beta-1}, \quad x \in [0,1], \alpha > 0, \beta > 0.$$

1.1

Find the mean and variance of π.

$$E[x^n] = \frac{\Gamma(\alpha+\beta)\,\Gamma(\alpha+n)}{\Gamma(\alpha)\,\Gamma(\alpha+\beta+n)}$$

1.2

Find the mean and variance of Y_i and show that the variance of Y_i is always larger than or equal to that of a Binomial random variable with the same batch size and mean.

$$\int_0^1 \text{pdf beta} (\alpha+1, \beta) = 1$$

1.1

$$E[\pi] = \int_0^1 \pi \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} \pi^{\alpha-1} (1-\pi)^{\beta-1} \, d\pi$$

$$= \frac{\Gamma(\alpha+\beta)\Gamma(\alpha+1)}{\Gamma(\alpha)\Gamma(\alpha+1+\beta)} \int_0^1 \frac{\Gamma(\alpha+1+\beta)\, \pi^{\alpha+1-1}(1-\pi)^{\beta-1}}{\Gamma(\alpha+1)\,\Gamma(\beta)} \, d\pi$$

$$= \frac{\Gamma(\alpha+\beta)\,\alpha\Gamma(\alpha)}{\Gamma(\alpha)(\alpha+\beta)\Gamma(\alpha+\beta)} = \frac{\alpha}{\alpha+\beta}$$

$$E[\pi^2] = \frac{\Gamma(\alpha+\beta)\,\Gamma(\alpha+2)}{\Gamma(\alpha)\Gamma(\alpha+\beta+n)} \cdot 1 = \frac{\Gamma(\alpha+\beta)(\alpha+1)\Gamma(\alpha+1)}{\Gamma(\alpha)(\alpha+\beta+1)\Gamma(\alpha+\beta+1)} = \frac{(\alpha+1)\Gamma(\alpha+\beta)\,\alpha\Gamma(\alpha)}{\Gamma(\alpha)(\alpha+\beta+1)(\alpha+\beta)\Gamma(\alpha+\beta)}$$

$$= \frac{(\alpha+1)\alpha}{(\alpha+\beta+1)(\alpha+\beta)}$$

$$\text{Var}[\pi] = E[\pi^2] - \left(E[\pi]\right)^2$$

$$= \frac{(\alpha+1)\alpha}{(\alpha+\beta+1)(\alpha+\beta)} - \left(\frac{\alpha^2}{(\alpha+\beta)^2}\right) = \frac{(\alpha+1)\alpha}{(\alpha+\beta+1)(\alpha+\beta)}\frac{(\alpha+\beta)}{(\alpha+\beta)} - \frac{\alpha^2}{(\alpha+\beta)^2}\frac{(\alpha+\beta+1)}{(\alpha+\beta+1)}$$

$$= \frac{(\alpha^2+\alpha)(\alpha+\beta) - \alpha^2(\alpha+\beta+1)}{(\alpha+\beta)^2(\alpha+\beta+1)}$$

$$= \frac{\alpha^3 + \alpha^2\beta + \alpha^2 + \alpha\beta - \alpha^3 - \alpha^2\beta - \alpha^2}{(\alpha+\beta)^2(\alpha+\beta+1)}$$

$$= \frac{\alpha\beta}{(\alpha+\beta)^2(\alpha+\beta+1)}$$

1.2

$$E(Y_i) = E_{\pi_i}\left(\bar{E}_{Y_i}(Y_i|\pi_i)\right)$$

$$= E_{\pi_i}(n_i \cdot \pi_i)$$

$$= n_i \, E(\pi_i)$$

$$= n_i \frac{\alpha}{\alpha+\beta}$$

$$\text{Var}(\pi_i) = E_{\pi_i}\left(\text{Var}(Y_i|\pi_i)\right) + \text{Var}_{\pi_i}\left(E(Y_i|\pi_i)\right)$$

$$= E_{\pi_i}\left(n_i \cdot \pi_i \cdot (1-\pi_i)\right) + \text{Var}(n_i \cdot \pi_i)$$

$$= n_i\left(E(\pi_i) - E(\pi_i^2)\right) + n_i^2 \, \text{Var}(\pi)$$

$$= n_i\left(\frac{\alpha}{\alpha+\beta} - \frac{(\alpha+1)\alpha}{(\alpha+\beta+1)(\alpha+\beta)}\right) + n_i^2\left(\frac{\alpha\beta}{(\alpha+\beta)^2(\alpha+\beta+1)}\right)$$

$$= n_i\left(\frac{\cancel{\alpha^2}+\alpha\beta+\alpha-\cancel{\alpha^2}=\alpha}{(\alpha+\beta)(\alpha+\beta+1)} + \frac{n_i\alpha\beta}{(\alpha+\beta)^2(\alpha+\beta+1)}\right)$$

$$= n_i\left(\frac{\alpha\beta(\alpha+\beta) + n_i\alpha\beta}{(\alpha+\beta)^2(\alpha+\beta+1)}\right)$$

$$= \frac{n_i\alpha^2\beta + n_i\alpha\beta^2 + n_i^2\alpha\beta}{(\alpha+\beta)^2(\alpha+\beta+1)}$$

compare to $\text{Var}(\pi) = \frac{\alpha\beta}{(\alpha+\beta)^2(\alpha+\beta+1)}$, since $n_i \geqslant 1$,

so $\text{Var}(Y_i)$ always longer than binomial variance, unless $n_i = 1$.

Q2. Poisson regression log-likelihood

Let $Y_1, \ldots, Y_n$ be independent random variables with $Y_i \sim \text{Poisson}(\mu_i)$ and $\log \mu_i = \mathbf{x}_i^T \boldsymbol{\beta}$, $i = 1, \ldots, n$.

2.1

Write down the log-likelihood function.

2.2

Derive the gradient vector of the log-likelihood function with respect to the regression coefficients $\boldsymbol{\beta}$, i.e. taking derivative with respect to each β_j.

2.3

Show that for the fitted values $\hat{\mu}_i$ from maximum likelihood estimates

$$\sum_i \hat{\mu}_i = \sum_i y_i.$$

Therefore the deviance reduces to

$$D = 2 \sum_i y_i \log \frac{y_i}{\hat{\mu}_i}.$$

2.1 $\quad P(Y_i = y) = e^{-\mu_i} \dfrac{\mu_i^{y_i}}{y_i!} \qquad i = 1, \ldots, n$

$\ell(\beta) = \prod_{i=1}^{n} P(Y_i = y) = \sum_{i=1}^{n} -\mu_i + y_i \log(\mu_i) - \log(y_i!)$

$\quad = \sum_{i=1}^{n} \left(y_i x_i^T \beta - e^{x_i^T \beta} - \log(y_i!) \right)$, by $\log \mu_i = x_i^T \beta$. linear predictor.

2.2

$\nabla \ell(\beta) = \sum_{i=1}^{n} \left(y_i x_{ij} - x_{ij} e^{x_i^T \beta} \right)$

2.3

$\ell(\hat{\mu}_i) = \prod_{i=1}^{n} P(Y_i = y) = \sum_{i=1}^{n} \left(-\hat{\mu}_i + y_i \log(\hat{\mu}_i) - \log(y_i!) \right)$

$\nabla \ell(\hat{\mu}_i) = \sum_{i=1}^{n} \left(-1 + y_i \frac{1}{\ln \hat{\mu}_i} \right)$, set equal to 0

$\sum_{i=1}^{n} y_i \frac{1}{\hat{\mu}_i} - 1 = 0$

$\quad \sum_{i=1}^{n} y_i = \sum_{i=1}^{n} \hat{\mu}_i$

by $D = \log \dfrac{L \hat{\Omega}}{L \hat{\omega}}$, $\hat{\mu}_i = y_i$

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

$$= 2 \sum_{i=1}^{n} \left[y_i \log\left(\frac{y_i}{\hat{\mu}_i}\right) - (y_i - \hat{\mu}_i) \right]$$

$$= 2 \sum_{i=1}^{n} y_i \log\left(\frac{y_i}{\hat{\mu}_i}\right) \qquad \Rightarrow \text{ deviance for poisson}$$

Biostat 200C Homework 2
Due Apr 26 @ 11:59PM

Jiahao Tian

Q3. Simpson's paradox

The dataset `death` contains data on murder cases in Florida in 1977. The data is cross-classified by the race (black or white) of the victim, of the defendant and whether the death penalty was given.

```
library(faraway)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(ggplot2)
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --
```

```
## v tibble  3.1.6      v purrr   0.3.4
## v tidyr   1.2.0      v stringr 1.4.0
## v readr   2.1.2      v forcats 0.5.1
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
data(death)
death <- death %>%
  as_tibble() %>%
  print(n = Inf)
```

```
## # A tibble: 8 x 4
##       y penalty victim defend
##   <dbl> <fct>   <fct>  <fct>
## 1    19 yes     w      w
## 2   132 no      w      w
## 3     0 yes     b      w
## 4     9 no      b      w
## 5    11 yes     w      b
## 6    52 no      w      b
## 7     6 yes     b      b
## 8    97 no      b      b
```

3.1

Consider the frequency with which the death penalty is applied to black and white defendants, both marginally and conditionally, with respect to the race of the victim. Is this an example of Simpson's paradox? Are the observed differences in the frequency of application of the death penalty statistically significant?

Solutuon:

```
ct <- xtabs(y ~ penalty + defend, data = death)
prop.table(ct, 1)
```

```
##        defend
## penalty          b         w
##     no   0.5137931 0.4862069
##     yes  0.4722222 0.5277778
```

```
show(ct)
```

```
##        defend
## penalty   b   w
##     no  149 141
##     yes  17  19
```

```
summary(ct)
```

```
## Call: xtabs(formula = y ~ penalty + defend, data = death)
## Number of cases in table: 326
## Number of factors: 2
## Test for independence of all factors:
##   Chisq = 0.22145, df = 1, p-value = 0.6379
```

```
cta <- xtabs(y ~ penalty + defend + victim, data = death)
prop.table(cta, 1)
```

```
## , , victim = b
##
##        defend
## penalty          b          w
##     no   0.33448276 0.03103448
```

```
##     yes 0.16666667 0.00000000
##
## , , victim = w
##
##         defend
## penalty           b          w
##     no   0.17931034 0.45517241
##     yes 0.30555556 0.52777778
```

```
show(cta)
```

```
## , , victim = b
##
##        defend
## penalty   b   w
##     no   97   9
##     yes   6   0
##
## , , victim = w
##
##        defend
## penalty   b   w
##     no   52 132
##     yes  11  19
```

```
summary(cta)
```

```
## Call: xtabs(formula = y ~ penalty + defend + victim, data = death)
## Number of cases in table: 326
## Number of factors: 3
## Test for independence of all factors:
##   Chisq = 122.4, df = 4, p-value = 1.642e-25
```

```
margin.table(cta, c(1,2))
```

```
##        defend
## penalty   b   w
##     no   149 141
##     yes  17  19
```

It is an example of Simpson's paradox. Checking the associations throughout all defend groups, we will see the marginal association where we add over the defend groups is different from the conditional association observed within defend groups.

```
ct3 <- xtabs(y ~ penalty + defend + victim, data = death)
dim(ct3)
```

```
## [1] 2 2 2
```

```
mantelhaen.test(ct3, exact = TRUE)
```

```
##
##  Exact conditional test of independence in 2 x 2 x k tables
##
## data:  ct3
## S = 149, p-value = 0.3005
## alternative hypothesis: true common odds ratio is not equal to 1
## 95 percent confidence interval:
##  0.2754218 1.5500653
## sample estimates:
## common odds ratio
##         0.6453241
```

From the Cochran-Mantel-Haenszel test, the p-value is larger than 0.05, which shows that dependence on
death penalty and the race of the defendants is not significant.Therefore, the observed differences in the
frequency of application of the death penalty are not statistically significant.

3.2

Determine the most appropriate dependence model between the variables.

Solution:

```
modi <- glm(y ~ penalty + victim + defend, family = poisson, data = death)
summary(modi)
```

```
##
## Call:
## glm(formula = y ~ penalty + victim + defend, family = poisson,
##     data = death)
##
## Deviance Residuals:
##       1        2        3        4        5        6        7        8
##  1.9881   3.7542  -3.4843  -7.0237  -0.3023  -5.0100  -0.1196   5.7623
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  3.92657    0.11075  35.455  < 2e-16 ***
## penaltyyes  -2.08636    0.17671 -11.807  < 2e-16 ***
## victimw      0.64748    0.11662   5.552 2.83e-08 ***
## defendw     -0.03681    0.11079  -0.332     0.74
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 395.92  on 7  degrees of freedom
## Residual deviance: 137.93  on 4  degrees of freedom
## AIC: 181.61
##
## Number of Fisher Scoring iterations: 5
```

```
## it is not independent.
```

```
## Joint Independent
glm(y ~ penalty * victim + defend, family = poisson, data = death) %>%
  summary()
```

```
##
## Call:
## glm(formula = y ~ penalty * victim + defend, family = poisson,
##     data = death)
##
## Deviance Residuals:
##     1       2       3       4       5       6       7       8
##  1.066   4.102  -2.427  -7.380  -1.152  -4.707   1.486   5.260
##
## Coefficients:
##                    Estimate Std. Error z value Pr(>|z|)
## (Intercept)         3.98853    0.11131  35.832  < 2e-16 ***
## penaltyyes         -2.87168    0.41964  -6.843 7.75e-12 ***
## victimw             0.55150    0.12194   4.523 6.10e-06 ***
## defendw            -0.03681    0.11079  -0.332   0.7397
## penaltyyes:victimw  1.05794    0.46354   2.282   0.0225 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 395.92  on 7  degrees of freedom
## Residual deviance: 131.68  on 3  degrees of freedom
## AIC: 177.36
##
## Number of Fisher Scoring iterations: 6
```

```
glm(y ~ penalty * defend + victim, family = poisson, data = death) %>%
  summary()
```

```
##
## Call:
## glm(formula = y ~ penalty * defend + victim, family = poisson,
##     data = death)
##
## Deviance Residuals:
##      1       2       3       4       5       6       7       8
##  1.7146  3.8506  -3.6132  -6.9704  -0.0479  -5.0906  0.0657  5.6902
##
## Coefficients:
##                    Estimate Std. Error z value Pr(>|z|)
## (Intercept)         3.93555    0.11212  35.100  < 2e-16 ***
## penaltyyes         -2.17073    0.25600  -8.480  < 2e-16 ***
## defendw            -0.05519    0.11749  -0.470    0.639
## victimw             0.64748    0.11662   5.552 2.83e-08 ***
## penaltyyes:defendw  0.16641    0.35391   0.470    0.638
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 395.92  on 7  degrees of freedom
## Residual deviance: 137.71  on 3  degrees of freedom
## AIC: 183.39
##
## Number of Fisher Scoring iterations: 5
```

```
## by assum the first two variables are dependent, and jointly independent of the third
## still lack of fit by looking at the interaction term or residual deviance.
```

```
## Conditinal Independence
glm(y ~  penalty * victim + defend * victim, family = poisson, data = death) %>%
  drop1(test = "Chi")
```

```
## Single term deletions
##
## Model:
## y ~ penalty * victim + defend * victim
##               Df Deviance     AIC    LRT Pr(>Chi)
## <none>             1.882  49.563
## penalty:victim  1    8.132  53.813   6.25  0.01242 *
## victim:defend   1  131.680 177.361 129.80  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
glm(y ~  penalty * victim + defend * victim, family = poisson, data = death) %>%
  summary()
```

```
##
## Call:
## glm(formula = y ~ penalty * victim + defend * victim, family = poisson,
##     data = death)
##
## Deviance Residuals:
##        1         2         3         4         5         6         7         8
## -0.47967   0.18976  -0.98198   0.16368   0.70243  -0.29660   0.20237  -0.04887
##
## Coefficients:
##                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)         4.5797     0.1011  45.314  < 2e-16 ***
## penaltyyes         -2.8717     0.4196  -6.843 7.75e-12 ***
## victimw            -0.5876     0.1639  -3.586 0.000336 ***
## defendw            -2.4375     0.3476  -7.013 2.34e-12 ***
## penaltyyes:victimw  1.0579     0.4635   2.282 0.022471 *
## victimw:defendw     3.3116     0.3786   8.748  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 395.9153  on 7  degrees of freedom
## Residual deviance:   1.8819  on 2  degrees of freedom
## AIC: 49.563
##
## Number of Fisher Scoring iterations: 4
```

```r
library(gtsummary)
glm(y ~ penalty * victim + defend * victim, family = poisson, data = death) %>%
  step(test = "Chi") %>%
  drop1(test = "Chi")
```

```
## Start:  AIC=49.56
## y ~ penalty * victim + defend * victim
##
##                 Df Deviance     AIC    LRT Pr(>Chi)
## <none>               1.882  49.563
## - penalty:victim  1    8.132  53.813   6.25  0.01242 *
## - victim:defend   1  131.680 177.361 129.80  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
## Single term deletions
##
## Model:
## y ~ penalty * victim + defend * victim
##                Df Deviance     AIC    LRT Pr(>Chi)
## <none>              1.882  49.563
## penalty:victim  1    8.132  53.813   6.25  0.01242 *
## victim:defend   1  131.680 177.361 129.80  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
lmod <- glm(
  y ~ penalty * victim + defend * victim, family = poisson, data = death)
lmod %>%
  tbl_regression(exponentiate = TRUE, intercept = TRUE)
```

```
## Table printed with 'knitr::kable()', not {gt}. Learn why at
## https://www.danieldsjoberg.com/gtsummary/articles/rmarkdown.html
## To suppress this message, include 'message = FALSE' in code chunk header.
```

Characteristic	IRR	95% CI	p-value
(Intercept)	97.5	79.4, 118	<0.001
penalty			
no			
yes	0.06	0.02, 0.12	<0.001
victim			
b			

Characteristic	IRR	95% CI	p-value
w	0.56	0.40, 0.76	<0.001
defend			
b			
w	0.09	0.04, 0.16	<0.001
penalty * victim			
yes * w	2.88	1.24, 7.87	0.022
victim * defend			
w * w	27.4	13.7, 61.4	<0.001

The most appropriate dependence model is y ~ penalty * victim + defend * victim fitted by poisson regression with coefficients shown in the table above. This is a conditional independence model, also, shows that this model has the best fit is compatible with our result from the the Cochran-Mantel-Haenszel test.

3.3

Fit a binomial regression with death penalty as the response and show the relationship to your model in the previous question.

Solution:

```
death1 = cbind(matrix(death$y, ncol = 2, byrow = TRUE),
               unique(death %>%
                        select(victim, defend)))
colnames(death1) [1:2] = c('yes', 'no')
glm(cbind(yes, no) ~ victim * defend, family = binomial, data = death1) %>%
  step(test = "Chi") %>%
  summary()
```

```
## Start:  AIC=20.31
## cbind(yes, no) ~ victim * defend
##
##                  Df Deviance    AIC     LRT Pr(>Chi)
## - victim:defend   1  0.70074 19.015 0.70074   0.4025
## <none>               0.00000 20.314
##
## Step:  AIC=19.01
## cbind(yes, no) ~ victim + defend
##
##          Df Deviance    AIC    LRT Pr(>Chi)
## - defend  1   1.8819 18.196 1.1812 0.277121
## <none>        0.7007 19.015
## - victim  1   7.9102 24.224 7.2094 0.007252 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Step:  AIC=18.2
## cbind(yes, no) ~ victim
##
##          Df Deviance    AIC    LRT Pr(>Chi)
## <none>        1.8819 18.196
## - victim  1   8.1316 22.445 6.2497  0.01242 *
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1


##
## Call:
## glm(formula = cbind(yes, no) ~ victim, family = binomial, data = death1)
##
## Deviance Residuals:
##       1        2        3        4
## -0.5158  -0.9955   0.7625   0.2082
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -2.8717     0.4196  -6.843 7.75e-12 ***
## victimw       1.0579     0.4635   2.282   0.0225 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 8.1316  on 3  degrees of freedom
## Residual deviance: 1.8819  on 2  degrees of freedom
## AIC: 18.196
##
## Number of Fisher Scoring iterations: 4
```

The final binomial model is penalty ~ victim, retains only main effect. And defend is not a significant predictor for death penalty and it is related with the model selected in the previous question with same deviance.