VGAM Family Functions for Bivariate Binomial Responses

T. W. Yee

May 22, 2008

Beta Version 0.7-6

© Thomas W. Yee

Department of Statistics,
University of Auckland,
New Zealand
yee@stat.auckland.ac.nz
http://www.stat.auckland.ac.nz/∼yee

Contents

[Important note: This document and code is not yet finished, but should be completed one day . . .]

1 Introduction

This document describes in detail VGAM family functions for modeling bivariate binomial responses. Such commonly arise in medical and biological studies, e.g., ophthalmic studies where each eye is a response, measurements on pairs of twins, presence/absence data on two species of plant at the same geographical site. We write $\boldsymbol{Y} = (Y_1, Y_2)^T$, where Y_1 and Y_2 takes only the values 0 and 1; it is customary to denote "failure" by 0 and "success" by 1. Let $p_{rs} = P(Y_1 = r, Y_2 = s),\ \ r, s = 0, 1$, be the joint probabilities, and $p_j = P(Y_j = 1),\ \ j = 1, 2$, be the marginal probabilities.

A general reference for bivariate binomial data is McCullagh and Nelder (1989). Many of VGAM's features come from `glm()` and `gam()` so that readers unfamiliar with these functions are referred to Chambers and Hastie (1993). Additionally, the VGAM *User Manual* should be consulted for general instructions about the software. Lastly, the VGAM documentation on log-linear models is also very relevant as it provides another alternative.

2 Models

This section describes two classes of models currently implemented by VGAM—via the `binom2.or()` and `binom2.rho()` family functions.

2.1 Bivariate logit model

The *bivariate logistic model* (or *bivariate logistic odds-ratio model*) (BLOM) described by Section 6.5.6 of McCullagh and Nelder (1989) and Palmgren (1989) is specified by modelling the marginal distributions of each Y_j, and also the odds ratio. The odds ratio, $\psi = p_{00}\, p_{11} / (p_{01}\, p_{10})$, is used to describe the association between the two responses. The model is:

$$\begin{aligned} \operatorname{logit} p_j &= \eta_j(\boldsymbol{x}) \qquad j = 1, 2\,, \qquad (1) \\ \log\ \psi(\boldsymbol{x}) &= \eta_3(\boldsymbol{x}), \end{aligned}$$

where $\eta_j = \boldsymbol{\beta}_j^T \boldsymbol{x}$. The probability p_{11} can be obtained from p_1, p_2 and ψ as:

$$p_{11} = \begin{cases} \frac{1}{2}(\psi - 1)^{-1}\{a - \sqrt{a^2 + b}\}, & \psi \neq 1; \\ p_1\, p_2, & \psi = 1, \end{cases}$$

where $a = 1 + (p_1 + p_2)(\psi - 1)$ and $b = -4\psi(\psi - 1)p_1 p_2$ (Dale, 1986). The other three joint probabilities p_{rs} can then be recovered easily from the marginals and p_{11}.

The BLOM is similar to the bivariate probit model (see next section) but has several advantages: it is computationally simpler, and odds ratios are preferred to correlation coefficients when describing the association between two binary variables. In theory, there is no reason why other link functions could not be used for the marginal probabilities.

The BLOM is implemented via `binom2.or()`, and is to be preferred over the BPM for both theoretical and practical reasons. For more information, see le Cessie and van Houwelingen (1994).

2.2 Bivariate probit model

The *bivariate probit model* (BPM; Ashford and Sowden (1970) can be written

$$\begin{aligned} P(Y_j = 1|\boldsymbol{x}) &= \Phi(\eta_j(\boldsymbol{x}))\ , \quad j = 1, 2, \\ P(Y_1 = 1, Y_2 = 1|\boldsymbol{x}) &= \Phi_2\left(\eta_1(\boldsymbol{x}), \eta_2(\boldsymbol{x}); \rho = \frac{\exp\{\eta_3(\boldsymbol{x})\} - 1}{\exp\{\eta_3(\boldsymbol{x})\} + 1}\right)\ . \end{aligned} \tag{2}$$

Here, the correlation parameter ρ is modelled as a function of the covariates and $\Phi(\cdot)$ is the distribution function of a standard normal distribution and $\Phi_2(\cdot, \cdot; \rho)$ is the distribution function of a bivariate normal with zero means, unit variances and correlation ρ.

The BPM has a nice interpretation in terms of latent variables. Note that, whereas each marginal is modelled as a logistic regression in the BLOM, each marginal is modelled as a "probit analysis" for the BPM. The multivariate probit model, of which the BPM is a special case, is generally applicable to $M > 3$ binary responses. However, it is computationally difficult to estimate because it requires integration of a N_M density. VGAM currently has no family function that will fit a $M \geq 3$ dimensional probit model.

The BPM is implemented via `binom2.rho()`.

Table 1: *The coalminers data set. Note: B =Breathlessness, W =Wheeze.*

Age Group	$(B = 1, W = 1)$	$(B = 1, W = 0)$	$(B = 0, W = 1)$	$(B = 0, W = 0)$
20–24	9	7	95	1841
25–29	23	9	105	1654
30–34	54	19	177	1863
35–39	121	48	257	2357
40–44	169	54	273	1778
45–49	269	88	324	1712
50–54	404	117	245	1324
55–59	406	152	225	967
60–64	372	106	132	526

2.3 The Frank Family of Distributions

The Frank family of copulas (see, e.g., Genest (1987)) can be used to model bivariate binary responses. However, being more general, it is discussed in a separate document.

3 Other Topics

3.1 Code and Classes

One has

```
> args(binom2.or)

function (lmu = "logit", lmu1 = lmu, lmu2 = lmu, loratio = "loge",
    emu = list(), emu1 = emu, emu2 = emu, eoratio = list(), imu1 = NULL,
    imu2 = NULL, ioratio = NULL, zero = 3, exchangeable = FALSE,
    tol = 0.001)
NULL

> args(binom2.rho)

function (lrho = "rhobit", erho = list(), init.rho = 0.4, zero = 3,
    exchangeable = FALSE)
NULL
```

Of course, `lp` and `lor` are the links of the marginals and odds ratio respectively. For the BPM, it doesn't really make sense to use different link functions for the marginals as the BPM is theoretically tied to the bivariate normal distribution. If an odds ratio is within `tol` of unity then it is considered as the case of independence. The `"rhobit"` transformation is for $-1 < \rho < 1$: $\eta_3 = \log((1+\rho)/(1-\rho))$ or

```
> rhobit("rho", short = FALSE)

[1] "log((1+rho)/(1-rho))"
```

3.2 Input

The response `y` in `vglm()`/`vgam()` for `binom2.or()` is of the form, e.g.,

```
> vglm(y ~ x, binom2.or, weights = w)
```

where `weights` is usually optional. Here, `y` may be one of three types:

1. a 4-column matrix of sample proportions, where the order of the columns correspond to $(y_1 = 0, y_2 = 0)$, $(y_1 = 0, y_2 = 1)$, $(y_1 = 1, y_2 = 0)$, $(y_1 = 1, y_2 = 1)$, respectively. Then `weights` must be assigned the number of observations (unless all $n_i = 1$).

2. a 2-column matrix $(\boldsymbol{y}_1\, \boldsymbol{y}_2)$ of 0's and 1's.

3. a vector containing 4 unique values (including a factor with 4 levels). When sorted or ordered, these correspond to $(y_1 = 0, y_2 = 0)$, $(y_1 = 0, y_2 = 1)$, $(y_1 = 1, y_2 = 0)$, $(y_1 = 1, y_2 = 1)$, respectively.

In the future more than two binary responses may be modelled by VGAM family functions for GEE1—and will be documented elsewhere when finished.

3.3 Output

Suppose `fit` is a fitted bivariate binomial VGAM object. Then the fitted values (in `fitted(fit)`) are held in a $n \times 4$ matrix of probabilities (whose rows sum to unity). The order of the columns are like that of input, viz., $(y_1 = 0, y_2 = 0)$, $(y_1 = 0, y_2 = 1)$, $(y_1 = 1, y_2 = 0)$, $(y_1 = 1, y_2 = 1)$. Furthermore, `weights(fit, type="prior")` contain the $n_i = \sum_{j=1}^{4} y_{ij}$.

The $n \times 4$ response matrix is saved in `fit@y`.

3.4 Constraints

VGAM family functions for bivariate binomial responses have the `parallel`, `exchangeable` and `zero` arguments. By default, `parallel=FALSE`, `exchangeable=FALSE` and `zero=3`; this means that the correlation parameters ψ and ρ are modelled as an intercept-only unless assigned a `NULL` value.

3.5 Convergence

The BPM seems sensitive to the initial value of ρ, i.e., has difficulties in converging sometimes. If the default value doesn't work, assign a different value into the argument `init.rho`.

3.6 Implementation Details

The S expression `process.binomial2.data.vgam` provides a unified way of handling the response variable. Similarly, the S expression `deviance.categorical.data.vgam` computes the deviance for all the models in this document.

The bivariate normal integrals are computed using C code in the file `gaut.c`.

4 Tutorial Examples

4.1 Coalminers Data

The following reproduces the models of §6.6 of McCullagh and Nelder (1989). The `summary()` produces results that agree with Table 6.7.

```
> data(coalminers)
> coalminers = transform(coalminers, Age = (age - 42)/5)
> coalminers

   BW BnW nBW nBnW age Age
1   9   7  95 1841  22  -4
2  23   9 105 1654  27  -3
3  54  19 177 1863  32  -2
4 121  48 257 2357  37  -1
5 169  54 273 1778  42   0
6 269  88 324 1712  47   1
7 404 117 245 1324  52   2
8 406 152 225  967  57   3
9 372 106 132  526  62   4
```

```
> fit = vglm(cbind(nBnW, nBW, BnW, BW) ~ Age, binom2.or(zero = NULL),
+     coalminers, trace = TRUE)

VGLM    linear loop  1 :  deviance = 30.4424
VGLM    linear loop  2 :  deviance = 30.3939
VGLM    linear loop  3 :  deviance = 30.3939
VGLM    linear loop  4 :  deviance = 30.3939

> round(fitted(fit), dig = 3)

     00    01    10    11
1 0.937 0.049 0.005 0.008
2 0.915 0.064 0.007 0.015
3 0.884 0.080 0.010 0.025
4 0.844 0.097 0.016 0.043
5 0.792 0.114 0.024 0.070
6 0.726 0.126 0.036 0.112
7 0.644 0.130 0.054 0.172
8 0.547 0.126 0.078 0.249
9 0.438 0.113 0.109 0.341

> summary(fit)

Call:
vglm(formula = cbind(nBnW, nBW, BnW, BW) ~ Age, family = binom2.or(zero = NULL),
    data = coalminers, trace = TRUE)

Pearson Residuals:
                Min       1Q    Median      3Q    Max
logit(mu1)  -1.9687 -1.02898 -0.433399 0.38046 2.6796
logit(mu2)  -1.1461 -0.86856 -0.112040 0.71249 1.1973
log(oratio) -1.5456 -0.49029 -0.041692 0.66471 1.3264

Coefficients:
                 Value Std. Error  t value
(Intercept):1 -2.26247  0.0298919 -75.6884
(Intercept):2 -1.48776  0.0205593 -72.3645
(Intercept):3  3.02191  0.0697319  43.3361
Age:1          0.51451  0.0120713  42.6226
Age:2          0.32545  0.0088686  36.6966
Age:3         -0.13136  0.0284417  -4.6187

Number of linear predictors:  3

Names of linear predictors: logit(mu1), logit(mu2), log(oratio)

Dispersion Parameter for binom2.or family:   1

Residual Deviance: 30.39386 on 21 degrees of freedom
```

```
Log-likelihood: -12858.01 on 21 degrees of freedom

Number of Iterations: 4
```

```
> coef(fit, matrix = TRUE)

            logit(mu1) logit(mu2) log(oratio)
(Intercept) -2.2624682 -1.4877603   3.0219085
Age          0.5145103  0.3254455  -0.1313647
```

And Table 6.8 agrees with

```
> round(c(weights(fit, type = "prior")) * fitted(fit), dig = 3)

        00      01      10      11
1 1829.946  96.446   9.049  16.559
2 1638.068 113.972  12.493  26.467
3 1868.118 169.100  22.179  53.602
4 2348.671 271.298  44.010 119.021
5 1800.740 258.869  54.257 160.134
6 1736.803 301.303  86.072 268.821
7 1346.050 272.491 112.363 359.096
8  956.839 219.814 137.156 436.191
9  497.345 128.511 123.321 386.823
```

The regression coefficients are highly interpretable—see §6.6 of McCullagh and Nelder (1989).

4.2 Chest Data

The data frame chest cross-classifies 10186 participants in a New Zealand cohort study by age and chest pain in the left and right sides of the body. For example, amongst 19 year olds, there were 65 without any chest pain, 1 with right-side chest pain only, 4 with left-side chest pain only, and 3 with chest pain on both sides[1]. One can fit a nonparametric bivariate logistic model to this data by

```
> data(chest)
> chest[1:5, ]

  age nolnor nolr lnor lr
1  16      2    0    0  0
2  17     16    0    0  1
3  18     34    1    2  0
4  19     65    1    4  3
5  20    126    4    6  1

> cvgam0 <- vgam(cbind(nolnor, nolr, lnor, lr) ~ s(age),
+     binom2.or(exch = FALSE, zero = 3), dat = chest)
> par(mfrow = c(3, 1), mar = c(5, 5, 0.2, 1) + 0.1, xpd = TRUE,
+     las = 1)
> plot(cvgam0, se = TRUE, scale = 2, scol = "blue")
```

[1]Recall the order of the columns is $(y_1, y_2) = (0,0), (0,1), (1,0), (1,1)$. Here, y_1 is left chest pain.

For illustration's sake, the object cvgam0 is a non-exchangeable model: the marginal probabilities are different. The top two plots of Fig. 1 show this model. The marginals looks similar. Another method of comparison is to overlay the fitted function by using

```
> plot(cvgam0, se = TRUE, overlay = TRUE, scale = 2, scol = "blue")
```

(not done here).

Let's try fitting an exchangeable model ($\eta_1 = \eta_2$) with the log odds ratio being an intercept.

```
> cvgam <- vgam(cbind(nolnor, nolr, lnor, lr) ~ s(age), binom2.or(exch = TRUE,
+     zero = 3), dat = chest)
> plot(cvgam, se = TRUE, scale = 2, scol = "blue")
```

It produces the bottom plot of Fig. 1. The scale argument is used to force the vertical axis of the plots to be equal—thus making the size of the functions comparable. Notice that the standard error band is noticeably more narrow because it effectively uses twice the data to estimate it. Interestingly, the prevalence of chest pain appears to decrease between ages 40 and 60 years. Lastly,

```
> summary(cvgam)
```

```
Call:
vgam(formula = cbind(nolnor, nolr, lnor, lr) ~ s(age), family = binom2.or(exch = TRUE,
    zero = 3), data = chest)

Number of linear predictors:    3

Names of linear predictors: logit(mu1), logit(mu2), log(oratio)

Dispersion Parameter for binom2.or family:   1

Residual Deviance:  544.4184 on 213.056 degrees of freedom

Log-likelihood: -4803.252 on 213.056 degrees of freedom

Number of Iterations:  6

DF for Terms and Approximate Chi-squares for Nonparametric Effects

              Df Npar Df Npar Chisq     P(Chi)
(Intercept):1  1
(Intercept):2  1
s(age)         1     2.9    21.9058 6.3682e-05
```

showing that there is very strong evidence that the common marginal is nonlinear in age.

As an exercise, explore whether the odds ratio is in fact constant over age. Try it linear with age. That is, fit

$$\begin{aligned} \text{logit } p_j(\text{age}) &= f(\text{age}), \\ \log\ \psi(\text{age}) &= \beta_{(3)0} + \beta_{(3)1} \times \text{age}. \end{aligned} \tag{3}$$

by

```
> fit2 <- vgam(cbind(nBnW, nBW, BnW, BW) ~ s(age, df = c(4,
+     1)), binom2.or(exch = TRUE, zero = NULL), chest)
```

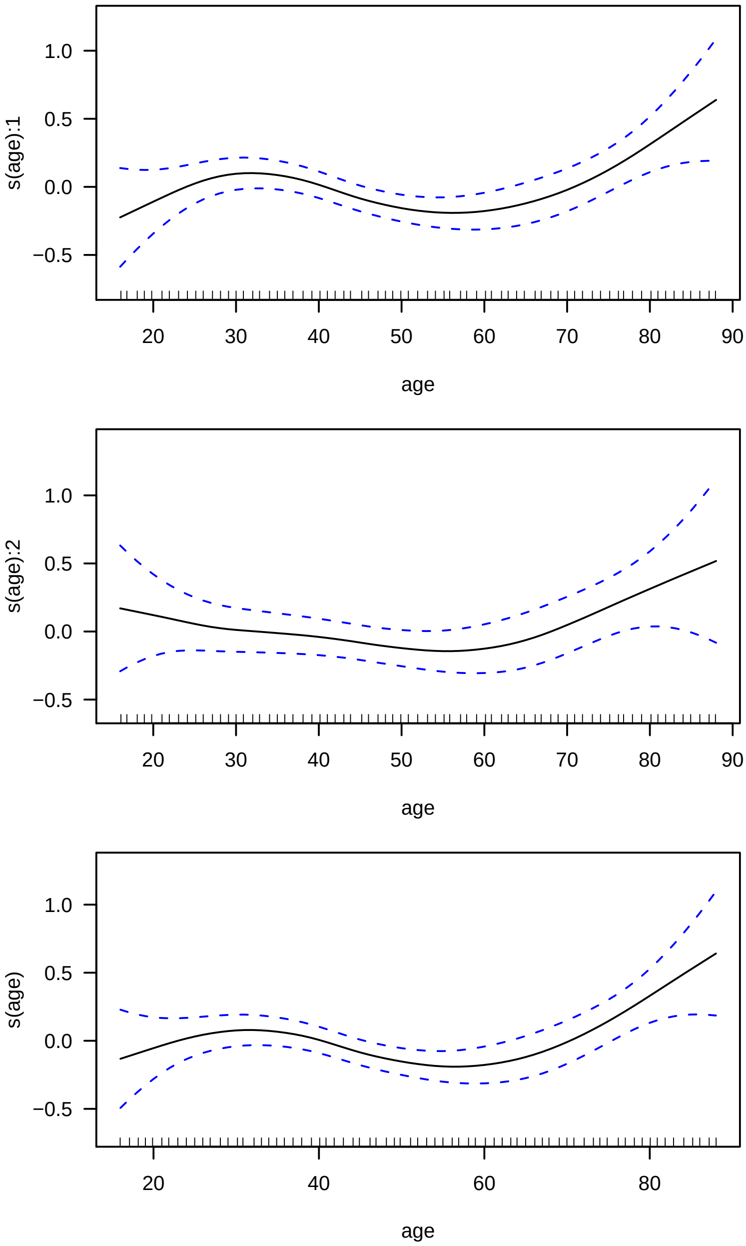

Figure 1: *Bivariate logistic model fitted to the chest pain data. The top two plots are a non-exchangeable model, whereas the bottom is exchangeable.*

4.3 Plotting Odds Ratios

Suppose you have a bivariate logit model with several variables and you want a plot of the odds ratio versus one of the variables. This can be achieved using the ideas of the following (artificial R) example.

```
> set.seed(123)
> n = 900
> y1 = round(runif(n) + 0.4)
> y2 = round(runif(n) + 0.4)
> x2 = rnorm(n)
> x3 = rnorm(n)
```

```
> x4 = rnorm(n)
> x5 = rnorm(n)
> COUNT = rep(10, n)
> fit <- vgam(cbind(y1, y2) ~ s(x2) + s(x3) + x4 + x5, binom2.or(zero = NULL,
+     exchangeable = TRUE), weight = COUNT)
> fit.terms = predict(fit, type = "terms", se = TRUE, raw = TRUE)
> newdat = data.frame(x2 = x2, x3 = rep(0, n), x4 = rep(0,
+     n), x5 = rep(0, n))
> pfit = predict(fit, newdat)
> pfit.lo = pfit - 2 * fit.terms$se.fit[, "s(x2):2"]
> pfit.hi = pfit + 2 * fit.terms$se.fit[, "s(x2):2"]
> oo = with(newdat, order(x2))
> with(newdat, matplot(x2[oo], exp(cbind(pfit[oo, "log(oratio)"],
+     pfit.lo[oo, "log(oratio)"], pfit.hi[oo, "log(oratio)"])),
+     lwd = 2, col = c("black", "blue", "blue"), lty = c(1,
+         2, 2), type = "l", xlab = "x2", ylab = "Odds Ratio",
+     main = ""))
```

This produces a plot of the odds ratio of Y_1 and Y_2 with respect to x_1, keeping all the other variables fixed at zero (Fig. 2). Standard error bands are included in the plot. One can easily modify the code to handle x_2. However, the validity of using ± 2 SE bands here needs justification which hasn't been obtained!

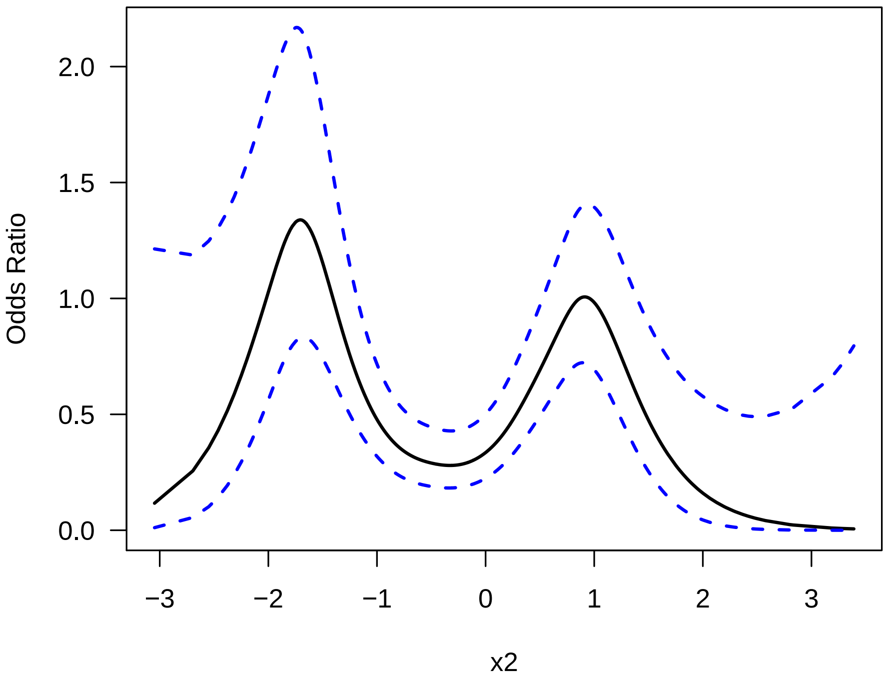

Figure 2: *Odds ratio plot.*

Exercises

1. Write a VGAM family function to fit a trivariate probit model. You will need to write/obtain code to perform integration of a N_3 random vector. Call it `binom3.rho()`. Note: what constraints on the three correlation parameters ρ_{12}, ρ_{13}, and ρ_{23} are needed?

Acknowledgements

Thanks to Dr C. Ananth for raising the OR-plotting problem.

References

Ashford, J. R., Sowden, R. R., 1970. Multi-variate probit analysis. Biometrics 26, 535–546.

Chambers, J. M., Hastie, T. J. (Eds.), 1993. Statistical Models in S. Chapman & Hall, New York.

Dale, J. R., 1986. Global cross-ratio models for bivariate discrete ordered responses. Biometrics 42, 909–917.

Genest, C., 1987. Frank's family of bivariate distributions. Biometrika 74, 549–555.

le Cessie, S., van Houwelingen, J. C., 1994. Logistic regression for correlated binary data. Applied Statistics 43, 95–108.

McCullagh, P., Nelder, J. A., 1989. Generalized Linear Models, 2nd Edition. Chapman & Hall, London.

Palmgren, J., 1989. Regression models for bivariate binary responses. Tech. Rep. 101, Department of Biostatistics, University of Washington, Seattle.