

Generalized linear models

1 Oct 2024

```
## it's nice to include packages at the top
## (and NOT automatically install them)
## try not to carry over packages you don't use
library(ggplot2); theme_set(theme_bw())
## diagnostics
library(performance)
library(DHARMA)
## downstream model evaluation
library(broom)
library(dotwhisker)
library(emmeans)
library(effects)
library(marginaleffects)
library(parameters)
## library(ggeffects)
```

References

Faraway (2016), McCullagh and Nelder (1989) (classic), Wood (2017, very rapid review)

Basics

- assume $\mathbf{y}_i \sim \text{Dist}(g^{-1}((\mathbf{X}\beta)_i))$
- g = **link function**
- $\eta = \mathbf{X}\beta$ = **linear predictor**
- **link scale** or **linear predictor** scale vs. **data** or **response** scale
- GLMs inverse-transform η , they don't transform y
- allows:

- separate control of heteroscedasticity and nonlinearity
- almost as convenient/efficient as LMs
- equivalent to MLE in many cases
- the vast majority of GLMs are logistic (binary data) or Poisson
- lots of inference, diagnostics, etc. inherited from LM framework

Exponential family

- $f(x|\theta) = h(x)g(\theta) \exp(\eta(\theta)T(x))$
- e.g. Poisson: $f(x|\theta) = \theta^x \exp(-\theta)/x! = (1/x!) \exp(-\theta) \exp(x \log(\theta))$
- $h(x) = 1/x!; g(\theta) = \exp(-\theta); \eta(\theta) = \log(\theta); T(x) = x$
- models with $T(x) = x$ are in the **exponential dispersion** (sub) family (Jørgensen 1987),
- $\eta(\theta)$ is the **canonical link** function for the family (nice mathematical properties, e.g. observed information = expected information [*Fisher(scoring)*])
- binomial, Poisson, Gamma (inverse Gaussian, von Mises distribution ...)

Computation

- iteratively reweighted least squares
 - if we know μ_i , we know V_i (up to a proportion)
 - we can do weighted least squares on the link scale
 - use new β estimates to recalculate μ_i and V
- needs starting values, but almost always robust to them
- alternatives/more complex versions: glmmTMB, VGAM packages
- lots more detail on the computational issues (McCullagh and Nelder 1989; Marschner 2011; Myers et al. 2010; Dobson and Barnett 2008; Mount 2012; Gelman 2011; Robinson 2010)
- basics (from Faraway (2016) p. 155)

IRLS

Can show (Faraway p. 154) from definition of exponential (dispersion) family that the score equations are equivalent to

$$\sum_i \frac{(y_i - \mu_i)}{V(\mu_i)} \frac{\partial \mu_i}{\partial \beta_j} = 0$$

which is equivalent to minimizing

$$\sum_i \frac{(y_i - \mu_i)^2}{V(\mu_i)}$$

- compute **adjusted dependent variate**:

$$Z_{(i)} = \hat{\eta}_{(i)} + (Y - \hat{\mu}_{(i)}) \left(\frac{d\eta}{d\mu} \right)_{(i)}$$

(note: $\frac{d\eta}{d\mu} = \frac{d\eta}{dg(\eta)} = 1/g'(\eta)$: translate from raw to linear predictor scale)

- compute **weights**

$$W_{(i)}^{-1} = \left(\frac{d\eta}{d\mu} \right)_{(i)}^2 V(\hat{\mu}_{(i)})$$

(translate variance from raw to linear predictor scale). This is the inverse variance of $Z_{(i)}$.

- regress $z_{(i)}$ on the covariates with weights $W_{(i)}$ to get new β estimates (\rightarrow new η , μ , $V(\mu)$...)

Tricky bits: starting values, non-convergence, etc.. (We will worry about these later!)

```
myglmfit <- function(y, X, family, tol=1e-8, maxit=50) {
  mu <- y ## set initial values
  ## set up 'oldbeta' and 'beta' so they're not identical
  oldbeta <- rep(0, ncol(X))
  beta <- rep(1, ncol(X))
  it <- 1 ## number of iterations
  while (it < maxit && max(abs((1-beta/oldbeta)))>tol) {
    oldbeta <- beta
    eta <- family$linkfun(mu) ## calc. linear predictor
    mm <- family$mu.eta(eta) ## calc. d(mu)/d(eta)
    adjdev <- eta + (y-mu)/mm ## adjusted response
    W <- c(1/(mm^2*family$variance(mu))) ## weights
    beta <- lm.wfit(X, adjdev, W)$coefficients ## weighted least-squares
    mu <- family$linkinv(X %*% beta) ## compute new mu
    it <- it+1 ## update
  }
  beta
}
X <- model.matrix(~wool*tension, data=warpbreaks)
y <- warpbreaks$breaks
myglmfit(y,X,poisson())
```

(Intercept)	woolB	tensionM	tensionH	woolB:tensionM
3.7967368	-0.4566272	-0.6186830	-0.5957987	0.6381768
woolB:tensionH				
0.1883632				

```
coef(glm(breaks~wool*tension, data=warpbreaks, family=poisson))
```

(Intercept)	woolB	tensionM	tensionH	woolB:tensionM
3.7967368	-0.4566272	-0.6186830	-0.5957987	0.6381768
woolB:tensionH				
0.1883632				

Mean-variance relations

- can show that we need only the link function and the **variance function** $V = f(\mu)$ for computation (may also depend multiplicatively on a **scale** or **dispersion parameter**, e.g. $V = \mu$ for Poisson, $V = \sigma^2$)

Link functions

- canonical doesn't always work best (e.g. Gamma/inverse link)
- probit vs logit; nearly the same shape, interpretational difference
- cloglog; *log-hazard* scale
- inverse link: linear changes in the *rate* of events

Log-hazards and log-hazard offsets

- if hazard is h , probability is $1 - \exp(-h)$
- $C(\mu) = \log(-\log(1 - \mu))$
- $C^{-1}(\eta) = 1 - \exp(-\exp(\eta))$
- $C^{-1}(\eta + \log(\Delta t)) = 1 - \exp(-\exp(\eta) \cdot \Delta t)$
- $\rightarrow 1 - (1 - \mu_0)^{\Delta t}$

in R

- “family” functions contain all of the components needed for GLM fitting, prediction, etc.
- some of the components are weird (e.g. `$aic`)
- canonical link is used by default

```
names(binomial())
```

```
[1] "family"      "link"        "linkfun"     "linkinv"     "variance"
[6] "dev.resids"  "aic"         "mu.eta"      "initialize"   "validmu"
[11] "valideta"    "simulate"    "dispersion"
```

Offsets

- allow for differential search effort, ratios, etc.
- typically add $\log(e)$
- e.g. $y \sim \text{Poisson}(X\beta + \log(A))$ is equivalent to modeling the response y/A , but without messing up the mean-variance relationship

Offset/link tricks

- fit an exponential curve with constant variance: `family = gaussian(link = "log")`
- Ricker function $y = ax \exp(-bx)$: log-link, `y ~ x + offset(log(x))`
- Michaelis-Menten $y = ax/(b + x) \rightarrow 1/y = (b/a) \cdot (1/x) + 1/a$: inverse-link, `y ~ I(1/x)`

Model interpretation, visualization, testing

Parameter interpretation

- log scale: easy
- logit scale: $\approx \log$ for low baseline, $\approx \log(1 - x)$ for high baseline, slope $\beta/4$ for intermediate values
- cloglog: **log-hazard** scale

Inference

- Wald tests (no finite-size corrections!)
- approximate Wald CIs (compute then back-transform)
- profile CIs

Overdispersion (diagnosis)

- too much variance
- SSQ of Pearson residuals $\sim \chi^2(n - p)$
 - but note Venables and Ripley (2002) p. 209 caution that this is **approximate**
 - simulation-based tests (e.g. DHARMA)

Overdispersion (solutions)

- quasi-likelihood (also handles **underdispersion**)
- compounded models (negative binomial, beta-binomial)
- observation-level random effects (== lognormal-Poisson)

Extended distributions

- VGAM, glmmTMB packages

Complete separation

- there is some linear combination of predictors that separates 0 from 1 responses (or 0 from non-zero responses in the case of count models)
- infinite MLE
- Hauck-Donner effect screws up Wald tests
- likelihood ratio tests still OK (sort of)
- Firth logistic regression (brglm2 package), Bayesian priors (arm::bayesglm)

Zero-inflation/hurdle models

- finite mixture models

Most common GLM problems

- binomial/Poisson models with non-integer data
- failing to specify family (default Gaussian: \rightarrow linear model); using glm() for linear models (unnecessary)
- predictions on effect scale
- using (k, N) rather than $(k, N - k)$ with family=binomial
- back-transforming SEs rather than CIs

- neglecting overdispersion
- Poisson for *underdispersed* responses
- equating negative binomial with binomial rather than Poisson
- worrying about overdispersion unnecessarily (binary/Gamma)
- ignoring random effects

Overdispersion in Bernoulli models?

I think many analysts read that binary models cannot be overdispersed and just do not question it. This happened with the deviance dispersion being the appropriate statistic to measure count model extra-dispersion. Some analysts simply took this on faith, so to speak. But they were mistaken.

Joseph Hilbe (2013)

References

- Dobson, Annette J., and Adrian Barnett. 2008. *An Introduction to Generalized Linear Models, Third Edition*. 3rd ed. Chapman; Hall/CRC.
- Faraway, Julian J. 2016. *Extending the Linear Model with R: Generalized Linear, Mixed Effects and Nonparametric Regression Models, Second Edition*. CRC Press.
- Gelman, Andrew. 2011. "Whassup with Glm()?" *Statistical Modeling, Causal Inference, and Social Science*. https://statmodeling.stat.columbia.edu/2011/05/04/whassup/_with/_gl/.
- Jørgensen, Bent. 1987. "Exponential Dispersion Models." *Journal of the Royal Statistical Society: Series B (Methodological)* 49 (2): 127–45. <https://doi.org/10.1111/j.2517-6161.1987.tb01685.x>.
- Marschner, Ian C. 2011. "Glm2: Fitting Generalized Linear Models with Convergence Problems." *The R Journal* 3 (2): 12–15. http://journal.r-project.org/archive/2011-2/RJournal_2011-2_Marschner.pdf.
- McCullagh, P., and J. A. Nelder. 1989. *Generalized Linear Models*. London: Chapman & Hall.
- Mount, John. 2012. "How Robust Is Logistic Regression?" *Win Vector LLC*. <https://win-vector.com/2012/08/23/how-robust-is-logistic-regression/>.
- Myers, Raymond H., Douglas C. Montgomery, G. Geoffrey Vining, and Timothy J. Robinson. 2010. "Appendix A.6: Computational Details for GLMs for a Noncanonical Link." In *Generalized Linear Models*, 481–83. John Wiley & Sons, Inc. <http://onlinelibrary.wiley.com/doi/10.1002/9780470556986.app6/summary>.
- Robinson, Andrew. 2010. "Why Do We Make a Big Fuss about Using Fisher Scoring When We Fit a GLM?" *Cross Validated*. <https://stats.stackexchange.com/q/205>.
- Venables, W., and Brian D. Ripley. 2002. *Modern Applied Statistics with S*. 4th ed. New York: Springer.

Wood, Simon N. 2017. *Generalized Additive Models: An Introduction with R*. CRC Texts in Statistical Science. Chapman & Hall.