Introduction to Generalized Linear Models

Heather Turner

ESRC National Centre for Research Methods, UK
and
Department of Statistics
University of Warwick, UK

WU, 2008–04–22-24

Introduction

This short course provides an overview of generalized linear models (GLMs).

We shall see that these models extend the linear modelling framework to variables that are not Normally distributed.

GLMs are most commonly used to model binary or count data, so we will focus on models for these types of data.

Plan

Part I: Introduction to Generalized Linear Models

Part II: Binary Data

Part III: Count Data

Part I: Introduction

Review of Linear Models

Generalized Linear Models

GLMs in R

Exercises

Part II: Binary Data

Binary Data

Models for Binary Data

Model Selection

Model Evaluation

Exercises

Part III: Count Data

Count Data

Modelling Rates

Modelling Contingency Tables

Exercises

Part I

Introduction to Generalized Linear Models

The General Linear Model

In a **general linear model**

$$y_i = \beta_0 + \beta_1 x_{1i} + ... + \beta_p x_{pi} + \epsilon_i$$

the **response** $y_i, i = 1, \ldots, n$ is modelled by a linear function of **explanatory** variables $x_j, j = 1, \ldots, p$ plus an error term.

General and Linear

Here **general** refers to the dependence on potentially more than one explanatory variable, v.s. the **simple linear model**:

$$y_i = \beta_0 + \beta_1 x_i + \epsilon_i$$

The model is *linear in the parameters*, e.g.

$$y_i = \beta_0 + \beta_1 x_1 + \beta_2 x_1^2 + \epsilon_i$$
$$y_i = \beta_0 + \gamma_1 \delta_1 x_1 + \exp(\beta_2) x_2 + \epsilon_i$$

but not e.g.

$$y_i = \beta_0 + \beta_1 x_1^{\beta_2} + \epsilon_i$$
$$y_i = \beta_0 \exp(\beta_1 x_1) + \epsilon_i$$

Error structure

(i.i.d)

We assume that the errors ϵ_i are independent and identically distributed such that

$$E[\epsilon_i] = 0$$

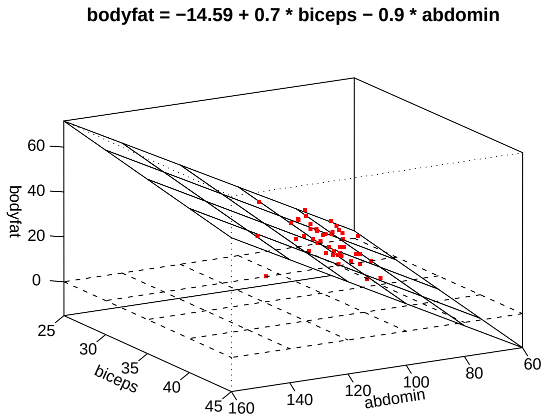
$$\text{and} \;\; \mathrm{var}[\epsilon_i] = \sigma^2$$

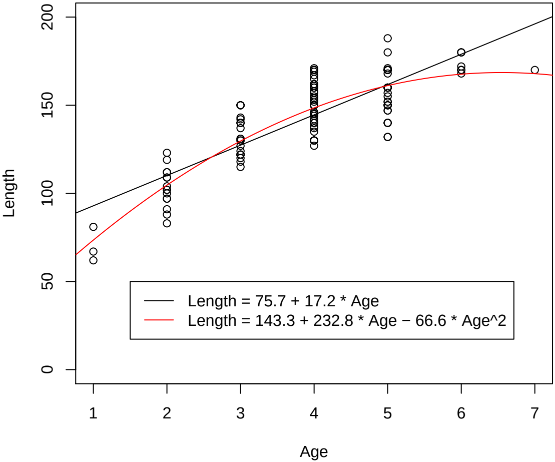
Typically we assume

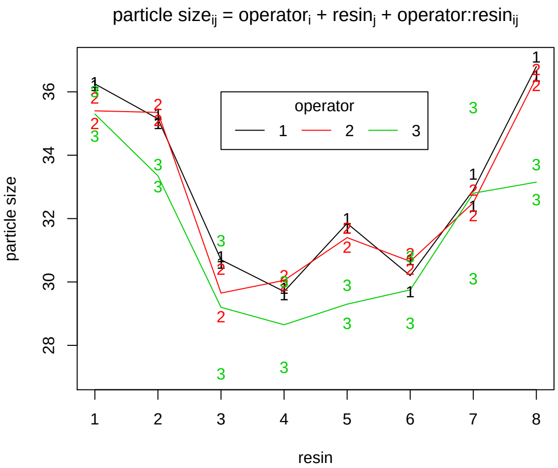
$$\epsilon_i \sim N(0, \sigma^2)$$

as a basis for inference, e.g. t-tests on parameters.

Some Examples




Length
0
50
100
150
200
1
2
3
4
5
6
7
Age
Length = 75.7 + 17.2 * Age
Length = 143.3 + 232.8 * Age − 66.6 * Age^2


particle size$_{ij}$ = operator$_i$ + resin$_j$ + operator:resin$_{ij}$
operator
1
2
3
particle size
28
30
32
34
36
resin
1
2
3
4
5
6
7
8

Restrictions of Linear Models

Although a very useful framework, there are some situations where general linear models are not appropriate

- the range of Y is restricted (e.g. binary, count)
- the variance of Y depends on the mean

Generalized linear models extend the general linear model framework to address both of these issues

Generalized Linear Models (GLMs)

A **generalized linear model** is made up of a **linear predictor**

$$\eta_i = \beta_0 + \beta_1 x_{1i} + \ldots + \beta_p x_{pi}$$

and two functions

- a **link** function that describes how the mean, $E(Y_i) = \mu_i$, depends on the linear predictor

$$g(\mu_i) = \eta_i$$

- a **variance** function that describes how the variance, $\mathrm{var}(Y_i)$ depends on the mean

$$\mathrm{var}(Y_i) = \phi V(\mu)$$

where the **dispersion parameter** ϕ is a constant

Normal General Linear Model as a Special Case

For the general linear model with $\epsilon \sim N(0, \sigma^2)$ we have the linear predictor

$$\eta_i = \beta_0 + \beta_1 x_{1i} + ... + \beta_p x_{pi}$$

the link function

$$g(\mu_i) = \mu_i$$

and the variance function

$$V(\mu_i) = 1$$

Modelling Binomial Data

(# successes in n trials w/ prob. pi)

Suppose

$$Y_i \sim \text{Binomial}(n_i, p_i)$$

and we wish to model the proportions Y_i/n_i. Then

$$E(Y_i/n_i) = p_i \qquad \text{var}(Y_i/n_i) = \frac{1}{n_i} p_i(1 - p_i)$$

So our variance function is

$$V(\mu_i) = \mu_i(1 - \mu_i)$$

spreads out the y range

↓

Our link function must map from $(0, 1) \rightarrow (-\infty, \infty)$. A common choice is

$$g(\mu_i) = \text{logit}(\mu_i) = \log\left(\frac{\mu_i}{1 - \mu_i}\right)$$

Modelling Poisson Data

Suppose

$$Y_i \sim \textsf{Poisson}(\lambda_i)$$

Then

$$E(Y_i) = \lambda_i \qquad\qquad \mathrm{var}(Y_i) = \lambda_i$$

So our variance function is

$$V(\mu_i) = \mu_i$$

Our link function must map from $(0, \infty) \rightarrow (-\infty, \infty)$. A natural choice is

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

Transformation vs. GLM

In some situations a response variable can be transformed to improve linearity and homogeneity of variance so that a general linear model can be applied.

This approach has some drawbacks

- response variable has changed!
- transformation must simulateneously improve linearity and homogeneity of variance
- transformation may not be defined on the boundaries of the sample space

For example, a common remedy for the variance increasing with the mean is to apply the log transform, e.g.

$$\begin{aligned} \log(y_i) &= \beta_0 + \beta_1 x_1 + \epsilon_i \\ \Rightarrow E(\log Y_i) &= \beta_0 + \beta_1 x_1 \end{aligned}$$

This is a linear model for the mean of $\log Y$ which may not always be appropriate. E.g. if Y is income perhaps we are really interested in the mean income of population subgroups, in which case it would be better to model $E(Y)$ using a glm :

$$\log E(Y_i) = \beta_0 + \beta_1 x_1$$

with $V(\mu) = \mu$. This also avoids difficulties with $y = 0$.

Exponential Family

Most of the commonly used statistical distributions, e.g. Normal, Binomial and Poisson, are members of the **exponential family of distributions** whose densities can be written in the form

$$f(y;\theta,\phi) = \exp\left\{\frac{y\theta - b(\theta)}{\phi + c(y,\phi)}\right\}$$

where ϕ is the dispersion parameter and θ is the **canonical parameter**.

It can be shown that

$$\begin{aligned} E(Y) &= b'(\theta) = \mu \\ \text{and} \quad \mathrm{var}(Y) &= \phi b''(\theta) = \phi V(\mu) \end{aligned}$$

Canonical Links

y_i

For a glm where the response follows an exponential distribution we have

$$g(\mu_i) = g(b'(\theta_i)) = \beta_0 + \beta_1 x_{1i} + ... + \beta_p x_{pi}$$

The **canonical link** is defined as

(just 1 choice)

1.

$$g = (b')^{-1}$$
$$\Rightarrow g(\mu_i) = \theta_i = \beta_0 + \beta_1 x_{1i} + ... + \beta_p x_{pi}$$

Canonical links lead to desirable statistical properties of the glm hence tend to be used by default. However there is no *a priori* reason why the systematic effects in the model should be additive on the scale given by this link.

Estimation of the Model Parameters

A single algorithm can be used to estimate the parameters of an exponential family glm using maximum likelihood.

The log-likelihood for the sample $y_1, \dots, y_n$ is

$$l = \sum_{i=1}^{n} \frac{y_i \theta_i - b(\theta_i)}{\phi_i} + c(y_i, \phi_i)$$

The maximum likelihood estimates are obtained by solving the score equations

$$s(\beta_j) = \frac{\partial l}{\partial \beta_j} = \sum_{i=1}^{n} \frac{y_i - \mu_i}{\phi_i V(\mu_i)} \times \frac{x_{ij}}{g'(\mu_i)} = 0$$

for parameters β_j.

We assume that

$$\phi_i = \frac{\phi}{a_i}$$

where ϕ is a single dispersion parameter and a_i are known **prior weights**; for example binomial proportions with known index n_i have $\phi = 1$ and $a_i = n_i$.

The estimating equations are then

$$\frac{\partial l}{\partial \beta_j} = \sum_{i=1}^{n} \frac{a_i(y_i - \mu_i)}{V(\mu_i)} \times \frac{x_{ij}}{g'(\mu_i)} = 0$$

which does not depend on ϕ (which may be unknown).

A general method of solving score equations is the iterative algorithm **Fisher's Method of Scoring** (derived from a Taylor's expansion of $s(\boldsymbol{\beta})$)

In the r-th iteration , the new estimate $\boldsymbol{\beta}^{(r+1)}$ is obtained from the previous estimate $\boldsymbol{\beta}^{(r)}$ by

$$\boldsymbol{\beta}^{(r+1)} = \boldsymbol{\beta}^{(r)} + s\left(\boldsymbol{\beta}^{(\boldsymbol{r})}\right) E\left(H\left(\boldsymbol{\beta}^{(\boldsymbol{r})}\right)\right)^{-1}$$

where H is the **Hessian matrix**: the matrix of second derivatives of the log-likelihood.

It turns out that the updates can be written as

$$\boldsymbol{\beta}^{(r+1)} = \left(X^T W^(r) X\right)^{-1} X^T W^{(r)} \boldsymbol{z}^{(r)}$$

i.e. the score equations for a weighted least squares regression of $\boldsymbol{z}^{(r)}$ on $\boldsymbol{X}$ with weights $W^{(r)} = diag(w_i)$, where

$$z_i^{(r)} = \eta_i^{(r)} + \left(y_i - \mu_i^{(r)}\right) g'\left(\mu_i^{(r)}\right)$$

$$\text{and} \quad w_i^{(r)} = \frac{a_i}{V\left(\mu_i^{(r)}\right)\left(g'\left(\mu_i^{(t)}\right)\right)^2}$$

Hence the estimates can be found using an **Iteratively (Re-)Weighted Least Squares** algorithm:

1. Start with initial estimates $\mu_i^{(r)}$
2. Calculate **working responses** $z_i^{(r)}$ and **working weights** $w_i^{(r)}$
3. Calculate $\boldsymbol{\beta}^{(r+1)}$ by weighted least squares
4. Repeat 2 and 3 till convergence

For models with the canonical link, this is simply the Newton-Raphson method.

Standard Errors

The estimates $\hat{\boldsymbol{\beta}}$ have the usual properties of maximum likelihood estimators. In particular, $\hat{\boldsymbol{\beta}}$ is asymptotically

$$N(\boldsymbol{\beta}, i^{-1})$$

where

$$i(\boldsymbol{\beta}) = \phi^{-1} X^T W X$$

Standard errors for the β_j may therefore be calculated as the square roots of the diagonal elements of

$$\hat{\mathrm{cov}}(\hat{\boldsymbol{\beta}}) = \phi(X^T \hat{W} X)^{-1}$$

in which $(X^T \hat{W} X)^{-1}$ is a by-product of the final IWLS iteration.

If ϕ is unknown, an estimate is required.

There are practical difficulties in estimating the dispersion ϕ by maximum likelihood.

Therefore it is usually estimated by **method of moments**. If $\boldsymbol{\beta}$ was known an unbiased estimate of $\phi = \{a_i \operatorname{var}(Y)\}/v(\mu_i)$ would be

$$\frac{1}{n}\sum_{i=1}^{n}\frac{a_i(y_i-\mu_i)^2}{V(\mu_i)}$$

Allowing for the fact that $\boldsymbol{\beta}$ must be estimated we obtain

$$\star \qquad \frac{1}{n-p}\sum_{i=1}^{n}\frac{a_i(y_i-\mu_i)^2}{V(\mu_i)} \quad = \phi$$

The glm Function

Generalized linear models can be fitted in R using the glm function, which is similar to the lm function for fitting linear models.

The arguments to a glm call are as follows

```
glm(formula, family = gaussian, data, weights, subset,
    na.action, start = NULL, etastart, mustart, offset,
    control = glm.control(...), model = TRUE,
    method = ”glm.fit”, x = FALSE, y = TRUE,
    contrasts = NULL, ...)
```

Formula Argument

The formula is specified to glm as, e.g.

```
y ~ x1 + x2
```

where `x1, x2` are the names of

- numeric vectors (continuous variables)
- factors (categorical variables)

All specified variables must be in the workspace or in the data frame passed to the `data` argument.

Other symbols that can be used in the formula include

- `a:b` for an interaction between `a` and `b`
- `a*b` which expands to `a + b + a:b`
- `.` for first order terms of all variables in `data`
- `-` to exclude a term or terms
- `1` to include an intercept (included by default)
- `0` to exclude an intercept

Family Argument

The `family` argument takes (the name of) a family function which specifies

- the link function
- the variance function
- various related objects used by `glm`, e.g. `linkinv`

The exponential family functions available in R are

- `binomial(link = "logit")`
- `gaussian(link = "identity")`
- `Gamma(link = "inverse")`
- `inverse.gaussian(link = "1/mu`^{2}`")`
- `poisson(link = "log")`

Extractor Functions

The `glm` function returns an object of class `c("glm", "lm")`.

There are several `glm` or `lm` methods available for accessing/displaying components of the `glm` object, including:

- `residuals()`
- `fitted()`
- `predict()`
- `coef()`
- `deviance()`
- `formula()`
- `summary()`

Example: Household Food Expenditure

Griffiths, Hill and Judge (1993) present a dataset on food expenditure for households that have three family members. We consider two variables, the logarithm of expenditure on food and the household income:

```
dat <- read.table("GHJ_food_income.txt", header = TRUE)
attach(dat)
plot(Food ~ Income, xlab = "Weekly Household Income ($)",
    ylab = "Weekly Household Expenditure on Food (Log $)")
```

It would seem that a simple linear model would fit the data well.

We will first fit the model using `lm`, then compare to the results using `glm`.

```
foodLM <- lm(Food ~ Income)
summary(foodLM)
foodGLM <- glm(Food ~ Income)
summary(foodGLM)
```

Summary of Fit Using lm

```
Call:
lm(formula = Food ~ Income)

Residuals:
      Min        1Q    Median        3Q       Max
-0.508368 -0.157815 -0.005357  0.187894  0.491421

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 2.409418   0.161976  14.875  < 2e-16 ***
Income      0.009976   0.002234   4.465 6.95e-05 ***
---
Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1

Residual standard error: 0.2766 on 38 degrees of freedom
Multiple R-squared: 0.3441,Adjusted R-squared: 0.3268
F-statistic: 19.94 on 1 and 38 DF,  p-value: 6.951e-05
```

Summary of Fit Using glm

The default family for glm is "gaussian" so the arguments of the call are unchanged.

A five-number summary of the **deviance residuals** is given. Since the response is assumed to be normally distributed these are the same as the residuals returned from lm.

```
Call:
glm(formula = Food ~ Income)

Deviance Residuals:
      Min         1Q     Median         3Q        Max
-0.508368  -0.157815  -0.005357   0.187894   0.491421
```

The estimated coefficients are unchanged

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 2.409418   0.161976  14.875  < 2e-16 ***
Income      0.009976   0.002234   4.465 6.95e-05 ***
---
Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1

(Dispersion parameter for gaussian family taken to be 0.07650739
```

Partial t-tests test the significance of each coefficient in the presence of the others. The dispersion parameter for the gaussian family is equal to the residual variance.

Wald Tests

For non-Normal data, we can use the fact that asymptotically

$$\hat{\boldsymbol{\beta}} \sim N(\boldsymbol{\beta}, \phi(\boldsymbol{X}'\boldsymbol{W}\boldsymbol{X})^{-1})$$

and use a z-test to test the significance of a coefficient.

Specifically, we test

$$H_0 : \beta_j = 0 \qquad \text{versus} \qquad H_1 : \beta_j \neq 0$$

using the test statistic

$$z_j = \frac{\hat{\beta}_j}{\sqrt{\phi}(\boldsymbol{X}'\hat{\boldsymbol{W}}\boldsymbol{X})^{-1}_{jj}}$$

which is asymptotically $N(0,1)$ under H_0.

Different model summaries are reported for GLMs. First we have the **deviance** of two models:

```
    Null deviance: 4.4325  on 39  degrees of freedom
Residual deviance: 2.9073  on 38  degrees of freedom
```

The first refers to the **null model** in which all of the terms are excluded, except the intercept if present. The degrees of freedom for this model are the number of data points n minus 1 if an intercept is fitted.

The second two refer to the fitted model, which has $n - p$ degees of freedom, where p is the number of parameters, including any intercept.

Deviance

The deviance of a model is defined as

$$D = 2\phi(l_{sat} - l_{mod})$$

where l_{mod} is the log-likelihood of the fitted model and l_{sat} is the log-likelihood of the **saturated model**.

In the saturated model, the number of parameters is equal to the number of observations, so $\hat{y} = y$.

For linear regression with Normal data, the deviance is equal to the residual sum of squares.

Akiake Information Criterion (AIC)

Finally we have:

```
AIC: 14.649

Number of Fisher Scoring iterations: 2
```

The AIC is a measure of fit that penalizes for the number of parameters p

$$AIC = -2l_{mod} + 2p$$

Smaller values indicate better fit and thus the AIC can be used to compare models (not necessarily nested).

Residual Analysis

Several kinds of residuals can be defined for GLMs:

- **response**: $y_i - \hat{\mu}_i$
- **working**: from the working response in the IWLS algorithm
- **Pearson**
$$r_i^P = \frac{y_i - \hat{\mu}_i}{\sqrt{V(\hat{\mu}_i)}}$$
  s.t. $\sum_i (r_i^P)^2$ equals the generalized Pearson statistic
- **deviance** r_i^D s.t. $\sum_i (r_i^D)^2$ equals the deviance

These definitions are all equivalent for Normal models.

Deviance residuals are the default used in R, since they reflect the same criterion as used in the fitting.

For example we can plot the deviance residuals against the fitted values (on the response scale) as follows:

```
plot(residuals(foodGLM) ~ fitted(foodGLM),
    xlab = expression(hat(y)[i]),
    ylab = expression(r[i]))
abline(0, 0, lty = 2)
```

The plot function gives the usual choice of residual plots, based on the deviance residuals. By default

- deviance residuals v. fitted values
- Normal Q-Q plot of deviance residuals standardised to unit variance
- scale-location plot of standardised deviance residuals
- standardised deviance residuals v. leverage with Cook's distance contours

Residual Plots

For the food expenditure data the residuals do not indicate any problems with the modelling assumptions:

```
plot(foodGLM)
```

Exercises

1. Load the SLID data from the car package and attach the data frame to the search path. Look up the description of the SLID data in the help file.

In the following exercises you will investigate models for the wages variable.

2. Produce appropriate plots to examine the bivariate relationships of wages with the other variables in the data set. Which variables appear to be correlated with wages?

3. Use lm to regress wages on the linear effect of the other variables. Look at a summary of the fit. Do the results appear to agree with your exploratory analysis? Use plot to check the residuals from the fit. Which modelling assumptions appear to be invalid?

4. Repeat the analysis of question 3 with `log(wages)` as the response variable. Confirm that the residuals are more consistent with the modelling assumptions. Can any variables be dropped from the model?

Investigate whether two-way and three-way interactions should be added to the model.

In the analysis of question 4, we have estimated a model of the form

$$\log y_i = \beta_0 + \sum_{r=1}^{p} \beta_r x_{ir} + \epsilon_i \tag{1}$$

which is equivalent to

$$y_i = \exp\left(\beta_0^* + \sum_{r=1}^{p} \beta_r x_{ir}\right) \times \epsilon_i^* \tag{2}$$

where $\epsilon_i = \log(\epsilon_i^*) - E(\log \epsilon_i^*)$.

Assuming ϵ_i to be normally distributed in Equation 1 implies that $\log(Y)$ is normally distributed. If $X = \log(Y) \sim N(\mu, \sigma^2)$, then Y has a log-Normal distribution with parameters μ and σ^2. It can be shown that

$$\begin{aligned} E(Y) &= \exp\left(\mu + \frac{1}{2}\sigma^2\right) \\ \text{var}(Y) &= \left\{\exp(\sigma^2) - 1\right\}\left\{E(Y)\right\}^2 \end{aligned}$$

so that

$$\text{var}(Y) \propto \{E(Y)\}^2$$

An alternative approach is to assume that Y has a Gamma distribution, which is the exponential family with this mean-variance relationship. We can then model $E(Y)$ using a GLM. The canonical link for Gamma data is $1/\mu$, but Equation 2 suggests we should use a log link here.

5. Use `gnm` to fit a Gamma model for `wages` with the same predictor variables as your chosen model in question 4. Look at a summary of the fit and compare with the log-Normal model – Are the inferences the same? Are the parameter estimates similar? Note that t statistics rather than z statistics are given for the parameters since the dispersion ϕ has had to be estimated.

6. (Extra time!) Go back and fit your chosen model in question 4 using `glm`. How does the deviance compare to the equivalent Gamma model? Note that the AIC values are not comparable here: constants in the likelihood functions are dropped when computing the AIC, so these values are only comparable when fitting models with the same error distribution.