MAST90104 - Lecture 8

Weichang Yu

Room 108, Old Geology South Bldg
School of Mathematics and Statistics, University of Melbourne

Generalised least squares

In Lecture 4 Part 1, we adopt assumptions (III) and (IV): $\mathbb{E}(\boldsymbol{\epsilon}) = \mathbf{0}$ and $Var(\boldsymbol{\epsilon}) = \sigma^2\mathbf{I}$.

What if the variance of $\boldsymbol{\epsilon}$ is not $\sigma^2\mathbf{I}$, i.e., assumption (IV) is violated?

Generalised least squares

We resolve the issue by converting our problem back to a paradigm where asumption (IV) holds.

Assume (I), (II), and (III) holds. Original regression equations:

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}, \text{ where } Var(\boldsymbol{\epsilon}) = \mathbf{V}.$$

Consider the matrix square root $\mathbf{V}^{1/2}$. Note that $\mathbf{V}^{-1/2}$ exists. Let $\widetilde{\boldsymbol{\epsilon}} = \mathbf{V}^{-1/2}\boldsymbol{\epsilon}$, $\widetilde{\mathbf{X}} = \mathbf{V}^{-1/2}\mathbf{X}$ and $\widetilde{\mathbf{y}} = \mathbf{V}^{-1/2}\mathbf{y}$.

Then, we have the new transformed regression equations:

$$\widetilde{\mathbf{y}} = \widetilde{\mathbf{X}}\boldsymbol{\beta} + \widetilde{\boldsymbol{\epsilon}},$$

Generalised least squares

We can check that the transformed regression model satisfy assumptions (I) to (IV). To check (I):

If $\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}$, then $\widetilde{\mathbf{y}} = \widetilde{\mathbf{X}}\boldsymbol{\beta} + \widetilde{\boldsymbol{\epsilon}}$. Hence, $\widetilde{\mathbf{y}}$ is linearly related to $\widetilde{\mathbf{X}}$.

To check (II), we note that since $\mathbf{V}$ is symmetric, there exists a orthogonal $\mathbf{P}$ such that

$$\mathbf{P}^T\mathbf{V}\mathbf{P} = \boldsymbol{\Lambda}$$

or equivalently

$$\mathbf{V} = \mathbf{P}\boldsymbol{\Lambda}\mathbf{P}^T,$$

where $\boldsymbol{\Lambda}$ is a diagonal matrix of eigenvalues of $\mathbf{V}$.

Generalised least squares

Moreover, by the positive-definiteness of $\mathbf{V}$, it can be shown that all eigenvalues of $\mathbf{V}$ are strictly positive.

Hence, by the definition of matrix square root, we can write

$$\mathbf{V}^{1/2} = \mathbf{P}\boldsymbol{\Lambda}^{1/2}\mathbf{P}^T$$

Consequently,

$$\begin{aligned} |\mathbf{V}^{1/2}| &= |\mathbf{P}\boldsymbol{\Lambda}^{1/2}\mathbf{P}^T| \\ &= |\boldsymbol{\Lambda}^{1/2}| \underbrace{|\mathbf{P}|^2}_{=1} \\ &= |\boldsymbol{\Lambda}^{1/2}| \\ &= \prod_{i=1}^{n} \sqrt{\lambda_i} > 0 \end{aligned}$$

Hence, $|\mathbf{V}^{-1/2}| = 1/|\mathbf{V}^{1/2}| > 0$.

Generalised least squares

To show that $\widetilde{\mathbf{X}}$ is full rank, we first note that $r(\widetilde{\mathbf{X}}) = r(\mathbf{V}^{-1/2}\mathbf{X}) \leq r(\mathbf{X})$.

On the other hand, $|\mathbf{V}^{-1/2}| \neq 0$ implies $r(\mathbf{V}^{-1/2}) = n$.

Then by Sylvester's rank inequality:

$$r(\widetilde{\mathbf{X}}) \geq r(\mathbf{X}) + \underbrace{r(\mathbf{V}^{-1/2}) - n}_{=0}.$$

and therefore $r(\widetilde{\mathbf{X}}) = r(\mathbf{X})$. Hence $\widetilde{\mathbf{X}}$ is full rank.

Generalised least squares

To check assumption (III): $\mathbb{E}(\widetilde{\boldsymbol{\epsilon}}) = \mathbf{V}^{-1/2}\mathbf{0} = \mathbf{0}$.

To check assumption (IV): $Var(\widetilde{\boldsymbol{\epsilon}}) = \mathbf{V}^{-1/2}\mathbf{V}\mathbf{V}^{-1/2} = \mathbf{I}$.

Generalised least squares

Since assumptions (I) to (IV) are satisfied in the transformed regression model, we can estimate $\boldsymbol{\beta}$ using the minimiser of the least squares criterion (based on transformed data):

$$\begin{aligned}
\widehat{\boldsymbol{\beta}} &= \underset{\mathbf{b}}{\operatorname{argmin}} (\widetilde{\mathbf{y}} - \widetilde{\mathbf{X}}\mathbf{b})^T (\widetilde{\mathbf{y}} - \widetilde{\mathbf{X}}\mathbf{b}) \\
&= \underset{\mathbf{b}}{\operatorname{argmin}} (\mathbf{V}^{-1/2}\mathbf{y} - \mathbf{V}^{-1/2}\mathbf{X}\mathbf{b})^T (\mathbf{V}^{-1/2}\mathbf{y} - \mathbf{V}^{-1/2}\mathbf{X}\mathbf{b}) \\
&= \underset{\mathbf{b}}{\operatorname{argmin}}\, \widetilde{C}(\mathbf{b}),
\end{aligned}$$

where $\widetilde{C}(\mathbf{b}) = (\mathbf{y} - \mathbf{X}\mathbf{b})^T \mathbf{V}^{-1} (\mathbf{y} - \mathbf{X}\mathbf{b})$ is the *generalised least squares criterion*.

Using matrix calculus, we can show that

$$\widehat{\boldsymbol{\beta}} = (\mathbf{X}^T \mathbf{V}^{-1} \mathbf{X})^{-1} \mathbf{X}^T \mathbf{V}^{-1} \mathbf{y}.$$

Generalised least squares

We have

$$\begin{aligned} \mathbb{E}[\widehat{\boldsymbol{\beta}}] &= \boldsymbol{\beta}, \\ \mathrm{Var}(\widehat{\boldsymbol{\beta}}) &= (\mathbf{X}^T\mathbf{V}^{-1}\mathbf{X})^{-1}. \end{aligned}$$

Moreover, it can be shown that the Gauss-Markov theorem still holds, i.e. the generalised least squares estimator is still BLUE.

The proof is left as an exercise.

Weighted least squares

In some situations, the errors are uncorrelated but do not have a common variance:

$$Var(\boldsymbol{\epsilon}) = diag(\sigma_1^2, \sigma_2^2, \ldots, \sigma_n^2).$$

The generalised least squares criterion simplifies:

$$\widetilde{C}(\mathbf{b}) = \sum_{i=1}^{n} \left(\frac{y_i - \mathbf{x}_i^T \mathbf{b}}{\sigma_i} \right)^2.$$

That is, we *weight* each residual by the inverse of the corresponding standard deviation. So a point with high variance influences $\widehat{\boldsymbol{\beta}}$ less than a point with low variance.

Caveat: In practice, the standard deviations σ_i's are unknown. We will need to replace them with some preliminary estimates.

Let's switch gear....back to generalised linear models

Exponential families

Y comes from an exponential family if it has a probability density or probability mass function (pdf or pmf) of the form

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

where

θ is the *canonical parameter* (captures location)

ϕ is the *dispersion parameter* (captures scale).

Observe that it is the log of the density function that must have a specific form involving θ and ϕ.

Example: Gaussian

$Y \sim N(\mu, \sigma^2)$

$$
\begin{aligned}
f(y) &= \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2}\frac{(y-\mu)^2}{\sigma^2}} \\
&= \exp\left[\frac{y\mu - \mu^2/2}{\sigma^2} - \frac{1}{2}\left(\frac{y^2}{\sigma^2} + \log(2\pi\sigma^2)\right)\right] \\
&= \exp\left[\frac{y\theta - b(\theta)}{a(\phi)} + c(y, \phi)\right]
\end{aligned}
$$

where $\theta = \mu$, $\phi = \sigma^2$, and

$$
\begin{aligned}
b(\theta) &= \theta^2/2 \\
a(\phi) &= \phi \\
c(y, \phi) &= -\frac{1}{2}\left(\frac{y^2}{\phi} + \log(2\pi\phi)\right)
\end{aligned}
$$

Example: Poisson

$Y \sim \text{Poisson}(\lambda)$

$$\begin{aligned} f(y) &= e^{-\lambda}\lambda^y/y! \text{ for } y = 0, 1, 2, \ldots \\ &= \exp\left[y \log \lambda - \lambda - \log y!\right] \\ &= \exp\left[\frac{y\theta - b(\theta)}{a(\phi)} + c(y, \phi)\right] \end{aligned}$$

where $\theta = \log \lambda$, $\phi = 1$, and

$$\begin{aligned} b(\theta) &= e^{\theta} \\ a(\phi) &= \phi \\ c(y, \phi) &= -\log y! \end{aligned}$$

Example: Binomial

$Y \sim \text{Binomial}(m, p)$ for known m (not a parameter)

$$
\begin{aligned}
f(y) &= \binom{m}{y} p^y (1-p)^{m-y} \text{ for } y = 0, 1, \ldots, m \\
&= \exp\left[y \log \frac{p}{1-p} + m \log(1-p) + \log \binom{m}{y} \right] \\
&= \exp\left[\frac{y\theta - b(\theta)}{a(\phi)} + c(y, \phi) \right]
\end{aligned}
$$

where $\theta = \log \frac{p}{1-p}$, $\phi = 1$, and

$$
\begin{aligned}
b(\theta) &= m \log(1 + e^\theta) \\
a(\phi) &= \phi \\
c(y, \phi) &= \log \binom{m}{y}
\end{aligned}
$$

Example: Scaled Binomial

$X \sim \text{Binomial}(m, p)$ and $Y = X/m$

$$f(y) = \binom{m}{my} p^{my}(1-p)^{m(1-y)} \text{ for } y = 0, 1/m, \ldots, 1$$

$$= \exp\left[\frac{y \log \frac{p}{1-p} + \log(1-p)}{1/m} + \log \binom{m}{my}\right]$$

$$= \exp\left[\frac{y\theta - b(\theta)}{a(\phi)} + c(y, \phi)\right]$$

where $\theta = \log \frac{p}{1-p}$, $\phi = 1/m$, and

$$b(\theta) = \log(1 + e^{\theta})$$
$$a(\phi) = \phi$$
$$c(y, \phi) = \log \binom{m}{my}$$

Exponential family: mean and variance

Lemma If Y is from an exponential family then

$$\mathbb{E}(Y) = b'(\theta)$$
$$\mathrm{Var}(Y) = b''(\theta)a(\phi)$$

Exponential family: variance function

Let $\mu = \mathbb{E}(Y)$ and write

$$Var(Y) = \psi(\mu)a(\phi)$$

where $\psi = b'' \circ (b')^{-1}$.

ψ is called the *variance function*

Examples:

normal $\psi(\mu) = 1$

Poisson $\psi(\mu) = \mu$

binomial $\psi(\mu) = \mu(1 - \mu/m)$

scaled binomial $\psi(\mu) = \mu(1 - \mu)$

Generalised Linear Model

Definition: Y is said to come from a *Generalised Linear Model (GLM)* if the pdf/pmf is from an exponential family, and

$$\mu := \mathbb{E}(Y) = g^{-1}(\mathbf{x}^T\boldsymbol{\beta})$$

where

- g is a monotonic differentiable function called the *link function*.
- $\mathbf{x}$ is a vector of predictor-variables, and
- $\boldsymbol{\beta}$ is a vector of parameters

Remark: We model *location* using $\eta = \mathbf{x}^T\boldsymbol{\beta}$, and let the *scale* be determined by the family of distributions.

That is, we do not model the scale explicitly in terms of predictor-variables.

Canonical link

Recall Y is from an exponential family if

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

If $g(\mu) = g(\mathbb{E}(Y)) = \theta$ then g is called the *canonical* link.

Since $\mu = b'(\theta)$, it follows that the canonical link must be $(b')^{-1}$.

Examples: canonical links

normal $\theta = \mu,\ g(\mu) = \mu$

Poisson $\theta = \log \lambda = \log \mu,\ g(\mu) = \log \mu$

Binomial $\theta = \log \frac{p}{1-p} = \log \frac{\mu}{m-\mu},\ g(\mu) = \log \frac{\mu}{m-\mu}$

scaled Binomial $\theta = \log \frac{p}{1-p} = \log \frac{\mu}{1-\mu},\ g(\mu) = \log \frac{\mu}{1-\mu}$

GLM estimation

We fit GLMs using maximum likelihood.

Suppose we have independent observations y_i from an exponential family, with canonical parameter θ_i and dispersion parameter ϕ, for $i = 1, \ldots, n$.

Furthermore suppose that y_i has mean

$$\mu_i = b'(\theta_i) = g^{-1}(\mathbf{x}_i^T \boldsymbol{\beta})$$

If g is the canonical link then $\theta_i = \mathbf{x}_i^T \boldsymbol{\beta}$, so the multiplier of y in the log pdf/pmf is the linear predictor $\mathbf{x}_i^T \boldsymbol{\beta}$.

The log-likelihood is then

$$\ell(\boldsymbol{\beta}, \phi; \mathbf{y}) = \sum_{i=1}^{n} \left(\frac{y_i \theta_i - b(\theta_i)}{a(\phi)} + c(y_i, \phi) \right)$$

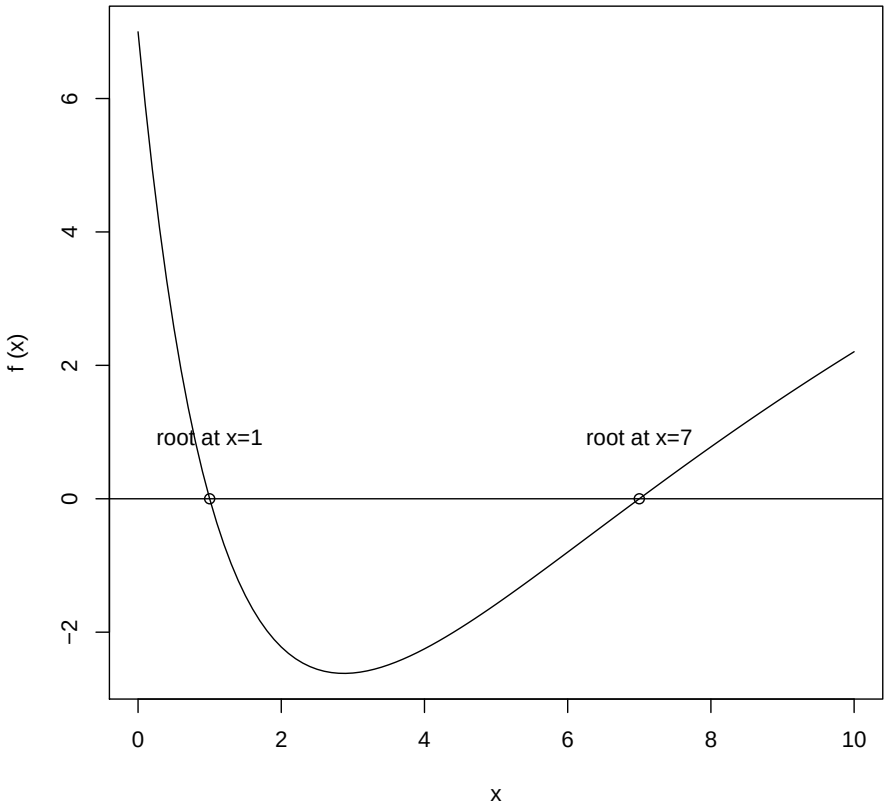
To find the maximum likelihood estimate, the log-likelihood needs to be differentiated and the result equated to zero.

This requires a procedure which works well for our generalised linear models.

Fortunately one is available using Newton's method.

Newton's method is discussed in the next slides.

Suppose that $f : \mathbb{R} \to \mathbb{R}$ is a continuous function. A *root* of f is a solution to the equation $f(x) = 0$



Newton's method for computing roots

Suppose the function f is differentiable with continuous derivative f' and at root a.
Let $x^{(0)} \in \mathbb{R}$ and think of $x^{(0)}$ as our current 'guess' at a.
The tangent line of f at point $(x^{(0)}, f(x^{(0)}))$ with slope $f'(x^{(0)})$ is the local straight line approximation to the function $f(x)$.
The equation of this tangent line is given by

$$y = f(x^{(0)}) + f'(x^{(0)})(x - x^{(0)}) = f(x^{(0)}) - f'(x^{(0)})x^{(0)} + \underbrace{f'(x^{(0)})}_{=\text{slope}} x.$$

This tangent line crosses the x-axis at a point $x^{(1)}$. This new $x^{(1)}$ is regarded as a better approximation than $x^{(0)}$ to a.
To find $x^{(1)}$ transform the previous equation:

$$f'(x^{(0)}) = \frac{f(x^{(0)}) - 0}{x^{(0)} - x^{(1)}} \quad \text{and so} \quad x^{(1)} = x^{(0)} - \frac{f(x^{(0)})}{f'(x^{(0)})}.$$

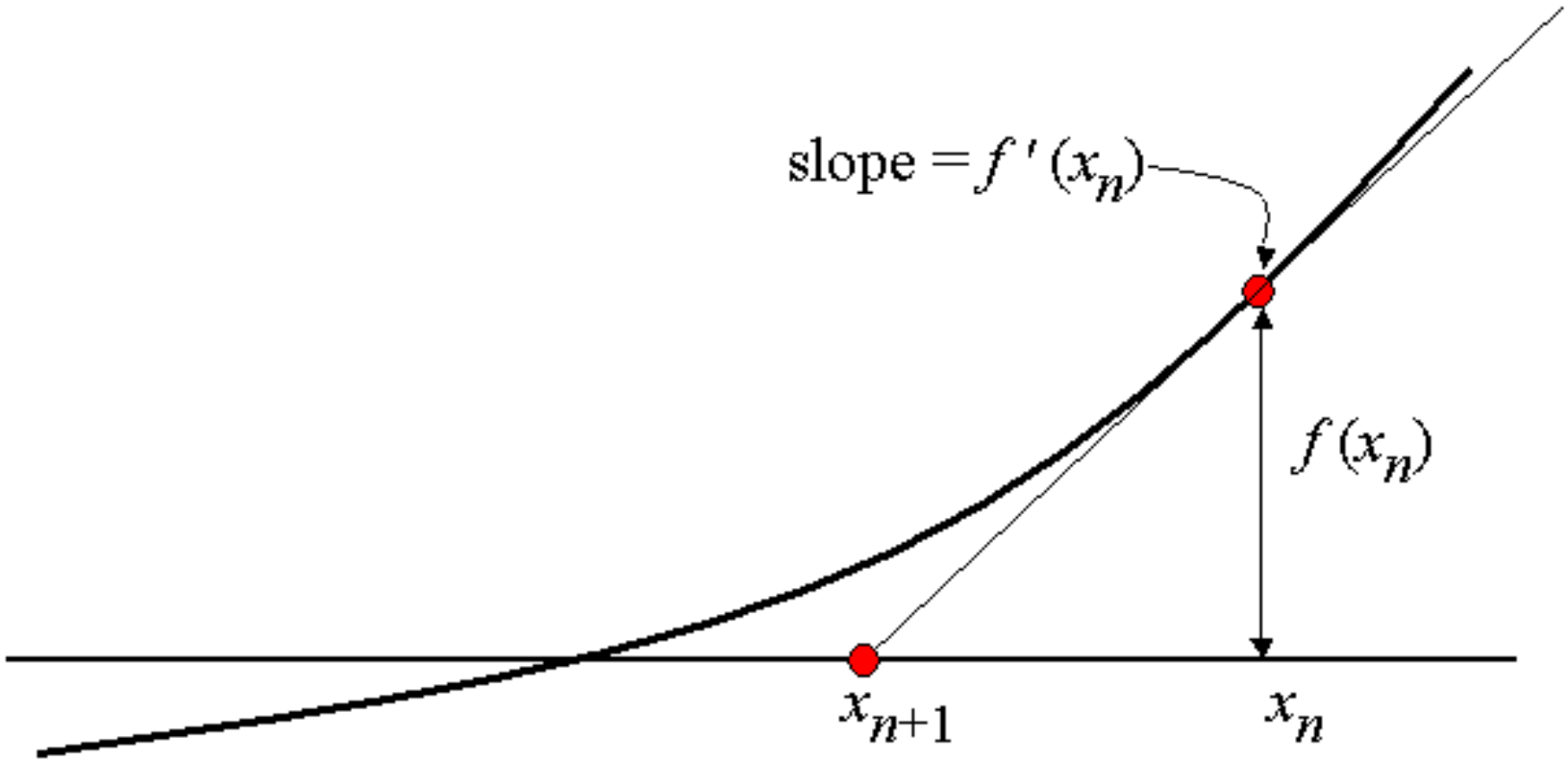


Figure: Illustration of one step in Newton's method

Newton's method for computing roots

A helpful animated illustration is readily available on wikipedia!

The final algorithm in summary

1. INPUT function f, initial guess $x^{(0)}$, and absolute tolerance α.
2. At iteration $t = 0, 1, 2, \ldots$, WHILE $|f(x^{(t)})| > \alpha$
   - Compute x-intercept of new tangent line:

$$x^{(t+1)} = x^{(t)} - \frac{f(x^{(t)})}{f'(x^{(t)})}.$$

3. Root $= x^{(t)}$.

Newton's method for computing stationary points

We can also apply the Newton's method for finding stationary points

The current approximation to the stationary point $x^{(t)}$ generates the next approximation via:

$$x^{(t+1)} = x^{(t)} - \frac{f'(x^{(t)})}{f''(x^{(t)})}.$$

Convergence is not guaranteed and the second derivative needs to be non-zero.

Newton's method - higher dimensions

The one dimensional Newton's method extends to functions which have more than one variable - like our log-likelihood.

Suppose we have a vector of variables $\mathbf{x} = (x_1, x_2, \ldots, x_k)^T$, and some scalar function of them:

$$z = f(\mathbf{x}).$$

Recall that the derivative of z with respect to $\mathbf{y}$ is defined as:

$$\frac{\partial z}{\partial \mathbf{x}} = (\partial z/\partial x_1, \partial z/\partial x_2, \ldots, \partial z/\partial x_k).$$

Newton's method - higher dimensions

Further, let $\mathbf{H}$ be the $k \times k$ *Hessian* matrix defined by:

$$\mathbf{H} = \begin{pmatrix} \frac{\partial^2 z}{\partial x_1 \partial x_1} & \frac{\partial^2 z}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 z}{\partial x_1 \partial x_k} \\ \vdots & \vdots & \vdots & \vdots \\ \frac{\partial^2 z}{\partial x_k \partial x_1} & \frac{\partial^2 z}{\partial x_k \partial x_2} & \cdots & \frac{\partial^2 z}{\partial x_k \partial x_k} \end{pmatrix}$$

In the multivariate case, the current approximation to critical point, $\mathbf{x}^{(t)}$, generates the next approximation via:

$$\mathbf{x}^{(t+1)} = \mathbf{x}^{(t)} - \mathbf{H}^{-1}(\mathbf{x}^{(t)}) \frac{\partial z}{\partial \mathbf{x}} (\mathbf{x}^{(t)})^T.$$

Multi-dimensional Newton's method for critical points

1. INPUT function f, initial guess $\mathbf{x}^{(0)}$, and absolute tolerance α.
2. At iteration $t = 0, 1, 2, \ldots$, WHILE $\|\frac{\partial z}{\partial \mathbf{x}}(\mathbf{x}^{(t)})\| > \alpha$
   - Compute intercept of new tangent plane:

$$\mathbf{x}^{(t+1)} = \mathbf{x}^{(t)} - \mathbf{H}^{-1}(\mathbf{x}^{(t)})\frac{\partial z}{\partial \mathbf{x}}(\mathbf{x}^{(t)})^T.$$

3. Critical point $= \mathbf{x}^{(t)}$.

Newton's method for MLE computation

Suppose we wish to maximise a log likelihood $\ell(\boldsymbol{\theta})$ using Newton's method.

Our update step is

$$\boldsymbol{\theta}^{(t+1)} = \boldsymbol{\theta}^{(t)} - \mathbf{H}(\boldsymbol{\theta}^{(t)})^{-1} \frac{\partial \ell}{\partial \boldsymbol{\theta}}(\boldsymbol{\theta}^{(t)})$$

where

$$H_{ij}(\boldsymbol{\theta}) = \frac{\partial^2 \ell(\boldsymbol{\theta})}{\partial \theta_i \partial \theta_j}$$

That is, $-H(\boldsymbol{\theta}) = \mathcal{J}(\boldsymbol{\theta})$, the observed information.

If we replace $\mathcal{J}$ by $\mathcal{I}$, the Fisher information, then the algorithm is called *Fisher scoring*.

The Fisher information is often easier/quicker to calculate, and is guaranteed to be positive definite (unlike the observed information).

Weighted Least Squares

For a GLM, a remarkable result is that the Fisher scoring version of Newton's method produces an iteration step which is the solution of a weighted least squares problem.
This problem involves the data as random variables. Define Z_i as

$$\begin{aligned} Z_i &:= g(\mu_i) + (Y_i - \mu_i)g'(\mu_i) \\ &= \mathbf{x}_i^T \boldsymbol{\beta} + \epsilon_i \end{aligned}$$

where $\epsilon_i = (Y_i - \mu_i)g'(\mu_i)$.

Observe that ϵ_i's do not have constant variance, i.e.,

$$\mathbb{E}(\epsilon_i) = 0, \quad \mathrm{Var}(\epsilon_i) = \{g'(\mu_i)\}^2 \mathrm{Var}(Y_i).$$

The random variable Z_i is the straight line approximation to $g(Y_i)$ starting at $(\mu_i, g(\mu_i))$.

Weighted LS for GLM

Recall from slide 8 that for a linear model $\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}$ where $Var(\mathbf{y}) = \mathbf{V}$ with $\mathbf{X}$ and $\mathbf{V}$ full rank, then the BLUE estimator of $\boldsymbol{\beta}$ is

$$(\mathbf{X}^T\mathbf{V}^{-1}\mathbf{X})^{-1}\mathbf{X}^T\mathbf{V}^{-1}\mathbf{y}.$$

This suggests that $\boldsymbol{\beta}$ is estimated using (replacing $\mathbf{y}$ with $\widehat{\mathbf{Z}}$)

$$\hat{\boldsymbol{\beta}} = (\mathbf{X}^T\widehat{\mathbf{V}}^{-1}\mathbf{X})^{-1}\mathbf{X}^T\widehat{\mathbf{V}}^{-1}\widehat{\mathbf{Z}},$$

where $\mathbf{V}$ is a diagonal matrix with

$$v_{ii} = \{g'(\mu_i)\}^2 a(\phi) b''(\theta_i)$$

and $\widehat{\mathbf{V}}$ and $\widehat{\mathbf{Z}}$ replaces all $\boldsymbol{\beta}$ with $\widehat{\boldsymbol{\beta}}$.

Note that $\widehat{\boldsymbol{\beta}}$ doesn't depend on $a(\phi)$. But we have another problem.....

Weighted LS for GLM

Observe that $\mu_i = g^{-1}(\mathbf{x}_i^T \boldsymbol{\beta}) = b'(\boldsymbol{\theta}_i)$

Problem: $\mathbf{Z}$ and $\mathbf{V}$ depends on $\boldsymbol{\beta}$.

Solution: Iterate! This produces the Fisher scoring version of Newton's algorithm for finding $\boldsymbol{\beta}$.

Note that the $a(\phi)$ factor in the expression for $\hat{\boldsymbol{\beta}}$ cancels out and that the covariance matrix $\mathbf{V}$ is diagonal.

Newton's method with Fisher scoring or Iterated Weighted Least Squares (IWLS)

1. Start with $\widehat{\boldsymbol{\mu}}^{(0)} = \mathbf{y}$. Specify tolerance α.
2. At iteration $t = 1, 2, \ldots,$ **DO**
   (I) Use $\widehat{\boldsymbol{\mu}}^{(t)}$ to calculate
   $\widehat{Z}_i^{(t)} = g(\widehat{\mu}_i^{(t)}) + (y_i - \widehat{\mu}_i^{(t)})g'(\widehat{\mu}_i^{(t)})$ and
   $\widehat{v}_{ii}^{(t)} = g'(\widehat{\mu}_i^{(t)})^2\psi(\widehat{\mu}_i^{(t)})$, for each i. Note $a(\phi)$ term has been omitted.
   (II) Let $\widehat{\mathbf{W}} = \widehat{\mathbf{V}}^{-1}$. Set $\widehat{\boldsymbol{\beta}}^{(t+1)} = (\mathbf{X}^T\widehat{\mathbf{W}}^{(t)}\mathbf{X})^{-1}\mathbf{X}^T\widehat{\mathbf{W}}^{(t)}\widehat{\mathbf{Z}}^{(t)}$ and
   $\widehat{\mu}_i^{(t+1)} = g^{-1}(\mathbf{x}_i^T\widehat{\boldsymbol{\beta}}^{(t+1)})$ for each i.
3. **WHILE** $\|\widehat{\boldsymbol{\beta}}^{(t+1)} - \widehat{\boldsymbol{\beta}}^{(t)}\| > \alpha$

In most cases, the algorithm converges to the maximum likelihood estimator.

Example: insecticide efficacy

An experiment measuring death rates for insects, with 30 insects at each of five treatment levels.

```
library(faraway)
data(bliss)
bliss
```

```
##   dead alive conc
## 1    2    28    0
## 2    8    22    1
## 3   15    15    2
## 4   23     7    3
## 5   27     3    4
```

We model this with a binomial regression model, and fit using IWLS...

Example: start

```
# IWLS
y <- bliss$dead
m <- bliss$dead + bliss$alive

mu <- y
eta <- logit(mu/m)
z <- eta + (y - mu)*m/mu/(m - mu)
w <- mu*(m - mu)/m
lmod <- lm(z ~ conc, weights=w, bliss)
coef(lmod)

## (Intercept)        conc
##   -2.302462    1.153587
```

Example: iteration

```
for (i in 1:5) {
eta <- lmod$fit
mu <- m*ilogit(eta)
z <- eta + (y - mu)*m/mu/(m - mu)
w <- mu*(m - mu)/m
lmod <- lm(z ~ conc, weights=w, bliss)
cat(i, coef(lmod), "\n")}

## 1 -2.323672 1.161847
## 2 -2.32379 1.161895
## 3 -2.32379 1.161895
## 4 -2.32379 1.161895
## 5 -2.32379 1.161895
```

Variance of $\hat{\boldsymbol{\beta}}$

Suppose that the IWLS algorithm converges to the estimate $\hat{\boldsymbol{\beta}}$, then

$$\hat{\boldsymbol{\beta}} = (\mathbf{X}^T\widehat{\mathbf{V}}^{-1}\mathbf{X})^{-1}\mathbf{X}^T\widehat{\mathbf{V}}^{-1}\widehat{\mathbf{Z}}$$

where for each $i = 1, \cdots, n$

$$\begin{aligned}
\widehat{\mu}_i &= g^{-1}(\mathbf{x}_i^T\widehat{\boldsymbol{\beta}}) \\
\widehat{z}_i &= \mathbf{x}_i^T\widehat{\boldsymbol{\beta}} + (y_i - \widehat{\mu}_i)g'(\widehat{\mu}_i) \\
\widehat{v}_{ii} &= (g'(\widehat{\mu}_i))^2\psi(\widehat{\mu}_i)a(\phi)
\end{aligned}$$

Since $Var(\mathbf{Z}) = \mathbf{V}$, by results in slide 9:

$$Var(\widehat{\boldsymbol{\beta}}) \approx (\mathbf{X}^T\mathbf{V}^{-1}\mathbf{X})^{-1}$$

Hence, an estimate of $Var(\widehat{\boldsymbol{\beta}})$ is $\widehat{Var}(\widehat{\boldsymbol{\beta}}) = (\mathbf{X}^T\widehat{\mathbf{V}}^{-1}\mathbf{X})^{-1}$

Note that the $a(\phi)$ term in $\widehat{\mathbf{V}}$ does not cancel here, as it did in the IWLS algorithm, so we need to estimate it.

Now $(Y_i - \mu_i)/\sqrt{\psi(\mu_i)}$ has mean 0 and variance $a(\phi)$, so it should come as no surprise that

$$X^2 := \sum_{i=1}^{n} \frac{(Y_i - \hat{\mu}_i)^2}{\psi(\hat{\mu}_i)} \approx a(\phi)\chi^2_{n-p}$$

where p is the number of parameters used to estimate $\boldsymbol{\mu}$.

The statistic $X^2/(n-p)$ is an estimator for $a(\phi)$.

X^2 is called residual deviance (or Pearson's χ^2-statistic). It can be shown that $X^2/(n-p)$ is a consistent estimator for $a(\phi)$.

Deviance

From here on, by reinterpreting the parameter if necessary, assume $a(\phi) = \phi$.

Definition: the *scaled deviance*, $\frac{D^A}{\phi}$, for model A is

$$\frac{D^A}{\phi} = -2 \log \frac{\mathcal{L}(\hat{\boldsymbol{\beta}}^A)}{\mathcal{L}(\text{full})}$$

where

- $\hat{\boldsymbol{\beta}}^A$ is the MLE of $\boldsymbol{\beta}^A$, the true parameter value for model A, and
- $\mathcal{L}(\text{full})$ is the maximum likelihood for the saturated model with one parameter for each observation.

The *deviance* is just D^A.

Example: Gaussian

The saturated Gaussian model has MLE, y_i, in estimating one mean, μ_i, for each of the n observations.

The deviance can be written as $D = \sum_i d_i$ where

$$d_i = (y_i - \hat{\mu}_i)^2$$

where $\hat{\mu}_i$ is the fitted mean using model A, so that D is the residual sum of squares for the fitted model.

The scaled deviance is $\sum_i d_i/\sigma^2$.

Example: Poisson

The saturated Poisson model has MLE y_i for $\mu_i = \lambda_i$.

The deviance (and the scaled deviance since $\phi = 1$) can be written as $D = \sum_i d_i$ where

$$d_i = -2\left(y_i \log \frac{\hat{\mu}_i}{y_i} - (\hat{\mu}_i - y_i)\right)$$

where $\hat{\mu}_i$ is the fitted mean using the model.

The scaled deviance equals deviance.

Example: Binomial

The saturated Binomial model has MLE y_i for $\mu_i = m_i p_i$.

Equivalently y_i/m_i is the saturated model MLE of p_i.

The deviance (and the scaled deviance since $\phi = 1$) can be written as $D = \sum_i d_i$ where

$$d_i = -2\left(y_i \log \frac{\hat{\mu}_i}{y_i} + (m_i - y_i) \log \frac{m_i - \hat{\mu}_i}{m_i - y_i}\right)$$

where $\hat{\mu}_i$ is the fitted mean using the model.

The scaled deviance equals deviance.

If the model is adequate then the scaled deviance will often (but not always) be $\approx \chi^2_{n-p}$, where the saturated model has n parameters (equal to the number of observations) and the fitted model has p parameters.

For nested models, if the smaller model is correct then the difference between two scaled deviances is the log likelihood ratio and will be $\approx \chi^2_s$ for large n, where s is the difference in the number of parameters.

The scaled deviance can be used to test model adequacy because the likelihood ratio statistic is accurately approximated by χ^2. But the χ^2-approximation for the saturated model maximum likelihood is not as reliable for large n.

If the difference between two scaled deviances is large enough, then the null hypothesis of the adequacy of the smaller model can be rejected (this is called a *likelihood ratio test*).

Recall that Poisson$(\mu) \approx N(\mu, \mu)$ and Binomial$(n, p) \approx N(np, np(1-p))$.

For the Binomial and Poisson models, the scaled deviance will be approximately χ^2 when the individual responses are somewhat normal (yes, we can use normal approximation of Binomial and Poisson distributions!).

As a rule of thumb we need the Poisson mean or the Binomial mean of both failures and successes to be at least 5.

For the Normal, Gamma or Inverse Gaussian models, $X^2/(n-p)$, or the residual deviance divided by the degrees of freedom, estimates the scale parameter ϕ.

For a Gaussian linear model A nested within linear model B with , under the null hypothesis that model A is correct ,

$$\frac{(D^A - D^B)/s}{X^2/(n-p)} \sim F_{s,n-p}$$

where there are n observations, A has $p - s$ parameters and B has p parameters.

For other GLM's this distributional result only holds approximately, but it can still be used for comparing models.

In particular it can be used to compare Gamma models (alternatively the AIC can be used).

Bliss(1935) dose-response

Recall that 5 sets of 30 insects were exposed to 5 different concentrations of insecticide and after exposure the number alive was recorded.

```
str(bliss)

## 'data.frame': 5 obs. of  3 variables:
##  $ dead : num  2 8 15 23 27
##  $ alive: num  28 22 15 7 3
##  $ conc : int  0 1 2 3 4
```

GLM command

The GLM command can fit the logistic model as follows:

```
modl <- glm(cbind(dead,alive) ~ conc,family=binomial, data=bliss)
summary(modl)
```

```
##
## Call:
## glm(formula = cbind(dead, alive) ~ conc, family = binomial, data = bliss)
##
## Deviance Residuals:
##       1        2        3        4        5
## -0.4510   0.3597   0.0000   0.0643  -0.2045
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -2.3238     0.4179  -5.561 2.69e-08 ***
## conc          1.1619     0.1814   6.405 1.51e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 64.76327  on 4  degrees of freedom
## Residual deviance:  0.37875  on 3  degrees of freedom
## AIC: 20.854
##
## Number of Fisher Scoring iterations: 4
```

Model adequacy

The residual deviance p-value can be found from:

```
pchisq(0.37875,df=3,lower=FALSE)

## [1] 0.9445964
```

There is a very large chance that a chi-square rv with 3 df would be more than the residual deviance so there is no evidence of model inadequacy.

Adequacy of null model (Null deviance)

The null deviance is the deviance if linear predictor equals to β_0 (intercept-only model).

The p-value for the null model is:

```
pchisq(64.76327,df=4,lower=FALSE)
```

```
## [1] 2.886305e-13
```

There is a tiny chance that a chi-square rv with 4 df would be more than the observed null deviance so the model with one probability for all concentrations is not adequate.

Significance of concentration

The null deviance is the deviance if only one probability is fitted for all 5 concentrations.

The difference between this and the residual deviance is the contribution of the insecticide concentration in explaining the response.

This difference is $64.76327 - 0.37875 = 64.38452$ and this has 1 degree of freedom with p-value

```
pchisq(64.38452,df=1,lower=FALSE)
```

```
## [1] 1.0235921e-15
```

There is a tiny chance that a chi-square rv with 1 df would be more than the observed difference so concentration is significant in the presence of an intercept term.

Use of GLM anova

An additional model could be fitted with a quadratic term as follows. This is done only for illustrative purposes since the low residual deviance makes any refinement of the model dubious.

```
mod12 <- glm(cbind(dead, alive) ~ conc+I(conc^2),
family=binomial,bliss)
anova(modl,mod12,test="Chi")

## Analysis of Deviance Table
##
## Model 1: cbind(dead, alive) ~ conc
## Model 2: cbind(dead, alive) ~ conc + I(conc^2)
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1         3    0.37875
## 2         2    0.19549  1  0.18325   0.6686
```

The large p-value confirms that the quadratic term is not necessary.

Akaike Information Criterion (AIC)

For model selection, the AIC often produces better models than comparing (scaled) deviances, and choosing a significance level is not necessary. The AIC is defined as

$$\text{AIC} = 2p - 2\log \mathcal{L}(\hat{\boldsymbol{\beta}})$$

where p is the number of parameters in the fitted model.

The model with smallest AIC is preferred.

If model B has s more parameters than model A (not necessarily nested within B), then

$$\begin{aligned} \text{AIC}^B - \text{AIC}^A &= 2s - 2\log \mathcal{L}(\hat{\boldsymbol{\beta}}^B) + 2\log \mathcal{L}(\hat{\boldsymbol{\beta}}^A) \\ &= 2s + \frac{D^B}{\phi} - \frac{D^A}{\phi}. \end{aligned}$$

In practice, we need an estimate of ϕ to compute AIC.

Example Galapogos islands: Data description

Data was collected on the number of species on each of 30 islands in the Galapogos Islands.

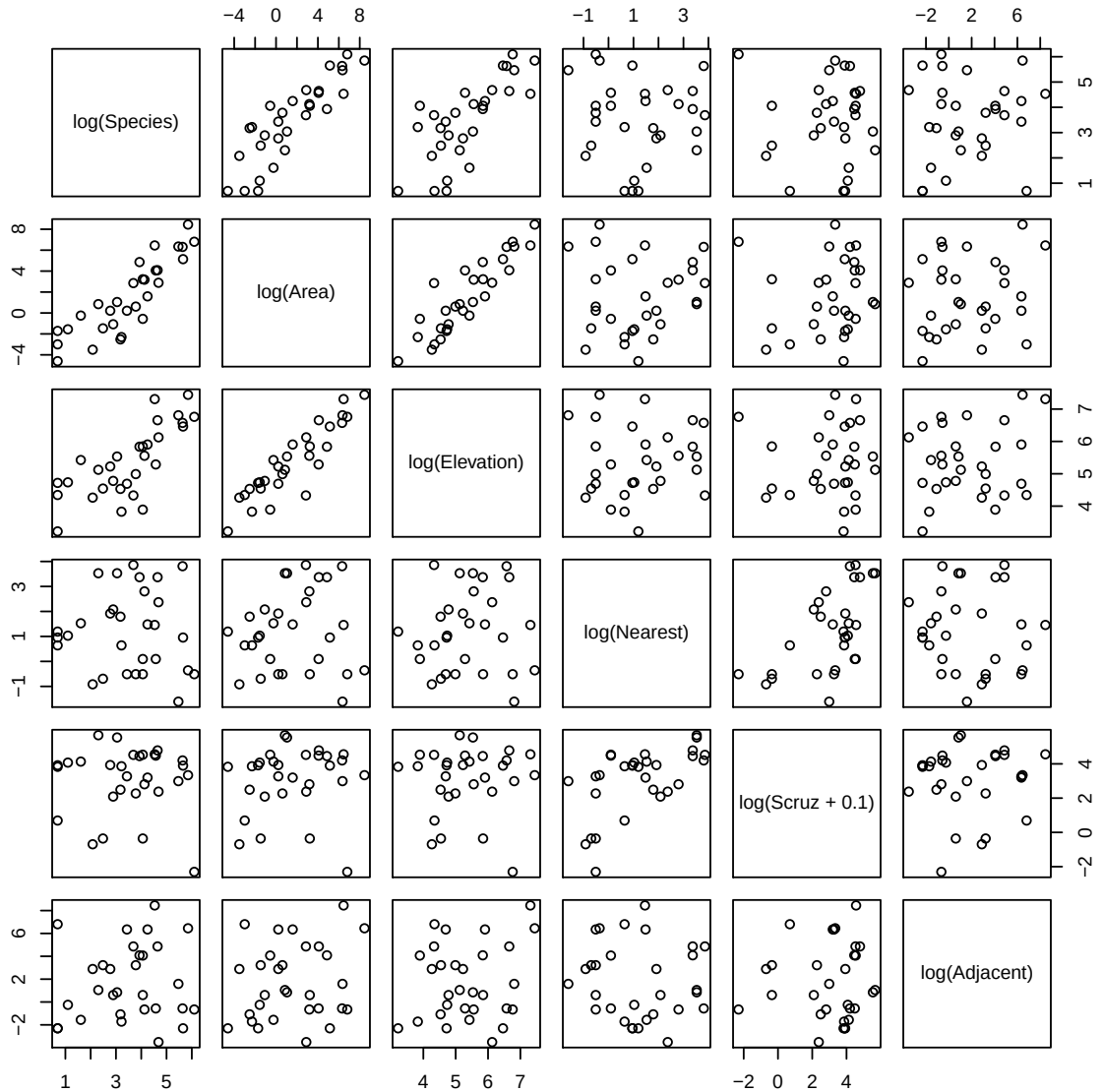
There were 5 geographic variables that can be used to explain the number of species on each island:

- Area - the area of the island (km^2)
- Elevation - the highest elevation of the island (m)
- Nearest - the distance from the nearest island (km)
- Scruz - the distance from Santa Cruz island (km)
- Adjacent - the area of the adjacent island (square km)

A lot of variability indicates taking logs might help.

Pairwise plots are shown in the next Figure.

Figure: Pairwise plots of logs of Galapogos variables



Poisson GLM

```
modp <- glm(Species ~ log(Area) + log(Elevation) + log(Nearest) + log(Scruz+0.1) + log(Adjacent),
family=poisson, gala)
summary(modp)
```

```
##
## Call:
## glm(formula = Species ~ log(Area) + log(Elevation) + log(Nearest) +
##     log(Scruz + 0.1) + log(Adjacent), family = poisson, data = gala)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -5.4479  -2.6717  -0.4547   2.5613   8.2970
##
## Coefficients:
##                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)       3.287941   0.284661  11.550  < 2e-16 ***
## log(Area)         0.348445   0.018029  19.327  < 2e-16 ***
## log(Elevation)    0.036421   0.056983   0.639  0.52272
## log(Nearest)     -0.040644   0.013781  -2.949  0.00318 **
## log(Scruz + 0.1) -0.030045   0.010492  -2.864  0.00419 **
## log(Adjacent)    -0.089014   0.006948 -12.812  < 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: 3510.73  on 29  degrees of freedom
## Residual deviance:  359.12  on 24  degrees of freedom
## AIC: 531.96
##
## Number of Fisher Scoring iterations: 5
```

Stepwise using AIC

The command to see what is the best model starting with a model that includes all predictors is:

```
step(modp, scope = ~ .)
```

```
## Start:  AIC=531.96
## Species ~ log(Area) + log(Elevation) + log(Nearest) + log(Scruz +
##     0.1) + log(Adjacent)
##
##                    Df Deviance    AIC
## - log(Elevation)    1   359.54 530.37
## <none>                  359.12 531.96
## - log(Scruz + 0.1)  1   367.27 538.10
## - log(Nearest)      1   367.79 538.62
## - log(Adjacent)     1   525.13 695.96
## - log(Area)         1   714.98 885.81
##
## Step:  AIC=530.37
## Species ~ log(Area) + log(Nearest) + log(Scruz + 0.1) + log(Adjacent)
##
##                    Df Deviance    AIC
## <none>                   359.5  530.4
## + log(Elevation)    1    359.1  532.0
## - log(Scruz + 0.1)  1    367.7  536.6
## - log(Nearest)      1    368.5  537.3
## - log(Adjacent)     1    528.6  697.4
## - log(Area)         1   3266.1 3434.9
##
## Call:  glm(formula = Species ~ log(Area) + log(Nearest) + log(Scruz +
##     0.1) + log(Adjacent), family = poisson, data = gala)
##
## Coefficients:
##      (Intercept)         log(Area)      log(Nearest)  log(Scruz + 0.1)
##          3.46648           0.35871           -0.04112           -0.03010
##    log(Adjacent)
##         -0.08822
##
## Degrees of Freedom: 29 Total (i.e. Null);  25 Residual
## Null Deviance:     3511
## Residual Deviance: 359.5  AIC: 530.4
```

Diagnostics: residuals

Response residuals: $y_i - \hat{\mu}_i$
Unless the variance function $\psi(\mu)$ is constant, as in the Gaussian case, plots with the response residuals are not very useful in assessing whether the model is reasonable.

Pearson residuals:

$$r_P(i) = \frac{y_i - \hat{\mu}_i}{\sqrt{\psi(\hat{\mu}_i)}}$$

Pearson residuals are (approximately) homoskedastic, and $\sum_i r_P(i)^2 = X^2$.

Deviance residuals:

$$r_D(i) = \text{sign}(y_i - \hat{\mu}_i)\sqrt{d_i}$$

where the deviance is $D = \sum_i d_i = \sum_i r_D(i)^2$. Caution: Do not confuse deviance residuals with residual deviance!
For GLMs deviance residuals are often the most useful, noting that the need for additional parameters in a model is assessed through reduction in deviance.

As for linear models, patterns in the residuals indicate structure in the data that has not been captured by the model.

We can plot the residuals against predictor variables, the responses, or the fitted means. Often a plot against the linear predictors, $\eta_i = \mathbf{x}_i^T \boldsymbol{\beta}$, works well.

With count data residual plots exhibit banding due to the discrete nature of the responses, and this can make it hard to see other patterns.

In this case we can use a smoothed fit of the residuals to help spot trends/patterns.

Residual plot commands

Three different plots of residuals are in Figures 3, 4 and 5.

```
par(mfrow=c(1,1))
#  Deviance residuals versus fitted values
plot(residuals(modp) ~ predict(modp,type="response"),
xlab=expression(hat(mu)), ylab="Deviance residuals",
main="Cramped in x")
# Deviances residuals versus linear predictor
plot(residuals(modp) ~ predict(modp,type="link"),
xlab=expression(hat(eta)), ylab="Deviance residuals",
main="Easier to look for patterns - none observed")
#  Residuals versus linear predictor
plot(residuals(modp,type="response") ~ predict(modp, type="link"),
xlab=expression(hat(eta)), ylab="Response residuals",
main= "Heteroskedastic")
```

Figure: Deviance residuals versus the fitted values for number of species

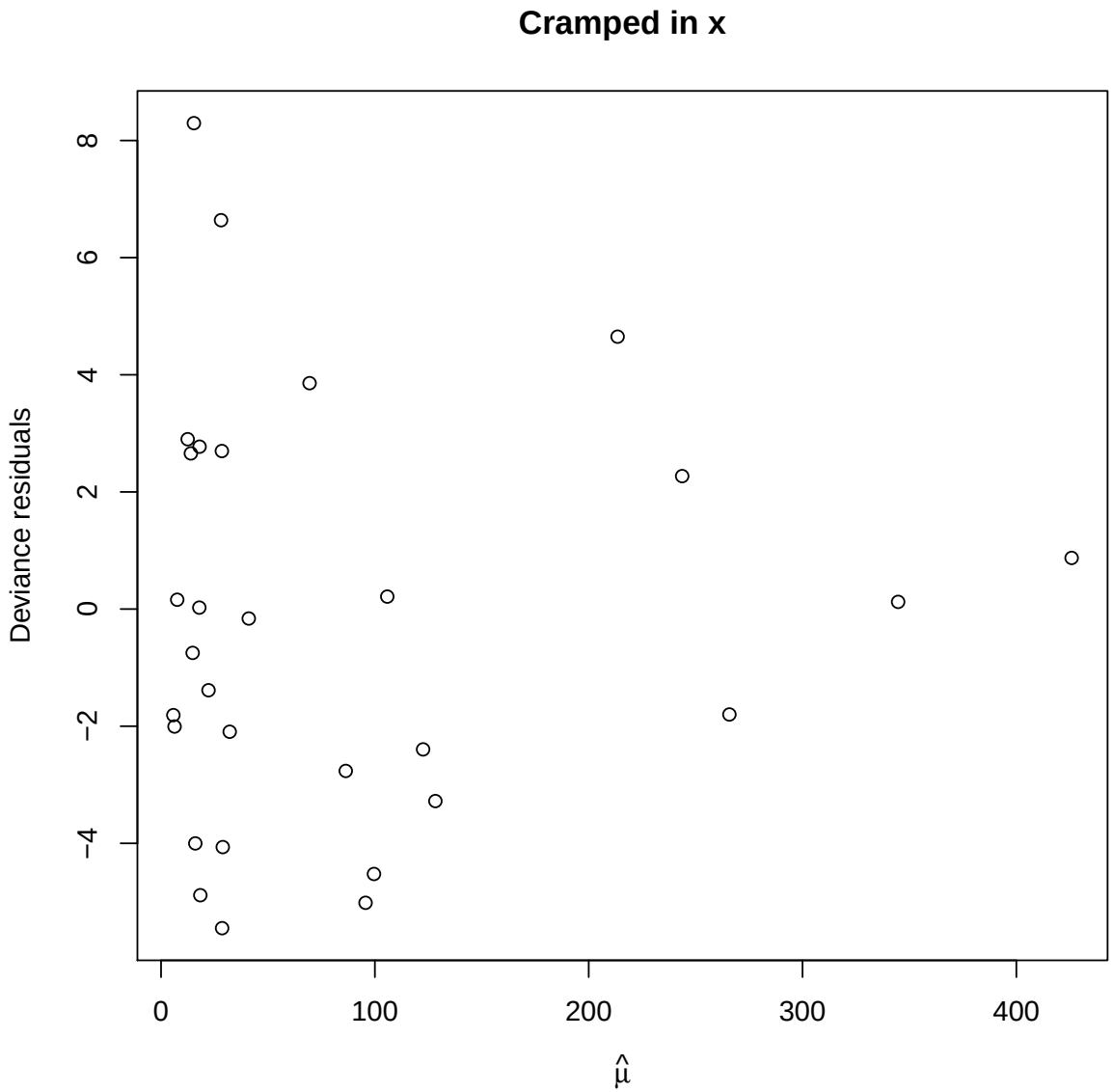


Figure: Deviance residuals versus the fitted values on linear predictor scale

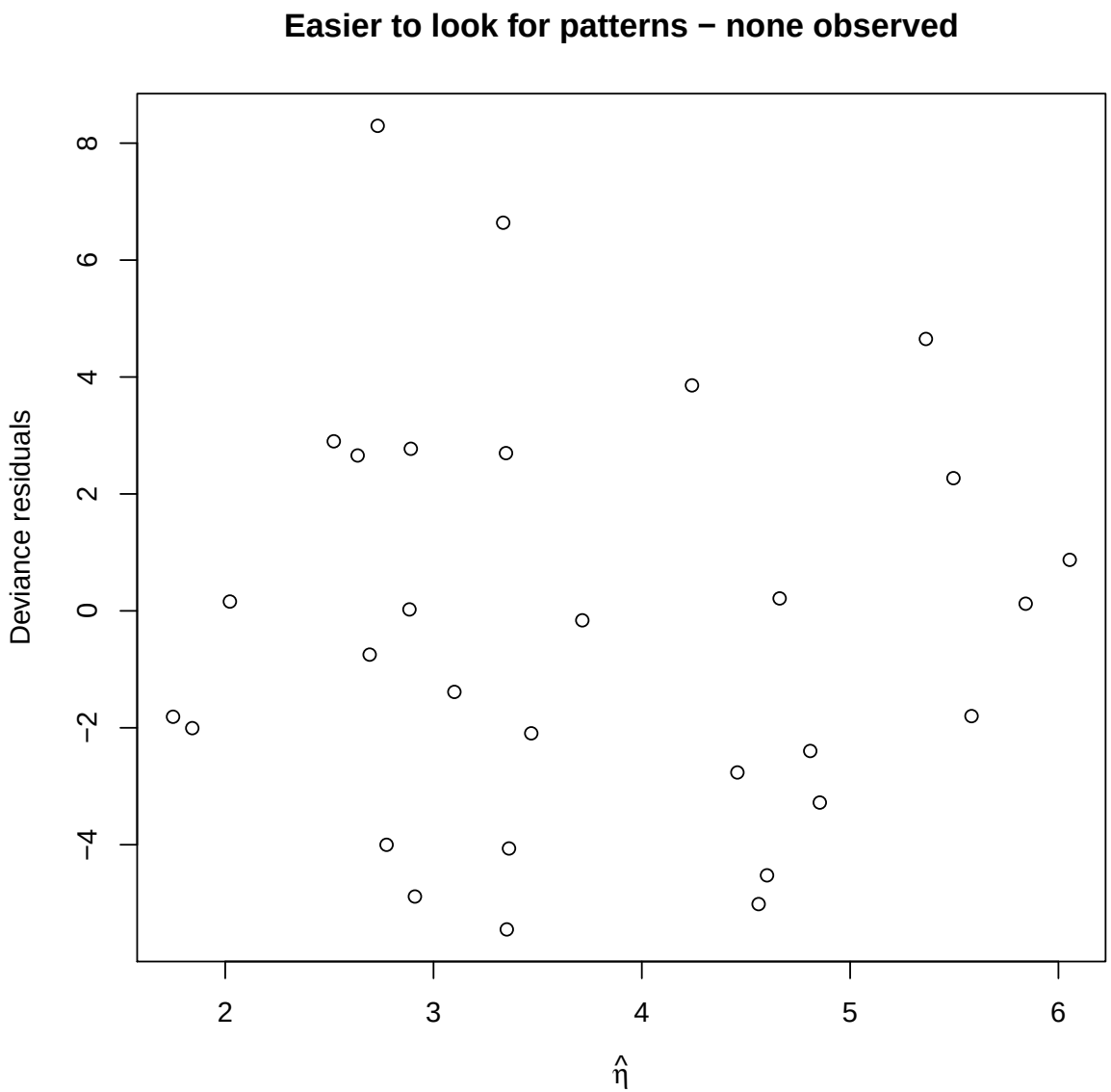
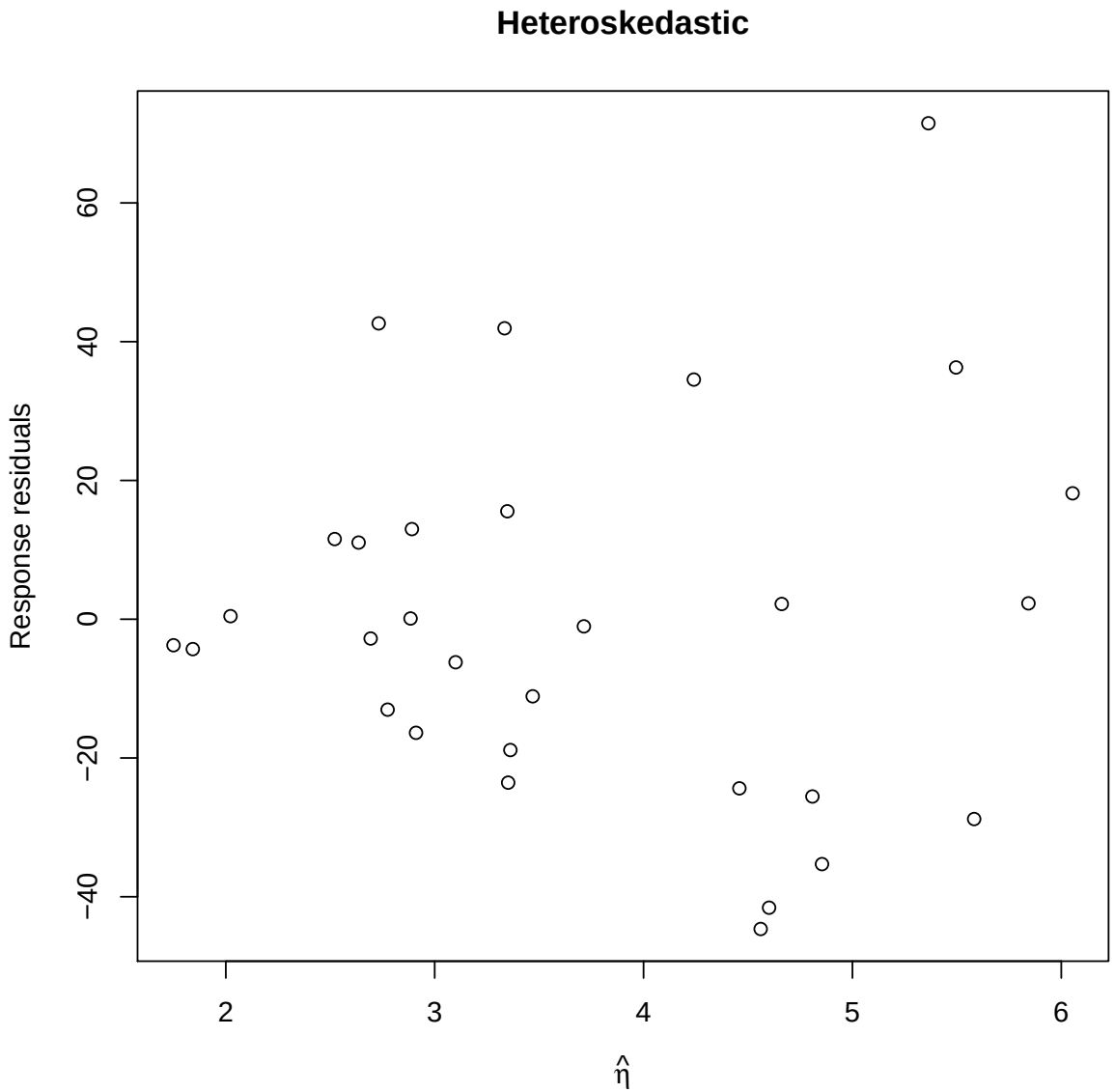


Figure: Residuals versus the fitted values on linear predictor scale



Leverage & Studentised residuals

Just as in linear models, the leverage measures the potential influence of a point on the fitted model.

The definition of leverage comes from the theory of linear models, using the hat matrix from the IWLS fitting.

Assuming we have oracle knowledge of $\boldsymbol{\beta}$, we may evaluate the pseudo-fitted value vector as

$$\mathbf{X}\widehat{\boldsymbol{\beta}} \approx \mathbf{X}(\mathbf{X}^T\mathbf{V}^{-1}\mathbf{X})^{-1}\mathbf{X}^T\mathbf{V}^{-1}\mathbf{Z},$$

where $\mathbf{Z}$ and $\mathbf{V}$ can be computed using our oracle knowledge of $\boldsymbol{\beta}$. Following linear model theory, let

$$\mathbf{H} = \mathbf{X}(\mathbf{X}^T\mathbf{V}^{-1}\mathbf{X})^{-1}\mathbf{X}^T\mathbf{V}^{-1}.$$

Leverage & Studentised residuals

Now, consider the covariance matrix "residual" of $\mathbf{Z}$ under oracle knowledge:

$$\begin{aligned}
Var(\mathbf{Z} - \mathbf{HZ}) &= Var(\{\mathbf{I} - \mathbf{H}\}\mathbf{Z}) \\
&= (\mathbf{I} - \mathbf{H})\mathbf{V}(\mathbf{I} - \mathbf{H})^T \\
&= (\mathbf{V} - \mathbf{HV})(\mathbf{I} - \mathbf{H}^T) \\
&= \mathbf{V} - \underbrace{\mathbf{VH}^T}_{=\mathbf{HV}} - \mathbf{HV} + \mathbf{H}\underbrace{\mathbf{VH}^T}_{=\mathbf{HV}} \\
&= \mathbf{V} - \mathbf{HV} - \mathbf{HV} + \underbrace{\mathbf{H}^2}_{=\mathbf{H}}\mathbf{V} \\
&= (\mathbf{I} - \mathbf{H})\mathbf{V},
\end{aligned}$$

where $\mathbf{VH}^T = \mathbf{VV}^{-1}\mathbf{X}(\mathbf{X}^T\mathbf{V}^{-1}\mathbf{X})^{-1}\mathbf{X}^T\mathbf{V}^{-1}\mathbf{V} = \mathbf{HV}$ and $\mathbf{H}^2 = \mathbf{H}$.

Leverage & Studentised residuals

Using some algebra, the i-th entry of $\mathbf{Z} - \mathbf{HZ}$ is:

$$\begin{aligned} Z_i - (\mathbf{HZ})_i &\approx g(\mu_i) + (Y_i - \mu_i)g'(\mu_i) - g(\mu_i) \\ &= (Y_i - \mu_i)g'(\mu_i) \end{aligned}$$

Hence, we have

$$Var\left\{\frac{Y_i - \mu_i}{\sqrt{(1 - (\mathbf{H})_{ii})\psi(\mu_i)\phi}}\right\} \approx 1$$

Leverage & Studentised residuals

But, in practice, we do not have oracle knowledge of $\boldsymbol{\beta}$. Hence, we use

$$\widehat{\boldsymbol{\beta}} = (\mathbf{X}^T\widehat{\mathbf{V}}^{-1}\mathbf{X})^{-1}\mathbf{X}^T\widehat{\mathbf{V}}^{-1}\widehat{\mathbf{Z}},$$

The generalised linear model hat matrix is

$$\widehat{\mathbf{H}} = \mathbf{X}(\mathbf{X}^T\widehat{\mathbf{V}}^{-1}\mathbf{X})^{-1}\mathbf{X}^T\widehat{\mathbf{V}}^{-1}.$$

The generalised linear model leverage h_{ii} for the i-th observation is the i-th diagonal element of $\widehat{\mathbf{H}}$.

Note that both $\mathbf{H}$ and $\widehat{\mathbf{H}}$ don't depend on ϕ.

The studentised Pearson residual is:

$$r_{SP}(i) := \frac{r_P(i)}{\sqrt{(1-h_{ii})\hat{\phi}}} = \frac{Y_i - \widehat{\mu}_i}{\sqrt{(1-h_{ii})\psi(\widehat{\mu}_i)\widehat{\phi}}}$$

Jack-knife residuals

Analogously, we define the i-th studentised deviance residual as

$$r_{SD}(i) := \frac{r_D(i)}{\sqrt{(1-h_{ii})\hat{\phi}}}$$

A large leverage does not necessarily mean a point *has* influenced the fit.

A direct measure of the influence of a point is the **jack-knife residual**, which is the difference between y_i and $\hat{\mu}_i^{(i)}$ predicted when you remove y_i from the set of observations, then scaled to standardise the variance.

The jack-knife residual can be approximated by

$$\approx \text{sign}(y_i - \hat{\mu}_i)\sqrt{(1-h_{ii})r_{SD}^2(i) + h_{ii}r_{SP}^2(i)}.$$

Cook's distance

Another measure of the influence of the i-th observation is **Cook's distance**:

$$\frac{(\widehat{\boldsymbol{\beta}}^{(i)} - \widehat{\boldsymbol{\beta}})^T \mathbf{X}^T \widehat{\mathbf{V}}^{-1} \mathbf{X} (\widehat{\boldsymbol{\beta}}^{(i)} - \widehat{\boldsymbol{\beta}})}{p}$$

where $\widehat{\boldsymbol{\beta}}^{(i)}$ is the estimate of $\boldsymbol{\beta}$ obtained when y_i is omitted.

(Formula is slightly different from Faraway's textbook but works out to be equivalent)

Recall that for linear models the Cook's distance can be expressed in terms of the leverage and the studentised (Pearson) residual, and is large when both of these are large. Values greater than 1 are usually of interest and/or values substantially larger than the majority.

The jack-knife residual and Cook's distance are both useful for detecting influential outliers.

Plotting residuals

When looking at residuals it is helpful to consider them ordered by absolute size.

If we plot the ordered absolute values against the percentage points of a half-normal dsitribution then it is easier to see if the largest values are in keeping with the others.

That is, plot the i-th ordered absolute residual against $\Phi^{-1}((n+i)/(2n+1))$, for $i = 1, \ldots, n$. If all is well we expect to see a smooth plot, while a jump or kink in the tail indicates a potential problem.

Note that for a glm the residuals will not in general be normal, so don't expect a straight line.

Checking linearity

A non-linear link g (anything except the identity) makes it harder to check the assumption that $g(\mu_i) = \mathbf{x}_i^T \boldsymbol{\beta}$.

The easiest thing to do is to plot $g(y_i)$ against $\{x_{ij}\}_{i=1}^n$ for each j and look for linear relationships.

A more sophisticated approach is to plot $z_i = g(\mu_i) + (y_i - \mu_i)g'(\mu_i)$ against $\{x_{ij}\}_{i=1}^n$, where $\hat{\mu}_i$ substitutes for μ_i.

From the IWLS scheme —provided the predictor variables are linearly independent—these plots should be linear.

If there are non-linearities present then we can consider transforming $\{x_{ij}\}_{i=1}^n$ or adding extra variables.

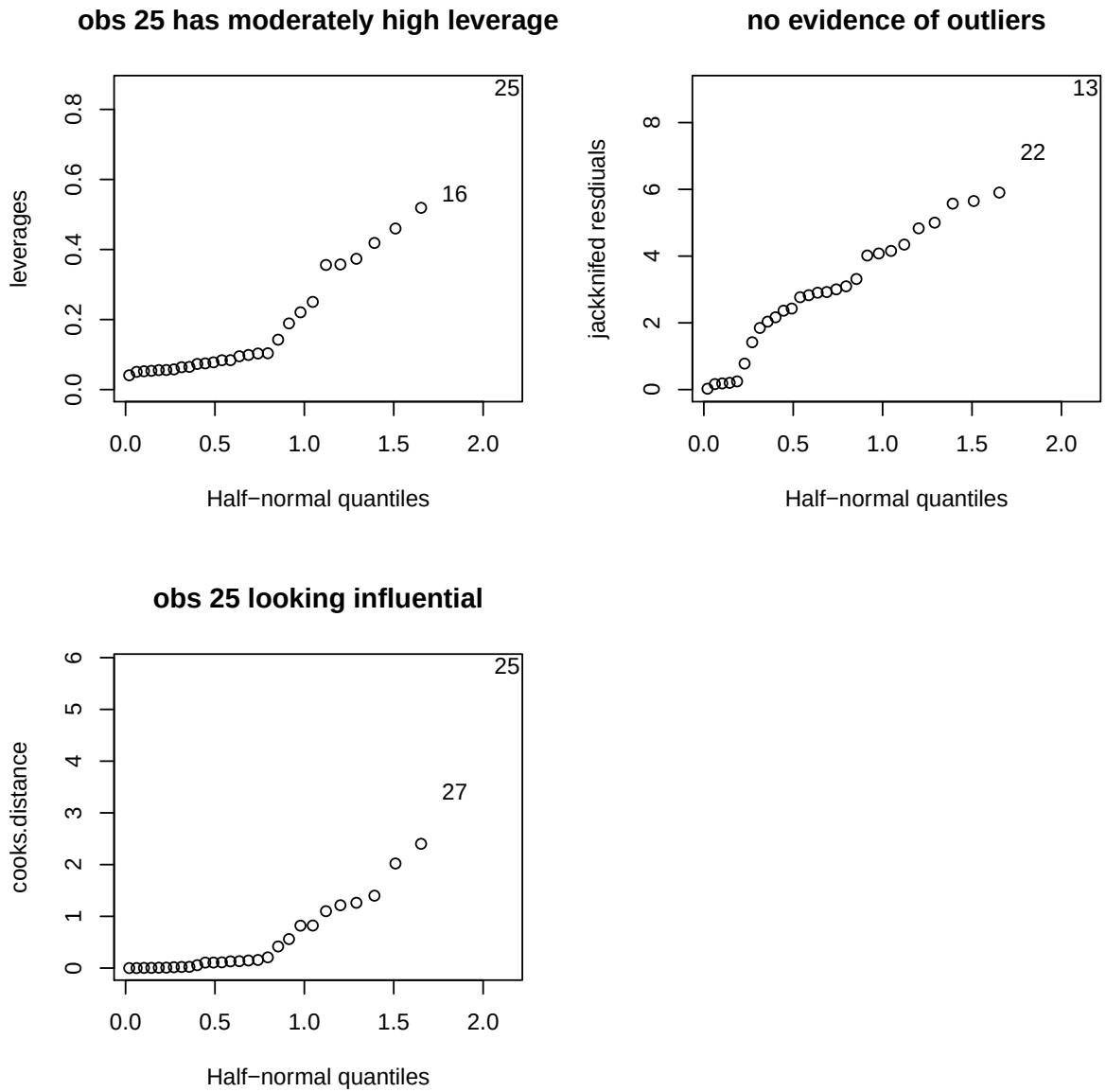
Transforming the responses y_i is often not a good idea for a glm, as this can break assumptions made about the distribution of Y_i.

Leverage, Jack-knife residuals, Cook's distance

Plots illustrating these are in the following page

```
par(mfrow=c(2,2))
# leverages
halfnorm(influence(modp)$hat,
ylab="leverages",
main= "obs 25 has moderately high leverage")
# jackknife residuals
halfnorm(rstudent(modp),
ylab="jackknifed resdiuals",
main= "no evidence of outliers")
# Cook's distance - obs 25 looking influential
halfnorm(cooks.distance(modp),
ylab="cooks.distance",
main = "obs 25 looking influential")
```

Figure: Leverage, Jack-knife residuals, Cook's distance



What to do about 25?

Observation 25 has $\texttt{Scruz} = 0$, which is $-\infty$ when taking log. We resolve this with $\log(\text{Scruz} + 0.1)$

```
# effect of removing obs 25 on model
modp2 <- glm(Species ~ log(Area) + log(Elevation) +
log(Nearest) + log(Scruz+0.1) + log(Adjacent),
family=poisson, gala, subset=-25)
summary(modp2)
step(modp2)
```

Refitting without observation 25

```
##
## Call:
## glm(formula = Species ~ log(Area) + log(Elevation) + log(Nearest) +
##     log(Scruz + 0.1) + log(Adjacent), family = poisson, data = gala,
##     subset = -25)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -5.7237  -2.7539  -0.3181   2.6401   7.9333
##
## Coefficients:
##                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)       3.05070    0.30033  10.158  < 2e-16 ***
## log(Area)         0.33453    0.01883  17.770  < 2e-16 ***
## log(Elevation)    0.05960    0.05743   1.038 0.299325
## log(Nearest)     -0.05255    0.01469  -3.578 0.000347 ***
## log(Scruz + 0.1)  0.01592    0.02218   0.718 0.472998
## log(Adjacent)    -0.08852    0.00696 -12.717  < 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: 2707.88  on 28  degrees of freedom
## Residual deviance:  353.42  on 23  degrees of freedom
## AIC: 518.32
```

Continued in the following page....

Refitting without observation 25

```
## Number of Fisher Scoring iterations: 5
## Start:  AIC=518.32
## Species ~ log(Area) + log(Elevation) + log(Nearest) + log(Scruz +
##     0.1) + log(Adjacent)
##
##                    Df Deviance    AIC
## - log(Scruz + 0.1)  1   353.94 516.84
## - log(Elevation)    1   354.51 517.41
## <none>                  353.42 518.32
## - log(Nearest)      1   366.21 529.11
## - log(Adjacent)     1   516.83 679.73
## - log(Area)         1   663.37 826.27
##
## Step:  AIC=516.84
## Species ~ log(Area) + log(Elevation) + log(Nearest) + log(Adjacent)
##
##                  Df Deviance    AIC
## - log(Elevation)  1   354.83 515.72
## <none>                353.94 516.84
## - log(Nearest)    1   368.20 529.09
## - log(Adjacent)   1   519.96 680.86
## - log(Area)       1   679.00 839.90
```

Continued in the following page....

Refitting without observation 25

```
## Step:  AIC=515.72
## Species ~ log(Area) + log(Nearest) + log(Adjacent)
##
##                 Df Deviance     AIC
## <none>               354.83  515.72
## - log(Nearest)   1   369.86  528.76
## - log(Adjacent)  1   521.71  680.60
## - log(Area)      1  2679.93 2838.82
##
## Call:  glm(formula = Species ~ log(Area) + log(Nearest) + log(Adjacent),
##     family = poisson, data = gala, subset = -25)
##
## Coefficients:
##   (Intercept)      log(Area)   log(Nearest)  log(Adjacent)
##       3.38465        0.35292       -0.04788       -0.08662
##
## Degrees of Freedom: 28 Total (i.e. Null);  25 Residual
## Null Deviance:     2708
## Residual Deviance: 354.8  AIC: 515.7
```

Final model

```
# without obs 25 log(Scruz+0.1) and log(Elevation) not significant
# refit using all data but omitting these variables
modp3 <- glm(Species ~ log(Area) + log(Nearest) + log(Adjacent),
family=poisson, gala)
summary(modp3)
```

```
##
## Call:
## glm(formula = Species ~ log(Area) + log(Nearest) + log(Adjacent),
##     family = poisson, data = gala)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -5.5064  -2.9908  -0.3175   2.6705   7.9670
##
## Coefficients:
##                Estimate Std. Error z value Pr(>|z|)
## (Intercept)    3.397868   0.048637  69.862  < 2e-16 ***
## log(Area)      0.362669   0.008200  44.229  < 2e-16 ***
## log(Nearest)  -0.061141   0.011695  -5.228 1.71e-07 ***
## log(Adjacent) -0.096593   0.006168 -15.660  < 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: 3510.73  on 29  degrees of freedom
## Residual deviance:  367.73  on 26  degrees of freedom
## AIC: 536.56
##
## Number of Fisher Scoring iterations: 5
```