1 Parametric generalized linear models for count data

R libraries and functions used in this chapter include:

```
library(MASS) #for fitting negative-binomial models
library(mpcmp) #taken off CRAN, but can download last version as local .ZIP file
glm #fitting generalized linear models (GLMs) in base R
glm.nb{MASS} #fitting negative-binomial GLMs in R
glm.cmp{mpcmp} #fitting mean-parametrized Conway-Maxwell-Poisson GLMs in R
histcompPIT{mpcmp} #probability inverse transform (PIT) histogram for fitted glm.cmp models
```

1.1 Introduction to generalized linear models

When one thinks about regression, one typically thinks of a model of the form

$$Y_i|X_i = X_i^T\beta + \epsilon_i\,, \quad i = 1, 2, \ldots$$

Usually implicit in the above model statement are four assumptions:

0.

1.

2.

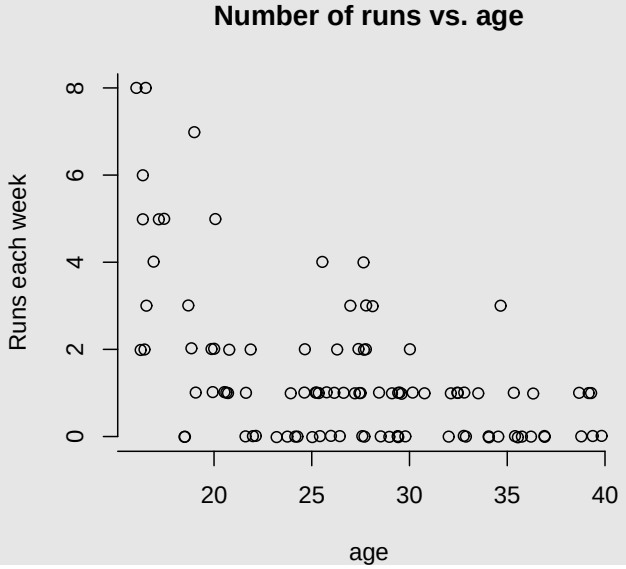
3.

Thus, a more transparent way to write the above model would be

$$Y_i|X_i \overset{ind}{\sim} N(X_i^T\beta, \sigma^2)\,.$$

Discuss the applicability of the above assumptions for the following two examples.

Example 1. In a fitness study, a hundred gym-goers were asked how many times they ran during past the week. The results, plotted against age of the respondent, are given below.



A1. Counts cannot be . This means that $E(Y|X)$ should generally not be modeled by a . One way to do this is to model the conditonal mean $\mu \equiv E(Y|X)$ via

$$\mu = \quad \Leftrightarrow \quad X^T\beta = \quad .$$

This ensures that μ is always nonnegative.

Note: we apply the log transformation to the conditional mean $\mu \equiv E(Y|X)$, and not to the data themselves (we cannot take the log of a zero count!). This transformation linking the conditional mean $E(Y|X)$ to the **linear predictor** $X^T\beta$ is called the **link function**.

A3. The responses (integer counts, with possible repeats) cannot possibly come from a normal distribution. Rather, it makes more sense to use a distribution. A reasonable model here would be

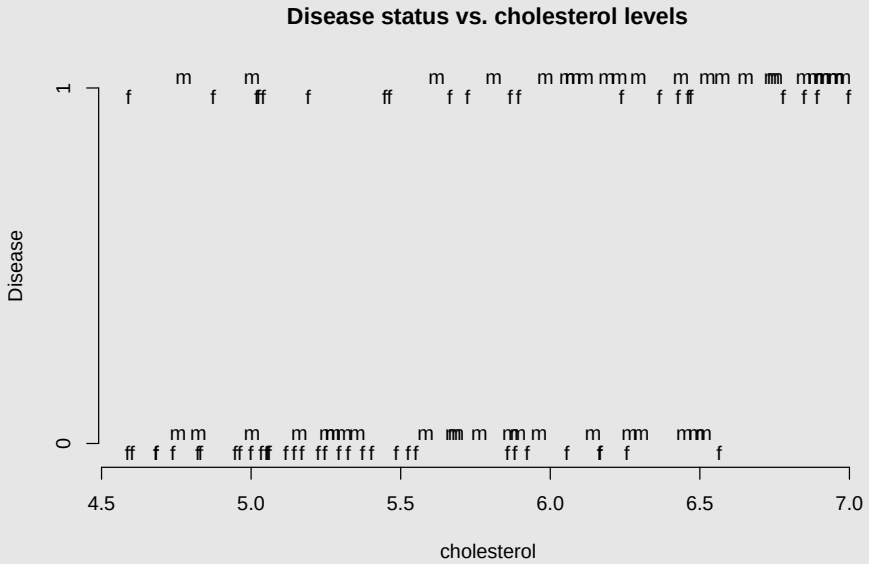
$$Y_i|X_i \overset{ind}{\sim} \text{Poisson}\left(\mu_i = \exp(X_i^T\beta)\right)$$

A2. Recall that for a Poisson random variable, the variance is

$$\text{Var}(Y_i|X_i) = \quad = \exp(X_i^T\beta)\,,$$

which is not constant. The variability of the responses increases with its mean.

Example 2. In a health study, a hundred patients were tested for their cholesterol levels and the presence of a certain disease. The results are plotted below.



A1. The expected value of Y given X (which is also the conditional probability p of success given X) cannot be smaller than or larger than . This means that $E(Y|X)$ should generally not be modeled by a . One way to do this is to model the conditional probability of success given X via

$$p = \quad \Leftrightarrow \quad X^T\beta =$$

This ensures that p is always between 0 and 1.

Note again that we apply the transformation to the conditional mean $p \equiv E(Y|X)$, and not to the data themselves (we cannot apply the logistic function to 0s and 1s!). The link function here is the logistic-link.

A3. The responses (zeros and ones, successes and failures) cannot possibly come from a normal distribution. Rather, it makes more sense to use a distribution. A reasonable model here would be

$$Y_i|X_i \overset{ind}{\sim} \text{Bernoulli}\left(p_i = \frac{\exp(X_i^T\beta)}{1+\exp(X_i^T\beta)}\right)$$

A2. Recall that for a Bernoulli random variable, the variance is

$$\text{Var}(Y_i|X_i) = \qquad\qquad = \frac{\exp(X_i^T\beta)}{(1+\exp(X_i^T\beta))^2},$$

which is not constant. The variability of responses with probabilities close to 1 or 0 is smaller than the variability of responses with probabilities around 0.5

We have briefly considered models of the form

$$\begin{aligned}
Y_i|X_i &\overset{\text{ind}}{\sim} N(X_i^T\beta, \sigma^2) \\
Y_i|X_i &\overset{\text{ind}}{\sim} \text{Poisson}(\mu_i = \exp(X_i^T\beta)) \\
Y_i|X_i &\overset{\text{ind}}{\sim} \text{Bernoulli}\left(p_i = \frac{\exp(X_i^T\beta)}{1+\exp(X_i^T\beta)}\right).
\end{aligned}$$

While we can conceivably think of (infinitely) more models using ad-hoc distributions, we will generally have to examine the properties of and construct new algorithms for *each new model we come up with*. Fortunately, there is something special about the normal, Poisson, and Bernoulli distributions, along with other commonly used distributions, that allow them to fall into a unified modelling framework called *generalized linear models* (GLMs). GLMs extend classical linear models to data from a wider class of distributions. Usually, these distributions form an *exponential family of distributions*.

1.2 Exponential Families of Distributions

A random variable Y whose distribution depends on a single parameter θ belongs to an exponential family if it has a probability (density) function of the form

$$f(y;\theta) = \exp\left\{\frac{a(y)b(\theta)+c(\theta)}{e(\phi)}\right\}d(y;\phi) \tag{1}$$

where $a(\cdot)$, $b(\cdot)$, $c(\cdot)$, $d(\cdot)$ and $e(\cdot)$ are specified functions, and ϕ is a *dispersion parameter* that is related to the variance of the distribution.

- If $b(\theta) = \theta$, distribution is in *canonical* (or natural) form.

- If $a(y) = y$, then $b(\theta)$ is called the *natural* (or canonical) parameter of the distribution and the family is called a *linear exponential family*
- The function $c(\theta)$ is a normalizing function (to ensure the probabilities sum to 1).
- $d(y;\phi)$ is a base measure, which determines the shape of the family of distributions.
- ϕ is regarded as a *nuisance parameter* and sometimes treated as known or given.

1.2.1 The Binomial distribution as an exponential family

Consider a series of n independent binary trials, each with only two possible outcomes: 'success' or 'failure' with success probability, p. Let Y be the number of 'successes' in these n trials; then Y has the $Binomial(n,p)$ distribution with probability function

$$f(y;p) = \left(\begin{array}{c} n \\ y \end{array} \right) p^y (1-p)^{n-y}.$$

This can be rewritten as

$$f(y;p) = \exp\Big\{ y \log p - y \log(1-p) + n \log(1-p) \Big\} \times \left(\begin{array}{c} n \\ y \end{array} \right)$$

This belongs to the linear exponential family with $a(y) = y$, natural parameter $b(p) = \log\left(\frac{p}{1-p}\right)$, normalizing function $c(p) = -n\log(1-p)$, and with dispersion parameter $e(\phi) = \phi =\equiv 1$. Note that the base measure $d(y) =^n C_y$ is essentially the binomial distribution with $p = 0.5$.

The binomial distribution is usually used to model counts from a process with binary outcomes. For example:

- The number of candidates from a class who pass a test
- The number of patients in a medical study who are alive at a specified time since diagnosis

1.2.2 The Poisson distribution as an exponential family

The probability mass function for random variable $Y \sim \text{Poisson}(\lambda)$ is

$$f(y;\lambda) = \frac{\lambda^y e^{-\lambda}}{y!}, \quad y = 0, 1, 2, \cdots$$

This probability function can be rewritten as

$$\begin{aligned} f(y;\lambda) &= \\ &= \\ &= \exp\Big\{ y \log \lambda - \lambda \Big\} \times 1/y!. \end{aligned}$$

Because $a(y) = y$, $f(y;\lambda)$ is in the canonical form with natural parameter is $\log\lambda$ normalizing function is $c(\lambda) = -\lambda$, and dispersion $e(\phi) = \phi \equiv 1$. Note that the base measure is $1/y!$ which is essentially the Poisson pmf with $\lambda = 1$.

The Poisson distribution is often used to model count data, which are typically the number of occurrences of some event in a defined time period or space. For example, it can be used to model

- The number of medical conditions reported for a person.
- The number of tropical cyclones during a season.
- The number of spelling mistakes on a page of a newspaper.

1.2.3 The negative-binomial distribution as an exponential family

The negative-binomial distribution is an extension of the Poisson distribution that allows the variance to be larger than the mean. It has pmf given by

$$f(y;\theta) = \begin{pmatrix} y+r-1 \\ r-1 \end{pmatrix} \theta^r (1-\theta)^y, \qquad y = 0, 1, 2, \dots,$$

where the size parameter r is typically fixed and θ is the parameter of interest.

Homework: for fixed r, show that the negative-binomial distribution is an exponential family, and find its natural statistic $a(y)$, canonical parameter $b(\theta)$, dispersion function $e(r)$, normalizing function $c(\theta)$ and base measure $d(y;r)$.

1.3 Generalized Linear Models

Generalized linear models consist of three components:

1. Response variables $Y_1, \cdots, Y_n$ have distributions from the same exponential family:
$$f_{Y_i}(y;\theta_i) = \exp\left\{ \frac{a(y)b(\theta_i) + c(\theta_i)}{e(\phi)} \right\} d(y;\phi)$$
2. A set of covariate vectors $X_1, \cdots, X_n$ and associated set of parameters $\beta_0, \cdots, \beta_d$ forming linear predictors $X_i^T \boldsymbol{\beta}$ for each observation i.
3. A link function $g(\cdot)$ such that
$$g(\mu_i) = X_i^\top \boldsymbol{\beta},$$
where $\mu_i = E(Y_i)$.

A link function $g(\cdot)$ is called a *canonical* link if $g(\mu_i) = b(\theta_i)$. Canonical links include:

- Normal: $g(\mu) = \mu$;
- Poisson: $g(\mu) = \log(\mu) = \log(\lambda)$ (note that $\mu = \lambda$ for Poisson);
- Binomial: $g(\mu) = \log\left(\frac{\mu}{1-\mu}\right) = \log\left(\frac{p}{1-p}\right)$ (note that $\mu = p$ for Bernoulli);

Canonical links are natural choices of link functions (but not always the most appropriate). They are also the default links in R. Other choices are possible: for example, the log-link is not the canonical link for the negative-binomial distribution, but it is by far the most commonly used link for its interpretability.

1.4 Models for Binomial counts

We first consider generalized linear models for response variables that are measured on a binary scale. That is, the response variable has only two possible outcomes, and can be represented by a binary indicator variable taking on values 1 and 0. 'Success' and 'failure' are used as generic terms for the two outcomes.

For example,

- In an analysis of whether or not business firms have an industrial relations department according to the size of firm.
- In a study of labor force participation of married women, as a function of age, number of children and husband's income
- In a health study, the presence of a certain disease as a function of gender and cholesterol level.

If Y_i are binary with $P(Y_i = 1|X_i) = p_i$ and $P(Y_i = 0|X_i) = 1 - p_i$, then $\mu_i = E(Y_i|X_i) = p_i$. We want to model the probability of success p_i in terms of explanatory variables X_i via some link function:

$$g(p_i) = X_i^\top \boldsymbol{\beta}$$

The following three link functions are commonly used for binary response variables:

- Logistic (or logit) models

$$\log\left(\frac{p_i}{1-p_i}\right) = X_i^\top \boldsymbol{\beta} \quad \text{or} \quad p_i = \frac{\exp(X_i^\top \boldsymbol{\beta})}{1+\exp(X_i^\top \boldsymbol{\beta})}$$

- Probit models

$$\Phi^{-1}(p_i) = X_i^\top \boldsymbol{\beta} \quad \text{or} \quad p_i = \Phi(X_i^\top \boldsymbol{\beta})$$

  where Φ is the standard normal cumulative distribution function

- Complementary log-log models

$$\log(-\log(1-p_i)) = X_i^\top \boldsymbol{\beta} \quad \text{or} \quad p_i = 1 - \exp[-\exp(X_i^\top \boldsymbol{\beta})]$$

The most popular of these is the logistic link, because it is the easiest to interpret.

Note that the *binary* (0–1) nature of the responses is handled by the *Bernoulli distribution*, while the covariates determine the mean of the distribution:

$$Y_i|X_i \overset{ind}{\sim} \text{Bernoulli}\Big(p_i = \mu(X_i^T \beta)\Big)$$

where $\mu(\cdot) = g^{-1}(\cdot)$ is the inverse-link, or equivalently, the mean function. I am much better at thinking on the mean scale, so I prefer working with $\mu(\cdot)$ rather than the link.

1.4.1 Parameter interpretation for logistic models

The logistic link is perhaps the easiest to interpret, and hence it is the most widely used. In particular, the interpretation of β is somewhat similar to that in a normal linear regression model.

Consider a logistic model with two covariates x_1 and x_2:

$$\log\left(\frac{p}{1-p}\right) = \beta_0 + \beta_1 x_1 + \beta_2 x_2$$

Then, a unit change in x_1 (keeping x_2 fixed) is associated with an increase in the log-odds by . Equivalently, a unit change in x_1 is associated with a change in the odds by a *factor* of . Recall that the odds is defined as $p/(1-p)$, the relative probability of success over failure.

Similarly, a unit change in x_2 (keeping x_1 fixed) is associated with an increase in the log-odds by an amount β_2, or, equivalently, a change in the odds by a *factor* of $\exp(\beta_2)$.

The interpretation of the intercept β_0 is the log-odds when both x_1 and x_2 are zero. Equivalently, $p = \exp(\beta_0)/(1+\exp(\beta_0))$ is the probability of success at baseline, when all covariates are zero. This may or may not have a meaningful interpretation

Example (2. continued)**.** Recall that in our health study, 100 patients were tested for their cholesterol levels and the presence of heart disease. The overall prevalence of heart disease in the study was 53% for male patients and 38% for female patients.

A preliminary logistic model fitted to this dataset is

$$\log\left(\frac{\hat{p}}{1-\hat{p}}\right) = -9.3207 - 0.1095 * I(\text{sex} = \text{male}) + 1.5843 * \text{cholesterol}$$

Interpret the intercept and slope(s) of the fitted model:
Intercept: the estimated intercept is , which means that the probability of disease at baseline ( ) is estimated as

$$\frac{\exp(\quad)}{1+\exp(\quad)} = 8.95 \times 10^{-5}$$

This is not actually meaningful, because a female with cholesterol level 0 would be probably dead.

Slope for cholesterol: the estimated slope for cholesterol is , which means that a unit increase in cholesterol levels (for the same gender) is associated with an increase in log-odds of disease by 1.5843, or, equivalently, an increase in the odds by a factor of exp( ) = 4.88 times.

Slope for sex: the estimated slope for sex is , which means that males are estimated to have lower log-odds of disease by 0.1095 than females (having the same cholesterol level). Equivalently, their odds of disease is exp( ) = 0.90 times smaller. This may seem counter-intuitive, but remember, this comparison between sex is for fixed cholesterol levels. Thus, it may well be sensible that a female with the same (high) level of cholesterol as a male is more at risk to disease, because it is more uncommon for females to have (high) cholesterol levels in the first place.

1.4.2 Estimation of parameters

For $Y_i \sim \text{Bernoulli}(p_i)$, the probability of observing value $Y_i = y_i$ is

$$p_i^{y_i}(1-p_i)^{1-y_i}$$

If we have a dataset of n independent binary responses, each having probability p_i of success, the probability of observing $Y_1 = y_1, Y_2 = y_2, \ldots, Y_n = y_n$ is therefore

$$\prod_{i=1}^{n} p_i^{y_i}(1-p_i)^{1-y_i}$$

The log-likelihood function for β is therefore given by

$$\begin{aligned} l(\beta) &= \sum_{i=1}^{n} \log\left(p_i^{y_i}(1-p_i)^{1-y_i}\right) \\ &= \sum_{i=1}^{n} \log\left(p_i^{y_i}\right) + \log\left((1-p_i)^{1-y_i}\right) \\ &= \sum_{i=1}^{n} y_i \log(p_i) + (1-y_i)\log(1-p_i) \ , \end{aligned}$$

keeping in mind that $p_i = \mu(X_i^T\beta)$ for some mean function μ.

We can now find the maximum likelihood estimate of β by maximizing $l(\beta)$ in β. To do this, we can set the derivative $\partial l/\partial\beta$ to 0:

$$\begin{aligned} 0 &= \frac{\partial l(\beta)}{\partial \beta} \\ &= \sum_{i=1}^{n} y_i \frac{\partial}{\partial \beta}\log(p_i) + (1-y_i)\frac{\partial}{\partial \beta}\log(1-p_i) \\ &= \sum_{i=1}^{n} y_i \frac{\partial}{\partial p_i}\log(p_i)\frac{\partial p_i}{\partial \beta} + (1-y_i)\frac{\partial}{\partial p_i}\log(1-p_i)\frac{\partial p_i}{\partial \beta} \\ &= \sum_{i=1}^{n} y_i \frac{1}{p_i}\mu'(X_i^T\beta)X_i - (1-y_i)\frac{1}{1-p_i}\mu'(X_i^T\beta)X_i \\ &= \sum_{i=1}^{n} \left[y_i\frac{1}{p_i} - (1-y_i)\frac{1}{1-p_i}\right]\mu'(X_i^T\beta)X_i \\ &= \sum_{i=1}^{n} \left[\frac{y_i - p_i}{p_i(1-p_i)}\right]\mu'(X_i^T\beta)X_i \ , \end{aligned}$$

keeping in mind that $p_i = \mu(X_i^T\beta)$. Thus, solving for the MLE $\hat{\beta}$ is a nonlinear root-finding problem. Fortunately, there is a universal algorithm that can be used to find the MLE in any GLM, using a Newton-Raphson procedure with iteratively reweighted least squares.

To see this, note that the score equation has the form

$$0 = \sum_{i=1}^{n} \frac{y_i - \mu_i}{\text{Var}(Y_i)} \mu'(X_i^T \beta) X_i \tag{2}$$

where $\mu_i = p_i = \mu(X_i^T \beta)$ is the mean of Y_i and $\text{Var}(Y_i) = p_i(1 - p_i)$ is the variance of Y_i. Don't forget that, in general, both μ_i and $\text{Var}(Y_i)$ may be functions of β. We can compare this to the score equations for normal linear regression:

$$0 = \sum_{i=1}^{n} \left(y_i - X_i^T \beta\right) X_i \quad \Leftrightarrow \quad 0 = \sum_{i=1}^{n} \frac{y_i - X_i^T \beta}{\sigma^2} X_i \, .$$

Thus in a normal linear model, $\mu(X_i^T \beta) = X_i^T \beta$ is the identity link so that $\mu'(\cdot) = 1$. This has the same form as (2), which suggests that the score equations for any GLM have the same form as (2).

An algorithm for fitting GLMs

The score equation (2) naturally suggests an algorithm for computing the MLE $\hat{\beta}$. Suppose we have an initial estimate $\beta^{(0)}$ of β. Then, we have an initial estimate $\mu'_{(0)} \equiv \mu'(X_i^T \beta^{(0)})$ of $\mu'(X_i^T \beta)$ and an initial estimate $V_i^{(0)}$ of $\text{Var}(Y_i)$. The score equation (2) can then be approximated by

$$0 = \sum_{i=1}^{n} \frac{y_i - \mu(X_i^T \beta)}{V_i^{(0)}} \mu'_{(0)} X_i.$$

Next, a linearization in β around $\beta^{(0)}$ gives

$$0 = \sum_{i=1}^{n} \frac{y_i - \mu(X_i^T \beta^{(0)})}{V_i^{(0)}} \mu'_{(0)} X_i - \sum_{i=1}^{n} \frac{\mu'(X_i^T \beta^{(0)})}{V_i^{(0)}} \mu'_{(0)} X_i X_i^T (\beta - \beta^{(0)})$$

Thus, if we write

$$U_{(0)} = \sum_{i=1}^{n} \frac{y_i - \mu(X_i^T \beta^{(0)})}{V_i^{(0)}} \mu'_{(0)} X_i \, ,$$

and

$$\mathcal{I}_{(0)} = \sum_{i=1}^{n} \frac{(\mu'_{(0)})^2}{V_i^{(0)}} X_i X_i^T \, ,$$

then we have a one-step update for β as

$$\beta^{(1)} = \beta^{(0)} + \mathcal{I}_{(0)}^{-1} U_{(0)}$$

We iterate these updates until there is minimal change in β.

Note that $\mathcal{I}$ is equal to $-E(\partial^2 l(\beta)/\partial\beta\partial\beta^T)$, which is the Fisher information matrix. For this reason, the above algorithm is also called *Fisher-scoring*.

Fortunately, we don't need to do these updates by hand. Computer software are very efficient in computing the MLE for GLMs, usually taking only a handful of iterations and a fraction of a second.

1.4.3 Standard errors and inferences

Estimation of parameters is only half the story in statistics. The other (and possibly more important) half is to quantify the precision of estimates and to make inferences on model parameters.

It can be shown that for large sample size n, $\hat{\beta}$ is asymptotically normal in distribution with mean β^*, and its covariance matrix is approximately:

$$\text{cov}(\hat{\boldsymbol{\beta}}) \approx \mathcal{I}^{-1}(\beta^*)\text{ , the Fisher information evaluated at true } \beta^*$$

In practice, we don't know β^* so we use the estimated Fisher information evaluated at the estimate $\hat{\beta}$:

$$\mathcal{I}(\hat{\beta}) = \sum_{i=1}^{n} \frac{(\mu'(X_i^T\hat{\beta}))^2}{\hat{\text{Var}}(Y_i)} X_i X_i^T \ .$$

(Marginal) confidence intervals for each β_j can then be obtained by the usual

$$\hat{\beta}_j \pm z_{\alpha/2}\hat{\text{se}}(\hat{\beta}_j)$$

where $\hat{\text{se}}(\hat{\beta}_j) = \sqrt{\mathcal{I}^{-1}(\hat{\beta})_{jj}}$, the square root of the jth diagonal component of $\mathcal{I}^{-1}(\hat{\beta})$. These confidence intervals will have approximately $(1-\alpha)100\%$ coverage even for moderate sample sizes.

Example (2. continued)**.** We demonstrate a full analysis of the disease and cholesterol dataset:

```
disease = c(rep(0,55), rep(1,45))
sex = c(rep(0,34), rep(1,21), rep(0,21),rep(1,24))
cholesterol = c(5.25, 5.03, 6.17, 5.32, 5.92, 5.88, 4.68, 6.56, 6.26, 4.68,
                5.55, 5.29, 5.12, 4.74, 5.37, 5.49, 5.00, 5.17, 4.83, 6.06,
                5.06, 5.53, 5.14, 4.59, 5.22, 4.95, 5.05, 5.06, 5.40, 4.60,
                5.86, 6.16, 4.96, 4.82, 5.68, 6.51, 4.82, 5.00, 6.49, 5.76,
                5.35, 5.89, 5.16, 6.14, 6.27, 5.69, 4.76, 5.58, 5.26, 5.87,
                5.31, 6.30, 6.45, 5.27, 5.96, 5.03, 5.72, 5.66, 4.59, 5.19,
                6.37, 5.46, 6.89, 5.02, 6.78, 5.87, 6.24, 6.85, 6.46, 5.04,
                5.89, 7.00, 6.47, 6.43, 4.87, 5.45, 6.74, 5.00, 4.78, 6.91,
                6.29, 6.23, 6.58, 5.98, 5.81, 6.75, 6.65, 6.19, 6.44, 6.96,
                6.91, 6.95, 6.12, 5.62, 6.85, 6.06, 6.08, 6.98, 6.89, 6.53)
fit1 = glm(disease~sex+cholesterol, family=binomial)
summary(fit1)
```

The default output is:

```
Call:
glm(formula = disease ~ gender + cholesterol, family = binomial)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
```

```
-1.6566  -0.8626  -0.5394   0.8735   2.0831

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  -9.3207     2.1703  -4.295 1.75e-05 ***
sex          -0.1095     0.4873  -0.225    0.822
cholesterol   1.5843     0.3834   4.132 3.59e-05 ***
---
Signif. codes:  0 `***' 0.001 `**' 0.01 `*' 0.05 `.' 0.1 ` ' 1

(Dispersion parameter for binomial family taken to be 1)
```

Observations: We see that cholesterol level has a effect on disease probability, but the effect of gender is given the cholesterol levels. A simpler model would therefore be one in which males and females have the same probability of disease given their cholesterol levels:

```
fit0 = glm(disease~cholesterol, family=binomial)
summary(fit0)
```

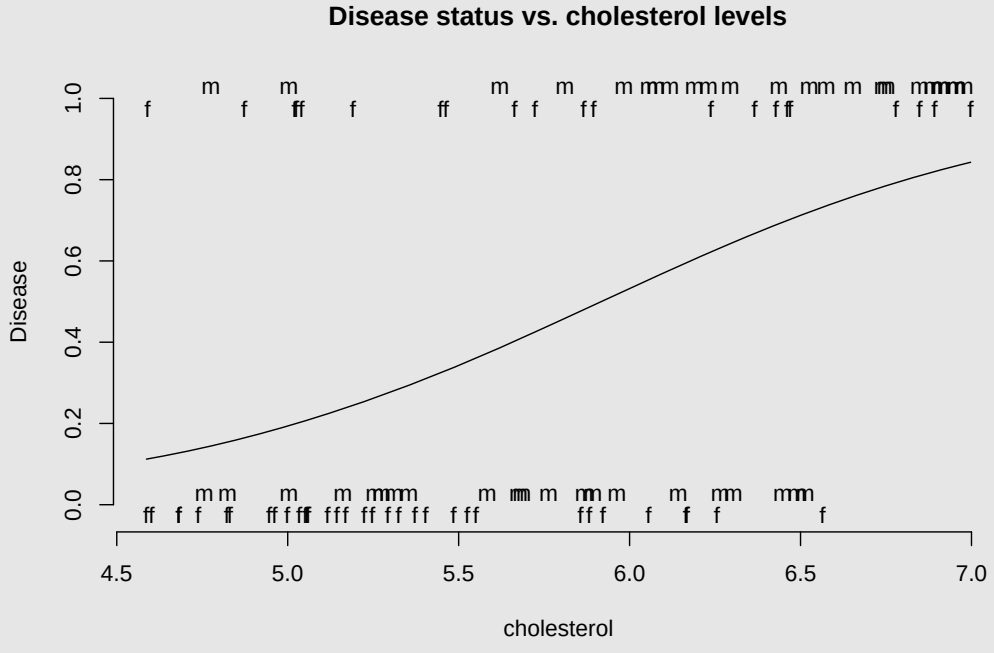
```
Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)   -9.202      2.098  -4.385 1.16e-05 ***
cholesterol    1.555      0.359   4.331 1.48e-05 ***
```

An approximate 95% confidence interval for the slope of cholesterol is

$$1.555 \pm 1.96 \times 0.359 = (0.85, 2.26)$$

We can therefore be 95% confident that a unit increase in cholesterol is associated with an increase in the odds of disease of between $\exp(0.85) = 2.34$ and $\exp(2.26) = 9.58$ times.

The final fitted mean model looks like this:



1.4.4 Models for Binomial counts

In experimental studies, we often apply treatments (covariates) to groups of units and count the total number of successes in that group. Note that the number of units in each group may be different. If the units within each group behave independently, we have binomial data instead of Bernoulli data. This poses no additional problem though, as the loglikelihood is exactly the same, except for an additive constant not involving β.

Example (3. Insecticide effectiveness)**.** The following table shows the numbers of dead beetles after five hours exposure to gaseous carbon disulphide at various concentrations (data from Bliss 1935):

Dose, x_i ($\log_{10} CS_2 mgl^{-1}$)	Number of beetles, n_i	Number killed, y_i
1.6907	59	6
1.7242	60	13
1.7552	62	18
1.7842	56	28
1.8113	63	52
1.8369	59	53
1.8610	62	61
1.8839	60	60

To fit a logistic model to these data in R, we encode the vector of **dose** as before, but we combine the success and failure counts into two columns in a **response matrix**:

```
dose = c(1.6907, 1.7242, 1.7552, 1.7842, 1.8113, 1.8369, 1.8610, 1.8839)
killed = c(6,13,18,28,52,53,61,60)
notkilled = c(53,47,48,28,11,6,1,0)
response = cbind(killed, notkilled)
fit = glm(response ~dose, family=binomial)
summary(fit)
```

The default R output is

```
Call:
glm(formula = response ~ dose, family = binomial)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.5098  -0.4385   0.9158   1.2845   1.5895

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  -61.249      5.206  -11.76   <2e-16 ***
dose          34.555      2.926   11.81   <2e-16 ***
---
```

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

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 288.940  on 7  degrees of freedom
Residual deviance:  12.141  on 6  degrees of freedom
AIC: 42.363

Number of Fisher Scoring iterations: 4
```

Conclusion: For each 0.01 unit increase in dosage, the odds of killing a beetle increases by an estimated $\exp(0.3456) = 1.41$ times. A 95% confidence interval for this increase is between $\exp(0.3456 - 1.96 \times 0.02926) =$ and $\exp(0.3456 + 1.96 \times 0.02926) =$ times.

1.5 Models for unbounded counts

In this section, we consider generalized linear models (GLMs) in which the response variables are nonnegative counts with no fixed upper bound. For example,

- In an insurance risk analysis, the number of claims over a given period of time as a function of insurance type and value of insured items.
- In a radiation study, the number of particles each second as recorded by a Geiger counter as a function on temperature and humidity.
- In behavioural studies, counts of incidents in a given time interval as a function of cognitive measurements.

The most popular (and arguably most natural) link function used for count response variables is a *log-linear model*,

$$\log(\mu_i) = X_i^\top \boldsymbol{\beta} \quad \text{or} \quad \mu_i = \exp(X_i^\top \boldsymbol{\beta})$$

This generates a mean-model that is nonnegative for any covariate value X and any parameter value β. The log-link is also the default link in R for count data, regardless of the discrete family used.

The second most popular link function for count data is a *linear model*,

$$\mu_i = X_i^\top \boldsymbol{\beta}$$

This is usually valid only over a narrow range of X's and only some values of β.

Comment: While a linear mean model may be a good enough fit to a particular dataset, there is a philosophical concern that the model is not a valid model for data that we *could have observed* but did not happen to observe. Moreover, over a narrow range of X's, an exponential mean curve is often indistinguishable from a linear mean curve. For these reasons, I tend to always use the log-link.

1.5.1 Poisson regression models

The *count* nature (0,1,2,3,...) of the responses is typically handled by the *Poisson distribution*, while the covariates determine the mean of the distribution:

$$Y_i|X_i \overset{\text{ind}}{\sim} \text{Poisson}\left(\mu_i = \exp(X_i^T\beta)\right)$$

Note that the Poisson model necessarily assumes that the conditional variance $\text{Var}(Y_i|X_i)$ is identical to the conditional mean μ_i for each observation. This may be too restrictive in some applications.

1.5.2 Parameter interpretation for log-linear models

The log-linear model is easier than the logistic model to interpret. In particular, the interpretation of β is very similar to that in a normal linear regression model.

Consider first a log-linear model with two covariates x_1 and x_2:

$$\log(\mu_i) = \beta_0 + \beta_1 x_1 + \beta_2 x_2$$

Then, a unit change in x_1 (keeping x_2 fixed) is associated with a change in the mean by a
of exp( ) times.

Similarly, a unit change in x_2 (keeping x_1 fixed) is associated with a change in the mean by a multiplicative factor of of $\exp(\beta_2)$ times.

The interpretation of the intercept β_0 is that the response mean when all covariates are at baseline is $\exp(\beta_0)$. Again, this may or may not have a meaningful interpretation

Example (4. Mine injuries)**.** Myers et al. (2010, pp. 181–183) describe a dataset on the number of injuries or fractures that occur in the upper seam of coal mines in West Virginia. A total of 44 observations were collected on mines in this area. In fitting a Poisson GLM to these data, Myers et al. (2010) found that three variables, namely, the inner burden thickness in feet (X_1), percent extraction of the lower previously mined seam (X_2), and time that the mine has opened (X_3), were important in explaining the number of injuries. The fitted mean model is

$$\hat{\mu} = \exp(-3.2707 - 0.0015X_1 + 0.0627X_2 - 0.0317X_3)$$

Interpret the intercept and slope(s) of the fitted model:

Intercept: the estimated intercept is , which implies that the mean number of injuries at baseline
( ) is estimated to be

$$\exp(\quad) = 0.038$$

This is not actually meaningful, because a mine with zero inner burden thickness is not really a mine...

Slope for X1: the estimated slope for X_1 is , which implies that an increase in inner burden thickness by a foot is associated with a reduction in the mean number of injuries by a factor of $exp(\quad) = 0.999$ times.

Slope for X_2: the estimated slope for X_2 is 0.0627, which implies that an increase in extraction by 1% is associated with an increase in the mean number of injuries by a factor of $\exp(0.0627) = 1.065$ times.

Slope for X_3: the estimated slope for X_3 is -0.0317, which implies that a mine that is one year older is expected to have fewer mean number of injuries by a factor of $\exp(-0.0317) = 1.032$ times.

1.5.3 Parameter estimation for Poisson regression

If we have a dataset of n independent Poisson responses, each having mean μ_i, the probability of observing $Y_1 = y_1, Y_2 = y_2, \ldots, Y_n = y_n$ is

$$\prod_{i=1}^{n} \frac{\mu_i^{y_i}}{y_i!} e^{-\mu_i} .$$

The log-likelihood function for β can then be written as

$$l(\beta) \quad = \quad \sum_{i=1}^{n} \{y_i \log(\mu_i) - \mu_i\} + \text{constant} ,$$

where $\mu_i = \exp(X_i^T \beta)$ for the log-link. We can now find the maximum likelihood estimate of β by maximizing $l(\beta)$ in β. To do this, we can set the derivative $\partial l / \partial \beta$ to 0:

$$0 = \frac{\partial l(\beta)}{\partial \beta} = \ldots = \sum_{i=1}^{n} \frac{y_i - \exp(X_i^T \beta)}{\exp(X_i^T \beta)} \exp(X_i^T \beta) X_i$$

which is again of the same form as

$$0 = \sum_{i=1}^{n} \frac{y_i - \mu_i}{\text{Var}(Y_i)} \mu'(X_i^T \beta) X_i .$$

1.5.4 Standard deviations and inferences

For large sample size n, $\hat{\beta}$ is asymptotically normal in distribution with mean β^* and its covariance matrix is approximately

$$\text{cov}(\hat{\boldsymbol{\beta}}) \approx \mathcal{I}^{-1}(\beta^*) , \text{ the Fisher information evaluated at true } \beta^*$$

In practice, we don't know β^* so we use the estimated covariance matrix evaluated at the estimate $\hat{\beta}$:

$$\mathcal{I}(\hat{\beta}) = \sum_{i=1}^{n} \frac{(\mu'(X_i^T \hat{\beta}))^2}{\hat{\text{Var}}(Y_i)} X_i X_i^T = \sum_{i=1}^{n} \exp(X_i^T \hat{\beta}) X_i X_i^T$$

(Marginal) confidence intervals for each β_j can then be obtained by the usual

$$\hat{\beta}_j \pm z_{\alpha/2} \hat{\text{se}}(\hat{\beta}_j)$$

where $\hat{\text{se}}(\hat{\beta}_j) = \sqrt{\mathcal{I}_{jj}^{-1}(\hat{\beta})}$, the square root of the jth diagonal component of $\mathcal{I}^{-1}(\hat{\beta})$. These confidence intervals will have approximately $(1-\alpha)100\%$ coverage even for moderate sample sizes.

Example (4, continued)**.** Myers et al. (2010, pp. 181–183) describe a dataset on the number of injuries (Y) that occur in the upper seam of coal mines in West Virginia, with covariates being the inner burden thickness in feet (X_1), percent extraction of the lower previously mined seam (X_2), and time that the mine has opened (X_3). The complete dataset is given below:

```
y = c(2,1,0,4,1,2,0,0,4,4,1,4,1,5,2,5,5,5,0,5,1,1,
      3,3,2,2,0,1,5,2,3,3,3,0,0,2,0,0,3,2,3,5,0,3)
x1 = c(50,230,125,75,70,65,65,350,350,160,145,145,180,43,42,42,45,83,300,190,145,510,
       65,470,300,275,420,65,40,900,95,40,140,150,80,80,145,100,150,150,210,11,100,50)
x2 = c(70,65,70,65,65,70,60,60,90,80,65,85,70,80,85,85,85,85,65,90,90,80,
       75,90,80,90,50,80,75,90,88,85,90,50,60,85,65,65,80,80,75,75,65,88)
x3 = c(1,6,1,0.5,0.5,3,1,0.5,0.5,0,10,0,2,0,12,0,0,10,10,6,12,10,
       5,9,9,4,17,15,15,35,20,10,7,5,5,5,9,9,3,0,2,0,25,20)
```

we can fit a log-linear Poisson model via

```
fit1 = glm(y~x1+x2+x3, family=poisson)
```

The summary output is

```
summary(fit1)


Call:
glm(formula = y ~ x1 + x2 + x3, family = poisson)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.7727  -0.9073  -0.0107   0.2716   2.1783

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -3.7206821  0.9788770  -3.801 0.000144 ***
x1          -0.0014793  0.0008244  -1.794 0.072757 .
x2           0.0627011  0.0122711   5.110 3.23e-07 ***
x3          -0.0316514  0.0163095  -1.941 0.052298 .
---
Signif. codes:  0 `***' 0.001 `**' 0.01 `*' 0.05 `.' 0.1 ` ' 1

(Dispersion parameter for poisson family taken to be 1)
```

Observations: percentage extraction is a highly significant predictor for the number of mine injuries, with inner burden thickness and age of the mine being borderline significant.

1.5.5 Negative-binomial regression models

A generalization of the Poisson model is the negative binomial, which is induced by a scale mixture of a Poisson(λ) with a gamma rate parameter λ. It can be characterized via:

$$E(Y) = \mu \quad \text{and} \quad \text{Var}(Y) = \mu + \mu^2/\nu, \text{ where } \nu > 0 \text{ is a dispersion parameter.}$$

Note that the variance is quadratic in the mean. It is always overdispersed compared to the Poisson distribution, but approaches the Poisson as a limiting case when $\nu \to \infty$.

Negative binomial regression models can be fit in R via `glm.nb` from the `MASS` package.

Example (5. Overdispersed class attendance data)**.** An attendance dataset examines the relationship between the number of days absent from high school and the gender, maths score (standardized score out of 100) and academic programme ("General", "Academic" and "Vocational") of 314 students sampled from two urban high schools. The dataset is included in the `mpcmp` package:

```
library(mpcmp)
data(attendance)
attach(attendance)
library(MASS)
fit1 = glm.nb(daysabs~gender+prog+math)
summary(fit1)

Coefficients:
               Estimate Std. Error z value Pr(>|z|)
(Intercept)    2.707484   0.204275  13.254  < 2e-16 ***
gendermale    -0.211086   0.121989  -1.730   0.0836 .
progAcademic  -0.424540   0.181725  -2.336   0.0195 *
progVocational -1.252615   0.199699  -6.273 3.55e-10 ***
math          -0.006236   0.002492  -2.502   0.0124 *
---
Signif. codes:  0 `***' 0.001 `**' 0.01 `*' 0.05 `.' 0.1 ` ' 1

(Dispersion parameter for Negative Binomial(1.0473) family taken to be 1)

    Null deviance: 431.67  on 313  degrees of freedom
Residual deviance: 358.87  on 309  degrees of freedom
AIC: 1740.3

Number of Fisher Scoring iterations: 1
```

Observations: The fitted model estimates that students in the General programme are expected to miss exp(+1.253) = 3.5 times more days of school compared to students in the Vocational programme, and exp(+0.425) = 1.52 times more days of school compared to the Academic programme.

Both of these comparisons are highly significant. Female students are estimated to miss an expected $\exp(+0.211) = 1.2$ times more days of school compared to male students, but this comparison is only borderline significant. A 10-point increase in maths scores is associated with a $\exp(-0.06) = 0.94$ times reduction in the expected days of absence from school, with this effect being rather significant. Finally, the estimated negative-binomial dispersion parameter of 1.0473 is much smaller than ∞, reflecting the strong overdispersion exhibited by the data. That is, even after conditioning on the explanatory variables, the variability in number of missed days is much larger than what a Poisson model would predict.

Indeed, fitting the standard Poisson model to the same dataset gives:

```
fit0 = glm(daysabs~gender+prog+math, family=poisson)
summary(fit0)


Coefficients:
                Estimate Std. Error z value Pr(>|z|)
(Intercept)    2.7594786  0.0637731  43.270  < 2e-16 ***
gendermale    -0.2424762  0.0467765  -5.184 2.18e-07 ***
progAcademic  -0.4260327  0.0567308  -7.510 5.92e-14 ***
progVocational -1.2707199  0.0779143 -16.309  < 2e-16 ***
math          -0.0069561  0.0009354  -7.437 1.03e-13 ***
```

Observations: We see that while the parameter estimates are all similar, the standard errors are an order of magnitude smaller and the p-values are far too significant. This is a consequence of the Poisson model not handling the conditional overdispersion exhibited in the data.

1.5.6 Conway-Maxwell-Poisson regression models

An alternative model that can handle both overdispersion and underdispersion is the Conway-Maxwell-Poisson (Conway & Maxwell, 1962). It has seen a recent resurgence in popularity ($>$ 1000 citations since 2005).

There are two flavours of Conway-Maxwell-Poisson distributions:

1. Shmueli et. al (2005, CMP):

$$f(y;\lambda,\nu) \propto \frac{\lambda^y}{(y!)^\nu}, \quad y = 0,1,2,\dots,$$

where λ is a latent rate parameter, $\nu \geq 0$ is a dispersion parameter
$\nu < 1 \Rightarrow$ overdispersion
$\nu = 1 \Rightarrow$ Poisson
$\nu > 1 \Rightarrow$ underdispersion
No closed form expression for the mean or variance.

2. Huang (2017, CMP_μ):

$$f(y;\mu,\nu) \propto \frac{\lambda(\mu,\nu)^y}{(y!)^\nu}, \quad y = 0,1,2,\ldots,$$

where μ is the mean, $\nu \geq 0$ is a dispersion parameter
$\nu < 1 \Rightarrow$ overdispersion
$\nu = 1 \Rightarrow$ Poisson
$\nu > 1 \Rightarrow$ underdispersion
No closed form expression for the variance.

Example (6. Number of takeover bids)**.** A dataset from Cameron & Johansson (1997) gives the number of bids received by 126 US firms that were successful targets of tender offers during the period 1978-85, along with the following set of explanatory variables:

- Defensive actions taken by management of target firm: indicator variable for legal defense by lawsuit (`leglrest`), proposed changes in asset structure (`rearest`), proposed change in ownership structure (`finrest`) and management invitation for friendly third-party bid (`whtknght`).
- Firm-specific characteristics: bid price divided by price 14 working days before bid (`bidprem`), percentage of stock held by institutions (`insthold`), total book value of assets in billions of dollars (`size`) and book value squared ($\texttt{size}^2$).
- Intervention by federal regulators: an indicator variable for Department of Justice intervention (`regulatn`).

A key feature of the dataset is that it exhibits strong underdispersion after accounting for the explanatory variables. We can fit a CMP_μ log-linear regression model to these data, accounting for this underdispersion, using the `mpcmp` package:

```
library(mpcmp)
data(takeoverbids)
attach(takeoverbids)
fit1 = glm.cmp(numbids~leglrest+rearest+finrest+whtknght+
                   bidprem+insthold+size+sizesq+regulatn)
summary(fit1)
```

```
Linear Model Coefficients:
             Estimate   Std.Err Z value Pr(>|z|)
(Intercept)  0.989630  0.435366   2.273 0.023020 *
leglrest     0.267879  0.122873   2.180 0.029248 *
rearest     -0.173177  0.154779  -1.119 0.263197
finrest      0.067744  0.174403   0.388 0.697693
whtknght     0.481281  0.131721   3.654 0.000258 ***
bidprem     -0.684822  0.307627  -2.226 0.026005 *
```

```
insthold    -0.367886  0.346799  -1.061 0.288780
size         0.179325  0.047627   3.765 0.000166 ***
sizesq      -0.007582  0.002485  -3.052 0.002276 **
regulatn    -0.037569  0.130303  -0.288 0.773101
---
Signif. codes:  0 `***' 0.001 `**' 0.01 `*' 0.05 `.' 0.1 ` ' 1

(Dispersion parameter for Mean-CMP estimated to be 1.752)
```

Model interpretation: For example, "A firm that engaged in a legal defense by lawsuit is estimated to have exp( ) = 1.31 times as many number of bids than a comparable firm that did not. The corresponding z-statistic is , which is significant at the 5% level".

The estimated dispersion parameter of 1.752 > 1 reflects a moderate level of underdispersion in the data.

Indeed, fitting the standard Poisson model to the same dataset gives:

```
fit0 = glm(numbids~leglrest+rearest+finrest+whtknght+
             bidprem+insthold+size+sizesq+regulatn, family=poisson)
summary(fit0)


Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)  0.986060   0.533920   1.847  0.06477 .
leglrest     0.260146   0.150959   1.723  0.08484 .
rearest     -0.195660   0.192631  -1.016  0.30976
finrest      0.074030   0.216522   0.342  0.73242
whtknght     0.481382   0.158870   3.030  0.00245 **
bidprem     -0.677696   0.376737  -1.799  0.07204 .
insthold    -0.361991   0.424329  -0.853  0.39361
size         0.178503   0.060022   2.974  0.00294 **
sizesq      -0.007569   0.003122  -2.425  0.01532 *
regulatn    -0.029439   0.160568  -0.183  0.85453
```

Here, the Poisson model gives similar parameter estimates, but the standard errors are slightly larger and the p-values are slightly weaker, reflecting the fact that the Poisson model does not account for the conditional underdispersion in this dataset.

1.5.7 Model diagnosis via Probability Inverse Transforms

Residual plots from count regression models exhibit banding artifacts due to the discreteness of the response. Instead of the usual residual plots, we can instead look at probability inverse transform

(PIT) plots.

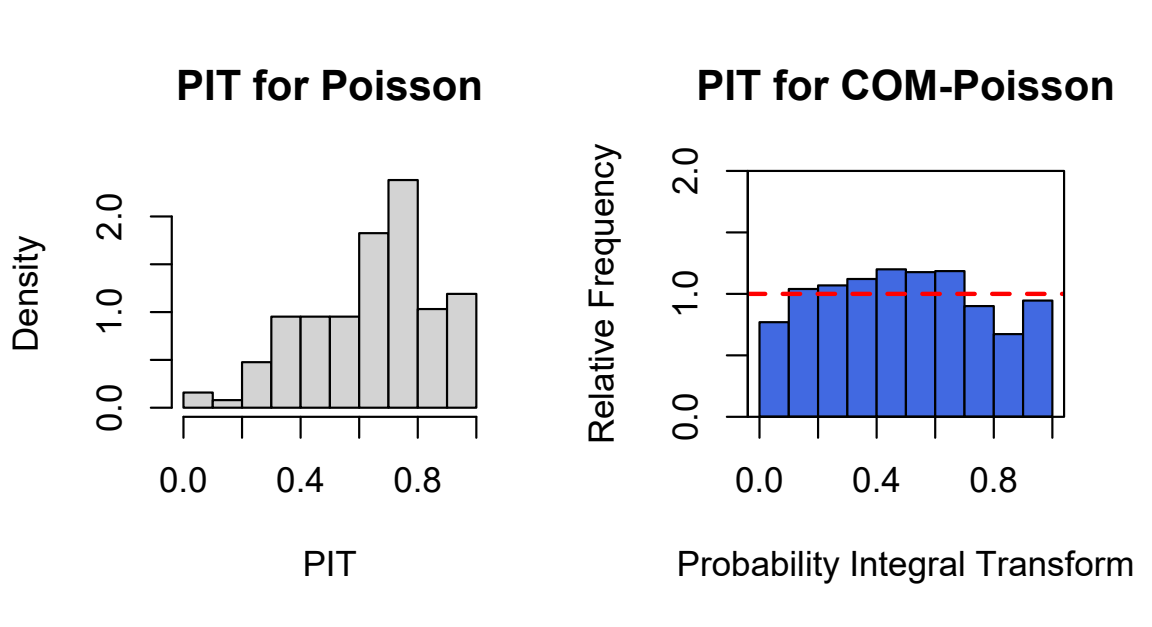
Recall that for a (continuous) random variable Y coming from some (cumulative) distribution F, we have $F(Y) \sim$Uniform$(0,1)$. Thus, if a fitted model is in fact a good fit to the data, then $\hat{F}(Y) \approx$Uniform$(0,1)$.

Example. (6. Number of bids, continued) For the Poison fitted model, the fitted cumulative probabilities $\hat{F}(Y_i)$ are given by

```
PIT = ppois(numbids, fit0$fitted)
PIT
  [1] 0.48651696 0.27185769 0.36570378 0.66622916 0.65690769 0.84017572
  [7] 0.72357890 0.71121364 0.81054462 0.31845063 0.73615084 0.69617021
      ...
[121] 0.91543884 0.96590461 0.79658764 0.83721443 0.40119278 0.07639473
```

If the fitted Poisson model is adequate, then these probabilities should resemble a random sample of 126 observations from a Uniform(0,1). We can compare the histogram of the PITs from the fitted Poisson model to the fitted CMP$_\mu$ model (using the built-in PIT plots from the `mpcmp` package for the latter):

```
par(mfrow=c(1,2))
hist(PIT, freq=F, main="PIT for Poisson")
histcompPIT(fit1)
```



The CMP-PIT plot is much closer to uniformity, indicating that it is the better fit of the two models. The Poison PIT exhibits a clear $\cap$ shape, which reflects the data not exhibiting enough "small" or "large" counts as predicted by the Poisson model due to underdispersion.

2 GLMs with nonparametric mean functions

R libraries and functions used in this chapter include:

```
library(mgcv) # cross-validation fitting of nonparametric splines
library(rpart) # source of kyphosis dataset
gam{mgcv} # fitting generalized additive models
plot{mgcv} # plotting function for gam objects
```

Parametric mean models are useful for relatively simple, easily interpretable analyses of data. They are particularly valuable in biological, engineering and agricultural contexts, where explicit *treatment effects*, or *covariate effects*, are of interest. The following should be familiar by now:

- in a linear model, $E(Y|X) = \beta_0 + \beta_1 x_1 + \ldots + \beta_p x_p$, we interpret β_j as the change in the expected response associated with a unit increase in x_j (fixing all other covariates)
- in a log-linear model, $E(Y|X) = \exp(\beta_0 + \beta_1 x_1 + \ldots + \beta_p x_p)$, we interpret $\exp(\beta_j)$ as the multiplicative change in the expected response associated with a unit increase in x_j (fixing all other covariates)
- in a logistic model, $\text{logit}(p) = \beta_0 + \beta_1 x_1 + \ldots + \beta_p x_p$, we interpret β_j as the change in the log-odds of success associated with a unit increase in x_j (fixing all other covariates)

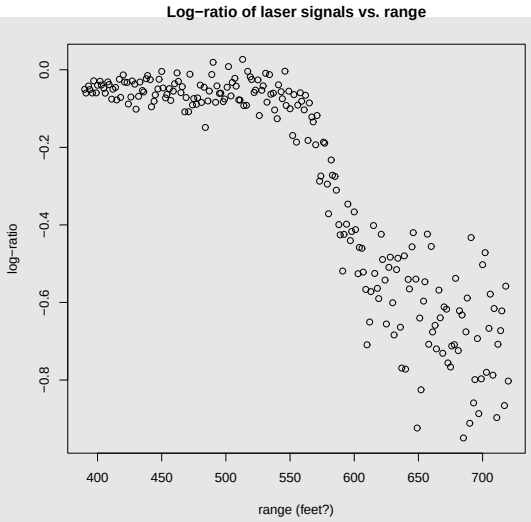
Of course, parametric models can be made more flexible by including non-linear terms in the X matrix, such as:

- polynomial effects, e.g. $X^T\beta = \beta_0 + \beta_1 x + \beta_2 x^2 + \ldots$
- transformation of covariates, e.g. $X^T\beta = \beta_0 + \beta_1 \log(x) + \ldots$
- interaction effects, e.g. $X^T\beta = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_{12} x_1 x_2 + \ldots.$

These non-linear effects can be combined with non-linear link functions to obtain a (very) wide range of models. However, the *specific parametric form of the terms in* X and the *specific form of the link function* have to be *chosen by the data analyst explicitly*.

Sometimes the link function and/or terms in the X matrix can be chosen on theoretical grounds. For example, if the responses are random variables governed by a rate parameter, then it often sensible to use the log-link so that we can speak about multiplicative changes in the rate. It is also sensible to think of certain covariates such as dosages and exposure times as being inherently multiplicative in magnitude, so that it would be natural to include $\log_2(\text{dose})$ or $\log_2(\text{exp.time})$ in the X matrix. However, there are often scenarios in which no specific form for the mean model stands out *a priori*.

Example (7. LIDAR data)**.** Ruppert, Wand & Carroll (2003) describe a light detection and ranging (LIDAR) dataset in which the responses (Y) are the log-ratios of received light from two laser sources and the covariate (x) is the distance travelled before the light is reflected back to its source. The data are plotted below.



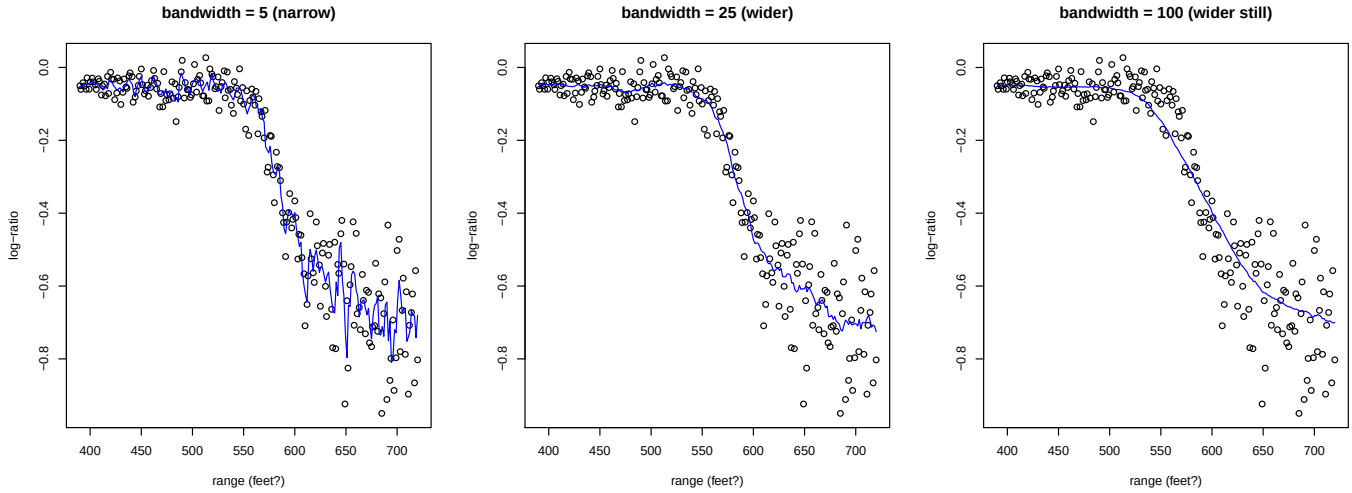
Comments:

In additional to the non-linear trend, the variances also seem to increase dramatically as the mean becomes more negative. This makes sense because laser signals not only get weaker over longer distances, but they are also expected to get noisier. We can handle non-constant variances using GLMs, so that should not concern us too much at this stage. The negative values might be a concern though, if we use gamma or inverse-gaussian families...

When the trends in the data are difficult to model with a relatively simple parametric model, we would like to let the data "speak for themselves" in a nonparametric and automated way. This motivates us to consider *scatterplot smoothing*, or *nonparametric regression*.

2.1 Nonparametric regression

We are all familiar with one method of nonparametric regression in the form of a LOWESS (locally weighted scatterplot smoothing) smoother. LOWESS smoothers are closely related to kernel smoothers and both estimate the conditional mean $E(Y|X)$ at each X by a (weighted) average of all data points in some neighbourhood of X. The size of the neighbourhood around each X is called the *bandwidth* (or window size) and plays a crucial role in the determining relative wiggliness of the fitted curve:



For noisier data, a larger bandwidth is required to get a cleaner signal. However, for less noisy data, an overly wide window will oversmooth the data and subtle fluctuations in the signal may be lost. If we do not account for possible non-constant variance, we may very well undersmooth noisier sections of the data and oversmooth more precise sections of the data. Since GLMs naturally account for non-constant variances, it makes sense to develop nonparametric regression methods within the GLM framework. A natural way to do this is via the roughness penalty approach of Green & Silverman (1994).

2.2 Nonparametric GLMs via roughness penalties

We first look at the simpler case with only one covariate x; we extend to two or more covariates a bit later on.

Recall that GLMs are generated from an exponential family of distributions:

$$Y_i|x_i \overset{ind}{\sim} \exp\left\{\frac{y\theta_i + c(\theta_i)}{e(\phi)}\right\} d(y;\phi) \,.$$

To obtain a more flexible GLM, we can replace the linear predictor $\theta_i = \beta_0 + \beta_1 x_i$ by simply

$$\theta_i = g(x_i) \,,$$

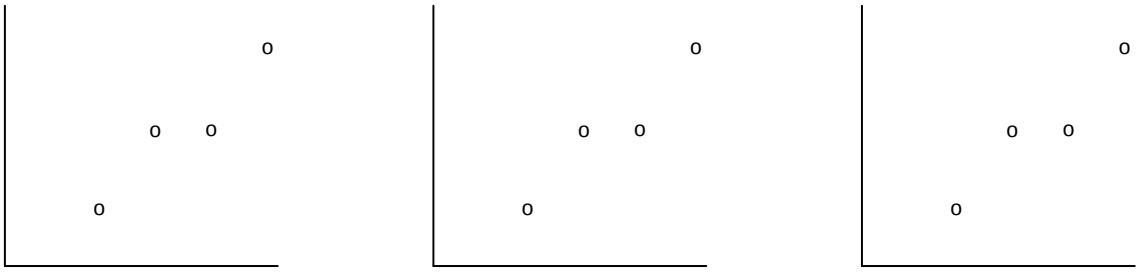
where g is some smooth but otherwise unknown function of x. This implies that the mean curve is

$$E(Y|x) = \mu(g(x_i)) \,,$$

also some smooth function in x.

2.2.1 How smooth is "smooth"?

If we attempt to find the best fit over all smooth functions g, the result is useless: it is always possible to choose g sufficient wiggly (but still "smooth") so that it essentially interpolates the data:



We need some way of measuring the wiggliness of a function, so that we can penalize functions g that are overly wiggly and favour simpler models that still provide adequate fits to the data.

Discuss: how might *you* quantify the wiggliness of a function?

The roughness penalty approach to nonparametric GLMs (Green & Silverman, 1994) is a natural way to formulate the problem:

1. replace linear predictor $\beta_0 + \beta_1 x$ with smooth but arbitrary $g(x)$.
2. connect mean response to $g(x)$ via the natural (canonical) link of the underlying exponential family:
$$E(Y|x) = \mu(g(x))$$
   - normal: $\mu(g(x)) = g(x)$, the identity link
   - Poisson: $\mu(g(x)) = \exp(g(x))$, the log-link
   - Bernoulli: $\mu(g(x)) = \exp(g(x))/(1 + \exp(g(x)))$, the logistic link.
3. estimate g by maximizing a penalized log-likelihood,
$$\sum_{i=1}^{n} \{Y_i g(x_i) - c(g(x_i))\} - \frac{1}{2}\lambda \int \{g''(x)\}^2 dx \,,$$
   over *all* functions g that are twice continuously differentiable.
   - as it turns out, the maximum penalized likelihood estimator $\hat{g} \equiv \hat{g}_\lambda$ is necessarily a cubic spline with knots at each x_i
4. the estimated mean function is then given by
$$\hat{E}(Y|x) = \mu(\hat{g}(x)) \,.$$

2.3 Fitting nonparametric GLMs in practice

The roughness penalty approach is mathematically elegant and leads *automatically* to a cubic spline estimator of g. However, it is not computationally efficient to use in practice, because it places a knot at each (unique) covariate value. This can pose a major computational challenge for larger sample sizes.

An alternative approach is to assume a cubic spline for g a priori and to choose only a handful of knot locations, either manually or in a systematic way. A typical default is to choose 10 knots, equally spaced between $x_{(1)}$ and $x_{(n)}$. Another common default is to use the deciles of x.

Another point of contention is the smoothing parameter λ, which determines the smoothnes of the fitted model. Larger λs penalize _____ in favour of _____.

Indeed, if $\lambda \to \infty$, we end up with _____, which leads to a classical parametric GLM. If $\lambda \to 0$, we end up with _____.

Interpolation may give the "best fit" to a dataset, but lacks predictive or inferential power, making the fitted model _____! We want to choose the penalty parameter λ to obtain an *adequate* fit to the data without overfitting.

2.3.1 Choosing smoothness via cross-validation

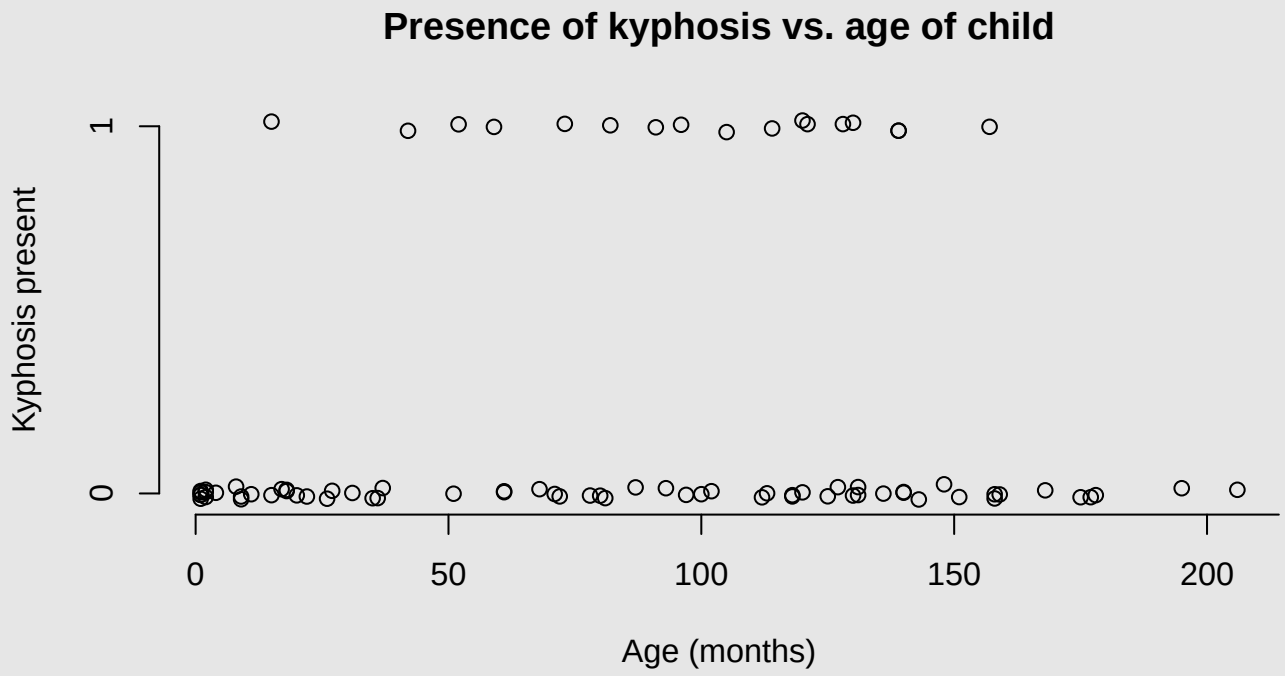
One of the most popular ways to choose the parameter λ is via *cross-validation*. The main idea behind cross-validation involves three simple steps:

1. hide part of the data
2. fit a model using only the remaining data
3. see how well our fitted model explains the data we initially hid

We then select λ that performs the best under these three steps. This idea is closely related to dividing data into training sets and testing sets in computer science and machine learning.

In many software packages, the smoothness parameter λ is not explicitly computed. Rather, a related parameter called the *(equivalent) degrees of freedom* is computed. This can be interpreted as a measure of how smooth a fitted $\hat{g}$ is compared to a polynomial fit. There is a one-to-one relationship between λ and the degrees of freedom.

Example (8. Kyphosis in children)**.** 81 children who have had corrective spinal surgery were followed up to see if kyphosis (a type of deformation) was present after the operation. We are interested in how the incidence rate of developing kyphosis changes with the age of children. The data are plotted below.



A quick look at the data suggests that a logistic model may not be appropriate for these data. Instead, we may consider a nonparametric binary GLM. That is,

$$P(\text{kyphosis}|\text{Age}) = \frac{\exp(g(\text{Age}))}{1+\exp(g(\text{Age}))},$$

where $g(\text{Age})$ is some arbitrary, smooth function of Age. Typically, an intercept term is explicitly included in the model,

$$P(\text{kyphosis}|\text{Age}) = \frac{\exp(\beta_0+g(\text{Age}))}{1+\exp(\beta_0+g(\text{Age}))},$$

so that $g(\text{Age})$ is some arbitrary, smooth function of Age centred around 0. This does **not** change the model, but makes things a bit easier to interpret.

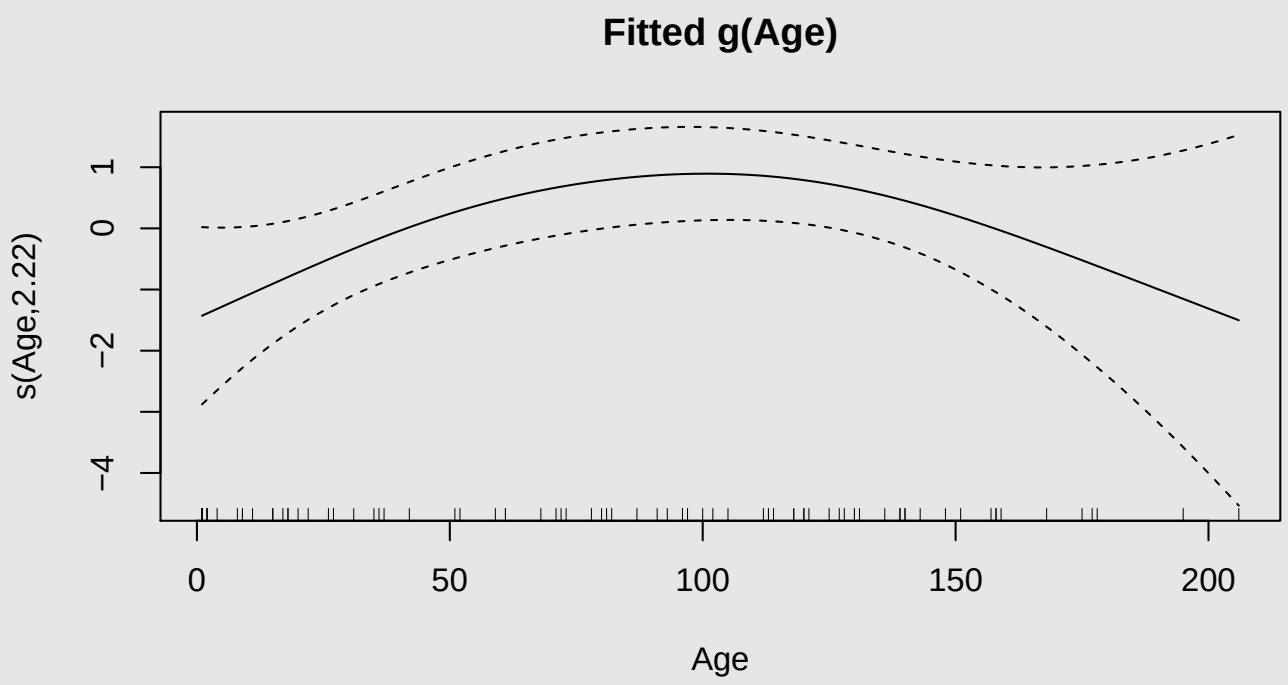
To fit this model in R, we can use the gam function in the mgcv package. The kyphosis dataset comes from the `rpart` package.

```
library(rpart)
data(kyphosis)
attach(kyphosis)
library(mgcv)
fit.1 = gam(Kyphosis ~s(Age), family=binomial)
```

Here, $s(\cdot)$ tells R that you want a "smooth" fit in Age. The gam function in the mgcv package automatically chooses the number of knots, knot locations and the smoothness, generally by cross-validation.

To see the fitted $\hat{g}$ function, we can use the plot command:

```
plot(fit.1)
```



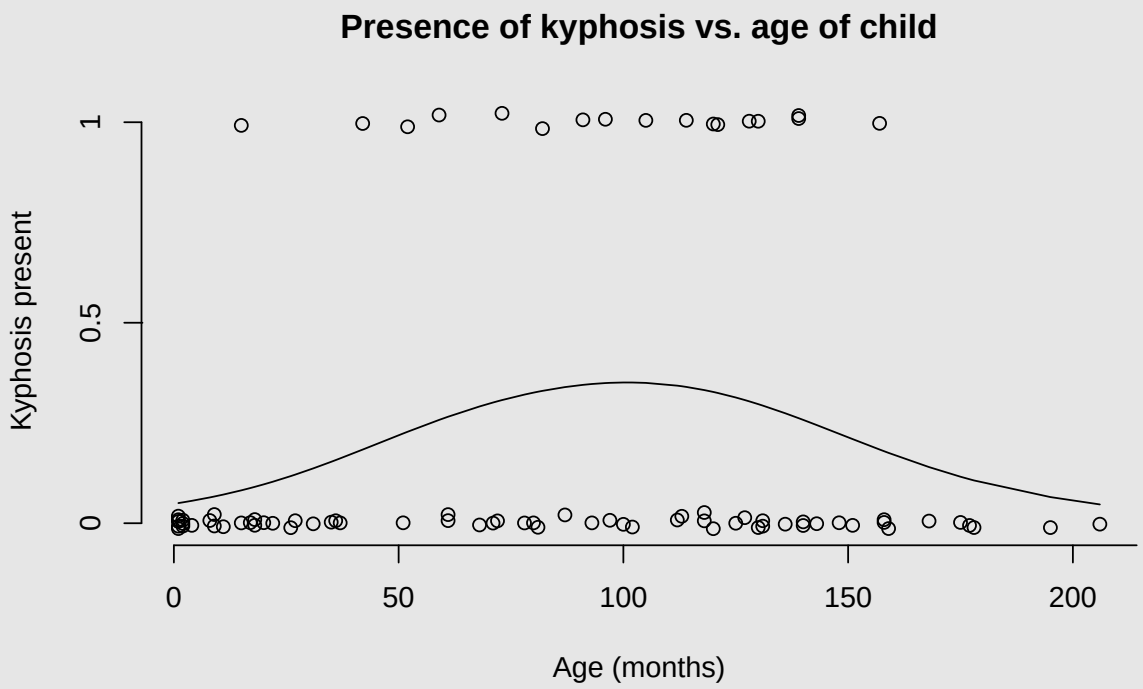
Here, the 2.22 is the equivalent degrees of freedom of the fit, as chosen by cross-validation. This tells us that the estimated $\hat{g}$ has complexity somewhere between a quadratic and a cubic polynomial. Note that $\hat{g}$ is itself a piecewise cubic polynomial with certain continuity constraints, but its overall "complexity" is 2.22.

The two sets of dotted lines are (simultaneous) $\pm 2 \times se$ confidence bands for $\hat{g}$. The confidence band (almost) includes 0 across all Ages, so we might be able to conclude that Age is actually not a significant predictor for kyphosis. We can look at the summary of our fitted model:

```
summary(fit.1)

Parametric coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)   -1.509      0.316  -4.776 1.78e-06 ***
---
Signif. codes:  0 `***' 0.001 `**' 0.01 `*' 0.05 `.' 0.1 ` ' 1

Approximate significance of smooth terms:
         edf Ref.df Chi.sq p-value
s(Age) 2.223  2.806  6.693  0.0714 .
---
Signif. codes:  0 `***' 0.001 `**' 0.01 `*' 0.05 `.' 0.1 ` ' 1

R-sq.(adj) =  0.0759   Deviance explained = 11.6%
```

The smooth function in Age is borderline significant, so we should keep it. We can add the fitted probability curve to our plot of the data:

```
lines(sort(Age), fit.1$fitted[order(Age)])
```



When we fit a nonparametric GLM our answer to questions like "How is kyphosis incidence associated with Age of children?" is just the above plot. Having an extremely flexible modelling framework comes at the cost of longer being able to give simple interpretations like "a unit increase in x is associated with..."

2.4 More than one covariate

There are various ways to extend the nonparametric GLM approach to scenarios with two or more covariates, although some of these models can be quite opaque in their interpretations. It suffices to consider three covariates and how to replace the linear predictor $\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3$ by a "smooth" function of x_1, x_2 and x_3.

2.4.1 Generalized Additive Models

Generalized additive models (Hastie & Tibshirani, 1990) replace the linear predictor with a sum of arbitrary, smooth univariate functions:

$$\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 \Longrightarrow \beta_0 + g_1(x_1) + g_2(x_2) + g_3(x_3)$$

where g_1 is an arbitrary, smooth function of x_1, g_2 is an arbitrary, smooth function of x_2 and g_3 is an arbitrary, smooth function of x_3. The effects of x_1, x_2 and x_3 are additive on the canonical scale, hence the name of these kinds of models.

Generalization to more than three covariates is immediate:

$$\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_p x_p \Longrightarrow \beta_0 + g_1(x_1) + g_2(x_2) + \ldots g_p(x_p)$$

Example (8. Kyphosis, continued)**.** In addition to Age, two other covariates were recorded, namely, Number: the number of vertabrae involved, and Start: the position of the top-most vertebra operated on. We can fit an additive nonparametric binary GLM to these data using all three covariates. Formally, this model is

$$P(\text{kyphosis}|\text{Age}) = \frac{\exp\{\beta_0 + g_1(\text{Number}) + g_2(\text{Start}) + g_3(\text{Age})\}}{1 + \exp\{\beta_0 + g_1(\text{Number}) + g_2(\text{Start}) + g_3(\text{Age})\}},$$

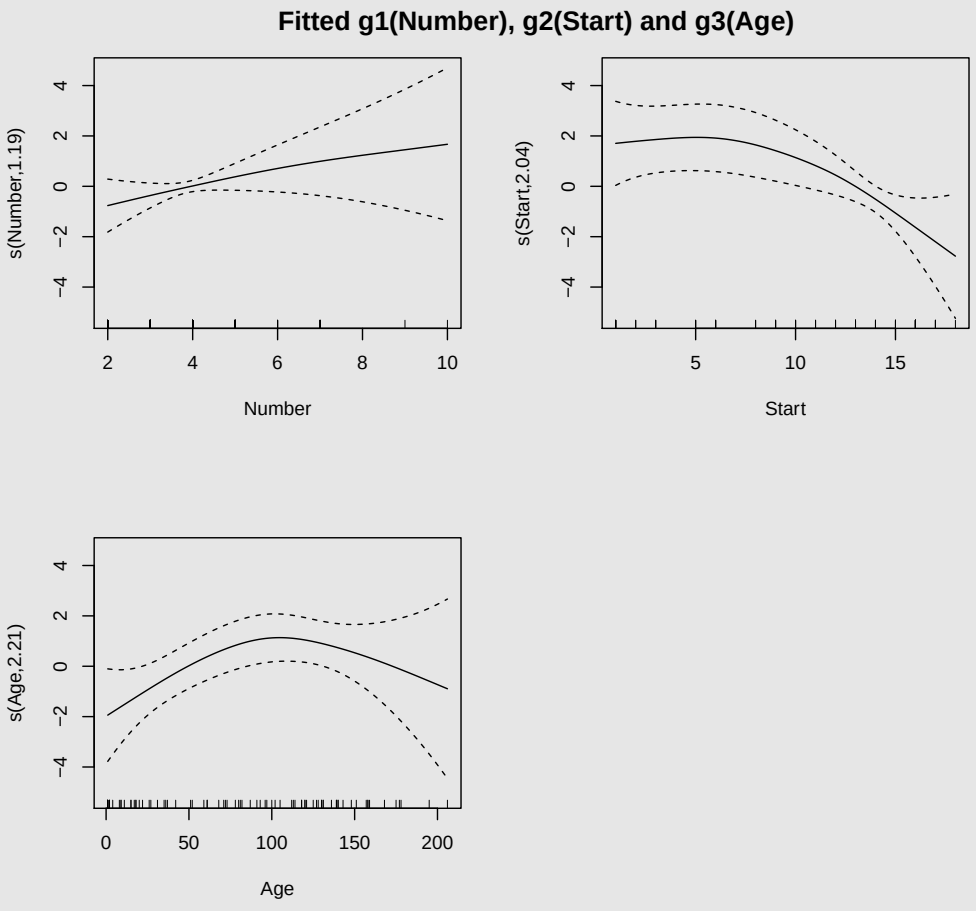
We can fit this model using the gam function in the mgcv package, paying special attention to the fact that we only have 8 unique values of Number and 16 unique values of Start:

```
fit.3 = gam(Kyphosis ~s(Number,k=8) + s(Start, k=16) + s(Age), family=binomial)
```

The additional k argument here tells R that we cannot exceed $k = 8$ degrees of freedom for the smooth function in Number and $k = 16$ degrees of freedom for the smooth function in Start.

We can look at our fitted $\hat{g}_1(\text{Number}), \hat{g}_2(\text{Start})$ and $\hat{g}_3(\text{Age})$ functions using the `plot` command:

```
plot(fit.3, pages=1)
```



We see that the effect of Number is almost exactly linear on the log-odds scale. The effects of Start and Age have complexity somewhere between a quadratic and cubic. As usual, each effect must be interpreted whilst fixing the other covariates.

2.4.2 Arbitrary nonparametric GLMs

Arbitrary nonparametric GLMs are obtained by replacing the linear predictor with a completely arbitrary but smooth multivariate function of all covariates:

$$\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 \Longrightarrow \beta_0 + g(x_1, x_2, x_3) \,,$$

where g is an aribtrary smooth function of x_1, x_2 and x_3. These are (almost) computationally impossible to fit in general, and perhaps even more impossible to interpret.

2.4.3 Partially-linear models

Partially linear, or semiparametric, GLMs assume that the effect of some covariates can be modeled with simple parametric forms, with the effects of other covariates having arbitrary smooth forms:

$$\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 \Longrightarrow \beta_0 + \beta_1 x_1 + g_2(x_2) + g_3(x_3) \,,$$

or

$$\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 \Longrightarrow \beta_0 + \beta_1 x_1 + \beta_2 x_2 + g_3(x_3) \,,$$

or

$$\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 \Longrightarrow \beta_0 + \beta_1 x_1 + g(x_2, x_3) \,.$$

Partially-linear GLMs are somewhere between fully parametric and generalized additive or abitrary nonparametric GLMs. They also have interpretability that is between the two extremes. For example, in the first case, β_1 is the change (on the canonical scale) in the mean response associated with a unit increase in x_1, fixing all the other covariates.

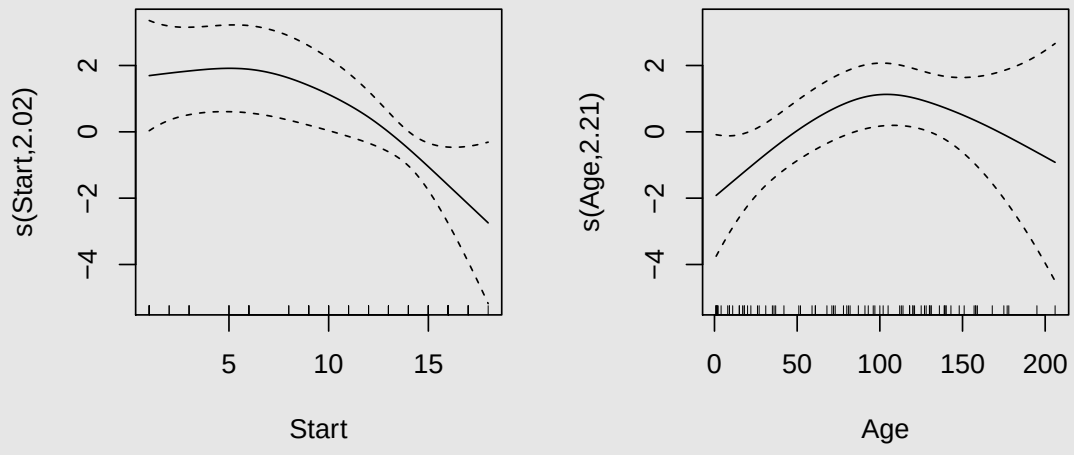
A useful rule of thumb is to consider arbitrary smooth fits for covariates that are not of primary interest (e.g. ambient temperature, weight of patient...), but try to use simple parametric forms for the covariate effects that *are* of primary interest (e.g. treatment effects, dosage effects...). This way, you can examine the effects of the things you want to look at, whilst adequately adjusting for systematic effects due to nuisance variables.

Example (8. Kyphosis, continued)**.** We can try fitting the following partially-linear binary model to the data:

$$P(\text{kyphosis}|\text{Age}) = \frac{\exp\{\beta_0 + \beta_1 \text{Number} + g_2(\text{Start}) + g_3(\text{Age})\}}{1 + \exp\{\beta_0 + +\beta_1 \text{Number} + g_2(\text{Start}) + g_3(\text{Age})\}} \,.$$

```
> fit.sp = gam(Kyphosis ~Number + s(Start, k=16) + s(Age), family=binomial)
> summary(fit.sp)

Parametric coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  -3.5926     1.1464  -3.134  0.00173 **
Number        0.3333     0.2324   1.434  0.15150
---
Signif. codes:  0 `***' 0.001 `**' 0.01 `*' 0.05 `.' 0.1 ` ' 1

Approximate significance of smooth terms:
           edf Ref.df Chi.sq p-value
s(Start) 2.025  2.531  9.957  0.0131 *
s(Age)   2.209  2.790  6.675  0.0711 .
---
Signif. codes:  0 `***' 0.001 `**' 0.01 `*' 0.05 `.' 0.1 ` ' 1

R-sq.(adj) =  0.355   Deviance explained = 39.4%
```

From the summary output, we see that given the smooth terms in Start and Age are in the model, the covariate Number is no longer significant. We can simplify the model by dropping Number, but we should probably keep Age in the model. The most important covariate is the starting vertebra, with an estimated effect that looks like this on the log-odds scale:



For children who have spinal surgery starting on higher vertebra (vertebrae 1–5), the odds of developing kyphosis is estimated to be around _____ times higher than children who had surgery starting on the lowest vertebra.

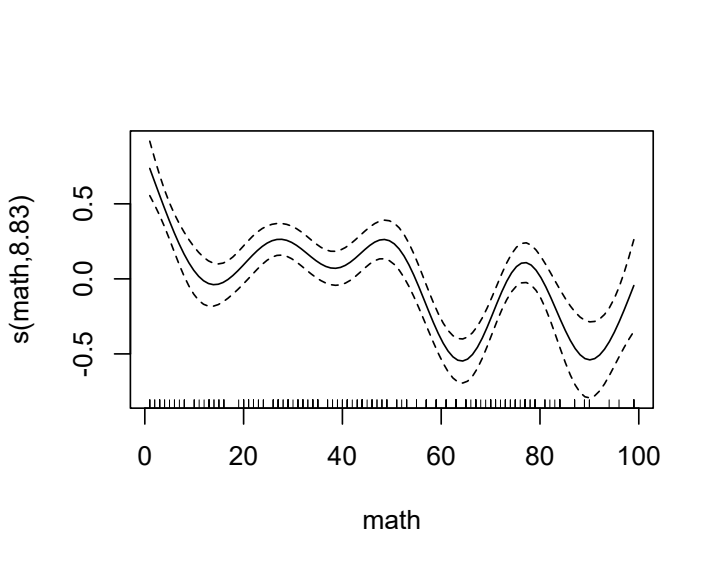
2.5 Nonparametric regression models for counts

In a similar way, we can develop generalized additive and partially linear models for unbounded counts.

Example (5. Class attendance, continued)**.** If we were to ignore possible overdispersion, then we can fit a partially linear model with main effects in gender and programme and a smooth function in math scores via:

```
fit0 = gam(daysabs~ gender+prog+s(math), family=poisson)
summary(fit0)
plot(fit0)
```

```
Parametric coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept)    2.35894    0.05370  43.928  < 2e-16 ***
gendermale    -0.23822    0.04762  -5.002 5.66e-07 ***
progAcademic  -0.38975    0.05750  -6.778 1.22e-11 ***
progVocational -1.20888    0.07871 -15.360  < 2e-16 ***
---
Signif. codes:
0 `***' 0.001 `**' 0.01 `*' 0.05 `.' 0.1 ` ' 1

Approximate significance of smooth terms:
          edf Ref.df Chi.sq p-value
s(math) 8.827  8.991  130.4  <2e-16 ***
```

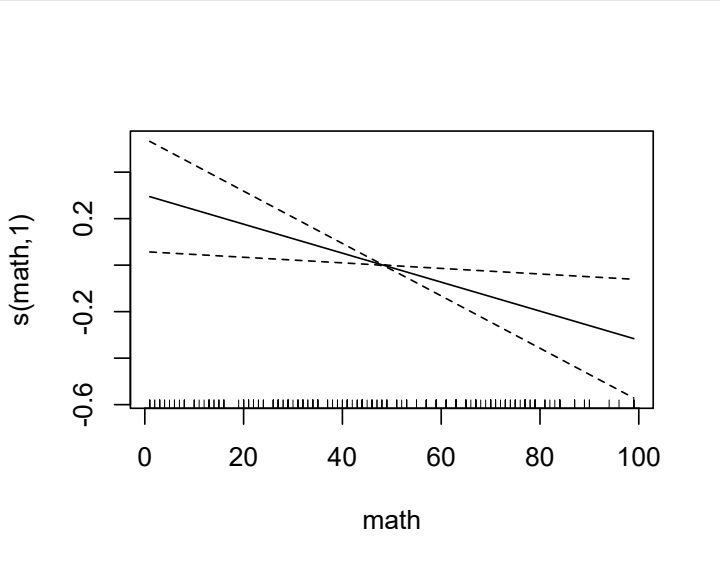


When using a conditional Poisson model, the smooth curve in maths scores looks overly complex. This is because the data shows too much fluctuation relative to a Poisson model, and so the partially linear model tries to account for this fluctuation by getting the mean curve closer to the observed values.

If, instead, we consider a conditional negative-binomial response distribution, the fitted model looks quite different:

```
fit1 = gam(daysabs~gender + prog + s(math), family=nb)
summary(fit1)
plot(fit1)
```

```
Parametric coefficients:
               Estimate Std. Error z value Pr(>|z|)
(Intercept)      2.4064     0.1731  13.904  < 2e-16 ***
gendermale      -0.2109     0.1231  -1.713   0.0867 .
progAcademic    -0.4245     0.1836  -2.313   0.0207 *
progVocational  -1.2525     0.2016  -6.213  5.2e-10 ***
---
Signif. codes:
0 `***' 0.001 `**' 0.01 `*' 0.05 `.' 0.1 ` ' 1

Approximate significance of smooth terms:
          edf Ref.df Chi.sq p-value
s(math) 1.001  1.001  6.136  0.0133 *
```



The negative-binomial model allows for extra variability around the fitted mean, and so we don't need the mean function to follow the observed counts so closely. Interestingly, the cross-validated "best" model is essentially a linear model in math scores, which coincides with our previous parametric analysis of the dataset.

3 GLMs with nonparametric response distribution

R libraries and functions used in this chapter include:

```
library(gldrm)
gldrm{gldrm} #fitting GLMs with nonparametric response distributions
gldrmPIT{gldrm} #PIT plots for model diagnostics
```

Another, somewhat orthogonal, direction in which we can relax the parametric GLM is to allow the conditional response distribution to be nonparametric. In all our examples, we could write the exponential family in a linear canonical form,

$$f(y;\theta) = \exp\{y\theta + c(\theta)\}\, d(y)$$

where we have absorbed the given/known dispersion parameter ϕ into θ, $c(\theta)$ and $d(y)$. The base measure $d(y)$ can be viewed as some density $dF(y)$ of a cumulative distribution F with respect to counting measure (or Lesbegue measure for continuous exponential families). The mean of the distribution is given by

$$\mu = \sum_{y=0}^{\infty} y \exp\{y\theta + c(\theta)\} dF(y) = \exp(c(\theta)) \sum_{y=0}^{\infty} y \exp\{y\theta\} dF(y)$$

Hence, for a given mean $\mu > 0$ and a given base measure F, the canonical parameter $\theta \equiv \theta(\mu, F)$ can be defined as a function of both the mean μ and base measure F via the solution to

$$0 = \sum_{y=0}^{\infty} (y - \mu) \exp\{y\theta\} dF(y)\,, \tag{3}$$

with the normalizing function c also being a function of μ and F via

$$c = -\log \sum_{y=0}^{\infty} \exp\{y\theta(\mu, F)\} dF(y) \tag{4}$$

Writing the model in this way allows us to view both the mean μ and the base measure F as parameters. For count data, the mean itself is typically modelled as $\mu = \exp(X^\top \beta)$ for some finite set of covariates X, and so the mean-parameters β are finite-dimensional. On the other hand, the parameter space for F is all distributions on $\mathbb{N}$ that have a cumulant generating function, and so is infinite-dimensional. Indeed, writing GLMs in this form covers the class of **all** discrete GLMs, and if we are able to maximize the likelihood over both β and F then we would have maximized over the space of **all** discrete GLMs!

Huang (2014) was able to show that this joint estimation is not only possible, but that little-to-no information is lost when estimating β while also simultaneously estimating F! The key intuition behind why this is achievable is that the mean μ and the base measure F in any GLM are in fact orthogonal parameters. While various estimation methods for F can be used, one particularly attractive method is to consider a histogram estimator that places probability mass only at the observed supports. This choice is optimal in the sense that it maximizes the so-called empirical likelihood.

Joint estimation of both the mean model and response distribution can be done in R using the `gldrm` package. GLDRM stands for "generalized linear density ratio models" because the linear canonical exponential family is a special case of a density ratio model.

Example (5. Class attendance, continued)**.** Instead of specifying a particular response distribution for the number of absent days, such as the Poisson or negative-binomial, we can instead estimate it along with the mean model via

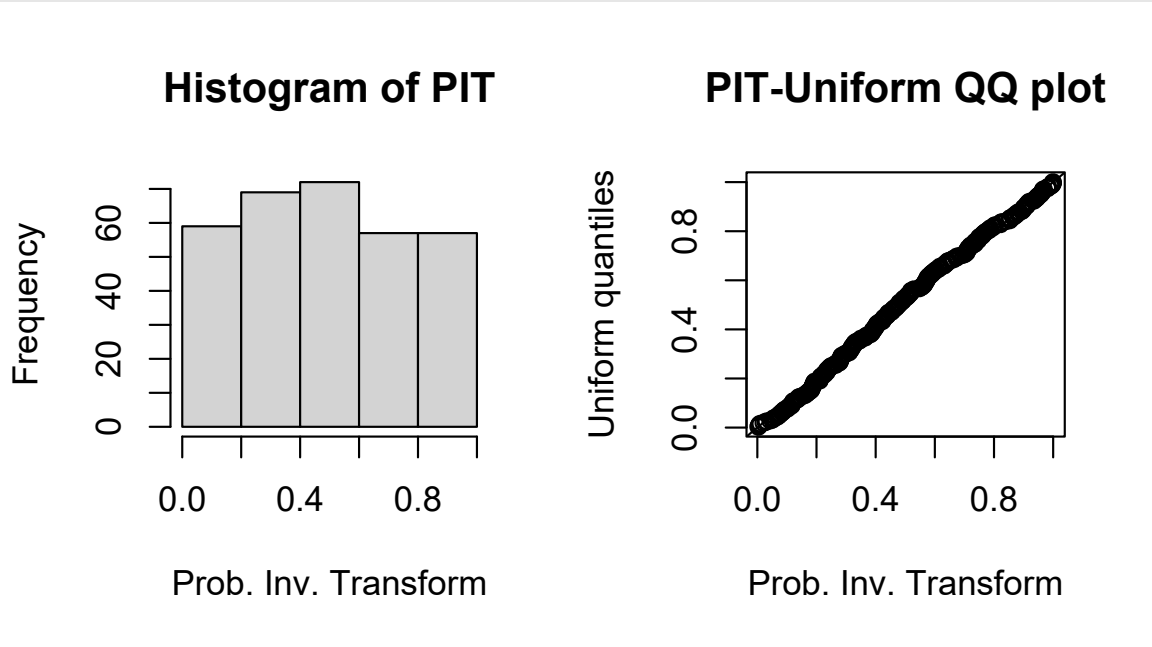
```
fit.sp = gldrm(daysabs~gender+prog+math, link="log")
fit.sp
```

```
Summary of gldrm fit

Coefficients:
               Estimate Std. Error t value Pr(>|t|)
(Intercept)     2.72318    0.17442   15.61  < 2e-16 ***
gendermale     -0.21835    0.11680   -1.87   0.0625 .
progAcademic   -0.42809    0.15845   -2.70   0.0073 **
progVocational -1.25891    0.18314   -6.87  3.5e-11 ***
math           -0.00637    0.00236   -2.70   0.0073 **
```

We see that while the estimated parameters are again similar, the standard errors here are somewhere between the negative-binomial fit and the Poisson fit to the data. This suggests that the data may be distributed as neither of the two parametric models.

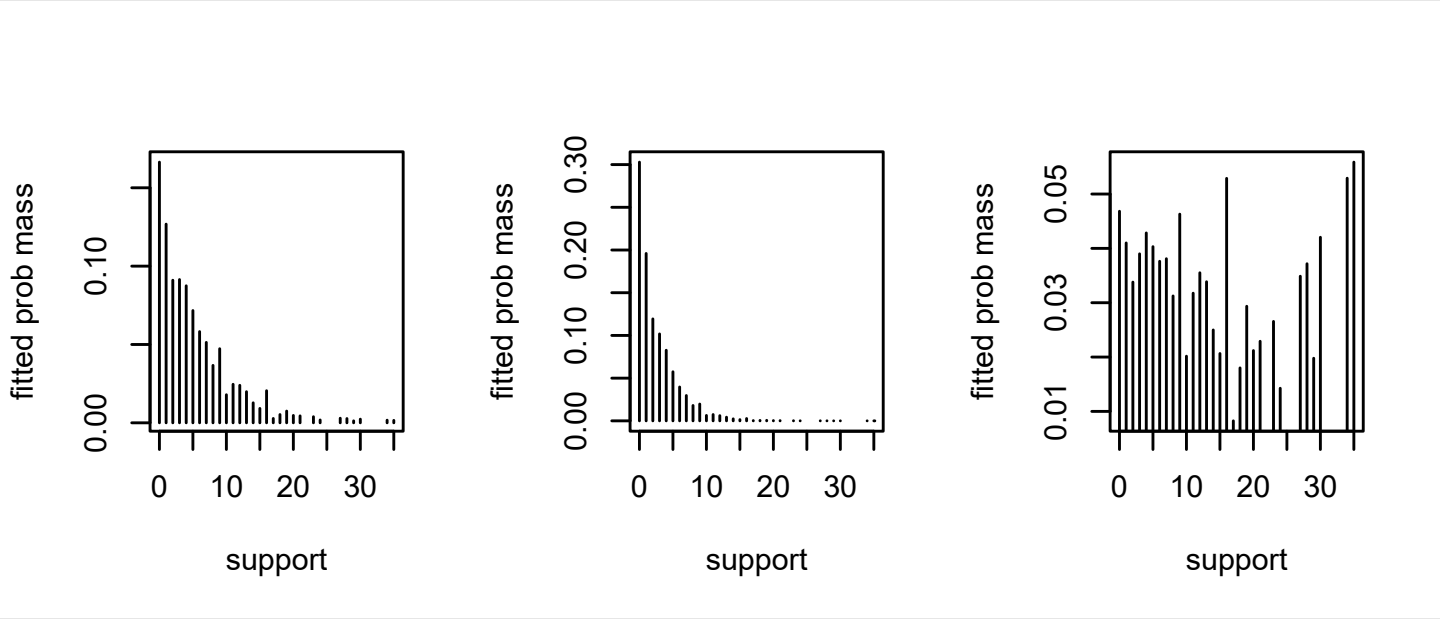
We can examine the goodness-of-fit of this semiparametric approach via PIT plots using the in-built function

```
gldrmPIT(fit.sp)
```



We see from the uniformity of these PIT plots that the estimated nonparametric base measure F has been able to adapt to the response distribution very well. We can examine the shape of the estimated response distributions at a few observations:

```
fit.sp$mu[1]
[1] 5.339606
fit.sp$mu[10]
[1] 2.543572
fit.sp$mu[107]
[1] 15.13189

support = fit.sp$spt
fitted.Fs = fit.sp$fTiltMatrix

par(mfrow=c(1,3))
plot(support, fitted.Fs[1,], type="h", ylab = "fitted prob mass", xlab = "support")
plot(support, fitted.Fs[10,], type="h", ylab = "fitted prob mass", xlab = "support")
plot(support, fitted.Fs[107,], type="h", ylab = "fitted prob mass", xlab = "support")
```



Example (6. Takeover bids, continued)**.** Instead of specifying a particular response distribution for the number of absent days, such as the Poisson or Conway-Maxwell-Poisson, we can instead estimate it along with the mean model via

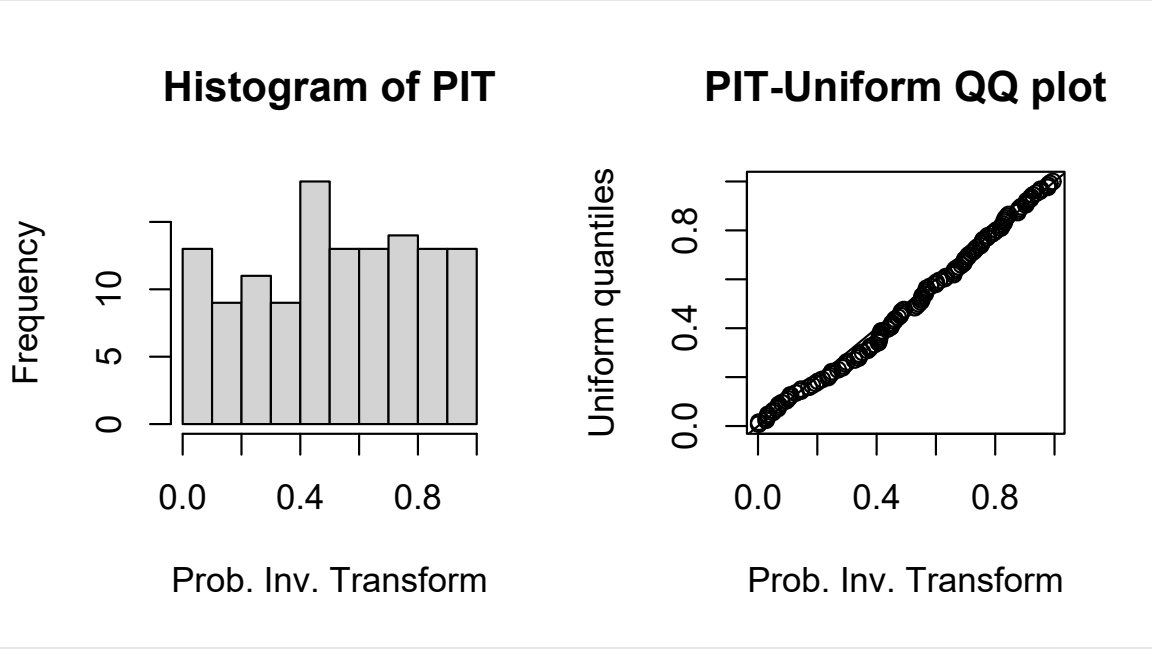
```
fit.sp = gldrm(numbids~leglrest+rearest+finrest+whtknght+
     bidprem+insthold+size+sizesq+regulatn, link="log")
fit.sp

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  1.16638    0.43465    2.68   0.0084 **
```

```
leglrest     0.21650    0.12498     1.73   0.0859 .
rearest     -0.37596    0.16625    -2.26   0.0256 *
finrest      0.14862    0.19772     0.75   0.4538
whtknght     0.49771    0.11840     4.20  5.2e-05 ***
bidprem     -0.78438    0.30802    -2.55   0.0122 *
insthold    -0.36539    0.33903    -1.08   0.2834
size         0.14963    0.06466     2.31   0.0224 *
sizesq      -0.00634    0.00321    -1.98   0.0503 .
regulatn     0.03872    0.13563     0.29   0.7758
```

We see that both the estimated parameters and the standard errors are very similar to those from the fitted Conway-Maxwell-Poisson model. This suggests that the CMP model may be adequate for the dataset.

We can examine the goodness-of-fit of this semiparametric approach via PIT plots using the in-built function

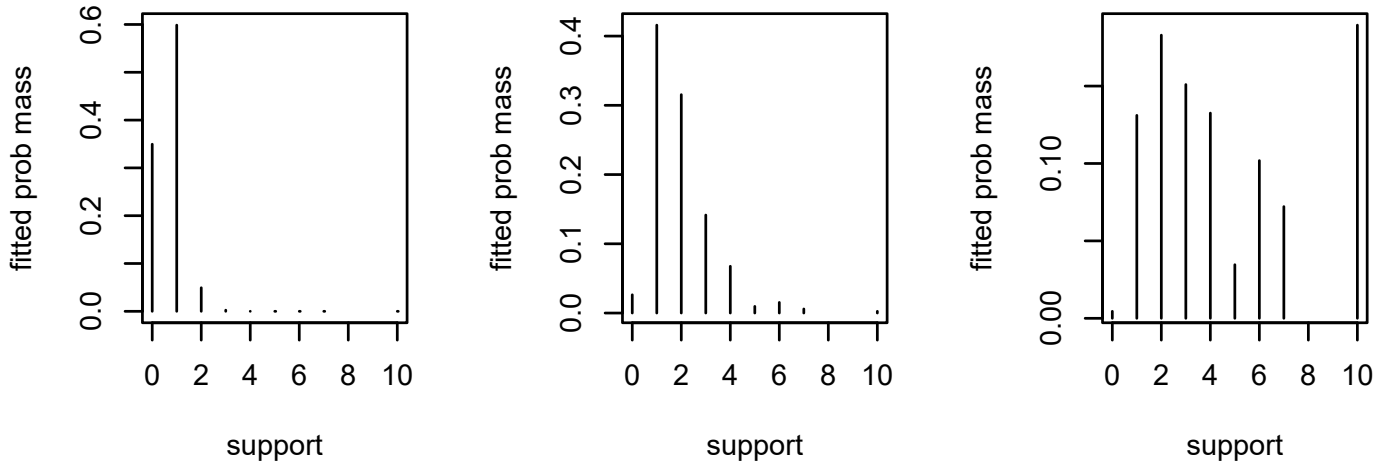
```
gldrmPIT(fit.sp)
```



We see that the estimated nonparametric base measure F has been able to fit to the response distribution as well as the parametrix CMP model. Finally, we can examine the shape of the estimated response distributions at a few observations:

```
fit.sp$mu[54]
[1] 0.7049818
fit.sp$mu[18]
[1] 1.947666
fit.sp$mu[126]
[1] 4.662918
```

```
support = fit.sp$spt
fitted.Fs = fit.sp$fTiltMatrix

par(mfrow=c(1,3))
plot(support, fitted.Fs[54,], type="h", ylab = "fitted prob mass", xlab = "support")
plot(support, fitted.Fs[18,], type="h", ylab = "fitted prob mass", xlab = "support")
plot(support, fitted.Fs[126,], type="h", ylab = "fitted prob mass", xlab = "support")
```



Future work: combine nonparametric mean curves with nonparametric response distribution, i.e., doubly nonparametric regression!