Custom Functions for Specifying Nonlinear Terms to gnm

Heather Turner, David Firth and Andy Batchelor

Department of Statistics
University of Warwick, UK

Motivating Application: Marriage Data

- We wish to investigate the propensity to marry for women living in Ireland based on the Living in Ireland Surveys (1994-2001)
- For women born between 1950 and 1975 we have
  - year of (first) marriage
  - year and month of birth
  - social class
  - highest level of education attained
  - year highest level of education was attained

Proposed Model

- We wish to model the **hazard** of marriage occuring at time t

$$h(t) = P(T = t | T \geq t)$$

using the discrete-time model

$$\begin{aligned} \text{logit}(h(t)) &= c + \beta_l \log(age_{it} - \alpha_l) + \beta_r \log(\alpha_r - age_{it}) + \boldsymbol{x}'_{it}\boldsymbol{\beta} \\ &= \eta \end{aligned}$$

- The parameters can be estimated by logistic regression of a marriage indicator on η — a **generalized nonlinear model**
  - need custom `"nonlin"` function to specify "log-excess" terms in the `formula` argument to `gnm`

Variables and Predictors

- A "nonlin" function creates a list of arguments for the internal function nonlinTerms
- First define the variables and predictors in the term

$$\beta_l \log(age_{it} - \alpha_l)$$

- Start to build "nonlin" function as follows

```
LogExcess <- function(age){
    list(predictors = list(beta = ~1, alpha = ~1),
         variables = list(substitute(age)))
}
class(LogExcess) <- "nonlin"
```

Term-specific Issues

- Want to use same function for both log-excess terms, so add argument

```
side = "left"
```

- To avoid taking logs of negative values, we let

$$\alpha_l = age_{[min]} - 10^{-5} - \exp(\alpha_l^*)$$
$$\alpha_r = age_{[max]} + 10^{-5} + \exp(\alpha_r^*)$$

and estimate α_l^* and α_r^* instead. In `LogExcess` define

```
constraint <- ifelse(side == "left",
                     min(age) - 1e-5, max(age) + 1e-5)
```

Term Definition

- The `term` argument of `nonlinTerms` takes labels for the predictors and variables and returns a deparsed expression of the term:

```
term = function(predLabels, varLabels) {
    paste(predLabels[1], " * log(",
        " -"[side == "right"], varLabels[1], " + ",
        " -"[side == "left"], constraint,
        " + exp(", predLabels[2], "))")
}
```

- So e.g.

```
> term(c("beta", "alpha*"), "age")
[1] "beta  * log(  age  +   - 14.99999  + exp( alpha* ))"
```

Parameter Labels

- Default parameter labels are taken from the `predictor` names, here `beta` and `alpha`
- To make parameter labels unique, save call to `LogExcess`:

  ```
  call <- sys.call()
  ```

  and specify `call` argument to `nonlinTerms`

  ```
  call = as.expression(call)
  ```

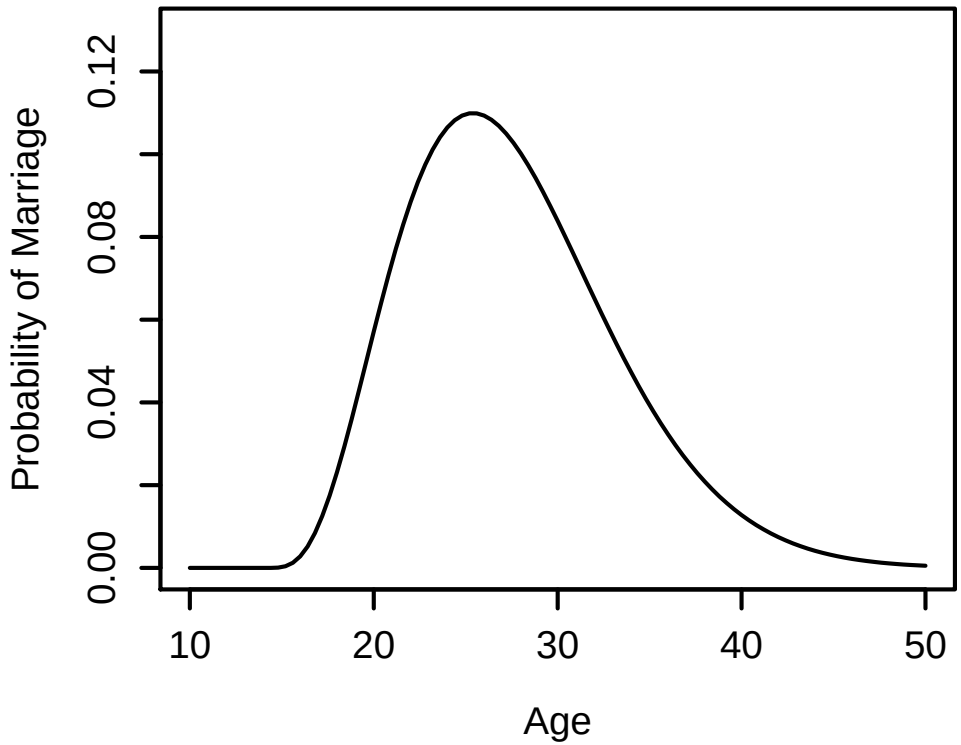
Complete Function

```
LogExcess <- function(age, side = "left"){
    call <- sys.call()
    constraint <- ifelse(side == "left",
                         min(age) - 1e-5, max(age) + 1e-5)
    list(predictors = list(beta = ~1, alpha = ~1),
         variables = list(substitute(age)),
         term = function(predLabels, varLabels) {
             paste(predLabels[1], " * log(",
                   " -"[side == "right"], varLabels[1], " + ",
                   " -"[side == "left"], constraint,
                   " + exp(", predLabels[2], "))")
         },
         call = as.expression(call))
 }
 class(LogExcess) <- "nonlin"
```

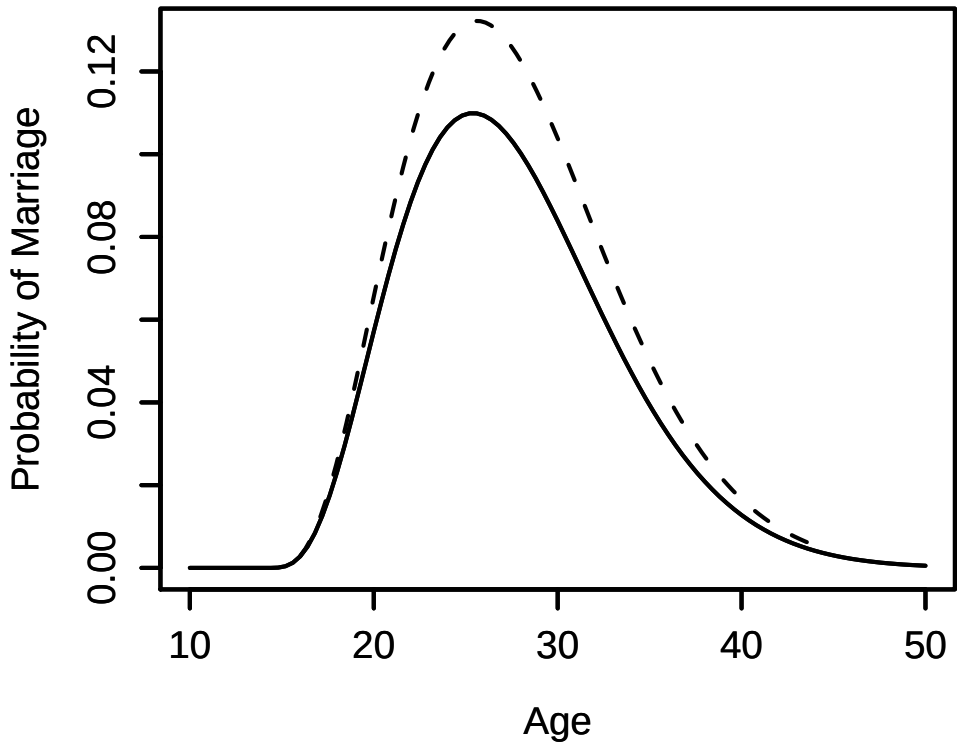
Summary of Baseline Model

```
Call:
gnm(formula = marriages/lives ~ LogExcess(age, side = "left") +
    LogExcess(age, side = "right"), family = binomial, data = fulldata,
    weights = lives, start = c(-20, 3, 0, 3, 0))

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-0.8098  -0.4441  -0.3224  -0.1528   4.0483

                                    Estimate Std. Error z value Pr(>|z|)
(Intercept)                         -118.5395         NA      NA       NA
LogExcess(age, side = "left")beta      3.6928         NA      NA       NA
LogExcess(age, side = "left")alpha    -0.1432         NA      NA       NA
LogExcess(age, side = "right")beta    24.8623         NA      NA       NA
LogExcess(age, side = "right")alpha    4.0247         NA      NA       NA

Std. Error is NA where coefficient has been constrained or is unidentified

Residual deviance: 12553 on 31004 degrees of freedom
AIC: 12748

Number of iterations: 76
```

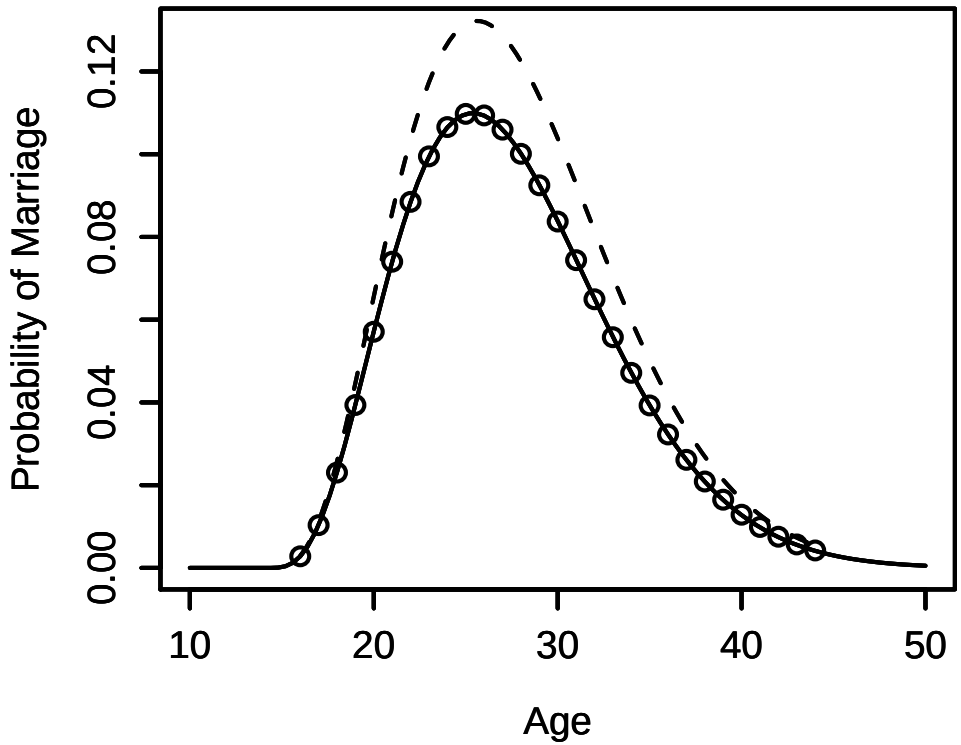
Example 'Recoil' Plot



Example 'Recoil' Plot



Example 'Recoil' Plot



Re-parameterization

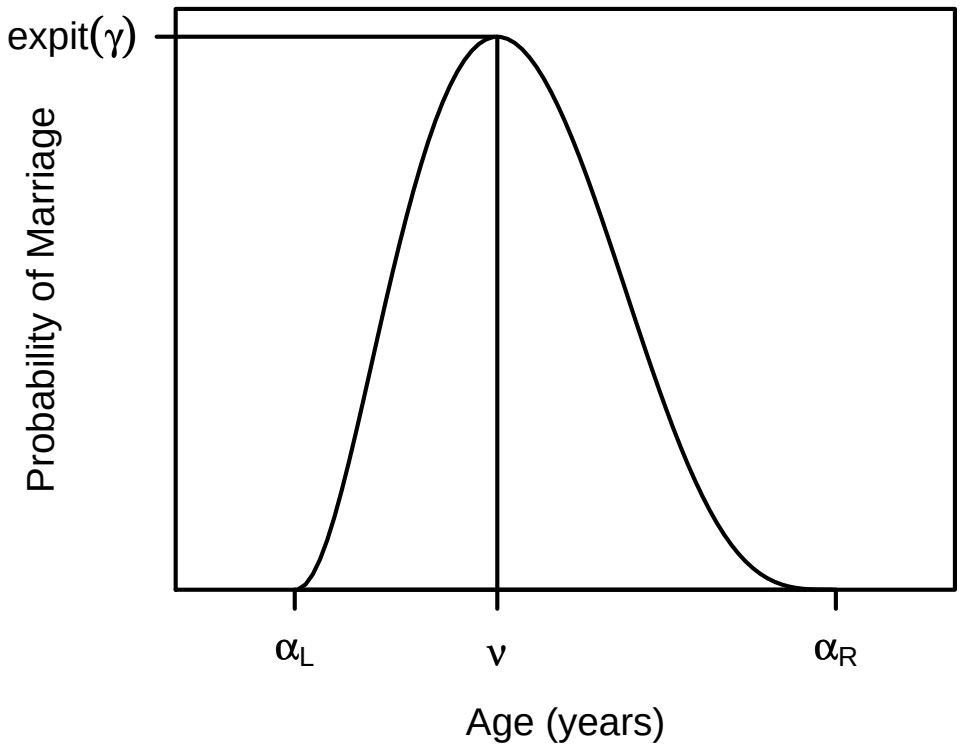
- The problem with aliasing can be overcome by re-parameterizing the baseline model:

$$\gamma - \delta \left\{ (\nu - \alpha_l) \log \left(\frac{\nu - \alpha_l}{age_{it} - \alpha_l} \right) \right\}$$
$$+ \delta \left\{ (\alpha_r - \nu) \log \left(\frac{\alpha_r - \nu}{\alpha_r - age_{it}} \right) \right\}$$

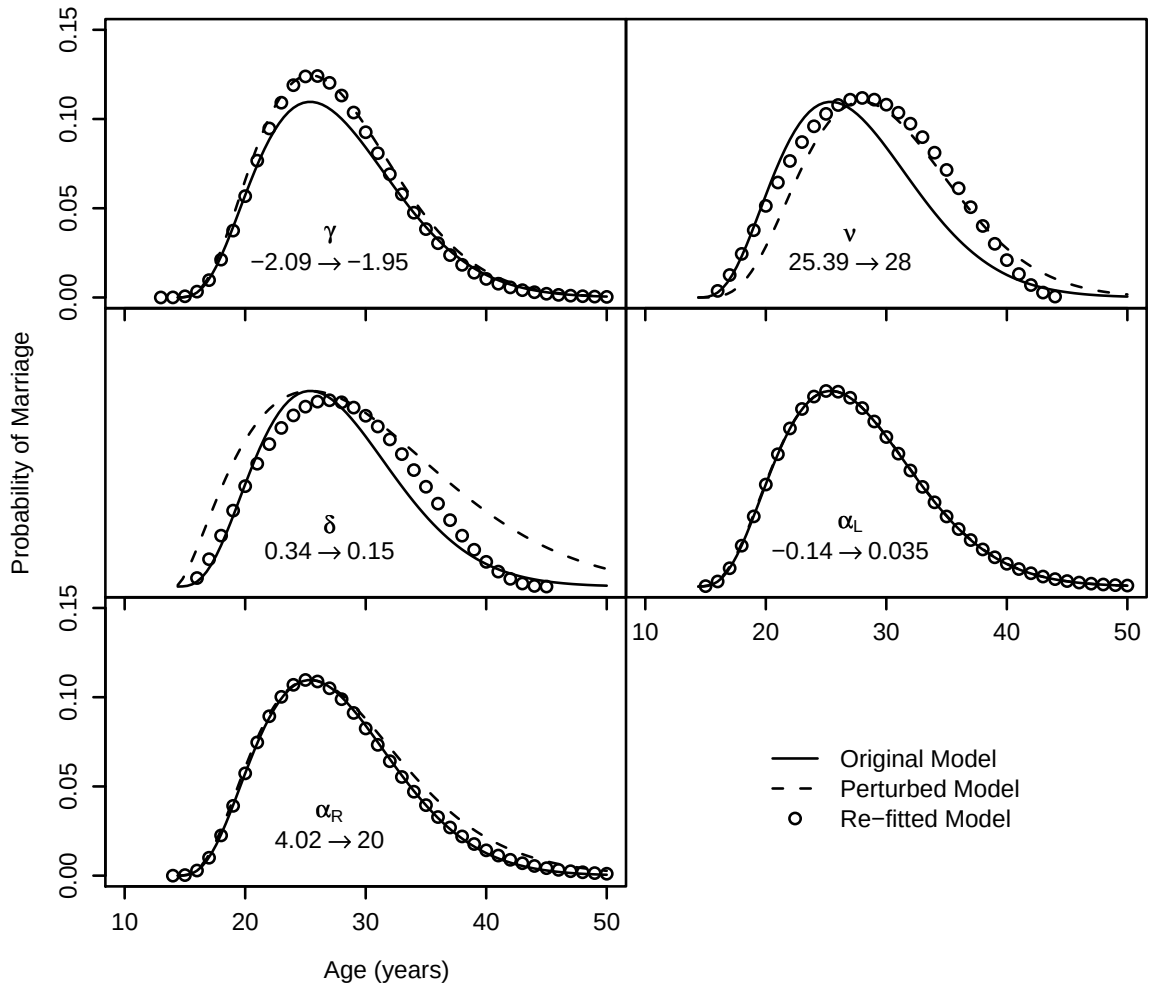
- A new `nonlin` function, `Bell`, is needed to specify this term

Interpretation of Parameters

- The parameters of the new parameterization have a more useful interpretation than before:



Recoil Plots for Reparameterised Model



Summary of Re-parameterized Model

```
Call:
gnm(formula = marriages/lives ~ Bell(age), family = binomial,
    data = fulldata, weights = lives,
    start = c(NA, 26, 0.2, NA, NA))

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-0.8098  -0.4441  -0.3224  -0.1528   4.0483

Coefficients:
                  Estimate Std. Error z value Pr(>|z|)
(Intercept)       -2.09446    0.03313 -63.225  < 2e-16
Bell(age)peak     25.39354    0.30405  83.517  < 2e-16
Bell(age)fallOff   0.32917    0.09065   3.631 0.000282
Bell(age)leftAdj  -0.14321    0.89350  -0.160 0.872663
Bell(age)rightAdj  4.02470    1.73753   2.316 0.020540

(Dispersion parameter for binomial family taken to be 1)

Residual deviance: 12553 on 31004 degrees of freedom
AIC: 12748
```

Further Analysis

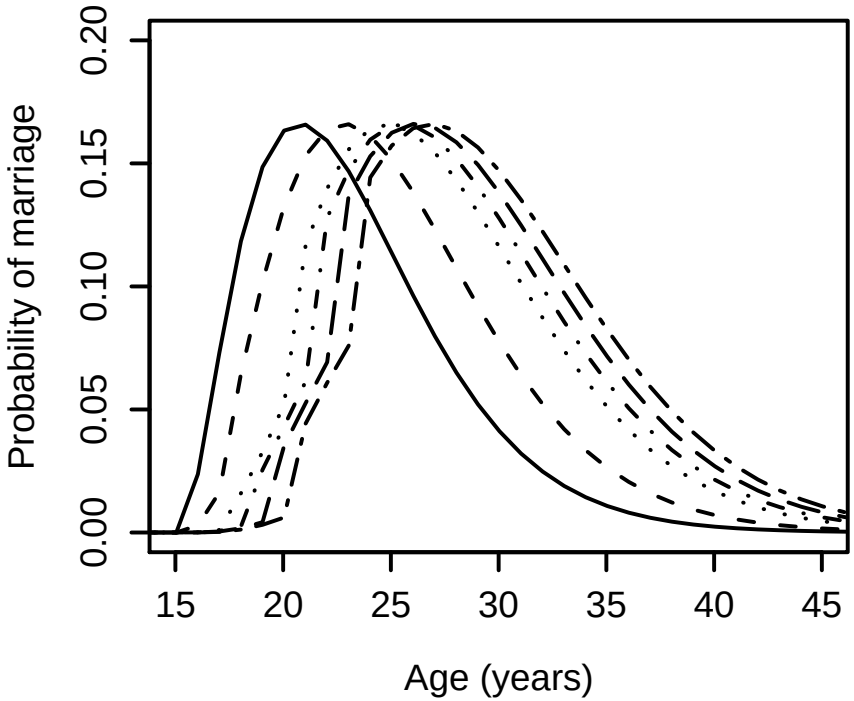
- We can write a simple function to compute the endpoints and their standard errors

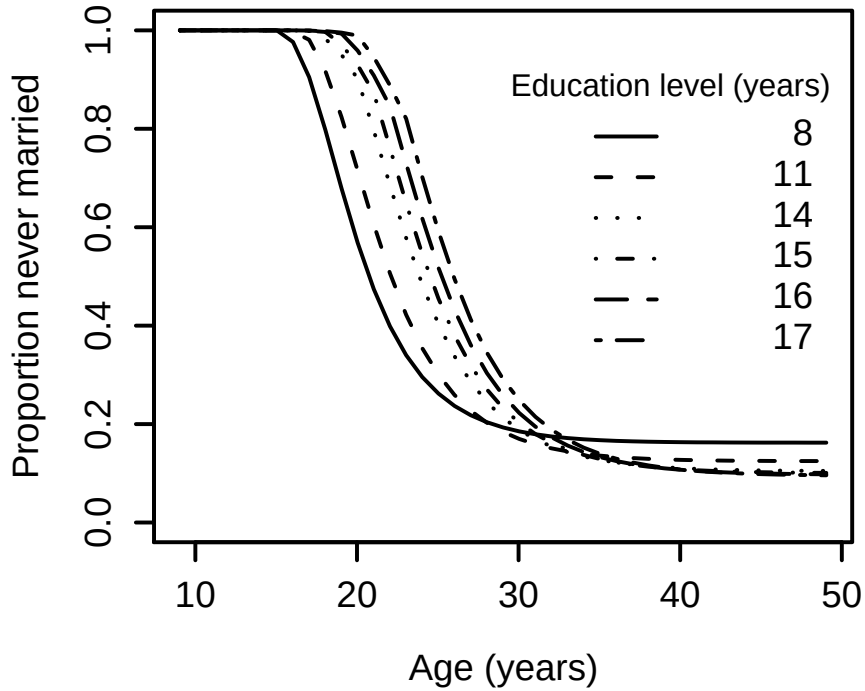
```
> BellEndpoints(bell.mod)
           [,1]       [,2]
left    14.17508  0.7742838
right 100.92183 97.2381849
```

- Adding theoretically important covariates produces even higher estimate of right endpoint
  - use simpler model with infinite right endpoint
- Residual analysis suggests **location** of hazard depends on education level
  - extend model to allow ν to depend on covariates

Final Model

- For women born in 1950





References/Acknowledgements

- More information about gnm can be found on www.warwick.ac.uk/go/gnm
- A working paper on the marriage application is available at www.warwick.ac.uk/go/crism/research/2007
- The marriage data are from The Economic and Social Research Institute Living in Ireland Survey Microdata File (©Economic and Social Research Institute).
- We gratefully acknowledge Carmel Hannan for introducing us to this application and providing background on the data.