gnm: a Package for Generalized Nonlinear Models

Heather Turner and David Firth

Department of Statistics
University of Warwick, UK

Generalized Nonlinear Models

- Same form as generalized linear models:

$$\begin{aligned} g(E(Y)) &= g(\mu) = \eta(x;\beta) \\ \text{Var}(Y) &= f(\mu) \end{aligned}$$

  except $\eta(x;\beta)$ can be nonlinear in β.
- E.g. row-column association model (Goodman, 1979)

$$\log \mu_{rc} = \alpha_r + \beta_c + \gamma_r \delta_c$$

Further Examples

- Diagonal reference (Sobel, 1981)

$$\text{e.g. } \mu_{rc} = w_1\gamma_r + w_2\gamma_c$$

- Stereotype (Anderson, 1984)

$$\text{e.g. } \log \mu_{ic} = \beta_{0c} + \gamma_c(\beta_1 x_{1i} + \beta_2 x_{2i})$$

- UNIDIFF (Erikson & Goldthorpe, 1992; Xie, 1992)

$$\text{e.g. } \log \mu_{ijk} = \alpha_{ik} + \beta_{jk} + \gamma_k \delta_{ij}$$

- GAMMI (van Eeuwijk, 1995)

$$\text{e.g. } \mu_{ij} = \alpha_i + \beta_j + \sigma_1\gamma_{1i}\delta_{1j} + \sigma_2\gamma_{2i}\delta_{2j}$$

The *gnm* Package

- Provides framework for fitting generalized nonlinear models
- Model-fitting function `gnm`
  - in-built mechanism to fit multiplicative terms
  - works with "plug-in" functions to fit other nonlinear terms
- Designed to be `glm`-like
  - common arguments, *gnm* objects inherit from *glm* objects, etc
- Uses over-parameterized representations of models

Model Specification

- *gnm* introduces two functions to specify nonlinear terms
  - `Mult` for standard multiplicative interactions, e.g.

    ```
    counts ~ row + column +
        Mult(-1 + row, -1 + column)
    ```

  - `Nonlin` for other terms that require a “plug-in” function, e.g.

    ```
    counts ~ row + column +
        Nonlin(MultHomog(row, column))
    ```

- Also functions to create factors for structured linear interactions
  - `Diag` (diagonal), `Symm` (symmetric), `Topo` (topological)

Working with Over-Parameterized Models

- `gnm` imposes minimal identifiability constraints
  - the same model can be represented by an infinite number of parameterisations, e.g.

$$\begin{aligned}\log \mu_{rc} &= \alpha_r + \beta_c + \gamma_r \delta_c \\ &= \alpha_r + \beta_c + (2\gamma_r)(0.5\delta_c) \\ &= \alpha_r + \beta_c + \gamma_r^{'} \delta_c^{'}\end{aligned}$$

  - `gnm` will return one of these parameterisations, at random
- Rules for constraining nonlinear parameters not required

Estimating Identifiable Parameter Combinations

- Through `gnm`
  - using arguments *constrain* and *constrainTo*
- From *gnm* object
  - estimate simple contrasts using `getContrasts`
  - estimate linear combinations of parameters using `se`

both `getContrasts` and `se` check estimability first

Example: Occupational Status Data

- Study of occupational status taken from Goodman (1979)
- Cross-classified by occupational status of father: orig[in] and son: dest[ination]

```
> status
    dest
orig   1   2   3   4   5   6   7   8
   1  50  19  26   8   7  11   6   2
   2  16  40  34  18  11  20   8   3
   3  12  35  65  66  35  88  23  21
   4  11  20  58 110  40 183  64  32
   5   2   8  12  23  25  46  28  12
   6  12  28 102 162  90 554 230 177
   7   0   6  19  40  21 158 143  71
   8   0   3  14  32  15 126  91 106
```

Row-Column Association Model

- Model diagonal effects separately and use standard multiplicative interaction

$$\log \mu_{rc} = \alpha_r + \beta_c + \theta_r(r = c) + \gamma_r \delta_c$$

```
> RC <- gnm(Freq ~ orig + dest + Diag(orig, dest) +
+           Mult(-1 + orig, -1 + dest),
+           family = poisson, data = status)
```

- Print summary of model

```
> summary(RC)
```

Row-Column Association Model Summary

```
Call: ...

Deviance Residuals: ...

Coefficients:
                    Estimate Std. Error z value Pr(>|z|)
(Intercept)          0.19649         NA      NA       NA
orig2                0.46730         NA      NA       NA
...
orig8                1.15890         NA      NA       NA
dest2                0.98771         NA      NA       NA
...
dest8                1.67662         NA      NA       NA
Diag(orig, dest)1    1.47923    0.45401   3.258  0.00112 **
...
Diag(orig, dest)8    0.40731    0.21930   1.857  0.06327 .
Mult1.Factor1.orig1  1.80430         NA      NA       NA
...
Mult1.Factor1.orig8 -1.48489         NA      NA       NA
Mult1.Factor2.dest1  1.23943         NA      NA       NA
...
Mult1.Factor2.dest8 -0.82438         NA      NA       NA

...
Residual deviance: 29.149 on 28 degrees of freedom
AIC: 423.49

Number of iterations: 9
```

$\}$ orig main effects

$\}$ dest main effects

$\}$ diagonal effects

$\}$ orig multiplier

$\}$ dest multiplier

Homogeneous Row-Column Association Model

- Compare to model with homogeneous multiplicative interaction

$$\log \mu_{ij} = \alpha_i + \beta_j + \theta_{ij} + \delta_i \delta_j$$

```
> RChomog <- update(RC, . ~ . - Mult(-1 + orig, -1 + dest)
+                   + Nonlin(MultHomog(orig, dest)),
+                   ofInterest = "Mult")
>
> anova(RChomog, RC, test = "Chisq")
Analysis of Deviance Table

Model 1: Freq ~ orig + ... + Nonlin(MultHomog(orig, dest))
Model 2: Freq ~ orig + ... + Mult(orig, dest)
  Resid. Df Resid. Dev Df Deviance P(>|Chi|)
1        34     32.561
2        28     29.149  6    3.412     0.756
```

Homogeneous Model Summary

```
> summary(RChomog)
Call:

gnm(formula = Freq ~ orig + dest + Diag(orig, dest) + Nonlin(MultHomog(orig,
    dest)), ofInterest = "Mult", family = poisson, data = status)


Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.6588  -0.4297   0.0000   0.3862   1.7208

Coefficients of interest:
                        Estimate Std. Error z value Pr(>|z|)
MultHomog(orig, dest).1 -1.50270         NA      NA       NA
MultHomog(orig, dest).2 -1.28440         NA      NA       NA
MultHomog(orig, dest).3 -0.68624         NA      NA       NA
MultHomog(orig, dest).4 -0.10236         NA      NA       NA
MultHomog(orig, dest).5 -0.08519         NA      NA       NA
MultHomog(orig, dest).6  0.42657         NA      NA       NA
MultHomog(orig, dest).7  0.84271         NA      NA       NA
MultHomog(orig, dest).8  1.08629         NA      NA       NA

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

Residual deviance: 32.561 on 34 degrees of freedom
AIC: 414.9

Number of iterations: 7
```

$\}$ coefficients of interest

Using getContrasts

```
> contr <- getContrasts(RChomog, ofInterest(RChomog))
> round(contr[[1]]$qvframe, 3)
                       estimate    SE quasiSE quasiVar
MultHomog(orig, dest).1    0.000 0.000   0.157    0.025
MultHomog(orig, dest).2    0.218 0.235   0.119    0.014
MultHomog(orig, dest).3    0.816 0.167   0.061    0.004
MultHomog(orig, dest).4    1.400 0.160   0.052    0.003
MultHomog(orig, dest).5    1.418 0.172   0.080    0.006
MultHomog(orig, dest).6    1.929 0.157   0.036    0.001
MultHomog(orig, dest).7    2.345 0.173   0.080    0.006
MultHomog(orig, dest).8    2.589 0.189   0.110    0.012
```

Quasi standard errors are independent of the parameterisation - for more detail see Firth and de Menezes (Biometrika, 2004)

More on *gnm* ...

- Package available on CRAN: `http://cran.r-project.org`
- `demo(gnm)` covers example from this talk
- `vignette("gnmOverview", package = "gnm")` covers more examples and details of further features
  - for `Mult` terms: `Exp`, multiplicity
  - other: `Dref`, eliminate, ...
- Vignette and slides from short course available from `http://www.warwick.ac.uk/go/heatherturner/gnm`

Further Work

- Integrate models which currently need work-arounds
  - > 1 instance of homogeneous multiplicative interaction, e.g.

$$\log \mu_{ij} = \alpha_i + \beta_j + \theta_{ij} + \delta_i\delta_j + \theta_i\theta_j$$

  - "sum-of-exponentials" models, e.g.

$$y = \alpha + \exp(\beta_1 + \gamma_1 x) + \exp(\beta_2 + \gamma_2 x) + e$$

- Handle exponentials and reciprocals as standard
  - add `Exp` and `Inv` terms
  - allow nested nonlinear terms?