

Sensitivity analysis of stochastic reserving models using bootstrap simulations

Othman El Hammouchi

June 20, 2024

Overview

The claims reserving problem

The bootstrap method

Mack's model

The ODP model

The claims reserving problem

Insurance industry

- ▶ Inverted production cycle
- ▶ Future liabilities not known today
- ▶ Prudential and regulatory requirement to make provisions

The actuarial reserving problem

- ▶ *Claims reserving*: forecast future funds needed to settle outstanding contracts
- ▶ Not just point estimate, but also variability and shape of distribution
- ▶ Traditional approach based on *claims*, *loss* or *run-off triangles*

Claims triangle example

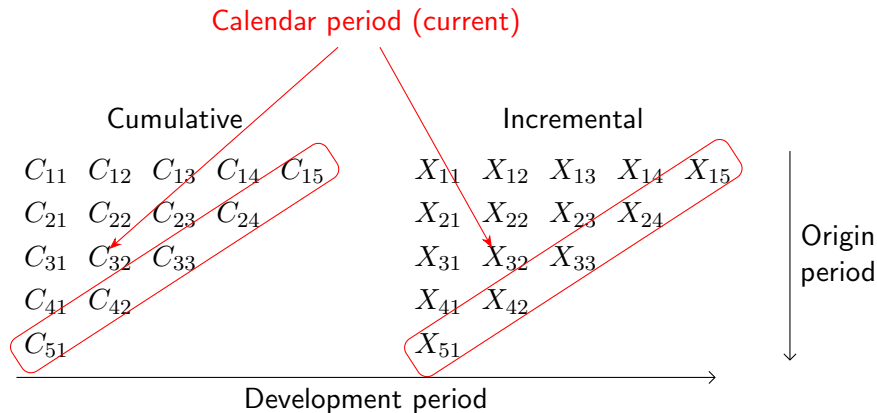
Origin	Dev						
	1	2	3	4	5	6	7
2007	3511	6726	8992	10704	11763	12350	12690
2008	4001	7703	9981	11161	12117	12746	
2009	4355	8287	10233	11755	12993		
2010	4295	7750	9773	11093			
2011	4150	7897	10217				
2012	5102	9650					
2013	6283						

Table: Cumulative payments triangle for a motor insurance account from the UK

Claims triangles in general

Cumulative					Incremental					Origin period ↓
C_{11}	C_{12}	C_{13}	C_{14}	C_{15}	X_{11}	X_{12}	X_{13}	X_{14}	X_{15}	
C_{21}	C_{22}	C_{23}	C_{24}		X_{21}	X_{22}	X_{23}	X_{24}		
C_{31}	C_{32}	C_{33}			X_{31}	X_{32}	X_{33}			
C_{41}	C_{42}				X_{41}	X_{42}				
C_{51}					X_{51}					
Development period →										

Claims triangles in general



Forecasting using claims triangles

C_{11} C_{12} C_{13} C_{14} C_{15}

C_{21} C_{22} C_{23} C_{24} \hat{C}_{25}

C_{31} C_{32} C_{33} \hat{C}_{34} \hat{C}_{35}

C_{41} C_{42} \hat{C}_{43} \hat{C}_{44} \hat{C}_{45}

C_{51} \hat{C}_{52} \hat{C}_{53} \hat{C}_{54} \hat{C}_{55}

X_{11} X_{12} X_{13} X_{14} X_{15}

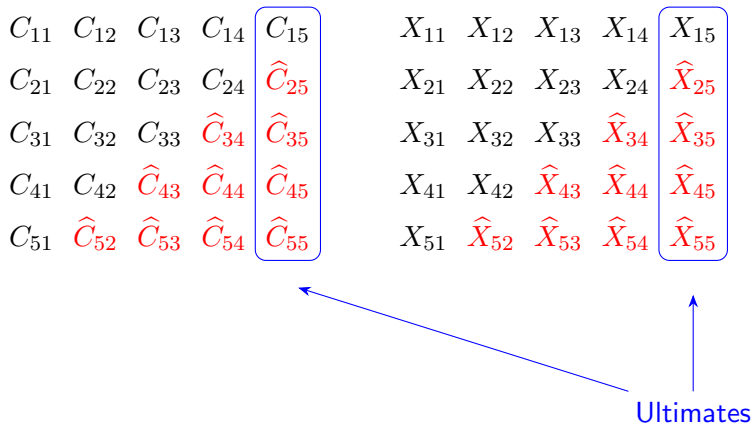
X_{21} X_{22} X_{23} X_{24} \hat{X}_{25}

X_{31} X_{32} X_{33} \hat{X}_{34} \hat{X}_{35}

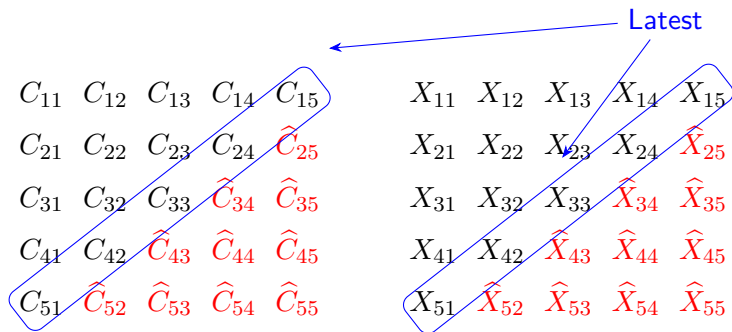
X_{41} X_{42} \hat{X}_{43} \hat{X}_{44} \hat{X}_{45}

X_{51} \hat{X}_{52} \hat{X}_{53} \hat{X}_{54} \hat{X}_{55}

Forecasting using claims triangles



Forecasting using claims triangles



More nomenclature

- ▶ I origin periods, J development periods
- ▶ We assume $I = J$ (square triangles)
- ▶ Reserve $R = \sum_{i=2}^I (C_{i,I} - C_{i,I+1-i}) = \sum_{j=2}^I \sum_{i=I+2-j}^I X_{ij}$

The chain ladder

- ▶ Most popular reserving method ¹
- ▶ Originally deterministic algorithm
- ▶ Various attempts to frame it as a stochastic model
- ▶ Main assumption: there exist *development factors* f_1, \dots, f_{I-1} such that

$$\mathbb{E} [C_{ij} | C_{i,j-1}, \dots, C_{i1}] = f_{j-1} C_{i,j-1}$$

¹According to the ASTIN 2016 Non-Life Reserving Practices Report 


Visualisation of the chain ladder

C_{11} C_{12} C_{13} C_{14} C_{15}

C_{21} C_{22} C_{23} C_{24}

C_{31} C_{32} C_{33}

C_{41} C_{42}

C_{51}

 \hat{f}_1 \hat{f}_2 \hat{f}_3 \hat{f}_4

Column sum average

$$\hat{f}_j = \frac{\sum_{i=1}^{I-j} C_{i,j+1}}{\sum_{i=1}^{I-j} C_{ij}}$$

Chain ladder prediction

$$\hat{C}_{ij} = C_{i,I+1-i} \prod_{k=I+1-i}^{j-1} \hat{f}_k$$

Visualisation of the chain ladder

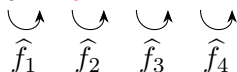
C_{11} C_{12} C_{13} C_{14} C_{15}

C_{21} C_{22} C_{23} C_{24}

C_{31} C_{32} C_{33}

C_{41} C_{42}

C_{51} \hat{C}_{52}



Column sum average

$$\hat{f}_j = \frac{\sum_{i=1}^{I-j} C_{i,j+1}}{\sum_{i=1}^{I-j} C_{ij}}$$

Chain ladder prediction

$$\hat{C}_{ij} = C_{i,I+1-i} \prod_{k=I+1-i}^{j-1} \hat{f}_k$$

Visualisation of the chain ladder

C_{11} C_{12} C_{13} C_{14} C_{15}

C_{21} C_{22} C_{23} C_{24}

C_{31} C_{32} C_{33}

C_{41} C_{42} \hat{C}_{43}

C_{51} \hat{C}_{52} \hat{C}_{53}

\hat{f}_1 \hat{f}_2 \hat{f}_3 \hat{f}_4

Column sum average

$$\hat{f}_j = \frac{\sum_{i=1}^{I-j} C_{i,j+1}}{\sum_{i=1}^{I-j} C_{ij}}$$

Chain ladder prediction

$$\hat{C}_{ij} = C_{i,I+1-i} \prod_{k=I+1-i}^{j-1} \hat{f}_k$$


Visualisation of the chain ladder

C_{11} C_{12} C_{13} C_{14} C_{15}

C_{21} C_{22} C_{23} C_{24}

C_{31} C_{32} C_{33} \hat{C}_{34}

C_{41} C_{42} \hat{C}_{43} \hat{C}_{44}

C_{51} \hat{C}_{52} \hat{C}_{53} \hat{C}_{54}

 \hat{f}_1 \hat{f}_2 \hat{f}_3 \hat{f}_4

Column sum average

$$\hat{f}_j = \frac{\sum_{i=1}^{I-j} C_{i,j+1}}{\sum_{i=1}^{I-j} C_{ij}}$$

Chain ladder prediction

$$\hat{C}_{ij} = C_{i,I+1-i} \prod_{k=I+1-i}^{j-1} \hat{f}_k$$

Visualisation of the chain ladder

C_{11}	C_{12}	C_{13}	C_{14}	C_{15}
C_{21}	C_{22}	C_{23}	C_{24}	\hat{C}_{25}
C_{31}	C_{32}	C_{33}	\hat{C}_{34}	\hat{C}_{35}
C_{41}	C_{42}	\hat{C}_{43}	\hat{C}_{44}	\hat{C}_{45}
C_{51}	\hat{C}_{52}	\hat{C}_{53}	\hat{C}_{54}	\hat{C}_{55}
\curvearrowright	\curvearrowright	\curvearrowright	\curvearrowright	
\hat{f}_1	\hat{f}_2	\hat{f}_3	\hat{f}_4	

Column sum average

$$\hat{f}_j = \frac{\sum_{i=1}^{I-j} C_{i,j+1}}{\sum_{i=1}^{I-j} C_{ij}}$$

Chain ladder prediction

$$\hat{C}_{ij} = C_{i,I+1-i} \prod_{k=I+1-i}^{j-1} \hat{f}_k$$

Stochastic chain ladder

- ▶ Many different variants
- ▶ Reproduce chain ladder point estimates
- ▶ Make different assumptions
- ▶ Difficult to verify with small data sizes
- ▶ Idea: detect violations by excluding points and gauging effect on bootstrapped reserve

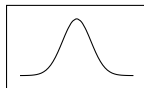
Normal

Data



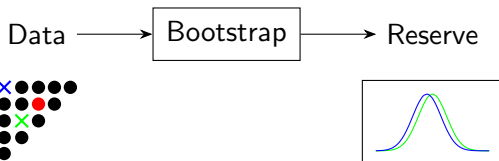
Bootstrap

Reserve

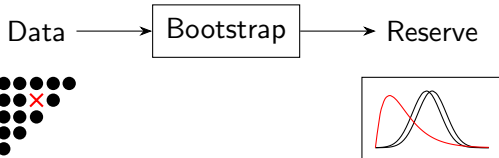


Detecting assumption violations

Wrong point



Right point

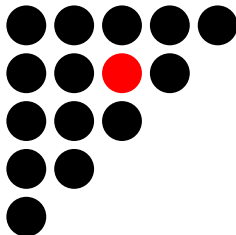


Detecting assumption violations

- ▶ Generate triangles which follow assumptions perfectly
- ▶ Apply perturbation
- ▶ Remove one point at a time and study impact on reserve
- ▶ Significant impact \Rightarrow reverse-engineer

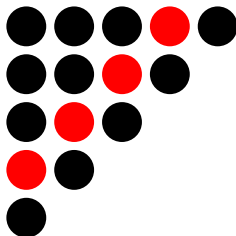
Perturbations

Single observation



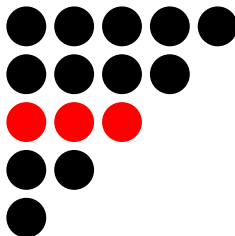
Perturbations

Calendar period



Perturbations

Origin period



The bootstrap method

Main idea

- ▶ Classical inference often intractable
- ▶ Relies on approximations and asymptotics
- ▶ Solution: resampling to produce pseudo-replicates

Main idea

- ▶ Classical inference often intractable
- ▶ Relies on approximations and asymptotics
- ▶ Solution: resampling to produce pseudo-replicates



Classical estimator

- ▶ Independent identically distributed sample X_1, \dots, X_n
- ▶ Parameter θ estimated by $\hat{\theta} := g(X_1, \dots, X_n)$
- ▶ For $b = 1, \dots, B$
 - ▶ Resample to obtain $X_1^{(b)}, \dots, X_n^{(b)}$
 - ▶ Compute $\hat{\theta}^{(b)} := g(X_1^{(b)}, \dots, X_n^{(b)})$
 - ▶ $\{\hat{\theta}^{(b)} \mid b = 1, \dots, B\}$ used for inference, e.g. variance estimation:

$$\widehat{\text{Var}}(\theta) := \frac{1}{B-1} \sum_{b=1}^B (\hat{\theta}^{(b)} - \bar{\theta}^B)^2$$

$$\text{with } \bar{\theta}^B := \frac{1}{B} \sum_{b=1}^B \hat{\theta}^{(b)}$$

Parametric vs. nonparametric

- ▶ How to do bootstrap resampling?
- ▶ Nonparametric: resample with replacement directly from data
- ▶ Parametric: fit model first, use this to simulate from RNG
- ▶ Can be extended to regression models

Regression

- ▶ Covariates X_1, \dots, X_p and response Y
- ▶ Parametrised function $f(X_1, \dots, X_p; \boldsymbol{\beta})$ modelling their relation
- ▶ Classic example: linear regression
 - ▶ $Y_i = \mathbf{x}_i^T \boldsymbol{\beta} + \varepsilon_i$
 - ▶ $\mathbb{E}[\varepsilon_i] = 0, \text{Var}(\varepsilon_i) = \sigma^2$
 - ▶ $\mathbb{E}[\varepsilon_i \varepsilon_j] = 0$
 - ▶ LS estimator: $\hat{\boldsymbol{\beta}} := (X^T X)^{-1} X^T \mathbf{y}$

Regression bootstrap

- ▶ Independent sample of predictor-response pairs $(\mathbf{x}_1, Y_1), \dots, (\mathbf{x}_n, Y_n)$
- ▶ For $b = 1, \dots, B$
 - ▶ Resample to obtain $(\mathbf{x}_1^{(b)}, Y_1^{(b)}), \dots, (\mathbf{x}_n^{(b)}, Y_n^{(b)})$
 - ▶ Compute $\hat{\beta}^{(b)}$
 - ▶ $\{\hat{\beta}^{(b)} \mid b = 1, \dots, B\}$ used for inference
- ▶ Parametric vs. nonparametric?

Nonparametric regression bootstrap

- ▶ Fundamental unit of resampling?
- ▶ Residuals \Rightarrow semiparametric
- ▶ Pairs \Rightarrow fully nonparametric

Semiparametric regression bootstrap

- ▶ Resample residuals, e.g.

$$r_i := Y_i - \mathbf{x}_i^T \hat{\boldsymbol{\beta}}$$

- ▶ Produce bootstrap replicates via

$$Y_i^{(b)} := \mathbf{x}_i^T \hat{\boldsymbol{\beta}} + r_i^{(b)}$$

- ▶ Only relies on parametrisation of first two moments

Fully nonparametric regression bootstrap

- ▶ Resample pairs to produce $(\mathbf{x}_1^{(b)}, Y_1^{(b)}), \dots, (\mathbf{x}_n^{(b)}, Y_n^{(b)})$
- ▶ Approximates multivariate distribution of (X_1, \dots, X_n, Y)
- ▶ Model refitted to pseudo-replicates to obtain $\hat{\beta}^{(b)}$
- ▶ Does not assume anything about data (except i.i.d.-ness of sample)

Parametric regression bootstrap

- ▶ Additional assumption about distribution of the ε_i
- ▶ Classic choice: $\varepsilon_i \sim \mathcal{N}(0, \sigma^2)$
- ▶ Fit model to obtain $\hat{\beta}$ and $\hat{\sigma}$
- ▶ Produce $Y_i^{(b)}$ by drawing from the estimated distribution $\mathcal{N}(\mathbf{x}_i^T \hat{\beta}, \hat{\sigma}^2)$
- ▶ Relies on correct specification of parametric model

Process error

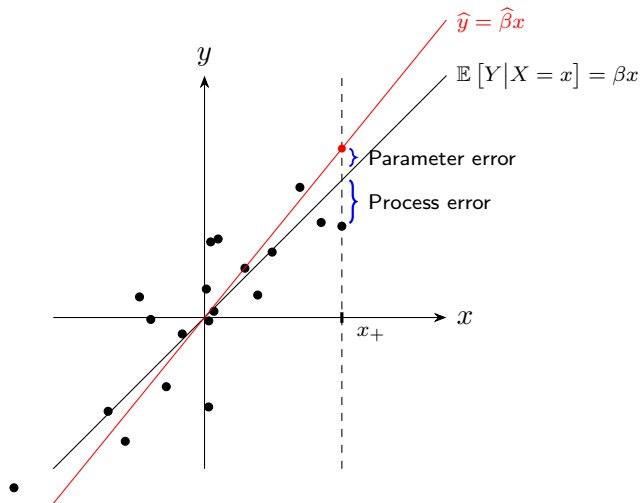
- ▶ Methods mentioned so far produce replicates of parameter vector β
- ▶ Can be used to simulate the *fitted* or *predicted response*

$$\hat{Y}_+^{(b)} = \mathbf{x}_+^T \hat{\beta}^{(b)}$$

at new value \mathbf{x}_+ of the regressors

- ▶ Incorporates estimation uncertainty or *parameter error*
- ▶ What about simulating Y_+ itself?
- ▶ Incorporate intrinsic variation or *process error*

Process error visualised



Predictive distribution

- ▶ Bayesian concept
- ▶ Incorporates both parameter and process error
- ▶ Semiparametric: resample residuals a second time and compute

$$Y_+^{(b,s)} := \mathbf{x}_+^T \hat{\boldsymbol{\beta}}^{(b)} + r^{(s)}$$

- ▶ Parametric: generate pseudo-response according to

$$Y_+^{(b,s)} \sim \mathcal{N}(\mathbf{x}_+^T \hat{\boldsymbol{\beta}}^{(b)}, \hat{\sigma}^2)$$

- ▶ Nonparametric: borrow one of the other approaches

Mack's model

Formulation

Model 1 (Mack chain ladder)

1. *There exist development factors f_1, \dots, f_{I-1} such that*

$$\mathbb{E}[C_{ij} \mid C_{i,j-1}, \dots, C_{i1}] = \mathbb{E}[C_{ij} \mid C_{i,j-1}] = f_{j-1} C_{i,j-1}$$

for $1 \leq i \leq I$

2. *There exist variance parameters $\sigma_1, \dots, \sigma_{I-1}$ such that*

$$\text{Var}[C_{ij} \mid C_{i,j-1}, \dots, C_{i1}] = \text{Var}[C_{ij} \mid C_{i,j-1}] = \sigma_{j-1}^2 C_{i,j-1}$$

for $1 \leq i \leq I$

3. *The cumulative claims processes $(C_{ij})_j, (C_{i'j})_j$ are independent for $i \neq i'$*

Properties

- ▶ Cumulative triangle
- ▶ Distribution-free
- ▶ Recursive
- ▶ For any pair of consecutive columns: equivalent to

$$\mathbf{c}_{j+1} = f_j \mathbf{c}_j + \boldsymbol{\varepsilon}$$

with

$$\mathbb{E} [\boldsymbol{\varepsilon} | C_{1,j}, \dots, C_{I-j,j}] = \mathbf{0}$$

$$\text{Var} [\boldsymbol{\varepsilon} | C_{1,j}, \dots, C_{I-j,j}] = \sigma_j^2 \begin{bmatrix} C_{1j} & & \\ & \ddots & \\ & & C_{I-j,j} \end{bmatrix}$$

Properties

- ▶ Model assumptions correspond to Gauss-Markov
- ▶ Optimal estimator: weighted least squares with

$$\mathbf{W} = \begin{bmatrix} 1/C_{1j} & & \\ & \ddots & \\ & & 1/C_{I-j,j} \end{bmatrix}$$

- ▶ Same as column sum estimator!

$$\hat{f}_j^{\text{WLS}} = (\mathbf{c}_j^T \mathbf{W} \mathbf{c}_j)^{-1} \mathbf{c}_j^T \mathbf{W} = \frac{\sum_{i=1}^{I-j} C_{i,j+1}}{\sum_{i=1}^{I-j} C_{i,j}}$$

- ▶ We can adapt the regression bootstrap!

Conditional vs. unconditional

- ▶ Recursivity leads to different bootstrap types
- ▶ Simulate next development year based on original data vs. generated bootstrap replicate
- ▶ Parametric example:

$$C_{i,j+1}^{(b)} \sim \mathcal{N}(\hat{f}_j C_{ij}, \hat{\sigma}_j^2) \quad \text{vs.} \quad C_{i,j+1}^{(b)} \sim \mathcal{N}(\hat{f}_j C_{ij}^{(b)}, \hat{\sigma}_j^2)$$

Wealth of configurations

- ▶ Conditional vs. unconditional
- ▶ Nonparametric: only conditional is possible!
- ▶ Parametric: which distribution?
 - ▶ Normal
 - ▶ Gamma
- ▶ Semiparametric: which residuals?
 - ▶ Standardised
 - ▶ Studentised
 - ▶ Log-normal
- ▶ Computationally very intensive!

Implementation

- ▶ R package {trngl}
- ▶ Front-end in R
- ▶ Heavy-duty numerical code in Fortran
- ▶ Parallelised using OpenMP
- ▶ Glued together with Rcpp
- ▶ Available on Github

The ODP model

Formulation

Model 2 (overdispersed Poisson GLM)

1. *The incremental claims are independent from each other*
2. *There exist parameters c, a_1, \dots, a_I and b_1, \dots, b_I such that*

$$\log(\mu_{ij}) = c + a_i + b_j$$

with $\mu_{ij} := \mathbb{E}[X_{ij}]$ and $a_1 = b_1 = 0$

3. *There exists a parameter ϕ such that*

$$\text{Var}[X_{ij}] = \phi \mu_{ij}$$

Properties

- ▶ Incremental triangle
- ▶ Belongs to family of *generalised linear models*
 - ▶ Extend normal linear model
 - ▶ Response can follow any distribution from the EDM family
 - ▶ Covariates related to response via *link function*
- ▶ Dispersion parameter allowing mean to differ from variance (cfr. Poisson)
- ▶ Fitted using quasi-maximum likelihood
- ▶ Equations solved iteratively using Fisher scoring

Triangle to regression

- ▶ Flatten triangle to obtain regression model
- ▶ Development and origin year become the covariates
- ▶ Cfr. two-way ANOVA (without interaction)
- ▶ We can adapt the regression bootstrap!

Triangle to regression

- ▶ Flatten triangle to obtain regression model
- ▶ Development and origin year become the covariates
- ▶ Cfr. two-way ANOVA (without interaction)
- ▶ We can adapt the regression bootstrap!

Origin	Dev	Value
2007	1	3511
2008	1	4001
2009	1	4355
2010	1	4295
2011	1	4150
2012	1	5102
2013	1	6283
2007	2	3215
2008	2	3702
2009	2	3932
2010	2	3455
2011	2	3747
2012	2	4548
2007	3	2266
2008	3	2278

Triangle to regression

$$\begin{array}{cccccc}
 c & & b_2 & b_3 & b_4 & b_5 \\
 \swarrow & & \uparrow & \uparrow & \uparrow & \uparrow \\
 & X_{11} & X_{12} & X_{13} & X_{14} & X_{15} \\
 a_2 \leftarrow & X_{21} & X_{22} & X_{23} & X_{24} & \\
 a_3 \leftarrow & X_{31} & X_{32} & X_{33} & & \\
 a_4 \leftarrow & X_{41} & X_{42} & & & \\
 a_5 \leftarrow & X_{51} & & & &
 \end{array}$$

$$\begin{bmatrix}
 1 & 0 & 0 & \dots & 0 \\
 1 & 1 & 0 & \dots & 0 \\
 1 & 0 & 1 & \dots & 0 \\
 \vdots & \vdots & \vdots & \ddots & \vdots \\
 1 & 0 & 0 & \dots & 1
 \end{bmatrix}
 \begin{bmatrix}
 c \\
 a_2 \\
 a_3 \\
 \vdots \\
 b_5
 \end{bmatrix}
 =
 \begin{bmatrix}
 \log(\mu_{11}) \\
 \log(\mu_{21}) \\
 \log(\mu_{31}) \\
 \vdots \\
 \log(\mu_{15})
 \end{bmatrix}$$

Configurations

- ▶ Parametric: which distribution?
 - ▶ Normal
 - ▶ Gamma
 - ▶ Poisson
- ▶ Semiparametric: which residuals?
 - ▶ Most popular ones for GLM: Pearson and deviance
 - ▶ Deviance suffer technical shortcoming which inhibits resampling
- ▶ Nonparametric: impossible
- ▶ Computationally very intensive!

Implementation

- ▶ R package {trngl}
- ▶ Front-end in R
- ▶ Heavy-duty numerical code in Fortran
- ▶ Parallelised using OpenMP
- ▶ Glued together with Rcpp
- ▶ Available on Github

And now for a demo!