A general multilevel estimation framework: Multivariate joint models and more

Statistical Analysis of Multi-Outcome Data
University of Liverpool
3rd July 2017

Michael J. Crowther

Biostatistics Research Group,
Department of Health Sciences,
University of Leicester, UK,
michael.crowther@le.ac.uk
@Crowther_MJ

Outline

My background

- I was awarded my PhD in November 2014, titled "Development and application of methodology for the parametric analysis of complex survival and joint longitudinal-survival data in biomedical research"

My background

- I was awarded my PhD in November 2014, titled "Development and application of methodology for the parametric analysis of complex survival and joint longitudinal-survival data in biomedical research"
- I was a post-doctoral biostatistician at Karolinska Institutet, 1st February 2015 - 29th February 2016, working mainly on parametric multi-state survival models

My background

- I was awarded my PhD in November 2014, titled "Development and application of methodology for the parametric analysis of complex survival and joint longitudinal-survival data in biomedical research"
- I was a post-doctoral biostatistician at Karolinska Institutet, 1st February 2015 - 29th February 2016, working mainly on parametric multi-state survival models
- I'm now a Lecturer in Biostatistics at the University of Leicester, where my work focuses on:
  - Joint modelling of longitudinal and survival data
  - Multi-state survival models
  - Survival analysis methods for analysis of electronic health records, mainly in CVD and cancer

Outline

Background 1

- Given the current trend in improved availability in both access to data, and volume of data, there is the ever increasing need for efficient, and appropriate statistical modelling techniques, and implementations

Background 1

- Given the current trend in improved availability in both access to data, and volume of data, there is the ever increasing need for efficient, and appropriate statistical modelling techniques, and implementations
- Consider the EHR, we inevitably have a complex hierarchical structure to the data, such as multiple biomarkers measured repeatedly $<$ patients $<$ GP practice area $<$ geographical regions, and so on

Background 1

- Given the current trend in improved availability in both access to data, and volume of data, there is the ever increasing need for efficient, and appropriate statistical modelling techniques, and implementations
- Consider the EHR, we inevitably have a complex hierarchical structure to the data, such as multiple biomarkers measured repeatedly $<$ patients $<$ GP practice area $<$ geographical regions, and so on
- Further challenges include time-dependent effects, and non-linear covariate effects, both of which are arguably commonplace in many settings

Background 1

- Given the current trend in improved availability in both access to data, and volume of data, there is the ever increasing need for efficient, and appropriate statistical modelling techniques, and implementations
- Consider the EHR, we inevitably have a complex hierarchical structure to the data, such as multiple biomarkers measured repeatedly $<$ patients $<$ GP practice area $<$ geographical regions, and so on
- Further challenges include time-dependent effects, and non-linear covariate effects, both of which are arguably commonplace in many settings
- Therefore, the need for appropriate modelling frameworks which can accommodate such complex structures is paramount

Background 2

- Joint longitudinal-survival models (JLSMs) [1]

Background 2

- Joint longitudinal-survival models (JLSMs) [1]
- A model is specified for each outcome, with some form of sharing between outcome models, often done through shared or correlated random effects

Background 2

- Joint longitudinal-survival models (JLSMs) [1]
- A model is specified for each outcome, with some form of sharing between outcome models, often done through shared or correlated random effects
- Commonplace in JLSMs is linking the 'current value' of the longitudinal outcome, directly to survival, through its expected value conditional on subject-specific random effects

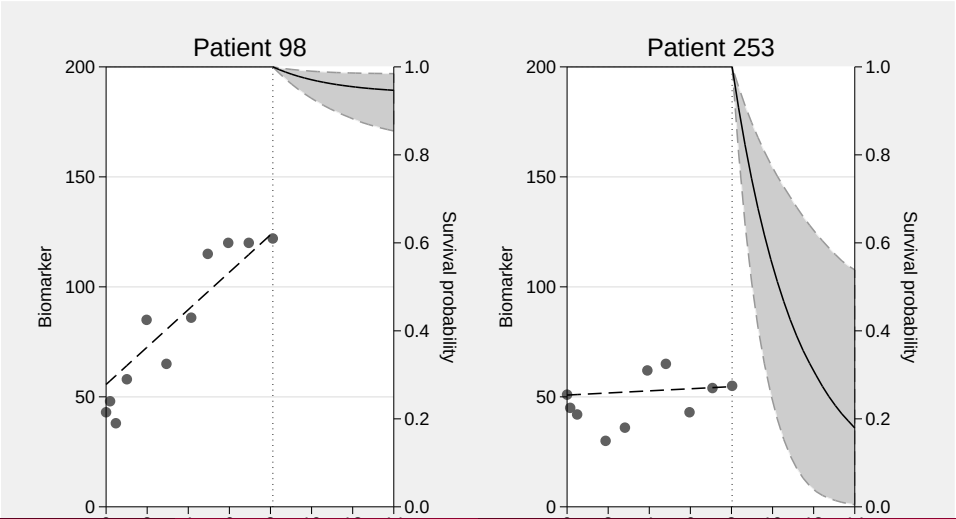
Background 2

- Joint longitudinal-survival models (JLSMs) [1]
- A model is specified for each outcome, with some form of sharing between outcome models, often done through shared or correlated random effects
- Commonplace in JLSMs is linking the 'current value' of the longitudinal outcome, directly to survival, through its expected value conditional on subject-specific random effects
- Alternatives include transformations of the current value, e.g. its gradient, or its integral

Background 2

- Joint longitudinal-survival models (JLSMs) [1]
- A model is specified for each outcome, with some form of sharing between outcome models, often done through shared or correlated random effects
- Commonplace in JLSMs is linking the 'current value' of the longitudinal outcome, directly to survival, through its expected value conditional on subject-specific random effects
- Alternatives include transformations of the current value, e.g. its gradient, or its integral
- These are clinically plausible ways to link such outcomes in many settings, and give us interpretable association parameters, irrespective of how complex the longitudinal model specification may be (such as when using splines).

This is where we want to get to



Numerous extensions

There has been a tremendous amount of work in this area

- Competing risks [2]
- Different types of outcomes [3]
- Multiple continuous outcomes [4]
- Delayed entry [5]
- Recurrent events and a terminal event [6]
- Prediction [7]
- Many others...

Software

We are always limited by availability of user-friendly software

- frailtypack in R [8]
- stjm in Stata [9]
- joineR in R [10]
- JM and JMBayes in R [11, 12]
- Many others...

My aim

My aim in this work is to provide a general framework for the analysis of clustered data, which can encompass;

- Multiple outcomes of varying types
- Measurement schedule can vary across outcomes
- Any number of levels
- Any number of random effects at each level
- Sharing and linking random effects between outcomes
- Sharing functions of the expected value of other outcomes
- Useful predictions
- A reliable estimation engine
- Easily extendable by the user
- Much more...

Outline

A general level model [13]

For example, for a one-level model with n response variables:

$$p(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{b}, \boldsymbol{\beta}) = \prod_{i=1}^{n} p_i(y_i|\boldsymbol{x}, \boldsymbol{b}, \boldsymbol{\beta})$$

For a two-level model:

$$p(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{b}, \boldsymbol{\beta}) = \prod_{i=1}^{n} \prod_{j=1}^{t} p_i(y_{ij}|\boldsymbol{x}, \boldsymbol{b}, \boldsymbol{\beta})$$

A general level model [13]

The log likelihood is obtained by integrating out the unobserved random effects, to obtain a marginal log likelihood,

$$ll(\boldsymbol{\beta}) = \log \int_{\mathcal{R}^r} p(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{b}, \boldsymbol{\beta}) \phi(\boldsymbol{b}|\Sigma_{\boldsymbol{b}}) \, \mathrm{d}\boldsymbol{b} \tag{1}$$

where $\mathcal{R}^r$ is the r-dimensional space, with each dimension spanning the real number line, and r the dimension of the random effects $\boldsymbol{b}$. We assume $\phi()$ is multivariate normal density for $\boldsymbol{b}$, with mean vector $\mathbf{0}$ and variance-covariance matrix $\Sigma_{\boldsymbol{b}}$. Equation (1) can be expanded with further levels of nesting, with $\Sigma_{\boldsymbol{b}}$ becoming block diagonal, with a block for each level. I'll refer to this as $ll1$.

Alternatively, exploiting conditional independence amongst level $l-1$ units, given the random effects at higher levels,

$$ll(\boldsymbol{\beta}) = \log \int \phi(\boldsymbol{b}^{(L)}|\boldsymbol{\Sigma}^{(L)}) \prod p^{(L-1)}(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{b}^{L}, \boldsymbol{\beta}) \, \mathrm{d}\boldsymbol{b}^{(L)}$$

where, for $l = 2, \ldots, L$

$$p^{(l)}(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{B}^{l+1}, \boldsymbol{\beta}) = \int \phi(\boldsymbol{b}^{(l)}|\boldsymbol{\Sigma}^{(l)}) \prod p^{(l-1)}(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{B}^{l}, \boldsymbol{\beta}) \, \mathrm{d}\boldsymbol{b}^{(l)}$$

I'll refer to this as $ll2$

Estimation challenges

- At each level, we need to integrate out our normally distributed random effects
- Generally this is done using Gauss-Hermite numerical quadrature

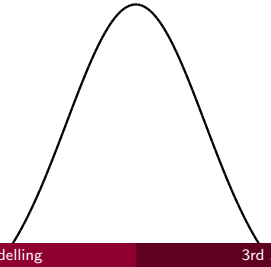
Gauss-Hermite quadrature

- Numerical method to approximate analytically intractable integrals [14]

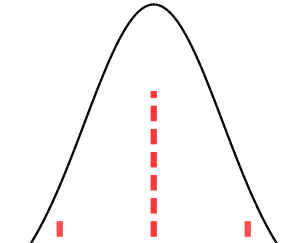
$$\int_{-\infty}^{\infty} e^{-x^2} f(x) dx \approx \sum_{q=1}^{m} w_q f(x_q)$$

- Can be extended to multivariate integrals i.e. multiple random effects

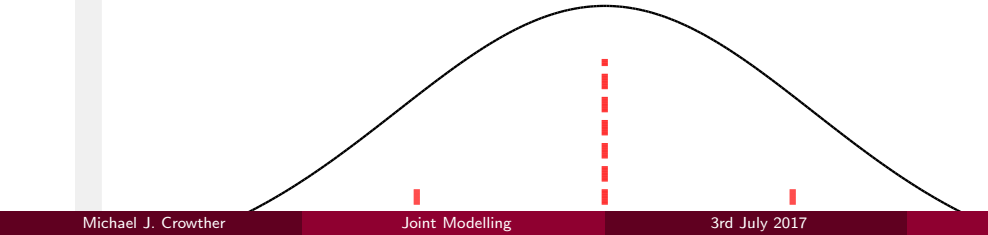
$u \sim N(0,1)$

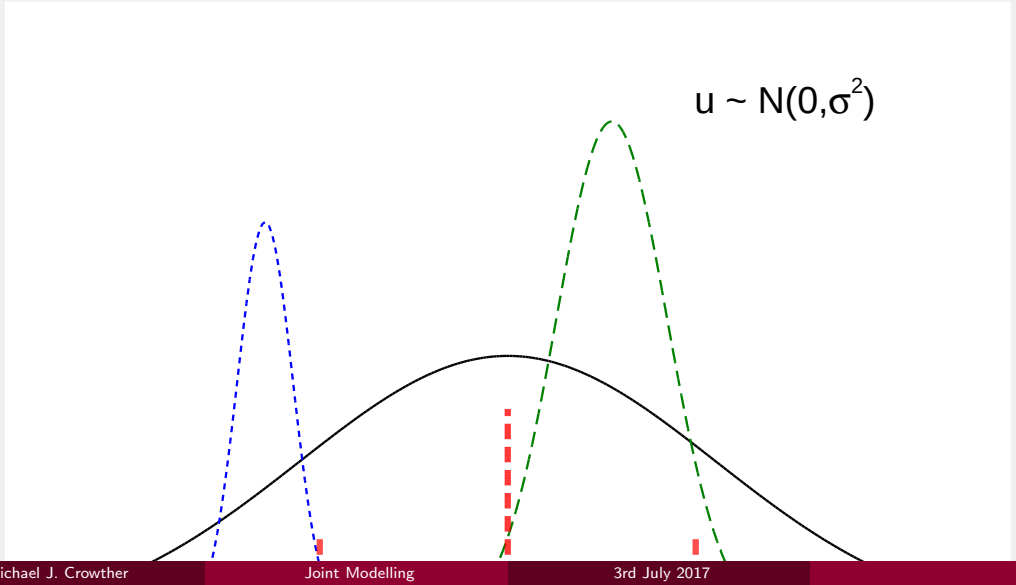


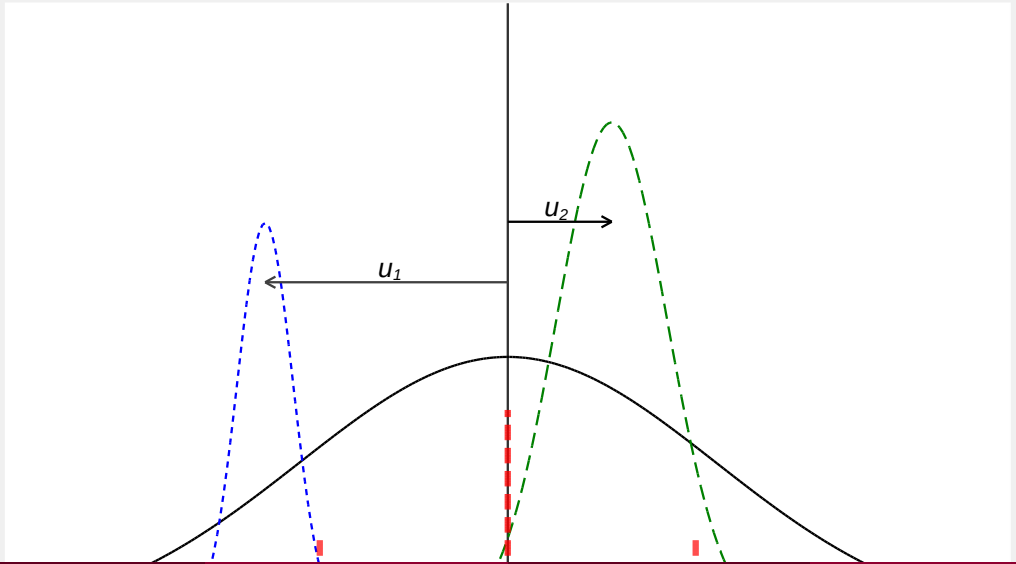
$u \sim N(0,1)$

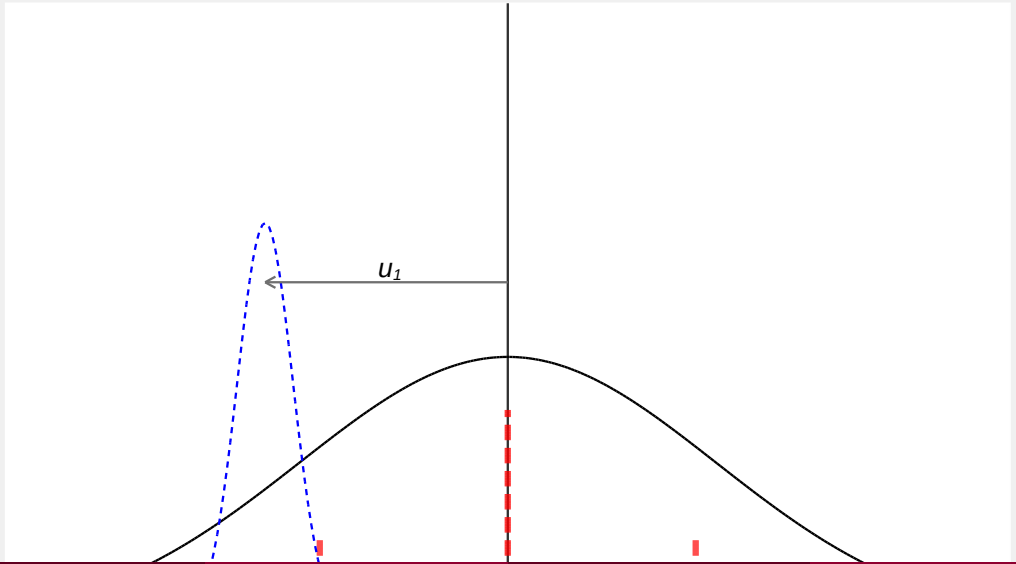


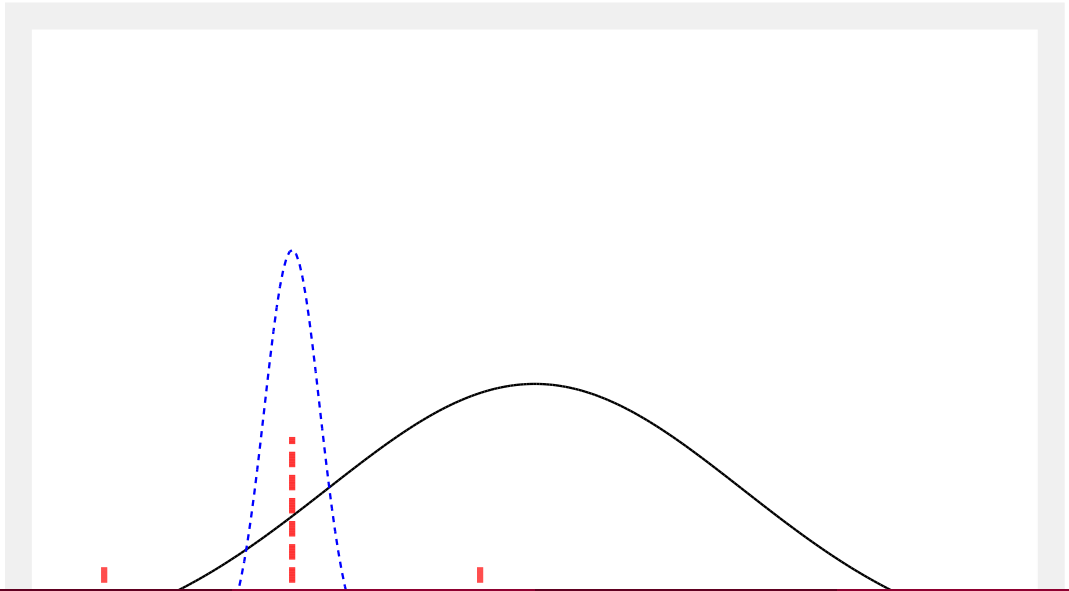
$u \sim N(0,\sigma^2)$

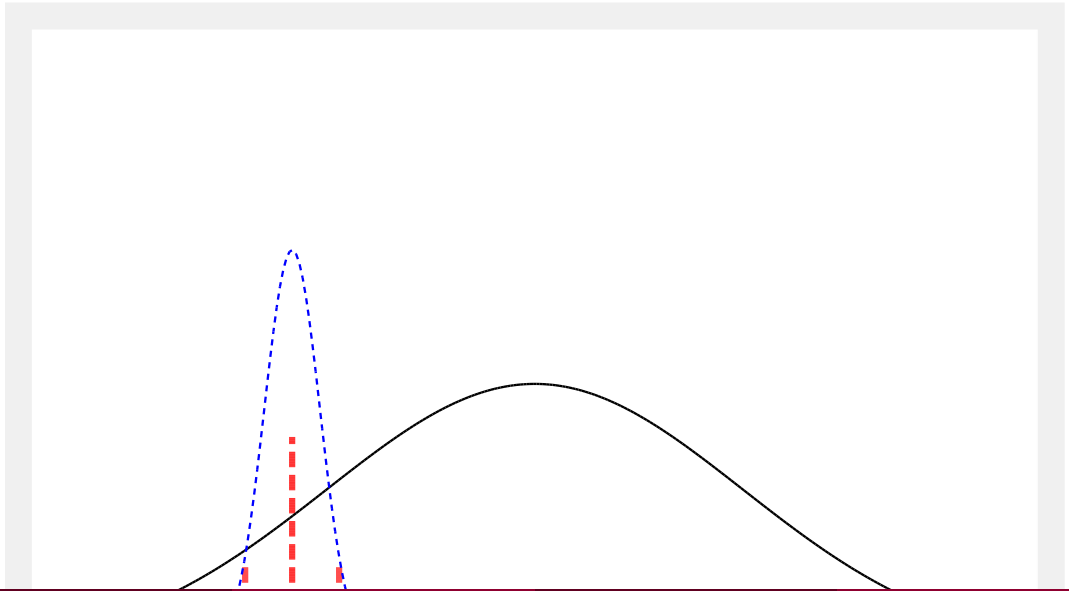



u ~ N(0,σ²)


u_2
u_1


u_1





The approximation is crucial

- When fitting models which rely on numerical approximations, the actual performance of the approximation is widely ignored
- Partly I think this is due to a lack of awareness of what is going on behind the scenes
- Within the context of joint modelling, we did some simulations a few years ago comparing non-adaptive and adaptive GH quadrature, showing you need at least 30 non-adaptive points to get close to the performance of 5-point adaptive [15]

Alternatives

- An issue with GH quadrature is it doesn't scale up well, for example, say we conduct 7-point quadrature, well for 1 random effect we evaluate our function at 7-points
- Say we have 3 biomarkers, each with a random intercept and linear slope, then for 6 random effects, we evaluate it at $7^6 = 117,649$ points
- An alternative is Monte Carlo integration

Monte Carlo integration

This is a rather brute force approach, but it's usefulness is in it's simplicity

$$L(\boldsymbol{\theta}) = \int f(\boldsymbol{y}|\boldsymbol{\theta}, \boldsymbol{b})\phi(\boldsymbol{b})\partial\boldsymbol{b} = \frac{1}{m}\sum_{u=1}^{m} f(y|\theta, \boldsymbol{b}_u)$$

The important thing to note is m doesn't have to change when extra random effects are added. It can be improved by:

- antithetic sampling [16]
- Halton sequences
- an adaptive procedure just like AGHQ, resulting in an importance sampling approximation.

Level-specific integration techniques and random effect distributions

The methods described above all assume the same distributional family for the random effects, across all levels of a model. Returning to $ll2$, our nested marginal likelihood, we can easily relax this,

$$ll(\boldsymbol{\theta}) = \log \int \phi_L(\boldsymbol{b}^{(L)}|\boldsymbol{\Sigma}^{(L)}) \prod p^{(L-1)}(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{b}^L, \boldsymbol{\beta}) \, \mathrm{d}\boldsymbol{b}^{(L)}$$

where, for $l = 2, \ldots, L$

$$p^{(l)}(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{B}^{l+1}, \boldsymbol{\beta}) = \int \phi_l(\boldsymbol{b}^{(l)})|\boldsymbol{\Sigma}^{(l)} \prod p^{(l-1)}(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{B}^l, \boldsymbol{\beta}) \, \mathrm{d}\boldsymbol{b}^{(l)}$$

where $\phi_l(\boldsymbol{b}^{(l)}|\boldsymbol{\Sigma}^{(l)})$ for $l = 2, \ldots, L$ is now level-specific

Level-specific integration techniques and random effect distributions

- This formulation now allows us to specify different distributions at each level
- Assess robustness using the t-distribution
- Issue of which integration techniques to apply at each level
  - e.g. one random effect at level 3, many at level 2, then use AGHQ at level 3, and MCI at level 2

Linear predictor

The standard linear predictor for a general level model can be written as follows,

$$\eta = \boldsymbol{X}\boldsymbol{\beta} + \sum_{l=2}^{L} \boldsymbol{X}^l \boldsymbol{b}^l$$

where subscripts are omitted for ease of exposition. We have $\boldsymbol{X}$ our vector of covariates, which could vary at any level, with associated fixed effect coefficient vector $\boldsymbol{\beta}$, and $\boldsymbol{X}^l$ the vector of covariates with random effects $\boldsymbol{b}^l$ at level l.

Extended linear predictor

$$\eta_i = g_i(E[y_i|\boldsymbol{X}, \boldsymbol{b}]) = \sum_{r=1}^{R_i} \prod_{s=1}^{S_{ir}} \psi_{irs}$$

where $g_i()$ is the link function for the ith outcome. To maintain generality, $\psi_{irs}(t)$ can take many forms, including,

$$\begin{aligned}
\psi_{irs}(t) &= X \\
\psi_{irs}(t) &= \beta \\
\psi_{irs}(t) &= b \\
\psi_{irs}(t) &= q(t) \\
\psi_{irs}(t) &= d_{rs}(E[y_j]), \qquad \text{where } j = 1, \ldots, k, j \neq i
\end{aligned}$$

megenreg in Stata

- Everything I've talked about will be available in the megenreg package in Stata
- It is a simplified/modified version of Stata's official gsem, which itself is ridiculously powerful, and was based on gllamm [13]
- megenreg will have many extensions, such as
  - Alternative models, such as spline based survival models
  - Extending sharing between outcomes, motivated by joint modelling
  - User-defined likelihood functions
  - Other things...

megenreg in Stata

Distributional choices

- Gaussian
- Poisson
- Binomial
- Beta
- negative binomial
- exponential, Weibull, Gompertz, log-normal, log-logistic, gamma, Royston-Parmar
- Non-linear outcome models
- User-defined hazard functions
- More to add...

Outline

1. A general level parametric survival model

The Royston-Parmar survival model uses restricted cubic splines of log time, on the log cumulative hazard scale, i.e.,

$$\log H(y) = s(\log(y)|\beta_{\boldsymbol{k}}) + \eta$$

```
. list patient time infect age female in 1/4, noobs
```

patient	time	infect	age	female
1	8	1	28	0
1	16	1	28	0
2	13	0	48	1
2	23	1	48	1

```
. megenreg (time age female M1[patient], family(rp, failure(infect) scale(h) df(3)))
```

1. A general level parametric survival model

Relax the normally dist. random effects assumption;

```
. megenreg (time age female M1[patient], family(rp, failure(infect) scale(h) df(3)))   ///
>                 , redistribution(t) df(3)
```

1. A general level parametric survival model

Relax the normally dist. random effects assumption;

```
. megenreg (time age female M1[patient], family(rp, failure(infect) scale(h) df(3)))   ///
>                 , redistribution(t) df(3)
```

Higher levels of clustering;

```
. megenreg (time trt M1[trial] M2[trial>patient], ...)
```

1. A general level parametric survival model

Relax the normally dist. random effects assumption;

```
. megenreg (time age female M1[patient], family(rp, failure(infect) scale(h) df(3)))   ///
>                 , redistribution(t) df(3)
```

Higher levels of clustering;

```
. megenreg (time trt M1[trial] M2[trial>patient], ...)
```

Random coefficients;

```
. megenreg (time trt M1[trial] trt#M1[trial] M2[trial>patient], ... )
```

1. A general level parametric survival model

Relax the normally dist. random effects assumption;

```
. megenreg (time age female M1[patient], family(rp, failure(infect) scale(h) df(3)))   ///
>                 , redistribution(t) df(3)
```

Higher levels of clustering;

```
. megenreg (time trt M1[trial] M2[trial>patient], ...)
```

Random coefficients;

```
. megenreg (time trt M1[trial] trt#M1[trial] M2[trial>patient], ... )
```

Time-dependent effects;

```
. megenreg (stime trt trt#{log(&t)} M1[id1] M2[id1>id2], ... timevar(stime))
```

1. A general level parametric survival model

Relax the normally dist. random effects assumption;

```
. megenreg (time age female M1[patient], family(rp, failure(infect) scale(h) df(3)))   ///
>                 , redistribution(t) df(3)
```

Higher levels of clustering;

```
. megenreg (time trt M1[trial] M2[trial>patient], ...)
```

Random coefficients;

```
. megenreg (time trt M1[trial] trt#M1[trial] M2[trial>patient], ... )
```

Time-dependent effects;

```
. megenreg (stime trt trt#{log(&t)} M1[id1] M2[id1>id2], ... timevar(stime))
```

Non-linear covariate effects

```
. gen age2 = age^2
. megenreg (stime trt trt#{log(&t)} age age2 M1[id1] M2[id1>id2], ... )
```

2. A general level relative survival model

Relative survival models are used widely, particularly in population based cancer epidemiology [17]. They model the excess mortality in a population with a particular disease, compared to a reference population. They do not rely on accurate cause of death information.

$$h(y) = h^*(y) + \lambda(y)$$

where $h^*(y)$ is the expected mortality in the reference population. Any of the previous models can be turned into a relative survival model;

```
. megenreg (stime trt trt#log(&t) M1[id1] M2[id1>id2],  ///
>                   family(rp, failure(died) df(3) scale(h) bhazard(bhaz)))
```

3. General level joint frailty survival models

- An area of intense research in recent years is in the field of joint frailty survival models, for the analysis of joint recurrent event and terminal event data
- Here I focus on the two most popular approaches, proposed by Liu et al. (2004) [18] and Mazroui et al. (2012) [19]
- In both, we have a survival model for the recurrent event process, and a survival model for the terminal event process, linked through shared random effects

3. General level joint frailty survival models

$$h_{ij}(y) = h_0(y)\exp(\boldsymbol{X}_{1ij}\boldsymbol{\beta}_1 + b_i)$$
$$\lambda_i(y) = \lambda_0(y)\exp(\boldsymbol{X}_{1i}\boldsymbol{\beta}_2 + \alpha b_i)$$

where $h_{ij}(y)$ is the hazard function for the jth event of the ith patient, $\lambda_i(y)$ is the hazard function for the terminal event, and $b_i \sim N(0, \sigma^2)$. We can fit such a model with `megenreg`, adjusting for treatment in each outcome model,

```
. megenreg (rectime trt M1[id1]       , family(rp, failure(recevent) scale(h) df(5)))  ///
>          (stime   trt M1[id1]@alpha , family(rp, failure(died)     scale(h) df(3)))
```

3. General level joint frailty survival models

$$h_{ij}(y) = h_0(y)\exp(\boldsymbol{X}_{1ij}\boldsymbol{\beta}_1 + b_{1i} + b_{2i})$$
$$\lambda_i(y) = \lambda_0(y)\exp(\boldsymbol{X}_{1i}\boldsymbol{\beta}_2 + b_{2i})$$

where $b_{1i} \sim N(0, \sigma_1^2)$ and $b_{2i} \sim N(0, \sigma_2^2)$. We give an example of how to fit this model with `megenreg`, this time illustrating how to use different distributions for the recurrent event and terminal event processes,

```
. megenreg (rectime trt M1[id1] M2[id1] , family(weibull, failure(recevent)))   ///
>          (stime   trt M2[id1]         , family(rp, failure(died) scale(h) df(3)))
```

4. Generalised multivariate JLSMs

Multiple longitudinal biomarkers

$$Y_1 \sim Weib(\lambda, \gamma), \quad Y_2 \sim N(\mu_2, \sigma_2^2), \quad Y_3 \sim N(\mu_3, \sigma_3^2)$$

The linear predictor of the survival outcome can be written as follows,

$$\eta_1(t) = \boldsymbol{X}\boldsymbol{\beta}_0 + E[y_2(t)|\eta_2(t)]\beta_1 + E[y_3(t)|\eta_3(t)]\beta_2 + E[y_2(t)|\eta_2(t)] \times E[y_3(t)|\eta_3(t)]\beta_3$$

```
. megenreg (stime trt EV[logb]@beta1 EV[logp]@beta2 EV[logb]#EV[logp]@beta3 ,
>                                    family(weibull, failure(died)))
>          (logb  {&t}@l1 {&t}#M2[id] M1[id] , family(gaussian) timevar(time))
>          (logp  {&t}@l2 {&t}#M4[id] M3[id] , family(gaussian) timevar(time))
>          , covariance(unstructured)
```

4. Generalised multivariate JLSMs

Competing risks

```
. list id logb logp time trt stime diedpbc diedother if id==3, noobs
```

id	logb	logp	time	trt	stime	diedpbc	diedother
3	.3364722	2.484907	0	D-penicil	2.77078	1	0
3	.0953102	2.484907	.481875	D-penicil	.	.	.
3	.4054651	2.484907	.996605	D-penicil	.	.	.
3	.5877866	2.587764	2.03428	D-penicil	.	.	.

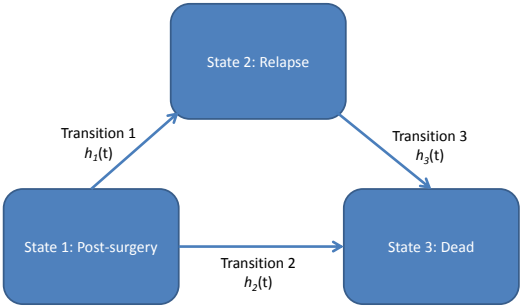
```
. megenreg (stime trt EV[logb]@a1 EV[logp]@a2 , family(weibull, failure(diedpbc)))
>          (stime trt EV[logb]@a3 EV[logp]@a4 , family(gompertz, failure(diedother)))
>          (logb  {&t}@l1 {&t}#M2[id] M1[id]  , family(gaussian) timevar(time))
>          (logp  {&t}@l2 {&t}#M4[id] M3[id]  , family(gaussian) timevar(time))
```

4. Generalised multivariate JLSMs

Joint frailty - The extensive `frailtypack` in R has recently been extended to fit a joint model of a continuous biomarker, a recurrent event process, and a terminal event [6, 8]. We can use `megenreg`,

```
. megenreg (canctime trt EV[logb]@a1 EV[logp]@a2 M5[id]       , ... )
>          (stime    trt EV[logb]@a4 EV[logp]@a5 M5[id]@alpha , ... )
>          (logb  {&t}@l1 {&t}#M2[id] M1[id]                  , ... )
>          (logp  {&t}@l2 {&t}#M4[id] M3[id]                  , ... )
```

4. Generalised multivariate JLSMs



5. A user-defined model

A Gaussian response model

$$y \sim N(\eta, \sigma^2)$$

```
real matrix gauss_logl(transmorphic gml)
{
     y        = gml_util_depvar(gml)          //dep. var.
     linpred  = gml_util_xzb(gml)             //lin. pred.
     sdre     = exp(gml_util_xb(gml,1))       //anc. param.
     return(lnnormalden(y,linpred,sdre))     //logl
}

. megenreg (logb time time#M2[id] M1[id], family(user, loglf(gauss_logl)) np(1))
```

6. A NLME example with multiple linear predictors

Consider Murawska et al. (2012), they developed a Bayesian NL joint model, with Gaussian response variable, and multiple non-linear predictors each with fixed effects and a random intercept. The overall non-linear predictor is defined as,

$$f(t) = \beta_{1i} + \beta_{2i} \exp^{-\beta_{3i} t}$$

where each linear predictor was constrained to be positive,

$$\begin{aligned}
\beta_{1i} &= \exp(X_1 \beta_1 + b_{1i}) \\
\beta_{2i} &= \exp(X_2 \beta_2 + b_{2i}) \\
\beta_{3i} &= \exp(X_3 \beta_3 + b_{3i})
\end{aligned}$$

and for the survival outcome

$$\lambda(t) = \lambda_0(t) \exp(\alpha_1 \beta_{1i} + \alpha_2 \beta_{2i} + \alpha_3 \beta_{3i})$$

6. A NLME example with multiple linear predictors

We can fit this, and extend it, easily with `megenreg`

```
. megenreg (resp age female M1[id], family(user, loglf(nlme_logl)) np(1) timevar(time))
>          (age female M2[id], family(null))
>          (age female M3[id], family(null))
>          (stime age female EV[resp]@alpha1 EV[2]@alpha2 EV[3]@alpha3,
>                       family(weibull, failure(died))),
>          covariance(unstructured)
```

```
real matrix nlme_logl(transmorphic gml, real matrix t)
{
     y         = gml_util_depvar(gml)                      //dep.var.
     linpred1  = exp(gml_util_xzb(gml))                    //main lin. pred.
     linpred2  = exp(gml_util_xzb2(gml,2))                 //extra lin. preds
     linpred3  = exp(gml_util_xzb2(gml,3))
     sdre      = exp(gml_util_xb(gml,1))                   //anc. param
     linpred   = linpred1 :+ linpred2:*exp(-linpred3:*t)  //nonlin. pred
     return(lnnormalden(y,linpred,sdre))                   //logl
}
```

7. Mixed effects for the level 1 variance function

A recent paper by Goldstein et al. (2017) [20] proposed a two-level model with complex level 1 variation, of the form,

$$
\begin{aligned}
y_{ij} &= \boldsymbol{X}_{1ij}\boldsymbol{\beta}_1 + \boldsymbol{Z}_{1ij}\boldsymbol{b}_{1j} + \epsilon_{ij} \\
\epsilon_{ij} &\sim N(0, \sigma_e^2) \\
\log(\sigma_e^2) &= \boldsymbol{X}_{2ij}\boldsymbol{\beta}_2 + \boldsymbol{Z}_{2ij}\boldsymbol{b}_{2j} \\
\begin{pmatrix} \boldsymbol{b}_{1j} \\ \boldsymbol{b}_{2j} \end{pmatrix} &\sim N\left[\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} \boldsymbol{\Sigma}_{b_1} & \\ \boldsymbol{\Sigma}_{b_1 b_2} & \boldsymbol{\Sigma}_{b_2} \end{pmatrix} \right]
\end{aligned}
$$

7. Mixed effects for the level 1 variance function

We can fit this, and extend it, easily with `megenreg`

```
real matrix lev1_logl(transmorphic gml, real matrix t)
{
     y        = gml_util_depvar(gml)                 //response
     linpred1 = gml_util_xzb(gml)                    //lin. pred.
     varresid = exp(gml_util_xzb2(gml,2))            //lev1 lin. pred
     return(lnnormalden(y,linpred,sqrt(varresid)))  //logl
}

. megenreg (resp female age age#M2[id] M1[id], family(user, loglf(lev1_logl)))
           (age female M3[id], family(null))
           covariance(unstructured)
```

Outline

Statistical software development

- I've written a few packages during and since my PhD, including:
  - `stjm` - joint modelling
  - `stmixed` - multilevel parametric survival
  - `survsim` - simulation of survival data
  - `multistate` - multi-state survival models

Statistical software development

- I've written a few packages during and since my PhD, including:
  - `stjm` - joint modelling
  - `stmixed` - multilevel parametric survival
  - `survsim` - simulation of survival data
  - `multistate` - multi-state survival models
- It baffles me that often methods papers do not come with a useable implementation

Statistical software development

- I've written a few packages during and since my PhD, including:
  - `stjm` - joint modelling
  - `stmixed` - multilevel parametric survival
  - `survsim` - simulation of survival data
  - `multistate` - multi-state survival models
- It baffles me that often methods papers do not come with a useable implementation
- Software development within an academic environment has its own unique aspects
  - Use of functional programming should enable rapid development
  - I am a big believer in the philosophy of to learn it, you have to code it!

Statistical software development

- I find it very rewarding

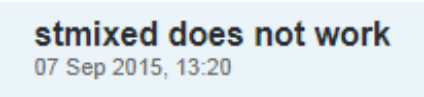
'On behalf of all of us in veterinary epidemiology (i.e. nearly ALWAYS dealing with clustered data) ... thank you very much'

Statistical software development

- I find it very rewarding

'On behalf of all of us in veterinary epidemiology (i.e. nearly ALWAYS dealing with clustered data) ... thank you very much'

- Then again...



Discussion

- A wealth of patient data is becoming available in registry sources, as electronic healthcare record linkage moves to the forefront of life science strategy [21]

Discussion

- A wealth of patient data is becoming available in registry sources, as electronic healthcare record linkage moves to the forefront of life science strategy [21]
- I've presented a very general, and hopefully usable, implementation which can fit a lot of different, and new, models, such as the Royston-Parmar models I showed

Discussion

- A wealth of patient data is becoming available in registry sources, as electronic healthcare record linkage moves to the forefront of life science strategy [21]
- I've presented a very general, and hopefully usable, implementation which can fit a lot of different, and new, models, such as the Royston-Parmar models I showed
- Through the complex linear predictor, we allow seamless development of novel models, and crucially, a way of making them immediately available to researchers through an accessible implementation

Discussion

- A wealth of patient data is becoming available in registry sources, as electronic healthcare record linkage moves to the forefront of life science strategy [21]
- I've presented a very general, and hopefully usable, implementation which can fit a lot of different, and new, models, such as the Royston-Parmar models I showed
- Through the complex linear predictor, we allow seamless development of novel models, and crucially, a way of making them immediately available to researchers through an accessible implementation
- I've incorporated level-specific random effect distributions, and integration techniques

Discussion

- Dynamic risk prediction, predictions will be a key focus of the `megenreg` engine

Discussion

- Dynamic risk prediction, predictions will be a key focus of the `megenreg` engine
- It's so general, and hence it can be slow. E.g. 5 times slower than setting specific implementation (multivariate joint models)

Discussion

- Dynamic risk prediction, predictions will be a key focus of the `megenreg` engine
- It's so general, and hence it can be slow. E.g. 5 times slower than setting specific implementation (multivariate joint models)
- Currently I'm using finite differences for the score and Hessian; however, I am implementing analytic derivatives which will provide substantial speed gains. They themselves rely on numerical approximations

Discussion

- Dynamic risk prediction, predictions will be a key focus of the `megenreg` engine
- It's so general, and hence it can be slow. E.g. 5 times slower than setting specific implementation (multivariate joint models)
- Currently I'm using finite differences for the score and Hessian; however, I am implementing analytic derivatives which will provide substantial speed gains. They themselves rely on numerical approximations
- Software will be released in the coming months...I am also porting it to R

Discussion

- Dynamic risk prediction, predictions will be a key focus of the `megenreg` engine
- It's so general, and hence it can be slow. E.g. 5 times slower than setting specific implementation (multivariate joint models)
- Currently I'm using finite differences for the score and Hessian; however, I am implementing analytic derivatives which will provide substantial speed gains. They themselves rely on numerical approximations
- Software will be released in the coming months...I am also porting it to R
- Crowther MJ. Extended generalised multivariate multilevel data analysis. (To submit).

References I

[1] Gould AL, Boye ME, Crowther MJ, Ibrahim JG, Quartey G, Micallef S, Bois FY. Joint modeling of survival and longitudinal non-survival data: current methods and issues. report of the dia bayesian joint modeling working group. *Statistics in medicine* 2015; **34**(14):2181–2195.

[2] Li N, Elashoff RM, Li G. Robust joint modeling of longitudinal measurements and competing risks failure time data. *Biom J* Feb 2009; **51**(1):19–30, doi:10.1002/bimj.200810491. URL `http://dx.doi.org/10.1002/bimj.200810491`.

[3] Rizopoulos D, Verbeke G, Lesaffre E, Vanrenterghem Y. A two-part joint model for the analysis of survival and longitudinal binary data with excess zeros. *Biometrics* 2008; **64**(2):pp. 611–619. URL `http://www.jstor.org/stable/25502097`.

[4] Lin H, McCulloch CE, Mayne ST. Maximum likelihood estimation in the joint analysis of time-to-event and multiple longitudinal variables. *Stat Med* Aug 2002; **21**(16):2369–2382, doi:10.1002/sim.1179. URL `http://dx.doi.org/10.1002/sim.1179`.

[5] Crowther MJ, Andersson TML, Lambert PC, Abrams KR, Humphreys K. Joint modelling of longitudinal and survival data: incorporating delayed entry and an assessment of model misspecification. *Statistics in medicine* 2016; **35**(7):1193–1209.

References II

[6] Król A, Ferrer L, Pignon JP, Proust-Lima C, Ducreux M, Bouché O, Michiels S, Rondeau V. Joint model for left-censored longitudinal data, recurrent events and terminal event: Predictive abilities of tumor burden for cancer evolution with application to the ffcd 2000–05 trial. *Biometrics* 2016; **72**(3):907–916.

[7] Barrett J, Su L. Dynamic predictions using flexible joint models of longitudinal and time-to-event data. *Statistics in Medicine* 2017; :n/a–n/adoi:10.1002/sim.7209. URL http://dx.doi.org/10.1002/sim.7209, sim.7209.

[8] Król A, Mauguen A, Mazroui Y, Laurent A, Michiels S, Rondeau V. Tutorial in joint modeling and prediction: a statistical software for correlated longitudinal outcomes, recurrent events and a terminal event. *arXiv preprint arXiv:1701.03675* 2017; .

[9] Crowther MJ, Abrams KR, Lambert PC, *et al.*. Joint modeling of longitudinal and survival data. *Stata J* 2013; **13**(1):165–184.

[10] Philipson P, Sousa I, Diggle P, Williamson P, Kolamunnage-Dona R, Henderson R. joineR - Joint Modelling of Repeated Measurements and Time-to-Event Data 2012. URL http://cran.r-project.org/web/packages/joineR/index.html.

References III

[11] Rizopoulos D. JM: An R Package for the Joint Modelling of Longitudinal and Time-to-Event Data. *J Stat Softw* 7 2010; **35**(9):1–33. URL `http://www.jstatsoft.org/v35/i09`.

[12] Rizopoulos D. Jmbayes: joint modeling of longitudinal and time-to-event data under a bayesian approach 2015.

[13] Rabe-Hesketh S, Skrondal A, Pickles A. Reliable estimation of generalized linear mixed models using adaptive quadrature. *Stata J* 2002; **2**:1–21.

[14] Pinheiro JC, Bates DM. Approximations to the log-likelihood function in the nonlinear mixed-effects model. *J Comput Graph Statist* 1995; **4**(1):pp. 12–35.

[15] Crowther MJ, Abrams KR, Lambert PC. Flexible parametric joint modelling of longitudinal and survival data. *Stat Med* 2012; **31**(30):4456–4471, doi:10.1002/sim.5644. URL `http://dx.doi.org/10.1002/sim.5644`.

[16] Henderson R, Diggle P, Dobson A. Joint modelling of longitudinal measurements and event time data. *Biostatistics* 2000; **1**(4):465–480.

[17] Dickman PW, Sloggett A, Hills M, Hakulinen T. Regression models for relative survival. *Stat Med* 2004; **23**(1):51–64, doi:10.1002/sim.1597. URL `http://dx.doi.org/10.1002/sim.1597`.

References IV

[18] Liu L, Wolfe RA, Huang X. Shared frailty models for recurrent events and a terminal event. *Biometrics* 2004; **60**(3):747–756.

[19] Mazroui Y, Mathoulin-Pelissier S, Soubeyran P, Rondeau V. General joint frailty model for recurrent event data with a dependent terminal event: application to follicular lymphoma data. *Statistics in medicine* 2012; **31**(11-12):1162–1176.

[20] Goldstein H, Leckie G, Charlton C, Tilling K, Browne WJ. Multilevel growth curve models that incorporate a random coefficient model for the level 1 variance function. *Statistical methods in medical research* Jan 2017; :962280217706 728doi:10.1177/0962280217706728.

[21] Jutte DP, Roos LL, Brownell MD. Administrative record linkage as a tool for public health research. *Annu Rev Public Health* 2011; **32**:91–108, doi:10.1146/annurev-publhealth-031210-100700. URL `http://dx.doi.org/10.1146/annurev-publhealth-031210-100700`.