Parametric transition-specific multi-state survival models with application to estimating clinically useful measures of risk

Centre for Statistical Methodology
LSHTM
31st March 2017

Michael J. Crowther

Department of Health Sciences
University of Leicester
and
Department of Medical Epidemiology and Biostatistics
Karolinska Institutet
michael.crowther@le.ac.uk

Plan

- Background
- Primary breast cancer example
- Multi-state survival models
  - Common approaches
  - Simulation approach
  - Some extensions
  - Clinically useful measures of absolute risk
  - New Stata `multistate` package
- Current and future research

Background

- In survival analysis, we often concentrate on the time to a single event of interest
- In practice, there are many clinical examples of where a patient may experience a variety of intermediate events
  - Cancer
  - Cardiovascular disease
- This can create complex disease pathways

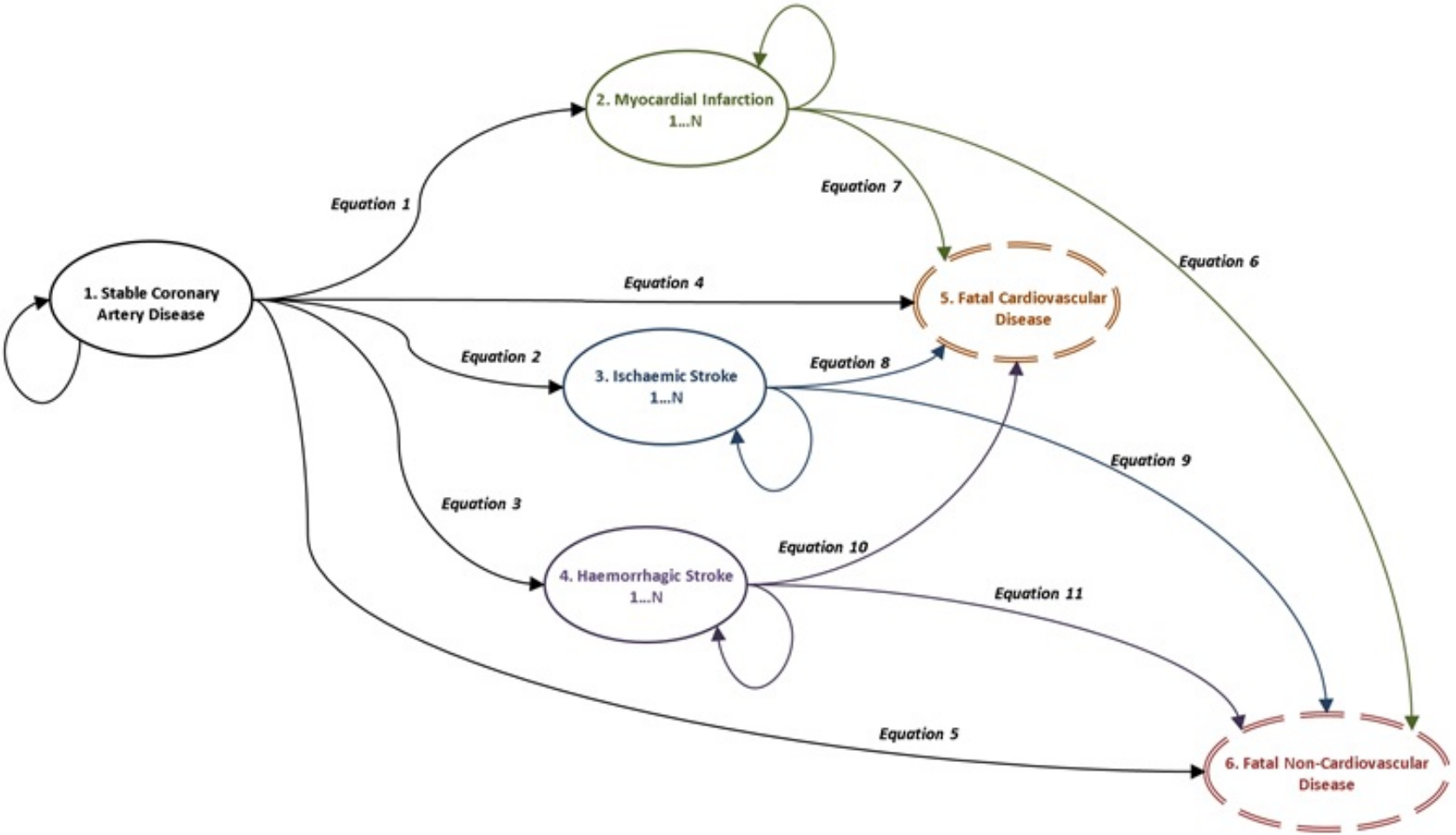


Figure: An example from stable coronary disease (Asaria et al., 2016)

- We want to investigate covariate effects for each specific transition between two states
- With the drive towards personalised medicine, and expanded availability of registry-based data sources, including data-linkage, there are substantial opportunities to gain greater understanding of disease processes, and how they change over time

Primary breast cancer (Sauerbrei et al., 2007)

- To illustrate, I use data from 2,982 patients with primary breast cancer, where we have information on the time to relapse and the time to death.
- All patients begin in the initial 'healthy' state, which is defined as the time of primary surgery, and can then move to a relapse state, or a dead state, and can also die after relapse.
- Covariates of interest include; age at primary surgery, tumour size (three classes; $\leq$ 20mm, 20-50mm, $>$ 50mm), number of positive nodes, progesterone level (fmol/l), and whether patients were on hormonal therapy (binary, yes/no). In all analyses we use a transformation of progesterone level ($\log(pgr + 1)$).

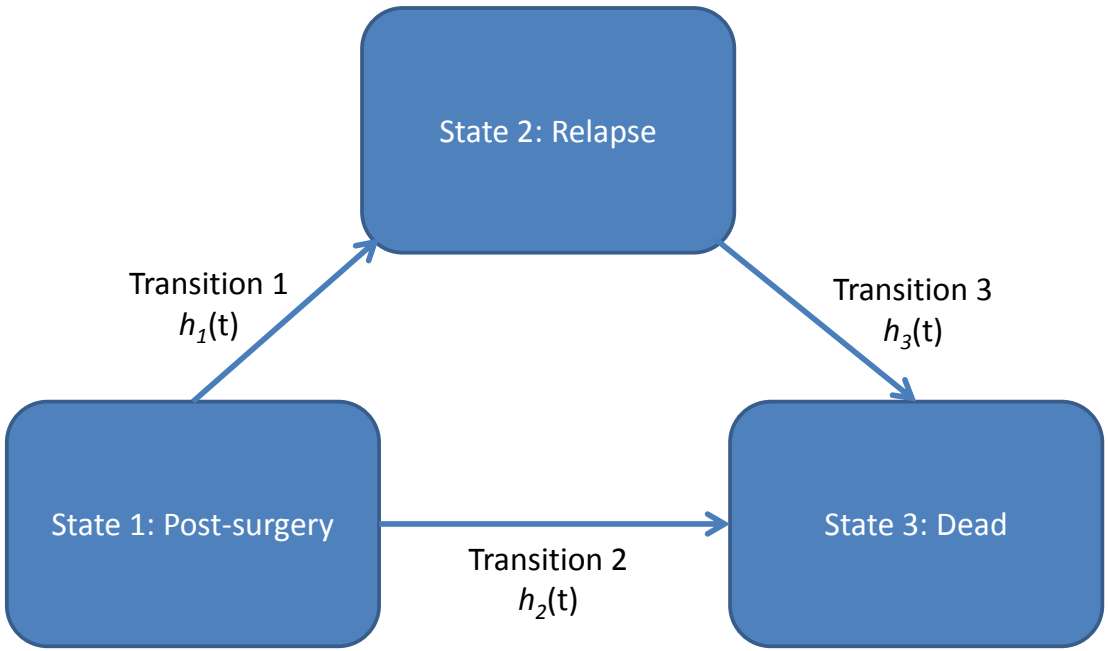


Figure: Illness-death model for primary breast cancer example.

Markov multi-state models

Consider a random process $\{Y(t), t \geq 0\}$ which takes the values in the finite state space $\mathcal{S} = \{1, \ldots, S\}$. We define the history of the process until time s, to be $\mathcal{H}_s = \{Y(u); 0 \leq u \leq s\}$. The transition probability can then be defined as,

$$P(Y(t) = b | Y(s) = a, \mathcal{H}_{s-})$$

where $a, b \in \mathcal{S}$. This is the probability of being in state b at time t, given that it was in state a at time s and conditional on the past trajectory until time s.

Markov multi-state models

A Markov multi-state model makes the following assumption,

$$P(Y(t) = b | Y(s) = a, \mathcal{H}_{s-}) = P(Y(t) = b | Y(s) = a)$$

which implies that the future behaviour of the process is only dependent on the present.

Markov multi-state models

The transition intensity is then defined as,

$$h_{ab}(t) = \lim_{\delta t \to 0} \frac{P(Y(t+\delta t) = b | Y(t) = a)}{\delta t}$$

Or, for the kth transition from state a_k to state b_k, we have

$$h_k(t) = \lim_{\delta t \to 0} \frac{P(Y(t+\delta t) = b_k | Y(t) = a_k)}{\delta t}$$

which represents the instantaneous risk of moving from state a_k to state b_k. Our collection of transitions intensities governs the multi-state model.

Estimating a multi-state models

- There are a variety of challenges in estimating transition probabilities in multi-state models, within both non-/semi-parametric and parametric frameworks (Putter et al., 2007), which I'm not going to go into today
- Essentially, a multi-state model can be specified by a combination of transition-specific survival models
- The most convenient way to do this is through the stacked data notation, where each patient has a row of data for each transition that they are at risk for, using start and stop notation (standard delayed entry setup)

Consider the breast cancer dataset, with recurrence-free and overall survival

```
. list pid rf rfi os osi if pid==1 | pid==1371, sepby(pid) noobs
```

pid	rf	rfi	os	osi
1	59.1	0	59.1	alive
1371	16.6	1	24.3	deceased

We can restructure using `msset`

Title

msset — data preparation for multi-state and competing risks analysis

Syntax

msset [*if*] [*in*] , **id(***varname***) states(***varlist***) times(***varlist***)** [*options*]

options	Description
id(*varname***)**	identification variable
states(*varlist***)**	indicator variables for each state
times(*varlist***)**	time variables for each state
transmatrix(*matname***)**	transition matrix
covariates(*varlist***)**	variables to expand into transition specific covariates

msset creates the following variables:

_from	starting state
_to	receiving state
_trans	transition number
_start	starting time for each transition
_stop	stopping time for each transition
_status	status variable, indicating a transition (coded 1) or censoring (coded 0)
_flag	indicator variable to show observations where changes to the original data have been made

Saved results

msset returns the following in r():

Matrices:

r(Nnextstates)	number of possible next states from starting state (row number)
r(transmatrix)	transition matrix
r(freqmatrix)	frequencies of transitions

```
. list pid rf rfi os osi if pid==1 | pid==1371, sepby(pid) noobs
```

pid	rf	rfi	os	osi
1	59.1	0	59.1	alive
1371	16.6	1	24.3	deceased

```
. list pid rf rfi os osi if pid==1 | pid==1371, sepby(pid) noobs
```

pid	rf	rfi	os	osi
1	59.1	0	59.1	alive
1371	16.6	1	24.3	deceased

```
. msset, id(pid) states(rfi osi) times(rf os) covariates(age)
  variables age_trans1 to age_trans3 created
```

```
. list pid rf rfi os osi if pid==1 | pid==1371, sepby(pid) noobs
```

pid	rf	rfi	os	osi
1	59.1	0	59.1	alive
1371	16.6	1	24.3	deceased

```
. msset, id(pid) states(rfi osi) times(rf os) covariates(age)
  variables age_trans1 to age_trans3 created
. matrix tmat = r(transmatrix)
```

```
. list pid rf rfi os osi if pid==1 | pid==1371, sepby(pid) noobs
```

pid	rf	rfi	os	osi
1	59.1	0	59.1	alive
1371	16.6	1	24.3	deceased

```
. msset, id(pid) states(rfi osi) times(rf os) covariates(age)
  variables age_trans1 to age_trans3 created
. matrix tmat = r(transmatrix)
. list pid _start _stop _from _to _status _trans if pid==1 | pid==1371
```

pid	_start	_stop	_from	_to	_status	_trans
1	0	59.104721	1	2	0	1
1	0	59.104721	1	3	0	2
1371	0	16.558521	1	2	1	1
1371	0	16.558521	1	3	0	2
1371	16.558521	24.344969	2	3	1	3

```
. list pid rf rfi os osi if pid==1 | pid==1371, sepby(pid) noobs
```

pid	rf	rfi	os	osi
1	59.1	0	59.1	alive
1371	16.6	1	24.3	deceased

```
. msset, id(pid) states(rfi osi) times(rf os) covariates(age)
  variables age_trans1 to age_trans3 created
. matrix tmat = r(transmatrix)
. list pid _start _stop _from _to _status _trans if pid==1 | pid==1371
```

pid	_start	_stop	_from	_to	_status	_trans
1	0	59.104721	1	2	0	1
1	0	59.104721	1	3	0	2
1371	0	16.558521	1	2	1	1
1371	0	16.558521	1	3	0	2
1371	16.558521	24.344969	2	3	1	3

```
. stset _stop, enter(_start) failure(_status==1) scale(12)
```

- Now our data is restructured and declared as survival data, we can use any standard survival model available within Stata
  - Proportional baselines across transitions
  - Stratified baselines
  - Shared or separate covariate effects across transitions
- This is all easy to do in Stata; however, calculating transition probabilities (what we are generally most interested in!) is not so easy

Calculating transition probabilities

$$P(Y(t) = b | Y(s) = a)$$

There are a variety of approaches

- Exponential distribution is convenient (Jackson, 2011)
- Numerical integration (Hsieh et al., 2002; Hinchliffe et al., 2013)
- Ordinary differential equations (Titman, 2011)
- Simulation (Iacobelli and Carstensen, 2013; Touraine et al., 2013; Jackson, 2016)

Simulation

- Given our estimated transition intensities, we simulate n patients through the transition matrix
- At specified time points, we simply count how many people are in each state, and divide by the total to get our transition probabilities
- To get confidence intervals, we draw from a multivariate normal distribution, with mean vector the estimated coefficients from the intensity models, and associated variance-covariance matrix, and repeated M times

Simulating survival times

Under a general hazard model

$$h(t) = h_0(t)\exp(X(t)\beta(t))$$

Simulating survival times

Under a general hazard model

$$h(t) = h_0(t)\exp(X(t)\beta(t))$$

$$H(t) = \int_0^t h(u)\ \mathrm{d}u, \quad S(t) = \exp[-H(t)]$$

Simulating survival times

Under a general hazard model

$$h(t) = h_0(t)\exp(X(t)\beta(t))$$

$$H(t) = \int_0^t h(u)\ \mathrm{d}u, \quad S(t) = \exp[-H(t)]$$

$$F(t) = 1 - \exp[-H(t)]$$

Simulating survival times

Under a general hazard model

$$h(t) = h_0(t)\exp(X(t)\beta(t))$$

$$H(t) = \int_0^t h(u)\ \mathrm{d}u, \quad S(t) = \exp[-H(t)]$$

$$F(t) = 1 - \exp[-H(t)]$$

$$U = \exp[-H(t)] \sim \mathrm{U}(0,1)$$

Solve for t...

Simulating survival times

Under a general hazard model

$$h(t) = h_0(t)\exp(X(t)\beta(t))$$

$$H(t) = \int_0^t h(u)\ \mathrm{d}u, \quad S(t) = \exp[-H(t)]$$

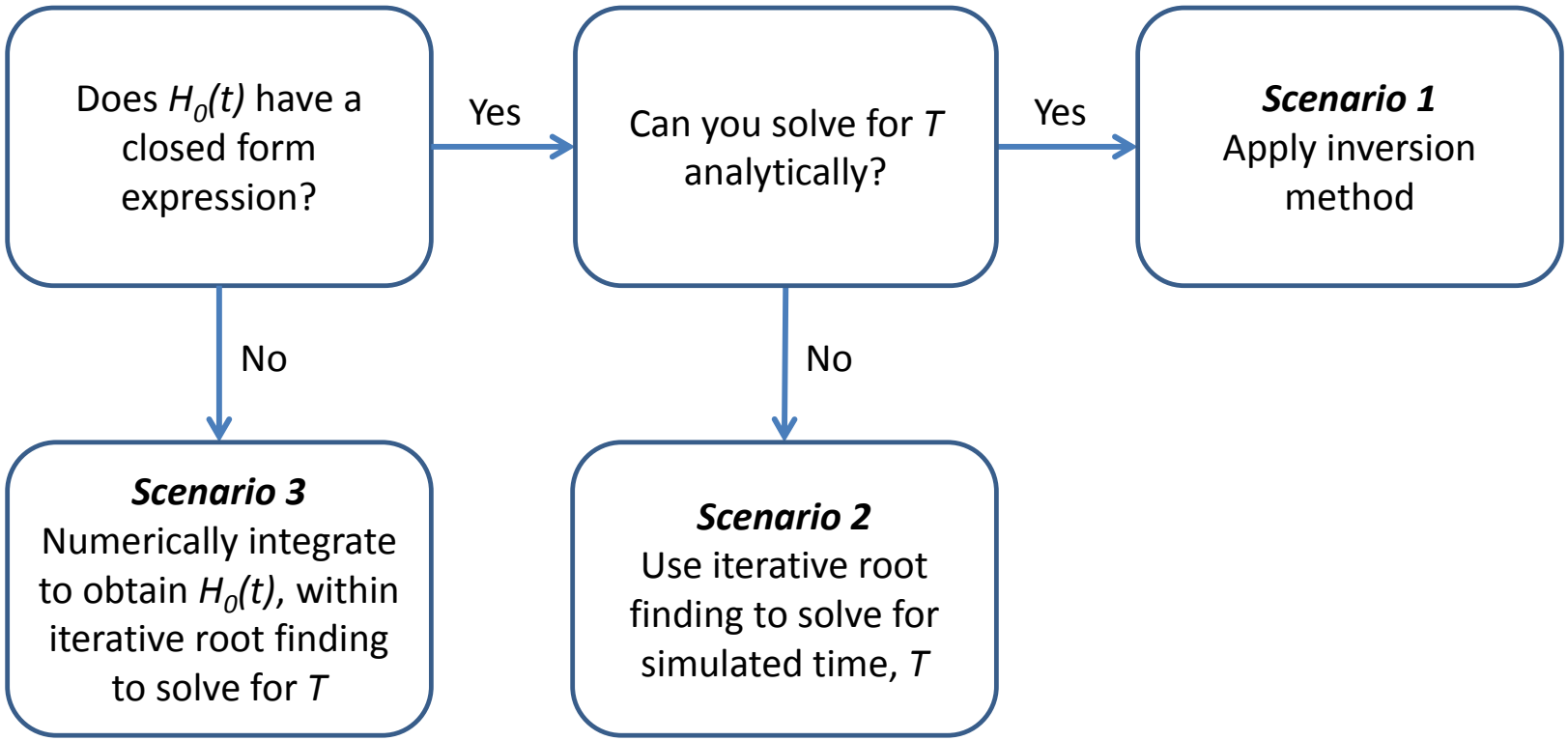
$$F(t) = 1 - \exp[-H(t)]$$

$$U = \exp[-H(t)] \sim \mathrm{U}(0,1)$$

Solve for t... Under a standard parametric PH model,

$$T = H_0^{-1}[-\log(U)\exp(-X\beta)]$$

Simulation methods (Crowther and Lambert, 2013)



Simulation methods

- Standard parametric models (Weibull, Gompertz, etc.) - closed form $H(t)$ and can invert -> extremely efficient
- Royston-Parmar model - closed form $H(t)$ but can't invert -> Brent's univariate root finder
- Splines on the log hazard scale - intractable $H(t)$ and can't invert -> numerical integration and root finding

The last two are not as computationally intensive as you would expect...

Computation time in Stata with `predictms`

- Predicting transition probabilities at 20 evenly spaced points in time across follow-up
- Starting in state 1 at time 0
- Times are in seconds
- Tolerance of <1E-08

n	Weibulls	Royston-Parmar (df=1,5,5)	Log-hazard splines (df=1,5,5)
10,000	0.05	0.31	3.23
100,000	0.30	2.60	32.10
1,000,000	2.50	29.70	302.04
10,000,000	22.35	300.46	3010.30

Baseline only models fit to ebmt3 data

Comparison with `simLexis`

- Predicting transition probabilities at 20 evenly spaced points in time across follow-up
- Starting in state 1 at time 0
- Times are in seconds
- Tolerance of <1E-08
- Poisson - 20 splits in first 2 years, then yearly

	Stata	R
n	Log-hazard splines (df=1,5,5)	Poisson/splines (df=1,5,5)
10,000	3.23	12.97
100,000	32.10	176.50
1,000,000	302.04	Memory error
10,000,000	3010.30	Dream on...

Baseline only models fit to ebmt3 data

Comparison with `flexsurv`

- Predicting transition probabilities at 20 evenly spaced points in time across follow-up
- Starting in state 1 at time 0, and starting in state 2 at time 0
- Times are in seconds

n	Weibull		RP (df=5)	
	Stata	R	Stata	R
10,000	0.08	0.61	2.03	$\approx$ 3600
100,000	0.38	6.79	5.34	...
1,000,000	3.88	59.73	56.36	...
10,000,000	38.71	Crashes	570.4	...

Baseline only models fit to bosms3 data

Extending multi-state models

- What I've described so far assumes the same underlying distribution for every transition
- Consider a set of available covariates X. We therefore define, for the kth transition, the hazard function at time t is,

$$h_k(t) = h_{0k}(t)\exp(X_k\beta_k)$$

where $h_{0k}(t)$ is the baseline hazard function for the $a_k \rightarrow b_k$ transition, which can take any parametric form such that $h_{0k}(t) > 0$. To maintain flexibility, we have a vector of patient-level covariates included in the $a_k \rightarrow b_k$ transition, X_k, where $X_k \in X$.

Proportional baseline, transition specific age effect

```
. streg age_trans1 age_trans2 age_trans3 _trans2 _trans3, dist(weibull)
Weibull regression -- log relative-hazard form
No. of subjects =        7,482                 Number of obs    =        7,482
No. of failures =        2,790
Time at risk    =  38474.53852
                                               LR chi2(5)       =      3057.11
Log likelihood  =   -5547.7893                 Prob > chi2      =       0.0000
```

_t	Haz. Ratio	Std. Err.	z	P>|z|	[95% Conf.	Interval]
age_trans1	.9977633	.0020646	-1.08	0.279	.993725	1.001818
age_trans2	1.127599	.0084241	16.07	0.000	1.111208	1.144231
age_trans3	1.007975	.0023694	3.38	0.001	1.003342	1.01263
_trans2	.0000569	.000031	-17.95	0.000	.0000196	.0001653
_trans3	1.85405	.325532	3.52	0.000	1.314221	2.615619
_cons	.1236137	.0149401	-17.30	0.000	.0975415	.1566547
/ln_p	-.1156762	.0196771	-5.88	0.000	-.1542426	-.0771098
p	.8907636	.0175276			.8570641	.9257882
1/p	1.122632	.0220901			1.080161	1.166774

predictms

```
. predictms, transmat(tmat) at(age 50)
```

predictms

```
. predictms, transmat(tmat) at(age 50) graph
```

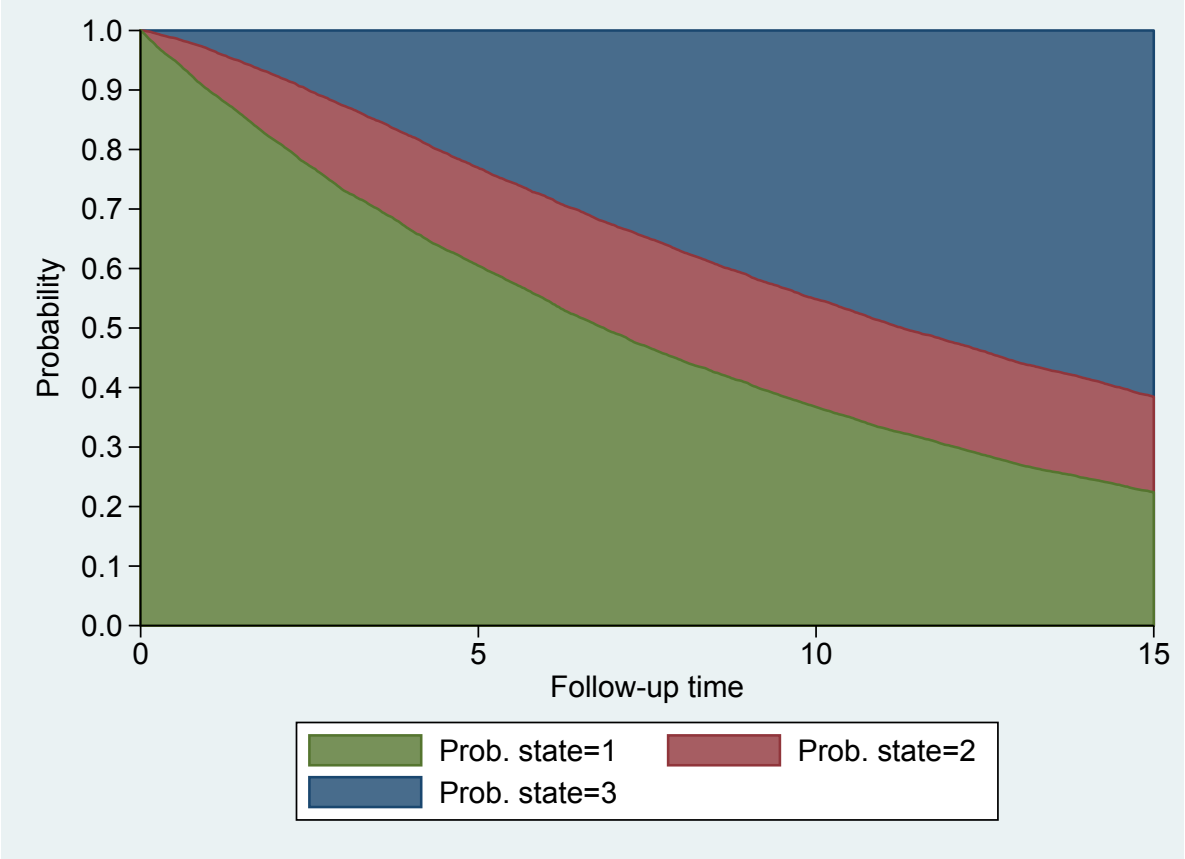


Figure: Predicted transition probabilities.

Extending multi-state models

```
 . streg age_trans1 age_trans2 age_trans3 _trans2 _trans3 ,
> dist(weibull) anc(_trans2 _trans3)

// Is equivalent to...

. streg age if _trans==1, dist(weibull)
. est store m1

. streg age if _trans==2, dist(weibull)

. est store m2

. streg age if _trans==3, dist(weibull)

. est store m3
```

Extending multi-state models

```
 . streg age_trans1 age_trans2 age_trans3 _trans2 _trans3 ,
> dist(weibull) anc(_trans2 _trans3)

// Is equivalent to...

. streg age if _trans==1, dist(weibull)
. est store m1

. streg age if _trans==2, dist(weibull)

. est store m2

. streg age if _trans==3, dist(weibull)

. est store m3


//Predict transition probabilities

. predictms, transmat(tmat) models(m1 m2 m3) at(age 50)
```

Separate models...we can now use *different* distributions

Building our model

Returning to the breast cancer dataset

- Choose the best fitting parametric survival model, using AIC and BIC
- We find that the best fitting model for transitions 1 and 3 is the Royston-Parmar model with 3 degrees of freedom, and the Weibull model for transition 2.
- Adjust for important covariates; age, tumour size, number of nodes, progesterone level
- Check proportional hazards assumption

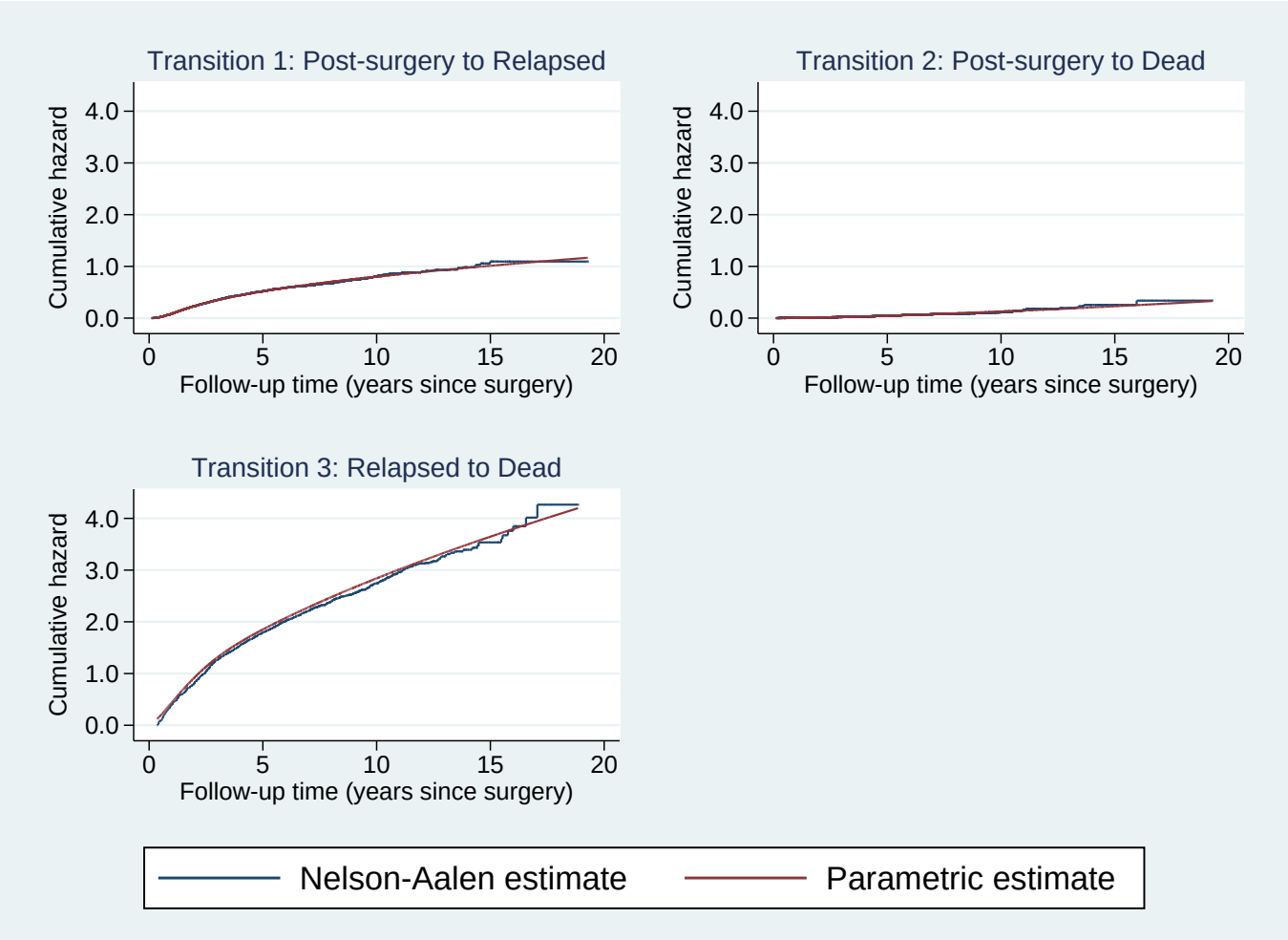


Figure: Best fitting parametric cumulative hazard curves overlaid on the Nelson-Aalen estimate for each transition.

Final model

- Transition 1: Royston-Parmar baseline with df=3, age, tumour size, number of positive nodes, hormonal therapy. Non-PH in tumour size (both levels) and progesterone level, modelled with interaction with log time.
- Transition 2: Weibull baseline, age, tumour size, number of positive nodes, hormonal therapy.
- Transition 3: Royston-Parmar with df=3, age, tumour size, number of positive nodes, hormonal therapy. Non-PH found in progesterone level, modelled with interaction with log time.

```
predictms, transmat(tmat) at(age 54 pr_1 3 sz2 1)
> models(m1 m2 m3)
```

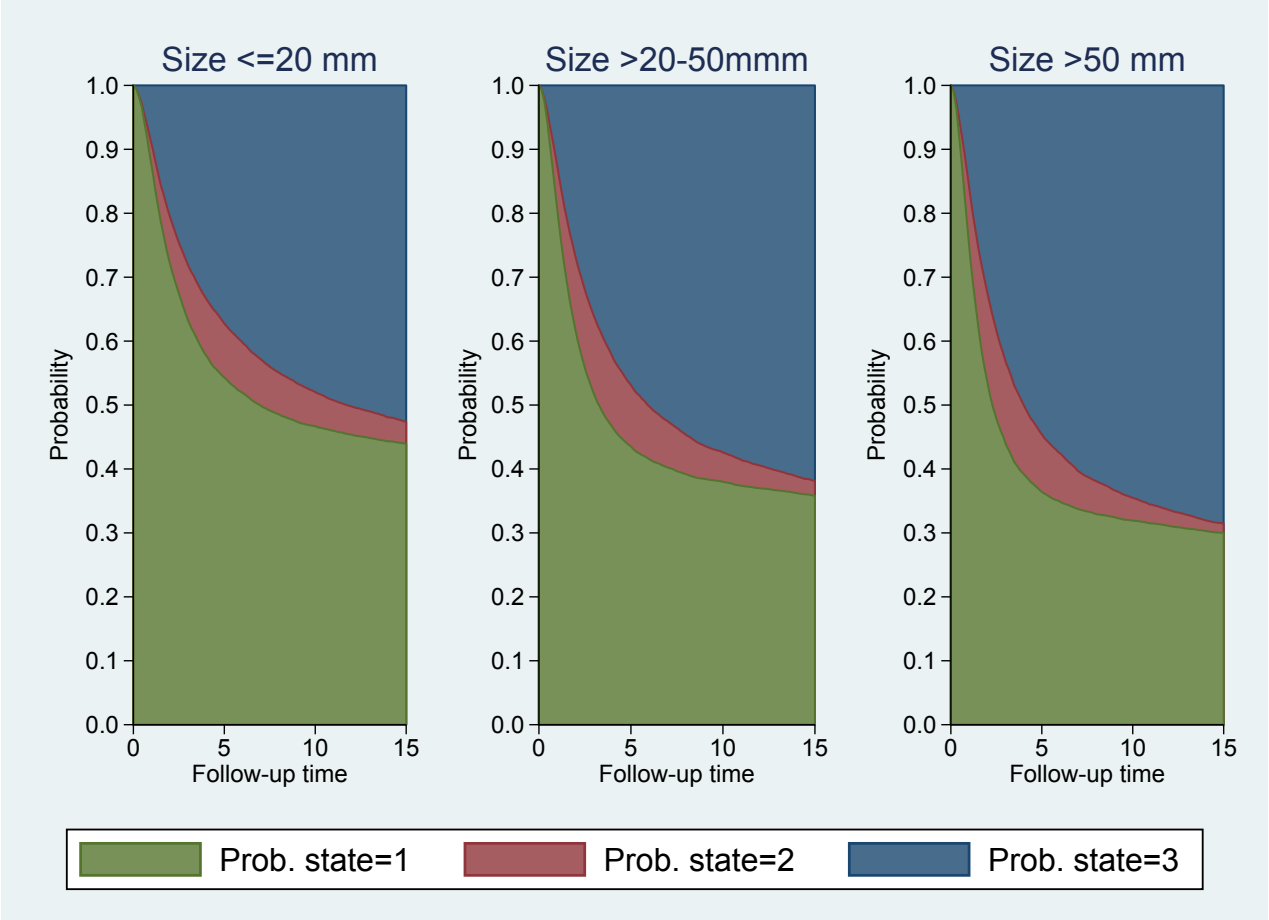


Figure: Probability of being in each state for a patient aged 54, with progesterone level (transformed scale) of 3.

```
predictms, transmat(tmat) at(age 54 pr_1 3 sz2 1)
> models(m1 m2 m3) ci
```

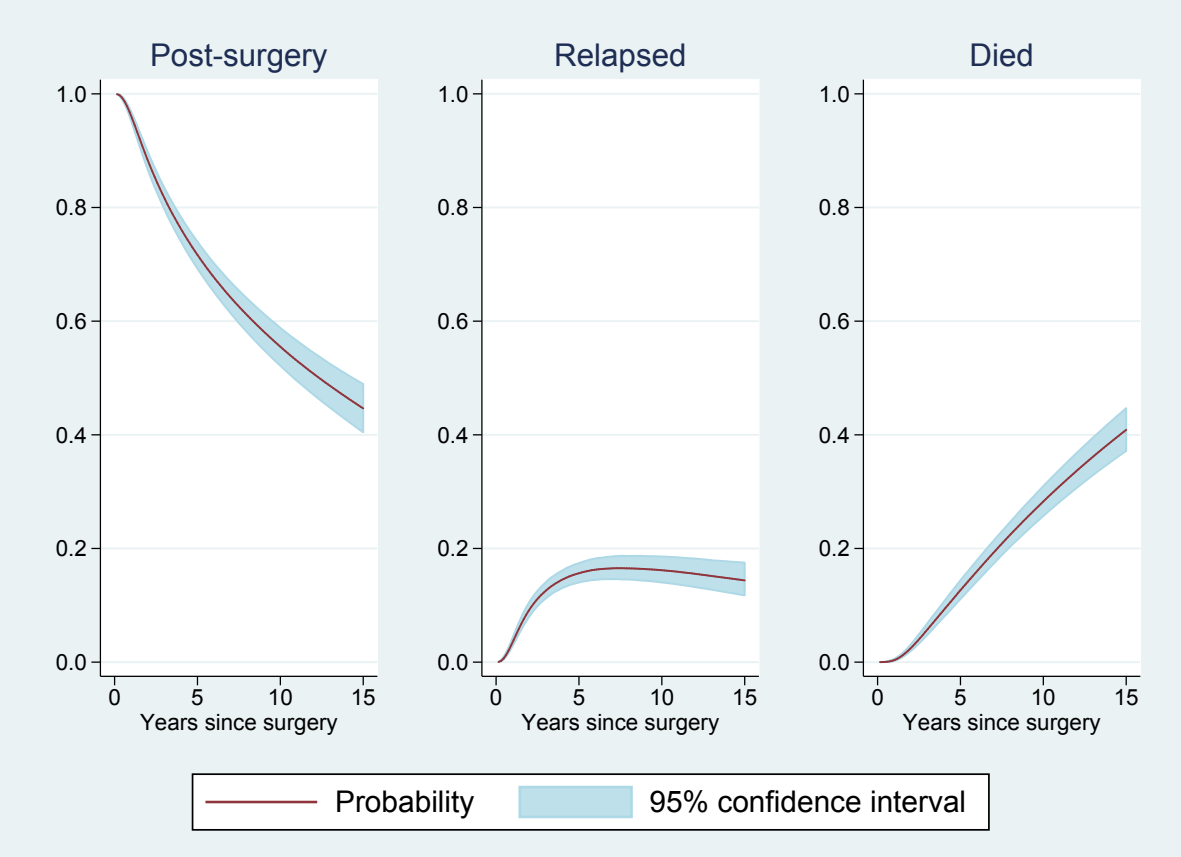
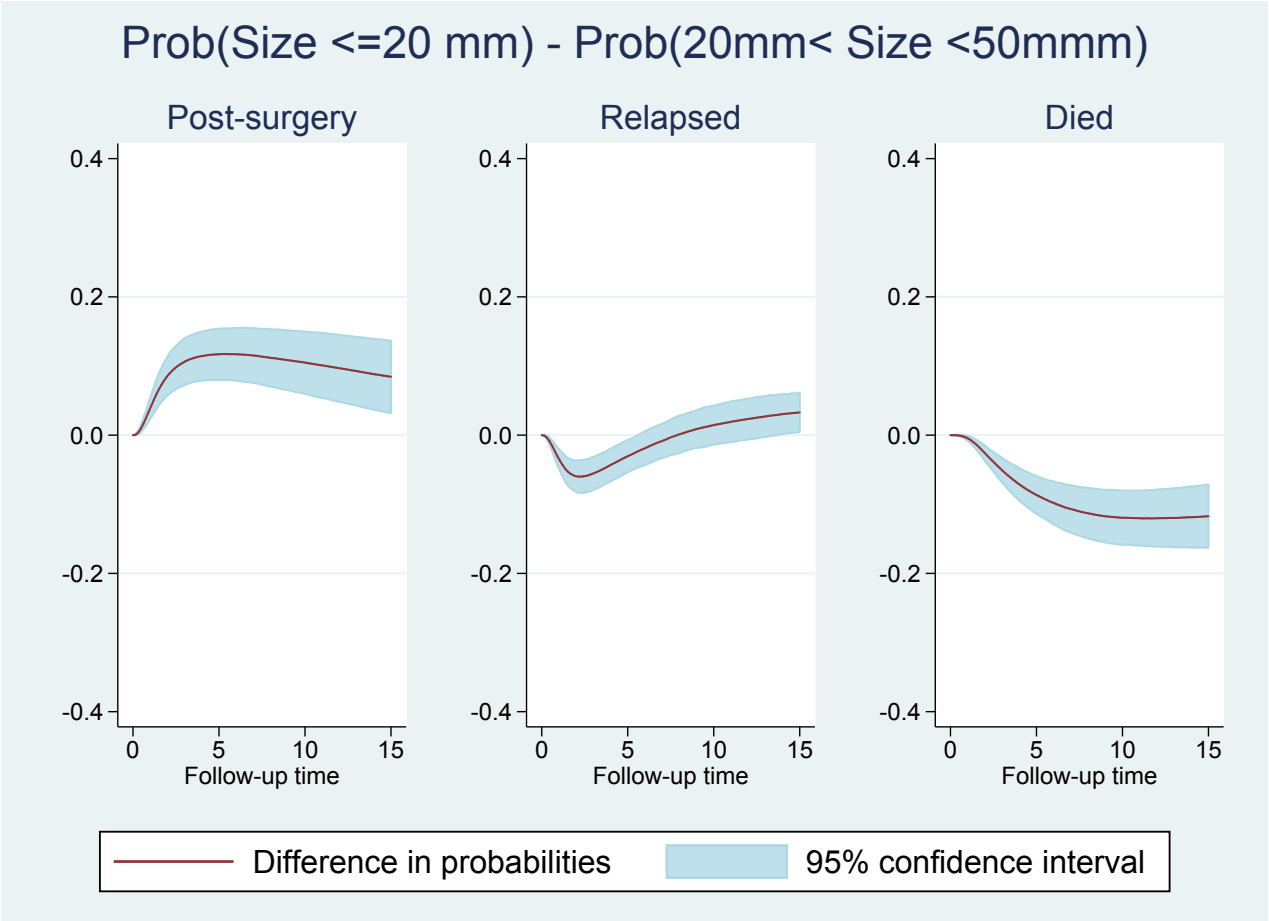


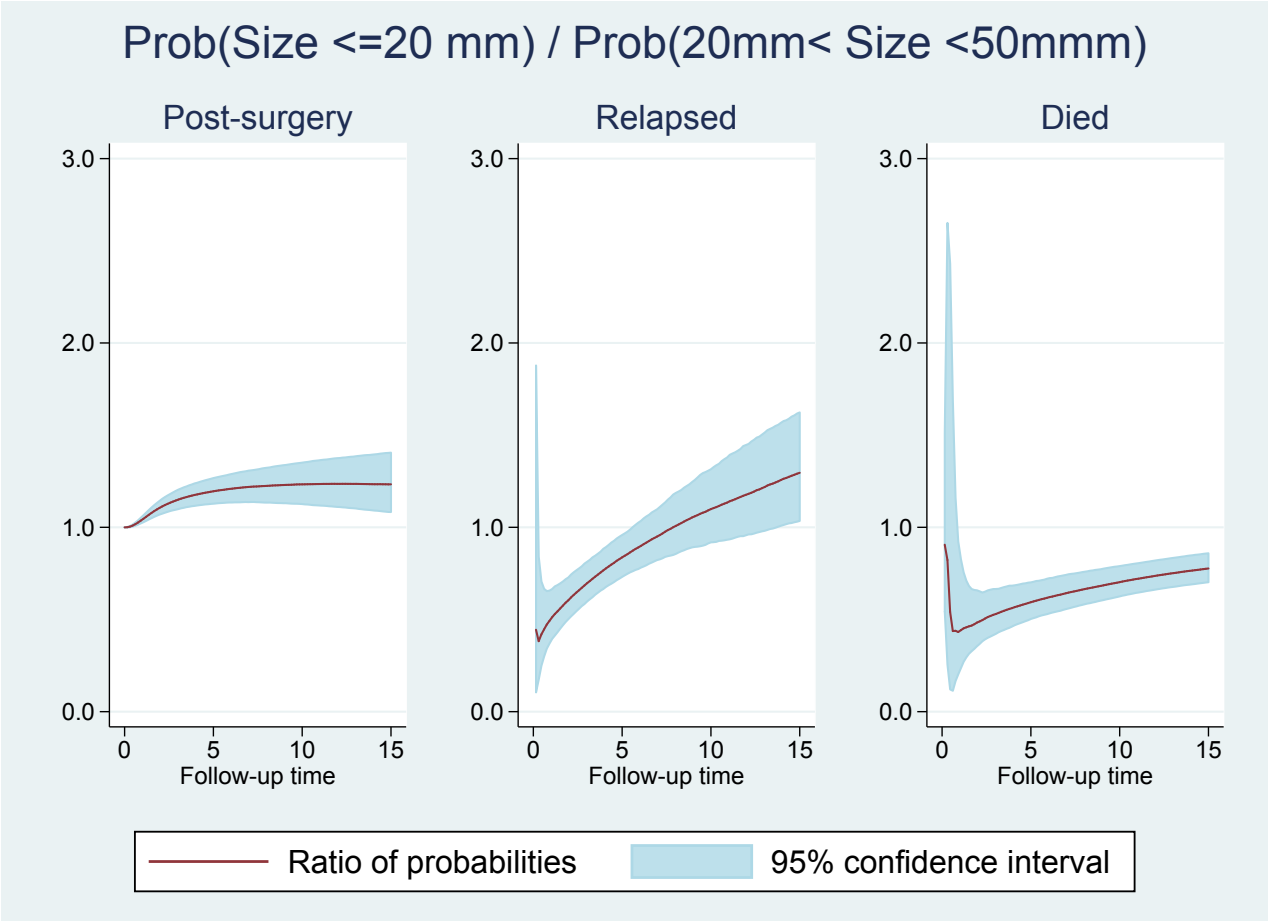
Figure: Probability of being in each state for a patient aged 54, 50> size ≥20 mm, with progesterone level (transformed scale) of 3, and associated confidence intervals.

Differences in transition probabilities




```
. predictms, transmat(tmat) models(m1 m2 m3) ///
.   at(age 54 pgr 3 size1 1) at2(age 54 pgr 3 size2 1) ci
```

Ratios of transition probabilities




```
. predictms, transmat(tmat) models(m1 m2 m3) ///
.   at(age 54 pgr 3 size1 1) at2(age 54 pgr 3 size2 1) ci ratio
```

Length of stay

A clinically useful measure is called length of stay, which defines the amount of time spent in a particular state.

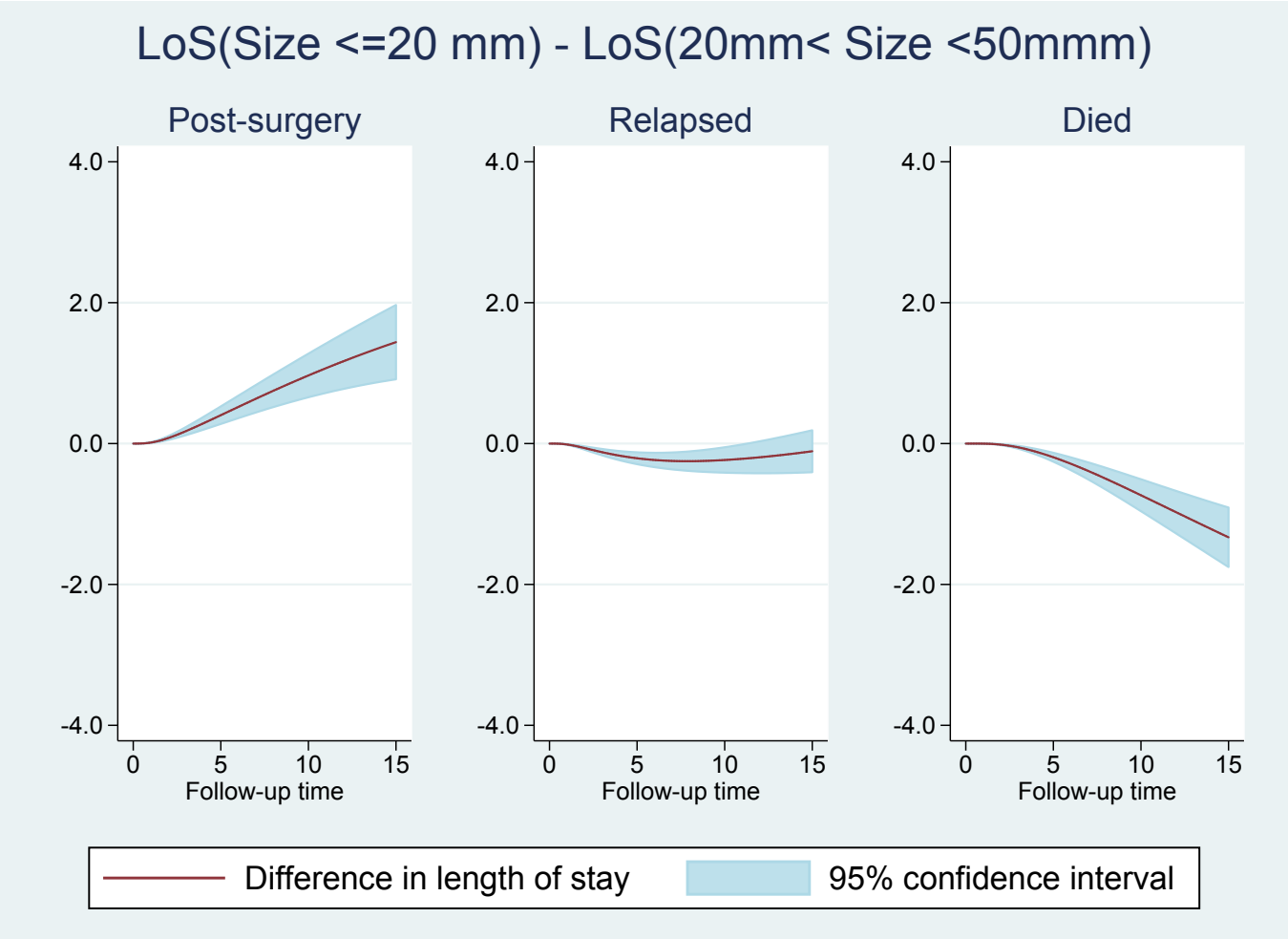
$$\int_s^t P(Y(u) = b | Y(s) = a) du$$

Using this we could calculate life expectancy if t $= \infty$, and $a = b = 1$ (Touraine et al., 2013). Thanks to the simulation approach, we can calculate such things extremely easily.

Length of stay

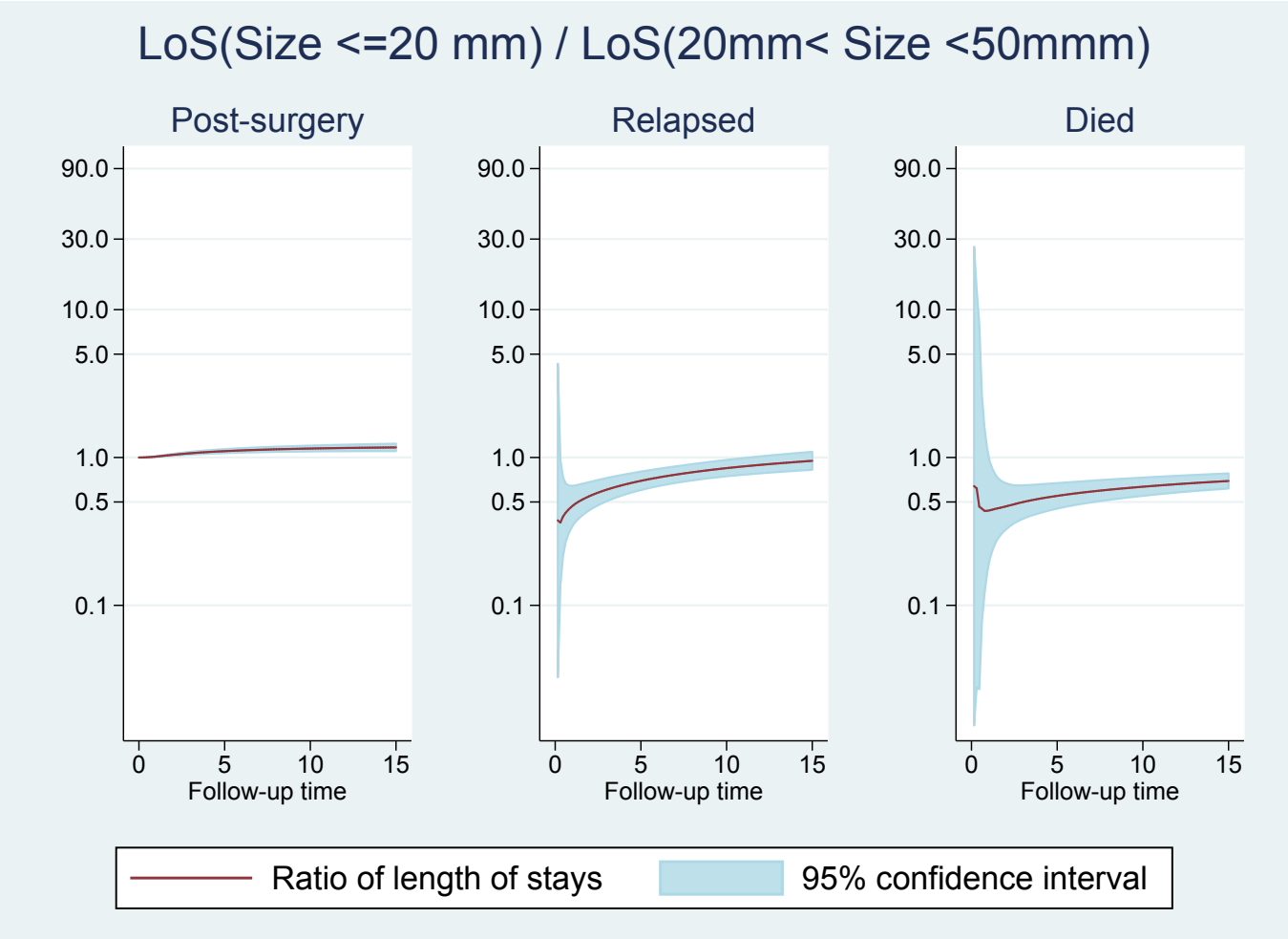

```
. predictms, transmat(tmat) models(m1 m2 m3) ///
.   at(age 54 pgr 3 size1 1) ci los
```

Differences in length of stay




```
. predictms, transmat(tmat) models(m1 m2 m3) ///
.   at(age 54 pgr 3 size1 1) at2(age 54 pgr 3 size2 1) ci los
```

Ratios in length of stay




```
. predictms, transmat(tmat) models(m1 m2 m3) ///
.   at(age 54 pgr 3 size1 1) at2(age 54 pgr 3 size2 1) ci los ratio
```

Sharing covariate effects

- Fitting models separately to each transition means we can no longer share covariate effects - one of the benefits of fitting to the stacked data
- We therefore want to fit different distributions, but jointly, to the stacked data, which will allow us to constrain parameters to be equal across transitions

Transition-specific distributions, estimated jointly

```
. stms (age sz2 sz3 nodes pr_1 hormon, model(rp) df(3) scale(h)) ///
.      (age sz2 sz3 nodes pr_1 hormon, model(weib)) ///
.      (age sz2 sz3 nodes pr_1 hormon, model(rp) df(3) scale(h)) ///
.      , transvar(_trans)
```

Transition-specific distributions, estimated jointly

```
. stms (age sz2 sz3 nodes pr_1 hormon, model(rp) df(3) scale(h)) ///
.      (age sz2 sz3 nodes pr_1 hormon, model(weib)) ///
.      (age sz2 sz3 nodes pr_1 hormon, model(rp) df(3) scale(h)) ///
.      , transvar(_trans) constrain(age 1 3 nodes 2 3)
```

Transition-specific distributions, estimated jointly

```
. stms (age sz2 sz3 nodes pr_1 hormon, model(rp) df(3) scale(h)) ///
.      (age sz2 sz3 nodes pr_1 hormon, model(weib)) ///
.      (age sz2 sz3 nodes pr_1 hormon, model(rp) df(3) scale(h)) ///
.      , transvar(_trans) constrain(age 1 3 nodes 2 3)


. predictms, transmat(tmat) at(age 34 sz2 1 nodes 5) ci
```

Current work

- Standardised/population averaged predictions (Sjölander, 2016)

$$S_{std}(t) = \frac{1}{n}\sum_{i=1}^{n}\hat{S}_i(t|Z_i)$$

- Contrasts

$$\frac{1}{n}\sum_{i=1}^{n}\hat{S}_i(t|X=0,Z_i) - \frac{1}{n}\sum_{i=1}^{n}\hat{S}_i(t|X=1,Z_i)$$

Current work

- Standardised/population averaged transition probabilities (Gran et al., 2015)

$$P_{ab}^{std}(t) = \frac{1}{n}\sum_{i=1}^{n}\hat{P}_{ab}(t|Z_i)$$

- Contrasts

$$\frac{1}{n}\sum_{i=1}^{n}\hat{P}_{ab}(t|X=0,Z_i) - \frac{1}{n}\sum_{i=1}^{n}\hat{P}_{ab}(t|X=1,Z_i)$$

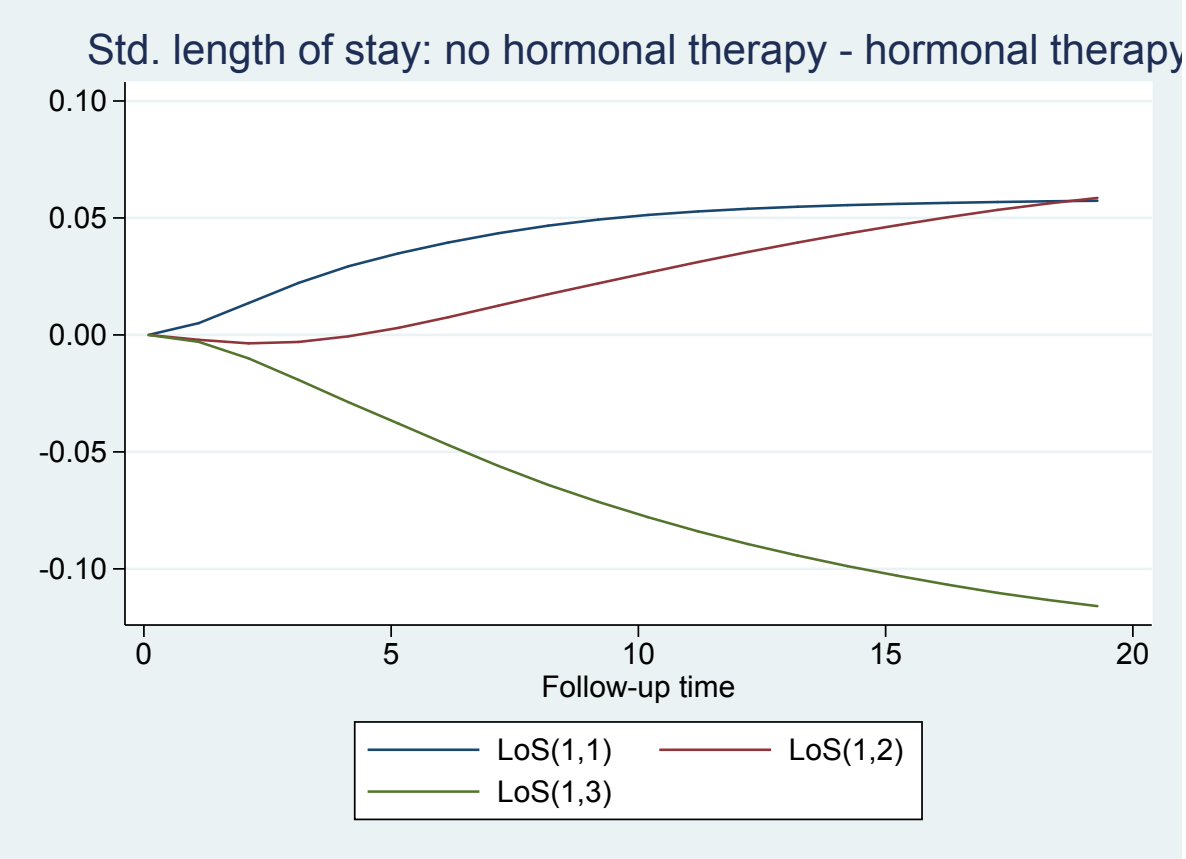
Current work

- Standardised/population averaged length of stay

$$LoS_{ab}^{std}(t) = \frac{1}{n}\sum_{i=1}^{n} \hat{LoS}_{ab}(t|Z_i)$$

- Contrasts

$$\frac{1}{n}\sum_{i=1}^{n} \hat{LoS}_{ab}(t|X=0, Z_i) - \frac{1}{n}\sum_{i=1}^{n} \hat{LoS}_{ab}(t|X=1, Z_i)$$




```
. predictms , transmatrix(tmat) at(hormon 0) at2(hormon 1) los std

Transition 1:  Standardising over -> age sz2 sz3 nodes pr1
Transition 2:  Standardising over -> age sz2 sz3 nodes pr1
Transition 3:  Standardising over -> age sz2 sz3 nodes pr1
```

Summary

- The transition-specific distribution approach I've described provides substantial flexibility

Summary

- The transition-specific distribution approach I've described provides substantial flexibility
- We can fit a very complex model, but immediately obtain interpretable measures of absolute and relative risk

Summary

- The transition-specific distribution approach I've described provides substantial flexibility
- We can fit a very complex model, but immediately obtain interpretable measures of absolute and relative risk
- The simulation approach is extremely versatile and generalisable
  - Further predictions include centiles, plus anything else you can think of!
- Software now makes them accessible
  - `ssc install multistate`

Summary

- The transition-specific distribution approach I've described provides substantial flexibility
- We can fit a very complex model, but immediately obtain interpretable measures of absolute and relative risk
- The simulation approach is extremely versatile and generalisable
  - Further predictions include centiles, plus anything else you can think of!
- Software now makes them accessible
  - `ssc install multistate`
- Other things to mention:
  - Semi-Markov - `reset` with `predictms`
  - Cox model will also be available (`mstate` in R)
  - Reversible transition matrix, interval censoring

References I

Asaria, M., Walker, S., Palmer, S., Gale, C. P., Shah, A. D., Abrams, K. R., Crowther, M., Manca, A., Timmis, A., Hemingway, H., et al. Using electronic health records to predict costs and outcomes in stable coronary artery disease. *Heart*, 102(10):755–762, 2016.

Crowther, M. J. and Lambert, P. C. Simulating biologically plausible complex survival data. *Stat Med*, 32(23): 4118–4134, 2013.

Gran, J. M., Lie, S. A., Øyeflaten, I., Borgan, Ø., and Aalen, O. O. Causal inference in multi-state models–sickness absence and work for 1145 participants after work rehabilitation. *BMC Public Health*, 15(1):1–16, 2015.

Hinchliffe, S. R., Scott, D. A., and Lambert, P. C. Flexible parametric illness-death models. *Stata Journal*, 13(4): 759–775, 2013.

Hsieh, H.-J., Chen, T. H.-H., and Chang, S.-H. Assessing chronic disease progression using non-homogeneous exponential regression Markov models: an illustration using a selective breast cancer screening in Taiwan. *Statistics in medicine*, 21(22):3369–3382, 2002.

Iacobelli, S. and Carstensen, B. Multiple time scales in multi-state models. *Stat Med*, 32(30):5315–5327, Dec 2013.

Jackson, C. flexsurv: A platform for parametric survival modeling in r. *Journal of Statistical Software*, 70(1):1–33, 2016.

Jackson, C. H. Multi-state models for panel data: the msm package for R. *Journal of Statistical Software*, 38(8): 1–29, 2011.

Putter, H., Fiocco, M., and Geskus, R. B. Tutorial in biostatistics: competing risks and multi-state models. *Stat Med*, 26(11):2389–2430, 2007.

Sauerbrei, W., Royston, P., and Look, M. A new proposal for multivariable modelling of time-varying effects in survival data based on fractional polynomial time-transformation. *Biometrical Journal*, 49:453–473, 2007.

Sjölander, A. Regression standardization with the r package stdreg. *European Journal of Epidemiology*, 31(6): 563–574, 2016.

Titman, A. C. Flexible nonhomogeneous Markov models for panel observed data. *Biometrics*, 67(3):780–787, Sep 2011.

Touraine, C., Helmer, C., and Joly, P. Predictions in an illness-death model. *Statistical methods in medical research*, 2013.