

Multistate models:

Occurrence rates, cumulative risks, competing risks,
state probabilities with multiple states and time scales using R and Epi::Lexis

Bendix Carstensen Steno Diabetes Center Copenhagen
Herlev, Denmark
<http://BendixCarstensen.com>

Baker HDI, 22-23 February 2023

<http://bendixcarstensen.com/AdvCoh/courses/Melb-2023>

Acknowledgement

I acknowledge the Traditional Owners of the land on which Baker Institute (Melbourne) resides, the Boon Wurrung peoples of the Yaluk-ut Weelam clan.

I pay my respects to all Elders past, present and future.

Survival and rate data

Rates and Survival

Multistate models:

Occurrence rates, cumulative risks, competing risks,
state probabilities with multiple states and time scales using **R** and `Epi::Lexis`
Baker HDI, 22-23 February 2023

<http://bendixcarstensen.com/AdvCoh/courses/Melb-2023>

surv-rate

Survival data

Persons enter the study at some date.

Persons exit at a later date, either dead or alive.

Observation:

Actual time span to death (“event”)

or

Some time alive (“at least this long”)

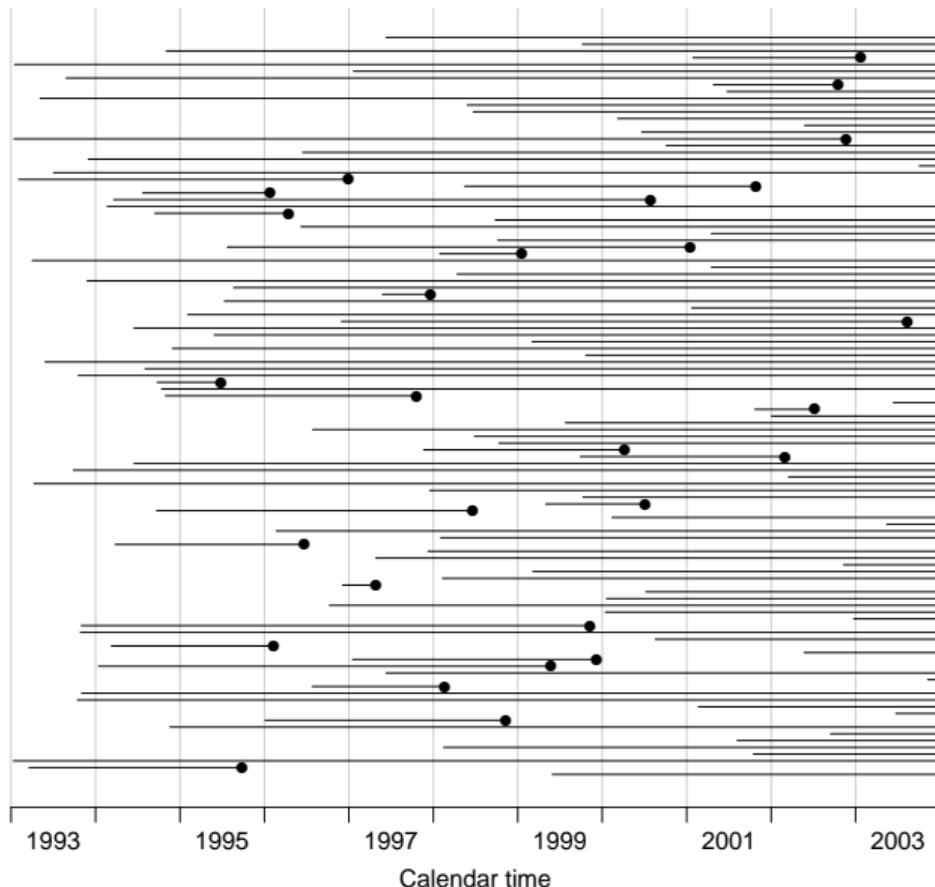
Examples of time-to-event measurements

- ▶ Time from diagnosis of cancer to death.
- ▶ Time from randomisation to death in a cancer clinical trial
- ▶ Time from HIV infection to AIDS.
- ▶ Time from marriage to 1st child birth.
- ▶ Time from marriage to divorce.
- ▶ Time to re-offending after being released from jail

Each line a person

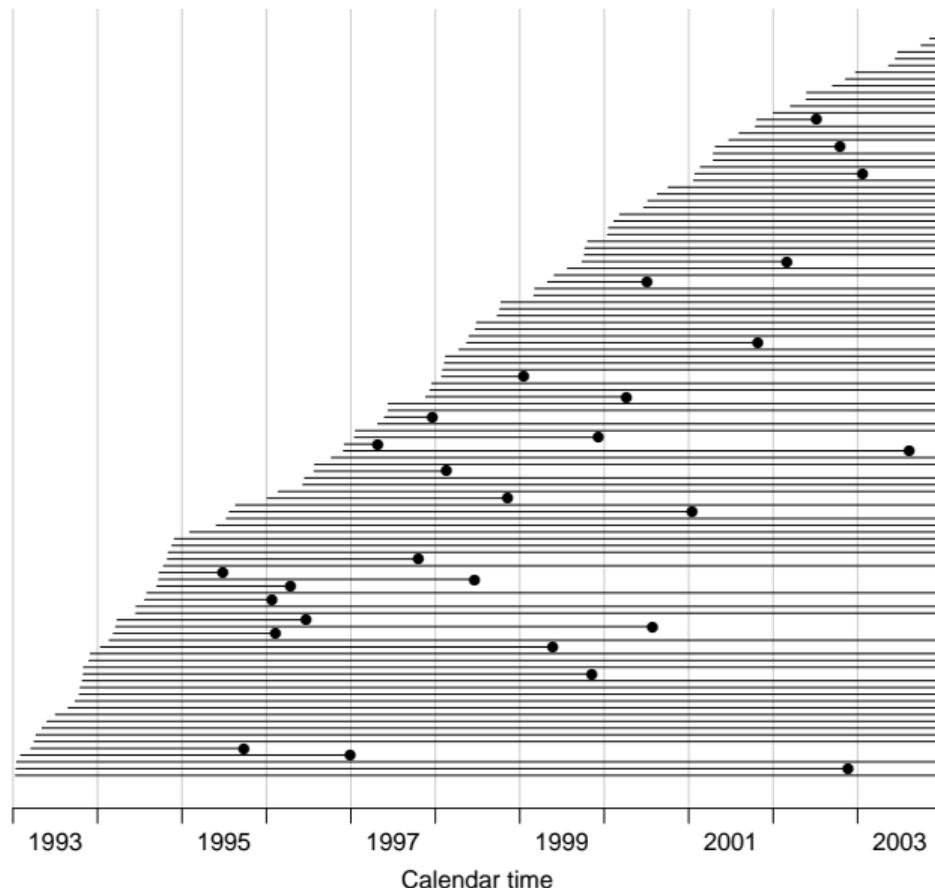
Each blob a death

Study ended at 31
Dec. 2003

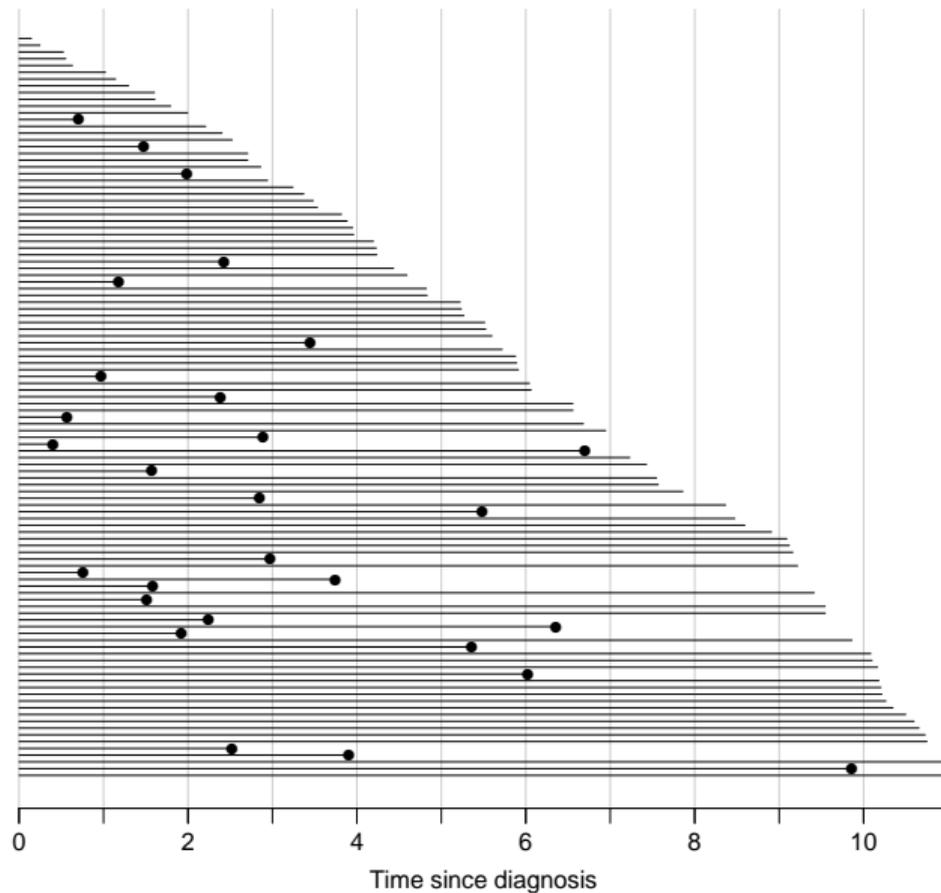


Ordered by date of entry

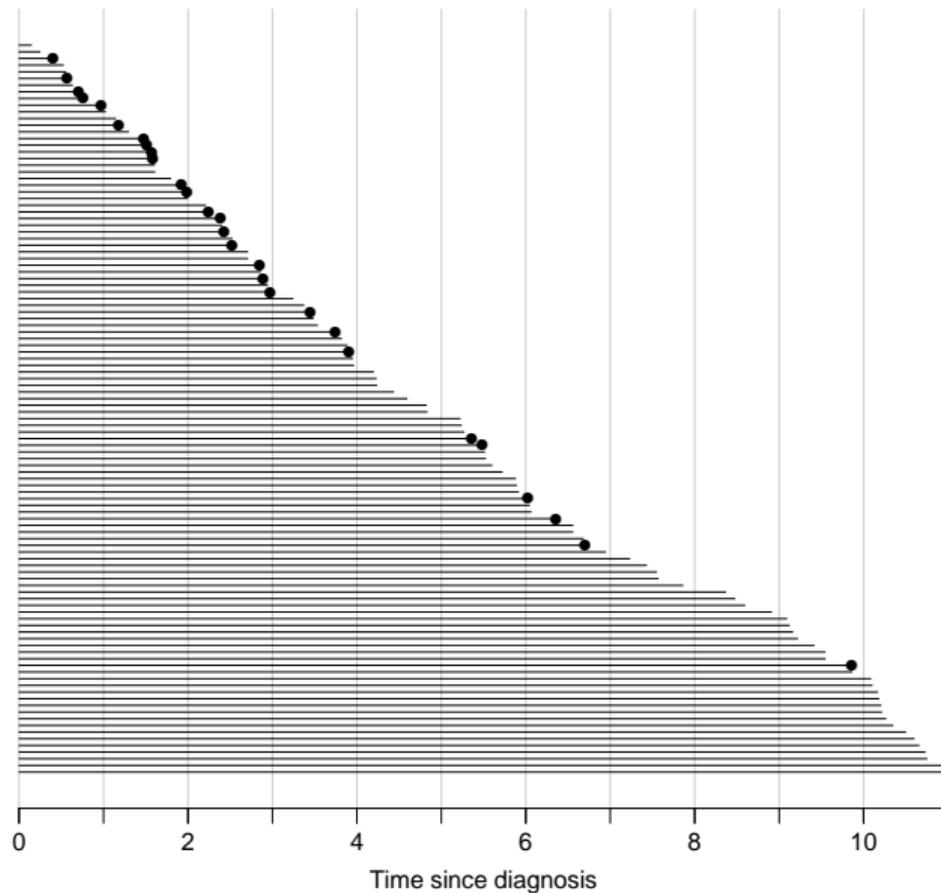
Most likely the order in your database.



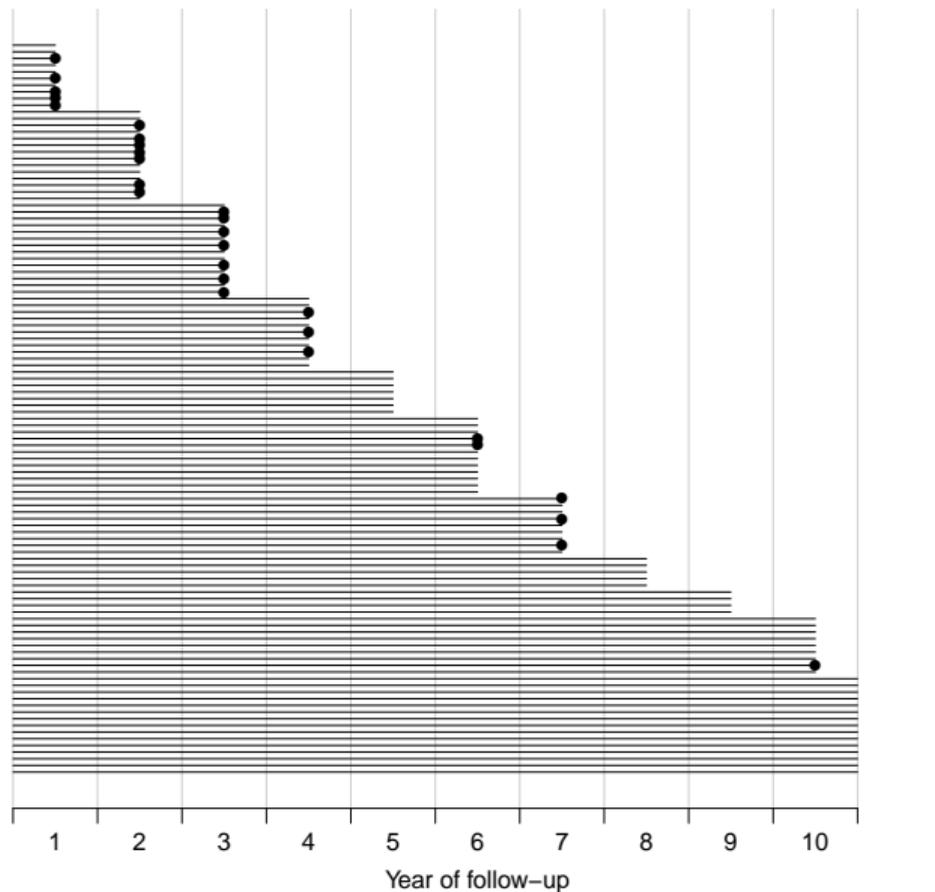
Timescale changed
to
“Time since
diagnosis”.



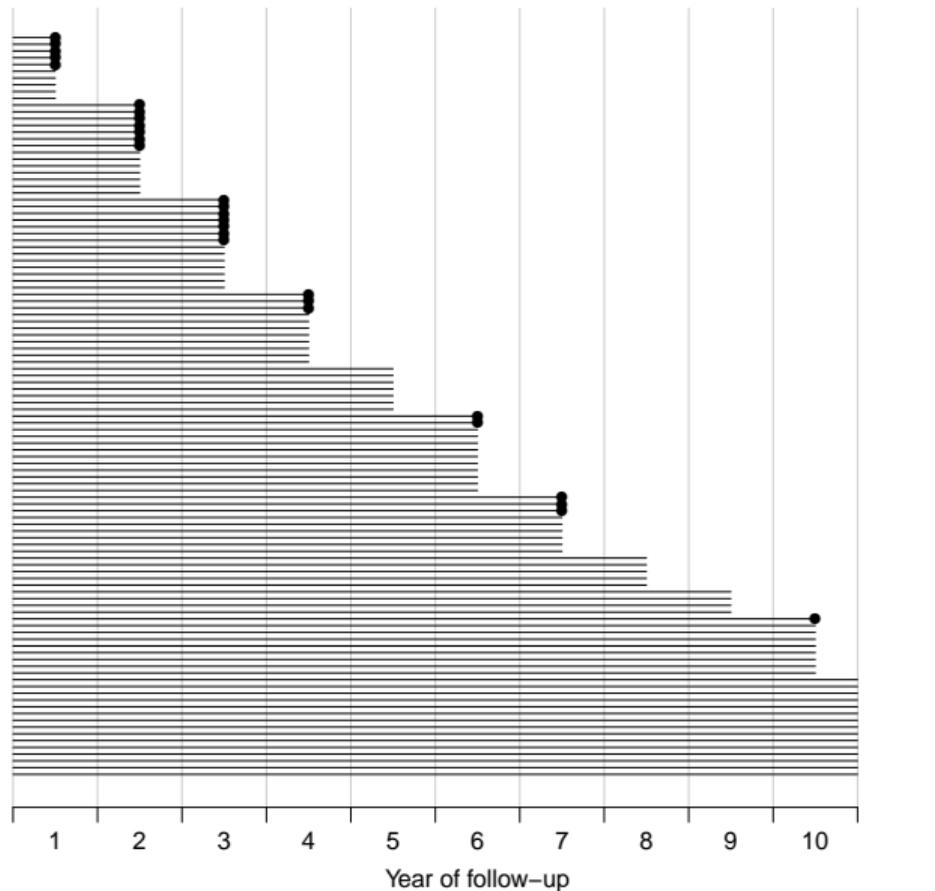
Patients ordered by survival time.



Survival times
grouped into bands
of survival.



Patients ordered by survival status within each band.



Survival after Cervix cancer

Year	Stage I			Stage II		
	<i>N</i>	<i>D</i>	<i>L</i>	<i>N</i>	<i>D</i>	<i>L</i>
1	110	5	5	234	24	3
2	100	7	7	207	27	11
3	86	7	7	169	31	9
4	72	3	8	129	17	7
5	61	0	7	105	7	13
6	54	2	10	85	6	6
7	42	3	6	73	5	6
8	33	0	5	62	3	10
9	28	0	4	49	2	13
10	24	1	8	34	4	6

Life-table estimator of death probability: $D/(N - L/2)$

Estimated risk of death in year 1 for Stage I women is $5/107.5 = 0.0465$

Estimated 1 year survival is $1 - 0.0465 = 0.9535$

Survival after Cervix cancer

Year	Stage I			Stage II		
	<i>N</i>	<i>D</i>	<i>L</i>	<i>N</i>	<i>D</i>	<i>L</i>
1	110	5	5	234	24	3
2	100	7	7	207	27	11
3	86	7	7	169	31	9

Estimated risk in year 1 for Stage I women is $5/107.5 = 0.0465$

Estimated risk in year 2 for Stage I women is $7/96.5 = 0.0725$

Estimated risk in year 3 for Stage I women is $7/82.5 = 0.0848$

Estimated 1 year survival is $1 - 0.0465 = 0.9535$

Estimated 2 year survival is $0.9535 \times (1 - 0.0725) = 0.8843$

Estimated 3 year survival is $0.8843 \times (1 - 0.0848) = 0.8093$

This is the **life-table estimator** of the survival curve.

- ▶ no need to use 1 year intervals: 1 day intervals could be used
- ▶ very small intervals will leave at most 1 censoring or 1 death in each
- ▶ interval with 1 death and n_t persons at risk:
 $P\{\text{Death}\} = 1/n_t$
- ▶ corresponding survival probability $1 - 1/n_t = (n_t - 1)/n_t$
- ▶ interval with 0 deaths has survival probability 1
- ▶ multiply these over times with event to get survival function:

$$S(t) = \prod_{t \text{ with event}} (n_t - 1)/n_t$$

... you have the **Kaplan-Meier estimator**

Multistate models

introduction

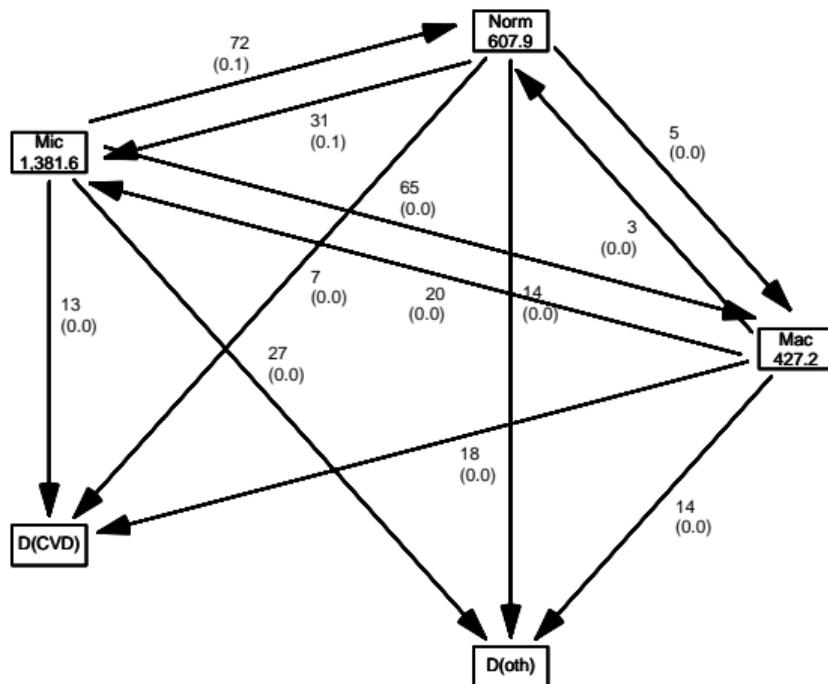
Multistate models:

Occurrence rates, cumulative risks, competing risks,
state probabilities with multiple states and time scales using **R** and `Epi::Lexis`
Baker HDI, 22-23 February 2023

<http://bendixcarstensen.com/AdvCoh/courses/Melb-2023>

MSintro

A multistate model



A multistate model: data

- ▶ Not really a model
- ▶ Data (observations)
 - ▶ sequence of transitions: (**when**, from state, to state)
 - ▶ sequence of: (current state, **time**, next state)
- ▶ Time: covariate or response? . . . both, actually:
 - ▶ **when** something happens
 - is a **covariate** for rates:
how large are rates at a given age, say
 - ▶ risk **time**: how long has the person been at risk
 - this is the part of the outcome
 - ▶ risk **time** is the difference between two **whens**
 - ▶ **whens** are usually dates

A multistate model

- ▶ Target parameters:
 - ▶ Rates (the arrows)
 - ▶ State probabilities (being in a state at a given time)
 - ▶ Survival probability (being alive)
 - ▶ Sojourn times (how long time do you spend in a state)
 - ▶ Expected life time
 - ▶ Probability of ever visiting a state

Data and parameter realms

- ▶ Data: events / (person)time
— the rate dimension (time^{-1})
- ▶ Target parameter dimensions:
 - ▶ rates (dimension time^{-1})
 - ▶ probabilities:
integrals of rates w.r.t. time, requires starting point
—dimension $\text{time}^{-1} \times \text{time} = \langle \text{none} \rangle$
 - ▶ sojourn times:
integrals of probabilities w.r.t. time.
—dimension $\langle \text{none} \rangle \times \text{time} = \text{time}$

What is a statistical model

- ▶ Specification of a statistical machinery that could have generated data
- ▶ ...so with a statistical model we can simulate a data set
- ▶ The basis for the likelihood of data is the statistical **model**
⇒ Estimation of parameters in the model
- ▶ Parameter estimates needed for prediction of rates (hazards)
- ▶ So we need the likelihood of
 the observed data
 given the model
—a function of (the parameters of) the rates.

Data assumptions

- ▶ Individual, accurate data:
- ▶ Exact time of transition between states for all persons

Lung cancer survival

computations

Multistate models:

Occurrence rates, cumulative risks, competing risks,
state probabilities with multiple states and time scales using **R** and `Epi::Lexis`
Baker HDI, 22-23 February 2023

<http://bendixcarstensen.com/AdvCoh/courses/Melb-2023>

surv

Prerequisites

```
> library(Epi)
> library(popEpi)
> # popEpi::splitMulti returns a data.frame rather than a data.table
> options("popEpi.datatable" = FALSE)
```

The lung data set

```
> library(survival)
> data(lung)
> lung$sex <- factor(lung$sex,
+                   levels = 1:2,
+                   labels = c("M", "W"))
> lung$time <- lung$time / (365.25/12)
> head(lung)
```

	inst	time	status	age	sex	ph.ecog	ph.karno	pat.karno	meal.cal	wt.loss
1	3	10.053388	2	74	M	1	90	100	1175	NA
2	3	14.948665	2	68	M	0	90	90	1225	15
3	3	33.182752	1	56	M	0	90	90	NA	15
4	5	6.899384	2	57	M	1	90	60	1150	11
5	1	29.010267	2	60	M	0	100	90	NA	0
6	12	33.577002	1	74	M	1	50	80	513	0

Survival function

- ▶ Use `survfit` to construct the Kaplan-Meier estimator of overall survival:

```
> ?Surv  
> ?survfit
```

```
> km <- survfit(Surv(time, status == 2) ~ 1, data = lung)
```

```
> km
```

```
Call: survfit(formula = Surv(time, status == 2) ~ 1, data = lung)
```

```
      n events median 0.95LCL 0.95UCL  
[1,] 228     165   10.2    9.36   11.9  
> # summary(km) # very long output
```

We can plot the survival curve—this is the default plot for a `survfit` object:

```
> plot(km)
```

What is the median survival? What does it mean? Explore if survival patterns between men and women are different:

```
> kms <- survfit(Surv(time, status == 2) ~ sex, data = lung)
> kms
```

```
Call: survfit(formula = Surv(time, status == 2) ~ sex, data = lung)
```

	n	events	median	0.95LCL	0.95UCL
sex=M	138	112	8.87	6.97	10.2
sex=W	90	53	14.00	11.43	18.1

We see that men have worse survival than women, but they are also a bit older (`age` is age at diagnosis of lung cancer):

```
> with(lung, tapply(age, sex, mean))
```

```
      M      W  
63.34058 61.07778
```

Formally there is a significant difference in survival between men and women

```
> survdiff(Surv(time, status==2) ~ sex, data = lung)
```

Call:

```
survdiff(formula = Surv(time, status == 2) ~ sex, data = lung)
```

	N	Observed	Expected	(O-E) ² /E	(O-E) ² /V
sex=M	138	112	91.6	4.55	10.3
sex=W	90	53	73.4	5.68	10.3

```
Chisq= 10.3 on 1 degrees of freedom, p= 0.001
```

Rates and rate-ratios

- ▶ Occurrence **rate**:

$$\lambda(t) = \lim_{h \rightarrow 0} P \{ \text{event in } (t, t + h] \mid \text{alive at } t \} / h$$

—measured in probability per time: time^{-1}

- ▶ observation in a survival study: (exit status, time alive)
- ▶ empirical rate $(d, y) = (\text{deaths, time})$
- ▶ the Cox model is a model for rates as function of time (t) and covariates (x_1, x_2) :

$$\lambda(t, x) = \lambda_0(t) \exp(\beta_1 x_1 + \beta_2 x_2)$$

—mortality depends on the person's sex and age, say.

- ▶ Data looks like data for a K-M analysis **plus** covariate values

Rates and rate-ratios: Simple Cox model

Now explore how sex and age (at diagnosis) influence the mortality—note that in a Cox-model we are addressing the mortality rate and not the survival:

```
> c0 <- coxph(Surv(time, status == 2) ~ sex, data = lung)
> c1 <- coxph(Surv(time, status == 2) ~ sex + age, data = lung)
> summary(c1)
> ci.exp(c0)
> ci.exp(c1)
```

What variables from `lung` are we using?

```
> c0 <- coxph(Surv(time, status == 2) ~ sex, data = lung)
> c1 <- coxph(Surv(time, status == 2) ~ sex + age, data = lung)
> summary(c1)
```

Call:

```
coxph(formula = Surv(time, status == 2) ~ sex + age, data = lung)
```

n= 228, number of events= 165

	coef	exp(coef)	se(coef)	z	Pr(> z)	
sexW	-0.513219	0.598566	0.167458	-3.065	0.00218	**
age	0.017045	1.017191	0.009223	1.848	0.06459	.

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

	exp(coef)	exp(-coef)	lower .95	upper .95
sexW	0.5986	1.6707	0.4311	0.8311
age	1.0172	0.9831	0.9990	1.0357

Concordance= 0.603 (se = 0.025)

Likelihood ratio test= 14.12 on 2 df, p=9e-04

Wald test = 13.47 on 2 df, p=0.001

Score (logrank) test = 13.72 on 2 df, p=0.001

```

> ci.exp(c0)
      exp(Est.)      2.5%      97.5%
sexW 0.5880028 0.4237178 0.8159848
> ci.exp(c1)
      exp(Est.)      2.5%      97.5%
sexW 0.598566 0.4310936 0.8310985
age 1.017191 0.9989686 1.0357467

```

What do these estimates mean?

$$\lambda(t, x) = \lambda_0(t) \exp(\beta_1 x_1 + \beta_2 x_2)$$

Where is β_1 ? Where is β_2 ? Where is $\lambda_0(t)$?

What is the mortality RR for a 10 year age difference?

If mortality is assumed constant ($\lambda(t) = \lambda$), then the likelihood for the Cox-model is equivalent to a Poisson likelihood, which can be fitted using the `poisreg` family from the `Epi` package:

```
> ?poisreg

> p1 <- glm(cbind(status == 2, time) ~ sex + age,
+          family = poisreg,
+          data = lung)
> ci.exp(p1) # Poisson

              exp(Est.)      2.5%      97.5%
(Intercept) 0.03255152 0.01029228 0.1029511
sexW         0.61820515 0.44555636 0.8577537
age          1.01574132 0.99777446 1.0340317

> ci.exp(c1) # Cox

              exp(Est.)      2.5%      97.5%
sexW  0.598566 0.4310936 0.8310985
age   1.017191 0.9989686 1.0357467
```

Sex and age effects are quite close between the Poisson and the Cox models.

Poisson model has an intercept term, the estimate of the (assumed) constant underlying mortality.

The risk time part of the response (second argument in the `cbind`) was entered in units of months (remember we rescaled in the beginning?), the `(Intercept)` (taken from the `ci.exp`) is a rate per 1 person-month.

What age and sex does the `(Intercept)` refer to?

```
> ci.exp(p1) # Poisson
              exp(Est.)      2.5%      97.5%
(Intercept) 0.03255152 0.01029228 0.1029511
sexW        0.61820515 0.44555636 0.8577537
age         1.01574132 0.99777446 1.0340317
```

poisreg and poisson

```
poisreg: cbind(d,y) ~ ...
```

```
> p1 <- glm(cbind(status == 2, time) ~ sex + age,  
+          family = poisreg,  
+          data = lung)
```

```
poisson: d ~ ... + offset(log(y))
```

```
> px <- glm(status == 2 ~ sex + age + offset(log(time)),  
+          family = poisson,  
+          data = lung)  
> ## or:  
> px <- glm(status == 2 ~ sex + age,  
+          offset = log(time),  
+          family = poisson,  
+          data = lung)
```

Likelihood and records

Suppose a person is alive from t_e (entry) to t_x (exit) and that the person's status at t_x is d , where $d = 0$ means alive and $d = 1$ means dead. If we choose, say, two time points, t_1, t_2 between t_e and t_x , standard use of conditional probability (formally, repeated use of Bayes' formula) gives

$$\begin{aligned} P \{d \text{ at } t_x \mid \text{entry at } t_e\} &= P \{\text{survive } (t_e, t_1] \mid \text{alive at } t_e\} \times \\ &\quad P \{\text{survive } (t_1, t_2] \mid \text{alive at } t_1\} \times \\ &\quad P \{\text{survive } (t_2, t_x] \mid \text{alive at } t_2\} \times \\ &\quad P \{d \text{ at } t_x \mid \text{alive just before } t_x\} \end{aligned}$$

Rates and likelihood

For a start assume that the mortality is constant over time $\lambda(t) = \lambda$:

$$\begin{aligned} P \{ \text{death during } (t, t + h] \} &\approx \lambda h & (1) \\ \Rightarrow P \{ \text{survive } (t, t + h] \} &\approx 1 - \lambda h \end{aligned}$$

where the approximation gets better the smaller h is.

Dividing follow-up time

- ▶ Survival for a time span: $y = t_x - t_e$
- ▶ Subdivided in N intervals, each of length $h = y/N$
- ▶ Survival probability for the entire span from t_e to t_x is the **product** of probabilities of surviving each of the small intervals, conditional on being alive at the beginning each interval:

$$P \{ \text{survive } t_e \text{ to } t_x \} \approx (1 - \lambda h)^N = \left(1 - \frac{\lambda y}{N} \right)^N$$

Dividing follow-up time

- ▶ From mathematics it is known that $(1 + x/n)^n \rightarrow \exp(x)$ as $n \rightarrow \infty$ (some define $\exp(x)$ this way).
- ▶ So if we divide the time span y in small pieces we will have that $N \rightarrow \infty$:

$$P \{\text{survive } t_e \text{ to } t_x\} \approx \left(1 - \frac{\lambda y}{N}\right)^N \rightarrow \exp(-\lambda y), \quad N \rightarrow \infty \quad (2)$$

- ▶ The contribution to the likelihood from a person observed for a time span of length y is $\exp(-\lambda y)$, and the contribution to the log-likelihood is therefore $-\lambda y$.

Dividing follow-up time

- ▶ A person dying at the end of the last interval, the contribution to the likelihood from the last interval will be
- ▶ the probability surviving till just before the end of the interval,
- ▶ **multiplied** by
- ▶ the probability of dying in the last tiny instant (of length ϵ) of the interval
- ▶ The probability of dying in this tiny instant is $\lambda\epsilon$
- ▶ log-likelihood contribution from this last instant is $\log(\lambda\epsilon) = \log(\lambda) + \log(\epsilon)$.

Total likelihood

The total likelihood for one person is the product of all these terms from the follow-up intervals (i) for the person; and the log-likelihood (ℓ) is therefore the sum of the log-likelihood terms:

$$\begin{aligned}\ell(\lambda) &= \sum_i (-\lambda y_i + d_i \log(\lambda) + d_i \log(\epsilon)) \\ &= \sum_i (d_i \log(\lambda) - \lambda y_i) + \sum_i d_i \log(\epsilon)\end{aligned}$$

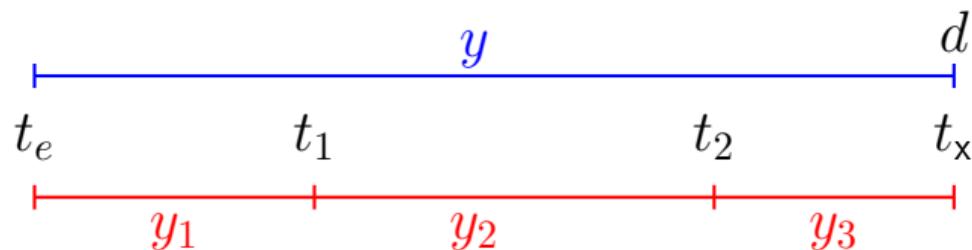
The last term does not depend on λ , so it can be ignored

Total log-likelihood

- ▶ ... for the follow up of 1 person is (the **rate** likelihood):

$$\sum_i (d_i \log(\lambda) - \lambda y_i)$$

- ▶ this is also the likelihood for independent Poisson variates d_i with means λy_i .
- ▶ even though the d_i s are neither Poisson nor independent
- ▶ Different models can have the same (log)likelihood:
 - ▶ model for follow-up of a person (d_i, y_i) , constant rate λ
 - ▶ model for independent Poisson variates (d_i) , mean λy_i



Probability

$$P(d \text{ at } t_x | \text{entry } t_e)$$

$$= P(\text{surv } t_e \rightarrow t_1 | \text{entry } t_e)$$

$$\times P(\text{surv } t_1 \rightarrow t_2 | \text{entry } t_1)$$

$$\times P(d \text{ at } t_x | \text{entry } t_2)$$

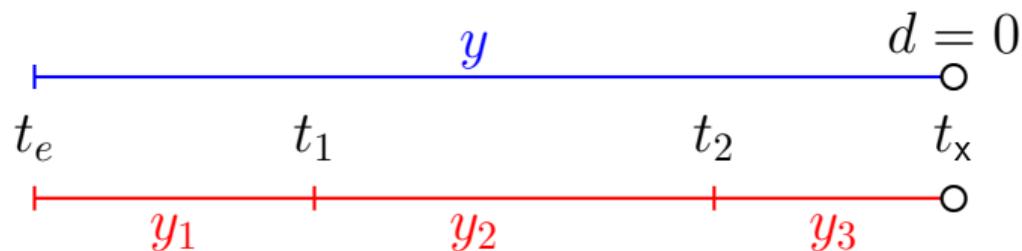
log-Likelihood

$$d \log(\lambda) - \lambda y$$

$$= 0 \log(\lambda) - \lambda y_1$$

$$+ 0 \log(\lambda) - \lambda y_2$$

$$+ d \log(\lambda) - \lambda y_3$$



Probability

$$P(\text{surv } t_e \rightarrow t_x | \text{entry } t_e)$$

$$= P(\text{surv } t_e \rightarrow t_1 | \text{entry } t_e)$$

$$\times P(\text{surv } t_1 \rightarrow t_2 | \text{entry } t_1)$$

$$\times P(\text{surv } t_2 \rightarrow t_x | \text{entry } t_2)$$

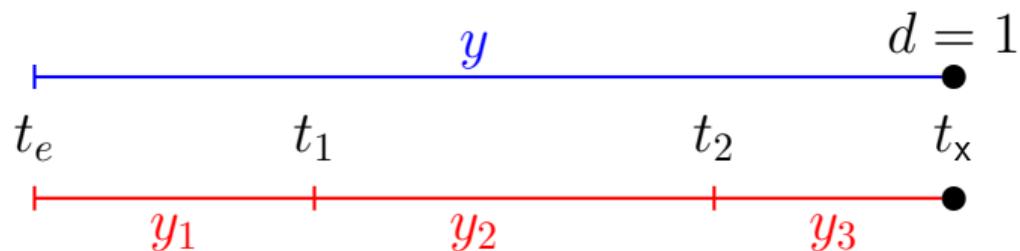
log-Likelihood

$$0 \log(\lambda) - \lambda y$$

$$= 0 \log(\lambda) - \lambda y_1$$

$$+ 0 \log(\lambda) - \lambda y_2$$

$$+ 0 \log(\lambda) - \lambda y_3$$



Probability

$$P(\text{event at } t_x | \text{entry } t_e)$$

$$= P(\text{surv } t_e \rightarrow t_1 | \text{entry } t_e)$$

$$\times P(\text{surv } t_1 \rightarrow t_2 | \text{entry } t_1)$$

$$\times P(\text{event at } t_x | \text{entry } t_2)$$

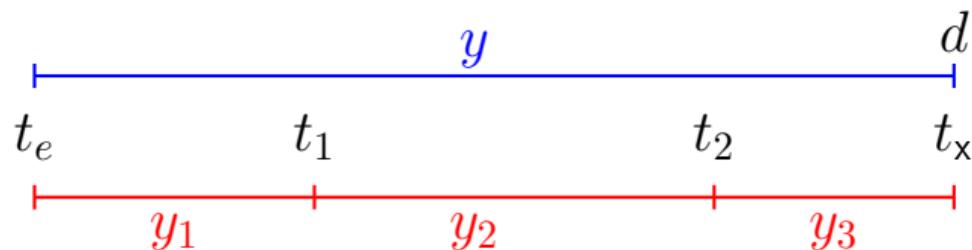
log-Likelihood

$$1 \log(\lambda) - \lambda y$$

$$= 0 \log(\lambda) - \lambda y_1$$

$$+ 0 \log(\lambda) - \lambda y_2$$

$$+ 1 \log(\lambda) - \lambda y_3$$



Probability

$$P(d \text{ at } t_x | \text{entry } t_e)$$

$$= P(\text{surv } t_e \rightarrow t_1 | \text{entry } t_e)$$

$$\times P(\text{surv } t_1 \rightarrow t_2 | \text{entry } t_1)$$

$$\times P(d \text{ at } t_x | \text{entry } t_2)$$

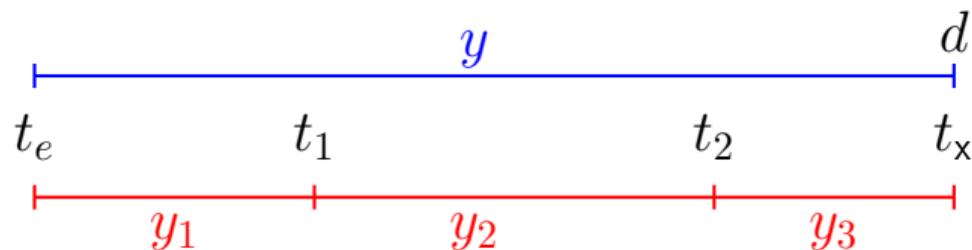
log-Likelihood

$$d \log(\lambda) - \lambda y$$

$$= 0 \log(\lambda) - \lambda y_1$$

$$+ 0 \log(\lambda) - \lambda y_2$$

$$+ d \log(\lambda) - \lambda y_3$$



Probability

$$P(d \text{ at } t_x | \text{entry } t_e)$$

$$= P(\text{surv } t_e \rightarrow t_1 | \text{entry } t_e)$$

$$\times P(\text{surv } t_1 \rightarrow t_2 | \text{entry } t_1)$$

$$\times P(d \text{ at } t_x | \text{entry } t_2)$$

log-Likelihood

$$d \log(\lambda) - \lambda y$$

$$= 0 \log(\lambda_1) - \lambda_1 y_1$$

$$+ 0 \log(\lambda_2) - \lambda_2 y_2$$

$$+ d \log(\lambda_3) - \lambda_3 y_3$$

— allows different rates (λ_i) in each interval

Representation of follow-up: Lexis object

```
> L1 <- Lexis(exit = list(tfl = time),
+             exit.status = factor(status,
+                                   levels = 1:2,
+                                   labels = c("Alive", "Dead")),
+             data = lung)
```

NOTE: entry.status has been set to "Alive" for all.

NOTE: entry is assumed to be 0 on the tfl timescale.

```
> head(L1)
```

lex.id	tfl	lex.dur	lex.Cst	lex.Xst	inst	time	status	age	sex	ph.ecog	ph.karno
1	0	10.05	Alive	Dead	3	10.053	2	74	M	1	90
2	0	14.95	Alive	Dead	3	14.949	2	68	M	0	90
3	0	33.18	Alive	Alive	3	33.183	1	56	M	0	90
4	0	6.90	Alive	Dead	5	6.899	2	57	M	1	90
5	0	29.01	Alive	Dead	1	29.010	2	60	M	0	100
6	0	33.58	Alive	Alive	12	33.577	1	74	M	1	50
pat.karno	meal.cal	wt.loss									
100	1175	NA									
90	1225	15									
90	NA	15									
60	1150	11									

New variables in a Lexis object

`tfl`: time from lung cancer **at the time of entry**, therefore it is 0 for all persons; the entry time is 0 from the date of lung cancer. Defines a **timescale** with name `tfl`.

`lex.dur`: the **length** of time a person is in state `lex.Cst`, here measured in months, because `time` is.

`lex.Cst`: Current `s`tate, the state in which the `lex.dur` time is spent.

`lex.Xst`: eXit `s`tate, the state to which the person moves after the `lex.dur` time in `lex.Cst`.

`lex.id`: an id of each record in the source dataset. Can be explicitly set by `id=`.

Lexis object: Overview of follow-up

Overkill?

The point is that the machinery generalizes to multistate data.

```
> summary(L1)
```

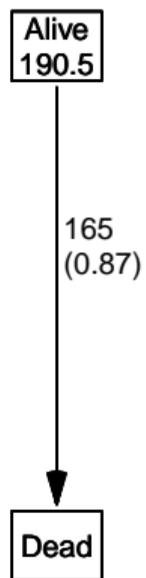
```
Transitions:
```

```
  To
```

```
From   Alive  Dead  Records:  Events:  Risk time:  Persons:
  Alive    63   165         228         165       2286.42         228
```

What is the average follow-up time for persons?

```
> boxes(L1, boxpos = TRUE, scale.Y = 12, digits.R = 2)
```



Explain the numbers in the graph.

Cox model using the **Lexis**-specific variables:

```
> c1 <- coxph(Surv(tfl,  
+             tfl + lex.dur,  
+             lex.Xst == "Dead") ~ sex + age,  
+             data = L1)
```

Surv(from-time, to-time, event indicator)

Using the **Lexis** features:

```
> cL <- coxph.Lexis(L1, tfl ~ sex + age)
```

survival::coxph analysis of Lexis object L1:

Rates for the transition:

Alive->Dead

Baseline timescale: tfl

```
> round(cbind(ci.exp(cL),  
+             ci.exp(c1)), 3)
```

	exp(Est.)	2.5%	97.5%	exp(Est.)	2.5%	97.5%
sexW	0.599	0.431	0.831	0.599	0.431	0.831
age	1.017	0.999	1.036	1.017	0.999	1.036

The crude Poisson model:

```
> pc <- glm(cbind(lex.Xst == "Dead", lex.dur) ~ sex + age,  
+          family = poisreg,  
+          data = L1)
```

or even simpler, by using the **Lexis** features:

```
> pL <- glm.Lexis(L1, ~ sex + age)
```

stats::glm Poisson analysis of Lexis object L1 with log link:

Rates for the transition:

Alive->Dead

```
> round(cbind(ci.exp(pL),  
+            ci.exp(pc)), 3)
```

	exp(Est.)	2.5%	97.5%	exp(Est.)	2.5%	97.5%
(Intercept)	0.033	0.010	0.103	0.033	0.010	0.103
sexW	0.618	0.446	0.858	0.618	0.446	0.858
age	1.016	0.998	1.034	1.016	0.998	1.034

Poisson and Cox model

The crude Poisson model is a Cox-model with the (quite brutal) assumption that baseline rate is constant over time.

But results are similar:

```
> round(cbind(ci.exp(cL),  
+             ci.exp(pL)[-1,]), 3)
```

	exp(Est.)	2.5%	97.5%	exp(Est.)	2.5%	97.5%
sexW	0.599	0.431	0.831	0.618	0.446	0.858
age	1.017	0.999	1.036	1.016	0.998	1.034

Baseline hazard: splitting time

```
> S1 <- splitMulti(L1, tfl = 0:36)
> summary(L1)
```

Transitions:

To

From	Alive	Dead	Records:	Events:	Risk time:	Persons:
Alive	63	165	228	165	2286.42	228

```
> summary(S1)
```

Transitions:

To

From	Alive	Dead	Records:	Events:	Risk time:	Persons:
Alive	2234	165	2399	165	2286.42	228

What happened to no. records?

What happened to amount of risk time?

What happened to no. events?

```

> wh <- names(L1)[1:10] # names of variables in some order
> subset(L1, lex.id == 10)[,wh]

lex.id tfl lex.dur lex.Cst lex.Xst inst time status age sex
    10   0   5.45   Alive   Dead    7 5.454     2  61   M

> subset(S1, lex.id == 10)[,wh]

lex.id tfl lex.dur lex.Cst lex.Xst inst time status age sex
    10   0   1.00   Alive   Alive    7 5.454     2  61   M
    10   1   1.00   Alive   Alive    7 5.454     2  61   M
    10   2   1.00   Alive   Alive    7 5.454     2  61   M
    10   3   1.00   Alive   Alive    7 5.454     2  61   M
    10   4   1.00   Alive   Alive    7 5.454     2  61   M
    10   5   0.45   Alive   Dead    7 5.454     2  61   M

```

In `S1` each record now represents a small interval of follow-up for a person, so each person has many records.

Natural splines for baseline hazard

```
> ps <- glm(cbind(lex.Xst == "Dead", lex.dur)
+           ~ Ns(tfl, knots = seq(0, 36, 12)) + sex + age,
+           family = poisreg,
+           data = S1)
```

or even simpler:

```
> ps <- glm.Lexis(S1, ~ Ns(tfl, knots = seq(0, 36, 12)) + sex + age)
```

stats::glm Poisson analysis of Lexis object S1 with log link:

Rates for the transition:

Alive->Dead

```
> ci.exp(ps)
```

	exp(Est.)	2.5%	97.5%
(Intercept)	0.0189837	0.005700814	0.06321569
Ns(tfl, knots = seq(0, 36, 12))1	2.4038681	0.809442081	7.13896863
Ns(tfl, knots = seq(0, 36, 12))2	4.1500822	0.436273089	39.47798357
Ns(tfl, knots = seq(0, 36, 12))3	0.8398973	0.043928614	16.05849662
sexW	0.5987171	0.431232662	0.83124998
age	1.0165872	0.998377104	1.03512945

Comparing with estimates from the Cox-model and from the model with constant baseline:

```
> round(cbind(ci.exp(c1),  
+           ci.exp(ps, subset = c("sex", "age")),  
+           ci.exp(pc, subset = c("sex", "age"))), 3)
```

	exp(Est.)	2.5%	97.5%	exp(Est.)	2.5%	97.5%	exp(Est.)	2.5%	97.5%
sexW	0.599	0.431	0.831	0.599	0.431	0.831	0.618	0.446	0.858
age	1.017	0.999	1.036	1.017	0.998	1.035	1.016	0.998	1.034

But where is the baseline hazard?

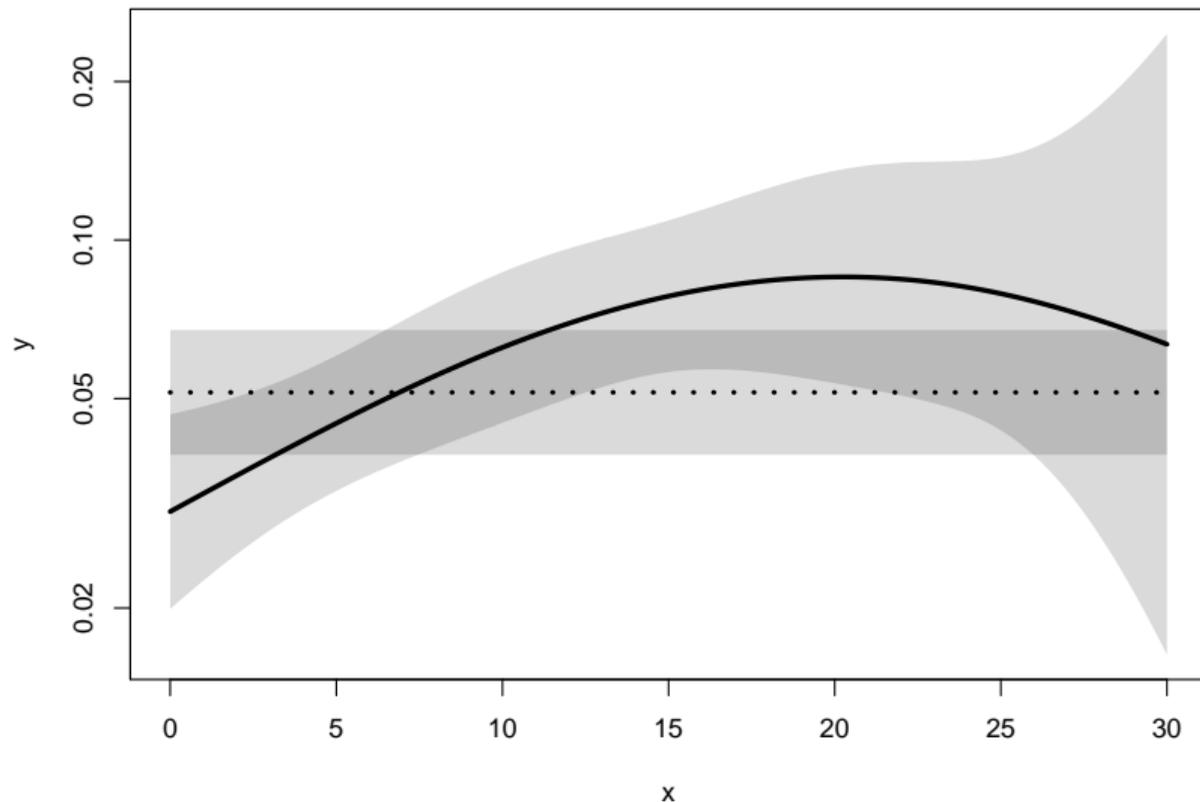
`ps` is a model for the hazard so we can predict the value of it at defined values for the covariates in the model:

```
> prf <- data.frame(tfl = seq(0, 30, 0.2),  
+                   sex = "W",  
+                   age = 60)
```

We can over-plot with the predicted rates from the model where mortality rates are constant, the only change is the model (`pc` instead of `ps`):

```
> matshade(prf$tfl, ci.pred(ps, prf),  
+          plot = TRUE, log = "y", lwd = 3)  
> matshade(prf$tfl, ci.pred(pc, prf), lty = 3, lwd = 3)
```

Here is the baseline hazard!



Survival function and hazard function

$$S(t) = \exp\left(-\int_0^t \lambda(u) \, du\right)$$

Simple, but the CI for $S(t)$ not so simple...

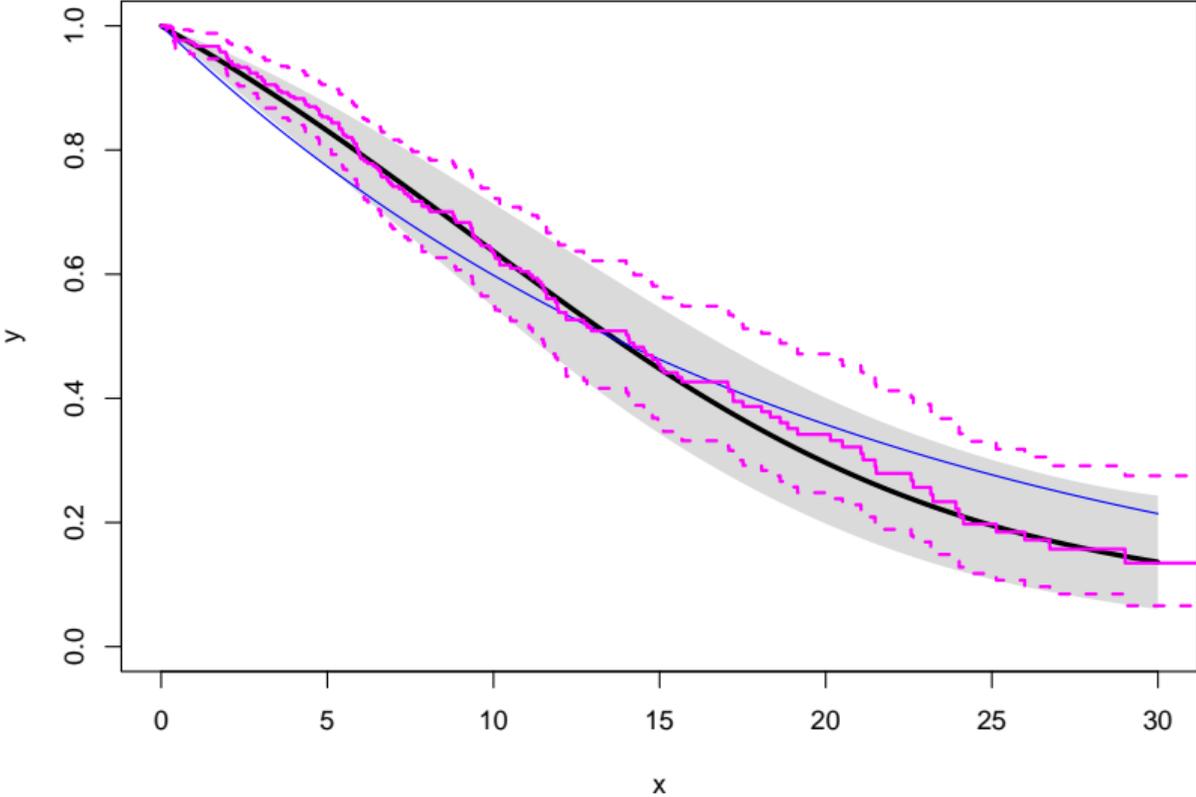
Implemented in the `ci.surv` function

Arguments: 1:model, 2:prediction data frame, 3:equidistance

Prediction data frame must correspond to a sequence of equidistant time points:

```
> matshade(prf$tfl, ci.surv(ps, prf, intl = 0.2),  
+          plot = TRUE, ylim = 0:1, lwd = 3)  
> lines(prf$tfl, ci.surv(pc, prf, intl = 0.2)[,1], col="blue")  
> lines(survfit(c1, newdata = data.frame(sex = "W", age = 60)),  
+       lwd = 2, lty = 1, col="magenta")
```

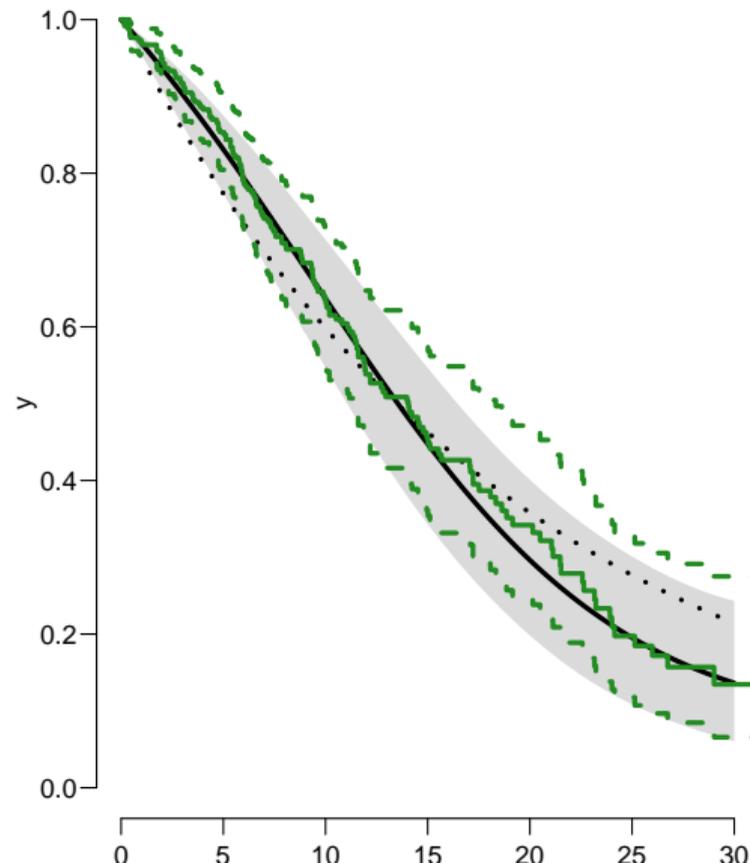
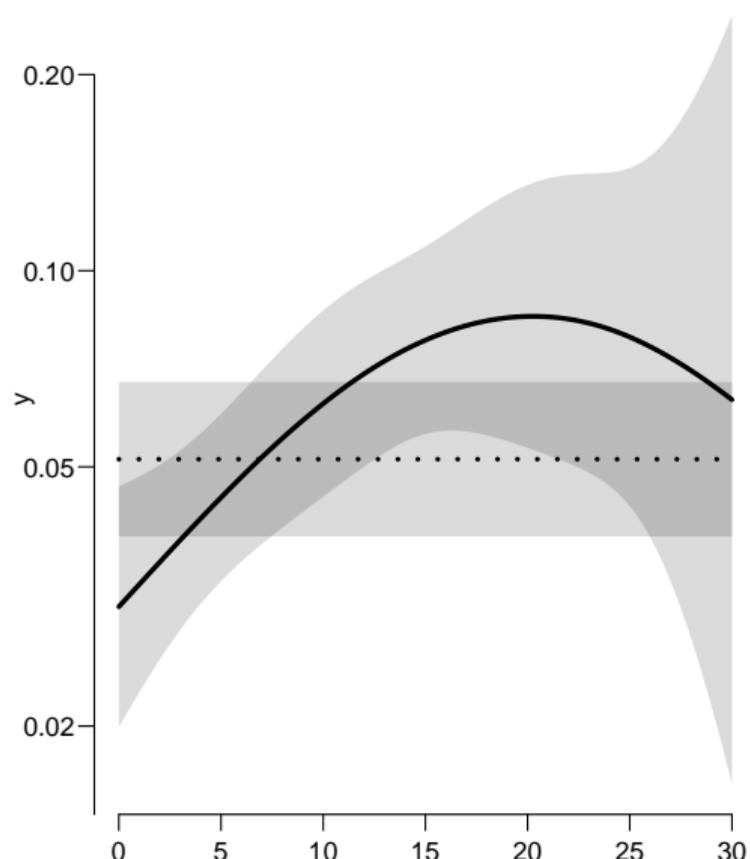
Survival functions



Hazard and survival functions

```
> par(mfrow = c(1,2), mar=c(3,3,1,1), mgp=c(3,1,0)/1.6)
> #
> # hazard scale
> matshade(prf$tfl, ci.pred(ps, prf),
+          plot = TRUE, log = "y", lwd = 3)
> matshade(prf$tfl, ci.pred(pc, prf), lty = 3, lwd = 3)
> #
> # survival
> matshade(prf$tfl, ci.surv(ps, prf, intl = 0.2),
+          plot = TRUE, ylim = 0:1, lwd = 3)
> lines(survfit(c1, newdata = data.frame(sex = "W", age = 60)),
+       col = "forestgreen", lwd = 3, conf.int = FALSE)
> lines(survfit(c1, newdata = data.frame(sex = "W", age = 60)),
+       col = "forestgreen", lwd = 1, lty = 1)
```

Hazard and survival functions

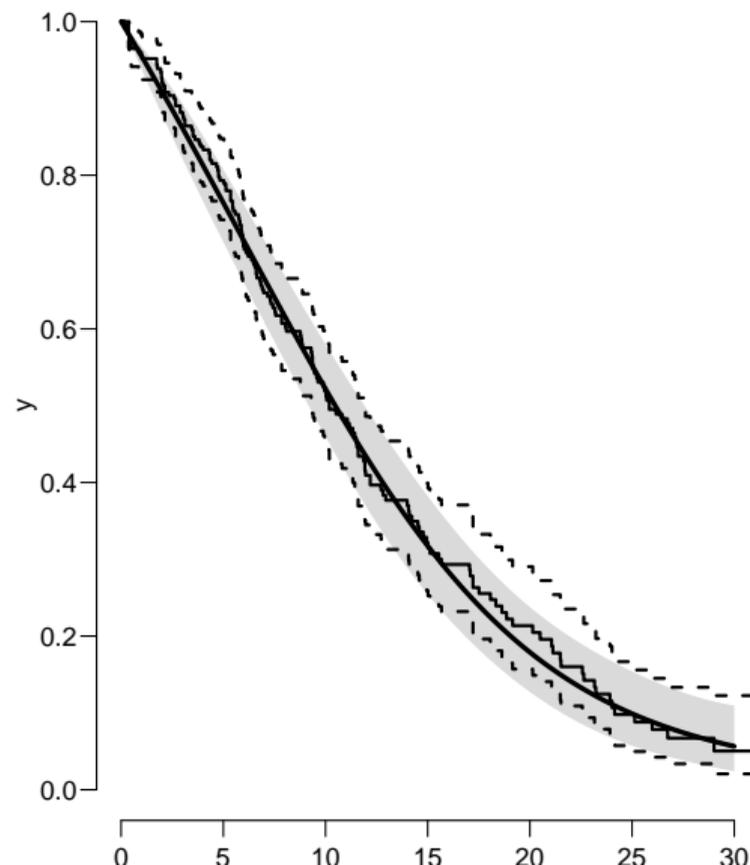
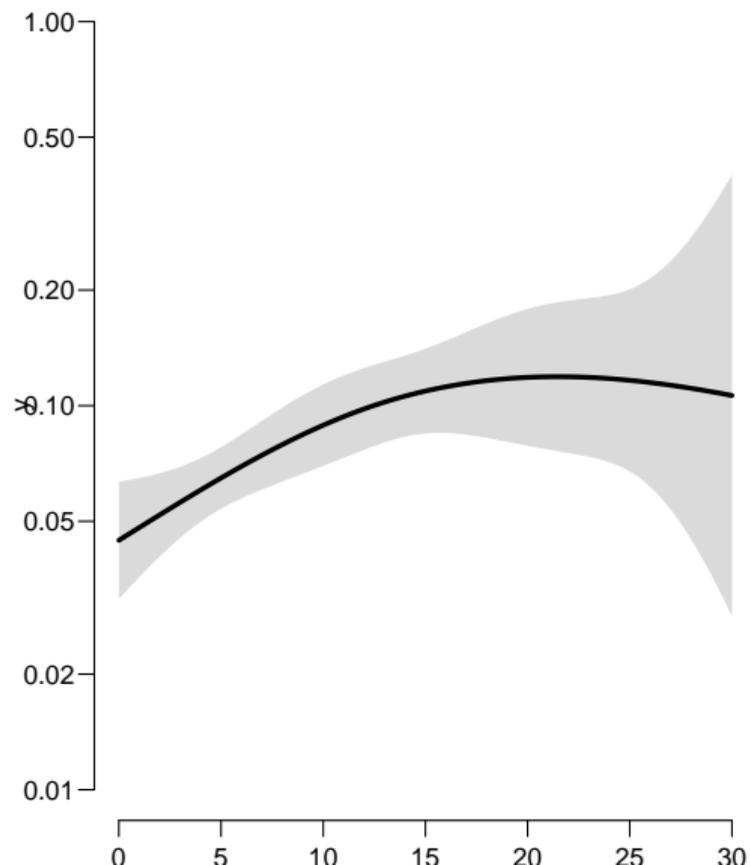


K-M estimator and smooth Poisson model

Kaplan-Meier estimator and compared to survival from corresponding Poisson-model, which is one with time (`tfl`) as the only covariate:

```
> par(mfrow=c(1,2))
> pk <- glm(cbind(lex.Xst == "Dead",
+               lex.dur) ~ Ns(tfl, knots = seq(0, 36, 12)),
+         family = poisreg,
+         data = S1)
> # hazard
> matshade(prf$tfl, ci.pred(pk, prf),
+         plot = TRUE, log = "y", lwd = 3, ylim = c(0.01,1))
> # survival from smooth model
> matshade(prf$tfl, ci.surv(pk, prf, intl = 0.2) ,
+         plot = TRUE, lwd = 3, ylim = 0:1)
> # K-M estimator
> lines(km, lwd = 2)
```

K-M estimator and smooth Poisson model

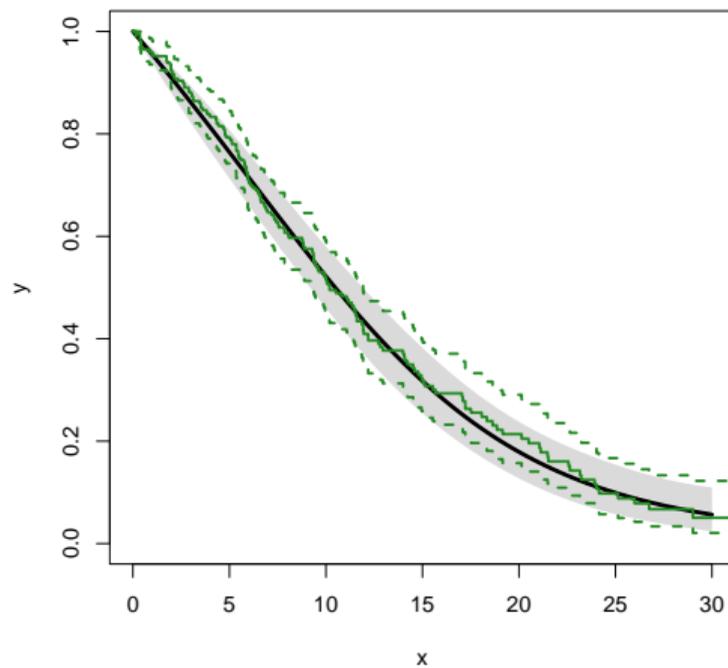
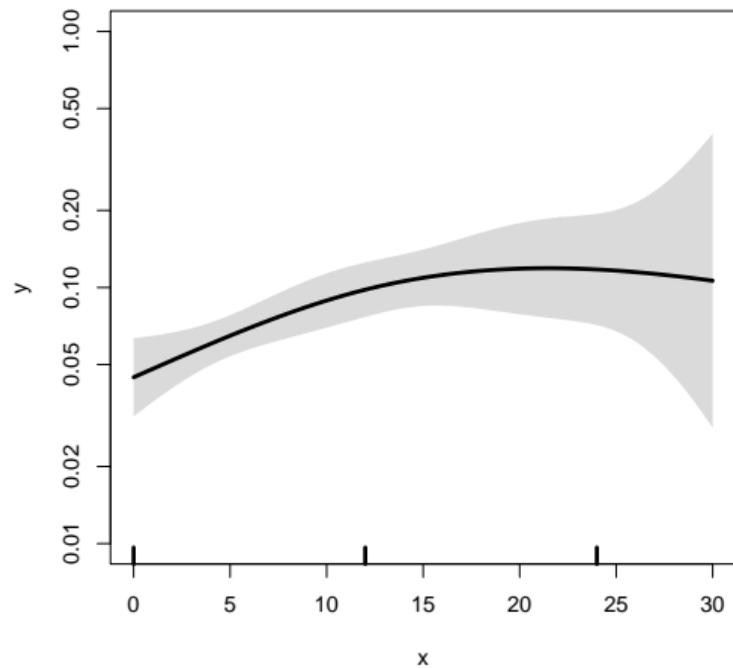


K-M estimator and smooth Poisson model

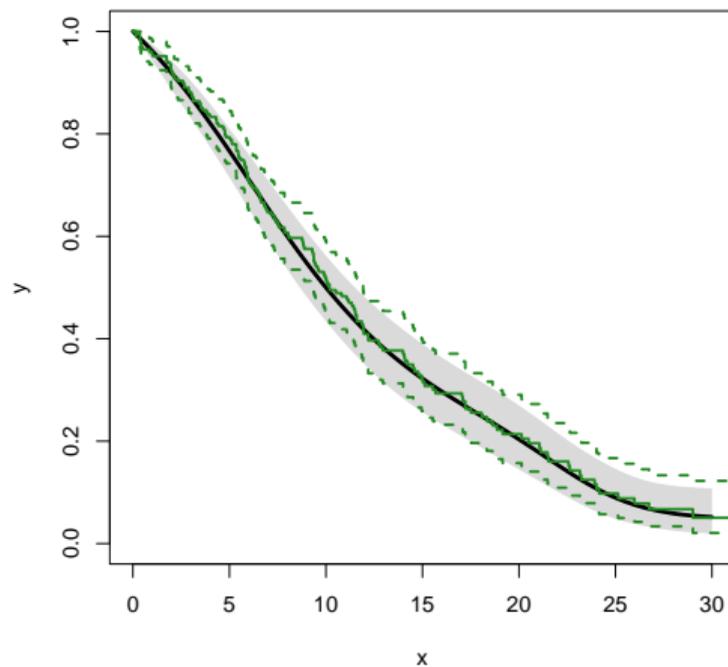
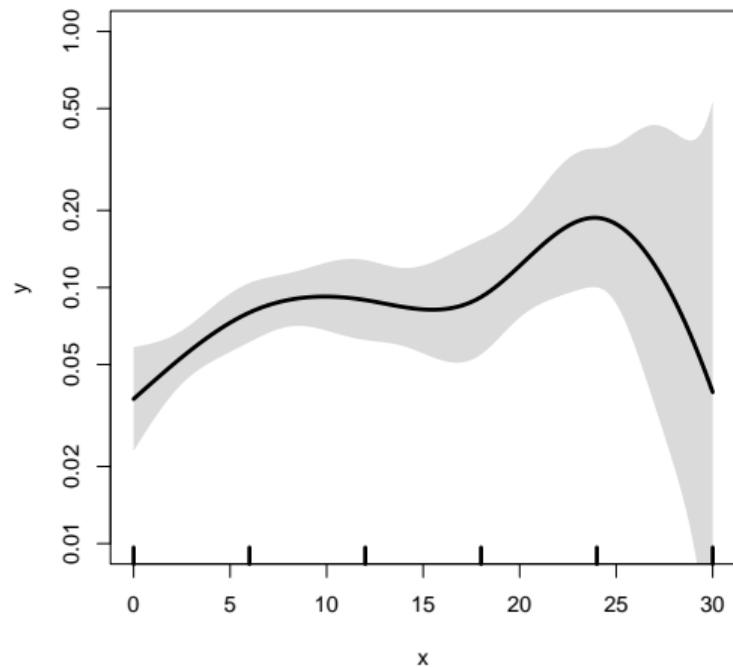
We can explore how the tightness of the knots in the smooth model influence the underlying hazard and the resulting survival function:

```
> zz <- function(dk) # distance between knots
+ {
+   par(mfrow=c(1,2))
+   kn <- seq(0, 36, dk)
+   pk <- glm(cbind(lex.Xst == "Dead",
+                   lex.dur) ~ Ns(tfl, knots = kn),
+             family = poisreg,
+             data = S1)
+   matshade(prf$tfl, ci.pred(pk, prf),
+            plot = TRUE, log = "y", lwd = 3, ylim = c(0.01,1))
+   rug(kn, lwd=3)
+
+   matshade(prf$tfl, ci.surv(pk, prf, intl = 0.2) ,
+            plot = TRUE, lwd = 3, ylim = 0:1)
+   lines(km, lwd = 2, col = "forestgreen")
+ }
> zz(12)
```

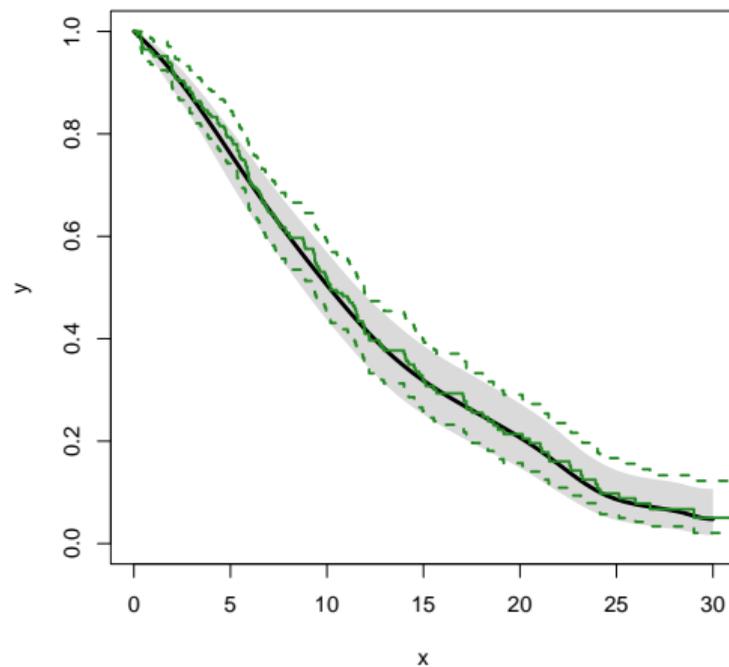
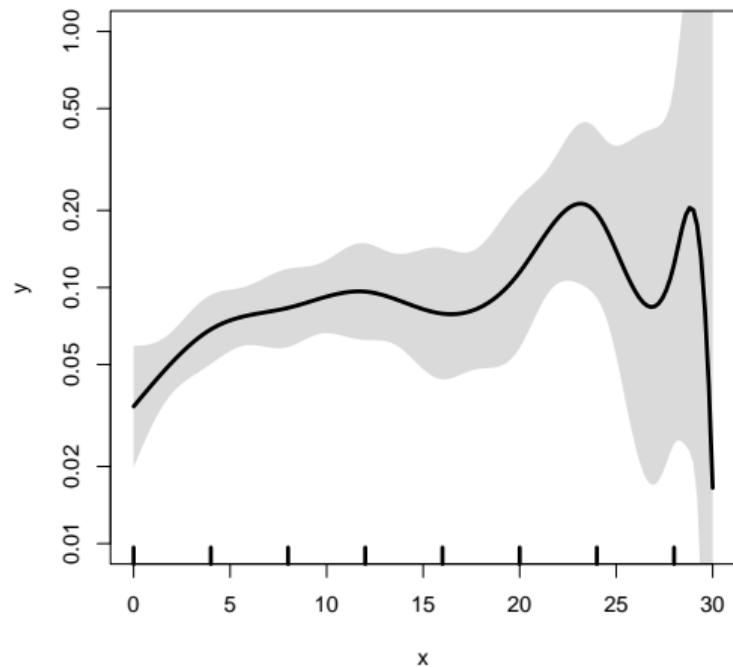
K-M estimator and smooth Poisson model



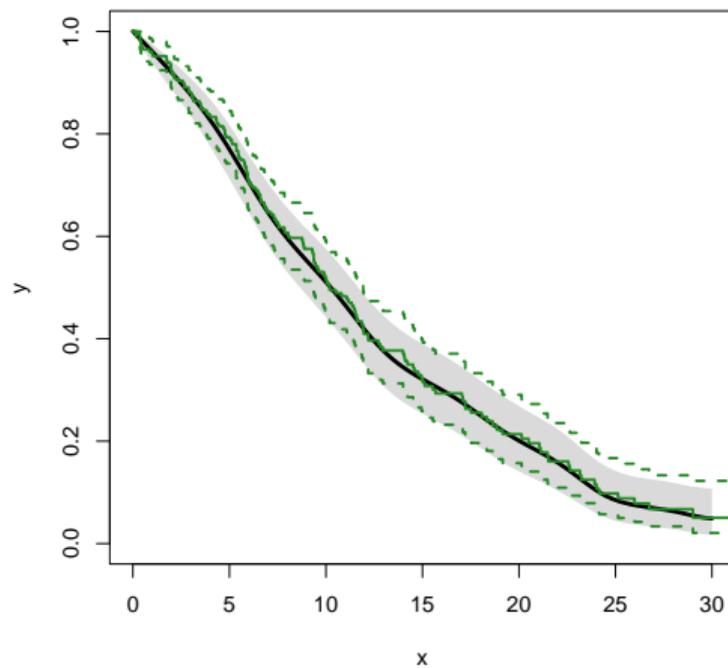
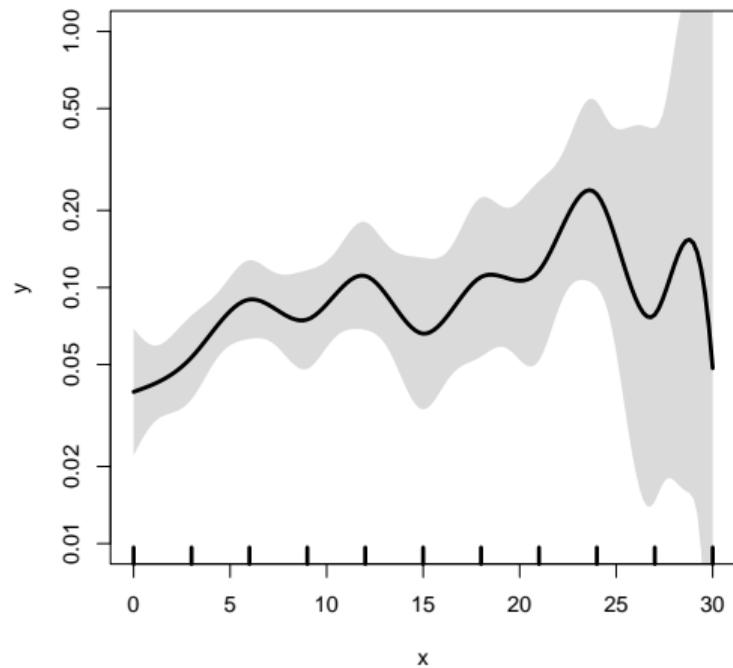
K-M estimator and smooth Poisson model



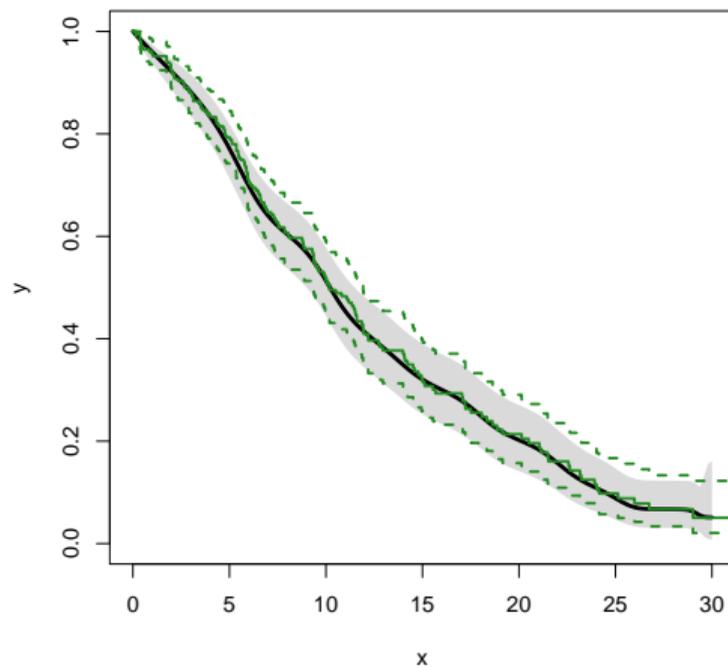
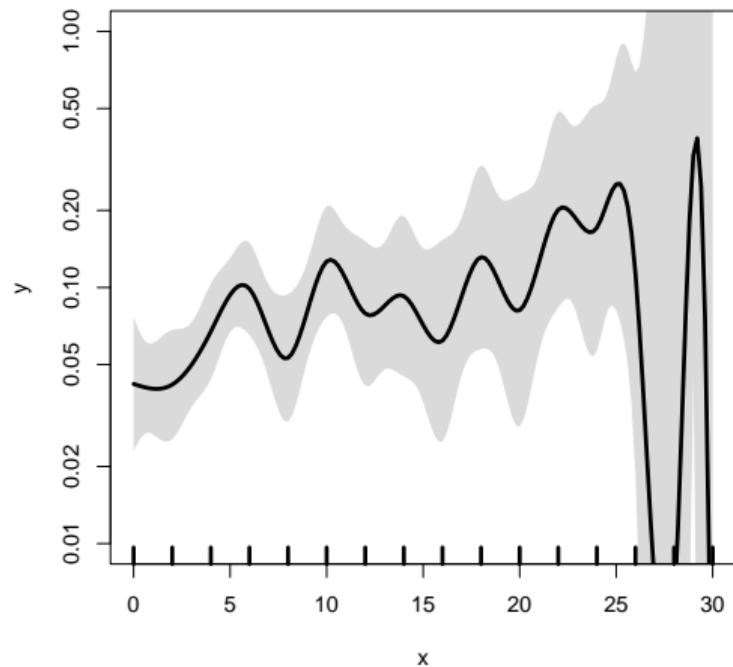
K-M estimator and smooth Poisson model



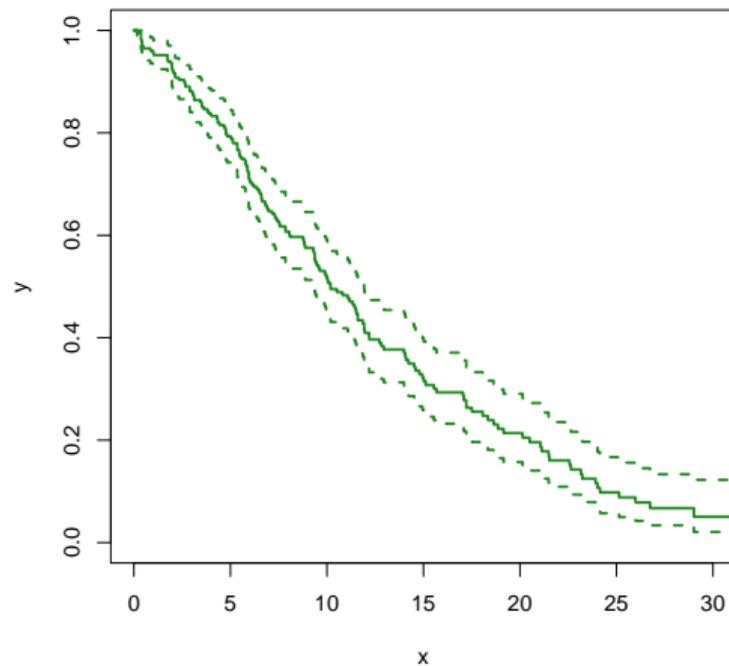
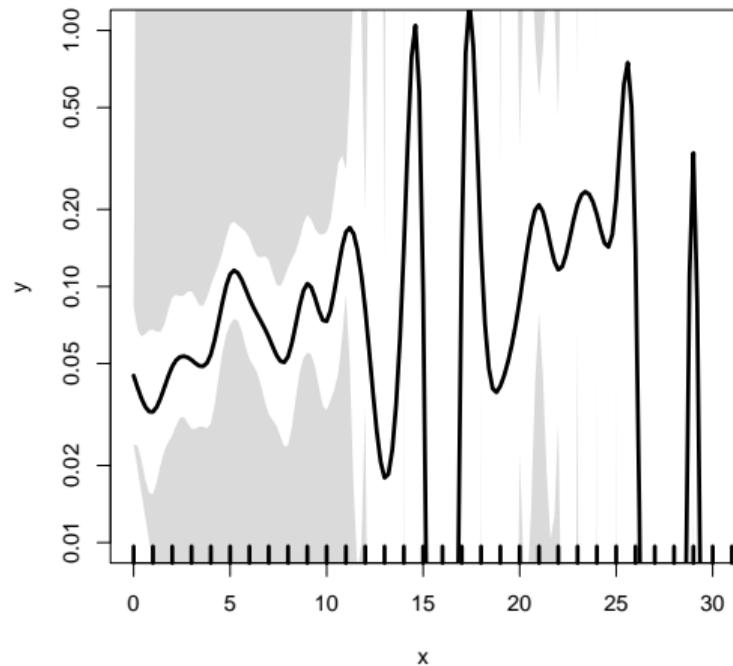
K-M estimator and smooth Poisson model



K-M estimator and smooth Poisson model



K-M estimator and smooth Poisson model



Survival analysis summary

- ▶ 1 to 1 correspondence between
 - ▶ hazard function and starting point
 - ▶ survival function
- ▶ K-M and Cox use a very detailed baseline hazard (omits it)
- ▶ Smooth parametric hazard function more credible:
 - ▶ Define `Lexis` object
 - ▶ Split along time
 - ▶ Fit Poisson model
 - ▶ Prediction data frame
 - ▶ `ci.pred` to get baseline rates
 - ▶ `ci.surv` to get baseline survival

```
> data(lung)
> lung$sex <- factor(lung$sex, labels=c("M", "F"))
> Lx <- Lexis(exit = list(tfe=time),
+           exit.status = factor(status, labels = c("Alive", "Dead")),
+           data = lung)
> sL <- splitMulti(Lx, tfe=seq(0, 1200, 10))
```

Smooth parametric hazard function

```
> m0 <- glm.Lexis(sL, ~ Ns(tfe, knots = seq(0, 1000, 200)) + sex + age)
```

Prediction data frame

```
> nd <- data.frame(tfe = seq(0, 900, 20) + 10, sex = "M", age = 65)
```

Predictions

```
> rate <- ci.pred(m0, nd) * 365.25 # per year, not per day
> surv <- ci.surv(m0, nd, int = 20)
```

Plot the rates

```
> matshade(nd$tfe, rate, log = "y", plot = TRUE)
```

Plot the survival function

```
> matshade(nd$tfe - 10, surv, ylim = c(0, 1), plot = TRUE)
```

Competing risks

estimation

Multistate models:

Occurrence rates, cumulative risks, competing risks,
state probabilities with multiple states and time scales using **R** and `Epi::Lexis`
Baker HDI, 22-23 February 2023

<http://bendixcarstensen.com/AdvCoh/courses/Melb-2023>

cmpr

```
> library(survival)
> library(Epi)
> library(popEpi)
> # popEpi::splitMulti returns a data.frame rather than a data.table
> options("popEpi.datatable" = FALSE)
> library(tidyverse)
> clear()
```

```
> data(DMlate)
> # str(DMlate)
> set.seed(1952)
> DMlate <- DMlate[sample(1:nrow(DMlate), 2000),]
> str(DMlate)
```

```
'data.frame':      2000 obs. of  7 variables:
 $ sex   : Factor w/ 2 levels "M","F": 2 1 2 1 1 1 1 1 1 1 ...
 $ dobth: num  1964 1944 1957 1952 1952 ...
 $ dodm  : num  2003 2006 2008 2007 2003 ...
 $ dodth: num  NA ...
 $ dooad: num  NA 2006 NA 2007 2006 ...
 $ doins: num  NA NA NA 2008 NA ...
 $ dox   : num  2010 2010 2010 2010 2010 ...
```

```
> head(DMlate)
```

Lexis object from DM to Death

```
> Ldm <- Lexis(entry = list(per = dodm,  
+                          age = dodm - dobth,  
+                          tfd = 0),  
+             exit = list(per = dox),  
+             exit.status = factor(!is.na(dodth),  
+                                 labels = c("DM", "Dead")),  
+             data = DMLate)
```

NOTE: entry.status has been set to "DM" for all.

NOTE: Dropping 1 rows with duration of follow up < tol

```
> summary(Ldm)
```

Transitions:

To

From	DM	Dead	Records:	Events:	Risk time:	Persons:
DM	1521	478	1999	478	10742.34	1999

Cut follow-up at the date of Ins

```
> Ldm <- sortLexis(Ldm)
> Cdm <- cutLexis(Ldm,
+               cut = Ldm$doins,
+               timescale = "per",
+               new.state = "Ins")
> summary(Cdm)
```

Transitions:

To

From	DM	Ins	Dead	Records:	Events:	Risk time:	Persons:
DM	1258	330	398	1986	728	9015.5	1986
Ins	0	263	80	343	80	1726.8	343
Sum	1258	593	478	2329	808	10742.3	1999

Cut follow-up at the date of Ins, doins

```
> subset(Ldm, lex.id %in% c(2,3,4,34))[,c(1:7,13)]
```

lex.id	per	age	tfd	lex.dur	lex.Cst	lex.Xst	doins
2	2005.6	61.52	0	4.35	DM	DM	NA
3	2007.9	51.10	0	2.11	DM	DM	NA
4	2007.0	54.61	0	3.03	DM	DM	2008.0
34	2002.8	69.65	0	4.01	DM	Dead	2002.9

```
> subset(Cdm, lex.id %in% c(2,3,4,34))[,c(1:7,13)]
```

lex.id	per	age	tfd	lex.dur	lex.Cst	lex.Xst	doins
2	2005.6	61.52	0.00	4.35	DM	DM	NA
3	2007.9	51.10	0.00	2.11	DM	DM	NA
4	2007.0	54.61	0.00	1.06	DM	Ins	2008.0
4	2008.0	55.67	1.06	1.97	Ins	Ins	2008.0
34	2002.8	69.65	0.00	0.07	DM	Ins	2002.9
34	2002.9	69.72	0.07	3.94	Ins	Dead	2002.9

Restrict to those alive in DM

```
> Adm <- subset(Cdm, lex.Cst == "DM")  
> summary(Adm)
```

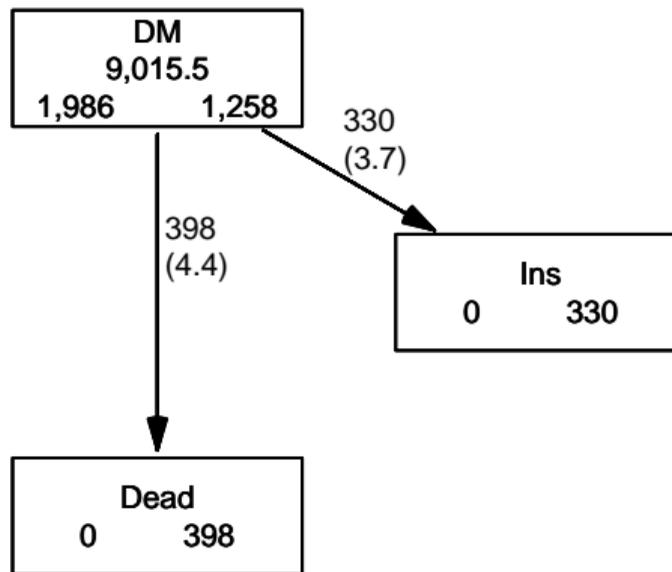
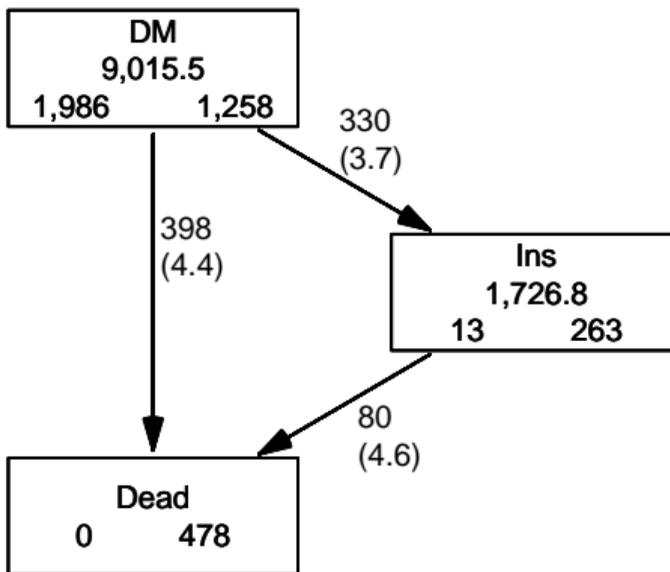
Transitions:

To

From	DM	Ins	Dead	Records:	Events:	Risk time:	Persons:
DM	1258	330	398	1986	728	9015.5	1986

```
> par(mfrow=c(1,2))  
> boxes(Cdm, boxpos = TRUE, scale.R = 100, show.BE = TRUE)  
> boxes(Adm, boxpos = TRUE, scale.R = 100, show.BE = TRUE)
```

Transitions in Cdm and Adm



Survival function?

$$S(t) = \exp \left(- \int_0^t \lambda(u) + \mu(u) \, du \right)$$

$$S(t) = \exp \left(- \int_0^t \lambda(u) \, du \right)$$

$$S(t) = \exp \left(- \int_0^t \mu(u) \, du \right)$$

Survival function?

- ▶ Regarding either Dead or Ins as censorings — or neither?
- ▶ **Simple survival**: what is the probability of being in each of the states Alive and Dead
—depends on **one** rate, Alive \rightarrow Dead
- ▶ **Competing risks**: the probability of being in each of the states DM, Ins and Dead
—depends on **two** rates, DM \rightarrow Ins and DM \rightarrow Dead

Survival function and Cumulative risk function

`survfit` does the trick; the requirements are:

1. (start, stop, event) arguments to `Surv`
2. the third argument to the `Surv` function is a factor
3. an `id` argument is given, pointing to an id variable that links together records belonging to the same person.
4. the initial state (DM) must be the first level of the factor `lex.Xst`

Survival function and Cumulative risk function

```
> levels(Adm$lex.Xst)
[1] "DM"  "Ins"  "Dead"

> m3 <- survfit(Surv(tfd, tfd + lex.dur, lex.Xst) ~ 1,
+              id = lex.id,
+              data = Adm)
> # names(m3)
> m3$states
[1] "(s0)" "Ins"  "Dead"

> head(cbind(time = m3$time, m3$pstate))
      time
[1,] 0.0054757 0.99950 0.0000000 0.00050352
[2,] 0.0082136 0.99748 0.0010070 0.00151057
[3,] 0.0109514 0.99547 0.0025184 0.00201435
[4,] 0.0136893 0.99396 0.0040297 0.00201435
[5,] 0.0164271 0.99295 0.0050373 0.00201435
[6,] 0.0191650 0.98942 0.0085637 0.00201435
```

—this is called the Aalen-Johansen estimator of state probabilities

Survival function and cumulative risks—formulae

$$S(t) = \exp\left(-\int_0^t \lambda(u) + \mu(u) \, du\right)$$

$$R_{\text{Dead}}(t) = \int_0^t \mu(u) S(u) \, du$$

$$\begin{aligned} R_{\text{Ins}}(t) &= \int_0^t \lambda(u) S(u) \, du \\ &= \int_0^t \lambda(u) \exp\left(-\int_0^u \lambda(s) + \mu(s) \, ds\right) \, du \end{aligned}$$

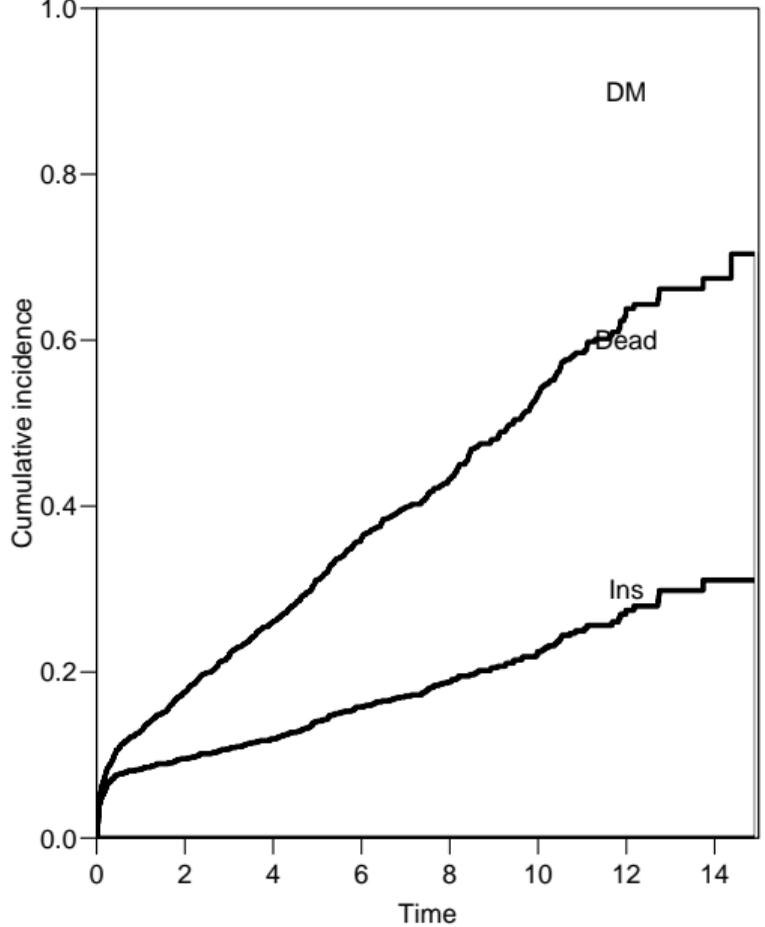
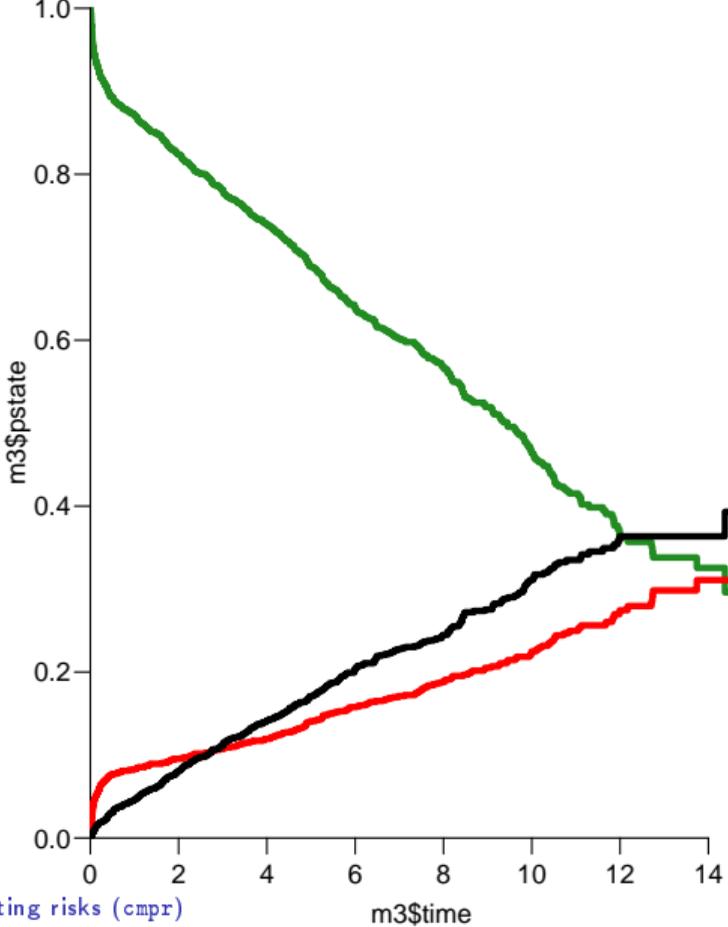
$$S(t) + R_{\text{Ins}}(t) + R_{\text{Dead}}(t) = 1, \quad \forall t$$

Survival function and cumulative risks

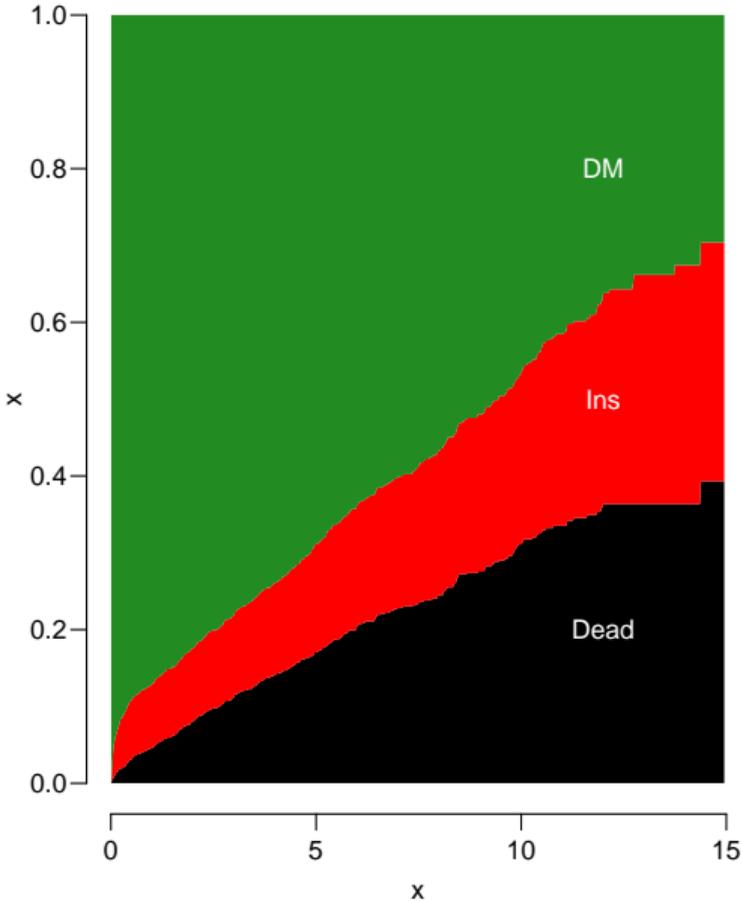
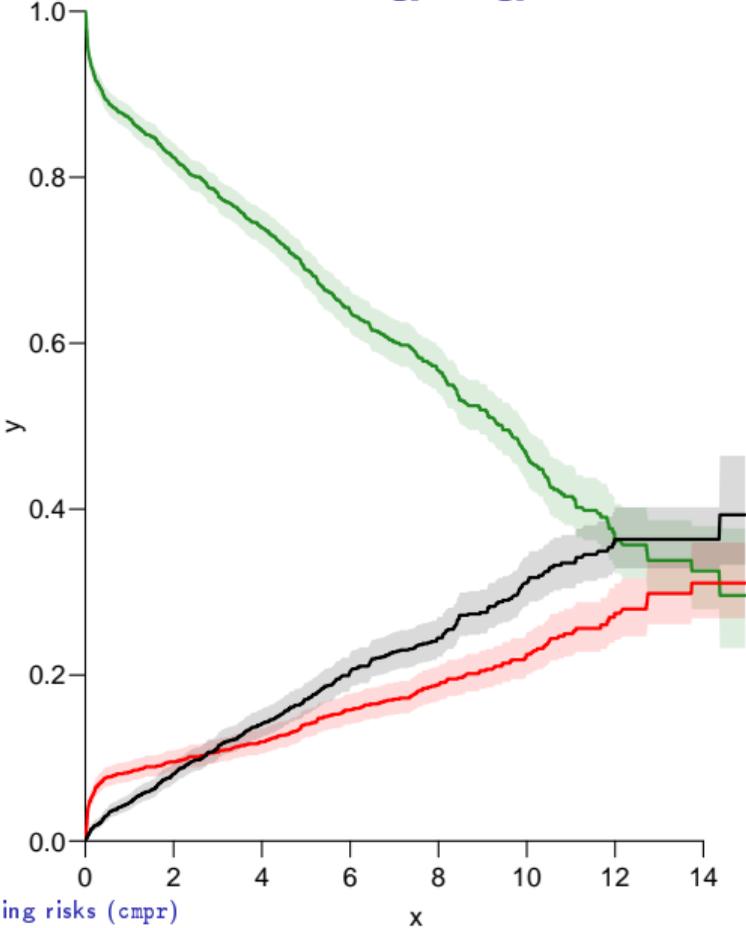
```
> par( mfrow=c(1,2) )
> matplot(m3$time, m3$pstate,
+         type="s", lty=1, lwd=4,
+         col=c("ForestGreen", "red", "black"),
+         xlim=c(0,15), xaxs="i",
+         ylim=c(0,1), yaxs="i" )
> stackedCIF(m3, lwd=3, xlim=c(0,15), xaxs="i", yaxs="i" )
> text(rep(12,3), c(0.9,0.3,0.6), levels(Cdm))
> box(bty="o")

> par(mfrow = c(1, 2))
> matshade(m3$time, cbind(m3$pstate,
+                         m3$lower,
+                         m3$upper)[, c(1, 4, 7, 2, 5, 8, 3, 6, 9)],
+         plot = TRUE, lty = 1, lwd = 2,
+         col = clr <- c("ForestGreen", "red", "black"),
+         xlim=c(0,15), xaxs="i",
+         ylim = c(0,1), yaxs = "i")
> mat2pol(m3$pstate, perm = 3:1, x = m3$time, col = clr[3:1])
> text(rep(12, 3), c(0.8, 0.5, 0.2), levels(Cdm), col = "white")
```

Survival and cumulative risk functions



Survival and cumulative risk functions



Survival function and cumulative risks—don't

$$S(t) = \exp\left(-\int_0^t \lambda(u) + \mu(u) du\right)$$

$$R_{\text{Dead}}(t) = \int_0^t \mu(u)S(u) du$$

$$R_{\text{Ins}}(t) = \int_0^t \lambda(u)S(u) du$$

$$= \int_0^t \lambda(u) \exp\left(-\int_0^u \lambda(s) + \mu(s) ds\right) du$$

$$\neq \int_0^t \lambda(u) \exp\left(-\int_0^u \lambda(s) ds\right) du$$

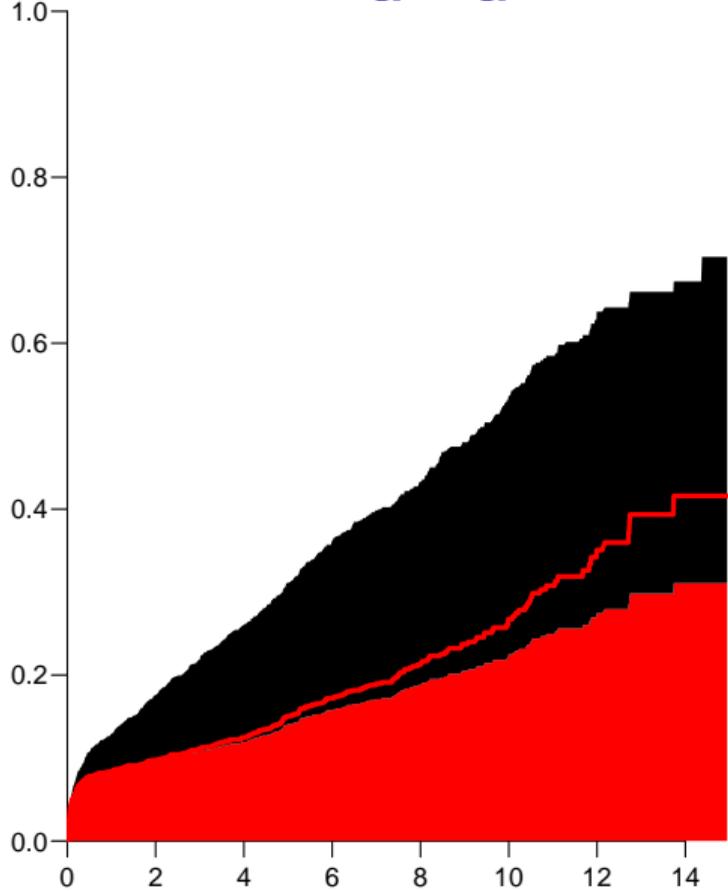
$$= 1 - \exp\left(-\int_0^t \lambda(s) ds\right) \text{ — nice formula, but wrong!}$$

Probability of Ins **assuming** Dead does not exist **and** rate of Ins unchanged!

Survival function and cumulative risks—don't

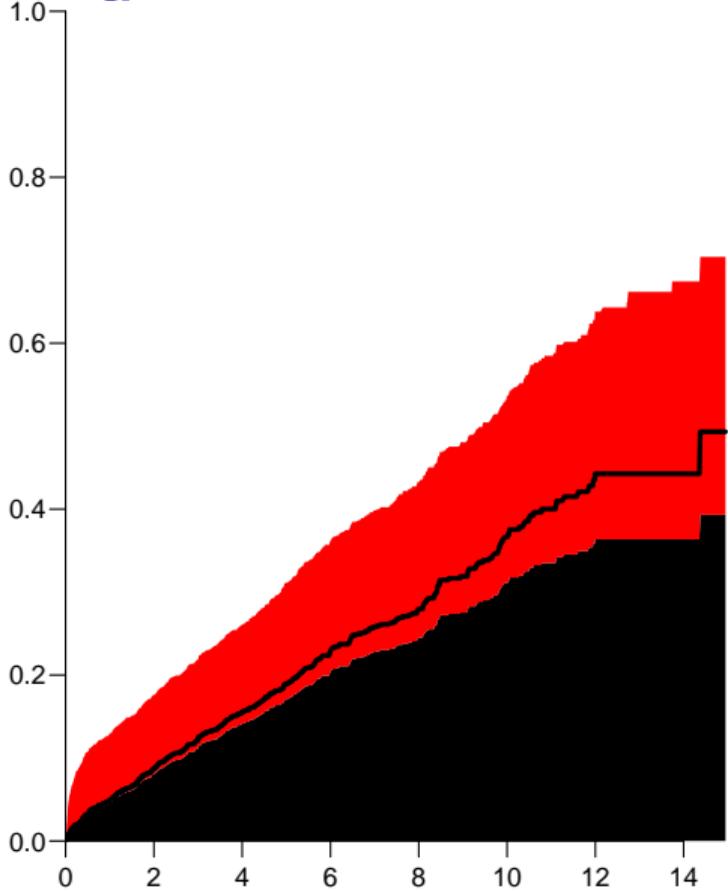
```
> m2 <- survfit(Surv(tfd,
+                 tfd + lex.dur,
+                 lex.Xst == "Ins" ) ~ 1,
+                 data = Adm)
> M2 <- survfit(Surv(tfd,
+                 tfd + lex.dur,
+                 lex.Xst == "Dead") ~ 1,
+                 data = Adm)
> par(mfrow = c(1,2))
> mat2pol(m3$pstate, c(2,3,1), x = m3$time,
+         col = c("red", "black", "transparent"),
+         xlim=c(0,15), xaxs="i",
+         yaxs = "i", xlab = "time since DM", ylab = "" )
> lines(m2$time, 1 - m2$surv, lwd = 3, col = "red" )
> mat2pol(m3$pstate, c(3,2,1), x = m3$time, yaxs = "i",
+         col = c("black","red","transparent"),
+         xlim=c(0,15), xaxs="i",
+         yaxs = "i", xlab = "time since DM", ylab = "" )
> lines(M2$time, 1 - M2$surv, lwd = 3, col = "black" )
```

Survival and cumulative risk functions



Competing risks (cmpr)

time since DM



time since DM

Cause-specific rates

- ▶ There is nothing wrong with modeling the cause-specific event-rates, the problem lies in how you transform them into probabilities.
- ▶ The relevant model for a competing risks situation normally consists of separate models for each of the cause-specific rates.
- ▶ These models have no common parameters (effects of time or other covariates are not constrained to be the same).
- ▶ ... not for technical or statistical reasons, but for **substantial** reasons:
it is unlikely that rates of different types of event (Insulin initiation and death, say) depend on time in the same way.

Cause-specific rates

```
> Sdm <- splitMulti(Adm, tfd = seq(0, 20, 0.1))  
> summary(Adm)
```

Transitions:

To

From	DM	Ins	Dead	Records:	Events:	Risk time:	Persons:
DM	1258	330	398	1986	728	9015.5	1986

```
> summary(Sdm)
```

Transitions:

To

From	DM	Ins	Dead	Records:	Events:	Risk time:	Persons:
DM	90419	330	398	91147	728	9015.5	1986

Cause-specific rates

```
> round(cbind(  
+ with(subset(Sdm, lex.Xst == "Ins" ), quantile(tfd + lex.dur, 0:4/4)),  
+ with(subset(Sdm, lex.Xst == "Dead"), quantile(tfd + lex.dur, 0:4/4))), 2)
```

	[,1]	[,2]
0%	0.01	0.01
25%	0.07	1.15
50%	1.07	3.01
75%	5.19	5.69
100%	13.74	14.38

```
> ikn <- c(0, 0.5, 3, 10)  
> dkn <- c(0, 2.0, 5, 9)  
> Ins.glm <- glm.Lexis(Sdm, ~ Ns(tfd, knots = ikn), to = "Ins" )
```

```
stats::glm Poisson analysis of Lexis object Sdm with log link:  
Rates for the transition:  
DM->Ins
```

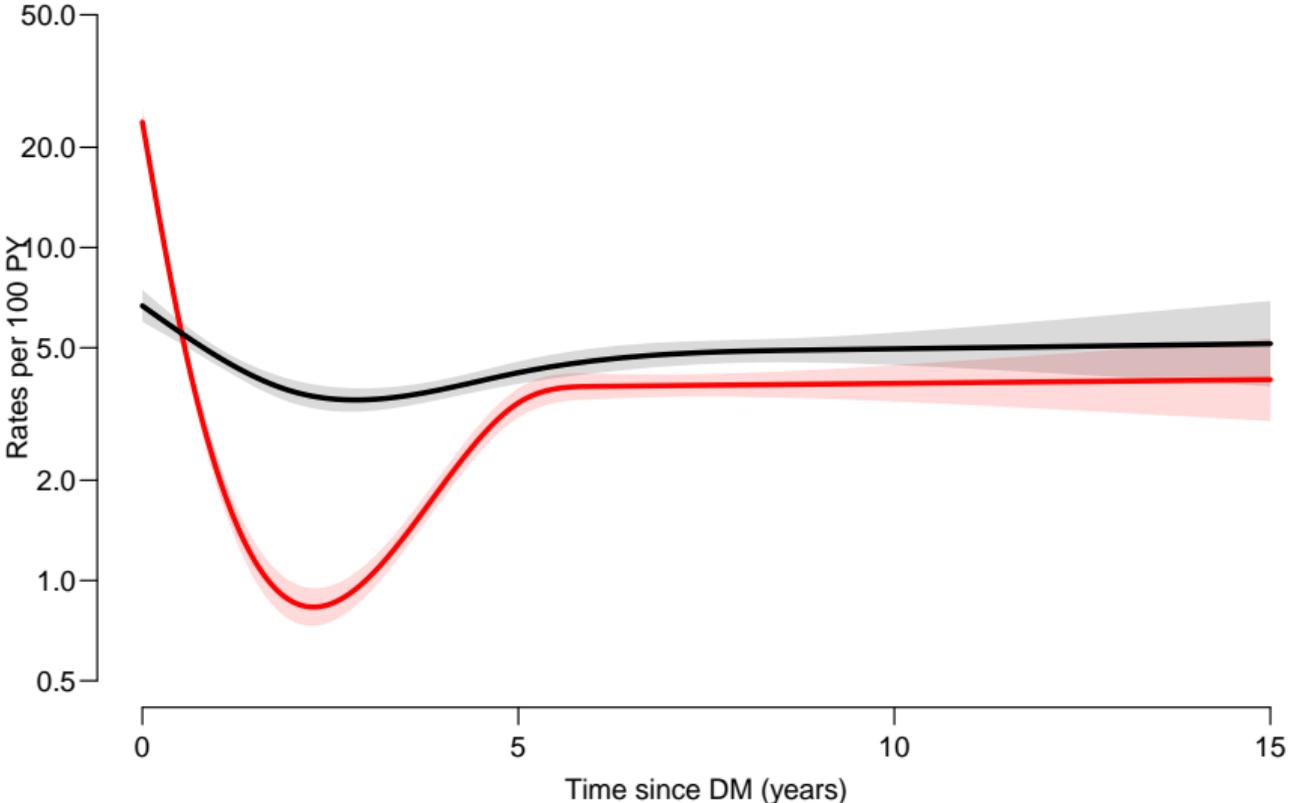
```
> Dead.glm <- glm.Lexis(Sdm, ~ Ns(tfd, knots = dkn), to = "Dead")
```

```
stats::glm Poisson analysis of Lexis object Sdm with log link:  
Rates for the transition:  
DM->Dead
```

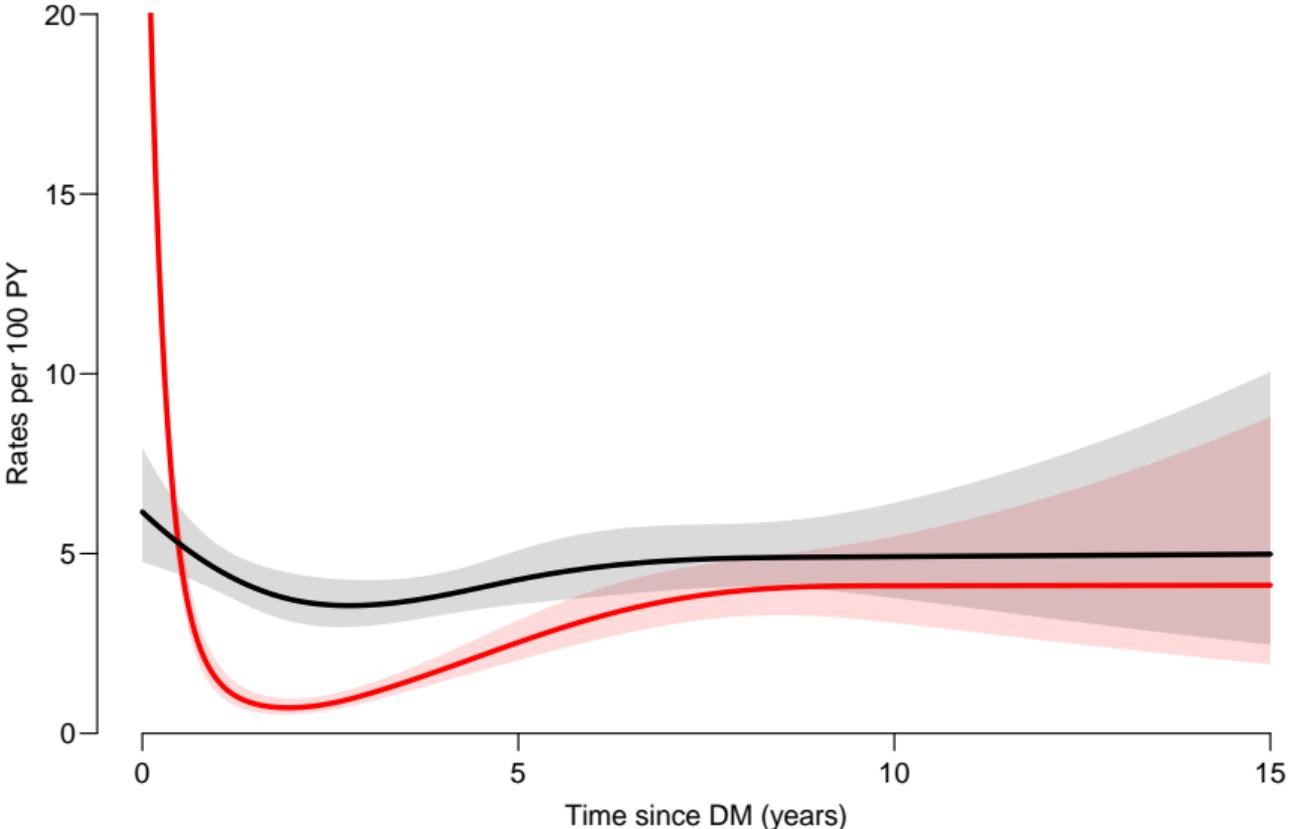
Cause-specific rates

```
> int <- 0.01
> nd <- data.frame(tfd = seq(0, 15, int))
> l.glm <- ci.pred( Ins.glm, nd)
> m.glm <- ci.pred(Dead.glm, nd)
> matshade(nd$tfd,
+          cbind(l.glm, m.glm) * 100,
+          plot = TRUE,
+          yaxs="i", ylim = c(0, 20),
+          # log = "y", ylim = c(2, 20),
+          col = rep(c("red","black"), 2), lwd = 3,
+          xlab = "Time since DM (years)",
+          ylab = "Rates per 100 PY")
```

Survival and cumulative risk functions



Survival and cumulative risk functions



Integrals with R

- ▶ Integrals look scary to many people, but they are really just areas under curves.
- ▶ In R, a curve of the function $\mu(t)$ is a set of two vectors: one vector of ts and one vector $y = \mu(t)s$.
- ▶ When we have a model such as the `glm` above that estimates the mortality as a function of time (`tfd`), we can get the mortality as a function of time by first choosing the timepoints, say from 0 to 15 years in steps of 0.01 year (≈ 4 days)
- ▶ Using `ci.pred` on this gives the predicted rates
- ▶ Then use the formulae with all the integrals to get the state probabilities.

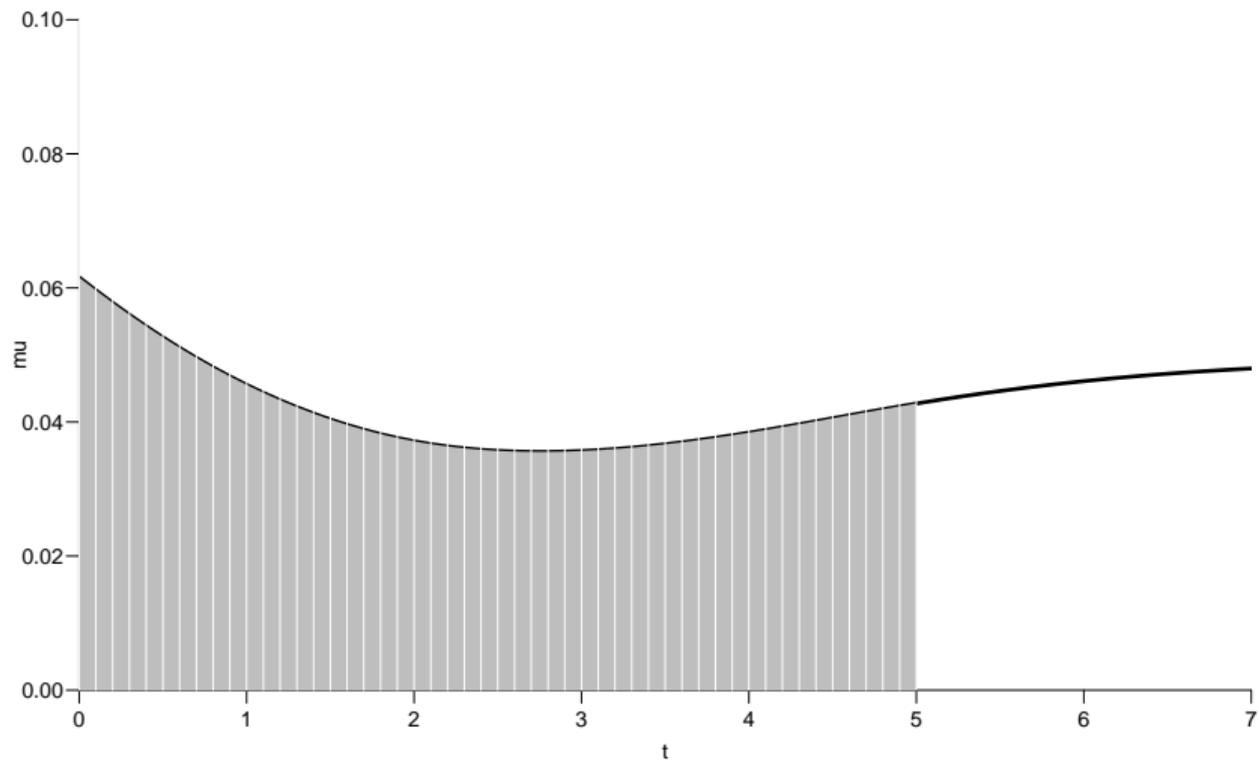
Integrals with R

```
> t <- seq(0, 15, 0.01)
> nd <- data.frame(tfd = t)
> mu <- ci.pred(Dead.glm, nd)[,1]
> head(cbind(t, mu))
```

```
      t      mu
1 0.00 0.061567
2 0.01 0.061372
3 0.02 0.061177
4 0.03 0.060983
5 0.04 0.060790
6 0.05 0.060597
```

```
> plot(t, mu, type="l", lwd = 3,
+       xlim = c(0, 7), xaxs = "i",
+       ylim = c(0, 0.1), yaxs = "i")
> polygon(t[c(1:501,501:1)], c(mu[1:501], rep(0, 501)),
+         col = "gray", border = "transparent")
> abline(v=0:50/10, col="white")
```

Integrals with R



Numerical integration with R

```
> mid <- function(x) x[-1] - diff(x) / 2
> (x <- c(1:5, 7, 10))
[1] 1 2 3 4 5 7 10
> mid(x)
[1] 1.5 2.5 3.5 4.5 6.0 8.5
```

`mid(x)` is a vector that is 1 shorter than the vector `x`, just as `diff(x)` is.

So if we want the integral over the period 0 to 5 years, we want the sum over the first 500 intervals, corresponding to the first 501 interval endpoints:

```
> cbind(diff(t), mid(mu))[1:5,]
  [,1] [,2]
2 0.01 0.061470
3 0.01 0.061275
4 0.01 0.061080
5 0.01 0.060887
6 0.01 0.060694
```

Numerical integration with R

In practice we will want the integral **function** of μ , so for every t we want $M(t) = \int_0^t \mu(s) d(s)$. This is easily accomplished by the function `cumsum`:

```
> Mu <- c(0, cumsum(diff(t) * mid(mu)))  
> head(cbind(t, Mu))
```

	t	Mu
	0.00	0.0000000
2	0.01	0.0006147
3	0.02	0.0012274
4	0.03	0.0018383
5	0.04	0.0024471
6	0.05	0.0030541

Note the first value which is the integral from 0 to 0, so by definition 0.

Cumulative risks from parametric models

If we have estimates of λ and μ as functions of time, we can derive the cumulative risks.

In practice this will be by numerical integration; compute the rates at closely spaced intervals and evaluate the integrals as sums. This is easy.

What is not so easy is to come up with confidence intervals for the cumulative risks.

Simulation of cumulative risks: `ci.Crisk`

1. a random vector from the multivariate normal distribution with
 - ▶ mean equal to the parameters of the model,
 - ▶ variance-covariance equal to the estimated variance-covariance of the parameter estimates
2. use this to generate a simulated set of rates $(\lambda(t), \mu(t))$, evaluated at closely spaced times
3. derive state probabilities at these times by numerical integration
4. repeat to obtain, say, 1000 sets of state probabilities at these times
5. derive confidence intervals for the state probabilities as the 2.5 and 97.5 percentiles of the state probabilities at each time

This machinery is implemented in the function `ci.Crisk` in `Epi`

Cumulative risks from parametric models

```
> cR <- ci.Crisk(mods = list(Ins = Ins.glm,  
+                           Dead = Dead.glm),  
+               nd = nd)
```

NOTE: Times are assumed to be in the column `tfd` at equal distances of 0.01

```
> str(cR)
```

List of 4

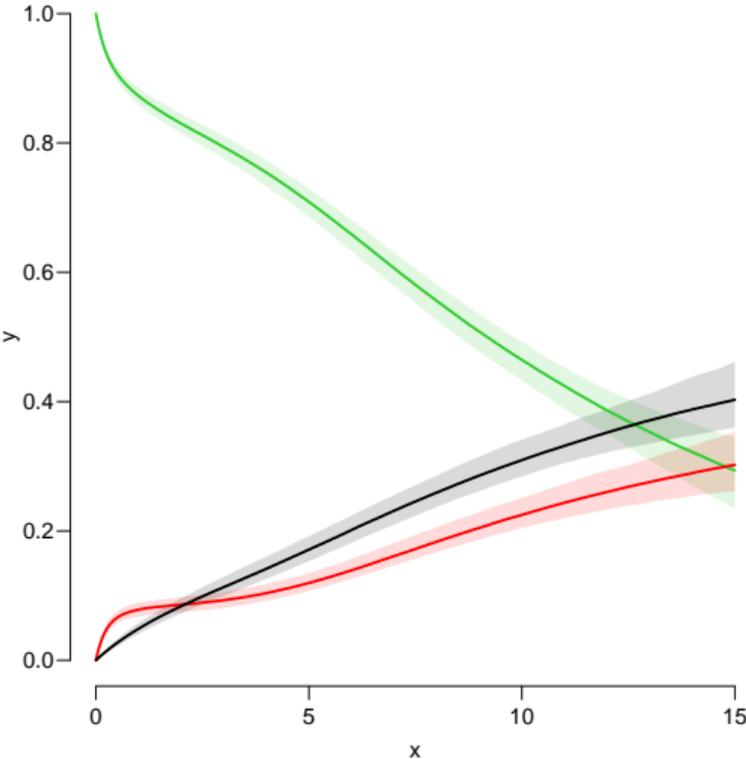
```
$ Crisk: num [1:1501, 1:3, 1:3] 1 0.996 0.993 0.989 0.986 ...  
  ..- attr(*, "dimnames")=List of 3  
  .. ..$ tfd   : chr [1:1501] "0" "0.01" "0.02" "0.03" ...  
  .. ..$ cause: chr [1:3] "Surv" "Ins" "Dead"  
  .. ..$      : chr [1:3] "50%" "2.5%" "97.5%"  
$ Srisk: num [1:1501, 1:2, 1:3] 0 0.000618 0.001232 0.001841 0.002447 ...  
  ..- attr(*, "dimnames")=List of 3  
  .. ..$ tfd   : chr [1:1501] "0" "0.01" "0.02" "0.03" ...  
  .. ..$ cause: chr [1:2] "Dead" "Dead+Ins"  
  .. ..$      : chr [1:3] "50%" "2.5%" "97.5%"  
$ Stime: num [1:1501, 1:3, 1:3] 0 0.00998 0.01993 0.02984 0.03972 ...  
  ..- attr(*, "dimnames")=List of 3  
  .. ..$ tfd   : chr [1:1501] "0" "0.01" "0.02" "0.03" ...  
  .. ..$ cause: chr [1:3] "Surv" "Ins" "Dead"
```

Cumulative risks from parametric models

So now plot the cumulative *risks* of being in each of the states (the **Crisk** component):

```
> matshade(as.numeric(dimnames(cR$Crisk)[[1]]),  
+         cbind(cR$Crisk[,1,],  
+             cR$Crisk[,2,],  
+             cR$Crisk[,3,]), plot = TRUE,  
+         lwd = 2, col = c("limegreen","red","black"))
```

Survival and cumulative risk functions



Stacked probabilities: (matrix 2 polygons)

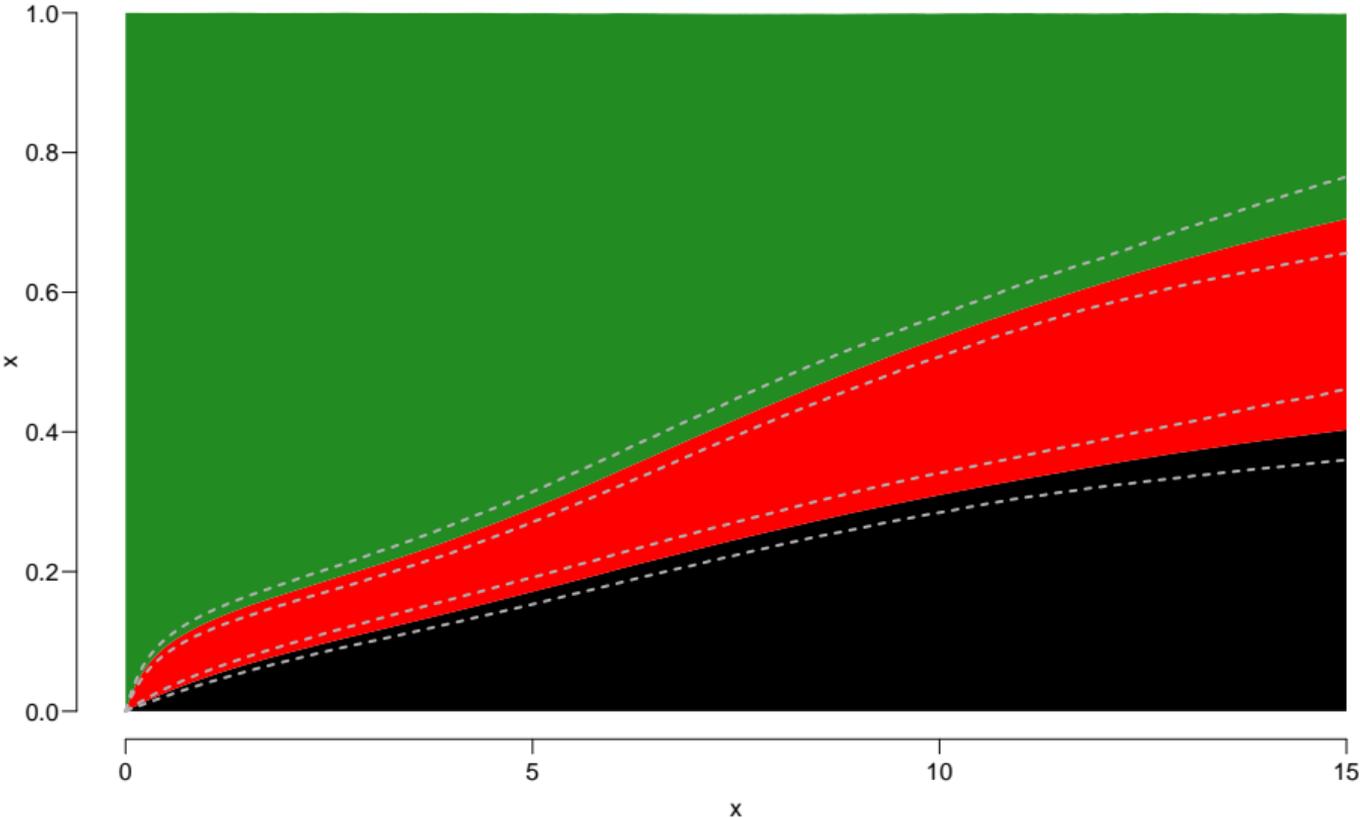
```
> mat2pol(cR$Crisk[,3:1,1], col = c("forestgreen","red","black")[3:1])
```

1st argument to `mat2pol` must be a 2-dimensional matrix, with rows representing the x -axis of the plot, and columns states.

The component `Srisk` has the confidence limits of the stacked probabilities:

```
> mat2pol(cR$Crisk[,3:1,1], col = c("forestgreen","red","black")[3:1])
> matlines(as.numeric(dimnames(cR$Srisk)[[1]]),
+         cbind(cR$Srisk[, "Dead"      ,2:3],
+         cR$Srisk[, "Dead+Ins",2:3]),
+         lty = "32", lwd = 2, col = gray(0.7))
```

Survival and cumulative risk functions



Expected life time: using simulated objects

The areas between the lines (up to say 10 years) are **expected sojourn times**, that is:

- ▶ expected years alive without Ins
- ▶ expected years lost to death without Ins
- ▶ expected years after Ins, including years dead after Ins

Not all of direct relevance; actually only the first may be so.

They are available (with simulation-based confidence intervals) in the component of `cR`, `Stime` (Sojourn time).

Expected life time: using simulated objects

A relevant quantity would be the expected time alive without Ins during the first 5, 10 and 15 years:

```
> str(cR$Stime)
num [1:1501, 1:3, 1:3] 0 0.00998 0.01993 0.02984 0.03972 ...
- attr(*, "dimnames")=List of 3
..$ tfd : chr [1:1501] "0" "0.01" "0.02" "0.03" ...
..$ cause: chr [1:3] "Surv" "Ins" "Dead"
..$      : chr [1:3] "50%" "2.5%" "97.5%"

> round(cR$Stime[c("5", "10", "15"), "Surv", ], 1)
tfd 50% 2.5% 97.5%
 5 4.1 4.0 4.2
10 7.0 6.8 7.2
15 8.9 8.5 9.2
```

Multistate model

simulation

Multistate models:

Occurrence rates, cumulative risks, competing risks,
state probabilities with multiple states and time scales using **R** and `Epi::Lexis`
Baker HDI, 22-23 February 2023

<http://bendixcarstensen.com/AdvCoh/courses/Melb-2023>

msmt

Background: Steno 2 trial

- ▶ Clinical trial for diabetes ptt. with kidney disease (micro-albuminuria)
- ▶ 80 ptt. randomised to either of
 - ▶ Conventional treatment
 - ▶ Intensified multifactorial treatment
- ▶ 1993–2001
- ▶ follow-up till 2018

Steno 2 trial: goal

- ▶ Is there a treatment effect on:
 - ▶ CVD mortality
 - ▶ non-CVD mortality
- ▶ Does the treatment effect depend on:
 - ▶ Albuminuria state
- ▶ Quantification of treatment effect:
 - ▶ Rate-ratios
 - ▶ Life times
 - ▶ Changes in clinical parameters

```

> data(steno2)
> steno2 <- cal.yr(steno2)
> steno2 <- transform(steno2,
+                       doEnd = pmin(doDth, doEnd, na.rm = TRUE))
> str(steno2)

'data.frame':      160 obs. of  14 variables:
 $ id      : num  1 2 3 4 5 6 7 8 9 10 ...
 $ allo    : Factor w/ 2 levels "Int","Conv": 1 1 2 2 2 2 2 1 1 1 ...
 $ sex     : Factor w/ 2 levels "F","M": 2 2 2 2 2 2 1 2 2 2 ...
 $ baseCVD : num  0 0 0 0 0 1 0 0 0 0 ...
 $ deathCVD: num  0 0 0 0 1 0 0 0 1 0 ...
 $ doBth   : 'cal.yr' num  1932 1947 1943 1945 1936 ...
 $ doDM    : 'cal.yr' num  1991 1982 1983 1977 1986 ...
 $ doBase  : 'cal.yr' num  1993 1993 1993 1993 1993 ...
 $ doCVD1  : 'cal.yr' num  2014 2009 2002 1995 1994 ...
 $ doCVD2  : 'cal.yr' num  NA 2009 NA 1997 1995 ...
 $ doCVD3  : 'cal.yr' num  NA 2010 NA 2003 1998 ...
 $ doESRD  : 'cal.yr' num  NaN NaN NaN NaN 1998 ...
 $ doEnd   : 'cal.yr' num  2015 2015 2002 2003 1998 ...
 $ doDth   : 'cal.yr' num  NA NA 2002 2003 1998 ...

```

A Lexis object

```
> L2 <- Lexis(entry = list(per = doBase,  
+                          age = doBase - doBth,  
+                          tfi = 0),  
+            exit = list(per = doEnd),  
+            exit.status = factor(deathCVD + !is.na(doDth),  
+                                labels=c("Mic", "D(oth)", "D(CVD)")),  
+            id = id,  
+            data = steno2)
```

NOTE: `entry.status` has been set to "Mic" for all.

Explain the coding of `exit.status`.

A Lexis object

```
> summary(L2, t = TRUE)
```

```
Transitions:
```

```
  To
```

From	Mic	D(oth)	D(CVD)	Records:	Events:	Risk time:	Persons:
Mic	67	55	38	160	93	2416.59	160

```
Timescales:
```

```
per age tfi  
"" "" ""
```

How many persons are there in the cohort?

How many deaths are there in the cohort?

How much follow-up time is there in the cohort?

How many states are there in the model (so far)?

Albuminuria status

```
> data(st2alb) ; head(st2alb, 3)
  id      doTr state
1  1 1993-06-12  Mic
2  1 1995-05-13  Norm
3  1 2000-01-26  Mic

> cut2 <- rename(cal.yr(st2alb),
+               lex.id = id,
+               cut = doTr,
+               new.state = state)
> with(cut2, addmargins(table(table(lex.id))))

  1  2  3  4  5 Sum
4  25 40 46 41 156
```

What does this table mean?

Albuminuria status as states

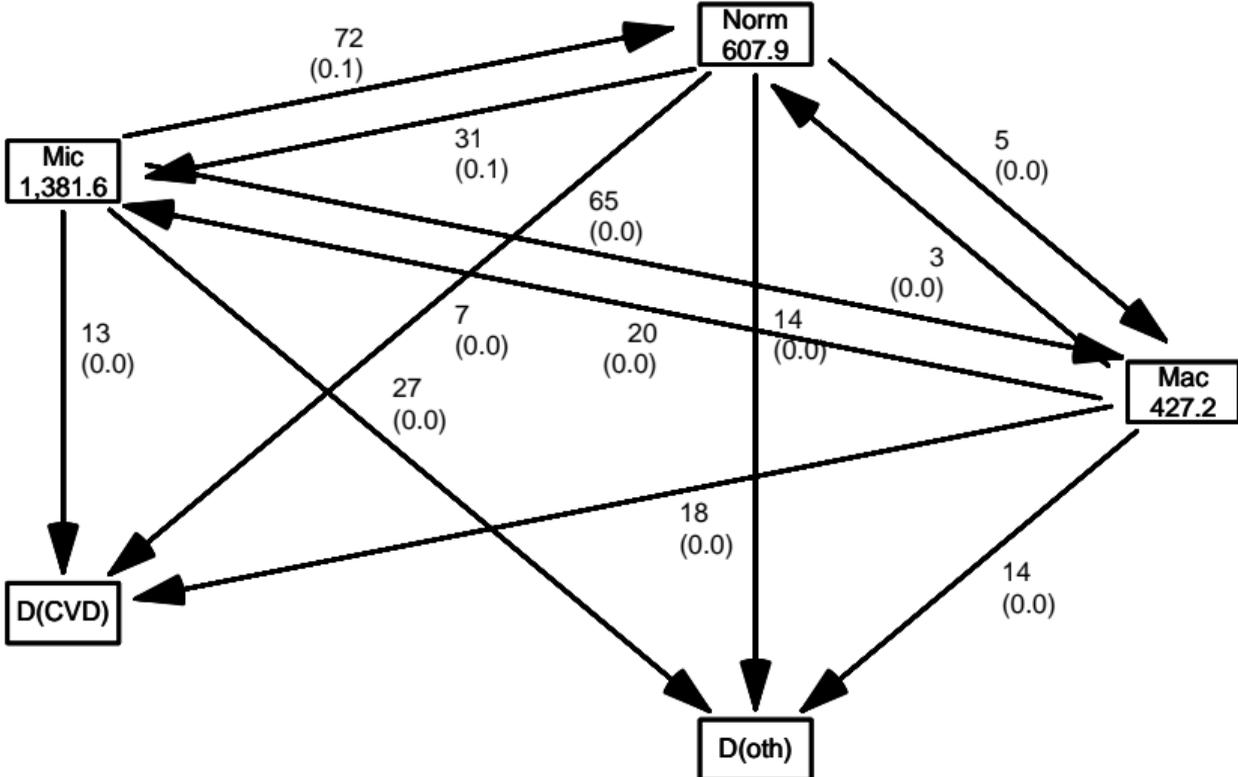
```
> L3 <- rcutLexis(L2, cut2, time = "per")  
> summary(L3)
```

Transitions:

	To								
From	Mic	Norm	Mac	D(oth)	D(CVD)	Records:	Events:	Risk time:	Persons:
Mic	299	72	65	27	13	476	177	1381.57	160
Norm	31	90	5	14	7	147	57	607.86	69
Mac	20	3	44	14	18	99	55	427.16	64
Sum	350	165	114	55	38	722	289	2416.59	160

```
> boxes(L3, boxpos = TRUE, cex = 0.8)
```

What's wrong with this



What's in jump

```
> (jump <-  
+ subset(L3, (lex.Cst == "Norm" & lex.Xst == "Mac") |  
+ (lex.Xst == "Norm" & lex.Cst == "Mac"))[,  
+ c("lex.id", "per", "lex.dur", "lex.Cst", "lex.Xst")])
```

lex.id	per	lex.dur	lex.Cst	lex.Xst
70	1999.49	2.67	Mac	Norm
86	2001.76	12.82	Norm	Mac
130	2000.91	1.88	Mac	Norm
131	1997.76	4.24	Norm	Mac
136	1997.21	0.47	Mac	Norm
136	1997.69	4.24	Norm	Mac
171	1996.39	5.34	Norm	Mac
175	2004.58	9.88	Norm	Mac

—and what will you do about it?

How to fix things

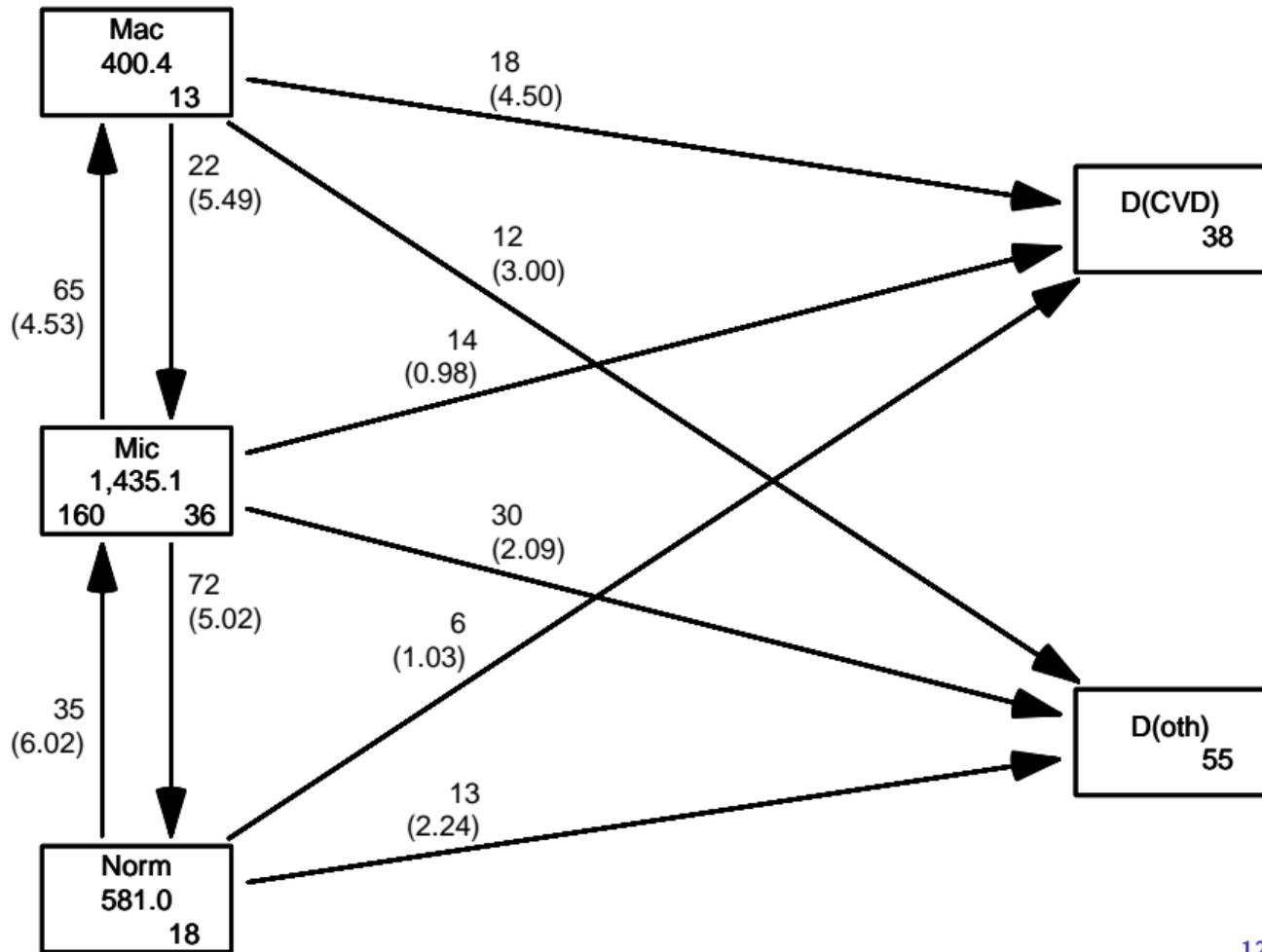
```
> set.seed(1952)
> xcut <- transform(jump,
+                   cut = per + lex.dur * runif(per, 0.1, 0.9),
+                   new.state = "Mic")
> xcut <- select(xcut, c(lex.id, cut, new.state))
> L4 <- rcutLexis(L3, xcut)
> L4 <- Relevel(L4, c("Norm", "Mic", "Mac", "D(CVD)", "D(oth)"))
> summary(L4)
```

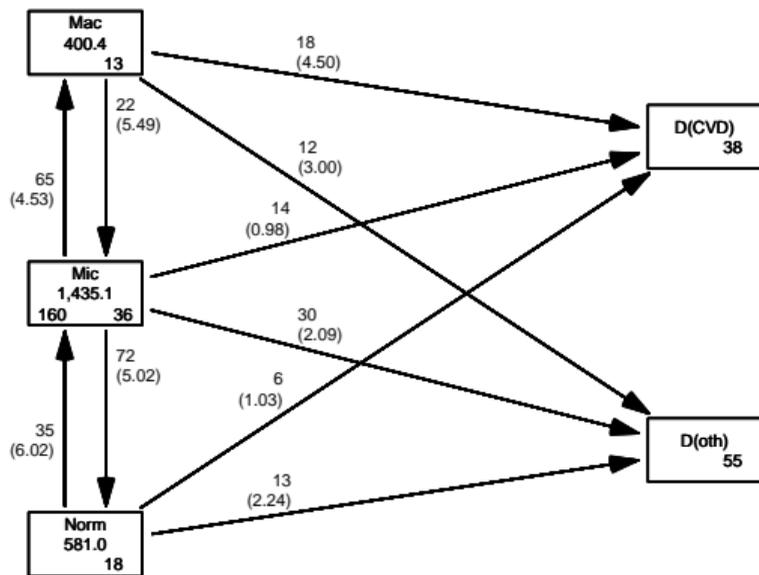
Transitions:

	To								
From	Norm	Mic	Mac	D(CVD)	D(oth)	Records:	Events:	Risk time:	Persons:
Norm	90	35	0	6	13	144	54	581.04	66
Mic	72	312	65	14	30	493	181	1435.14	160
Mac	0	22	41	18	12	93	52	400.41	60
Sum	162	369	106	38	55	730	287	2416.59	160

Plot the boxes

```
> boxes(L4, boxpos = list(x = c(20, 20, 20, 80, 80),  
+                          y = c(10, 50, 90, 75, 25)),  
+      show.BE = "nz",  
+      scale.R = 100, digits.R = 2,  
+      cex = 0.9, pos.arr = 0.3)
```





Explain all the numbers in the graph.

Describe the overall effect of albuminuria on the two mortality rates.

Modeling transition rates

- ▶ A model with a smooth effect of timescales on the rates require follow-up in small bits
- ▶ Achieved by `splitLexis` (or `splitMulti` from `popEpi`)
- ▶ Compare the `Lexis` objects

```
> S4 <- splitMulti(L4, tfi = seq(0, 25, 1/2))
> summary(L4)
```

Transitions:

	To									
From	Norm	Mic	Mac	D(CVD)	D(oth)	Records:	Events:	Risk time:	Persons:	
Norm	90	35	0	6	13	144	54	581.04	66	
Mic	72	312	65	14	30	493	181	1435.14	160	
Mac	0	22	41	18	12	93	52	400.41	60	
Sum	162	369	106	38	55	730	287	2416.59	160	

```
> summary(S4)
```

Transitions:

	To									
From	Norm	Mic	Mac	D(CVD)	D(oth)	Records:	Events:	Risk time:	Persons:	
Norm	1252	35	0	6	13	1306	54	581.04	66	
Mic	72	3101	65	14	30	3282	181	1435.14	160	
Mac	0	22	844	18	12	896	52	400.41	60	
Sum	1324	3158	909	38	55	5484	287	2416.59	160	

How the split works:

```
> subset(L4, lex.id == 96)[,1:7]
```

lex.id	per	age	tfi	lex.dur	lex.Cst	lex.Xst
96	1993.65	51.53	0.00	0.45	Mic	Norm
96	1994.10	51.99	0.45	2.58	Norm	Norm
96	1996.68	54.57	3.03	1.90	Norm	Norm
96	1998.59	56.47	4.94	2.90	Norm	D(CVD)

```
> s4 <- subset(S4, lex.id == 96)[,1:7]
```

```
> s4[c(1:4,NA,nrow(s4)+(-3:0)),]
```

lex.id	per	age	tfi	lex.dur	lex.Cst	lex.Xst
96	1993.65	51.53	0.00	0.45	Mic	Norm
96	1994.10	51.99	0.45	0.05	Norm	Norm
96	1994.15	52.03	0.50	0.50	Norm	Norm
96	1994.65	52.53	1.00	0.50	Norm	Norm
NA	NA	NA	NA	NA	<NA>	<NA>
96	1999.65	57.53	6.00	0.50	Norm	Norm
96	2000.15	58.03	6.50	0.50	Norm	Norm
96	2000.65	58.53	7.00	0.50	Norm	Norm
96	2001.15	59.03	7.50	0.33	Norm	D(CVD)

```
> subset(L4, lex.id == 159)[,1:7]
```

lex.id	per	age	tfi	lex.dur	lex.Cst	lex.Xst
159	1994.02	67.50	0.00	0.13	Mic	Mic
159	1994.16	67.63	0.13	2.66	Mic	Norm
159	1996.82	70.29	2.80	2.37	Norm	Mic
159	1999.20	72.67	5.17	7.32	Mic	Mac
159	2006.52	79.99	12.49	3.95	Mac	D(CVD)

```
> subset(S4, lex.id == 159)[c(1:2,NA,6:7,NA,12:13,NA,27:28,NA,36:37),1:7]
```

lex.id	per	age	tfi	lex.dur	lex.Cst	lex.Xst
159	1994.02	67.50	0.00	0.13	Mic	Mic
159	1994.16	67.63	0.13	0.37	Mic	Mic
NA	NA	NA	NA	NA	<NA>	<NA>
159	1996.02	69.50	2.00	0.50	Mic	Mic
159	1996.52	70.00	2.50	0.30	Mic	Norm
NA	NA	NA	NA	NA	<NA>	<NA>
159	1998.52	72.00	4.50	0.50	Norm	Norm
159	1999.02	72.50	5.00	0.17	Norm	Mic
NA	NA	NA	NA	NA	<NA>	<NA>
159	2005.52	79.00	11.50	0.50	Mic	Mic
159	2006.02	79.50	12.00	0.49	Mic	Mac
NA	NA	NA	NA	NA	<NA>	<NA>
159	2009.52	83.00	15.50	0.50	Mac	Mac
159	2010.02	83.50	16.00	0.44	Mac	D(CVD)

How the split works



Same amount of follow-up

Same transitions

More intervals (5, resp. 37)

Different value of time scales between intervals

Purpose of the split

- ▶ Assumption of constant rate in each interval
- ▶ All intervals are (shorter than) 0.5 years
- ▶ Magnitude of the rates depend on covariates:
 - ▶ fixed covariates
 - ▶ time scales
 - ▶ randomly varying covariates (not now)
- ▶ values of covariates differ between intervals
- ▶ each interval contributes to the (log-)likelihood for a specific rate **from** a given origin state (`lex.Cst`) **to** a given destination state (`lex.Xst`).
- ▶ —looks as the likelihood for a single Poisson observation

Modeling the rate: Mic \rightarrow D(CVD)

```
> mr <- glm(cbind(lex.Xst == "D(CVD)" & lex.Cst != lex.Xst,
+               lex.dur)
+          ~ Ns(tfi, knots = seq( 0, 20, 5)) +
+            Ns(age, knots = seq(50, 80, 10)),
+          family = poisreg,
+          data = subset(S4, lex.Cst == "Mic"))
```

... the same as:

```
> mp <- glm((lex.Xst == "D(CVD)" & lex.Cst != lex.Xst)
+          ~ Ns(tfi, knots = seq( 0, 20, 5)) +
+            Ns(age, knots = seq(50, 80, 10)),
+          offset = log(lex.dur),
+          family = poisson,
+          data = subset(S4, lex.Cst == "Mic"))
> summary(coef(mr) - coef(mp))
```

	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
	-1.368e-12	-2.364e-13	-2.887e-14	-1.625e-13	-7.883e-15	6.839e-13

Modeling the rate: Mic \rightarrow D(CVD)

A convenient wrapper for `Lexis` objects simplifies things substantially:

```
> mL <- glm.Lexis(S4, ~ Ns(tfi, knots = seq( 0, 20, 5)) +  
+                   Ns(age, knots = seq(50, 80, 10)),  
+                   from = "Mic",  
+                   to = "D(CVD)")
```

```
stats::glm Poisson analysis of Lexis object S4 with log link:  
Rates for the transition:  
Mic->D(CVD)
```

```
> summary(coef(mr) - coef(mL))
```

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
0	0	0	0	0	0

```
> summary(coef(mp) - coef(mL))
```

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
-6.839e-13	7.883e-15	2.887e-14	1.625e-13	2.364e-13	1.368e-12

`glm.Lexis` by default models all transitions **to** absorbing states,
from states preceding these

```
> mX <- glm.Lexis(S4, ~ Ns(tfi, knots = seq( 0, 20, 5)) +  
+                   Ns(age, knots = seq(50, 80, 10)) +  
+                   lex.Cst)
```

NOTE:

Multiple transitions **from** state ' Mac', 'Mic', 'Norm ' - are you sure?

The analysis requested is effectively merging outcome states.

You may want analyses using a **stacked** dataset - see `?stack.Lexis`

`stats::glm` Poisson analysis of Lexis object S4 with log link:

Rates for transitions:

Norm->D(CVD)

Mic->D(CVD)

Mac->D(CVD)

Norm->D(oth)

Mic->D(oth)

Mac->D(oth)

Describe the model(s) in mX (look at the figure with the boxes)

- ▶ What rates are modeled ?
- ▶ How are they modeled (assumptions about shapes) ?
- ▶ What are the differences between the rates modeled?
- ▶ What would you rather do?