Survival models and Cox-regression

Bendix Carstensen Steno Diabetes Center Copenhagen,
Gentofte, Denmark
& Department of Biostatistics,
University of Copenhagen
b@bxc.dk
http://BendixCarstensen.com

IDEG 2017 training day, Abu Dhabi,
11 December 2017
http://BendixCarstensen/Epi/Courses/IDEG2017

From /home/bendix/teach/Epi/sdc/surv/surv-cox/slides.tex

Steno Diabetes Center Copenhagen

Rates and Survival

Bendix Carstensen
Senior Statistician, Steno Diabetes Center

Survival models and Cox-regression
IDEG 2017 training day, Abu Dhabi,
11 December 2017

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”)

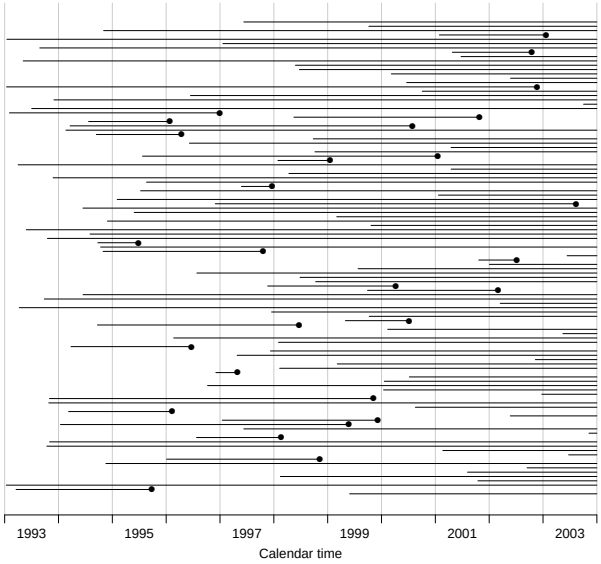
Examples of time-to-event measurements

- 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

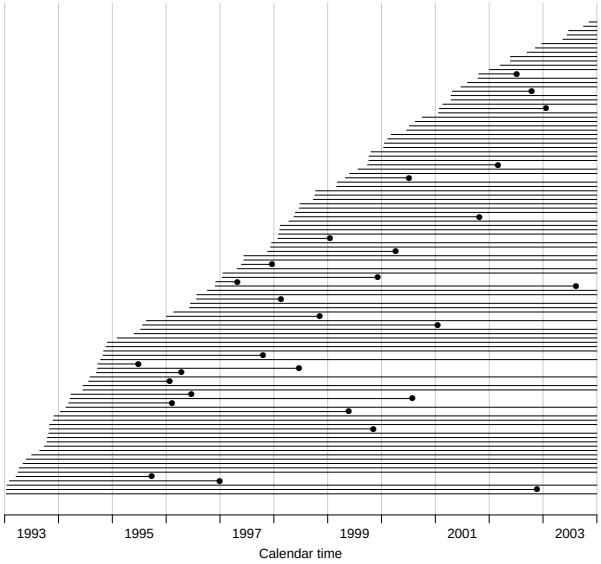
Each blob a death

Study ended at 31 Dec. 2003

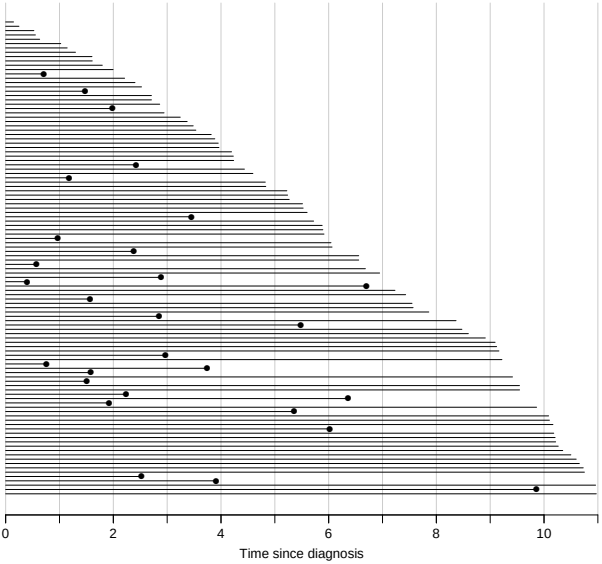


Ordered by date of entry

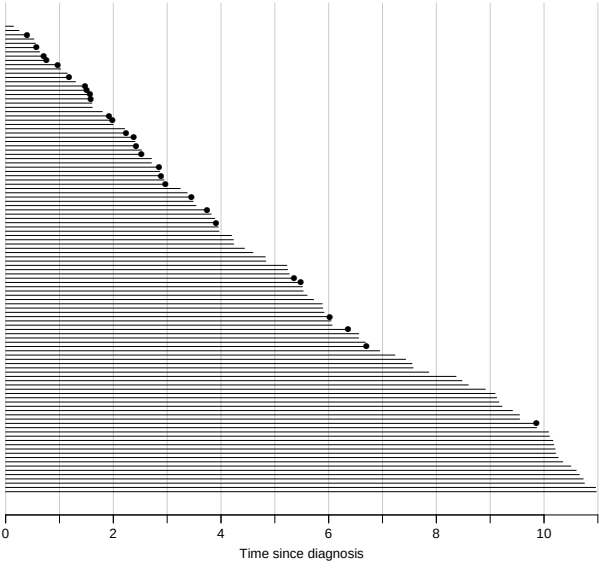
Most likely the order in your database.



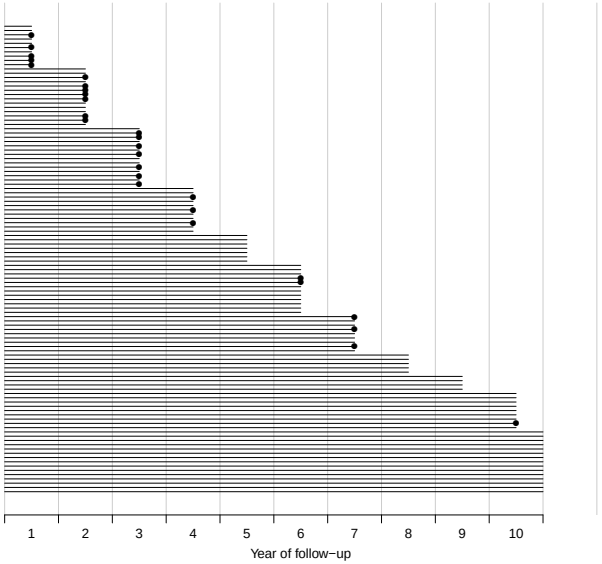
Timescale changed to "Time since diagnosis".



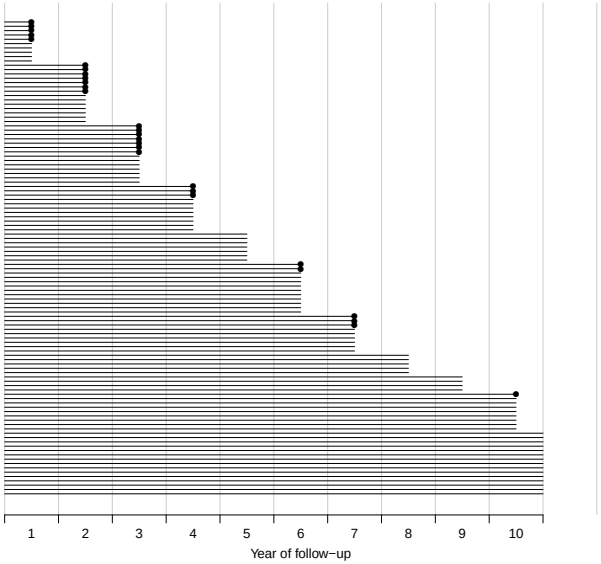
Patients ordered by survival time.



Survival times grouped into bands of survival.



Patients ordered by survival status within each band.



Survival after Cervix cancer

	Stage I			Stage II		
Year	N	D	L	N	D	L
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

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

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

Life-table estimator.

Survival function

Persons enter at time 0:
Date of birth, date of randomization, date of diagnosis.

How long do they survive?
Survival time T — a stochastic variable.

Distribution is characterized by the survival function:

$$\begin{aligned} S(t) &= \mathrm{P}\{\text{survival at least till } t\} \\ &= \mathrm{P}\{T > t\} = 1 - \mathrm{P}\{T \leq t\} = 1 - F(t) \end{aligned}$$

$F(t)$ is the cumulative risk of death before time t.

Intensity / rate / hazard — same same

- The **intensity** or **hazard function**
- Probability of event in interval, reltive to interval length:

$$\lambda(t) = \mathrm{P}\left\{\text{event in } (t, t+h] \mid \text{alive at } t\right\}/h$$

- Characterizes the distribution of survival times as does
  f (density) or
  F (cumulative distibution).
- Theoretical counterpart of a(n empirical) **rate**.

Rate and survival

$$S(t) = \exp\left(-\int_0^t \lambda(s)\,\mathrm{d}s\right) \qquad \lambda(t) = \frac{S'(t)}{S(t)}$$

Survival is a *cumulative* measure, the rate is an *instantaneous* measure.

Note: A cumulative measure requires an origin!

. . . it is always survival **since** some timepoint.

Observed survival and rate

- **Survival studies**:
  Observation of (right censored) survival time:

$$X = \min(T, Z), \quad \delta = 1\{X = T\}$$

  — sometimes conditional on $T > t_0$
  (left truncation, delayed entry).
- **Epidemiological studies**:
  Observation of (components of) a rate:

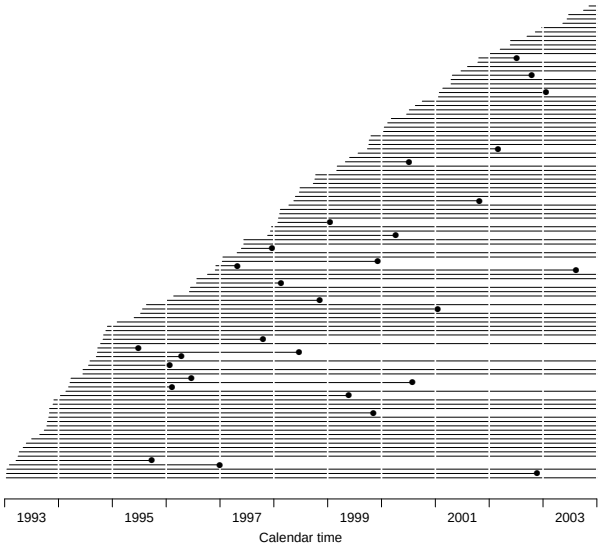
$$D / Y$$

  D: no. events, Y no of person-years, in a prespecified time-frame.

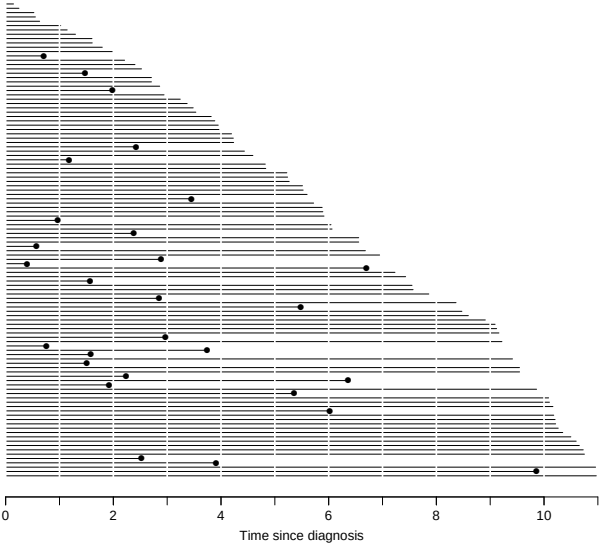
Empirical rates for individuals

- At the *individual* level we introduce the **empirical rate:** (d, y),
  — number of events ($d \in \{0, 1\}$) during y risk time.
- A person contributes several observations of (d, y), with associated covariate values.
- Empirical rates are **responses** in survival analysis.
- The timescale t is a **covariate** — varies within each individual:
  t: age, time since diagnosis, calendar time.
- Don't confuse with y — difference between two points on **any** timescale we may choose.

Empirical rates by calendar time.



Empirical rates by time since diagnosis.



Statistical inference: Likelihood

Two things needed:

- **Data** — what did we actually observe
  Follow-up for each person:
  Entry time, exit time, exit status, covariates
- **Model** — how was data generated
  Rates as a function of time:
  Probability machinery that generated data

Likelihood is the probability of observing the data, assuming the model is correct.

Maximum likelihood estimation is choosing parameters of the model that makes the likelihood maximal.

Likelihood from one person

The likelihood from several empirical rates from one individual is a product of conditional probabilities:

$$\begin{aligned} \mathrm{P}\left\{\text{event at } t_4 | t_0\right\} &= \mathrm{P}\left\{\text{survive } (t_0, t_1) | \text{ alive at } t_0\right\} \times \\ &\quad \mathrm{P}\left\{\text{survive } (t_1, t_2) | \text{ alive at } t_1\right\} \times \\ &\quad \mathrm{P}\left\{\text{survive } (t_2, t_3) | \text{ alive at } t_2\right\} \times \\ &\quad \mathrm{P}\left\{\text{event at } t_4 | \text{ alive at } t_3\right\} \end{aligned}$$

Log-likelihood from one individual is a sum of terms.

Each term refers to one empirical rate (d, y)
— $y = t_i - t_{i-1}$ and mostly $d = 0$.

t_i is the timescale (covariate).

Poisson likelihood

The log-likelihood contributions from follow-up of **one** individual:

$$d_t \log\big(\lambda(t)\big) - \lambda(t) y_t, \quad t = t_1, \ldots, t_n$$

is also the log-likelihood from several independent Poisson observations with mean $\lambda(t) y_t$, i.e. log-mean $\log\big(\lambda(t)\big) + \log(y_t)$

Analysis of the rates, (λ) can be based on a Poisson model with log-link applied to empirical rates where:

- d is the response variable.
- $\log(\lambda)$ is modelled by covariates
- $\log(y)$ is the offset variable.

Likelihood for follow-up of many persons

Adding empirical rates over the follow-up of persons:

$$D = \sum d \qquad Y = \sum y \quad \Rightarrow \quad D\log(\lambda) - \lambda Y$$

- Persons are assumed independent
- Contribution from the same person are **conditionally** independent, hence give separate contributions to the log-likelihood.
- Therefore equivalent to likelihood for independent Poisson variates
- No need to correct for dependent observations; the likelihood is a product.

Likelihood

Probability of the data and the parameter:

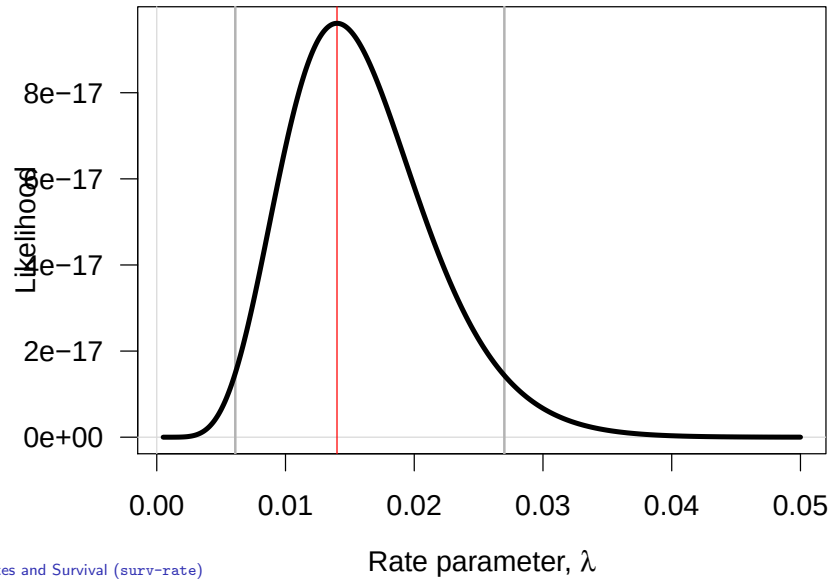
Assuming the rate (intensity) is constant, λ, the probability of observing 7 deaths in the course of 500 person-years:

$$\begin{aligned} \mathrm{P}\left\{D = 7, Y = 500|\lambda\right\} &= \lambda^D \mathrm{e}^{\lambda Y} \times K \\ &= \lambda^7 \mathrm{e}^{\lambda 500} \times K \\ &= L(\lambda|\text{data}) \end{aligned}$$

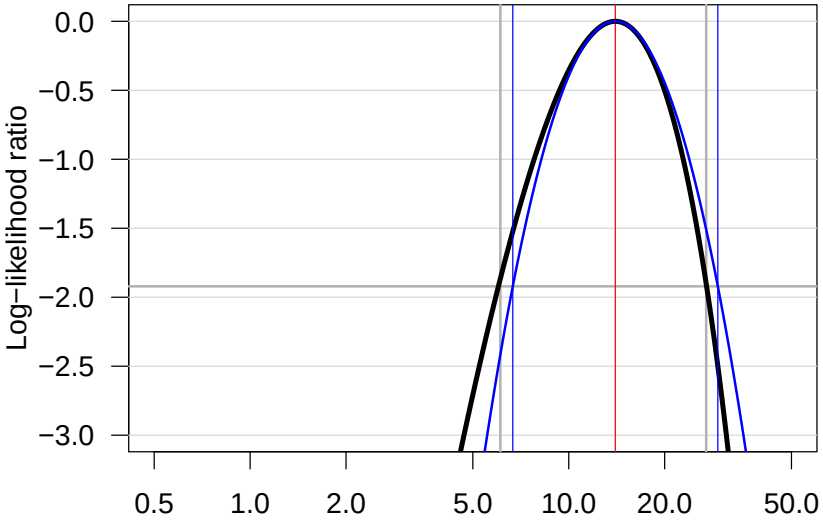
Best guess of λ is where this function is as large as possible.

Confidence interval is where it is not too far from the maximum

Likelihood function



Likelihood function



Example using R

Poisson likelihood, for one rate, based on 17 events in 843.7 PY:

```
library( Epi )
D <- 17 ; Y <- 843.7
m1 <- glm( D ~ 1, offset=log(Y/1000), family=poisson)
ci.exp( m1 )
```

```
            exp(Est.)     2.5%    97.5%
(Intercept)  20.14934 12.52605 32.41213
```

Poisson likelihood, two rates, or one rate and RR:

```
D <- c(17,28) ; Y <- c(843.7,632.3) ; gg <- factor(0:1)
m2 <- glm( D ~ gg, offset=log(Y/1000), family=poisson)
ci.exp( m2 )
```

```
            exp(Est.)      2.5%     97.5%
(Intercept) 20.149342 12.526051 32.412130
gg1          2.197728  1.202971  4.015068
```

Example using R

Poisson likelihood, two rates, or one rate and RR:

```
D <- c(17,28) ; Y <- c(843.7,632.3) ; gg <- factor(0:1)
m2 <- glm( D ~ gg, offset=log(Y/1000), family=poisson)
ci.exp( m2 )
```

```
            exp(Est.)      2.5%     97.5%
(Intercept) 20.149342 12.526051 32.412130
gg1          2.197728  1.202971  4.015068
```

```
m3 <- glm( D ~ gg - 1, offset=log(Y/1000), family=poisson)
ci.exp( m3 )
```

```
    exp(Est.)     2.5%    97.5%
gg0  20.14934 12.52605 32.41213
gg1  44.28278 30.57545 64.13525
```

Kaplan-Meier estimators

Bendix Carstensen
Senior Statistician, Steno Diabetes Center

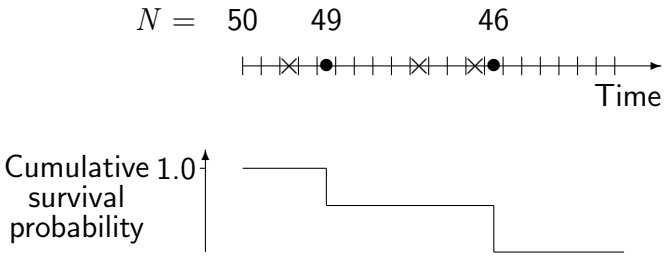
Survival models and Cox-regression
IDEG 2017 training day, Abu Dhabi,
11 December 2017

The Kaplan-Meier Method

- The most common method of estimating the survival function.
- A non-parametric method.
- Divides time into small intervals where the intervals are defined by the unique times of failure (death).
- Based on conditional probabilities as we are interested in the probability a subject surviving the next time interval given that they have survived so far.

Kaplan–Meier method illustrated

(• = failure and × = censored):



- Steps caused by multiplying by $(1 - 1/49)$ and $(1 - 1/46)$ respectively
- Late entry can also be dealt with

Using R: Surv()

```
library( survival )
data( lung )
head( lung, 3 )
```

```
  inst time status age sex ph.ecog ph.karno pat.karno meal.cal wt.loss
1    3  306      2  74   1       1       90       100      1175      NA
2    3  455      2  68   1       0       90        90      1225      15
3    3 1010      1  56   1       0       90        90        NA      15
```

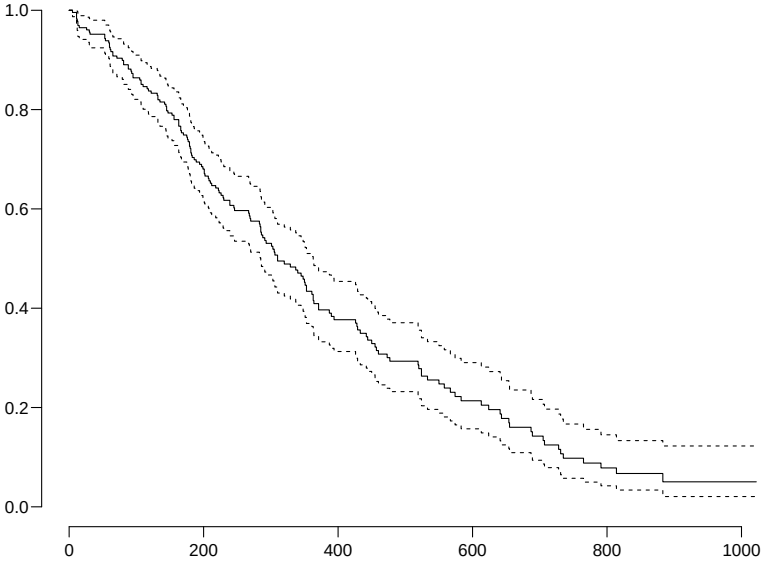
```
with( lung, Surv( time, status==2 ) )[1:10]
```

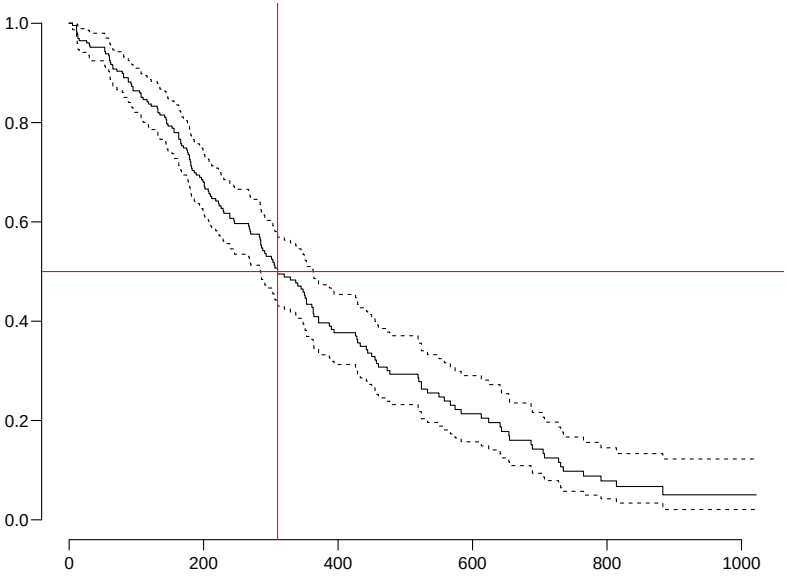
```
 [1]  306   455  1010+  210   883  1022+  310   361   218   166
```

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

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

      n  events  median 0.95LCL 0.95UCL
    228     165     310     285     363
```

```
plot( s.km )
abline( v=310, h=0.5, col="red" )
```


1.0
0.8
0.6
0.4
0.2
0.0
0
200
400
600
800
1000


1.0
0.8
0.6
0.4
0.2
0.0
0
200
400
600
800
1000

The Cox-model

Bendix Carstensen
Senior Statistician, Steno Diabetes Center

Survival models and Cox-regression
IDEG 2017 training day, Abu Dhabi,
11 December 2017

The proportional hazards model

$$\lambda(t, x) = \lambda_0(t) \times \exp(x'\beta)$$

The partial log-likelihood for the regression parameters (βs):

$$\ell(\beta) = \sum_{\text{death times}} \log\left(\frac{\mathrm{e}^{\,x_{\text{death}}\beta}}{\sum_{i\in\mathcal{R}_t}\mathrm{e}^{\,x_i\beta}}\right)$$

- This is David Cox's invention.
- Extremely efficient from a computational point of view.
- The baseline hazard $\lambda_0(t)$ is bypassed (profiled out).

Proportional Hazards model

- The baseline hazard rate, $\lambda_0(t)$, is the hazard rate when all the covariates are 0.
- The form of the above equation means that covariates act **multiplicatively** on the baseline hazard rate.
- Time is a covariate (albeit modeled in a special way).
- The baseline hazard is a function of time and thus varies with time.
- No assumption about the shape of the underlying hazard function.
- — but you will never see the shape of the baseline hazard . . .

Interpreting Regression Coefficients

- If x_j is binary $\exp(\beta_j)$ is the estimated hazard ratio for subjects corresponding to $x_j = 1$ compared to those where $x_j = 0$.
- If x_j is continuous $\exp(\beta_j)$ is the estimated increase/decrease in the hazard rate for a unit change in x_j.
- With more than one covariate interpretation is similar, i.e. $\exp(\beta_j)$ is the hazard ratio for subjects who **only** differ with respect to covariate x_j.

Fitting a Cox- model in R

```
library( survival )
data(bladder)
bladder <- subset( bladder, enum<2 )
head( bladder)
```

```
   id rx number size stop event enum
1   1  1      1    3    1     0    1
5   2  1      2    1    4     0    1
9   3  1      1    1    7     0    1
13  4  1      5    1   10     0    1
17  5  1      4    1    6     1    1
21  6  1      1    1   14     0    1
```

Fitting a Cox-model in R

```
c0 <- coxph( Surv(stop,event) ~ number + size, data=bladder )
c0
```

```
Call:
coxph(formula = Surv(stop, event) ~ number + size, data = bladder)

          coef exp(coef) se(coef)     z       p
number 0.20491   1.22742  0.07036 2.912 0.00359
size   0.06135   1.06327  0.10328 0.594 0.55254

Likelihood ratio test=7.04  on 2 df, p=0.02963
n= 85, number of events= 47
```

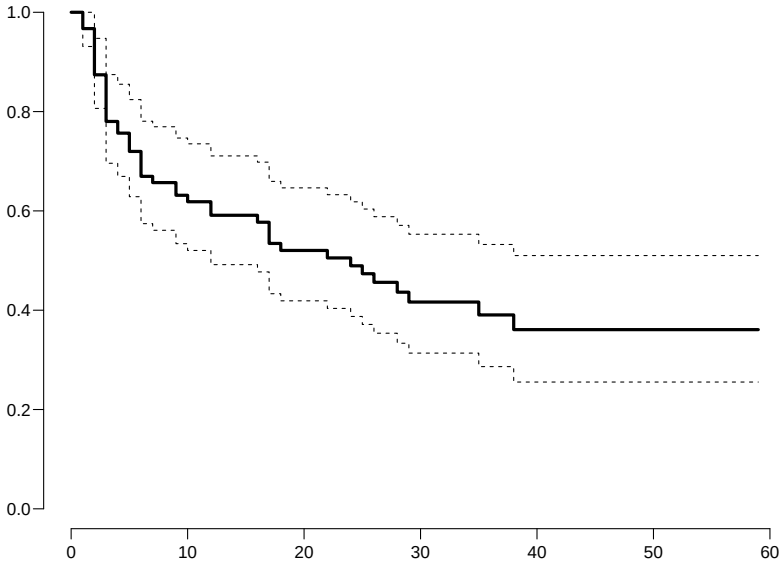
Plotting the base survival in R

```
plot( survfit(c0) )
lines( survfit(c0), conf.int=F, lwd=3 )
```

The `plot.coxph` plots the survival curve for a person with an average covariate value

— which is **not** the average survival for the population considered...

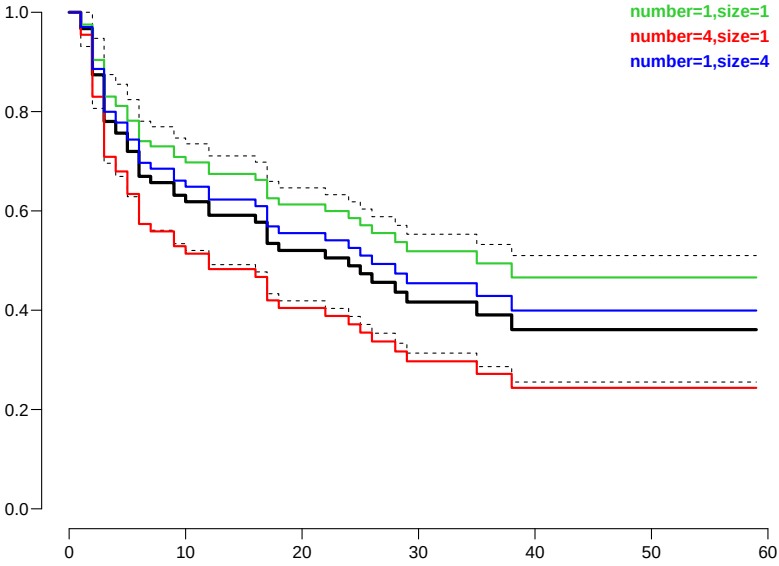
— and not necessarily meaningful


1.0
0.8
0.6
0.4
0.2
0.0
0
10
20
30
40
50
60

Plotting the base survival in R

You can plot the survival curve for specific values of the covariates, using the `newdata=` argument:

```
plot( survfit(c0) )
lines( survfit(c0), conf.int=F, lwd=3 )
lines( survfit(c0, newdata=data.frame(number=1,size=1)),
       lwd=2, col="limegreen" )
text( par("usr")[2]*0.98, 1.00, "number=1,size=1",
      col="limegreen", font=2, adj=1 )
```


number=1,size=1
number=4,size=1
number=1,size=4
1.0
0.8
0.6
0.4
0.2
0.0
0
10
20
30
40
50
60

Who needs the Cox-model anyway?

Bendix Carstensen
Senior Statistician, Steno Diabetes Center

Survival models and Cox-regression
IDEG 2017 training day, Abu Dhabi,
11 December 2017

A look at the Cox model

$$\lambda(t, x) = \lambda_0(t) \times \exp(x'\beta)$$

A model for the rate as a function of t and x.

The covariate t has a special status:

- Computationally, because all individuals contribute to (some of) the range of t.
- . . . the scale along which time is split (the risk sets)
- Conceptually t is just a covariate that varies within individual.
- Cox's approach profiles $\lambda_0(t)$ out from the model

The Cox-likelihood as profile likelihood

- One parameter per death time to describe the effect of time (i.e. the chosen timescale).

$$\log\big(\lambda(t, x_i)\big) = \log\big(\lambda_0(t)\big) + \beta_1 x_{1i} + \cdots + \beta_p x_{pi} = \alpha_t + \eta_i$$

- Profile likelihood:
  - Derive estimates of α_t as function of data and βs — assuming constant rate between death times
  - Insert in likelihood, now only a function of data and βs
  - Turns out to be Cox's partial likelihood

The Cox-likelihood: mechanics of computing

- The likelihood is computed by summing over risk-sets at each event time t:

$$\ell(\eta) = \sum_t \log\left(\frac{e^{\eta_{\text{death}}}}{\sum_{i\in\mathcal{R}_t} e^{\eta_i}}\right)$$

- this is essentially splitting follow-up time at event- (and censoring) times
- . . . repeatedly in every cycle of the iteration
- . . . simplified by not keeping track of risk time
- . . . but only works along **one** time scale

$$\log\big(\lambda(t, x_i)\big) = \log\big(\lambda_0(t)\big) + \beta_1 x_{1i} + \cdots + \beta_p x_{pi} = \alpha_t + \eta_i$$

- Suppose the time scale has been divided into small intervals with at most one death in each:
- Empirical rates: (d_{it}, y_{it}) — each t has at most one $d_{it} = 0$.
- Assume w.l.o.g. the ys in the empirical rates all are 1.
- Log-likelihood contributions that contain information on a specific time-scale parameter α_t will be from:
  - the (only) empirical rate $(1, 1)$ with the death at time t.
  - all other empirical rates $(0, 1)$ from those at risk at time t.

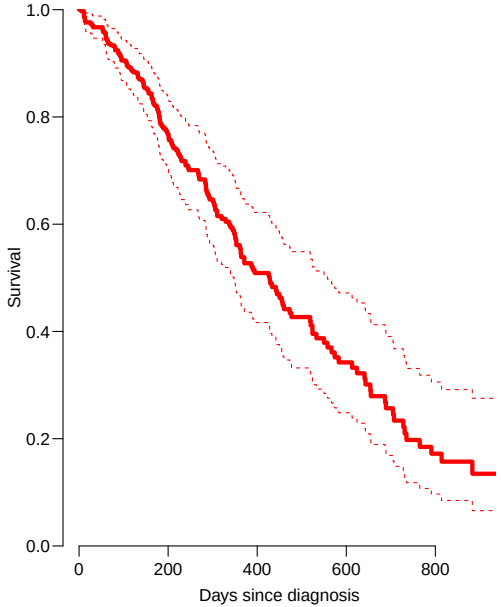
Splitting the dataset a priori

- The Poisson approach needs a dataset of empirical rates (d, y) with suitably small values of y.
- — each individual contributes many empirical rates
- (one per risk-set contribution in Cox-modelling)
- From each empirical rate we get:
  - Poisson-response d
  - Risk time $y \to \log(y)$ as offset
  - Covariate value for the timescale (time since entry, current age, current date, ...)
  - other covariates

Example: Mayo Clinic lung cancer

- Survival after lung cancer
- Covariates:
  - Age at diagnosis
  - Sex
  - Time since diagnosis
- Cox model
- Split data:
  - Poisson model, time as factor
  - Poisson model, time as spline

Mayo Clinic lung cancer 60 year old woman



Example: Mayo Clinic lung cancer I

```
> library( survival )
> library( Epi )
> Lung <- Lexis( exit = list( tfe=time ),
+                exit.status = factor(status,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 tfe timescale.
```

Example: Mayo Clinic lung cancer II

```
> mL.cox <- coxph( Surv( tfe, tfe+lex.dur, lex.Xst=="Dead" ) ~
+                  age + factor( sex ),
+                  method="breslow", eps=10^-8, iter.max=25, data=Lung )
> Lung.s <- splitLexis( Lung,
+                       breaks=c(0,sort(unique(Lung$time))),
+                       time.scale="tfe" )
> Lung.S <- splitLexis( Lung,
+                       breaks=c(0,sort(unique(Lung$time[Lung$lex.Xst=="Dead"]))),
+                       time.scale="tfe" )
> summary( Lung.s )

Transitions:
     To
From    Alive Dead  Records:  Events: Risk time:  Persons:
  Alive 19857  165     20022      165      69593       228

> summary( Lung.S )
```

Example: Mayo Clinic lung cancer III

```
Transitions:
     To
From    Alive Dead  Records:  Events: Risk time:  Persons:
  Alive 15916  165     16081      165      69593       228
```

```
> subset( Lung.s, lex.id==96 )[,1:11]
```

```
     lex.id tfe lex.dur lex.Cst lex.Xst inst time status age sex ph.ecog
9235     96   0       5   Alive   Alive   12   30      2  72   1       2
9236     96   5       6   Alive   Alive   12   30      2  72   1       2
9237     96  11       1   Alive   Alive   12   30      2  72   1       2
9238     96  12       1   Alive   Alive   12   30      2  72   1       2
9239     96  13       2   Alive   Alive   12   30      2  72   1       2
9240     96  15      11   Alive   Alive   12   30      2  72   1       2
9241     96  26       4   Alive    Dead   12   30      2  72   1       2
```

```
> nlevels( factor( Lung.s$tfe ) )
```

```
[1] 186
```

Example: Mayo Clinic lung cancer IV

```
> system.time(
+ mLs.pois.fc <- glm( lex.Xst=="Dead" ~ - 1 + factor( tfe ) +
+                                  age + factor( sex ),
+                                  offset = log(lex.dur),
+                        family=poisson, data=Lung.s, eps=10^-8, maxit=25 )
+              )

   user  system elapsed
 11.489  18.016   8.202

> length( coef(mLs.pois.fc) )

[1] 188

> system.time(
+ mLS.pois.fc <- glm( lex.Xst=="Dead" ~ - 1 + factor( tfe ) +
+                                  age + factor( sex ),
+                                  offset = log(lex.dur),
+                        family=poisson, data=Lung.S, eps=10^-8, maxit=25 )
+              )
```

Example: Mayo Clinic lung cancer V

```
   user  system elapsed
  4.096   6.018   2.717
```

```
> length( coef(mLS.pois.fc) )
```

```
[1] 142
```

```
> t.kn <- c(0,25,100,500,1000)
> dim( Ns(Lung.s$tfe,knots=t.kn) )
```

```
[1] 20022     4
```

```
> system.time(
+ mLs.pois.sp <- glm( lex.Xst=="Dead" ~ Ns( tfe, knots=t.kn ) +
+                                        age + factor( sex ),
+                      offset = log(lex.dur),
+                      family=poisson, data=Lung.s, eps=10^-8, maxit=25 )
+             )
```

Example: Mayo Clinic lung cancer VI

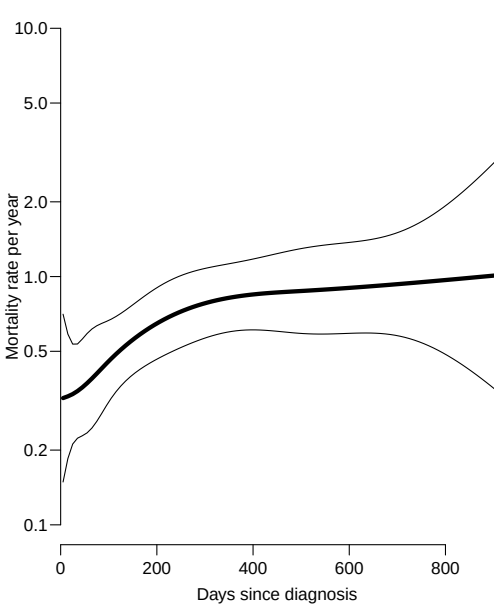
```
   user  system elapsed
  0.331   0.469   0.246
```

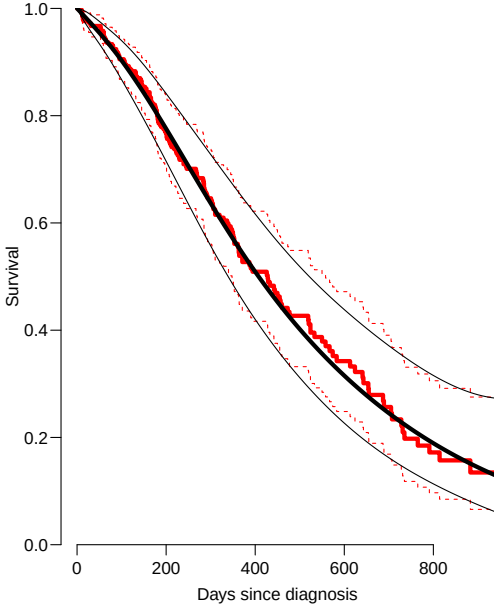
```
> ests <-
+ rbind( ci.exp(mL.cox),
+        ci.exp(mLs.pois.fc,subset=c("age","sex")),
+        ci.exp(mLS.pois.fc,subset=c("age","sex")),
+        ci.exp(mLs.pois.sp,subset=c("age","sex")) )
> cmp <- cbind( ests[c(1,3,5,7)  ,],
+               ests[c(1,3,5,7)+1,] )
> rownames( cmp ) <- c("Cox","Poisson-factor","Poisson-factor (D)","Poisson-spline")
> colnames( cmp )[c(1,4)] <- c("age","sex")
```

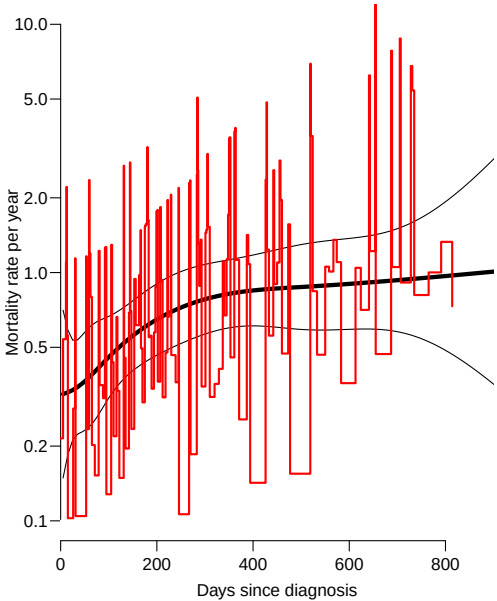
```
> round( cmp, 7 )
```

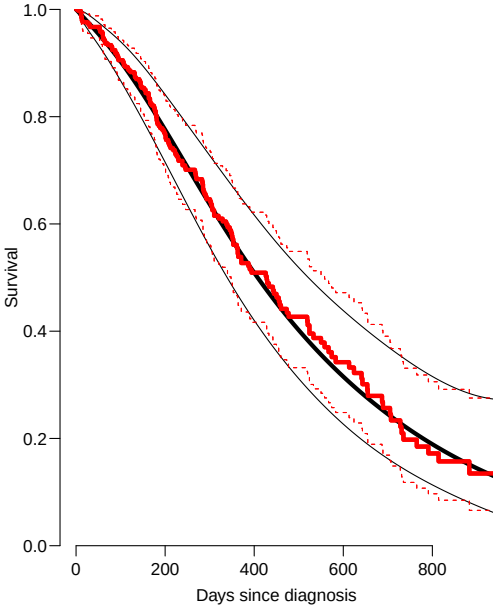
Example: Mayo Clinic lung cancer VII

	age	2.5%	97.5%	sex	2.5%	97.5%
Cox	1.017158	0.9989388	1.035710	0.5989574	0.4313720	0.8316487
Poisson-factor	1.017158	0.9989388	1.035710	0.5989574	0.4313720	0.8316487
Poisson-factor (D)	1.017332	0.9991211	1.035874	0.5984794	0.4310150	0.8310094
Poisson-spline	1.016189	0.9980329	1.034676	0.5998287	0.4319932	0.8328707


10.0
5.0
2.0
1.0
0.5
0.2
0.1
Mortality rate per year
0
200
400
600
800
Days since diagnosis


1.0
0.8
0.6
0.4
0.2
0.0
Survival
0
200
400
600
800
Days since diagnosis


10.0
5.0
2.0
1.0
0.5
0.2
0.1
Mortality rate per year
0
200
400
600
800
Days since diagnosis


1.0
0.8
0.6
0.4
0.2
0.0
Survival
0
200
400
600
800
Days since diagnosis

Deriving the survival function

```
> mLs.pois.sp <- glm( lex.Xst=="Dead" ~ Ns( tfe, knots=t.kn ) +
+                                  age + factor( sex ),
+                       offset = log(lex.dur),
+                       family=poisson, data=Lung.s, eps=10^-8, maxit=25 )
```

```
> CM <- cbind( 1, Ns( seq(10,1000,10)-5, knots=t.kn ), 60, 1 )
> lambda <- ci.exp( mLs.pois.sp, ctr.mat=CM )
> Lambda <- ci.cum( mLs.pois.sp, ctr.mat=CM, intl=10 )[,-4]
> survP  <- exp(-rbind(0,Lambda))
```

Code and output for the entire example avaiable in
http://bendixcarstensen.com/AdvCoh/WNtCMa/

What the Cox-model really is

Taking the life-table approach *ad absurdum* by:

- dividing time very finely and
- modeling one covariate, the time-scale, with one parameter per distinct value.
- the **model** for the time scale is really with exchangeable time-intervals.
- $\Rightarrow$ difficult to access the baseline hazard (which looks terrible)
- $\Rightarrow$ uninitiated tempted to show survival curves where irrelevant

Models of this world

- Replace the α_ts by a parametric function $f(t)$ with a limited number of parameters, for example:
  - Piecewise constant
  - Splines (linear, quadratic or cubic)
  - Fractional polynomials
- the two latter brings model into "this world":
  - smoothly varying rates
  - parametric closed form representation of baseline hazard
  - finite no. of parameters
- Makes it really easy to use rates directly in calculations of
  - expected residual life time
  - state occupancy probabilities in multistate models
  - ...

Multiple time scales and continuous rates

Bendix Carstensen
Senior Statistician, Steno Diabetes Center

Survival models and Cox-regression
IDEG 2017 training day, Abu Dhabi,
11 December 2017

Testis cancer

Testis cancer in Denmark:

```
> options( show.signif.stars=FALSE )
> library( Epi )
> data( testisDK )
> str( testisDK )

'data.frame': 4860 obs. of  4 variables:
 $ A: num  0 1 2 3 4 5 6 7 8 9 ...
 $ P: num  1943 1943 1943 1943 1943 ...
 $ D: num  1 1 0 1 0 0 0 0 0 0 ...
 $ Y: num  39650 36943 34588 33267 32614 ...

> head( testisDK )

  A    P D        Y
1 0 1943 1 39649.50
2 1 1943 1 36942.83
3 2 1943 0 34588.33
4 3 1943 1 33267.00
5 4 1943 0 32614.00
6 5 1943 0 32020.33
```

Cases, PY and rates

```
> stat.table( list(A=floor(A/10)*10,
+                  P=floor(P/10)*10),
+             list( D=sum(D),
+                   Y=sum(Y/1000),
+                rate=ratio(D,Y,10^5) ),
+             margins=TRUE, data=testisDK )
```

	P						
A	1940	1950	1960	1970	1980	1990	Total
0	10.00	7.00	16.00	18.00	9.00	10.00	70.00
	2604.66	4037.31	3884.97	3820.88	3070.87	2165.54	19584.22
	0.38	0.17	0.41	0.47	0.29	0.46	0.36
10	13.00	27.00	37.00	72.00	97.00	75.00	321.00
	2135.73	3505.19	4004.13	3906.08	3847.40	2260.97	19659.48
	0.61	0.77	0.92	1.84	2.52	3.32	1.63
20	124.00	221.00	280.00	535.00	724.00	557.00	2441.00
	2225.55	2923.22	3401.65	4028.57	3941.18	2824.58	19344.74
	5.57	7.56	8.23	13.28	18.37	19.72	12.62

Linear effects in glm

Two ways of fitting a Poisson model, D and Y must be there, note poisson, resp. poisreg.

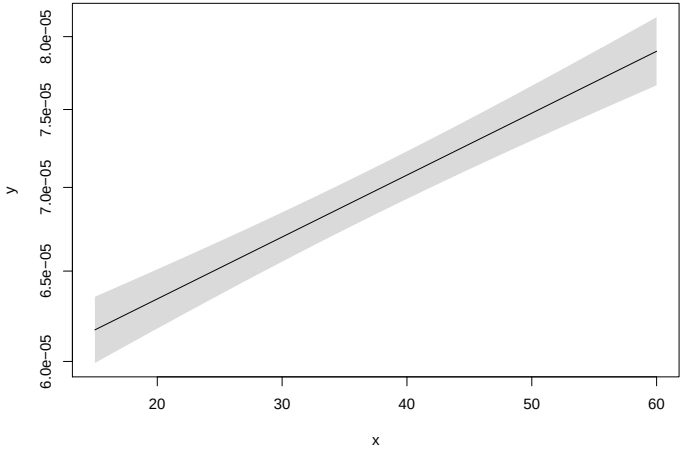
```
> m0 <- glm(       D    ~ A, offset=log(Y), family=poisson, data=testisDK )
> m1 <- glm( cbind(D,Y) ~ A                , family=poisreg, data=testisDK )
> round( ci.exp( m0 ), 4 )

            exp(Est.)   2.5%  97.5%
(Intercept)    0.0001 0.0001 0.0001
A              1.0055 1.0046 1.0064

> round( ci.exp( m1 ), 4 )

            exp(Est.)   2.5%  97.5%
(Intercept)    0.0001 0.0001 0.0001
A              1.0055 1.0046 1.0064
```

Linear increase of log-rates by age

Linear effects in glm

```
> nd <- data.frame( A=15:60 )
> pr <- ci.pred( ml, newdata=nd )
> head( pr )

      Estimate         2.5%        97.5%
1 6.170105e-05 5.991630e-05 6.353897e-05
2 6.204034e-05 6.028525e-05 6.384652e-05
3 6.238149e-05 6.065547e-05 6.415663e-05
4 6.272452e-05 6.102689e-05 6.446937e-05
5 6.306943e-05 6.139944e-05 6.478485e-05
6 6.341624e-05 6.177301e-05 6.510319e-05

> matplot( nd$A, pr,
+          type="l", lty=1, lwd=c(3,1,1), col="black", log="y" )
```

Linear effects in glm




```
> matshade( nd$A, pr, log="y", plot=TRUE )
```

Quadratic effects in `glm`

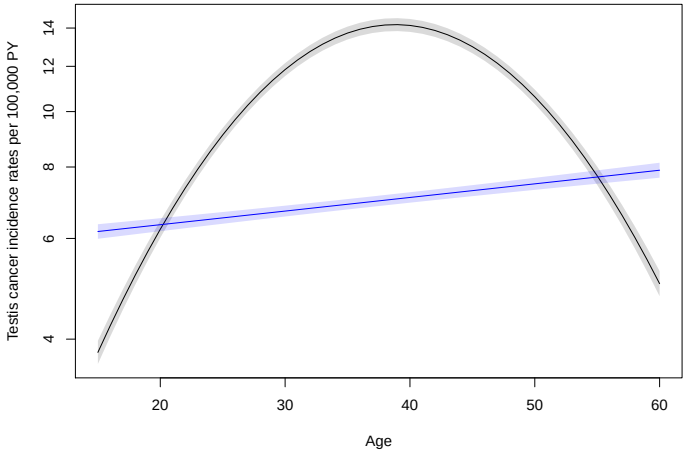
How do rates depend on age?

```
> mq <- glm( cbind(D,Y) ~ A + I(A^2), family=poisreg, data=testisDK )
> round( ci.lin( mq ), 4 )

             Estimate StdErr         z P     2.5%    97.5%
(Intercept) -12.3656 0.0596 -207.3579 0 -12.4825 -12.2487
A             0.1806 0.0033   54.8282 0   0.1741   0.1871
I(A^2)       -0.0023 0.0000  -53.6997 0  -0.0024  -0.0022

> round( ci.exp( mq ), 4 )

             exp(Est.)   2.5%  97.5%
(Intercept)     0.0000 0.0000 0.0000
A               1.1979 1.1902 1.2057
I(A^2)          0.9977 0.9976 0.9978
```

Quadratic effect in glm

```
> matshade( nd$A, ci.pred( mq, nd )*10^5, plot=TRUE, log="y",
+           xlab="Age", ylab="Testis cancer incidence rates per 100,000 PY")
```

Quadratic effect in glm



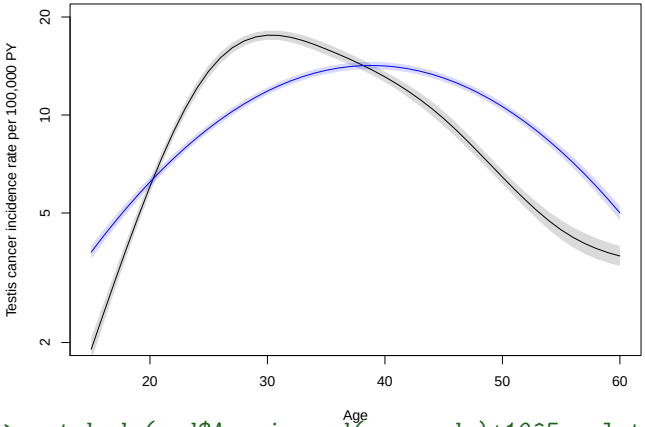

```
> matshade( nd$A, ci.pred( mq, nd )*10^5, plot=TRUE, log="y",
+           xlab="Age", ylab="Testis cancer incidence rates per 100,000 PY")
> matshade( nd$A, ci.pred( ml, nd )*10^5, col="blue" )
```

Spline effects in glm

```
> library( splines )
> ms <- glm( cbind(D,Y) ~ Ns(A,knots=seq(15,65,10)), family=poisreg, data=testisDK )
> round( ci.exp( ms ), 3 )
```

	exp(Est.)	2.5%	97.5%
(Intercept)	0.000	0.000	0.000
Ns(A, knots = seq(15, 65, 10))1	8.548	7.650	9.551
Ns(A, knots = seq(15, 65, 10))2	5.706	4.998	6.514
Ns(A, knots = seq(15, 65, 10))3	1.002	0.890	1.128
Ns(A, knots = seq(15, 65, 10))4	14.402	11.896	17.436
Ns(A, knots = seq(15, 65, 10))5	0.466	0.429	0.505

Spline effects in `glm`




```
> matshade( nd$A, ci.pred( ms, nd )*10^5, plot=TRUE, log="y", ylim=c(2,20),
+           xlab="Age", ylab="Testis cancer incidence rate per 100,000 PY" )
> matshade( nd$A, ci.pred( mq, nd )*10^5, col="blue" )
```

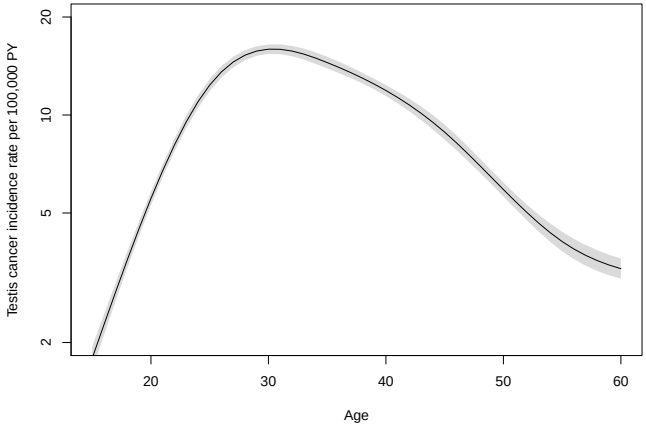
Adding a linear period effect

```
> msp <- glm( cbind(D,Y) ~ Ns(A,knots=seq(15,65,10)) + P, family=poisreg, data=testi
> round( ci.lin( msp ), 3 )
```

	Estimate	StdErr	z	P	2.5%	97.5%
(Intercept)	-58.105	1.444	-40.229	0.000	-60.935	-55.274
Ns(A, knots = seq(15, 65, 10))1	2.120	0.057	37.444	0.000	2.009	2.231
Ns(A, knots = seq(15, 65, 10))2	1.700	0.068	25.157	0.000	1.567	1.832
Ns(A, knots = seq(15, 65, 10))3	0.007	0.060	0.110	0.913	-0.112	0.125
Ns(A, knots = seq(15, 65, 10))4	2.596	0.097	26.631	0.000	2.405	2.787
Ns(A, knots = seq(15, 65, 10))5	-0.780	0.042	-18.748	0.000	-0.861	-0.698
P	0.024	0.001	32.761	0.000	0.023	0.025

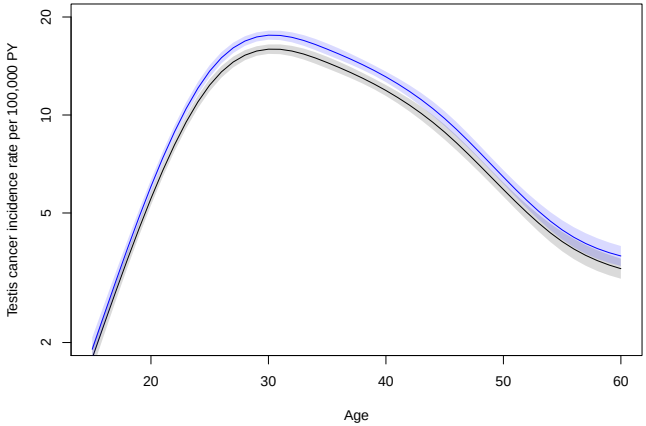
```
> matshade( nd$A, ci.pred( msp, cbind(nd,P=1970) )*10^5, plot=TRUE,
+           log="y", ylim=c(2,20),
+           xlab="Age", ylab="Testis cancer incidence rate per 100,000 PY" )
> matshade( nd$A, ci.pred( ms, nd )*10^5, col="blue" )
```

Adding a linear period effect




```
> matshade( nd$A, ci.pred( msp, cbind(nd,P=1970) )*10^5, plot=TRUE,
+           log="y", ylim=c(2,20),
+           xlab="Age", ylab="Testis cancer incidence rate per 100,000 PY" )
```

Adding a linear period effect




```
> matshade( nd$A, ci.pred( msp, cbind(nd,P=1970) )*10^5, plot=TRUE,
+           log="y", ylim=c(2,20),
+           xlab="Age", ylab="Testis cancer incidence rate per 100,000 PY" )
> matshade( nd$A, ci.pred( ms, nd )*10^5, col="blue" )
```

The period effect

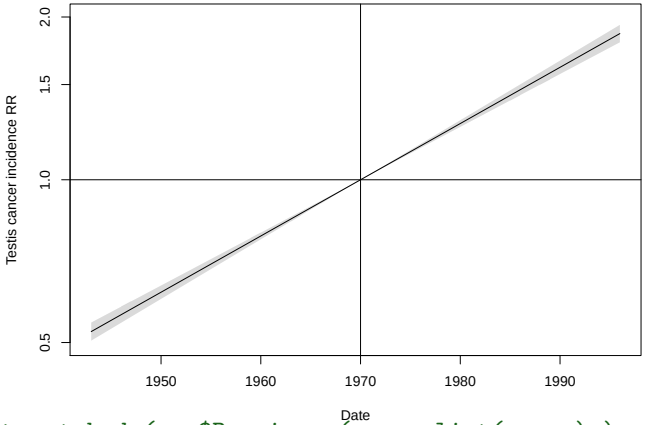
It is **relative** to some reference, say P=1970.

It is the same for any age, so we just choose one, A=40

The rate ratio is the ratio of predictions from two data frames:

```
> np <- data.frame( A=40, P=1943:1996 )
> nr <- data.frame( A=40, P=1970 )
> ci.exp( msp, subset="P" )

  exp(Est.)     2.5%    97.5%
P  1.024235 1.022769 1.025704

> matshade( np$P, ci.exp( msp, list(np,nr) ), plot=TRUE,
+           log="y", ylim=c(0.5,2),
+           xlab="Date", ylab="Testis cancer incidence RR" )
> abline( h=1, v=1970, pch=3 )
```

Period effect




```
> matshade( np$P, ci.exp( msp, list(np,nr) ), plot=TRUE,
+           log="y", ylim=c(0.5,2),
+           xlab="Date", ylab="Testis cancer incidence RR" )
> abline( h=1, v=1970, pch=3 )
```

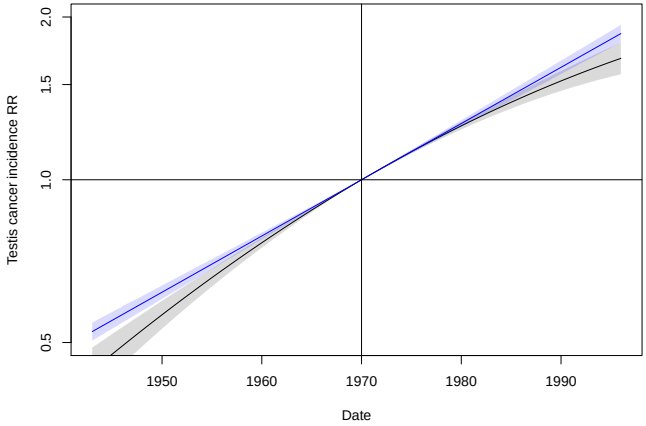
A quadratic period effect

```
> mspq <- glm( cbind(D,Y) ~ Ns(A,knots=seq(15,65,10)) + P + I(P^2),
+              family=poisreg, data=testisDK )
> round( ci.exp( mspq ), 3 )
```

```
                                  exp(Est.)   2.5%  97.5%
(Intercept)                           0.000  0.000  0.000
Ns(A, knots = seq(15, 65, 10))1       8.356  7.478  9.337
Ns(A, knots = seq(15, 65, 10))2       5.513  4.829  6.295
Ns(A, knots = seq(15, 65, 10))3       1.006  0.894  1.133
Ns(A, knots = seq(15, 65, 10))4      13.439 11.101 16.269
Ns(A, knots = seq(15, 65, 10))5       0.458  0.422  0.497
P                                     2.189  1.457  3.291
I(P^2)                                1.000  1.000  1.000
```

```
> matshade( np$P, ci.exp( mspq, list(np,nr) ), plot=TRUE,
+           log="y", ylim=c(0.5,2),
+           xlab="Date", ylab="Testis cancer incidence RR" )
> matshade( np$P, ci.exp( msp, list(np,nr) ), col="blue" )
> abline( h=1, v=1970, lty=3 )
```

A quadratic period effect




```
> matshade( np$P, ci.exp( mspq, list(np,nr) ), plot=TRUE,
+           log="y", ylim=c(0.5,2),
+           xlab="Date", ylab="Testis cancer incidence RR" )
> matshade( np$P, ci.exp( msp, list(np,nr) ), col="blue" )
> abline( h=1, v=1970 )
```

A spline period effect

Because we have the age-effect with the rate dimension, the period effect is a RR, the ratio of two predictions (`np` and `nr`):

```
> msps <- glm( cbind(D,Y) ~ Ns(A,knots=seq(15,65,10)) +
+                           Ns(P,knots=seq(1950,1990,10)),
+              family=poisreg, data=testisDK )
> round( ci.exp( msps ), 3 )
```

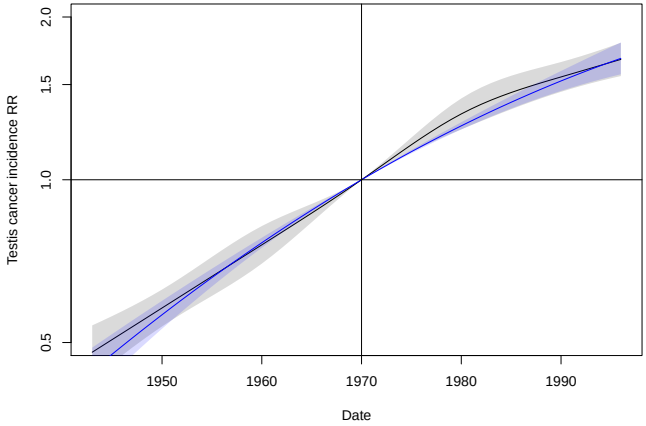
	exp(Est.)	2.5%	97.5%
(Intercept)	0.000	0.000	0.000
Ns(A, knots = seq(15, 65, 10))1	8.327	7.452	9.305
Ns(A, knots = seq(15, 65, 10))2	5.528	4.842	6.312
Ns(A, knots = seq(15, 65, 10))3	1.007	0.894	1.133
Ns(A, knots = seq(15, 65, 10))4	13.447	11.107	16.279
Ns(A, knots = seq(15, 65, 10))5	0.458	0.422	0.497
Ns(P, knots = seq(1950, 1990, 10))1	1.711	1.526	1.918
Ns(P, knots = seq(1950, 1990, 10))2	2.190	2.028	2.364
Ns(P, knots = seq(1950, 1990, 10))3	3.222	2.835	3.661
Ns(P, knots = seq(1950, 1990, 10))4	2.299	2.149	2.459

```
> matshade( np$P, ci.exp( msps, list(np,nr) ), plot=TRUE,
```

A spline period effect

```
> matshade( np$P, ci.exp( msps, list(np,nr) ), plot=TRUE,
+           log="y", ylim=c(0.5,2),
+           xlab="Date", ylab="Testis cancer incidence RR" )
> matshade( np$P, ci.exp( mspq, list(np,nr) ), col="blue" )
> abline( h=1, v=1970 )
```

Period effect

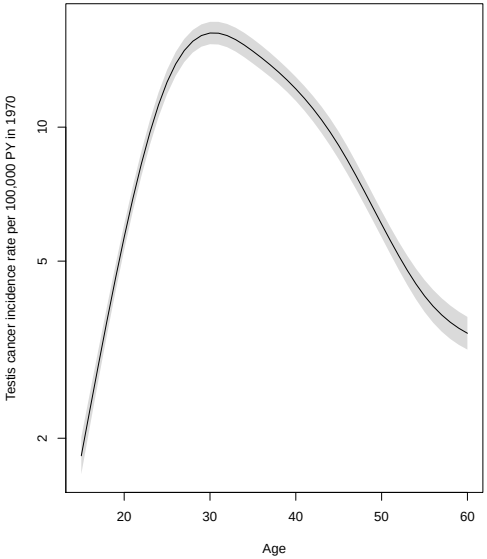


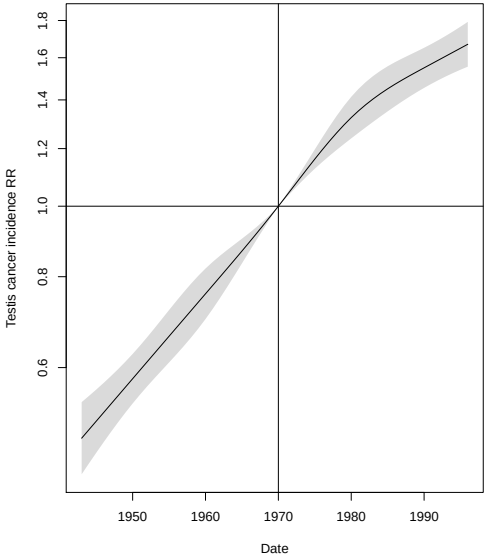

```
> matshade( np$P, ci.exp( msps, list(np,nr) ), plot=TRUE,
+           log="y", ylim=c(0.5,2),
+           xlab="Date", ylab="Testis cancer incidence RR" )
> matshade( np$P, ci.exp( mspq, list(np,nr) ), col="blue" )
> abline( h=1, v=1970 )
```

Period effect

```
> par( mfrow=c(1,2) )
> matshade( nd$A, ci.pred( msps, cbind(nd,P=nr$P) )*10^5, plot=TRUE, log="y", xlab="
+           ylab="Testis cancer incidence rate per 100,000 PY in 1970" )
> matshade( np$P, ci.exp( msps, list(np,nr) ), plot=TRUE,
+           log="y", xlab="Date", ylab="Testis cancer incidence RR" )
> abline( h=1, v=1970 )
```

Age and period effect

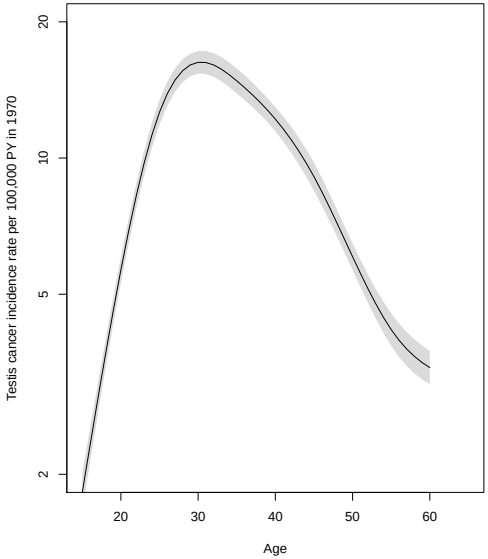


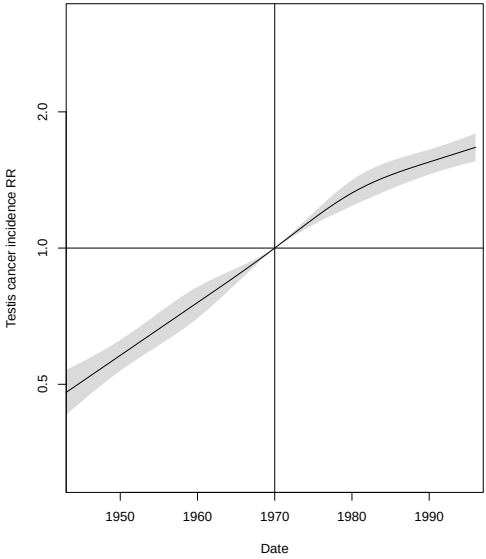


Period effect

```
> par( mfrow=c(1,2) )
> par( mfrow=c(1,2) )
> matshade( nd$A, ci.pred( msps, cbind(nd,P=nr$P) )*10^5, plot=TRUE, log="y", xlab="
+           ylim=c(2,20), xlim=c(15,65),
+           ylab="Testis cancer incidence rate per 100,000 PY in 1970" )
> matshade( np$P, ci.exp( msps, list(np,nr) ), plot=TRUE,
+           ylim=c(2,20)/sqrt(2*20), xlim=c(15,65)+1930,
+           log="y", xlab="Date", ylab="Testis cancer incidence RR" )
> abline( h=1, v=1970 )
```

Age and period effect





Age and period effect with `ci.exp`

- In rate models there is always one term with the **rate** dimension — usually **age**
- But it must refer to a specific **reference** value for **all other** variables (`P`).
- **All** parameters must be used in computing rates, at some reference value(s).
- For the "other" variables, report the RR **relative** to the reference point.
- Contrast matrix (2nd argument to `ci.exp`) is a **difference** between the prediction points and the reference point.