# Prevalence, mortality and survival in diabetes epidemiology

**Bendix Carstensen** Steno Diabetes Center Copenhagen
Herlev, Denmark

IDEG 2025 Training day, Bangkok, 3 April 2025

`http://bendixcarstensen.com/AdvCoh/courses/IDEG2025`

From `C:\Bendix\teach\AdvCoh\courses\IDEG2025\slides/slides.tex` Sunday 30th March, 2025, 08:22

# Prevalence: Tables

**Bendix Carstensen**

Prevalence, mortality and survival in diabetes epidemiology
IDEG 2025 Training day, Bangkok, 3 April 2025

# Prevalence

- the fraction of a population that suffer a particular condition, diabetes for example
- for several conditions (T1D, T2D) — mutually exclusive?
- at a given point in time (date, age, other time)
- depends on age, sex, . . .

# get the data

```
> nhis <- read.csv(
+ "https://bendixcarstensen.com/AdvCoh/courses/IDEG2025/data/NHIS_IDEG.csv",
+                    header = TRUE)
> str(nhis)
```

```
'data.frame':       29522 obs. of  10 variables:
 $ HHX        : chr  "H029691" "H028812" "H045277" "H021192" ...
 $ WTFA_A     : num  7371 3147 10808 4662 10930 ...
 $ SEX_A      : int  1 1 1 2 2 2 2 1 2 1 ...
 $ AGEP_A     : int  67 73 48 42 50 46 36 44 80 61 ...
 $ EDUCP_A    : int  1 8 5 9 7 8 8 10 8 1 ...
 $ DIBEV_A    : int  2 1 2 2 2 2 2 2 2 2 ...
 $ DIBAGETC_A : int  NA 61 NA NA NA NA NA NA NA NA ...
 $ DIFYRSTC1_A: int  NA 12 NA NA NA NA NA NA NA NA ...
 $ DIBTYPE_A  : int  NA 2 NA NA NA NA NA NA NA NA ...
 $ BMICAT_A   : int  3 3 4 3 2 3 2 4 4 3 ...
```

# make names simpler

```
> newn <- tolower(gsub("_A", "", names(nhis)))
> cbind(names(nhis), newn)
```

```
                             newn
 [1,] "HHX"                  "hhx"
 [2,] "WTFA_A"               "wtfa"
 [3,] "SEX_A"                "sex"
 [4,] "AGEP_A"               "agep"
 [5,] "EDUCP_A"              "educp"
 [6,] "DIBEV_A"              "dibev"
 [7,] "DIBAGETC_A"           "dibagetc"
 [8,] "DIFYRSTC1_A"          "difyrstc1"
 [9,] "DIBTYPE_A"            "dibtype"
[10,] "BMICAT_A"             "bmicat"
```

```
> names(nhis) <- newn
> str(nhis)
```

```
'data.frame':         29522 obs. of  10 variables:
 $ hhx      : chr  "H029691" "H028812" "H045277" "H021192" ...
 $ wtfa     : num  7371 3147 10808 4662 10930 ...
 $ sex      : int  1 1 1 2 2 2 2 1 2 1 ...
 $ agep     : int  67 73 48 42 50 46 36 44 80 61 ...
```

# sensible names for groups

```
> nhis <- mutate(nhis, dibev = factor(dibev, labels = c("Y","N","R","U")),
+                    dibtype = factor(dibtype, labels = c("T1","T2","O","O","O"))
+                        agr = cut(agep, seq(0, 100, 10), right = FALSE),
+                        sex = factor(sex,  labels = c("M","W","U","U")))
> str(nhis)
```

```
'data.frame':        29522 obs. of  11 variables:
 $ hhx      : chr  "H029691" "H028812" "H045277" "H021192" ...
 $ wtfa     : num  7371 3147 10808 4662 10930 ...
 $ sex      : Factor w/ 3 levels "M","W","U": 1 1 1 2 2 2 2 1 2 1 ...
 $ agep     : int  67 73 48 42 50 46 36 44 80 61 ...
 $ educp    : int  1 8 5 9 7 8 8 10 8 1 ...
 $ dibev    : Factor w/ 4 levels "Y","N","R","U": 2 1 2 2 2 2 2 2 2 2 ...
 $ dibagetc : int  NA 61 NA NA NA NA NA NA NA NA ...
 $ difyrstc1: int  NA 12 NA NA NA NA NA NA NA NA ...
 $ dibtype  : Factor w/ 3 levels "T1","T2","O": NA 2 NA NA NA NA NA NA NA NA ...
 $ bmicat   : int  3 3 4 3 2 3 2 4 4 3 ...
 $ agr      : Factor w/ 10 levels "[0,10)","[10,20)",..: 7 8 5 5 6 5 4 5 9 7 ...
```

# table of overall prevalence

```
> (tb <- with(nhis, table(dibev, exclude = NULL)))
```

```
dibev
    Y     N     R     U
 3294 26195    23    10
```

```
> tb["Y"] / (tb["Y"] + tb["N"]) * 100
```

```
       Y
11.17027
```

# table of prevalence by age

```
> with(nhis, table(Age = agr,
+                  Diabetes = dibev,
+                   exclude = NULL)) |> addmargins() -> diab
> diab
```

```
          Diabetes
Age            Y     N     R     U   Sum
  [0,10)       0     0     0     0     0
  [10,20)      3   426     0     0   429
  [20,30)     44  3308     1     0  3353
  [30,40)    132  4534     0     1  4667
  [40,50)    281  3863     4     1  4149
  [50,60)    528  3953     4     0  4485
  [60,70)    976  4596     5     1  5578
  [70,80)    920  3556     5     3  4484
  [80,90)    408  1899     1     4  2312
  [90,100)     2    60     3     0    65
  Sum       3294 26195    23    10 29522
```

. . . we want to get rid of `R` and `U` categories

# table by age

```
> (diab <- addmargins(diab[, 1:2], 2))

          Diabetes
Age            Y     N   Sum
  [0,10)       0     0     0
  [10,20)      3   426   429
  [20,30)     44  3308  3352
  [30,40)    132  4534  4666
  [40,50)    281  3863  4144
  [50,60)    528  3953  4481
  [60,70)    976  4596  5572
  [70,80)    920  3556  4476
  [80,90)    408  1899  2307
  [90,100)     2    60    62
  Sum       3294 26195 29489

> cbind(round(diab[,"Y"] / diab[,"Sum"] * 100, 1))

          [,1]
[0,10)     NaN
[10,20)    0.7
[20,30)    1.3
[30,40)    2.8
```

# table of % by age

```
> cbind(round(diab[,"Y"] / diab[,"Sum"] * 100, 1))
          [,1]
[0,10)     NaN
[10,20)    0.7
[20,30)    1.3
[30,40)    2.8
[40,50)    6.8
[50,60)   11.8
[60,70)   17.5
[70,80)   20.6
[80,90)   17.7
[90,100)   3.2
Sum       11.2
```

# table by age and type

```
> with(nhis, table(agr, dibtype, exclude = NULL)) |> addmargins() -> dtyp
> dtyp
          dibtype
agr           T1    T2     0  <NA>   Sum
  [0,10)       0     0     0     0     0
  [10,20)      0     2     1   426   429
  [20,30)     23    15     6  3309  3353
  [30,40)     36    80    16  4535  4667
  [40,50)     21   236    24  3868  4149
  [50,60)     43   447    38  3957  4485
  [60,70)     73   849    54  4602  5578
  [70,80)     49   818    53  3564  4484
  [80,90)     24   341    43  1904  2312
  [90,100)     0     1     1    63    65
  Sum        269  2789   236 26228 29522
```

# table by age

```
> round(100 * dtyp[, 1:2] / dtyp[, "Sum"], 1)
           dibtype
agr            T1   T2
  [0,10)
  [10,20)   0.0  0.5
  [20,30)   0.7  0.4
  [30,40)   0.8  1.7
  [40,50)   0.5  5.7
  [50,60)   1.0 10.0
  [60,70)   1.3 15.2
  [70,80)   1.1 18.2
  [80,90)   1.0 14.7
  [90,100)  0.0  1.5
  Sum       0.9  9.4
```

# Prevalence: Probability

**Bendix Carstensen**

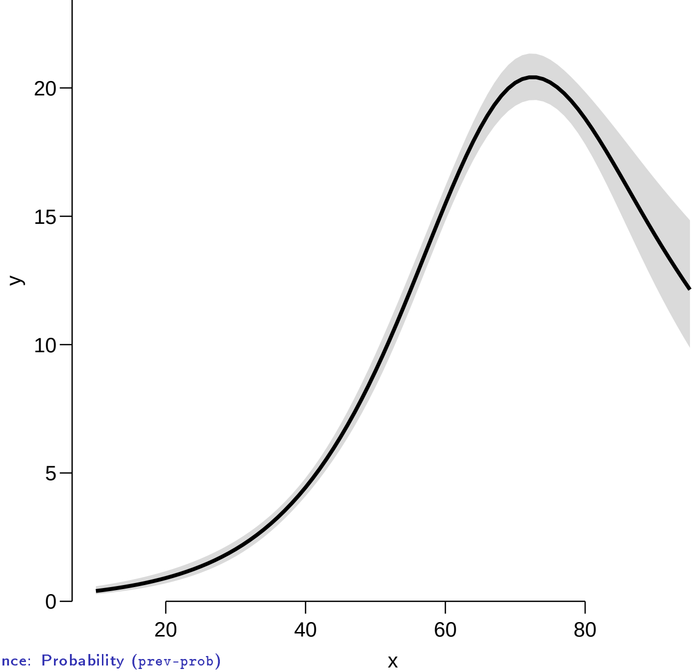
Prevalence, mortality and survival in diabetes epidemiology
IDEG 2025 Training day, Bangkok, 3 April 2025

# Prevalence and probability

- prevalence defined as the **fraction** of a population that suffers from a given disease
- theoretically, prevalence can be defined as the **probability** that a randomly selected person has the disease
- opens the possibility of statistical modeling to address the question on how the prevalence of diabetes depends on age and sex, for example
- **statistical model**: a probability machinery that could have generated data
- properties of the machinery assumed to be properties of the real world

# probability model with agep

```
> nh <- subset(nhis, dibev %in% c("Y", "N") & sex %in% c("M", "W"))
> # model
> ma <- glm((dibev == "Y") ~ Ns(agep, knots = seq(30, 90,, 4)),
+           family = binomial,
+           data = nh)
> #
> # prediction frame
> da <- data.frame(agep = 10:95)
> head(pa <- ci.pred(ma, da) * 100)

   Estimate      2.5%     97.5%
1 0.4034395 0.2770811 0.5870824
2 0.4375991 0.3041219 0.6292888
3 0.4746372 0.3337856 0.6745235
4 0.5147940 0.3663231 0.7230036
5 0.5583292 0.4020088 0.7749616
6 0.6055237 0.4411424 0.8306467

> # plot prediction
> matshade(da[,"agep"], pa, plot = TRUE, lwd = 3, ylim = c(0, 25), yaxs = "i")
```

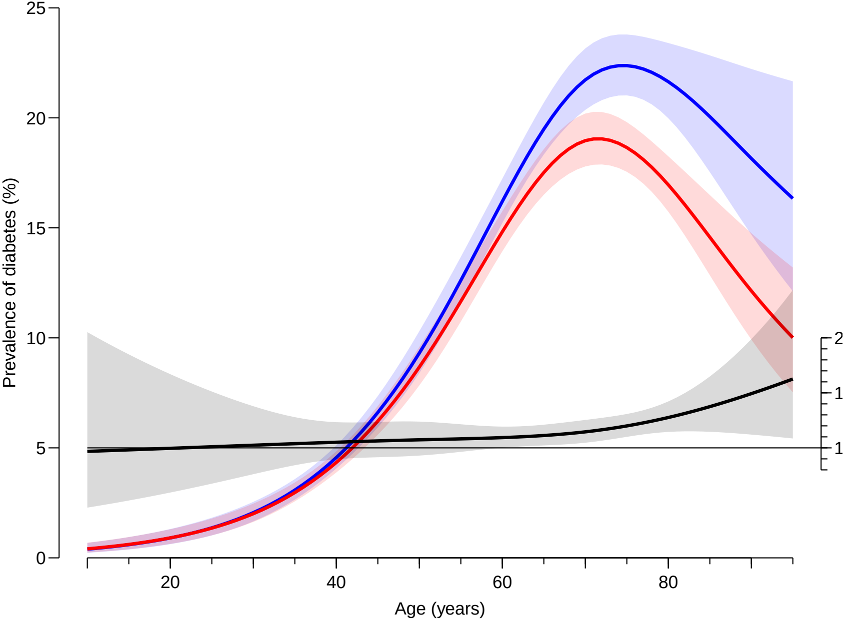

20
15
10
5
0
y
20
40
60
80
x

# model separately for men and women

```
> # model for men
> Ma <- glm((dibev == "Y") ~ Ns(agep, knots = seq(30, 90,, 4)),
+                    family = binomial,
+                    data = subset(nh, sex == "M"))
> # model for women
> Wa <- update(Ma, data = subset(nh, sex == "W"))
> #
> # predicted prevalences
> pM <- ci.pred(Ma, da)
> pW <- ci.pred(Wa, da)
> #
> # ratio of predicted prevalences
> mw <- ci.ratio(pM, pW)
```

# plot prevalences for men and women

```
> # plot prevalences for men and women
> matshade(da$agep, cbind(pM, pW) * 100,
+          plot = TRUE,
+          xlab = "Age (years)",
+          ylab = "Prevalence of diabetes (%)", ylim = c(0, 25), yaxs = "i",
+          col = c("blue", "red"), lwd = 3)
> axis(side = 1, at = seq(15, 95, 5), labels = NA, tcl = -0.3)
> axis(side = 1, at = seq(10, 90,10), labels = NA, tcl = -0.5)
> #
> # plot ratio of prevalences
> matshade(da$agep, mw * 5, lwd =3)
> lines(c(10,100), c(5,5))
> axis(side = 4, at = c(1, 1.5, 2) * 5, labels = c(1, 1.5, 2))
> axis(side = 4, at = seq(0.8, 2, 0.1) * 5, labels = NA, tcl = -0.3)
```


25
20
15
10
5
0
Prevalence of diabetes (%)
20
40
60
80
Age (years)
2
1
1

# Mortality: concepts

Bendix Carstensen

Prevalence, mortality and survival in diabetes epidemiology
IDEG 2025 Training day, Bangkok, 3 April 2025

`http://bendixcarstensen.com/AdvCoh/courses/IDEG2025` `mort-conc`

# mortality: concepts

- mortality rate is the “force of mortality”
- speed with which deaths occur
- a time aspect—how **long** have people been at risk of dying
- in practice:
  - how many persons for how long time (the **risk time** or “**person-years**”)
  - how many deaths (events) have occurred.
- the **individual** level we need to know how long time a person has been at risk of dying
- a person is at risk at a given time if she would have been recorded as dead had she died at the time

# mortality: concepts

- at the **individual** level we need to know how long time a person has been at risk of dying
- at the **theoretical** level we need a precise (probabilistic) definition of mortality
- a mortality **rate** is defined as the conditional probability of death in a small interval, conditional on
  —being alive at a given time $t$
  —divided by the length of the risk interval $h$:

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

- something that relates to the **time point** $t$
- in pratise we will use some small value of $h$

# mortality: concepts

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

- the $t$ here is the time**scale**
  —**when** the person is at risk;
- the $h$ is the **risk** time
  —**how long** the person has been at risk
- The rate has dimension time$^{-1}$—probability/events per time.
- The mortality is a function of $t$, but one possibility for this function is that it is constant, the same at all times:
  $\lambda(t) = \lambda \ \forall t$.

# mortality: data

```
> data(DMlate)
> set.seed(1952)
> DMlate <- DMlate[sample(1:nrow(DMlate), 2000),]
> rownames(DMlate) <- 1:2000
> 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 NA NA NA NA NA NA NA NA NA ...
 $ dooad: num  NA 2006 NA 2007 2006 ...
 $ doins: num  NA NA NA 2008 NA ...
 $ dox  : num  2010 2010 2010 2010 2010 ...
```

```
> head(DMlate)
```

```
  sex    dobth     dodm dodth    dooad    doins      dox
1   F 1963.591 2003.481    NA       NA       NA 2009.997
2   M 1944.127 2005.644    NA 2005.778       NA 2009.997
3   F 1956.790 2007.886    NA       NA       NA 2009.997
4   M 1952.355 2006.969    NA 2006.969 2008.026 2009.997
5   M 1952.240 2003.361    NA 2005.852       NA 2009.997
```

# mortality: risk time and deaths

```
> (y <- with(DMlate, sum(dox - dodm)))
[1] 10742.34
> (d <- with(DMlate, sum(!is.na(dodth))))
[1] 479
> d / y
[1] 0.04458991
> round(d / y * 100, 1)
[1] 4.5
```

What are the units of these numbers?

# mortality: risk time and deaths

```
> tt <- xtabs(cbind(D = !is.na(dodth),
+                   Y = dox - dodm) ~
+                   agr,
+             data = mutate(DMlate,
+                           agr = cut(dodm - dobth,
+                                     seq(0, 100, 10),
+                                     right = FALSE)))
> tt
```

```
agr                D          Y
  [0,10)     0.00000  104.37235
  [10,20)    1.00000  146.11088
  [20,30)    0.00000  271.86037
  [30,40)    3.00000  710.50513
  [40,50)   14.00000 1503.36208
  [50,60)   55.00000 2323.28268
  [60,70)   99.00000 2942.04244
  [70,80)  188.00000 2027.45517
  [80,90)   98.00000  660.26557
  [90,100)  21.00000   53.08419
```

# mortality: risk time and deaths

```
> cbind(mort = tt[, "D"] / tt[, "Y"] * 100)
                 mort
[0,10)     0.0000000
[10,20)    0.6844117
[20,30)    0.0000000
[30,40)    0.4222348
[40,50)    0.9312461
[50,60)    2.3673400
[60,70)    3.3650092
[70,80)    9.2727081
[80,90)   14.8425125
[90,100)  39.5598019
```

# Mortality: models for age

**Bendix Carstensen**

Prevalence, mortality and survival in diabetes epidemiology
IDEG 2025 Training day, Bangkok, 3 April 2025

# age and age is not the same

- What we have done is to classify follow-up (deaths and risk time) by the age **at diagnosis**
- more relevant to classify the follow-up by **current** age (also called **attained** age)
  —the age of the person as it changes during follow-up
- requires follow-up for each person to be classified according to current age, so persons would potentially contribute follow-up in more than one age class
- how large should the age-classes be?

# current age as time scale

```
> Lx <- Lexis(entry = list(age = dodm - dobth),
+               exit = list(age = dox  - dobth),
+        exit.status = factor(!is.na(dodth), labels = c("Alive","Dead")),
+               data = DMlate)

NOTE: entry.status has been set to "Alive" for all.
NOTE: Dropping  1  rows with duration of follow up < tol

> subset(DMlate, near(dodm, dox))

     sex    dobth     dodm    dodth dooad doins      dox
1895   F 1936.067 1996.984 1996.984    NA    NA 1996.984

> summary(Lx)

Transitions:
     To
From    Alive Dead  Records:  Events: Risk time:  Persons:
  Alive  1521  478      1999      478   10742.34      1999
```

# current age as time scale

```
> summary(Lx)

Transitions:
     To
From    Alive Dead  Records:  Events: Risk time:  Persons:
  Alive  1521  478      1999       478   10742.34      1999

> head(Lx, 10)

 lex.id   age lex.dur lex.Cst lex.Xst sex   dobth    dodm dodth   dooad   doins  d
      1 39.89    6.52   Alive   Alive   F 1963.59 2003.48    NA      NA      NA 2
      2 61.52    4.35   Alive   Alive   M 1944.13 2005.64    NA 2005.78      NA 2
      3 51.10    2.11   Alive   Alive   F 1956.79 2007.89    NA      NA      NA 2
      4 54.61    3.03   Alive   Alive   M 1952.35 2006.97    NA 2006.97 2008.03 2
      5 51.12    6.64   Alive   Alive   M 1952.24 2003.36    NA 2005.85      NA 2
      6 23.19    8.05   Alive   Alive   M 1978.76 2001.95    NA      NA 2001.97 2
      7 47.14    4.93   Alive   Alive   M 1957.93 2005.07    NA      NA      NA 2
      8 82.15    4.67   Alive   Alive   M 1923.17 2005.32    NA 2005.49      NA 2
      9 60.18    6.73   Alive   Alive   M 1943.09 2003.27    NA      NA      NA 2
     10 46.74    5.29   Alive   Alive   M 1957.96 2004.70    NA      NA      NA 2
```

# current age as time scale

```
> str(Lx)
```

```
Classes 'Lexis' and 'data.frame':       1999 obs. of  12 variables:
 $ age    : num  39.9 61.5 51.1 54.6 51.1 ...
 $ lex.dur: num  6.52 4.35 2.11 3.03 6.64 ...
 $ lex.Cst: Factor w/ 2 levels "Alive","Dead": 1 1 1 1 1 1 1 1 1 1 ...
 $ lex.Xst: Factor w/ 2 levels "Alive","Dead": 1 1 1 1 1 1 1 1 1 1 ...
 $ lex.id : int  1 2 3 4 5 6 7 8 9 10 ...
 $ 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 NA NA NA NA NA NA NA NA NA ...
 $ dooad  : num  NA 2006 NA 2007 2006 ...
 $ doins  : num  NA NA NA 2008 NA ...
 $ dox    : num  2010 2010 2010 2010 2010 ...
 - attr(*, "time.scales")= chr "age"
 - attr(*, "time.since")= chr ""
 - attr(*, "breaks")=List of 1
  ..$ age: NULL
```

what does `str(Lx)` tell you?

# current age as time scale

splitting follow-up along current age

```
> Sx <- splitLexis(Lx, breaks = seq(0, 100, 5))
> summary(Lx)

Transitions:
     To
From    Alive Dead  Records:  Events: Risk time:  Persons:
  Alive  1521  478      1999       478   10742.34      1999

> summary(Sx)

Transitions:
     To
From    Alive Dead  Records:  Events: Risk time:  Persons:
  Alive  3656  478      4134       478   10742.34      1999
```

# current age as time scale

tabulate deaths and risk-time by current age:

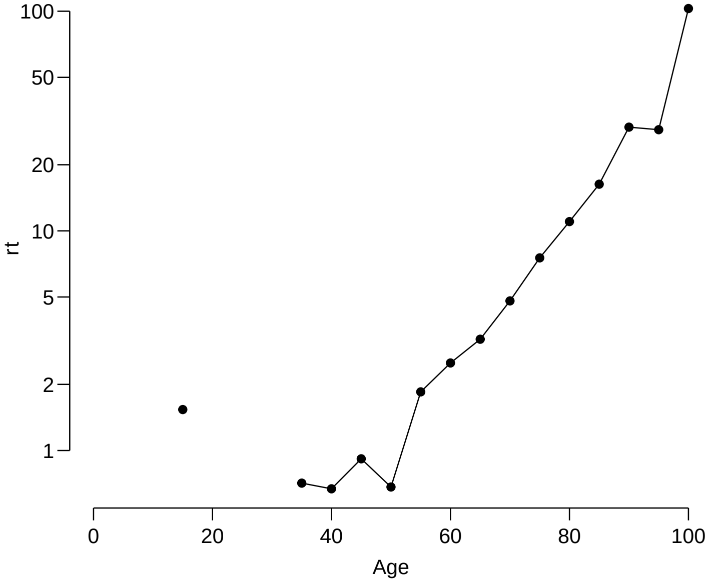
```
> tt <- xtabs(cbind(D = lex.Xst == "Dead",
+                   Y = lex.dur) ~
+                   I(floor(age / 5) * 5),
+             data = Sx)
> tt

I(floor(age/5) * 5)           D           Y
                0     0.000000   13.258727
                5     0.000000   44.838467
                10    0.000000   81.636550
                15    1.000000   65.104038
                20    0.000000   75.470910
                25    0.000000   97.076660
                30    0.000000  203.124572
                35    2.000000  281.568104
                40    3.000000  448.275838
                45    6.000000  654.642710
                50    6.000000  879.850787
                55   22.000000 1189.978097
                60   33.000000 1318.171116
```

# deaths and risk-time by current age

```
> (rt <- cbind(mort = tt[, "D"] / tt[, "Y"] * 100))
          mort
0    0.0000000
5    0.0000000
10   0.0000000
15   1.5360030
20   0.0000000
25   0.0000000
30   0.0000000
35   0.7103077
40   0.6692308
45   0.9165305
50   0.6819338
55   1.8487735
60   2.5034686
65   3.2105086
70   4.7989375
75   7.5337874
80  11.0200101
85  16.3104372
90  29.6655924
```

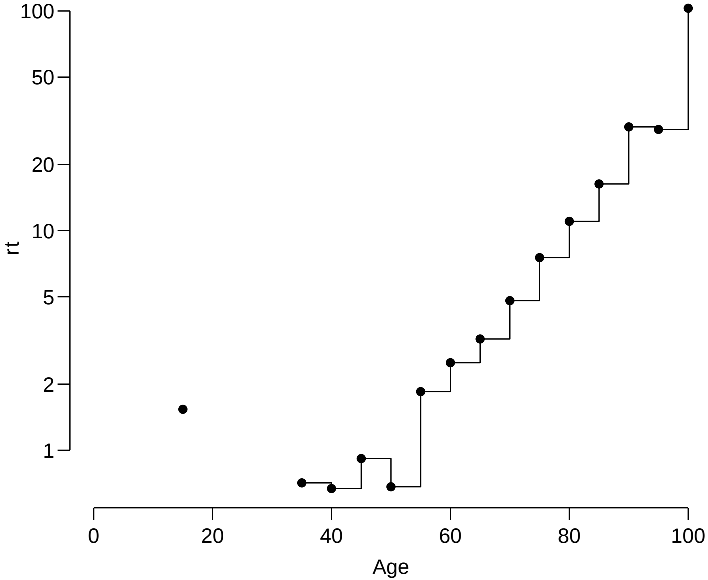
# current age as time scale

```
> plot(rownames(tt), rt, log ="y", type = "o", xlab = "Age", pch = 16)
```


100
50
20
10
5
2
1
rt
0
20
40
60
80
100
Age

# current age as time scale

```
> plot(rownames(tt), rt, log ="y", type = "s", xlab = "Age")
> points(rownames(tt), rt, pch = 16)
```


100
50
20
10
5
2
1
rt
0
20
40
60
80
100
Age

# smooth model for age

- the assumption of constant rates in 5 year intervals is a bit coarse, 1 year would be a more reasonable approximation
- we estimated 21 parameters (one per 5 year age class)
- highly unrealistic that wee would need 101 parameters to describe mortality by age.
- solution: put a parametric restriction on mortality rates in the 1-year age classes
- parametric modeling also has the advantage that we do not need to tabulate data, we can directly fit a model for age to the split data.

# smooth model for age: 1 year classes

```
> Sx <- splitLexis(Lx, breaks = 0:100, time.scale = "age")
> summary(Lx)

Transitions:
     To
From    Alive Dead  Records:  Events: Risk time:  Persons:
  Alive  1521  478      1999       478   10742.34      1999

> summary(Sx)

Transitions:
     To
From    Alive Dead  Records:  Events: Risk time:  Persons:
  Alive 12201  478     12679       478   10742.34      1999
```

# smooth model for age

```
> subset(Lx, lex.id %in% 16:17)

 lex.id   age lex.dur lex.Cst lex.Xst sex   dobth    dodm   dodth dooad doins
     16 66.97    3.85   Alive   Alive   F 1939.18 2006.15      NA    NA    NA 201
     17 73.11    2.93   Alive    Dead   F 1932.69 2005.79 2008.72    NA    NA 200

> subset(Sx, lex.id %in% 16:17)

 lex.id   age lex.dur lex.Cst lex.Xst sex   dobth    dodm   dodth dooad doins
     16 66.97    0.03   Alive   Alive   F 1939.18 2006.15      NA    NA    NA 201
     16 67.00    1.00   Alive   Alive   F 1939.18 2006.15      NA    NA    NA 201
     16 68.00    1.00   Alive   Alive   F 1939.18 2006.15      NA    NA    NA 201
     16 69.00    1.00   Alive   Alive   F 1939.18 2006.15      NA    NA    NA 201
     16 70.00    0.82   Alive   Alive   F 1939.18 2006.15      NA    NA    NA 201
     17 73.11    0.89   Alive   Alive   F 1932.69 2005.79 2008.72    NA    NA 200
     17 74.00    1.00   Alive   Alive   F 1932.69 2005.79 2008.72    NA    NA 200
     17 75.00    1.00   Alive   Alive   F 1932.69 2005.79 2008.72    NA    NA 200
     17 76.00    0.04   Alive    Dead   F 1932.69 2005.79 2008.72    NA    NA 200
```

# smooth model for age

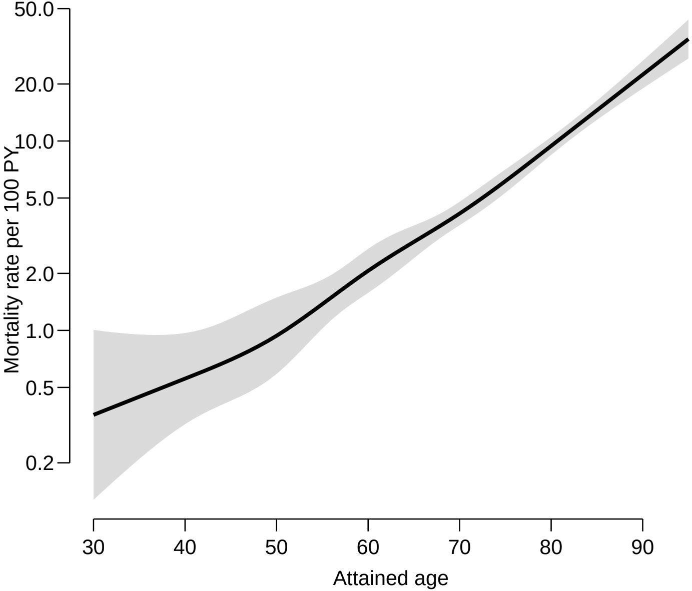
```
> mL <- glmLexis(Sx, ~ Ns(age, knots = seq(40, 80, 10)))

stats::glm Poisson analysis of Lexis object Sx with log link:
Rates for the transition:
Alive->Dead


> mP <- glm((lex.Xst == "Dead") ~ Ns(age, knots = seq(40, 80, 10)),
+           offset = log(lex.dur),
+           family = poisson,
+           data = Sx)
> round(cbind(ci.exp(mL),
+             ci.exp(mP)), 4)

                                   exp(Est.)   2.5%   97.5% exp(Est.)   2.5%   97.5
(Intercept)                           0.0056 0.0032  0.0097    0.0056 0.0032  0.009
Ns(age, knots = seq(40, 80, 10))1     3.9144 1.7492  8.7598    3.9144 1.7490  8.760
Ns(age, knots = seq(40, 80, 10))2     6.2849 3.7176 10.6249    6.2849 3.7173 10.625
Ns(age, knots = seq(40, 80, 10))3    18.3976 7.0794 47.8107   18.3976 7.0743 47.845
Ns(age, knots = seq(40, 80, 10))4    13.2045 7.7908 22.3800   13.2045 7.7907 22.380
```

# smooth model for age: predicted rates

```
> nd <- data.frame(age = 30:95)
> pr.rates <- ci.pred(mL, nd) * 100
> head(pr.rates)
```

```
   Estimate      2.5%     97.5%
1 0.3582582 0.1275648 1.0061469
2 0.3743926 0.1413134 0.9919077
3 0.3912537 0.1563451 0.9791130
4 0.4088741 0.1727125 0.9679558
5 0.4272881 0.1904434 0.9586845
6 0.4465314 0.2095274 0.9516190
```

```
> matshade(nd$age, pr.rates, plot = TRUE, log = "y", lwd = 3,
+          xlab = "Attained age", ylab = "Mortality rate per 100 PY")
```


50.0
20.0
10.0
5.0
2.0
1.0
0.5
0.2
Mortality rate per 100 PY
30
40
50
60
70
80
90
Attained age

# Mortality: models for duration

**Bendix Carstensen**

Prevalence, mortality and survival in diabetes epidemiology
IDEG 2025 Training day, Bangkok, 3 April 2025

# smooth model for age

with age as covariate:

```
> Lx <- Lexis(entry = list(age = dodm - dobth),
+              exit = list(age = dox  - dobth)               )
```

with time from diagnosis as covariate:

```
> Lx <- Lexis(entry = list(tfd = dodm - dodm),
+              exit = list(tfd = dox  - dodm),
+       exit.status = factor(!is.na(dodth), labels = c("Alive", "Dead")),
+              data = DMlate)
NOTE: entry.status has been set to "Alive" for all.
NOTE: Dropping  1  rows with duration of follow up < tol
> summary(Lx)
Transitions:
     To
From    Alive Dead  Records:  Events: Risk time:  Persons:
  Alive  1521  478      1999      478   10742.34      1999
```

```
> head(Lx, 17)
 lex.id tfd lex.dur lex.Cst lex.Xst sex   dobth    dodm   dodth   dooad   doins
      1   0    6.52   Alive   Alive   F 1963.59 2003.48      NA      NA      NA 2
      2   0    4.35   Alive   Alive   M 1944.13 2005.64      NA 2005.78      NA 2
      3   0    2.11   Alive   Alive   F 1956.79 2007.89      NA      NA      NA 2
      4   0    3.03   Alive   Alive   M 1952.35 2006.97      NA 2006.97 2008.03 2
      5   0    6.64   Alive   Alive   M 1952.24 2003.36      NA 2005.85      NA 2
      6   0    8.05   Alive   Alive   M 1978.76 2001.95      NA      NA 2001.97 2
      7   0    4.93   Alive   Alive   M 1957.93 2005.07      NA      NA      NA 2
      8   0    4.67   Alive   Alive   M 1923.17 2005.32      NA 2005.49      NA 2
      9   0    6.73   Alive   Alive   M 1943.09 2003.27      NA      NA      NA 2
     10   0    5.29   Alive   Alive   M 1957.96 2004.70      NA      NA      NA 2
     11   0    9.20   Alive   Alive   M 1948.25 2000.80      NA      NA      NA 2
     12   0   11.86   Alive    Dead   M 1934.42 1995.10 2006.96      NA      NA 2
     13   0    6.61   Alive   Alive   M 1933.24 2003.38      NA 2005.63      NA 2
     14   0   13.94   Alive   Alive   F 1938.97 1996.06      NA      NA      NA 2
     15   0    0.43   Alive    Dead   M 1927.43 2004.84 2005.27      NA      NA 2
     16   0    3.85   Alive   Alive   F 1939.18 2006.15      NA      NA      NA 2
     17   0    2.93   Alive    Dead   F 1932.69 2005.79 2008.72      NA      NA 2
```

# splitting along duration

```
> Sx <- splitLexis(Lx, breaks = seq(0, 20, 0.5))
> summary(Lx)

Transitions:
     To
From    Alive Dead  Records:  Events: Risk time:  Persons:
  Alive  1521  478      1999      478   10742.34      1999

> summary(Sx)

Transitions:
     To
From    Alive Dead  Records:  Events: Risk time:  Persons:
  Alive 22020  478     22498      478   10742.34      1999
```

```
> subset(Lx, lex.id %in% 15:17)

 lex.id tfd lex.dur lex.Cst lex.Xst sex   dobth    dodm   dodth dooad doins      d
     15   0    0.43   Alive    Dead   M 1927.43 2004.84 2005.27    NA    NA 2005.
     16   0    3.85   Alive   Alive   F 1939.18 2006.15      NA    NA    NA 2010.
     17   0    2.93   Alive    Dead   F 1932.69 2005.79 2008.72    NA    NA 2008.

> subset(Sx, lex.id %in% 15:17)

 lex.id tfd lex.dur lex.Cst lex.Xst sex   dobth    dodm   dodth dooad doins      d
     15 0.0    0.43   Alive    Dead   M 1927.43 2004.84 2005.27    NA    NA 2005.
     16 0.0    0.50   Alive   Alive   F 1939.18 2006.15      NA    NA    NA 2010.
     16 0.5    0.50   Alive   Alive   F 1939.18 2006.15      NA    NA    NA 2010.
     16 1.0    0.50   Alive   Alive   F 1939.18 2006.15      NA    NA    NA 2010.
     16 1.5    0.50   Alive   Alive   F 1939.18 2006.15      NA    NA    NA 2010.
     16 2.0    0.50   Alive   Alive   F 1939.18 2006.15      NA    NA    NA 2010.
     16 2.5    0.50   Alive   Alive   F 1939.18 2006.15      NA    NA    NA 2010.
     16 3.0    0.50   Alive   Alive   F 1939.18 2006.15      NA    NA    NA 2010.
     16 3.5    0.35   Alive   Alive   F 1939.18 2006.15      NA    NA    NA 2010.
     17 0.0    0.50   Alive   Alive   F 1932.69 2005.79 2008.72    NA    NA 2008.
     17 0.5    0.50   Alive   Alive   F 1932.69 2005.79 2008.72    NA    NA 2008.
     17 1.0    0.50   Alive   Alive   F 1932.69 2005.79 2008.72    NA    NA 2008.
     17 1.5    0.50   Alive   Alive   F 1932.69 2005.79 2008.72    NA    NA 2008.
     17 2.0    0.50   Alive   Alive   F 1932.69 2005.79 2008.72    NA    NA 2008.
     17 2.5    0.43   Alive    Dead   F 1932.69 2005.79 2008.72    NA    NA 2008.
```

# model the effect of time since diagnosis

```
> tL <- glmLexis(Sx, ~ Ns(tfd, knots = c(0, 1, 3, 6, 10)))
stats::glm Poisson analysis of Lexis object Sx with log link:
Rates for the transition:
Alive->Dead
> round(ci.exp(tL), 3)
                                      exp(Est.)  2.5% 97.5%
(Intercept)                               0.061 0.048 0.077
Ns(tfd, knots = c(0, 1, 3, 6, 10))1       0.774 0.517 1.159
Ns(tfd, knots = c(0, 1, 3, 6, 10))2       0.944 0.662 1.344
Ns(tfd, knots = c(0, 1, 3, 6, 10))3       0.425 0.221 0.817
Ns(tfd, knots = c(0, 1, 3, 6, 10))4       1.190 0.899 1.574
```

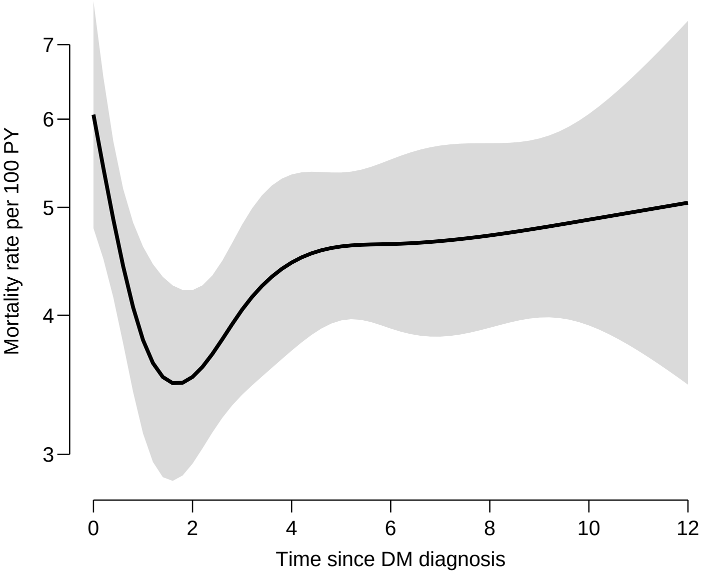
# predicted mortality rates

plot the mortality rates, but now as a function of time since diagnosis:

```
> nd <- data.frame(tfd = seq(0, 12, 0.2))
> pr.rates <- ci.pred(tL, nd)
> head(pr.rates)
```

```
    Estimate       2.5%      97.5%
1 0.06056585 0.04788313 0.07660781
2 0.05424695 0.04494245 0.06547778
3 0.04879401 0.04150913 0.05735739
4 0.04426341 0.03771416 0.05194997
5 0.04066812 0.03413191 0.04845600
6 0.03800474 0.03132808 0.04610433
```

```
> matshade(nd$tfd, pr.rates * 100, plot = TRUE,
+          log = "y", lwd = 3,
+          xlab = "Time since DM diagnosis",
+          ylab = "Mortality rate per 100 PY")
```



7
6
5
4
3
Mortality rate per 100 PY
0
2
4
6
8
10
12
Time since DM diagnosis

# survival with diabetes

- data is follow-up of a random sample of diabetes patients from the date of diagnosis of diabetes
- natural to ask for the survival probability as a function of time from diagnosis.
- link between
  a) mortality, $\lambda(t)$ and the origin $t = 0$ on one hand and
  b) the survival function $S(t)$ on the other hand

  is:

$$S(t) = \exp\Big( - \int_0^t \lambda(u)\,\mathrm{d}u \Big)$$

# survival and mortality

$$S(t) = \exp\Big( - \int_0^t \lambda(u)\,\mathrm{d}u \Big)$$

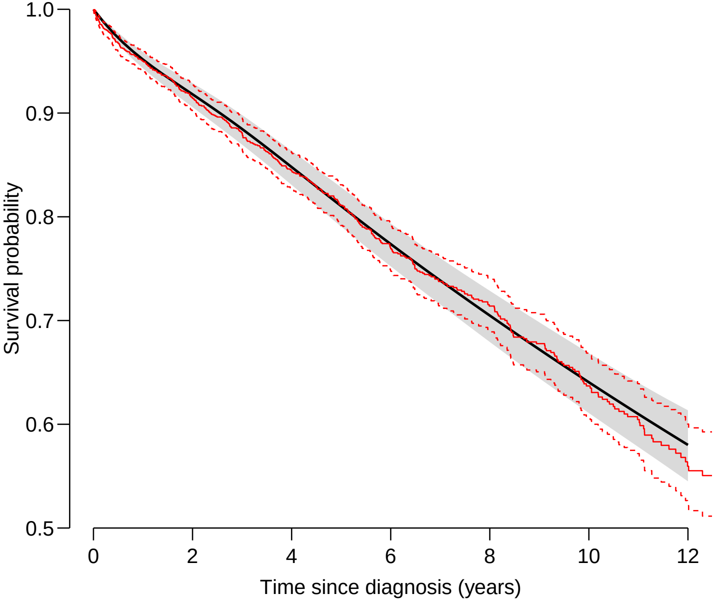
...easy to derive the survival function by numerical integration of the mortality in `pr.rates[,1]`

```
> head(cbind(nd, pr.rates))

  tfd   Estimate       2.5%      97.5%
1 0.0 0.06056585 0.04788313 0.07660781
2 0.2 0.05424695 0.04494245 0.06547778
3 0.4 0.04879401 0.04150913 0.05735739
4 0.6 0.04426341 0.03771416 0.05194997
5 0.8 0.04066812 0.03413191 0.04845600
6 1.0 0.03800474 0.03132808 0.04610433

> surv <- exp(-cumsum(pr.rates[, 1]) * 0.2)
> head(surv)

        1         2         3         4         5         6
0.9879599 0.9772991 0.9678082 0.9592783 0.9515075 0.9443026
```

# survival and mortality

- tricky to get a confidence interval for the survival function
- implemented in the function `ci.surv`
- for comparison we overlay the Kaplan-Meier estimate of the survival function (`survfit()`)

```
> matshade(nd$tfd, ci.surv(tL, nd), plot = TRUE,
+          lwd = 2, ylim = c(0.5,1), yaxs = "i",
+          xlab = "Time since diagnosis (years)",
+          ylab = "Survival probability")
NOTE: interval length chosen from  as tfd[2] - tfd[1]
> lines(survfit(Surv(dox - dodm, !is.na(dodth)) ~ 1, data = Lx), col = "red")
```


1.0
0.9
0.8
0.7
0.6
0.5
Survival probability
0
2
4
6
8
10
12
Time since diagnosis (years)

# survival and mortality

An exposition of how to use parametric rate models is in the document "Who needs the Cox model anyway?" at `https://bendixcarstensen.com/WntCma.pdf`.

The document also explains how the Cox-model is a special case of the Poisson model.

Further reading in the `Epi` package vignettes:

```
> vignette(package = "Epi")
> # vignette("aaflup", package = "Epi")
```