Competing risks with `ci.Crisk`

Bendix Carstensen Steno Diabetes Center
Gentofte, Denmark
http://BendixCarstensen.com

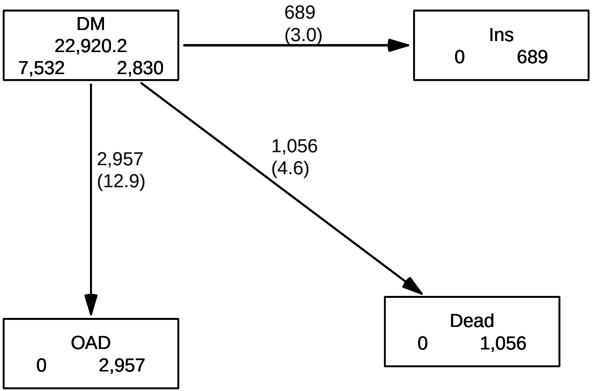
SDCC - R study group, 16 April 2021

From /home/bendix/teach/AdvCoh/00/slides/crisk.tex

Friday 16th April, 2021, 09:22

Vignette in the Epi:
Competing risks with Lexis, parametric rates and simulation based confidence intervals

or in http://BendixCarstensen.com/Epi



Competing risks

. . . is where persons are subject to a number of different causes of death, “cause1”, “cause2” etc.

- Causes of death are required to be **exhaustive** and mutually **exclusive**:
    - you will eventually die from one of the designated causes
    - you can only die from one.
- The observed data will be a survival time and an exit status which is either “censored alive” or one of the causes.

Competing risks

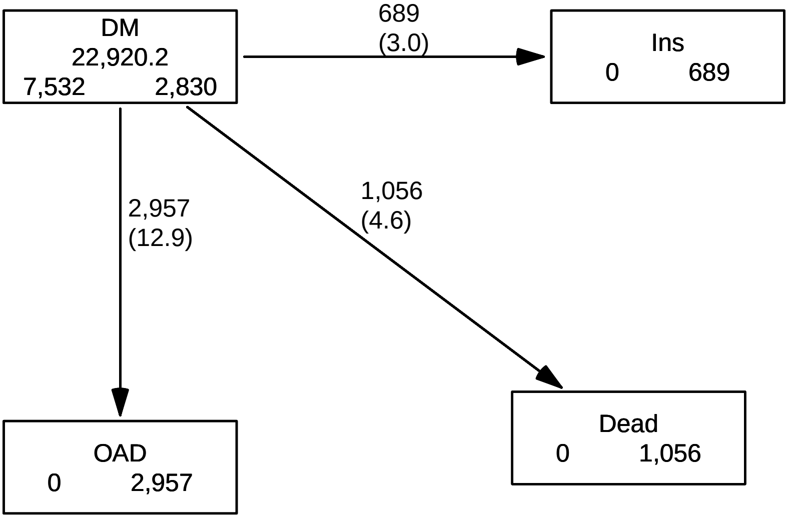
- In situations where the causes are not causes of death but other events, it is implicit that we only consider the first occurrence of an event from the state "alive", and ignore what occurs after.
- Suppose we consider:
  - OAD initiation
  - Insulin initiation

  . . . we must also always include
  - Death (without OAD or Insulin)
- to make sure possible events are **exhaustive**

Cause specific rates

$$\lambda_c(t) = \mathrm{P}\left\{\text{event } c \text{ in } (t, t+h) \middle| \text{ alive at } t\right\}/h$$

- Analysis is in principle straight forward: estimate a model for each of the cause-specific rates, $\lambda_c, c = 1, 2, \ldots$
- modeling for λ_c:
    - Event is occurrence of cause c
    - All other events are lumped together with the censoring
    - (or: Just use the persons-years in the "alive" state)
- the cause-specific rates will together form a complete model for the entire competing risks problem
- If the cause-specific rates are all we want, then we are done



DM
22,920.2
7,532 2,830

689
(3.0)

Ins
0 689

2,957
(12.9)

1,056
(4.6)

OAD
0 2,957

Dead
0 1,056

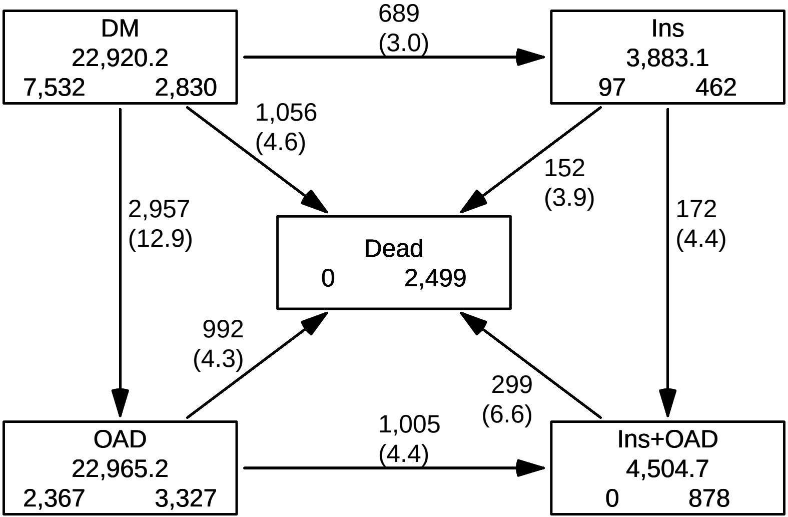
Example data

```
> 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  4  rows with duration of follow up < tol

> Mdm <- mcutLexis(Ldm,
+                   wh = c('dooad','doins'),
+           new.states = c('OAD','Ins'),
+           seq.states = FALSE,
+                 ties = TRUE)

NOTE: Precursor states set to DM
NOTE: 15 records with tied events times resolved (adding 0.01 random uniform),
      so results are only reproducible if the random number seed was set.
```

Example data

```
> summary(Mdm)

Transitions:
     To
From        DM Dead  OAD  Ins Ins+OAD  Records:  Events: Risk time:  Persons:
  DM      2830 1056 2958  688       0      7532     4702   22920.19      7532
  OAD        0  992 3327    0    1006      5325     1998   22965.26      5325
  Ins        0  152    0  462     171       785      323    3883.08       785
  Ins+OAD    0  299    0    0     878      1177      299    4504.73      1177
  Sum     2830 2499 6285 1150    2055     14819     7322   54273.27      9996
```

Cause-specific rates



DM
22,920.2
7,532 2,830

689
(3.0)

Ins
3,883.1
97 462

1,056
(4.6)

152
(3.9)

2,957
(12.9)

Dead
0 2,499

172
(4.4)

992
(4.3)

299
(6.6)

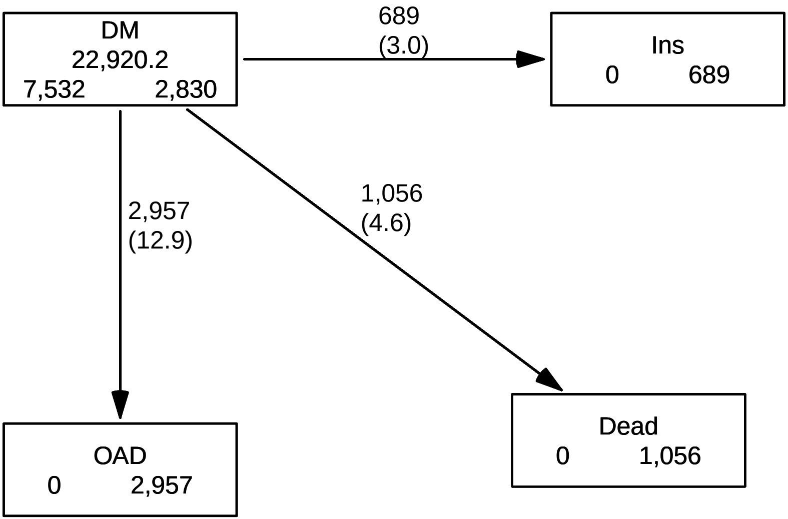
OAD
22,965.2
2,367 3,327

1,005
(4.4)

Ins+OAD
4,504.7
0 878

Split data—only time in DM

```
> Sdm <- splitMulti(factorize(subset(Mdm, lex.Cst == "DM")),
+                   tfd = seq(0, 20, 1/12))

NOTE: lex.Cst and lex.Xst now have levels:
 DM Dead OAD Ins

> summary(Sdm)

Transitions:
     To
From      DM Dead  OAD Ins   Records:  Events: Risk time:  Persons:
  DM 274263 1056 2958 688    278965     4702   22920.19      7532
```


DM
22,920.2
7,532 2,830

689
(3.0)

Ins
0 689

2,957
(12.9)

1,056
(4.6)

OAD
0 2,957

Dead
0 1,056

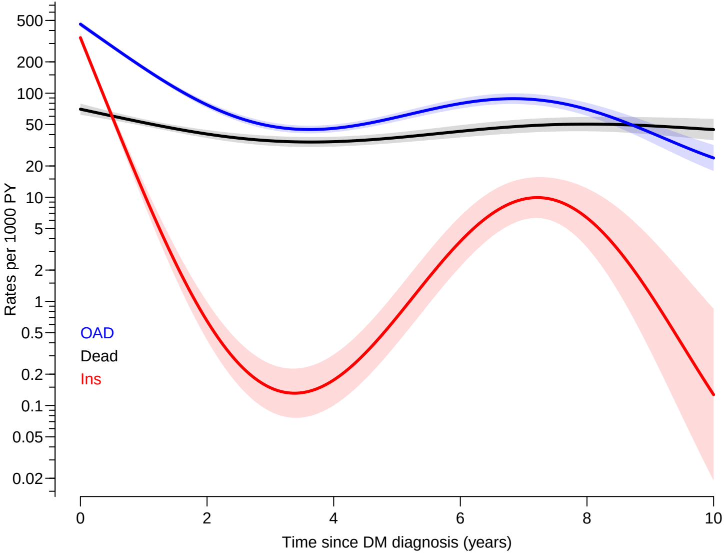
Poisson analysis of time-split data

```
> mD <- gam.Lexis(Sdm, ~ s(tfd, k = 5), to = 'Dead')

mgcv::gam Poisson analysis of Lexis object Sdm with log link:
Rates for the transition: DM->Dead

> mO <- gam.Lexis(Sdm, ~ s(tfd, k = 5), to = 'OAD' )

mgcv::gam Poisson analysis of Lexis object Sdm with log link:
Rates for the transition: DM->OAD

> mI <- gam.Lexis(Sdm, ~ s(tfd, k = 5), to = 'Ins' )

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

. . . wrapper for:

```
> mI <- gam(cbind(lex.Xst == "Ins" & lex.Cst != lex.Xst,
+                 lex.dur)
+           ~ s(tfd, k = 5),
+           family = poisreg,
+             data = subset(Sdm, lex.Cst == "DM"))
```

Cause-specific rates



Cumulative risks

In addition to the rates we are interested in:

- **survival** probability, and
- the **cumulative risks** — probability of dying from each cause
- i.e. the probabilities of being en each of the 4 boxes at any one time

Cumulative risks

- Survival probability

$$S(t) = \exp\left(-\int_0^t \sum_c \lambda_c(s)\,\mathrm{d}s\right) = \exp\left(-\sum_c \Lambda_c(t)\right)$$

  —the cumulative risk of survival. . .

- Cumulative risks

$$R_c(t) = \int_0^t \lambda_c(u) S(u)\,\mathrm{d}u = \int_0^t \lambda_c(u) \exp\left(-\sum_c \Lambda_c(u)\right)\mathrm{d}u$$

- These are probabilities, dimensionless.
- Note that integrals are just sums, so easy to compute numerically

Computing the rates

We need $\lambda_{\mathsf{D}}(t)$, $\lambda_{\mathsf{I}}(t)$ and $\lambda_{\mathsf{O}}(t)$.

So first choose the ts, then compute rates at these times

```
> # data frame with times where we want predicted rates, space: int
> int <- 1 / 100
> nd <- data.frame(tfd = seq(0, 10, int))
> #
> # utility function that calculates the midpoints between values in a vector
> mp <- function(x) x[-1] - diff(x) / 2
> #
> # rates at midpoints (mp) of intervals
> lD <- mp(ci.pred(mD, nd)[,1])
> lI <- mp(ci.pred(mI, nd)[,1])
> lO <- mp(ci.pred(mO, nd)[,1])
> #
> head(cbind(mp(nd$tfd), lD, lI, lO))
```

```
                 lD        lI        lO
2 0.005 0.07008416 0.3350871 0.4586874
3 0.015 0.06987361 0.3235601 0.4541849
```

Computing the survival function

$$S(t) = \exp\left(-\sum_c \int_0^t \lambda_c(s)\,\mathrm{d}s\right) = \exp\left(-\sum_c \Lambda_c(t)\right)$$

```
> # cumulative rates and survival function at right border of the intervals
> LD <- cumsum(lD) * int
> LI <- cumsum(lI) * int
> LO <- cumsum(lO) * int
> Sv <- exp(- LD - LI - LO )
> head(cbind(Sv, LD, LI, LO))

         Sv           LD          LI          LO
2 0.9913986 0.0007008416 0.003350871 0.004586874
3 0.9830309 0.0013995777 0.006586472 0.009128723
4 0.9748877 0.0020962145 0.009710769 0.013625990
5 0.9669607 0.0027907585 0.012727593 0.018079112
6 0.9592415 0.0034832158 0.015640641 0.022488524
7 0.9517225 0.0041735928 0.018453485 0.026854654
```

Computing the cumulative risk functions

$$R_c(t) = \int_0^t \lambda_c(u) S(u) \, \mathrm{d}u$$

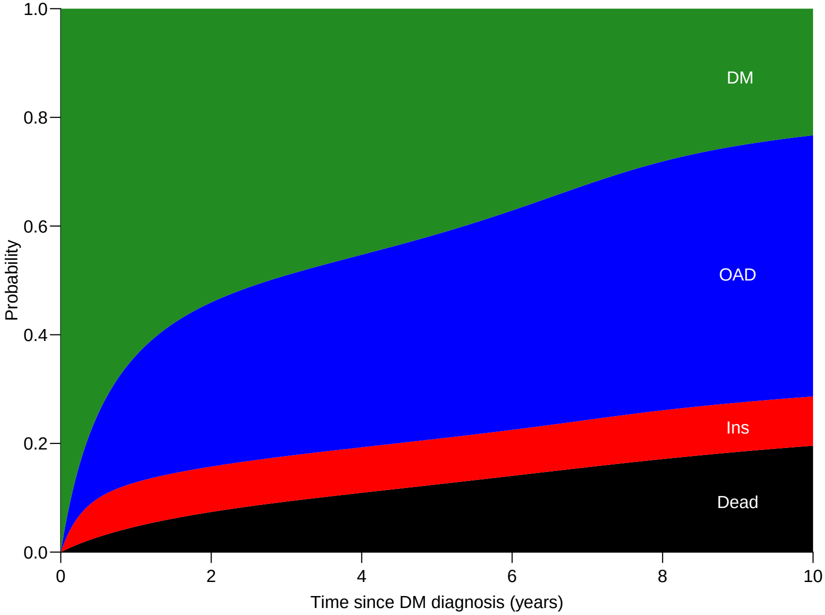
Note that the survival starts at 1, cumulative risks at 0

```
> # when integrating to get the cumulative risks we use the average (mp) Sv
> Sv <- c(1, Sv)
> rD <- c(0, cumsum(lD * mp(Sv)) * int)
> rI <- c(0, cumsum(lI * mp(Sv)) * int)
> r0 <- c(0, cumsum(l0 * mp(Sv)) * int)
> head(cbind(Sv, rD, rI, r0))
```

```
         Sv           rD          rI          r0
  1.0000000 0.0000000000 0.000000000 0.000000000
2 0.9913986 0.0006978275 0.003336460 0.004567147
3 0.9830309 0.0013876301 0.006530693 0.009050927
4 0.9748877 0.0020696092 0.009589253 0.013453568
5 0.9669607 0.0027439587 0.012518360 0.017777213
6 0.9592415 0.0034108651 0.015323920 0.022023922
```

Stacking the probabilities with `mat2pol`, `mat`rix to `pol`ygons

```
> zz <- mat2pol(cbind(rD, rI, rO, Sv), x = nd$tfd,
+               xlim = c(0,10), xaxs = "i", yaxs = "i", las = 1,
+               xlab = "Time since DM diagnosis (years)",
+               ylab = "Probability",
+                col =  c("black","red","blue","forestgreen"))
> text(9, mp(zz["9", ]), c("Dead", "Ins", "OAD"," DM"), col = "white")
> box(col = "white", lwd = 3)
```


1.0
0.8
0.6
0.4
0.2
0.0
Probability
DM
OAD
Ins
Dead
0
2
4
6
8
10
Time since DM diagnosis (years)

Confidence intervals

- …do not carry easily over through these calculations
- No analytical expression available (not even approximate)
- Thus, **simulation**:
- From the parameters of the model we got rates,
  from these we computed survival and cumulative risks.
- Instead of the parameters from the model use a random sample from the normal distribution with:
  - **mean**: estimates of parameters
  - **variance**: var-cov of parameters

Confidence intervals

- Instead of the parameters from the model use a random sample from the normal distribution with:
  - **mean**: estimates of parameters
  - **variance**: var-cov of parameters
- Compute rates, survival and cumulative risks from this
- Repeat many times (2000 for example)
- Use the 2.5 and 97.5 percentiles for the 95% c.i.
- Implemented in `ci.Crisk`

Using ci.Crisk

```
> res <- ci.Crisk(list(OAD = m0,
+                      Ins = mI,
+                     Dead = mD),
+                             nd = data.frame(tfd = 0:1000 / 100),
+                             nB = 500,
+                           perm = 4:1)
```

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

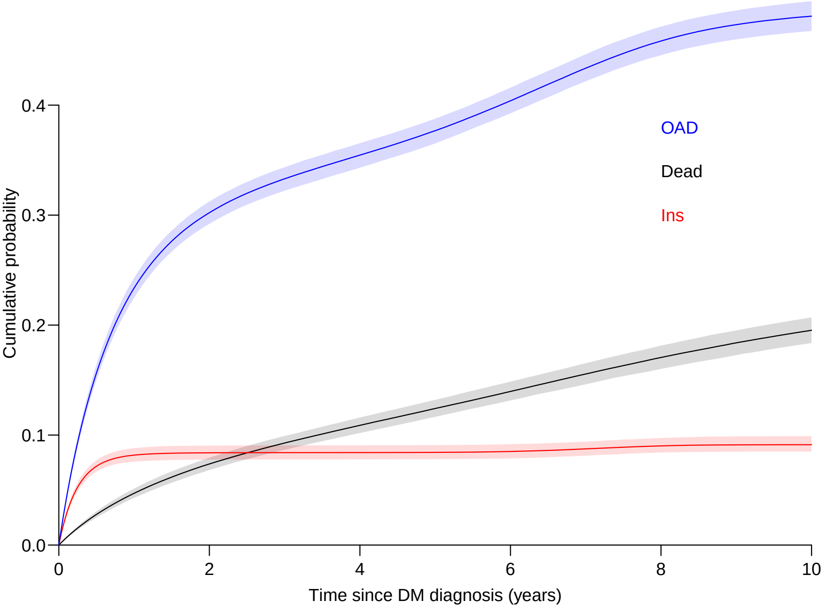
```
> str(res)
```

```
List of 4
 $ Crisk: num [1:1001, 1:4, 1:3] 1 0.991 0.983 0.975 0.967 ...
  ..- attr(*, "dimnames")=List of 3
  .. ..$ tfd  : chr [1:1001] "0" "0.01" "0.02" "0.03" ...
  .. ..$ cause: chr [1:4] "Surv" "OAD" "Ins" "Dead"
  .. ..$      : chr [1:3] "50%" "2.5%" "97.5%"
 $ Srisk: num [1:1001, 1:3, 1:3] 0 0.000703 0.001398 0.002084 0.002764 ...
  ..- attr(*, "dimnames")=List of 3
  .. ..$ tfd  : chr [1:1001] "0" "0.01" "0.02" "0.03" ...
  .. ..$ cause: chr [1:3] "Dead" "Dead+Ins" "Dead+Ins+OAD"
  .. ..$      : chr [1:3] "50%" "2.5%" "97.5%"
```

```
> str(res)

List of 4
 $ Crisk: num [1:1001, 1:4, 1:3] 1 0.991 0.983 0.975 0.967 ...
  ..- attr(*, "dimnames")=List of 3
  .. ..$ tfd  : chr [1:1001] "0" "0.01" "0.02" "0.03" ...
  .. ..$ cause: chr [1:4] "Surv" "OAD" "Ins" "Dead"
  .. ..$      : chr [1:3] "50%" "2.5%" "97.5%"
 $ Srisk: num [1:1001, 1:3, 1:3] 0 0.000703 0.001398 0.002084 0.002764 ...
  ..- attr(*, "dimnames")=List of 3
  .. ..$ tfd  : chr [1:1001] "0" "0.01" "0.02" "0.03" ...
  .. ..$ cause: chr [1:3] "Dead" "Dead+Ins" "Dead+Ins+OAD"
  .. ..$      : chr [1:3] "50%" "2.5%" "97.5%"
 $ Stime: num [1:1001, 1:4, 1:3] 0 0.00996 0.01983 0.02962 0.03933 ...
  ..- attr(*, "dimnames")=List of 3
  .. ..$ tfd  : chr [1:1001] "0" "0.01" "0.02" "0.03" ...
  .. ..$ cause: chr [1:4] "Surv" "OAD" "Ins" "Dead"
  .. ..$      : chr [1:3] "50%" "2.5%" "97.5%"
 $ time : num [1:1001] 0 0.01 0.02 0.03 0.04 0.05 0.06 0.07 0.08 0.09 ...
 - attr(*, "int")= num 0.01
```

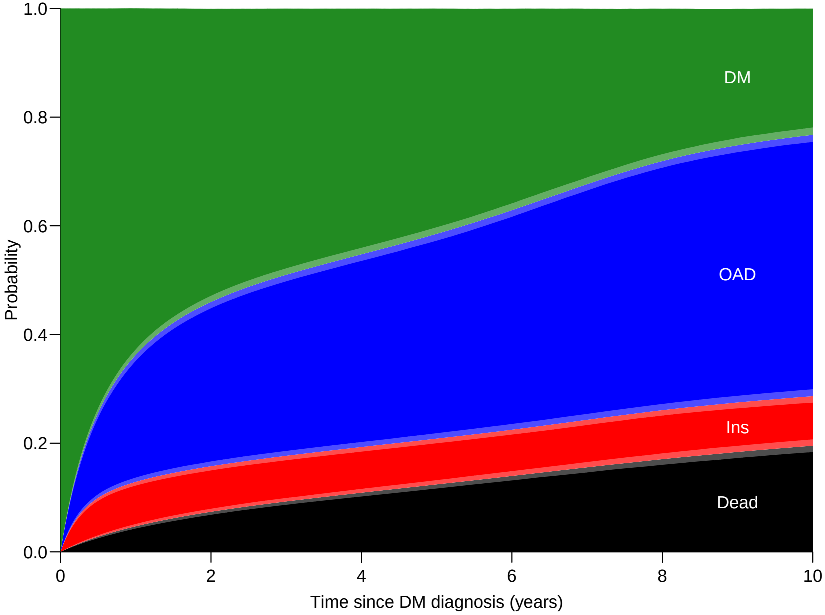
Plotting the results ci.Crisk

```
> matshade(res$time,
+          cbind(res$Crisk[,"Dead",],
+                res$Crisk[,"Ins" ,],
+                res$Crisk[,"OAD" ,]), plot = TRUE,
+          xlim = c(0,10), xaxs = "i", yaxs = "i", las = 1,
+          xlab = "Time since DM diagnosis (years)",
+          ylab = "Cumulative probability",
+           col = c("black","red","blue"))
> text(8, 0.3 + c(1,0,2)/25,
+      c("Dead","Ins","OAD"),
+      col = c("black","red","blue"), adj = 0)
```


0.4
0.3
0.2
0.1
0.0
Cumulative probability
0
2
4
6
8
10
Time since DM diagnosis (years)
OAD
Dead
Ins

Plotting the stacked probabilities from `ci.Crisk`

```
> zz <- mat2pol(res$Crisk[,c("Dead","Ins","OAD","Surv"),1],
+               x = res$time,
+            xlim = c(0,10), xaxs = "i", yaxs = "i", las = 1,
+            xlab = "Time since DM diagnosis (years)",
+            ylab = "Probability",
+             col =  c("black","red","blue","forestgreen") )
> text( 9, mp(zz["9",]), c("Dead","Ins","OAD","DM"), col = "white" )
> matshade(res$time,
+          cbind(res$Srisk[,1,],
+                res$Srisk[,2,],
+                res$Srisk[,3,]),
+          col = 'transparent', col.shade = "white", alpha = 0.3)
```


1.0
0.8
0.6
0.4
0.2
0.0
Probability
0
2
4
6
8
10
Time since DM diagnosis (years)
DM
OAD
Ins
Dead

Competing risks

- Different cases of death
- Cause specific rates analyzed separately
  there is no such thing as “competing risks analysis”
- With multiple causes, the survival depends on **all** cause specific rates.
- . . . and so does the cumulative risks
- There is no information about what happen if one cause of death is eliminated
- `ci.Crisk`:
- Input: rate models and common prediction data frame
- Result: Arrays

References