Frailty Models: Theory & Practice
Computer Practical

Theodor Balan and Hein Putter

11/3/2017

Introduction

The goal of this practical is to introduce you to fitting frailty models in R. We will try out functions from the `survival` and `frailtyEM` packages. We will not focus on covariate selection or cover the medical implications of the data analysis. Make sure that you have an up-to-date R installation.

The data that we will use comes from a placebo controlled trial of gamma interferon in chronic granulotomous disease (CGD). The main study based on these data was published as Gallin, J. I., et al. *A controlled trial of interferon gamma to prevent infection in chronic granulomatous disease*, in The New England Journal of Medicine 324.8 (1991): 509-516. The data set is available in numerous R packages, including the `survival` package.

Variables

The structure of the data is as follows:

- `id`: subject id
- `center`: the centers where the trial took place
- `random`: randomization date
- `treat`: treatment arm: `rIFN-g` or `placebo`
- `sex`: `female` or `male`
- `age`: age in years at time of study entry
- `height`: height in cm at time of study entry
- `weight`: weight in kg at time of study entry
- `inherit`: pattern of inheritance: `autosomal` (autosomal recessive) or `X-linked`
- `propylac`: use of prophylactic antibiotics at time of study entry (`0` for no, and `1` for yes)
- `hos.cat`: institution category: `US:other`, `US:NIH`, `Europe: Amsterdam` and `Europe:other`
- `tstart`, `tstop`, `status`: event times from randomization (time 0) to infections
- `enum` the number of the event

Descriptive analysis

First we load the data:

```
library(survival)
data(cgd)
```

This loads two data sets: `cgd` and `cgd0`. We will use `cgd`, which is already in the Andersen-Gill format that we discussed in the course. You can find a brief description of this data set in the `survival` package:

```
?cgd
head(cgd)
```

(Q1) Which event is the outcome of interest here? How many individuals are there in the data set? How many individuals are within each center?

```r
length(unique(cgd$id))
```

```
## [1] 128
```

```r
tapply(cgd[cgd$enum == 1,"id"], cgd[cgd$enum == 1, "center"], length)
```

```
##    Harvard Medical Sch      Scripps Institute          Copenhagen
##                      4                     16                   4
##                    NIH  L.A. Children's Hosp  Mott Children's Hosp
##                     26                      8                   9
##          Univ. of Utah   Univ. of Washington    Univ. of Minnesota
##                      4                      4                   6
##        Univ. of Zurich Texas Children's Hosp            Amsterdam
##                     16                      8                  19
## Mt. Sinai Medical Ctr
##                      4
```

The outcome of interest is the occurence of recurrent infections after the interventions. It is determined by 3 columns: tstart, tstop and status. We can see that there are 128 individuals in the data set. In terms of centers we see that there are a few large ones (NIH, Amsterdam, Scripps and Zurich) and the rest are smaller.

(Q2) How many events are there in the whole data set?

```r
sum(cgd$status)
```

```
## [1] 76
```

(A2) There are 76 events in the whole data set.

(Q3) How many events are there for each individual? Make a histogram of this. What percentage of individuals experience no serious infections? What are the maximum number of observed events / individual?

```r
hist(tapply(cgd$status, cgd$id, sum), main = "events / individual", xlab = "events")
```
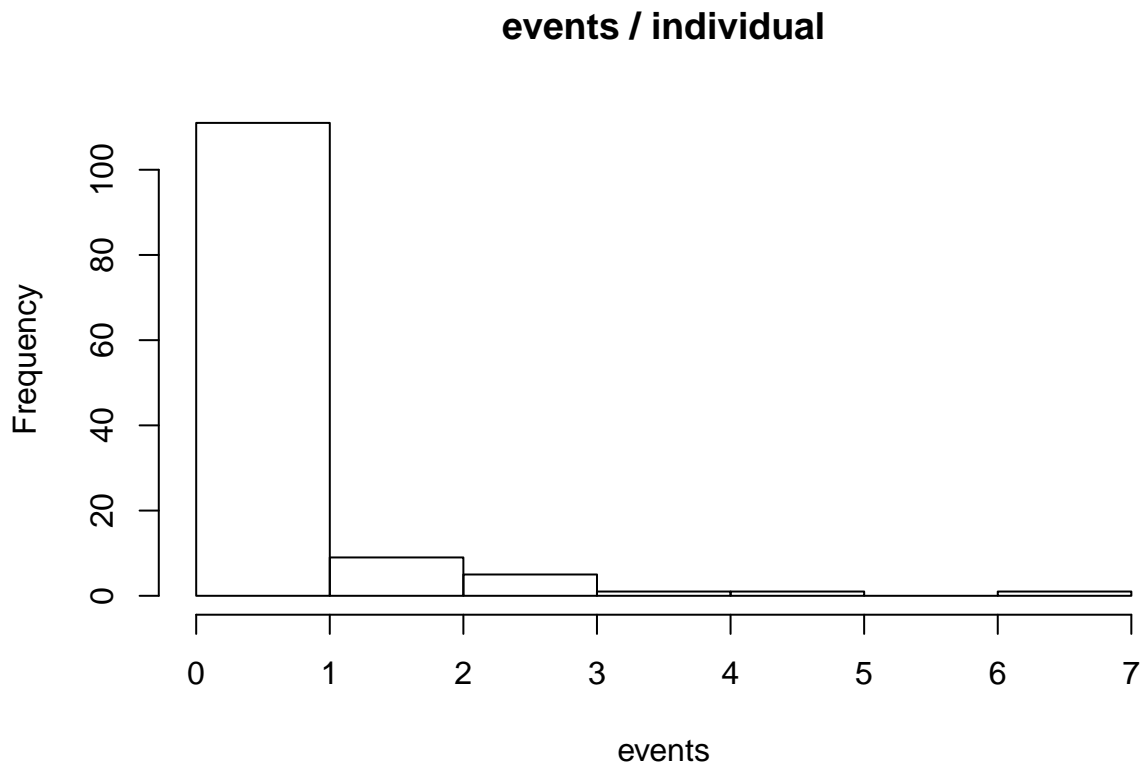

events / individual

```
mean(tapply(cgd$status, cgd$id, sum) == 0)
```

```
## [1] 0.65625
```

(A3) In the histogram (barplot) we see that the most individuals actually have 0 events, that is around 65% of patients. The maximum number of events for an individual is 7.

With recurrent events, often the cumulative intensity is more interesting than the survival. From counting process theory, a counting process $N(t)$ with intensity $\lambda(t)$ may be decomposed into:

$$N(t) = \Lambda(t) + M(t).$$

This classical result is known as the Doobs-Meyer decomposition and it is fundamental in applying martingale theory to survival analysis. The interpretation here is that $N(t)$ is the number of events of an individual up to time t, $M(t)$ is the martingale residual at time t and $\Lambda(t)$ is the cumulative intensity (hazard), which for a proportional intensity (hazard) would be

$$\Lambda(t) = \int_0^t \lambda_i(s)ds = \int_0^t \exp(\beta^\top x_i)\lambda_0(s)ds.$$

We will not dwelve on this here, but $M(t)$ has expectation 0, which implies that $\Lambda(t)$ is equal to the expected number of events of an individual up to time t. When the outcome is a single event (such as death), then $N(t) \leq 1$ and the intensity $\lambda(t)$ is the hazard function.

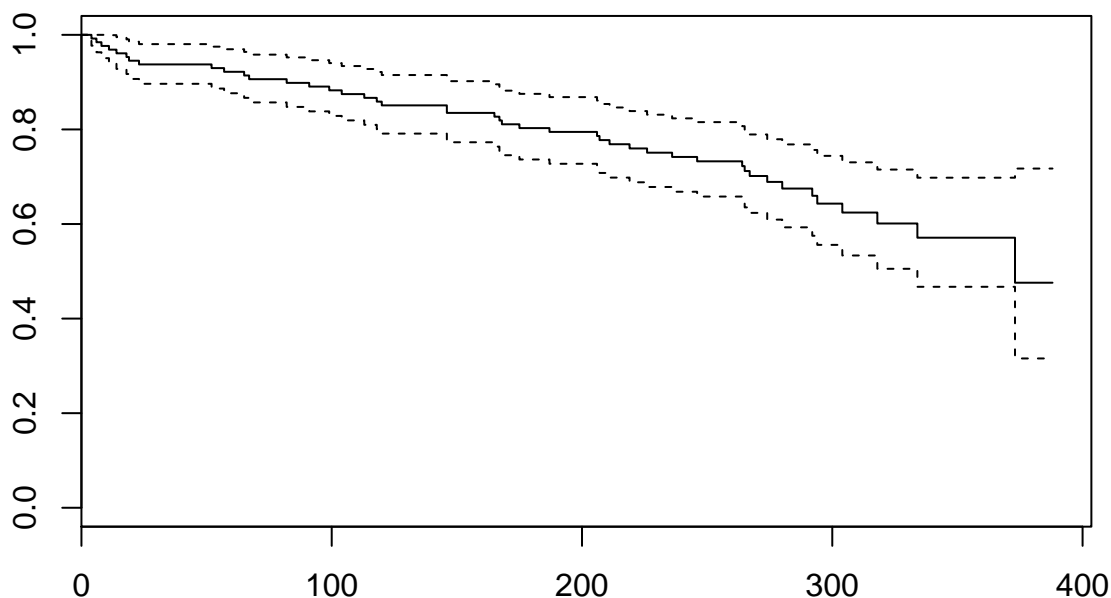The survival, in the recurrent event case, would be

$$S(t) = \exp(-\Lambda(t))$$

which is not a quantity that can be easily interpreted.

Time to first event for each individual

First, we will look at the time to the first event for each individual.

```
cgd1 <- cgd[cgd$enum==1,]
plot(survfit(Surv(tstop, status) ~ 1, cgd1))
```


(Q4) What is the interpretation of this curve?

(A4) This is a Kaplan Meier curve for the time to the first event. As we saw before, most individuals actually never get an event. After 200 days, around 80% of individuals still have not had an event.

(Q5) Which variables influence the time to the first event, and how? Tip: you can call `summary()` on an object for an easier to read output.

```
coxph(Surv(tstop, status) ~ frailty(id), cgd1)
```

```
coxph(Surv(tstop, status) ~ sex + treat + age, cgd1)
coxph(Surv(tstop, status) ~ sex + treat + age + frailty(id), cgd1)
```

```
mod_univ <- coxph(Surv(tstop, status) ~ sex + treat + age + inherit + steroids +
                      propylac + frailty(id), cgd1)
summary(mod_univ)
```

```
## Call:
## coxph(formula = Surv(tstop, status) ~ sex + treat + age + inherit +
##     steroids + propylac + frailty(id), data = cgd1)
##
##   n= 128, number of events= 44
##
##                    coef    se(coef) se2     Chisq DF p
## sexfemale         -0.49607 0.50812  0.50812  0.95 1  0.3300
## treatrIFN-g       -1.12082 0.34439  0.34439 10.59 1  0.0011
## age               -0.03743 0.01851  0.01851  4.09 1  0.0430
## inheritautosomal   0.49277 0.39981  0.39981  1.52 1  0.2200
## steroids           1.10624 0.75469  0.75469  2.15 1  0.1400
## propylac          -0.40837 0.42269  0.42269  0.93 1  0.3300
## frailty(id)                                  0.00 0  0.9400
##
##                    exp(coef) exp(-coef) lower .95 upper .95
## sexfemale             0.6089    1.6422    0.2249    1.6484
## treatrIFN-g           0.3260    3.0674    0.1660    0.6403
## age                   0.9633    1.0381    0.9289    0.9988
## inheritautosomal      1.6368    0.6109    0.7476    3.5837
## steroids              3.0230    0.3308    0.6887   13.2687
## propylac              0.6647    1.5044    0.2903    1.5221
##
## Iterations: 6 outer, 22 Newton-Raphson
##      Variance of random effect= 5e-07   I-likelihood = -184.9
## Degrees of freedom for terms= 1 1 1 1 1 1 0
## Concordance= 0.681  (se = 0.047 )
## Likelihood ratio test= 18.34  on 6 df,    p=0.005444
```

(A5) In the last model we can see that treatment and age are the only significant variables. Treatment and age both decrease the time to the first event. Treatment has a very significant effect: it decreases the hazard by about 68%!

(Q6) Do you think that the `frailty()` statement here is useful? Try to fit the same models without the `frailty()` statement. Does anything change?

(A6) Probably it will not make any difference: the estimated frailty variance is virtually 0. This means that the fitted model is basically a Cox model. You can see that the results are completely identical if we fit the same model without frailty.

```
mod_univ_nofr <- coxph(Surv(tstop, status) ~ sex + treat + age + inherit + steroids +
                      propylac, cgd1)
summary(mod_univ_nofr)
```

```
## Call:
## coxph(formula = Surv(tstop, status) ~ sex + treat + age + inherit +
##     steroids + propylac, data = cgd1)
##
##   n= 128, number of events= 44
##
##                      coef exp(coef) se(coef)      z Pr(>|z|)
## sexfemale        -0.49607   0.60892  0.50812 -0.976  0.32893
## treatrIFN-g      -1.12082   0.32601  0.34439 -3.255  0.00114 **
## age              -0.03743   0.96326  0.01851 -2.022  0.04315 *
## inheritautosomal  0.49277   1.63684  0.39981  1.232  0.21776
## steroids          1.10624   3.02296  0.75469  1.466  0.14270
## propylac         -0.40837   0.66473  0.42269 -0.966  0.33398
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##                  exp(coef) exp(-coef) lower .95 upper .95
## sexfemale           0.6089     1.6422    0.2249    1.6484
## treatrIFN-g         0.3260     3.0674    0.1660    0.6403
## age                 0.9633     1.0381    0.9289    0.9988
## inheritautosomal    1.6368     0.6109    0.7476    3.5837
## steroids            3.0230     0.3308    0.6887   13.2687
## propylac            0.6647     1.5044    0.2903    1.5221
##
## Concordance= 0.677  (se = 0.047 )
## Rsquare= 0.133   (max possible= 0.952 )
## Likelihood ratio test= 18.34  on 6 df,   p=0.005444
## Wald test            = 16.97  on 6 df,   p=0.009404
## Score (logrank) test = 18.16  on 6 df,   p=0.005839
```

(Q7) From the output of `mod_univ`, what do you think that the empirical Bayes estimates of the fraily will look like? Try to find the estimates. You can see the fields corresponding to an object in R by calling `str()` on that object. On which scale do you think that the frailty estimates are on?

```r
str(mod_univ)
```

(A7) The frailty estimates are in the `$frail` field. The estimates are virtually 0, but we can see that a number of them are negative. In fact, they are the logarithm of the empirical Bayes estimates (*not* the estimates of the log-frailty!).

```r
hist(mod_univ$frail)
```

Histogram of mod_univ$frail
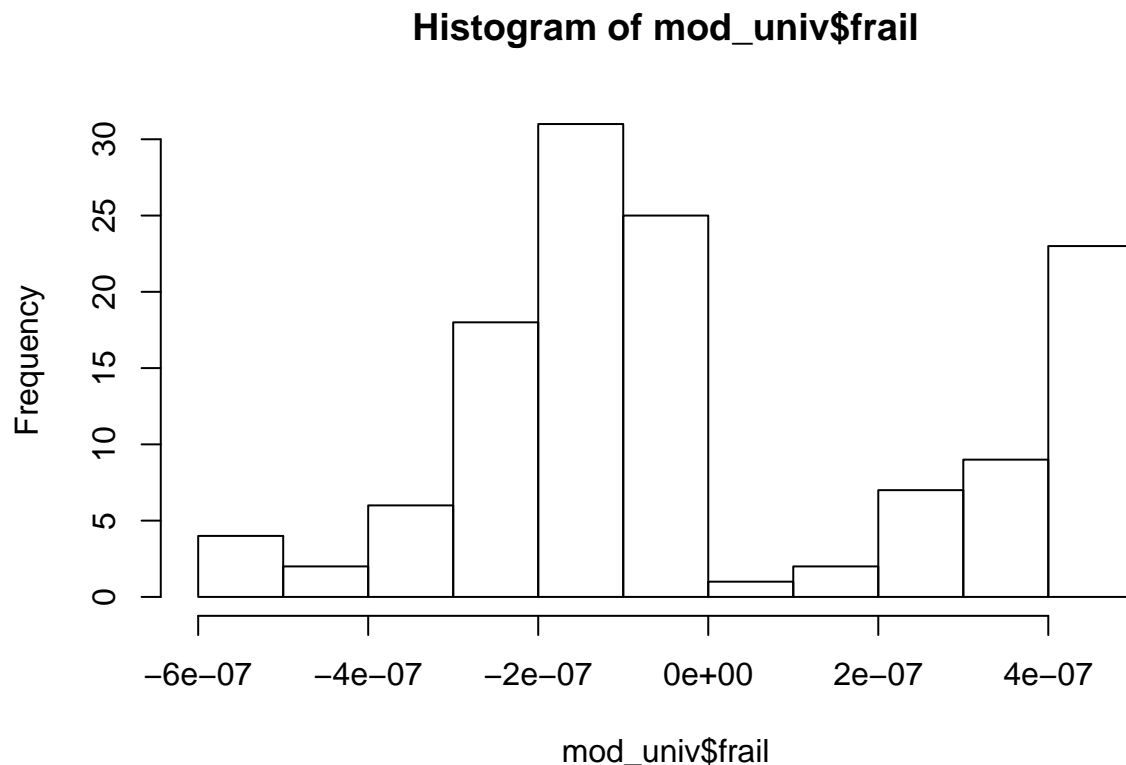
(Q8) Do you think that non-proportional covariate effect might be a big problem for this model? Call `cox.zph()` on it and check the results.

(A8) We can't really know that unless we test it.

```
cox.zph(mod_univ)
```

```
##                     rho  chisq     p
## sexfemale        -0.0186 0.0146 0.904
## treatrIFN-g      -0.0194 0.0193 0.889
## age               0.1044 0.5180 0.472
## inheritautosomal -0.0168 0.0157 0.900
## steroids          0.0272 0.0273 0.869
## propylac         -0.0271 0.0331 0.856
## GLOBAL                NA 0.7096 0.994
```

Nothing appears to be significant which shows that there is no evidence of non-proportionality. If we would have had a significant frailty, probably that there would be some non-proportionality in the Cox model (since the univariate frailty model is confounded with non-proportional covariate effects).

Modeling all the observations

(Q8b) Now back to analyzing the whole data set. In what time scale are the event times measured in this data set? Do you think that the calendar time or the gap time are more relevant in this case?

(A8b) I noticed that I put question 8 twice. Sorry! The time scale is time since intervention here. Calendar time is probably more relevant since there is a clear time origin.

We will start with the marginal models with working independence. Below is the fit using the calendar time scale.

6

(Q9) What is the left hand side in the argument of `coxph`? Why was it all right to use only `(tstop, status)` before?

```
mod_cal_wi <- coxph(Surv(tstart, tstop, status) ~ sex + treat + age + inherit +
                    steroids + propylac + cluster(id),
                ties = "breslow", cgd)
summary(mod_cal_wi)
```

```
## Call:
## coxph(formula = Surv(tstart, tstop, status) ~ sex + treat + age +
##     inherit + steroids + propylac + cluster(id), data = cgd,
##     ties = "breslow")
##
##   n= 203, number of events= 76
##
##                      coef exp(coef) se(coef) robust se      z Pr(>|z|)
## sexfemale        -0.73418   0.47990  0.39036   0.43343 -1.694 0.090290 .
## treatrIFN-g      -1.05399   0.34855  0.26504   0.30914 -3.409 0.000651 ***
## age              -0.04235   0.95853  0.01395   0.01479 -2.863 0.004194 **
## inheritautosomal  0.68068   1.97522  0.27413   0.38858  1.752 0.079820 .
## steroids          1.39243   4.02463  0.56298   0.63176  2.204 0.027521 *
## propylac         -0.54460   0.58007  0.30357   0.35074 -1.553 0.120494
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##                  exp(coef) exp(-coef) lower .95 upper .95
## sexfemale           0.4799     2.0838    0.2052    1.1222
## treatrIFN-g         0.3485     2.8691    0.1902    0.6388
## age                 0.9585     1.0433    0.9311    0.9867
## inheritautosomal    1.9752     0.5063    0.9223    4.2303
## steroids            4.0246     0.2485    1.1667   13.8829
## propylac            0.5801     1.7239    0.2917    1.1535
##
## Concordance= 0.702  (se = 0.035 )
## Rsquare= 0.168   (max possible= 0.966 )
## Likelihood ratio test= 37.37  on 6 df,   p=1.495e-06
## Wald test            = 26.9  on 6 df,   p=0.0001515
## Score (logrank) test = 36.78  on 6 df,   p=1.94e-06,   Robust = 12.62  p=0.0495
##
##   (Note: the likelihood ratio and score tests assume independence of
##      observations within a cluster, the Wald and robust score tests do not).
```

(A9) The left hand side is now `Surv(tstart, tstop, status)`. Before, when looking at the first event only, `tstart` was always 0. That is what `R` considers if you do not include `tstart`.

(Q10) We used `+ cluster(id)` to let `coxph` know which observations to take together. What happens if we would not add that statement? Fit the same model without that statment. Do you notice any differences?

(A10) The difference is that `+cluster()` gives the robust standard errors (the `robust se` column in the output), which is a correct estimate when the observations are not independent.

(Q11) Suppose now that we want to go in the other direction - and try a fixed effects model. We can do that by adding instead of `+ cluster(id)`, `+ as.factor(id)`. Fit this model and examine the output. Do you see anything problematic? How could you explain that?

```
mod_cal_fe <- coxph(Surv(tstart, tstop, status) ~ sex + treat + age + inherit +
                    steroids + propylac + as.factor(id),
```

```
                ties = "breslow", cgd)
mod_cal_fe
```

```
## Call:
## coxph(formula = Surv(tstart, tstop, status) ~ sex + treat + age +
##     inherit + steroids + propylac + as.factor(id), data = cgd,
##     ties = "breslow")
##
##                      coef  exp(coef)  se(coef)      z    p
## sexfemale        -6.39e+01   1.69e-28  5.85e+02  -0.11 0.91
## treatrIFN-g      -9.59e+01   2.26e-42  7.87e+02  -0.12 0.90
## age               2.91e+00   1.83e+01  2.61e+01   0.11 0.91
## inheritautosomal  2.92e+00   1.85e+01  1.18e+02   0.02 0.98
## steroids         -4.98e+01   2.41e-22  4.43e+02  -0.11 0.91
## propylac         -7.81e+01   1.18e-34  6.32e+02  -0.12 0.90
## as.factor(id)2   -8.92e+01   1.86e-39  8.16e+02  -0.11 0.91
## as.factor(id)3   -1.15e+01   1.05e-05  1.42e+02  -0.08 0.94
## as.factor(id)4    8.88e+00   7.21e+03  1.24e+02   0.07 0.94
## as.factor(id)5   -9.33e+01   2.94e-41  8.43e+02  -0.11 0.91
## as.factor(id)6   -1.01e+02   1.04e-44  8.36e+02  -0.12 0.90
## as.factor(id)7   -1.08e+02   8.99e-48  9.72e+02  -0.11 0.91
## as.factor(id)8    2.33e+01   1.38e+10  2.31e+02   0.10 0.92
## as.factor(id)9   -1.26e+02   2.57e-55  1.13e+03  -0.11 0.91
## as.factor(id)10   2.92e+01   4.70e+12  2.79e+02   0.10 0.92
## as.factor(id)11  -4.85e+01   8.91e-22  4.60e+02  -0.11 0.92
## as.factor(id)12   2.80e+01   1.49e+12  2.02e+02   0.14 0.89
## as.factor(id)13   8.88e+00   7.21e+03  1.24e+02   0.07 0.94
## as.factor(id)14  -4.57e+01   1.44e-20  4.35e+02  -0.11 0.92
## as.factor(id)15   2.35e+01   1.65e+10  1.94e+02   0.12 0.90
## as.factor(id)16  -1.31e+02   1.78e-57  1.10e+03  -0.12 0.91
## as.factor(id)17  -8.46e+01   1.80e-37  7.66e+02  -0.11 0.91
## as.factor(id)18  -4.73e+01   2.87e-21  4.35e+02  -0.11 0.91
## as.factor(id)19   8.87e+00   7.13e+03  1.69e+02   0.05 0.96
## as.factor(id)20  -1.32e+02   5.69e-58  1.18e+03  -0.11 0.91
## as.factor(id)21   1.47e+01   2.41e+06  1.62e+02   0.09 0.93
## as.factor(id)22   2.33e+01   1.34e+10  2.32e+02   0.10 0.92
## as.factor(id)23  -6.68e+01   9.54e-30  5.25e+02  -0.13 0.90
## as.factor(id)24   8.72e+00   6.12e+03  1.52e+02   0.06 0.95
## as.factor(id)25  -6.45e+01   9.31e-29  5.86e+02  -0.11 0.91
## as.factor(id)26  -5.52e+01   1.07e-24  4.38e+02  -0.13 0.90
## as.factor(id)27   3.49e+01   1.42e+15  3.30e+02   0.11 0.92
## as.factor(id)28  -7.26e+01   3.04e-32  5.89e+02  -0.12 0.90
## as.factor(id)29   3.70e+01   1.19e+16  2.75e+02   0.13 0.89
## as.factor(id)30  -4.72e+01   3.10e-21  4.35e+02  -0.11 0.91
## as.factor(id)31  -5.82e+01   5.36e-26  5.36e+02  -0.11 0.91
## as.factor(id)32  -1.08e+02   2.03e-47  8.96e+02  -0.12 0.90
## as.factor(id)33   5.84e+00   3.44e+02  1.15e+02   0.05 0.96
## as.factor(id)34  -1.62e+02   3.06e-71  1.32e+03  -0.12 0.90
## as.factor(id)35  -1.22e+02   1.05e-53  1.03e+03  -0.12 0.91
## as.factor(id)36   1.50e+01   3.37e+06  1.01e+02   0.15 0.88
## as.factor(id)37   3.78e+01   2.61e+16  3.54e+02   0.11 0.92
## as.factor(id)38  -5.81e+00   3.01e-03  1.17e+02  -0.05 0.96
## as.factor(id)39  -2.35e+02   8.35e-103 1.96e+03  -0.12 0.90
## as.factor(id)40   2.33e+01   1.30e+10  2.32e+02   0.10 0.92
```

```
## as.factor(id)41     2.79e+00   1.63e+01   1.01e+02   0.03 0.98
## as.factor(id)42     2.04e+01   7.12e+08   2.09e+02   0.10 0.92
## as.factor(id)43    -8.97e+01   1.13e-39   7.42e+02  -0.12 0.90
## as.factor(id)44     3.99e+01   2.17e+17   3.00e+02   0.13 0.89
## as.factor(id)45     7.85e+01   1.26e+34   7.11e+02   0.11 0.91
## as.factor(id)46    -1.25e+02   3.18e-55   1.13e+03  -0.11 0.91
## as.factor(id)47    -4.06e+01   2.22e-18   2.92e+02  -0.14 0.89
## as.factor(id)48    -1.25e+02   5.42e-55   1.05e+03  -0.12 0.91
## as.factor(id)49    -6.63e+01   1.55e-29   6.12e+02  -0.11 0.91
## as.factor(id)50    -1.20e+02   9.64e-53   1.08e+03  -0.11 0.91
## as.factor(id)51     6.69e+01   1.11e+29   6.08e+02   0.11 0.91
## as.factor(id)52    -1.01e+02   1.39e-44   9.21e+02  -0.11 0.91
## as.factor(id)53    -1.36e+02   1.30e-59   1.16e+03  -0.12 0.91
## as.factor(id)54    -5.81e+00   2.99e-03   1.18e+02  -0.05 0.96
## as.factor(id)55     1.69e+01   2.20e+07   1.17e+02   0.14 0.88
## as.factor(id)56    -2.32e+01   8.36e-11   2.59e+02  -0.09 0.93
## as.factor(id)57    -8.58e+01   5.30e-38   7.29e+02  -0.12 0.91
## as.factor(id)58    -1.48e+02   4.27e-65   1.26e+03  -0.12 0.91
## as.factor(id)59     2.62e+01   2.28e+11   2.57e+02   0.10 0.92
## as.factor(id)60     7.60e+01   9.67e+32   6.86e+02   0.11 0.91
## as.factor(id)61    -1.49e+02   2.82e-65   1.27e+03  -0.12 0.91
## as.factor(id)62    -5.88e+00   2.80e-03   5.22e+01  -0.11 0.91
## as.factor(id)63    -5.88e+01   2.94e-26   5.36e+02  -0.11 0.91
## as.factor(id)64    -2.40e+01   3.84e-11   3.00e+02  -0.08 0.94
## as.factor(id)65    -2.91e+01   2.33e-13   3.05e+02  -0.10 0.92
## as.factor(id)66    -1.48e+02   3.73e-65   1.26e+03  -0.12 0.91
## as.factor(id)67     1.45e+01   2.01e+06   2.04e+02   0.07 0.94
## as.factor(id)68    -4.04e+01   2.96e-18   2.94e+02  -0.14 0.89
## as.factor(id)69     2.62e+01   2.29e+11   2.57e+02   0.10 0.92
## as.factor(id)70          NA         NA   0.00e+00     NA   NA
## as.factor(id)71     5.52e+01   9.41e+23   5.07e+02   0.11 0.91
## as.factor(id)72    -1.20e+01   6.05e-06   2.32e+02  -0.05 0.96
## as.factor(id)73    -6.97e+01   5.17e-31   5.64e+02  -0.12 0.90
## as.factor(id)74     7.85e+01   1.20e+34   7.12e+02   0.11 0.91
## as.factor(id)75    -7.56e+01   1.52e-33   6.14e+02  -0.12 0.90
## as.factor(id)76    -8.43e+01   2.49e-37   6.92e+02  -0.12 0.90
## as.factor(id)77    -9.30e+01   4.06e-41   7.69e+02  -0.12 0.90
## as.factor(id)78    -1.33e+02   1.21e-58   1.13e+03  -0.12 0.91
## as.factor(id)79    -1.95e+02   2.66e-85   1.68e+03  -0.12 0.91
## as.factor(id)80     3.78e+01   2.52e+16   3.56e+02   0.11 0.92
## as.factor(id)81     2.32e+01   1.23e+10   2.35e+02   0.10 0.92
## as.factor(id)82    -5.52e+01   1.04e-24   4.40e+02  -0.13 0.90
## as.factor(id)83    -7.56e+01   1.53e-33   6.17e+02  -0.12 0.90
## as.factor(id)84    -8.14e+01   4.54e-36   6.67e+02  -0.12 0.90
## as.factor(id)85    -1.25e+02   7.48e-55   9.71e+02  -0.13 0.90
## as.factor(id)86     4.03e+01   3.19e+17   3.00e+02   0.13 0.89
## as.factor(id)87    -9.87e+01   1.37e-43   8.95e+02  -0.11 0.91
## as.factor(id)88    -7.27e+01   2.79e-32   5.90e+02  -0.12 0.90
## as.factor(id)89     2.03e+01   6.74e+08   2.12e+02   0.10 0.92
## as.factor(id)90    -7.60e+01   1.03e-33   6.89e+02  -0.11 0.91
## as.factor(id)91    -7.85e+01   8.33e-35   6.41e+02  -0.12 0.90
## as.factor(id)92    -1.45e+01   4.88e-07   1.68e+02  -0.09 0.93
## as.factor(id)93    -1.54e+02   1.26e-67   1.31e+03  -0.12 0.91
## as.factor(id)94     2.90e+01   4.11e+12   2.83e+02   0.10 0.92
```

```
## as.factor(id)95     2.62e+01   2.29e+11   2.57e+02   0.10 0.92
## as.factor(id)96     1.16e+01   1.10e+05   1.50e+02   0.08 0.94
## as.factor(id)97     2.90e+01   4.11e+12   2.83e+02   0.10 0.92
## as.factor(id)98    -5.46e+01   1.94e-24   5.08e+02  -0.11 0.91
## as.factor(id)99    -8.74e+00   1.61e-04   1.36e+02  -0.06 0.95
## as.factor(id)100    2.90e+01   3.88e+12   2.32e+02   0.12 0.90
## as.factor(id)101    4.07e+01   4.64e+17   3.84e+02   0.11 0.92
## as.factor(id)102   -2.91e+01   2.36e-13   2.82e+02  -0.10 0.92
## as.factor(id)103   -1.42e+02   1.42e-62   1.21e+03  -0.12 0.91
## as.factor(id)104    1.75e+01   3.85e+07   1.87e+02   0.09 0.93
## as.factor(id)105   -2.62e+01   4.30e-12   2.60e+02  -0.10 0.92
## as.factor(id)106   -2.62e+01   4.34e-12   1.69e+02  -0.15 0.88
## as.factor(id)107    1.45e+01   2.01e+06   1.72e+02   0.08 0.93
## as.factor(id)108   -1.28e+02   4.12e-56   1.08e+03  -0.12 0.91
## as.factor(id)109   -1.28e+02   2.94e-56   1.08e+03  -0.12 0.91
## as.factor(id)110    3.21e+01   8.53e+13   2.26e+02   0.14 0.89
## as.factor(id)111   -1.28e+02   2.95e-56   1.08e+03  -0.12 0.91
## as.factor(id)112   -2.32e+01   8.04e-11   1.44e+02  -0.16 0.87
## as.factor(id)113   -1.02e+02   6.65e-45   8.46e+02  -0.12 0.90
## as.factor(id)114   -8.14e+01   4.55e-36   6.66e+02  -0.12 0.90
## as.factor(id)115   -2.03e+02   6.44e-89   1.68e+03  -0.12 0.90
## as.factor(id)116   -1.91e+02   7.19e-84   1.58e+03  -0.12 0.90
## as.factor(id)117   -8.72e+01   1.37e-38   7.18e+02  -0.12 0.90
## as.factor(id)118    3.20e+01   7.64e+13   3.12e+02   0.10 0.92
## as.factor(id)119   -5.68e+01   2.10e-25   5.34e+02  -0.11 0.92
## as.factor(id)121   -5.52e+01   1.06e-24   4.26e+02  -0.13 0.90
## as.factor(id)122   -1.02e+02   6.73e-45   8.47e+02  -0.12 0.90
## as.factor(id)123   -9.86e+01   1.52e-43   8.95e+02  -0.11 0.91
## as.factor(id)125          NA         NA   0.00e+00     NA   NA
## as.factor(id)131          NA         NA   0.00e+00     NA   NA
## as.factor(id)132   -2.73e+00   6.55e-02   2.61e+01  -0.10 0.92
## as.factor(id)133          NA         NA   0.00e+00     NA   NA
## as.factor(id)134          NA         NA   0.00e+00     NA   NA
## as.factor(id)135          NA         NA   0.00e+00     NA   NA
##
## Likelihood ratio test=178  on 127 df, p=0.00198
## n= 203, number of events= 76
```

(A11) The estimates of the fixed effect for individuals are either 0 or very large. This is a probelm of collinearity: if we have fixed individual effects, for example, we can't tell the effect of sex or other variables that are constant within the individual.

(Q12) We can also try to stratify, so that we let each individual have their own baseline hazard. In this case, we would use + strata(id) instead of + cluster(id). What happens? Any idea why?

```r
try(mod_cal_strat <- coxph(Surv(tstart, tstop, status) ~ sex + treat + age + inherit +
                    steroids + propylac + strata(id),
                 ties = "breslow", cgd))
```

(A12) This won't work. The error is quite funny (*This should never happen. Please contact the author.*), but the reason is the same as before: since each individual has its own baseline hazard, we can't decide on covariates effect if they do not vary within the strata.

(Q13) What do you think is the key for the fixed effects and the stratified models to work? In which case could you use those models?

(A13) We would need covariates to vary within the individual. Strata would be a good idea if there would be

many events within each cluster/individual. For instance, when there are large centers with many patients. Fixed effects assume proportionality between clusters, and that might work even when there are not so many events per cluster or individual.

Shared frailty models

(Q14) We now include the `+ frailty(id)` statement in `coxph`. What kind of frailty model does this fit? Check `?frailty`.

```
mod_cal_fr <- coxph(Surv(tstart, tstop, status) ~ sex + treat + age + inherit +
                    steroids + propylac + frailty(id),
                ties = "breslow", cgd)
summary(mod_cal_fr)
```

```
## Call:
## coxph(formula = Surv(tstart, tstop, status) ~ sex + treat + age +
##     inherit + steroids + propylac + frailty(id), data = cgd,
##     ties = "breslow")
##
##   n= 203, number of events= 76
##
##                    coef    se(coef) se2     Chisq DF    p
## sexfemale         -0.80360 0.47560  0.40853  2.85 1.00 0.09100
## treatrIFN-g       -0.99104 0.29828  0.26530 11.04 1.00 0.00089
## age               -0.04188 0.01656  0.01434  6.40 1.00 0.01100
## inheritautosomal   0.67320 0.33488  0.27457  4.04 1.00 0.04400
## steroids           1.45357 0.75420  0.60175  3.71 1.00 0.05400
## propylac          -0.60236 0.39406  0.31774  2.34 1.00 0.13000
## frailty(id)                                 33.01 23.88 0.10000
##
##                    exp(coef) exp(-coef) lower .95 upper .95
## sexfemale             0.4477     2.2336    0.1763    1.1372
## treatrIFN-g           0.3712     2.6940    0.2069    0.6660
## age                   0.9590     1.0428    0.9284    0.9906
## inheritautosomal      1.9605     0.5101    1.0170    3.7793
## steroids              4.2784     0.2337    0.9757   18.7610
## propylac              0.5475     1.8264    0.2529    1.1853
##
## Iterations: 7 outer, 29 Newton-Raphson
##      Variance of random effect= 0.4964725   I-likelihood = -321.1
## Degrees of freedom for terms=  0.7  0.8  0.8  0.7  0.6  0.7 23.9
## Concordance= 0.818  (se = 0.035 )
## Likelihood ratio test= 90.29  on 28.11 df,    p=1.915e-08
```

(A14) The default in `coxph` is to fit a gamma frailty model.

(Q15) How do the estimates of the covariate effects here compare to the ones from the marginal model with `+ cluster(id)`? What is the estimated frailty variance? Do you think it is significant?

(A15) The point estimates (log-hazard ratios) are similar, with only small changes. The p-values are generally smaller in the frailty model. In some sense, that is because the random effect explaines a part of previously unexplained variability in the data. The estimated frailty variance is 0.49. That is definetely far from 0. Whether it is significant or not it's hard to tell, since we don't know the variance of this estimate. The best idea would be to do a likelihood ratio test between this model and a frailty-less model. Note that there is a p-value of a Wald-type test in the row corresponding to `frailty(id)`. Unfortunately, this test does lacks a

theoretical foundation (it's more of a by-product of the estimation procedure, the same like the *degrees of freedom* (`DF` column) for the frailty).

(Q16) The output also shows a likelihood ratio test. Which models does this test compare? What type of hazard ratios are shown in the `exp(coef)` column? Are they relevant for a particular individual or for the whole population?

(A16) That likelihood ratio test is between the frailty model and a model with no frailty and no covariates. That is difficult to find out; a hint is that the "degrees of freedom" is the sum of all elements in the `DF` column. The hazard ratios in this output are *conditional* effects that are interpreted at an individual level.

(Q17) We can also fit a model with log-normal frailty. Do you see a big difference in terms of estimates between this distribution and the gamma frailty?

```
mod_cal_fr_lognorm <- coxph(Surv(tstart, tstop, status) ~ sex + treat + age + inherit +
                    steroids + propylac + frailty(id, distribution = "gaussian"),
                ties = "breslow", cgd)
summary(mod_cal_fr_lognorm)
```

```
## Call:
## coxph(formula = Surv(tstart, tstop, status) ~ sex + treat + age +
##      inherit + steroids + propylac + frailty(id, distribution = "gaussian"),
##      data = cgd, ties = "breslow")
##
##   n= 203, number of events= 76
##
##                         coef     se(coef) se2      Chisq DF    p
## sexfemale              -0.73200 0.47519  0.40953   2.37  1.00 0.1200
## treatrIFN-g            -0.96966 0.29935  0.26680  10.49  1.00 0.0012
## age                    -0.04029 0.01658  0.01443   5.91  1.00 0.0150
## inheritautosomal        0.62470 0.34053  0.27776   3.37  1.00 0.0670
## steroids                1.37722 0.73363  0.60038   3.52  1.00 0.0600
## propylac               -0.58179 0.38928  0.31768   2.23  1.00 0.1400
## frailty(id, distribution                         32.72 22.04 0.0670
##
##                 exp(coef) exp(-coef) lower .95 upper .95
## sexfemale          0.4809     2.0792    0.1895    1.2206
## treatrIFN-g        0.3792     2.6370    0.2109    0.6819
## age                0.9605     1.0411    0.9298    0.9922
## inheritautosomal   1.8677     0.5354    0.9582    3.6405
## steroids           3.9639     0.2523    0.9411   16.6949
## propylac           0.5589     1.7892    0.2606    1.1986
##
## Iterations: 6 outer, 30 Newton-Raphson
##      Variance of random effect= 0.4640607
## Degrees of freedom for terms=  0.7  0.8  0.8  0.7  0.7  0.7 22.0
## Concordance= 0.821  (se = 0.035 )
## Likelihood ratio test= 92.15  on 26.33 df,    p=3.113e-09
```

(A17) I don't see a big difference. Usually you would expect results to be similar.

frailtyEM

Although the `coxph` and `frailty` functions are sometimes enough to get an idea of the model fit, we can get some more results with the `frailtyEM` package. If you don't have it already installed, you can do that with `install.packages("frailtyEM")`. The syntax is similar but a bit different from `coxph`.

(Q18) How do you identify the clusters with the `emfrail()` function? What is the default frailty distribution? What frailty distributions can be fitted with this function?

```
library(frailtyEM)
?emfrail
?emfrail_dist
```

(A18) In `frailtyEM` we use `+cluster()` to tell `R` which observations belong to the same cluster/individual. This plays a similar role with `+frailty()` from `coxph`. The default frailty distribution is the gamma, but also positive stable and PVF can be fitted. The PVF also has a parameter `m`, which by default is -0.5, defining the inverse Gaussian distribution. With positive `m`, we would obtain compound Poisson distributions.

(Q19) A gamma frailty fit is very similar to what we have seen with `coxph`. Do you notice any differences in estimates with the `coxph` fit?

```
mod_gam_1 <- emfrail(Surv(tstart, tstop, status) ~  sex + treat + age + propylac + inherit +
                     steroids + cluster(id), cgd)
summary(mod_gam_1)
```

```
## Call:
## emfrail(formula = Surv(tstart, tstop, status) ~ sex + treat +
##     age + propylac + inherit + steroids + cluster(id), data = cgd)
##
## Regression coefficients:
##                        coef exp(coef)  se(coef) adjusted se          z
## sexfemale         -0.804123  0.447480  0.475733    0.477113 -1.690281
## treatrIFN-g       -0.991684  0.370951  0.299325    0.299997 -3.313073
## age               -0.041893  0.958973  0.016605    0.016605 -2.522952
## propylac          -0.601626  0.547920  0.393676    0.393932 -1.528225
## inheritautosomal   0.674073  1.962214  0.334486    0.334486  2.015253
## steroids           1.452841  4.275245  0.754092    0.754893  1.926611
##                        p
## sexfemale         0.0910
## treatrIFN-g       0.0009
## age               0.0116
## propylac          0.1265
## inheritautosomal 0.0439
## steroids          0.0540
## Estimated distribution: gamma / left truncation: FALSE
##
## Fit summary:
## Commenges-Andersen test for heterogeneity: p-val  0.0449
## (marginal) no-frailty Log-likelihood: -323.606
## (marginal) Log-likelihood: -321.087
## LRT: 1/2 * pchisq(5.04), p-val 0.0124
##
## Frailty summary:
## theta = 2.019 (1.29) / 95% CI: [0.744, 23.631]
## variance = 0.495 / 95% CI: [0.042, 1.344]
## Kendall's tau: 0.199 / 95% CI: [0.021, 0.402]
## Median concordance: 0.195 / 95% CI: [0.02, 0.406]
## E[log Z]: -0.268 / 95% CI: [-0.805, -0.021]
## Var[log Z]: 0.637 / 95% CI: [0.043, 2.572]
## Confidence intervals based on the likelihood function
```

(A19) The estimates are identical, although what happens behind the scenes is quite different.

(Q20) You noticed probably that here is more output here. The field `LRT` also shows a likelihood ratio test. From the output shown here, what models does this test contrast? Do you see any other evidence for the presence of frailty from the output? What is θ in this case?

(A20) This time the LRT is a test between the model with frailty and the same model without frailty, but with the same covariates. It is indeed significant. The Commenges-Andersen test is a generally less powerful score test for heterogeneity, which is also siginificant in this case. θ is the inverse of the frailty variance.

(Q21) Look into the contents of the `emfrail` fit with `str()`. Can you find the empirical Bayes frailty estimates? How are they different from the ones from the `coxph()` fit?

```
plot(exp(mod_cal_fr$frail), mod_gam_1$frail, xlab = "coxph exp($frail)", ylab = "emfrail $frail")
abline(0,1)
```
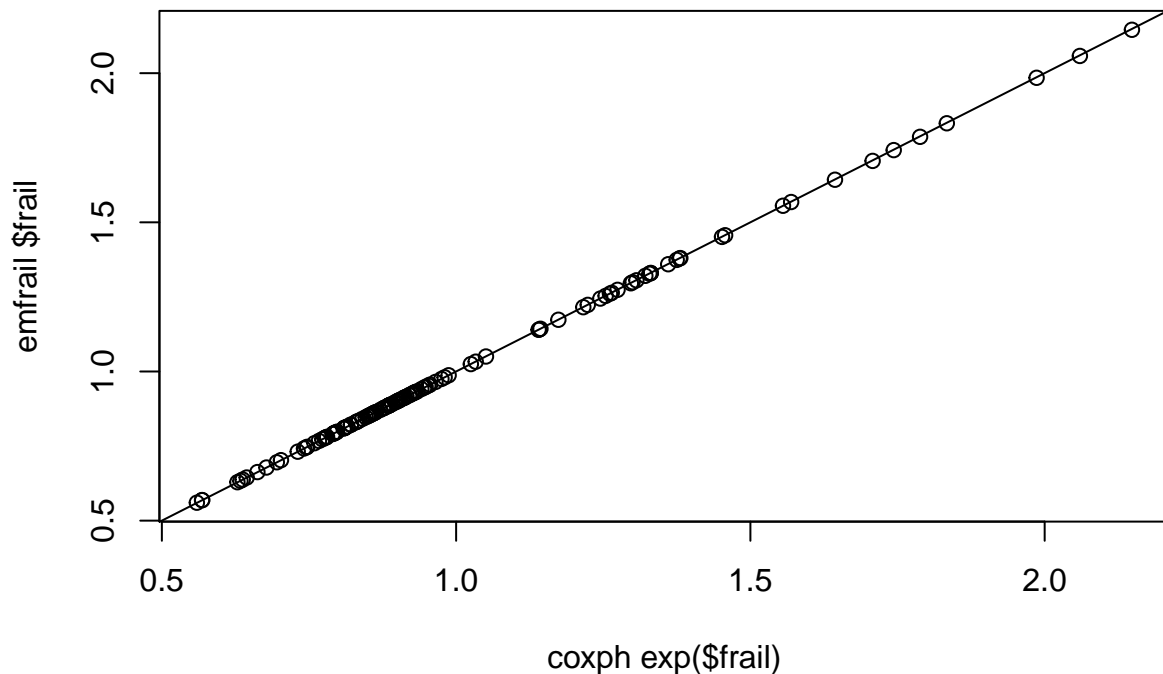

They are identical, just that with `frailtyEM` what is in '$frail$ are the actual estimates, not the logarithm.

(Q22) We can also calculate the empirical Bayes estimates by hand (only for the gamma frailty). The formula is:

$$\widehat{z}_i = \frac{\theta + N_i}{\theta + \Lambda_i}$$

where N_i is the number of events observed on individual i, Λ_i is the summed up cumulative intensity from cluster i. What do you think θ is? How is it related to the estimated frailty variance? Take a look at the summary of `mod_gam_1`. Here are the ingredients that we need to calculate that:

```
exp(mod_gam_1$logtheta)
mod_gam_1$nevents_id
mod_gam_1$residuals$group
```

(Q23) Calculate the $\widehat{z}_i$ from this. Does it agree with the empirical Bayes estimates from `coxph` or from `frailtyEM`?

(A23) Let's see:

```
myzi <- (exp(mod_gam_1$logtheta) + mod_gam_1$nevents_id) /
  (exp(mod_gam_1$logtheta) + mod_gam_1$residuals$group)
plot(myzi, mod_gam_1$frail)
abline(0,1)
```
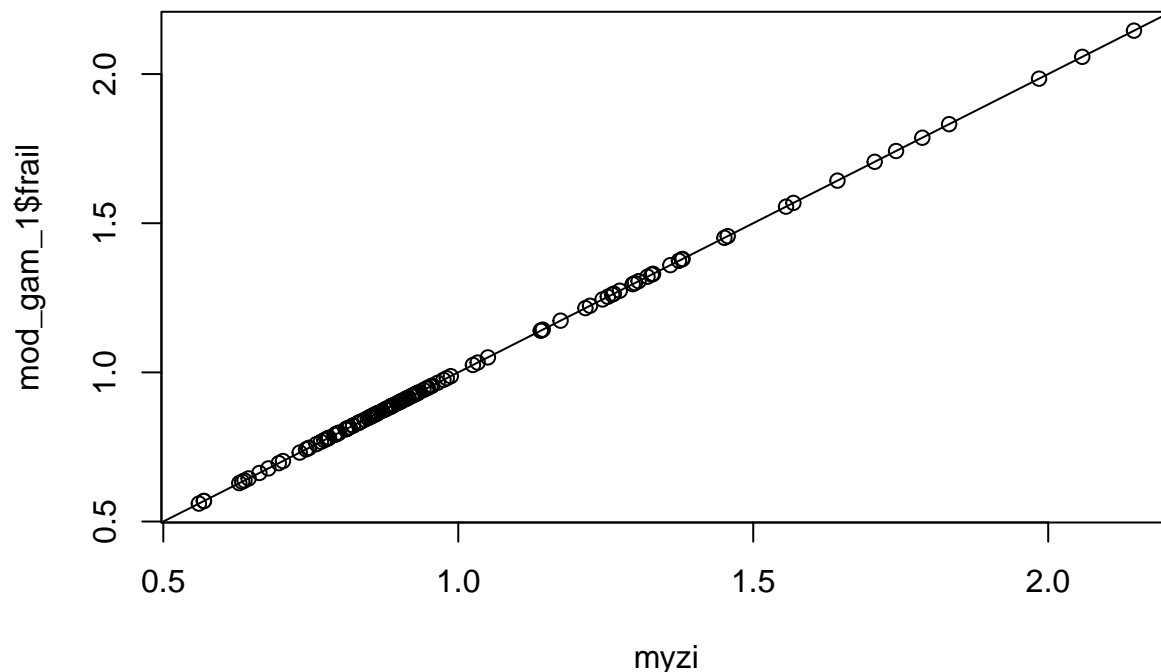
They are the same as the ones from `frailtyEM` (and the `exp()` of the ones from `coxph`).

(Q24) Calculate the *sample* mean and variance of the estimated $\widehat{z}_i$. Does the sample variance agree with the estimated frailty variance? Should they? (A24) Well:

```
mean(myzi)
```

```
## [1] 0.9999277
```

```
var(myzi)
```

```
##              [,1]
## [1,] 0.09718679
```

The mean is always 1 because of the parametrizations. Note that the sample variance of the estimates is *not* the estimate of the variance (and there's no reason why it should be).

We can make several types of plots from an `emfrail` fit. I will use those based on the `ggplot2` package here. More info on those can be found in `?autoplot.emfrail`. For conventional plots, `?plot.emfrail` describes what can be obtained. The advantage of the `ggplot2` plots is that they are objects in R and can be easily edited after they are produced.

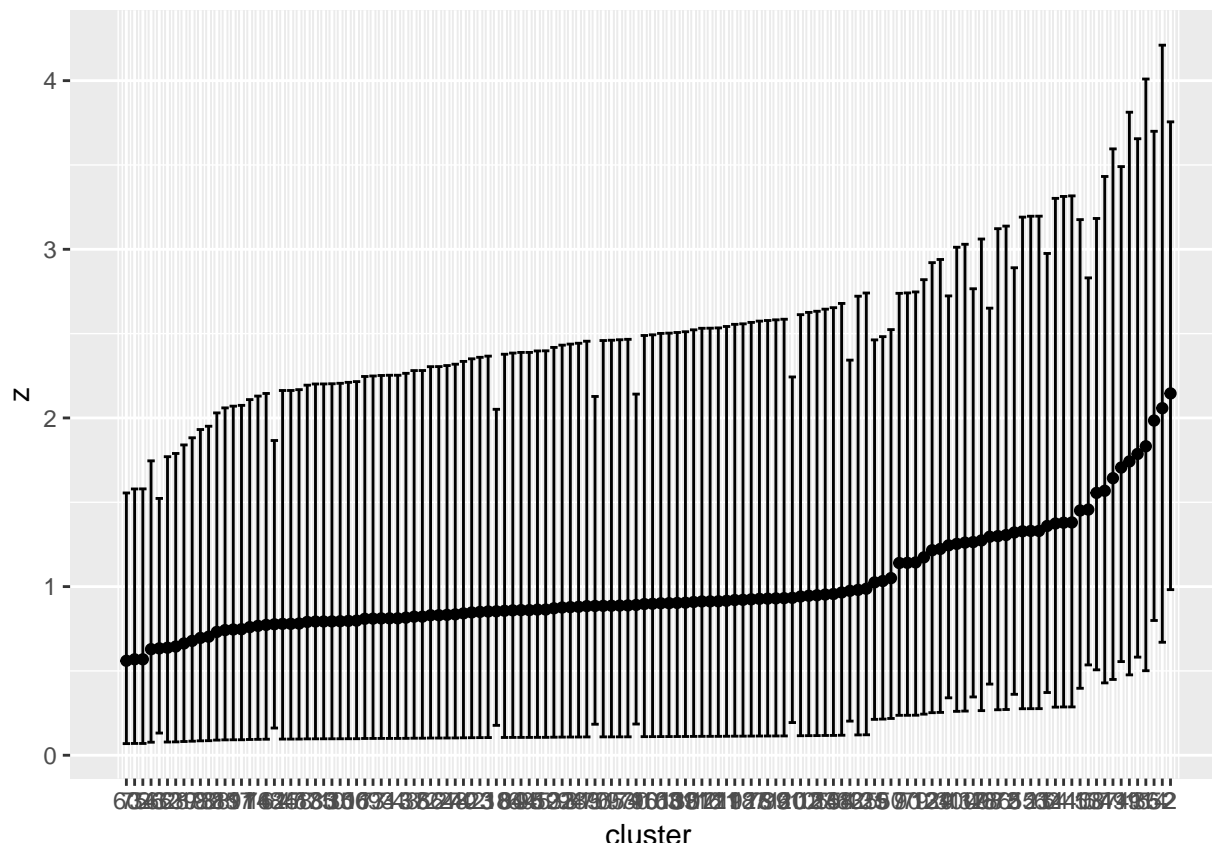Start with a histogram of the frailty estimates:

```
autoplot(mod_gam_1, type = "hist")
```

15
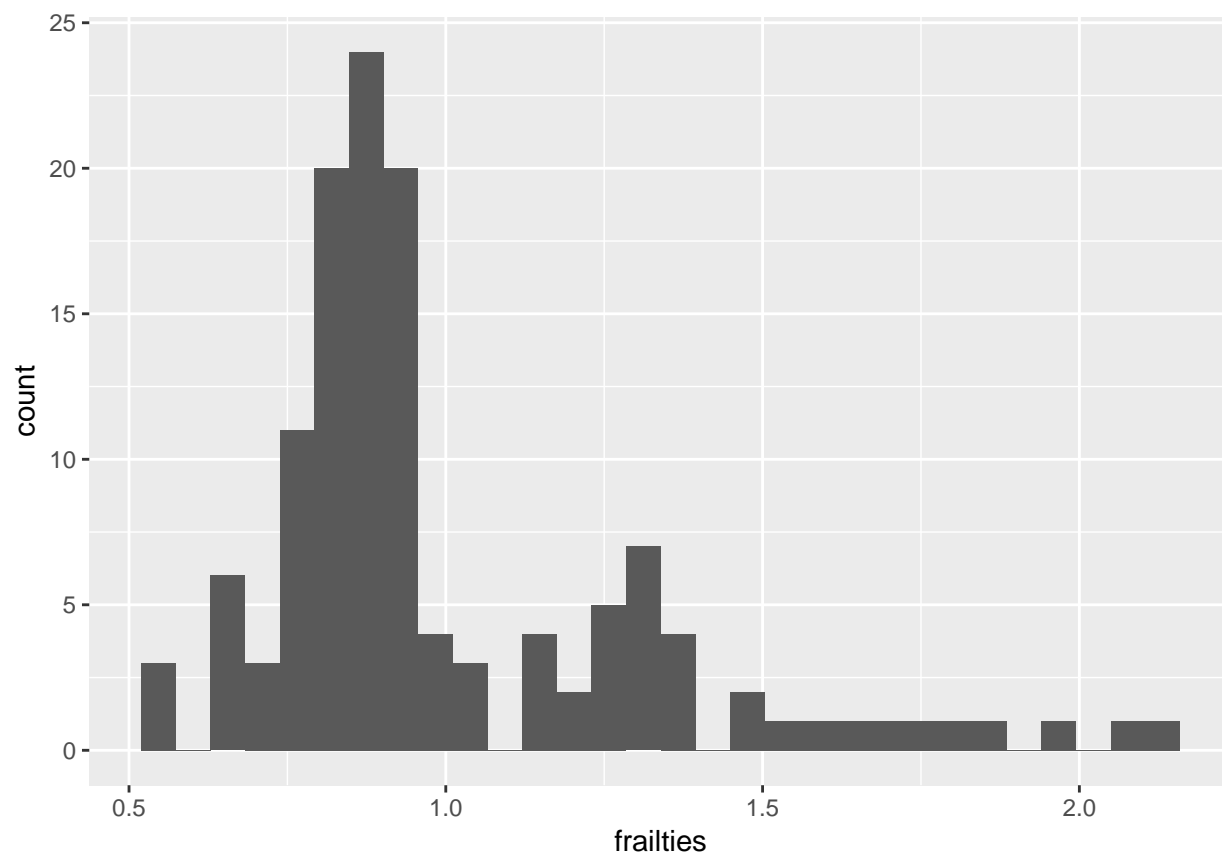
From the histogram it's quite difficult to tell which individual has which value of the frailty. We can make a so-called *catterpillar plot*:

```r
autoplot(mod_gam_1, type = "frail")
```

However, it is still difficult to tell to which individual which frailty value belongs to, since there are so many of them. We can use an interactive visualization for this: install the `plotly` package with `install.packages("plotly")` and then try this:

```r
library(plotly)
ggplotly(autoplot(mod_gam_1, type = "frail"))
```

(Q25) Which are the individuals with the highest and lowest frailty estimates?

(A25) Individual 60 has the lowest frailty and individual 2 has the highest frailty. If you have `plotly` installed, then you can just hover the mouse and see the name of each individual.

(Q26) From the output of `mod_gam_1` we can already read a number of estimates that correspond to log-hazard ratios. How would a marginal hazard ratio look like? Let's see this, between a treated and an untreated female, with `age = 24`, with baseline values for the other covariates. What do you expect that will happen with the marginal hazard ratio? How drastic do you think that the effect of the frailty will be, given the strength of the evidence for the frailty?

(A26) The marginal hazard ratio would be shrunken towards 1. Remember that the frailty was significant (p = 0.012). But I do not think that the shrinking will be extreme - I would maybe expect that if the p-value of the frailty would be *very* small. However, the amount of shrinking does not only depend on the strength of the frailty effect, but also on the other estimates. The only way to find out for sure is to plot it!

```r
sex <- rep("female", 2)
treat <- c("placebo", "rIFN-g")
age <- c(24, 24)
propylac = c(0, 0)
inherit <- rep("X-linked", 2)
steroids <- rep(0, 2)
newdata <- data.frame(sex, treat, age, propylac, inherit, steroids)
```

```
autoplot(mod_gam_1, type = "hr", newdata = newdata)
```
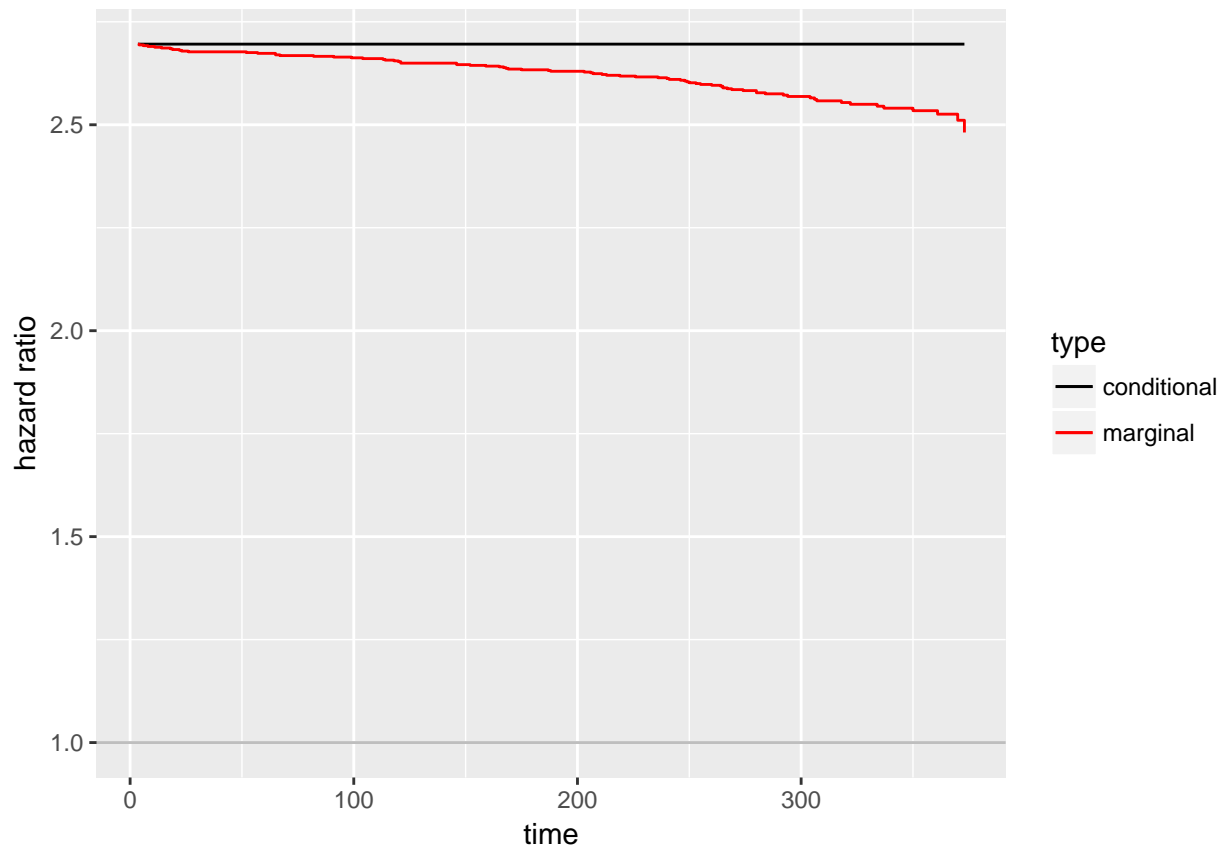

(Q27) Now suppose that we want to predict the cumulative intensity (hazard) for a certain individual. There comes a question: which one is more relevant, the conditional one or the marginal one? Which one would we observe at the level of the population? Which one would be more relevant for a patient?

(A27) The marginal one is what we would observe at the level of the population. For health economics for example that would be more relevant. The patient however cares about what will happen to her, so the conditional hazard ratio is in that case more relevant.

(Q28) Let's say we take the first woman, on placebo, from the previous question. What does this return? Can you tell what are the fields in this result? Hint: check with `?predict.emfrail`

```
head(predict(mod_gam_1, newdata = newdata[1,]))
```

```
##   time        lp      cumhaz    se_logH      cumhaz_l   cumhaz_r se_logH_adj
## 1  3.5 -1.809551 0.000000000 0.0000000 0.0000000000 0.00000000   0.0000000
## 2  4.0 -1.809551 0.004522549 1.0891757 0.0005348823 0.03823916   1.0892153
## 3  6.0 -1.809551 0.009045098 0.8284345 0.0017833459 0.04587658   0.8284866
## 4  8.0 -1.809551 0.013567648 0.7208586 0.0033029145 0.05573292   0.7209184
## 5 11.0 -1.809551 0.018090197 0.6605330 0.0049566276 0.06602377   0.6605982
## 6 14.0 -1.809551 0.022612746 0.6215334 0.0066879556 0.07645629   0.6216028
##      cumhaz_l_a cumhaz_r_a  survival survival_l survival_r survival_l_a
## 1 0.0000000000 0.00000000 1.0000000  1.0000000  1.0000000    1.0000000
## 2 0.0005348408 0.03824213 0.9954877  0.9994653  0.9624827    0.9994653
## 3 0.0017831639 0.04588126 0.9909957  0.9982182  0.9551598    0.9982184
## 4 0.0033025272 0.05573945 0.9865240  0.9967025  0.9457917    0.9967029
## 5 0.0049559934 0.06603221 0.9820724  0.9950556  0.9361086    0.9950563
```

```
## 6 0.0066870462 0.07646669 0.9776410  0.9933344  0.9263934     0.9933353
##    survival_r_a survival_m survival_m_l survival_m_r survival_m_l_a
## 1     1.0000000  1.0000000    1.0000000    1.0000000      1.0000000
## 2     0.9624799  0.9954927    0.9994653    0.9628270      0.9994654
## 3     0.9551554  0.9910157    0.9982190    0.9556504      0.9982192
## 4     0.9457855  0.9865688    0.9967052    0.9465065      0.9967056
## 5     0.9361007  0.9821516    0.9950617    0.9370983      0.9950623
## 6     0.9263838  0.9777639    0.9933453    0.9277026      0.9933462
##    survival_m_r_a     cumhaz_m    cumhaz_m_l cumhaz_m_r cumhaz_m_l_a
## 1       1.0000000 0.000000000 0.0000000000 0.00000000 0.0000000000
## 2       0.9628242 0.004517491 0.0005348114 0.03788152 0.0005347699
## 3       0.9556461 0.009024896 0.0017825587 0.04536308 0.0017823769
## 4       0.9465004 0.013522259 0.0033002155 0.05497749 0.0032998289
## 5       0.9370906 0.018009626 0.0049505528 0.06496712 0.0049499201
## 6       0.9276933 0.022487041 0.0066769021 0.07504407 0.0066759957
##    cumhaz_m_r_a
## 1   0.00000000
## 2   0.03788444
## 3   0.04536766
## 4   0.05498385
## 5   0.06497530
## 6   0.07505409
```

This returns a prediction of the cumulative hazard and survival (rather, $\exp(-\Lambda(t))$), both marginal and conditional, with 95% confidence intervals.

(Q29) We can look at the plot for the cumulative intensity of our patient of interest. How do you expect that the marginal cumulative intensity will compare to the conditional cumulative intensity?

```
autoplot.emfrail(mod_gam_1, type = "pred", newdata = newdata[1,])
```
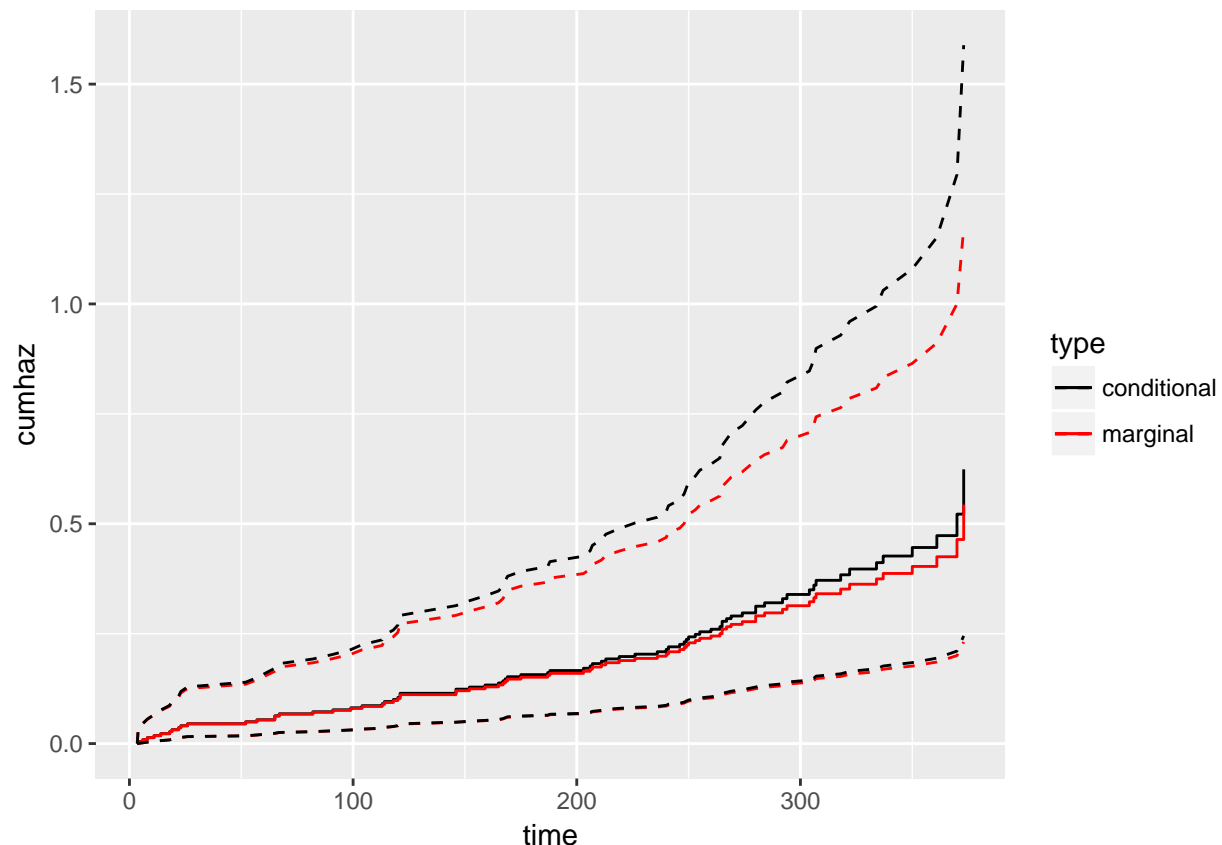
The dotted lines represent a 95% confidence band.

(A29) Usually the marginal hazard/intensity is "dragged down" by the frailty.

Other distributions

(Q30) Let's play a bit with the `distribution` argument in `emfrail`. I'll fit 3 more distributions here. Which one is the inverse Gaussian? (hint: `?emfrail_dist()`)

```
mod_pvf1 <- emfrail(formula = Surv(tstart, tstop, status) ~ sex + treat +
    age + propylac + inherit + steroids + cluster(id),
    distribution = emfrail_dist(dist = "pvf"),
    data = cgd)

mod_pvf2 <- emfrail(formula = Surv(tstart, tstop, status) ~ sex + treat +
    age + propylac + inherit + steroids + cluster(id),
    distribution = emfrail_dist(dist = "pvf", pvfm = 0.5),
    data = cgd)

mod_stab <- emfrail(formula = Surv(tstart, tstop, status) ~ sex + treat +
    age + propylac + inherit + steroids + cluster(id),
    distribution = emfrail_dist(dist = "stable"),
    data = cgd)
```

(A30) The default in `emfrail_dist(dist = "pvf")` is to have `pvfm = -0.5`. That is the inverse Gaussian distribution. The second model has a compound Poisson distribution.

(Q31) Look at the output of the positive stable frailty model. Is the frailty significant here? Compare the

estimates with those from the gamma frailty model and the marginal model with `+ cluster(id)`. To which ones is it closer? Is there anything extra in the output? What do you think it means?

```
summary(mod_stab)
```

```
## Call:
## emfrail(formula = Surv(tstart, tstop, status) ~ sex + treat +
##     age + propylac + inherit + steroids + cluster(id), data = cgd,
##     distribution = emfrail_dist(dist = "stable"))
##
## Regression coefficients:
##                     coef exp(coef)  se(coef) adjusted se         z
## sexfemale       -0.669860  0.511780  0.436831    0.441698 -1.533453
## treatrIFN-g     -1.045188  0.351626  0.293880    0.294395 -3.556511
## age             -0.043091  0.957824  0.015464    0.015505 -2.786602
## propylac        -0.599076  0.549319  0.333476    0.340483 -1.796458
## inheritautosomal 0.634477  1.886036  0.330350    0.333529  1.920620
## steroids         1.400698  4.058033  0.609885    0.610805  2.296659
##                        p
## sexfemale         0.1252
## treatrIFN-g       0.0004
## age               0.0053
## propylac          0.0724
## inheritautosomal  0.0548
## steroids          0.0216
## Estimated distribution: stable / left truncation: FALSE
##
## Fit summary:
## Commenges-Andersen test for heterogeneity: p-val  0.0449
## (marginal) no-frailty Log-likelihood: -323.606
## (marginal) Log-likelihood: -323.416
## LRT: 1/2 * pchisq(0.38), p-val 0.269
##
## Frailty summary:
## theta = 24.099 (39.25) / 95% CI: [4.736, Inf]
## Kendall's tau: 0.04 / 95% CI: [0, 0.174]
## Median concordance: 0.038 / 95% CI: [0, 0.171]
## E[log Z]: 0.024 / 95% CI: [0, 0.122]
## Var[log Z]: 0.139 / 95% CI: [0, 0.768]
## Attenuation factor: 0.96 / 95% CI: [0.826, 1]
## Confidence intervals based on the likelihood function
```

(A31) This shows that in the positive stable (PS) model the frailty is not significant. In general it is a bad idea to fit the PS model when individuals do not have at least 2 events, since this distribution can not be estimated in that case.

(Q32) Optional: how do you think that the marginal and conditional hazard ratios would look like, if we did the same thing with `mod_stab`? Try it!

(A32) The marginal and conditional hazard ratios would be parallel lines, but quite close to each other, since the frailty was not significant.

(Q33) How about model `mod_pvf2`? Notice anything in its summary that wasn't there before?

```
summary(mod_pvf2)
```

```
## Call:
```

```
## emfrail(formula = Surv(tstart, tstop, status) ~ sex + treat +
##     age + propylac + inherit + steroids + cluster(id), data = cgd,
##     distribution = emfrail_dist(dist = "pvf", pvfm = 0.5))
##
## Regression coefficients:
##                      coef exp(coef)  se(coef) adjusted se         z
## sexfemale        -0.816874  0.441811  0.476519    0.478196 -1.714251
## treatrIFN-g      -0.985376  0.373299  0.298845    0.300314 -3.297282
## age              -0.042005  0.958865  0.016567    0.016568 -2.535387
## propylac         -0.598970  0.549377  0.393811    0.393876 -1.520958
## inheritautosomal  0.681071  1.975994  0.332952    0.332952  2.045553
## steroids          1.467207  4.337103  0.764430    0.765591  1.919347
##                       p
## sexfemale        0.0865
## treatrIFN-g      0.0010
## age              0.0112
## propylac         0.1283
## inheritautosomal 0.0408
## steroids         0.0549
## Estimated distribution: pvf / left truncation: FALSE
## PVF m = 0.5
##
## Fit summary:
## Commenges-Andersen test for heterogeneity: p-val  0.0449
## (marginal) no-frailty Log-likelihood: -323.606
## (marginal) Log-likelihood: -321.048
## LRT: 1/2 * pchisq(5.12), p-val 0.0118
##
## Frailty summary:
## theta = 2.058 (1.25) / 95% CI: [0.811, 22.007]
## variance = 0.486 / 95% CI: [0.045, 1.233]
## Estimated mass at 0: 0.002081216
## Confidence intervals based on the likelihood function
```

(A33) The estimated mass at 0 field is new. That is a feature of the compound Poisson distributions.

(Q34) Compare the likelihoods of all the frailty models we fitted here. Which one is the highest? Was this to be expected? Think of one of the first questions, about how many individuals have how many events.

```r
lapply(list(mod_gam_1, mod_stab, mod_pvf1, mod_pvf2), logLik)
```

```
## [[1]]
## 'log Lik.' -321.0875 (df=7)
##
## [[2]]
## 'log Lik.' -323.4158 (df=7)
##
## [[3]]
## 'log Lik.' -321.292 (df=7)
##
## [[4]]
## 'log Lik.' -321.0475 (df=7)
```

(A34) The compound Poisson has the highest log-likelihood out of all the models. That is to be expected in some sense, since we have seen that most individuals have 0 events: so an estimated mass at 0 is to be expected.

Lastly, we will note that the estimated *mass at 0* depends heavily of the `pvfm` parameter, that defines the large classes of distribution from within the PVF family. This is not estimated by `frailtyEM`, but rather set by the user. We can however try to estimate it, just by trying out a number of values.

(Q35) I collect here the log-likelihood values for every distribution (for different values of `pvfm`). What is the value of `m` for which the highest log-likelihood of the model is obtained?

```
mvals <- seq(from = 0.1, to = 2, by = 0.2)
models <- lapply(mvals, function(x) emfrail(formula = Surv(tstart, tstop, status) ~ sex +
    treat + age + propylac + inherit + steroids + cluster(id), data = cgd,
    distribution = emfrail_dist(dist = "pvf", pvfm = x)) )

likelihoods <- sapply(models, function(x) x$loglik[2])

plot(mvals, likelihoods)
```
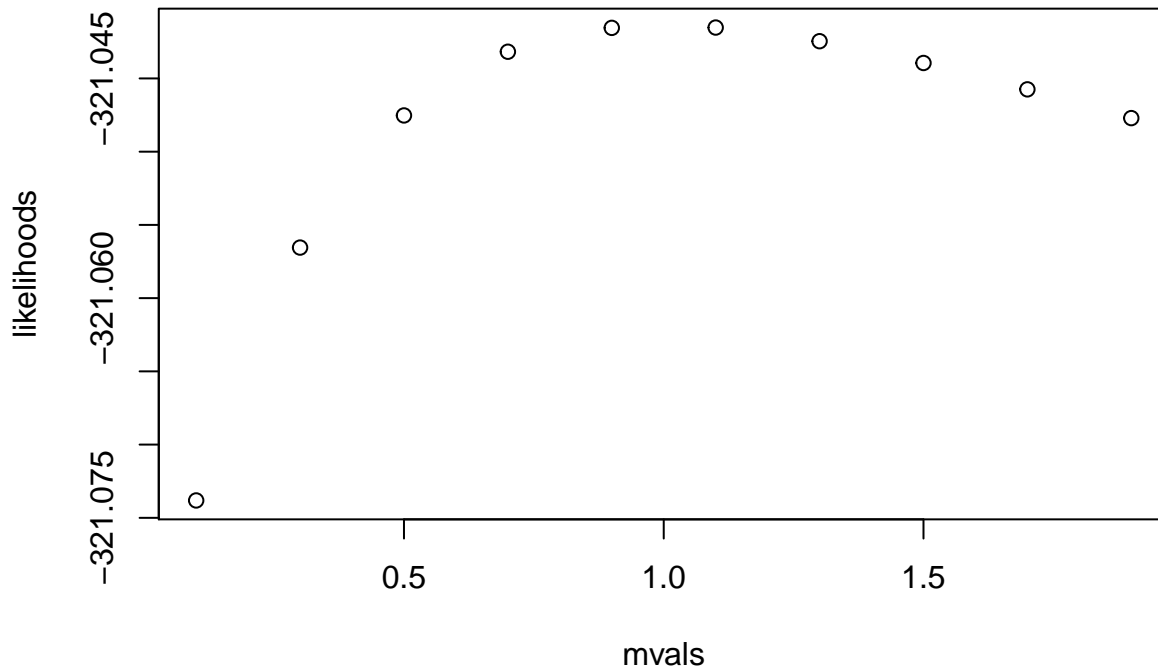

(A35)

An `m` around 1 gives the best fit.

Gap time

Another option with recurrent events data is to analyze it in gap-time. For this we will have to add another column in our data:

```
cgd$gap <- cgd$tstop - cgd$tstart
```

The right hand side of the formulas we used before would stay the same, but the left hand side should now be:

```
mod_gap_fr <- coxph(Surv(gap, status) ~ sex + treat + age + inherit +
                  steroids + propylac + frailty(id),
              ties = "breslow", cgd)
summary(mod_gap_fr)

## Call:
## coxph(formula = Surv(gap, status) ~ sex + treat + age + inherit +
##      steroids + propylac + frailty(id), data = cgd, ties = "breslow")
```

```
##
##   n= 203, number of events= 76
##
##                   coef    se(coef) se2     Chisq DF    p
## sexfemale        -0.88169 0.53542  0.42001  2.71 1.00 0.1000
## treatrIFN-g      -1.05828 0.33091  0.27776 10.23 1.00 0.0014
## age              -0.04506 0.01857  0.01487  5.89 1.00 0.0150
## inheritautosomal  0.72385 0.37952  0.27900  3.64 1.00 0.0560
## steroids          1.55948 0.91448  0.62636  2.91 1.00 0.0880
## propylac         -0.72039 0.46262  0.33332  2.42 1.00 0.1200
## frailty(id)                               57.94 39.03 0.0260
##
##                  exp(coef) exp(-coef) lower .95 upper .95
## sexfemale           0.4141     2.4150    0.1450    1.1826
## treatrIFN-g         0.3471     2.8814    0.1814    0.6638
## age                 0.9559     1.0461    0.9218    0.9914
## inheritautosomal    2.0624     0.4849    0.9802    4.3393
## steroids            4.7563     0.2102    0.7923   28.5546
## propylac            0.4866     2.0552    0.1965    1.2048
##
## Iterations: 6 outer, 31 Newton-Raphson
##     Variance of random effect= 0.9728984   I-likelihood = -340.5
## Degrees of freedom for terms=  0.6  0.7  0.6  0.5  0.5  0.5 39.0
## Concordance= 0.852  (se = 0.036 )
## Likelihood ratio test= 119.6  on 42.52 df,    p=2.925e-09
```

(Q36) Why can't we use `Surv(tstop, status)`? What would that correspond to? What is the time scale now?

(A36) Using `Surv(tstop, status)` would be equivalent with saying that each observation is from 0 to the `tstop`, and that would count too much at-risk time. We are using the time scale *time since previous event* now.

Look at the first individual:

```
cgd[cgd$id == 1, c("treat", "sex", "age", "inherit", "gap", "status")]
```

```
##    treat    sex age   inherit gap status
## 1 rIFN-g female  12 autosomal 219      1
## 2 rIFN-g female  12 autosomal 154      1
## 3 rIFN-g female  12 autosomal  41      0
```

In `mod_gap_fr`, R doesn't know that we are talking about recurrent events, since the syntax is exactly the same as for clustered data. (Q37) Do you think that matters here?

(A37) It does not actually! Clustered failures and recurrent events in gap-time are actually estimated in the same way.

(Q38) Below is a Kaplan-Meier curve. What do you think that this means here? Do you think that, if we analyze the models in gap time, the survival is more meaningful?

```
plot(survfit(coxph(Surv(gap, status) ~ 1, cgd)))
```
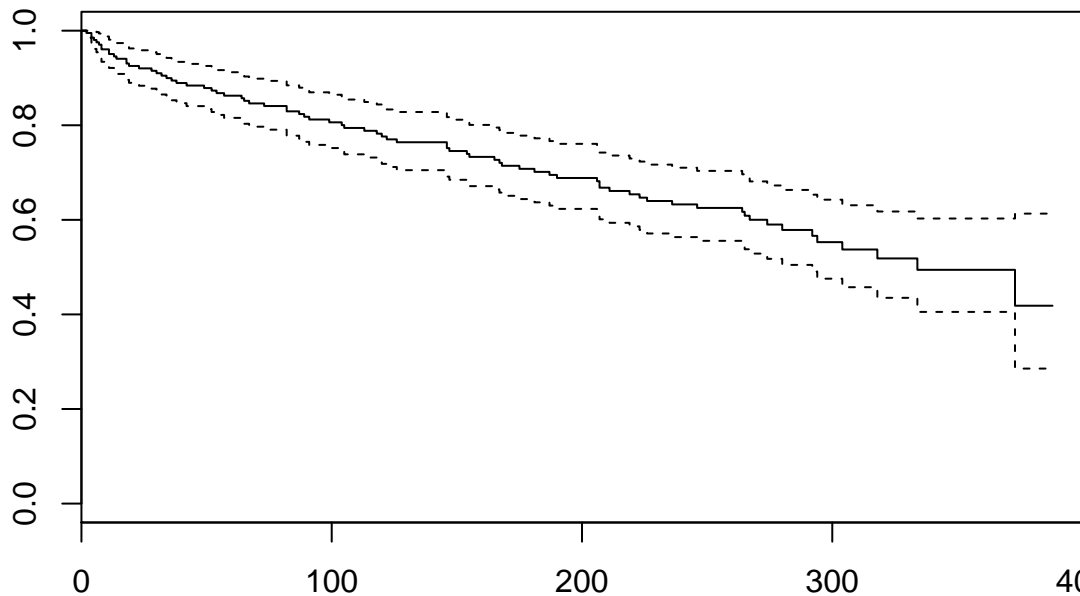
(A38) This curve is the "survival" of the gaptime - it tells us for example that about 70% of the gaptimes are large than 200 (we look at time 200 on the x axis.)

(Q39) Compare the estimated conditional hazard ratios (coefficients) between `mod_cal_fr` and `mod_gap_fr`. Are there similarities? Are the differences notable from a clinical point of view, do you think? Would you expect to have large differences between the two models? Do you think anything is lost by looking at the gap times instead of the calendar time?

(A39) The clinical conclusions are pretty similar, although the estimates are different and have a different meaning now. For example, if the treatment increases the intensity of recurrent event process in calendar time, we would also expect that it makes the times between events longer. However, there is no 1-1 correspondence in general between these estimates.

(Q40) Furthermore, compare the frailty estimates from the gap time model and the calendar time model. Do you expect them to be similar? What is different between them? Take a look again at the formula for the gamma empirical Bayes estimates that we looked at before.

```r
plot(exp(mod_gap_fr$frail), exp(mod_cal_fr$frail), xlab = "gaptime frailty", ylab = "calendartime frail
abline(0,1)
```

$$(A40)$$
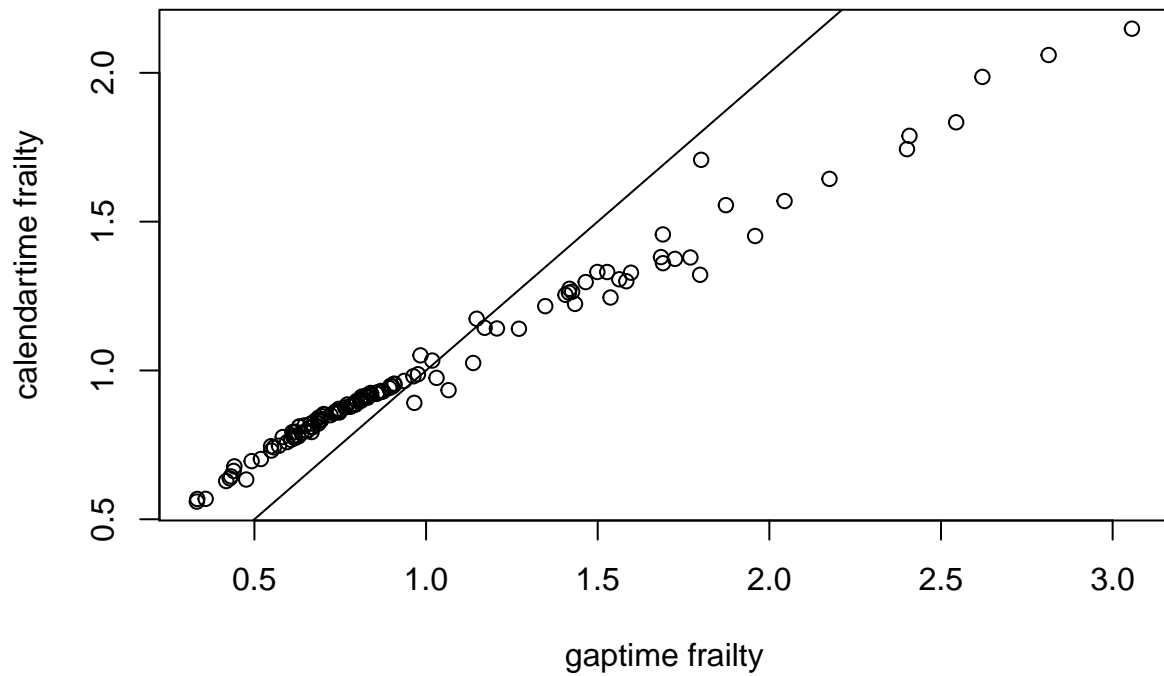
They are definetely positively correlated. If a high frailty individual has a high rate of events in calendar time, that would also correspond to shorter gaptimes. The numerator of the formula still contains the number of events for the individual, which stays unchanged in the two models.