Chapter 9

Correlated events and repeated events

9.1 Introduction

The survival models that we have considered depend upon the fundamental assumption that event times are independent. There are many settings where this assumption is unreasonable:

- **Clustered data**: There may be multiple observations for a single individual, or for a group with correlated times within the group. For example, the `diabetes` data set (in the `SurvCorr` package in `R`) includes data for 197 patients being treated for diabetic retinopathy — loss of vision due to diabetes. Each patient has two eyes, and all eyes are at risk until the event (measured serious vision loss). It would be unreasonable to assume that the loss of vision is independent between the two eyes.

  Below is an excerpt from the data set. One peculiarity is that the comparison was made within individuals, with each individual having one treated and one untreated eye, with the variable `TRT_EYE` recording which eye was treated. For more information about the data set see section 8.4.2 of [24].

```
> head(diabetes)
  ID LASER TRT_EYE AGE_DX ADULT  TIME1 STATUS1 TIME2 STATUS2
1  5     2       2     28     2 46.23       0 46.23       0
2 14     2       1     12     1 42.50       0 31.30       1
3 16     1       1      9     1 42.27       0 42.27       0
4 25     2       2      9     1 20.60       0 20.60       0
5 29     1       2     13     1 38.77       0  0.30       1
6 46     1       1     12     1 65.23       0 54.27       1
```

- **Multiple events**: A study may consider times of events that do not remove the subject from risk of further events. For example, a study of hospitalisation events for an elderly population will likely see some individuals being hospitalised multiple times.

- **Competing events**: Right censoring may be understood as a process with "multiple endpoints" — an individual leaves the study either through an event or through censoring. There is an asymmetry because we choose to treat the censoring distribution as a nuisance parameter. But in some studies there are multiple endpoints equally of interest: A medical study participant may exit through multiple relevant causes of death, or through the alternatives death or recovery. A participant in a sociological study of marriage may leave the "cohabiting" state through marriage or dissolution of the partnership. In each of these

cases, there are distinct outcomes, each with its own hazard rate, that each serves as "censoring" for observation of the others.

9.2 Time-to-first-event analysis

When we are confronted with multiple events for a single individual, one easy approach to eliminating the complication is to throw out the correlated data. A common approach, called *time-to-first-event analysis*, is to define the event of interest to be simply the first event. Later events for the same individual are simply ignored.

This trivially produces independent observations, and will yield results that are consistent and unbiased, but at the expense of losing a substantial amount of relevant information from the data. Of course, this approach is possible only when the correlated events correspond to a single individual, so that all covariates are identical. It would not apply to an example such as the diabetetic retinopathy study described above, where the two correlated events for one individual differ in their treatment group membership.

9.3 Clustered data

9.3.1 Stratified baseline

If there are a small number of large correlated groups of survival times, we may represent the correlation within the groups by using a semiparametric model and stratifying the baseline hazard by group. Note that there is no way to distinguish between a cluster of survival times being "dependent", and times sharing a group-determined hazard function.

Suppose we have k categories of individuals, $c_i = 1, \ldots, k$, each with its own baseline hazard $h_0^{(c)}(t)$, so that individuals in category c_i with covariates $\mathbf{x}_i(t)$ have hazard

$$h_i(t) = h_0^{(c_i)}(t) r(\beta, \mathbf{x}_i(t))$$

at time t. Then we have a partial likelihood for the observation that individual i_j had the unique event at time t_j

$$L_P(\beta) = \prod_{t_j} \frac{r(\beta, \mathbf{x}_{i_j}(t_j))}{\sum_{i \in \mathcal{R}_j : c_i = c_{i_j}} r(\beta, \mathbf{x}_i(t_j))}.$$

The only change, compared with the standard partial likelihood given in (5.5)

As an example, we consider a data set based on the NHANES (National Health and Nutrition Survey) wave 3, from which we have measures of systolic and diastolic blood pressure and about 15 years of survival follow-up. It is certainly not the case that men and women, or different ethnic groups, have the same baseline mortality rate. We could analyse the effect of blood pressure on mortality in the groups separately, but that will produce six different parameter estimates, each of which will be subject to more random noise. It is at least plausible that we would like to produce a joint estimate of the influence of blood pressure on mortality, that we suppose acts in approximately the same way against the background of distinct baseline mortality.

```
Call:
coxph(formula = with(nhanesC, Surv(age, age + yrsfu, eventhrt)) ~
meandias + meansys + strata(race) + strata(female), data = nhanesC)

coef exp(coef)  se(coef)      z       p
meandias -0.006495  0.993527  0.002763 -2.351  0.0187
meansys   0.011528  1.011595  0.001320  8.733 <2e-16

Likelihood ratio test=75.9  on 2 df, p=< 2.2e-16
```

```
10  n= 15295, number of events= 1459
```

If we plot the output of the `survfit` function applied to this result, we get the picture in Figure 9.1, showing six different baseline survival functions.

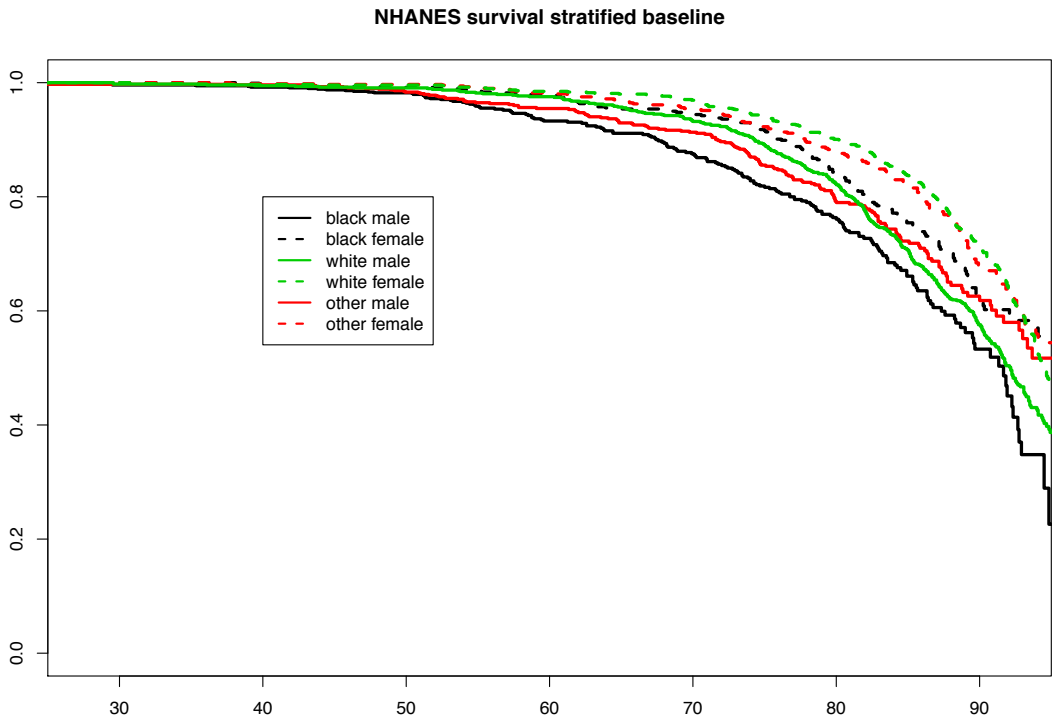

Figure 9.1: Baseline survival estimates for different population groups from the NHANES data.

9.3.2 The sandwich estimator for variance

In an example like the diabetic retinopathy study of course there is no possibility of allowing each "cluster" of two eyes its own baseline hazard. The clustering does not substantially change the parameter estimates, but it does affect the estimation of variance, and hence of the confidence interval for the parameters. In the extreme case, imagine that we had a dataset where each individual had the same survival time repeated multiple times. More observations reduces the variance estimate; but we would want to have a procedure that would be able to recognise that the duplicated observations are not actually providing additional information, and that would hence return the same variance estimate as the data set without duplication.

The standard approach in such cases is to replace the Fisher-information-based estimate for variance $J_n(\hat{\beta})^{-1}$ by the sandwich estimator

$$J_n(\hat{\beta})^{-1} V_n(\hat{\beta}) J_n(\hat{\beta})^{-1}, \tag{9.1}$$

where $V_n(\hat{\beta})$ is an estimate of the variance-covariance matrix of the score function. As described in [18], in the case of Cox regression, if we have C clusters of observations, with n_c individuals in

cluster c, we may take

$$V_n(\hat{\beta}) = \sum_{c=1}^{C} \hat{u}_c(\hat{\beta})\hat{u}_c(\hat{\beta})^T,$$

where

$$\hat{u}_c(\hat{\beta}) = \sum_{i=1}^{n_c} \sum_{t_j : c_j = c} \left\{ x_{ic}(t_j) - \bar{x}(t_j, \hat{\beta}) \right\} - \sum_{i=1}^{n_c} \sum_{t_j} \left\{ x_{ic}(t_j) - \bar{x}(t_j, \hat{\beta}) \right\} Y_{ic} \mathrm{e}^{\beta \cdot x_{ic}(t_j)} \, \mathrm{d}\hat{H}_0(t_j),$$

and

$$\mathrm{d}\hat{H}_0(t_j) = \frac{1}{\sum_{(i,c) \in \mathcal{R}_j} \mathrm{e}^{\beta \cdot x_{ic}(t_j)}}.$$

It is not necessary to memorise this formula (it is *not examinable*), but it is important to know that such a formula exists, and needs to be applied whenever one is analysing clustered data. It will automatically be applied in `R` if the `coxph` function is applied with an extra term in the formula of the form `+cluster(group)`. Commonly the clustering is by individual identifying number, so the term becomes `+cluster(id)`.

9.4 Multiple events

9.4.1 The Poisson model

The simplest model for repeated events is the Poisson model, taking the intensity for each individual to be a fixed constant α when at risk. Assuming changes in the at-risk process is independent of the counting process, the number of events observed for each individual, and in total, will be Poisson distributed, conditioned on the total time at risk. If we observe individual i for a total time $T_i = \int_0^\infty Y_i(t)dt$, and observe $N = N(\infty)$ events, then the log likelihood is

$$\ell(\alpha) = N \log \alpha - \alpha \sum T_i,$$

with the MLE

$$\hat{\alpha} = \frac{N}{\sum T_i}.$$

9.4.2 The Poisson regression model

A simple generalisation would be to say that each individual has Poisson number of events, with the intensity being constant during the time when that individual is at risk, and a function — to be determined — of some measured covariates. If individual i is at risk for total time T_i, the number of events N_i is Poisson distributed with parameter $T_i\alpha(\mathbf{x}_i)$. Conditioned on the time at risk the log likelihood is then

$$\ell(\alpha) = \sum_{i=1}^{n} N_i \log \alpha(\mathbf{x}_i) - T_i\alpha(\mathbf{x}_i).$$

The most common parametric form is $\alpha = \exp\{\beta \cdot \mathbf{x}\}$, where $\beta = (\beta_0, \ldots, \beta_p)$, and we take $x_{i0} \equiv 1$. The log likelihood then becomes

$$\ell(\beta) = \sum_{k=0}^{p} \beta_k \sum_{i=1}^{n} N_i x_{ik} - \sum_{i=1}^{n} T_i e^{\beta \cdot \mathbf{x}_i}. \tag{9.2}$$

The MLE then satisfies the equations

$$\sum_{i=1}^{n} N_i x_{ik} = \sum_{i=1}^{n} x_{ik} T_i e^{\hat{\beta}\cdot \mathbf{x}_i}.$$

This fits into the framework of GLM (generalised linear model), and may be fit in R using any of the standard GLM functions. Note that we are modelling $N_i \sim \text{Po}(\mu_i)$, where

$$\log \mu_i = \log T_i + \beta \cdot \mathbf{x}_i. \tag{9.3}$$

We call $\log T_i$ an *offset* in the model.

As an example, we consider data from a trial to determine the effect of nutritional supplements on prisoners' rate of disciplinary offenses. There were 771 prisoners, observed for anywhere from two weeks to half a year for a baseline period, after which half were randomly given the supplements, the others were given placebos, and they were observed (and offences recorded) for a variable period, mostly at least 10 weeks. We consider here only the treatment (second) period, and try to model the effect of the treatment, and of the difference in rates between prisons (which we call here A, B, and C). Thus our model is

$$\log \alpha = \beta_0 + \beta_1 \mathbf{1}\{\textit{treatment}\} + \beta_2 \mathbf{1}\{\text{prison B}\} + \beta_3 \mathbf{1}\{\text{prison C}\}.$$

```
#sdb0=start date baseline, sdt0= start date treatment, edt0=end date
#base.count=# events baseline, treat.count=#events treatment
# pgroup= indicator of active treatment
risktime=edt0-sdt0
poisreg=glm(treat.count~pris+pgroup,family=poisson,offset=log(risktime))
summary(poisreg)
Call:
glm(formula = treat.count ~ pris + pgroup, family = poisson,
offset = log(risktime))

Deviance Residuals:
Min       1Q   Median       3Q      Max
-4.5092  -1.5806  -0.7360   0.4907   8.3629

Coefficients:
Estimate Std. Error z value Pr(>|z|)
(Intercept) -2.39045    0.05971 -40.031   <2e-16 ***
prisB       -0.99509    0.06611 -15.051   <2e-16 ***
prisC       -1.99417    0.07069 -28.212   <2e-16 ***
pgroup      -0.10239    0.05051  -2.027   0.0426 *
---
Signif. codes:  0 `***` 0.001 `**` 0.01 `*` 0.05 `.` 0.1 ` ` 1

(Dispersion parameter for poisson family taken to be 1)

Null deviance: 2887.0  on 770  degrees of freedom
Residual deviance: 2133.2  on 767  degrees of freedom
AIC: 3349.5

Number of Fisher Scoring iterations: 6
```

This fitted model gives us a predicted expected number of events for each individual. The difference between the observed number of events and the expected number predicted by the model is the residual. In Figure 9.2 we plot the residuals against the fitted values. (This is the automatic output of the command `plot(poisreg)`, where `poisreg` is the output of the `glm` fit above.

This would be obvious from a casual examination of the data. The mean number of events is about 2, but some individuals have as many as 25, which is not something you would see in a Poisson distribution. These data are *over-dispersed*, meaning that their variance is higher than it would be for a Poisson distribution of the same mean.

We also note that the deviance residuals (which should be approximately standard normal distributed if the model is correct) range from -4.5 to $+8.36$. The sum of their squares, called the residual deviance, is 2133.2, which is much too large for a chi-squared variable on 767 degrees of freedom.

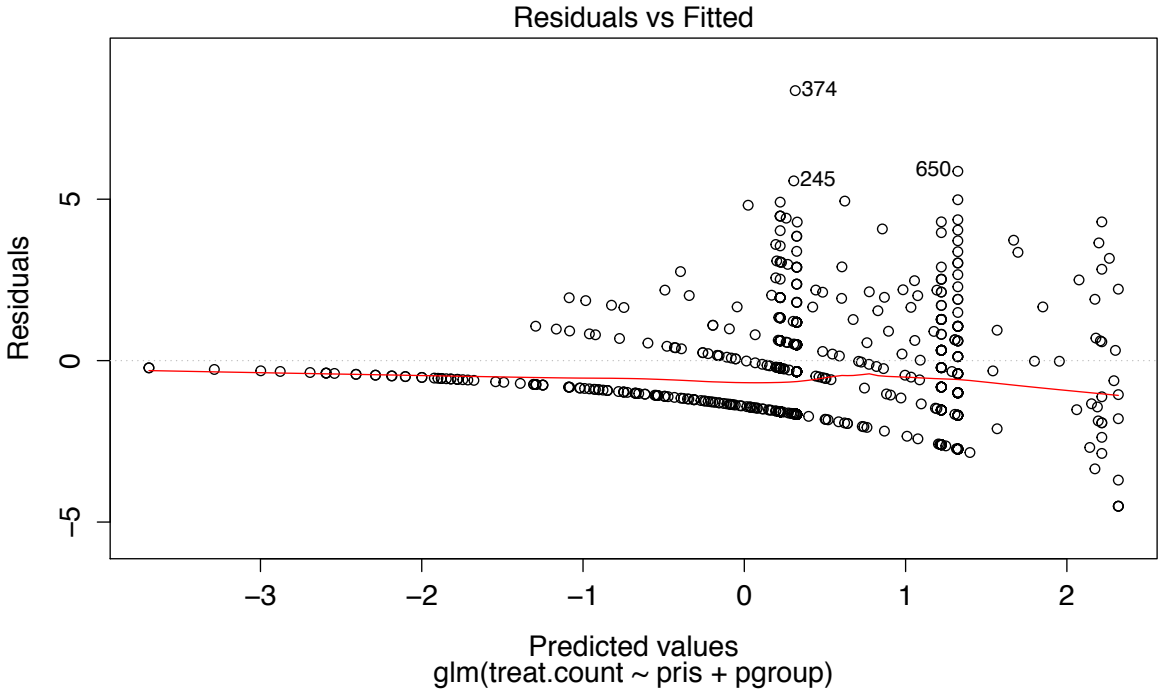

Figure 9.2: Residual plot for the prison-data Poisson regression.

9.4.3 The Andersen–Gill model

The Poisson regression model makes sense if we believe the event intensity is constant, or if all individuals Another popular generalisation of the Poisson model, introduced in 1982 by Andersen and Gill [2], is a semi-parametric relative-risk regression model, essentially equivalent to the Cox proportional hazards regression model. The only change is that the at-risk indicator $Y_i(t)$ for an individual will not, in general, become 0 after an event. Partial likelihood is defined exactly as in (5.5), and Breslow's formula still defines an estimate of cumulative intensity (rather than cumulative hazard).

We can fit the model in `R` by using the `coxph` command. All we need to do is to represent the data appropriately in a `Surv` object. To do this, the record for an individual gets duplicated, with one row for each event time or censoring time. An event time will be the "stop" time in one row, and will then be the "start" time in the next row. The covariates will repeat from row to row for the same individual.

The model assumes that differences between individuals are completely described by the relative-risk function determined by their covariates. If we are unsure — as we generally will be — we can robustify the variance estimates as in section 9.3.2 by adding a `+cluster(id)` term. Alternatively, we can add a hidden frailty term to the model, as described below in section 9.5.

9.5 Shared frailty model

One way to deal with the correlation among multiple events for the same individual (or for linked individuals) is by explicitly modelling the variation in hazard rate with a random effect, generally called a *frailty* term in the survival context. The most common version is a relative risk term $e^{\omega_{\text{group}}}$, where ω_{group} is an unobserved covariate with distribution assumed to have a particular form, usually either gamma or Gaussian.

9.5.1 Negative-binomial model

The simplest version of the frailty model generalises the Poisson model: Individuals accrue events at a constant rate, but with the unknown constant dependent on the individual. For example, the Poisson model doesn't really make much sense in the example discussed in section 9.4.2. Individuals may be presumed to have differing predispositions to offend. Thus, it is not surprising that the number of offences is more spread out than you would expect under the Poisson model, which posits that everyone offended at the same rate.

We may generalise the Poisson regression model to better fit overdispersed data by adding a frailty term. That is, in place of (9.3) we represent the individual intensity by

$$\log \mu_i = \log \lambda_i + \log T_i + \beta \cdot \mathbf{x}_i. \tag{9.4}$$

The term λ_i, called a *multiplicative frailty*, represents the individual relative rate of producing events. The λ_i are treated as random effects, meaning that they are not to be estimated individually — which would not make sense — but rather, they are taken to be i.i.d. samples from a simple parametric distribution. When the frailty λ has a gamma distribution (with parameters (θ, θ), because we conventionally take the frailty distribution to have mean 1), and N is a Poisson count conditioned on λ with mean $\lambda\alpha$, then N has probability mass function

$$\mathbb{P}\{N = n\} = \frac{\Gamma(n+\theta)}{n!\Gamma(\theta)} \left(\frac{\theta}{\theta+\alpha}\right)^{\theta-1} \left(\frac{\alpha}{\theta+\alpha}\right)^{n},$$

which is the negative binomial distribution with parameters θ and $\alpha/(\theta+\alpha)$. (The calculation is left as an exercise.) Therefore this is called the *negative binomial regression model*. We can fit it with the `glm.nb` command in `R`. If we apply it to the same data as before, we get the following output:

```
> summary(poisreg2)

Call:
glm.nb(formula = treat.count ~ pris + pgroup + offset(log(risktime)),
init.theta = 0.8418678047, link = log)

Deviance Residuals:
```

```
Min       1Q   Median       3Q      Max
-2.1179  -1.2228  -0.4695   0.2785   3.6767

Coefficients:
Estimate Std. Error z value Pr(>|z|)
(Intercept) -2.28974    0.17977 -12.737  < 2e-16 ***
prisB       -1.08624    0.19047  -5.703 1.18e-08 ***
prisC       -2.08056    0.18716 -11.117  < 2e-16 ***
pgroup      -0.15331    0.09984  -1.536    0.125
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for Negative Binomial(0.8419) family taken to be 1)

Null deviance: 956.27  on 770  degrees of freedom
Residual deviance: 755.61  on 767  degrees of freedom
AIC: 2645.4

Number of Fisher Scoring iterations: 1


Theta:  0.8419
Std. Err.:  0.0781

2 x log-likelihood:  -2635.3550
```

We note that, while the largest deviance residual of 3.68 suggests a possible outlier, the total residual deviance is now quite plausible.

9.5.2 Frailty in proportional hazards models

We can use shared frailty to account for correlated times in proportional hazards regression, whether these are unordered (clustered) times, or recurrent events. The model fitting functions numerically exactly like any other random-effects model: We treat the individual unknown frailties as unobserved data, whose expected values given the observed data may be calculated. Given the individual frailties, we may maximise the parameters, and so loop through the EM algorithm. The calculations are carried through automatically by the `coxph` function in `R`, as long as we add a `+ frailty(id)` (or whichever variable we are grouping by) term to the formula. The output will include a p-value estimate for the individual

Is the frailty term actually appropriate to the data? We may test the null hypothesis that there is no individual frailty with a likelihood ratio test. The null-hypothesis log-likelihood is simply the log partial likelihood for a traditional model without a frailty term. The alternative log likelihood is the log of the integrated partial likelihood — that is, integrated over the distribution of the frailty — called the `I-likelihood` in the `R` output.

Note that the model fit automatically produces estimates of the individual frailties. If desired, these may be used for individualised survival projections.

9.6 Example: The diabetic retinopathy data set

We analyse the data set consisting of 197 pairs of eyes, where one of each has been treated, for the proportional effect of treatment on the hazard rate of vision loss. We first create a new data set by doubling the original, creating one entry for each eye in the original data set.

```
Diabetes <- data.frame(id = rep(ID,2), time = c(TIME1,TIME2),
status = c(STATUS1,STATUS2),
adult=(rep(ADULT,2)==2), trt = c(TRT_EYE==1,TRT_EYE==2))
```

Now we fit the Cox model:

```
coxph(formula = Surv(time, status) ~ trt + adult + cluster(id),
data = Diabetes)

n= 394, number of events= 155

             coef  exp(coef) se(coef) robust se      z    Pr(>|z|)
trtTRUE    -0.28091   0.75509  0.16167   0.14527 -1.934   0.0531
adultTRUE   0.02401   1.02431  0.16196   0.17399  0.138   0.8902

trtTRUE    .
adultTRUE
---


exp(coef) exp(-coef) lower .95 upper .95
trtTRUE       0.7551     1.3243    0.5680     1.004
adultTRUE     1.0243     0.9763    0.7283     1.441

Concordance= 0.534  (se = 0.023 )
Rsquare= 0.008   (max possible= 0.988 )
Likelihood ratio test= 3.06  on 2 df,   p=0.2
Wald test            = 3.74  on 2 df,   p=0.2
Score (logrank) test = 3.06  on 2 df,   p=0.2,   Robust = 3.72  p=0.2

(Note: the likelihood ratio and score tests assume independence of
observations within a cluster, the Wald and robust score tests do not).
```

Note that in this case, because of the paired design, the robust SE is actually smaller than the model-based (inverse-information) SE. If we had stratified instead of clustering we would obtain:

```
coxph(formula = Surv(time, status) ~ trt + adult + strata(id),
data = Diabetes)

coef exp(coef) se(coef)     z     p
trtTRUE    -0.2122    0.8088   0.1813 -1.17 0.242
adultTRUE      NA        NA   0.0000    NA    NA

Likelihood ratio test=1.38  on 1 df, p=0.2407
n= 394, number of events= 155
```

Note that the standard error is increased substantially relative to the model-based estimate.

If we instead fit a gamma-frailty model to account for the correlation between two eyes we get a very similar result to that obtained from the clustered model:

```
coxph(formula = Surv(time, status) ~ trt + adult + frailty(id),
data = Diabetes)

```

```
  coef se(coef)      se2     Chisq    DF      p
trtTRUE      -0.31173  0.16501  0.16265  3.56886  1.0  0.059
adultTRUE     0.00239  0.20242  0.16341  0.00014  1.0  0.991
frailty(id)                             81.76565 65.1  0.079

Iterations: 7 outer, 31 Newton-Raphson
Variance of random effect= 0.587   I-likelihood = -863
Degrees of freedom for terms=  1.0  0.7 65.1
Likelihood ratio test=138  on 66.7 df, p=7e-07
n= 394, number of events= 155
```

Fitting a Gaussian frailty produces only a slight change:

```
coxph(formula = Surv(time, status) ~ trt + adult + frailty(id,
distribution = "gaussian"), data = Diabetes)

                              coef se(coef)    se2     Chisq      DF     p
trtTRUE                    -0.30731  0.16510  0.16268  3.46479   1.0 0.063
adultTRUE                   0.01567  0.19626  0.16241  0.00637   1.0 0.936
frailty(id, distribution                                78.53744 57.2 0.032

Iterations: 6 outer, 22 Newton-Raphson
Variance of random effect= 0.547
Degrees of freedom for terms=  1.0  0.7 57.2
Likelihood ratio test=136  on 58.9 df, p=5e-08
n= 394, number of events= 155
```

9.7 Example: Bladder cancer data set

We follow the discussion in section 8.5.4 of [24] of a famous data set on recurrence of bladder cancer. The data describe the time until up to four recurrences for 85 patients The version we use is included in the `survival` package as the object `bladder2`. We wish to apply a Cox regression analysis to evaluate the effect of the variables `rx` (treatment: 1=placebo, 2=active), `size` (size in cm of largest tumour), and `number` (number of tumours), on the time (in months) to recurrence.

The recurrences are numbered successively by the `enum` variable, so if we include only those with `enum==1` we will have a time-to-first-event analysis:

```
coxph(formula = Surv(start, stop, event) ~ rx + number + size,
data = bladder2, subset = (enum == 1))

  coef exp(coef) se(coef)      z       p
rx      -0.52598   0.59097  0.31583 -1.665  0.0958
number   0.23818   1.26894  0.07588  3.139  0.0017
size     0.06961   1.07209  0.10156  0.685  0.4931

Likelihood ratio test=9.92  on 3 df, p=0.01927
n= 85, number of events= 47
```

Including all the events in an Anderson–Gill model increases the number of events from 47 to 112. Naïvely we might expect the standard errors to be reduced by a factor of about

$\sqrt{47/112} = .648$, reducing the SE of the `number` coefficient from .076 to about .049; and the SE of the `rx` coefficient from .316 to about .205. Carrying out the calculation yields

```
coxph(formula = Surv(start, stop, event) ~ rx + number + size +
cluster(id), data = bladder2)

  coef exp(coef) se(coef) robust se      z       p
rx      -0.46469   0.62833  0.19973   0.26556 -1.750 0.08015
number   0.17496   1.19120  0.04707   0.06304  2.775 0.00551
size    -0.04366   0.95728  0.06905   0.07762 -0.563 0.57376

Likelihood ratio test=17.52  on 3 df, p=0.0005531
n= 178, number of events= 112
```

The `se(coef)` output is exactly what we predicted, but the `robust se` is substantially larger, due to correlation among the observations. The actual reduction in SE is not the 35% that would have been produced if the additional recurrences had been independent, but only about 17%. It is as though we had only about 15 additional independent observations, rather than the 65 that we might have naïvely supposed.

Applying a gamma frailty again yields a very similar result:

```
coxph(formula = Surv(start, stop, event) ~ rx + number + size +
frailty(id), data = bladder2)

coef se(coef)      se2    Chisq    DF       p
rx          -0.6077   0.3330  0.2197   3.3295   1.0 0.06805
number       0.2387   0.0932  0.0570   6.5567   1.0 0.01045
size        -0.0215   0.1140  0.0717   0.0357   1.0 0.85009
frailty(id)                          82.9157 45.4 0.00056

Iterations: 6 outer, 29 Newton-Raphson
Variance of random effect= 1.08   I-likelihood = -436.8
Degrees of freedom for terms=  0.4  0.4  0.4 45.4
Likelihood ratio test=144  on 46.6 df, p=8e-12
n= 178, number of events= 112
```

And with Gaussian frailty:

```
  coxph(formula = Surv(start, stop, event) ~ rx + number + size +
  frailty(id, distribution = "gaussian"), data = bladder2)

  coef se(coef)      se2    Chisq    DF       p
  rx                        -0.5612   0.3161  0.2175   3.1517   1.0   0.0758
  number                     0.2259   0.0835  0.0534   7.3248   1.0   0.0068
  size                      -0.0202   0.1065  0.0711   0.0358   1.0   0.8499
  frailty(id, distribution                             87.8815 37.8 7.2e-06

  Iterations: 7 outer, 27 Newton-Raphson
  Variance of random effect= 0.89
  Degrees of freedom for terms=  0.5  0.4  0.4 37.8
  Likelihood ratio test=138  on 39.1 df, p=6e-13
  n= 178, number of events= 112
```