Chapter 19
Case Study in Parametric Survival Modeling and Model Approximation

Consider the random sample of 1000 patients from the SUPPORT study,[352] described in Section 3.12. In this case study we develop a parametric survival time model (accelerated failure time model) for time until death for the acute disease subset of SUPPORT (acute respiratory failure, multiple organ system failure, coma). We eliminate the chronic disease categories because the shapes of the survival curves are different between acute and chronic disease categories. To fit both acute and chronic disease classes would require a log-normal model with σ parameter that is disease-specific.

Patients had to survive until day 3 of the study to qualify. The baseline physiologic variables were measured during day 3.

19.1 Descriptive Statistics

First we create a variable `acute` to flag the categories of interest, and print univariable descriptive statistics for the data subset.

```
require(rms)
```

```
getHdata(support)      # Get data frame from web site
acute ← support$dzclass %in% c('ARF/MOSF','Coma')
latex(describe(support[acute,]), file='')
```


F.E. Harrell, Jr., *Regression Modeling Strategies*, Springer Series in Statistics, DOI 10.1007/978-3-319-19425-7_19

support[acute,]
35 Variables 537 Observations

---

age : Age 

n	missing	unique	Info	Mean	.05	.10	.25	.50	.75	.90	.95
537	0	529	1	60.7	28.49	35.22	47.93	63.67	74.49	81.54	85.56

```
lowest : 18.04 18.41 19.76 20.30 20.31
highest: 91.62 91.82 91.93 92.74 95.51
```

---

death : Death at any time up to NDI date:31DEC94

n	missing	unique	Info	Sum	Mean
537	0	2	0.67	356	0.6629

---

sex

n	missing	unique
537	0	2

```
female (251, 47%), male (286, 53%)
```

---

hospdead : Death in Hospital

n	missing	unique	Info	Sum	Mean
537	0	2	0.7	201	0.3743

---

slos : Days from Study Entry to Discharge 

n	missing	unique	Info	Mean	.05	.10	.25	.50	.75	.90	.95
537	0	85	1	23.44	4.0	5.0	9.0	15.0	27.0	47.4	68.2

```
lowest :   3   4   5   6   7, highest: 145 164 202 236 241
```

---

d.time : Days of Follow-Up 

n	missing	unique	Info	Mean	.05	.10	.25	.50	.75	.90	.95
537	0	340	1	446.1	4	6	16	182	724	1421	1742

```
lowest :    3    4    5    6    7, highest: 1977 1979 1982 2011 2022
```

---

dzgroup

n	missing	unique
537	0	3

```
ARF/MOSF w/Sepsis (391, 73%), Coma (60, 11%), MOSF w/Malig (86, 16%)
```

---

dzclass

n	missing	unique
537	0	2

```
ARF/MOSF (477, 89%), Coma (60, 11%)
```

---

num.co : number of comorbidities

n	missing	unique	Info	Mean
537	0	7	0.93	1.525

	0	1	2	3	4	5	6
Frequency	111	196	133	51	31	10	5
%	21	36	25	9	6	2	1

---

edu : Years of Education

n	missing	unique	Info	Mean	.05	.10	.25	.50	.75	.90	.95
411	126	22	0.96	12.03	7	8	10	12	14	16	17

```
lowest :  0  1  2  3  4, highest: 17 18 19 20 22
```

---

income

n	missing	unique
335	202	4

```
under $11k (158, 47%), $11-$25k (79, 24%), $25-$50k (63, 19%)
>$50k (35, 10%)
```

---

scoma : SUPPORT Coma Score based on Glasgow D3

n	missing	unique	Info	Mean	.05	.10	.25	.50	.75	.90	.95
537	0	11	0.82	19.24	0	0	0	0	37	55	100

	0	9	26	37	41	44	55	61	89	94	100
Frequency	301	50	44	19	17	43	11	6	8	6	32
%	56	9	8	4	3	8	2	1	1	1	6

---

charges : Hospital Charges

n	missing	unique	Info	Mean	.05	.10	.25	.50	.75	.90	.95
517	20	516	1	86652	11075	15180	27389	51079	100904	205562	283411

```
lowest :   3448   4432   4574   5555   5849
highest: 504660 538323 543761 706577 740010
```

---

totcst : Total RCC cost

n	missing	unique	Info	Mean	.05	.10	.25	.50	.75	.90	.95
471	66	471	1	46360	6359	8449	15412	29308	57028	108927	141569

```
lowest :      0   2071   2522   3191   3325
highest: 269057 269131 338955 357919 390460
```

---

totmcst : Total micro-cost

n	missing	unique	Info	Mean	.05	.10	.25	.50	.75	.90	.95
331	206	328	1	39022	6131	8283	14415	26323	54102	87495	111920

```
lowest :      0   1562   2478   2626   3421
highest: 144234 154709 198047 234876 271467
```

---

avtisst : Average TISS, Days 3–25

n	missing	unique	Info	Mean	.05	.10	.25	.50	.75	.90	.95
536	1	205	1	29.83	12.46	14.50	19.62	28.00	39.00	47.17	50.37

```
lowest :  4.000  5.667  8.000  9.000  9.500
highest: 58.500 59.000 60.000 61.000 64.000
```

---

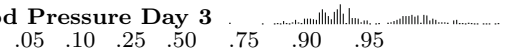
race

n	missing	unique
535	2	5

	white	black	asian	other	hispanic
Frequency	417	84	4	8	22
%	78	16	1	1	4

---

meanbp : Mean Arterial Blood Pressure Day 3



n	missing	unique	Info	Mean	.05	.10	.25	.50	.75	.90	.95
537	0	109	1	83.28	41.8	49.0	59.0	73.0	111.0	124.4	135.0

```
lowest :   0  20  27  30  32, highest: 155 158 161 162 180
```

---

wblc : White Blood Cell Count Day 3

n	missing	unique	Info	Mean	.05	.10	.25	.50	.75	.90	.95
532	5	241	1	14.1	0.8999	4.5000	7.9749	12.3984	18.1992	25.1891	30.1873

```
lowest :   0.05000   0.06999   0.09999   0.14999   0.19998
highest:  51.39844  58.19531  61.19531  79.39062 100.00000
```

---

hrt : Heart Rate Day 3

n	missing	unique	Info	Mean	.05	.10	.25	.50	.75	.90	.95
537	0	111	1	105	51	60	75	111	126	140	155

```
lowest :   0  11  30  36  40, highest: 189 193 199 232 300
```

---

resp : Respiration Rate Day 3

n	missing	unique	Info	Mean	.05	.10	.25	.50	.75	.90	.95
537	0	45	1	23.72	8	10	12	24	32	39	40

```
lowest :  0  4  6  7  8, highest: 48 49 52 60 64
```

---

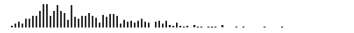
temp : Temperature (celcius) Day 3

n	missing	unique	Info	Mean	.05	.10	.25	.50	.75	.90	.95
537	0	61	1	37.52	35.50	35.80	36.40	37.80	38.50	39.09	39.50

```
lowest : 32.50 34.00 34.09 34.90 35.00
highest: 40.20 40.59 40.90 41.00 41.20
```

---

pafi : PaO2/(.01*FiO2) Day 3



n	missing	unique	Info	Mean	.05	.10	.25	.50	.75	.90	.95
500	37	357	1	227.2	86.99	105.08	137.88	202.56	290.00	390.49	433.31

```
lowest :  45.00  48.00  53.33  54.00  55.00
highest: 574.00 595.12 640.00 680.00 869.38
```

---

alb : Serum Albumin Day 3

n	missing	unique	Info	Mean	.05	.10	.25	.50	.75	.90	.95
346	191	34	1	2.668	1.700	1.900	2.225	2.600	3.100	3.400	3.800

```
lowest : 1.100 1.200 1.300 1.400 1.500
highest: 4.100 4.199 4.500 4.699 4.800
```

---

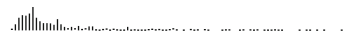
bili : Bilirubin Day 3

n	missing	unique	Info	Mean	.05	.10	.25	.50	.75	.90	.95
386	151	88	1	2.678	0.3000	0.4000	0.6000	0.8999	2.0000	6.5996	13.1743

```
lowest :  0.09999  0.19998  0.29999  0.39996  0.50000
highest: 22.59766 30.00000 31.50000 35.00000 39.29688
```

---

crea : Serum creatinine Day 3



n	missing	unique	Info	Mean	.05	.10	.25	.50	.75	.90	.95
537	0	84	1	2.232	0.6000	0.7000	0.8999	1.3999	2.5996	5.2395	7.3197

```
lowest :  0.3  0.4  0.5  0.6  0.7, highest: 10.4 10.6 11.2 11.6 11.8
```

---

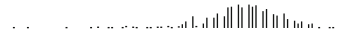
sod : Serum sodium Day 3

n	missing	unique	Info	Mean	.05	.10	.25	.50	.75	.90	.95
537	0	38	1	138.1	129	131	134	137	142	147	150

```
lowest : 118 120 121 126 127, highest: 156 157 158 168 175
```

---

ph : Serum pH (arterial) Day 3

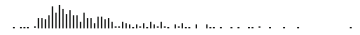


n	missing	unique	Info	Mean	.05	.10	.25	.50	.75	.90	.95
500	37	49	1	7.416	7.270	7.319	7.380	7.420	7.470	7.510	7.529

```
lowest : 6.960 6.989 7.069 7.119 7.130
highest: 7.560 7.569 7.590 7.600 7.659
```

---

glucose : Glucose Day 3

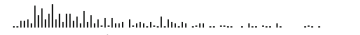


n	missing	unique	Info	Mean	.05	.10	.25	.50	.75	.90	.95
297	240	179	1	167.7	76.0	89.0	106.0	141.0	200.0	292.4	347.2

```
lowest :  30  42  52  55  68, highest: 446 468 492 576 598
```

---

bun : BUN Day 3

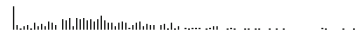


n	missing	unique	Info	Mean	.05	.10	.25	.50	.75	.90	.95
304	233	100	1	38.91	8.00	11.00	16.75	30.00	56.00	79.70	100.70

```
lowest :   1   3   4   5   6, highest: 123 124 125 128 146
```

---

urine : Urine Output Day 3



n	missing	unique	Info	Mean	.05	.10	.25	.50	.75	.90	.95
303	234	262	1	2095	20.3	364.0	1156.5	1870.0	2795.0	4008.6	4817.5

```
lowest :    0    5    8   15   20, highest: 6865 6920 7360 7560 7750
```

---

adlp : ADL Patient Day 3

n	missing	unique	Info	Mean
104	433	8	0.87	1.577

	0	1	2	3	4	5	6	7
Frequency	51	19	7	6	4	7	8	2
%	49	18	7	6	4	7	8	2

---

adls : ADL Surrogate Day 3

n	missing	unique	Info	Mean
392	145	8	0.89	1.86

	0	1	2	3	4	5	6	7
Frequency	185	68	22	18	17	20	39	23
%	47	17	6	5	4	5	10	6

---

sfdm2

n	missing	unique
468	69	5

```
no(M2 and SIP pres) (134, 29%), adl>=4 (>=5 if sur) (78, 17%)
SIP>=30 (30, 6%), Coma or Intub (5, 1%), <2 mo. follow-up (221, 47%)
```

```
adlsc : Imputed ADL Calibrated to Surrogate
       n missing unique Info Mean   .05   .10   .25   .50   .75   .90   .95
     537       0    144 0.96 2.119 0.000 0.000 0.000 1.839 3.375 6.000 6.000

  lowest : 0.0000 0.4948 0.4948 1.0000 1.1667
  highest: 5.7832 6.0000 6.3398 6.4658 7.0000
```

Next, patterns of missing data are displayed.

```
plot(naclus(support[acute,]))        # Figure 19.1
```

The `Hmisc` `varclus` function is used to quantify and depict associations between predictors, allowing for general nonmonotonic relationships. This is done by using Hoeffding's D as a similarity measure for all possible pairs of predictors instead of the default similarity, Spearman's ρ.

```
ac ← support[acute,]
ac$dzgroup ← ac$dzgroup[drop=TRUE]     # Remove unused levels
label(ac$dzgroup) ← 'Disease Group'
attach(ac)
vc ← varclus(∼ age + sex + dzgroup + num.co + edu + income +
               scoma + race + meanbp + wblc + hrt + resp +
               temp + pafi + alb + bili + crea + sod + ph +
               glucose + bun + urine + adlsc, sim='hoeffding')
plot(vc)                              # Figure 19.2
```

19.2 Checking Adequacy of Log-Normal Accelerated Failure Time Model

Let us check whether a parametric survival time model will fit the data, with respect to the key prognostic factors. First, Kaplan–Meier estimates stratified by disease group are computed, and plotted after inverse normal transformation, against $\log t$. Parallelism and linearity indicate goodness of fit to the log normal distribution for disease group. Then a more stringent assessment is made by fitting an initial model and computing right-censored residuals. These residuals, after dividing by $\hat{\sigma}$, should all have a normal distribution if the model holds. We compute Kaplan–Meier estimates of the distribution of the residuals and overlay the estimated survival distribution with the theoretical Gaussian one. This is done overall, and then to get more stringent assessments of fit, residuals are stratified by key predictors and plots are produced that contain multiple Kaplan–Meier curves along with a single theoretical normal curve. All curves should hover about the normal distribution. To gauge the natural variability of stratified residual distribution estimates, the residuals are also stratified by a random number that has no bearing on the goodness of fit.

```
dd ← datadist(ac)
# describe distributions of variables to rms
```

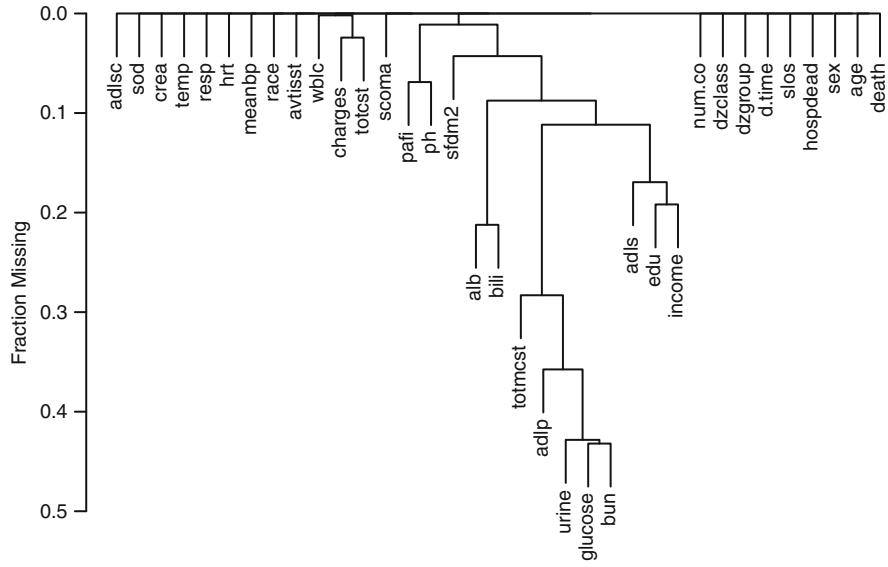


Fig. 19.1 Cluster analysis showing which predictors tend to be missing on the same patients

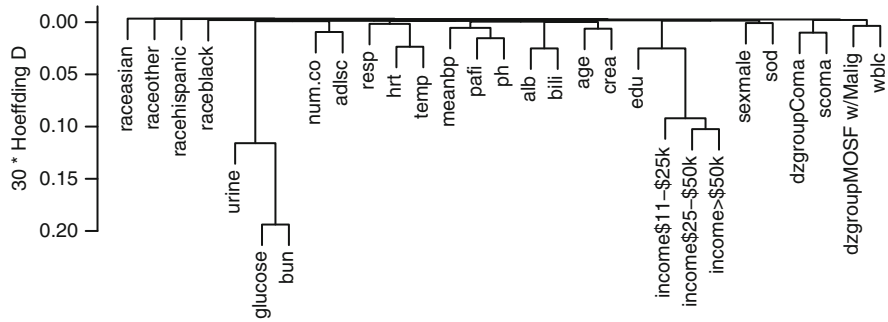


Fig. 19.2 Hierarchical clustering of potential predictors using Hoeffding D as a similarity measure. Categorical predictors are automatically expanded into dummy variables.

```
options(datadist='dd')

# Generate right-censored survival time variable
years ← d.time/365.25
units(years) ← 'Year'
S ← Surv(years, death)

# Show normal inverse Kaplan-Meier estimates
# stratified by dzgroup
survplot(npsurv(S ~ dzgroup), conf='none',
         fun=qnorm,logt=TRUE)    # Figure 19.3
```

```
f ← psm(S ~ dzgroup + rcs(age,5) + rcs(meanbp,5),
        dist='lognormal', y=TRUE)
r ← resid(f)

survplot(r, dzgroup, label.curve=FALSE)
survplot(r, age,     label.curve=FALSE)
survplot(r, meanbp,  label.curve=FALSE)
random ← runif(length(age)); label(random) ← 'Random Number'
survplot(r, random, label.curve=FALSE)  # Fig. 19.4
```

Now remove from consideration predictors that are missing in more than 0.2 of patients. Many of these were collected only for the second half of SUPPORT. Of those variables to be included in the model, find which ones have enough potential predictive power to justify allowing for nonlinear relationships or multiple categories, which spend more d.f. For each variable compute Spearman ρ^2 based on multiple linear regression of rank(x), rank$(x)^2$, and the

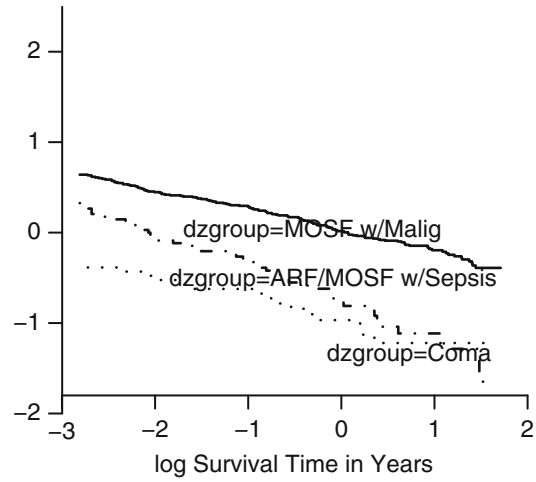


Fig. 19.3 $\Phi^{-1}(S_{KM}(t))$ stratified by `dzgroup`. Linearity and semi-parallelism indicate a reasonable fit to the log-normal accelerated failure time model with respect to one predictor.

survival time, truncating survival time at the shortest follow-up for survivors (356 days; see Section 4.1).

```
shortest.follow.up ← min(d.time[death==0], na.rm=TRUE)
d.timet ← pmin(d.time, shortest.follow.up)

w ← spearman2(d.timet ~ age + num.co + scoma + meanbp +
              hrt + resp + temp + crea + sod + adlsc +
              wblc + pafi + ph + dzgroup + race, p=2)
plot(w, main='')        # Figure 19.5
```

A better approach is to use the complete information in the failure and censoring times by computing Somers' D_{xy} rank correlation allowing for censoring.

```
w ← rcorrcens(S ~ age + num.co + scoma + meanbp + hrt + resp +
                temp + crea + sod + adlsc + wblc + pafi + ph +
                dzgroup + race)
plot(w, main='')            # Figure 19.6
```

Remaining missing values are imputed using the "most normal" values, a procedure found to work adequately for this particular study. Race is imputed using the modal category.

```
# Compute number of missing values per variable
sapply(llist(age,num.co,scoma,meanbp,hrt,resp,temp,crea,sod,
             adlsc,wblc,pafi,ph), function(x) sum(is.na(x)))
```

```
   age num.co  scoma meanbp    hrt   resp   temp   crea    sod  adlsc
     0      0      0      0      0      0      0      0      0      0
  wblc   pafi     ph
     5     37     37
```

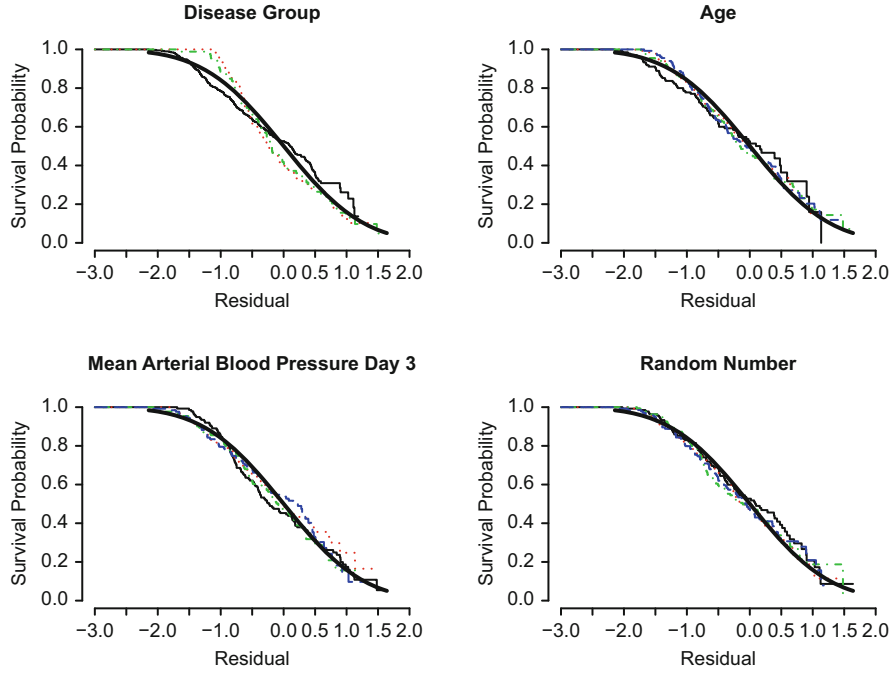


Fig. 19.4 Kaplan-Meier estimates of distributions of normalized, right-censored residuals from the fitted log-normal survival model. Residuals are stratified by important variables in the model (by quartiles of continuous variables), plus a random variable to depict the natural variability (in the lower right plot). Theoretical standard Gaussian distributions of residuals are shown with a thick solid line.

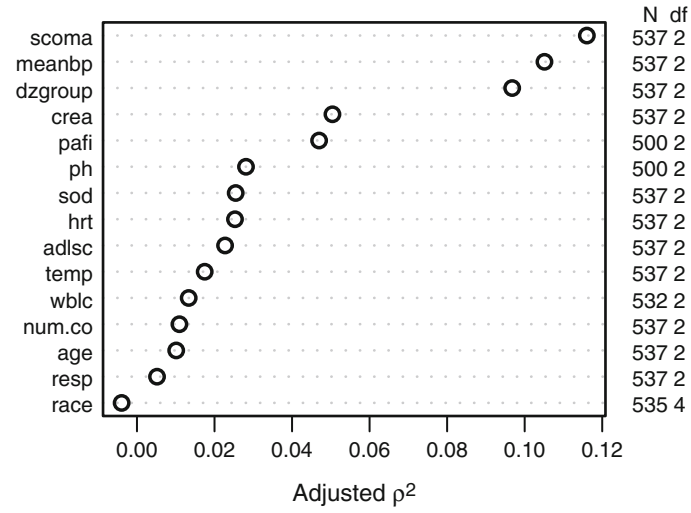


Fig. 19.5 Generalized Spearman ρ^2 rank correlation between predictors and truncated survival time

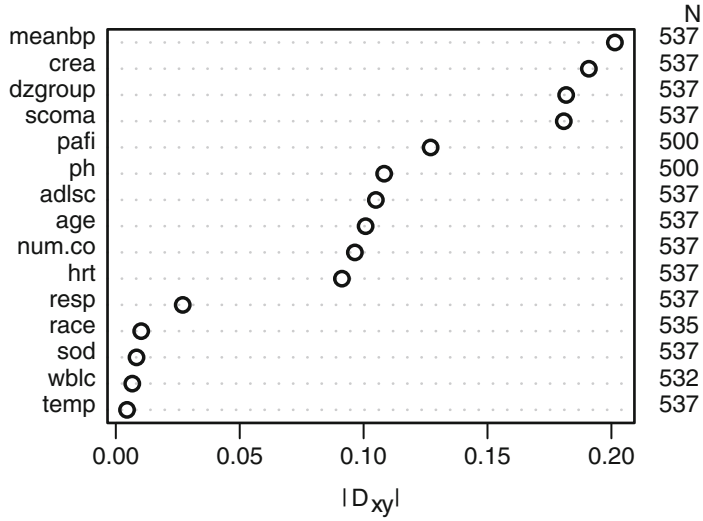


Fig. 19.6 Somers' D_{xy} rank correlation between predictors and original survival time. For `dzgroup` or `race`, the correlation coefficient is the maximum correlation from using a dummy variable to represent the most frequent or one to represent the second most frequent category.',scap='Somers' D_{xy} rank correlation between predictors and original survival time

```
# Can also do naplot(naclus(support[acute,]))
# Can also use the Hmisc naclus and naplot functions
# Impute missing values with normal or modal values
wblc.i ← impute(wblc, 9)
pafi.i ← impute(pafi, 333.3)
ph.i   ← impute(ph,   7.4)
race2  ← race
levels(race2) ← list(white='white',other=levels(race)[-1])
race2[is.na(race2)] ← 'white'
dd ← datadist(dd, wblc.i, pafi.i, ph.i, race2)
```

Now that missing values have been imputed, a formal multivariable redundancy analysis can be undertaken. The `Hmisc` package's `redun` function goes farther than the `varclus` pairwise correlation approach and allows for non-monotonic transformations in predicting each predictor from all the others.

```
redun(~ crea + age + sex + dzgroup + num.co + scoma + adlsc +
      race2 + meanbp + hrt + resp + temp + sod + wblc.i +
      pafi.i + ph.i, nk=4)
```

```
Redundancy Analysis

redun(formula = ~crea + age + sex + dzgroup + num.co + scoma +
    adlsc + race2 + meanbp + hrt + resp + temp + sod + wblc.i +
    pafi.i + ph.i, nk = 4)

n: 537  p: 16   nk: 4

Number of NAs:   0

Transformation of target variables forced to be linear

R^2 cutoff: 0.9  Type: ordinary

R^2 with which each variable can be predicted from all other variables:

   crea      age      sex dzgroup  num.co   scoma   adlsc   race2  meanbp
  0.133    0.246    0.132   0.451   0.147   0.418   0.153   0.151   0.178
    hrt     resp     temp     sod  wblc.i  pafi.i    ph.i
  0.258    0.131    0.197   0.135   0.093   0.143   0.171

No redundant variables
```

Now turn to a more efficient approach for gauging the potential of each predictor, one that makes maximal use of failure time and censored data is to all continuous variables to have a maximum number of knots in a log-normal survival model. This approach must use imputation to have an adequate sample size. A semi-saturated main effects additive log-normal model is fitted. It is necessary to limit restricted cubic splines to 4 knots, force `scoma` to be linear, and to omit `ph.i` in order to avoid a singular covariance matrix in the fit.

```
k ← 4
f ← psm(S ~ rcs(age,k)+sex+dzgroup+pol(num.co,2)+scoma+
          pol(adlsc,2)+race+rcs(meanbp,k)+rcs(hrt,k)+
```

```
          rcs(resp,k)+rcs(temp,k)+rcs(crea,3)+rcs(sod,k)+
          rcs(wblc.i,k)+rcs(pafi.i,k), dist='lognormal')
plot(anova(f))    # Figure 19.7
```

Figure 19.7 properly blinds the analyst to the form of effects (tests of linearity). Next fit a log-normal survival model with number of parameters corresponding to nonlinear effects determined from the partial χ^2 tests in Figure 19.7. For the most promising predictors, five knots can be allocated, as there are fewer singularity problems once less promising predictors are simplified.

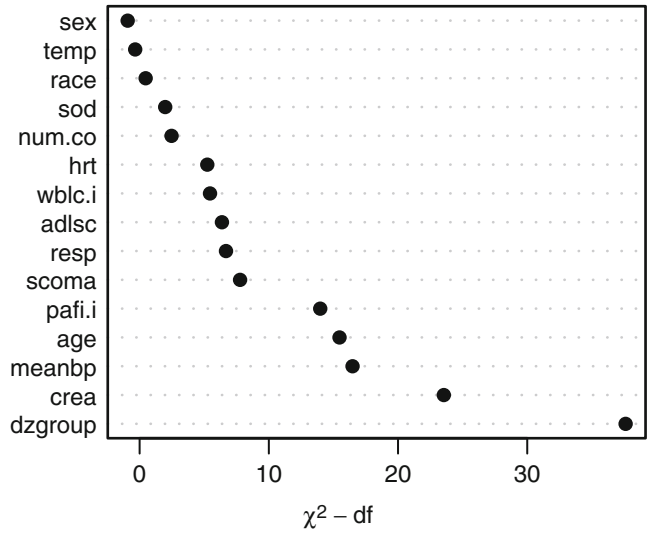


Fig. 19.7 Partial χ^2 statistics for association of each predictor with response from saturated main effects model, penalized for d.f.

```
f ← psm(S ~ rcs(age,5)+sex+dzgroup+num.co+
            scoma+pol(adlsc,2)+race2+rcs(meanbp,5)+
            rcs(hrt,3)+rcs(resp,3)+temp+
            rcs(crea,4)+sod+rcs(wblc.i,3)+rcs(pafi.i,4),
        dist='lognormal')
print(f, latex=TRUE, coefs=FALSE)
```

Parametric Survival Model: Log Normal Distribution

```
psm(formula = S ~ rcs(age, 5) + sex + dzgroup + num.co + scoma +
    pol(adlsc, 2) + race2 + rcs(meanbp, 5) + rcs(hrt, 3) + rcs(resp,
    3) + temp + rcs(crea, 4) + sod + rcs(wblc.i, 3) + rcs(pafi.i,
    4), dist = "lognormal")
```

	Model Likelihood Ratio Test		Discrimination Indexes	
Obs	537	LR χ^2	236.83	R^2 0.594
Events	356	d.f.	30	D_{xy} 0.485
σ	2.230782	Pr($> \chi^2$)	< 0.0001	g 0.033
				g_r 1.959

```
a ← anova(f)
```

Table 19.1 Wald Statistics for `S`

	χ^2	d.f.	P
age	15.99	4	0.0030
Nonlinear	0.23	3	0.9722
sex	0.11	1	0.7354
dzgroup	45.69	2	< 0.0001
num.co	4.99	1	0.0255
scoma	10.58	1	0.0011
adlsc	8.28	2	0.0159
Nonlinear	3.31	1	0.0691
race2	1.26	1	0.2624
meanbp	27.62	4	< 0.0001
Nonlinear	10.51	3	0.0147
hrt	11.83	2	0.0027
Nonlinear	1.04	1	0.3090
resp	11.10	2	0.0039
Nonlinear	8.56	1	0.0034
temp	0.39	1	0.5308
crea	33.63	3	< 0.0001
Nonlinear	21.27	2	< 0.0001
sod	0.08	1	0.7792
wblc.i	5.47	2	0.0649
Nonlinear	5.46	1	0.0195
pafi.i	15.37	3	0.0015
Nonlinear	6.97	2	0.0307
TOTAL NONLINEAR	60.48	14	< 0.0001
TOTAL	261.47	30	< 0.0001

19.3 Summarizing the Fitted Model

First let's plot the shape of the effect of each predictor on log survival time. All effects are centered so that they can be placed on a common scale. This allows the relative strength of various predictors to be judged. Then Wald χ^2 statistics, penalized for d.f., are plotted in descending order. Next, relative effects of varying predictors over reasonable ranges (survival time ratios varying continuous predictors from the first to the third quartile) are charted.

```
ggplot(Predict(f, ref.zero=TRUE), vnames='names',
       sepdiscrete='vertical', anova=a)  # Figure 19.8
```

```
latex(a, file='',label='tab:support-anovat') # Table 19.1
```

```
plot(a)              # Figure 19.9
```

```
options(digits=3)
plot(summary(f), log=TRUE, main='')    # Figure 19.10
```

19.4 Internal Validation of the Fitted Model Using the Bootstrap

Let us decide whether there was significant overfitting during the development of this model, using the bootstrap.

```
# First add data to model fit so bootstrap can re-sample
#  from the data
g ← update(f, x=TRUE, y=TRUE)
set.seed(717)
latex(validate(g, B=300), digits=2, size='Ssize')
```

Index	Original Sample	Training Sample	Test Sample	Optimism	Corrected Index	n
D_{xy}	0.49	0.51	0.46	0.05	0.43	300
R^2	0.59	0.66	0.54	0.12	0.47	300
Intercept	0.00	0.00	−0.05	0.05	−0.05	300
Slope	1.00	1.00	0.90	0.10	0.90	300
D	0.48	0.55	0.42	0.13	0.35	300
U	0.00	0.00	−0.01	0.01	−0.01	300
Q	0.48	0.56	0.43	0.12	0.36	300
g	1.96	2.05	1.87	0.19	1.77	300

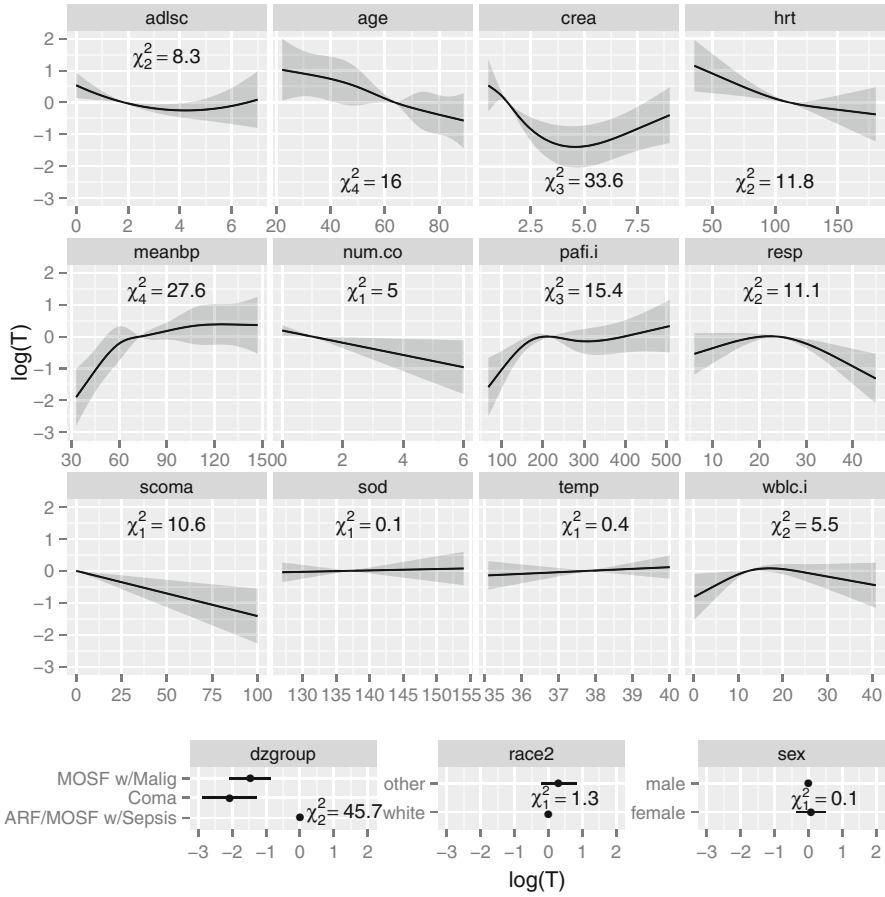


Fig. 19.8 Effect of each predictor on log survival time. Predicted values have been centered so that predictions at predictor reference values are zero. Pointwise 0.95 confidence bands are also shown. As all y-axes have the same scale, it is easy to see which predictors are strongest.

Judging from D_{xy} and R^2 there is a moderate amount of overfitting. The slope shrinkage factor (0.9) is not troublesome, however. An almost unbiased estimate of future predictive discrimination on similar patients is given by the corrected D_{xy} of 0.43. This index equals the difference between the probability of concordance and the probability of discordance of pairs of predicted survival times and pairs of observed survival times, accounting for censoring.

Next, a bootstrap overfitting-corrected calibration curve is estimated. Patients are stratified by the predicted probability of surviving one year, such that there are at least 60 patients in each group.

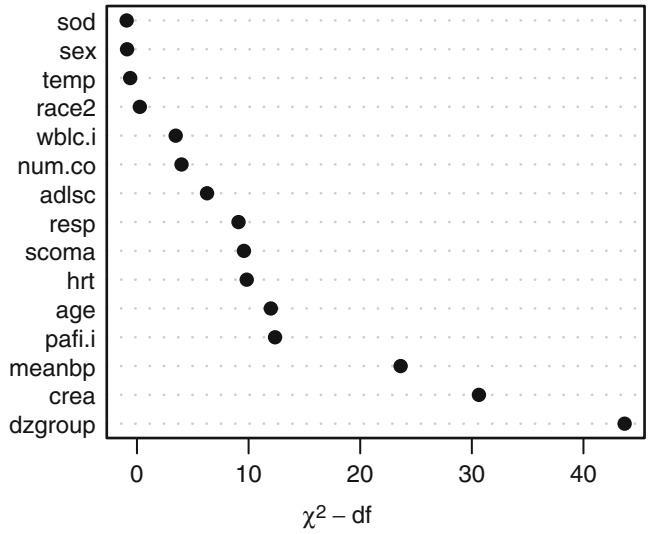


Fig. 19.9 Contribution of variables in predicting survival time in log-normal model

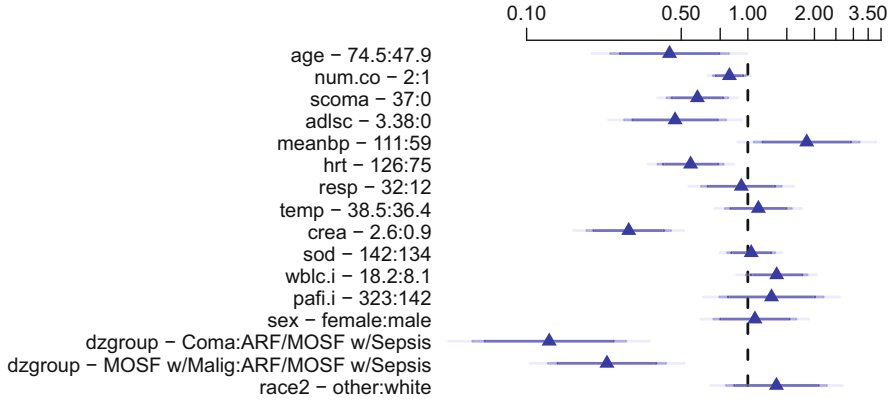


Fig. 19.10 Estimated survival time ratios for default settings of predictors. For example, when age changes from its lower quartile to the upper quartile (47.9y to 74.5y), median survival time decreases by more than half. Different shaded areas of bars indicate different confidence levels (.9, 0.95, 0.99).

```
set.seed(717)
cal ← calibrate(g, u=1, B=300)
plot(cal, subtitles=FALSE)
cal ← calibrate(g, cmethod='KM', u=1, m=60, B=120, pr=FALSE)
plot(cal, add=TRUE)      # Figure 19.11
```

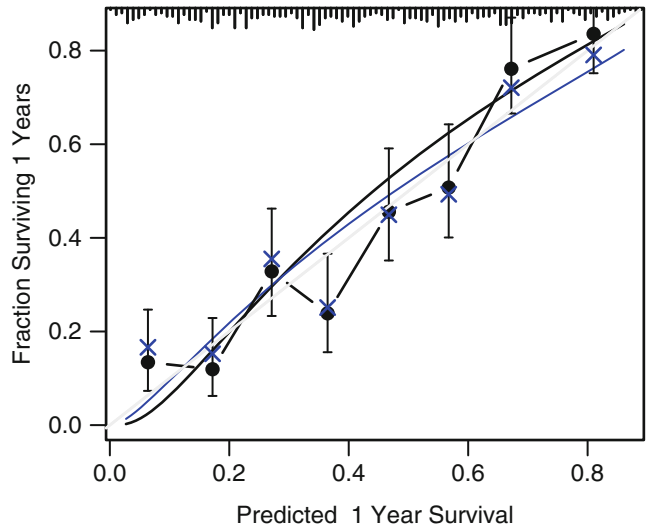


Fig. 19.11 Bootstrap validation of calibration curve. Dots represent apparent calibration accuracy; × are bootstrap estimates corrected for overfitting, based on binning predicted survival probabilities and computing Kaplan-Meier estimates. Black curve is the estimated observed relationship using `hare` and the blue curve is the overfitting-corrected `hare` estimate. The gray-scale line depicts the ideal relationship.

19.5 Approximating the Full Model

The fitted log-normal model is perhaps too complex for routine use and for routine data collection. Let us develop a simplified model that can predict the predicted values of the full model with high accuracy ($R^2 = 0.967$). The simplification is done using a fast backward step-down against the full model predicted values.

```
Z ← predict(f)     # X*beta hat
a ← ols(Z ~ rcs(age,5)+sex+dzgroup+num.co+
            scoma+pol(adlsc,2)+race2+
            rcs(meanbp,5)+rcs(hrt,3)+rcs(resp,3)+
            temp+rcs(crea,4)+sod+rcs(wblc.i,3)+
            rcs(pafi.i,4), sigma=1)
# sigma=1 is used to prevent sigma hat from being zero when
# R2=1.0 since we start out by approximating Z with all
#  component variables
fastbw(a, aics=10000)     # fast backward stepdown
```

```
Deleted Chi-Sq d.f. P      Residual d.f. P      AIC    R2
sod       0.43 1    0.512    0.43   1   0.5117   -1.57 1.000
sex       0.57 1    0.451    1.00   2   0.6073   -3.00 0.999
temp      2.20 1    0.138    3.20   3   0.3621   -2.80 0.998
race2     6.81 1    0.009   10.01   4   0.0402    2.01 0.994
wblc.i   29.52 2    0.000   39.53   6   0.0000   27.53 0.976
```

```
 num.co    30.84 1    0.000   70.36   7   0.0000   56.36 0.957
 resp      54.18 2    0.000  124.55   9   0.0000  106.55 0.924
 adlsc     52.46 2    0.000  177.00  11   0.0000  155.00 0.892
 pafi.i    66.78 3    0.000  243.79  14   0.0000  215.79 0.851
 scoma     78.07 1    0.000  321.86  15   0.0000  291.86 0.803
 hrt       83.17 2    0.000  405.02  17   0.0000  371.02 0.752
 age       68.08 4    0.000  473.10  21   0.0000  431.10 0.710
 crea     314.47 3    0.000  787.57  24   0.0000  739.57 0.517
 meanbp   403.04 4    0.000 1190.61  28   0.0000 1134.61 0.270
 dzgroup 441.28 2    0.000 1631.89  30   0.0000 1571.89 0.000

Approximate Estimates after Deleting Factors

        Coef    S.E. Wald Z P
[1,] -0.5928 0.04315 -13.74 0

Factors in Final Model

None
```

```
f.approx ← ols(Z ~ dzgroup + rcs(meanbp,5) + rcs(crea,4) +
                 rcs(age,5) + rcs(hrt,3) + scoma +
                 rcs(pafi.i,4) + pol(adlsc,2)+
                 rcs(resp,3), x=TRUE)
f.approx$stats
```

```
         n Model L.R.       d.f.         R2          g      Sigma
   537.000   1688.225     23.000      0.957      1.915      0.370
```

We can estimate the variance–covariance matrix of the coefficients of the reduced model using Equation 5.2 in Section 5.5.2. The computations below result in a covariance matrix that does not include elements related to the scale parameter. In the code `x` is the matrix T in Section 5.5.2.

```
V ← vcov(f,regcoef.only=TRUE)       # var(full model)
X ← cbind(Intercept=1, g$x)         # full model design
x ← cbind(Intercept=1, f.approx$x) # approx. model design
w ← solve(t(x) %*% x, t(x)) %*% X  # contrast matrix
v ← w %*% V %*% t(w)
```

Let's compare the variance estimates (diagonals of `v`) with variance estimates from a reduced model that is fitted against the actual outcomes.

```
f.sub ← psm(S ~ dzgroup + rcs(meanbp,5) + rcs(crea,4) +
              rcs(age,5) + rcs(hrt,3) + scoma + rcs(pafi.i,4) +
              pol(adlsc,2)+ rcs(resp,3), dist='lognormal')

diag(v)/diag(vcov(f.sub,regcoef.only=TRUE))
```

```
          Intercept         dzgroup=Coma dzgroup=MOSF w/Malig
              0.981                0.979               0.979
             meanbp               meanbp'             meanbp''
              0.977                0.979               0.979
           meanbp'''                 crea                crea'
              0.979                0.979               0.979
             crea''                  age                 age'
              0.979                0.982               0.981
              age''               age'''                 hrt
              0.981                0.980               0.978
```

```
          hrt'          scoma         pafi.i
         0.976          0.979          0.980
      pafi.i'       pafi.i''          adlsc
         0.980          0.980          0.981
     adlsc^2           resp          resp'
         0.981          0.978          0.977
```

```
r ← diag(v)/diag(vcov(f.sub,regcoef.only=TRUE))
r[c(which.min(r), which.max(r))]
```

```
 hrt'   age
0.976 0.982
```

The estimated variances from the reduced model are actually slightly smaller than those that would have been obtained from stepwise variable selection in this case, had variable selection used a stopping rule that resulted in the same set of variables being selected. Now let us compute Wald statistics for the reduced model.

```
f.approx$var ← v
latex(anova(f.approx, test='Chisq', ss=FALSE), file='',
      label='tab:support-anovaa')
```

The results are shown in Table 19.2. Note the similarity of the statistics to those found in the table for the full model. This would not be the case had deleted variables been very collinear with retained variables.

The equation for the simplified model follows. The model is also depicted graphically in Figure 19.12. The nomogram allows one to calculate mean and median survival time. Survival probabilities could have easily been added as additional axes.

```
# Typeset mathematical form of approximate model
latex(f.approx, file='')
```

$$\mathrm{E}(\mathrm{Z}) = X\beta, \quad \text{where}$$

$$
\begin{aligned}
X\hat{\beta} = \; & \\
& -2.51 \\
& -1.94[\text{Coma}] - 1.75[\text{MOSF w/Malig}] \\
& +0.068\text{meanbp} - 3.08\times 10^{-5}(\text{meanbp}-41.8)_{+}^{3} + 7.9\times 10^{-5}(\text{meanbp}-61)_{+}^{3} \\
& -4.91\times 10^{-5}(\text{meanbp}-73)_{+}^{3} + 2.61\times 10^{-6}(\text{meanbp}-109)_{+}^{3} - 1.7\times 10^{-6}(\text{meanbp}-135)_{+}^{3} \\
& -0.553\text{crea} - 0.229(\text{crea}-0.6)_{+}^{3} + 0.45(\text{crea}-1.1)_{+}^{3} - 0.233(\text{crea}-1.94)_{+}^{3} \\
& +0.0131(\text{crea}-7.32)_{+}^{3} \\
& -0.0165\text{age} - 1.13\times 10^{-5}(\text{age}-28.5)_{+}^{3} + 4.05\times 10^{-5}(\text{age}-49.5)_{+}^{3} \\
& -2.15\times 10^{-5}(\text{age}-63.7)_{+}^{3} - 2.68\times 10^{-5}(\text{age}-72.7)_{+}^{3} + 1.9\times 10^{-5}(\text{age}-85.6)_{+}^{3} \\
& -0.0136\text{hrt} + 6.09\times 10^{-7}(\text{hrt}-60)_{+}^{3} - 1.68\times 10^{-6}(\text{hrt}-111)_{+}^{3} + 1.07\times 10^{-6}(\text{hrt}-140)_{+}^{3} \\
& -0.0135\ \text{scoma} \\
& +0.0161\text{pafi.i} - 4.77\times 10^{-7}(\text{pafi.i}-88)_{+}^{3} + 9.11\times 10^{-7}(\text{pafi.i}-167)_{+}^{3}
\end{aligned}
$$

Table 19.2 Wald Statistics for `Z`

	χ^2	d.f.	P
dzgroup	55.94	2	< 0.0001
meanbp	29.87	4	< 0.0001
Nonlinear	9.84	3	0.0200
crea	39.04	3	< 0.0001
Nonlinear	24.37	2	< 0.0001
age	18.12	4	0.0012
Nonlinear	0.34	3	0.9517
hrt	9.87	2	0.0072
Nonlinear	0.40	1	0.5289
scoma	9.85	1	0.0017
pafi.i	14.01	3	0.0029
Nonlinear	6.66	2	0.0357
adlsc	9.71	2	0.0078
Nonlinear	2.87	1	0.0904
resp	9.65	2	0.0080
Nonlinear	7.13	1	0.0076
TOTAL NONLINEAR	58.08	13	< 0.0001
TOTAL	252.32	23	< 0.0001

$$-5.02\times10^{-7}(\text{pafi.i}-276)_+^3+6.76\times10^{-8}(\text{pafi.i}-426)_+^3-0.369\,\text{adlsc}+0.0409\,\text{adlsc}^2$$
$$+0.0394\text{resp}-9.11\times10^{-5}(\text{resp}-10)_+^3+0.000176(\text{resp}-24)_+^3-8.5\times10^{-5}(\text{resp}-39)_+^3$$

and $[c] = 1$ if subject is in group c, 0 otherwise; $(x)_+ = x$ if $x > 0$, 0 otherwise.

```
# Derive S functions that express mean and quantiles
# of survival time for specific linear predictors
# analytically
expected.surv ← Mean(f)
quantile.surv ← Quantile(f)
latex(expected.surv, file='', type='Sinput')
```

```
expected.surv ← function (lp = NULL,
                          parms = 0.802352037606488)
{
    names(parms) ← NULL
    exp(lp + exp(2 * parms)/2)
}
```

```
latex(quantile.surv, file='', type='Sinput')
```

```
quantile.surv ← function (q = 0.5, lp = NULL,
                          parms = 0.802352037606488)
```

```
{
    names(parms) ← NULL
    f ← function(lp, q, parms) lp + exp(parms) * qnorm(q)
    names(q) ← format(q)
    drop(exp(outer(lp, q, FUN = f, parms = parms)))
}
```

```
median.surv   ← function(x) quantile.surv(lp=x)
```

```
# Improve variable labels for the nomogram
f.approx ← Newlabels(f.approx, c('Disease Group',
          'Mean Arterial BP','Creatinine','Age','Heart Rate',
          'SUPPORT Coma Score','PaO2/(.01*FiO2)','ADL',
          'Resp. Rate'))
nom ←
  nomogram(f.approx,
           pafi.i=c(0, 50, 100, 200, 300, 500, 600, 700, 800,
                    900),
           fun=list('Median Survival Time'=median.surv,
                    'Mean Survival Time'  =expected.surv),
           fun.at=c(.1,.25,.5,1,2,5,10,20,40))
plot(nom, cex.var=1, cex.axis=.75, lmgp=.25)
# Figure 19.12
```

19.6 Problems

Analyze the Mayo Clinic PBC dataset.

1. Graphically assess whether Weibull (extreme value), exponential, log-logistic, or log-normal distributions will fit the data, using a few apparently important stratification factors.
2. For the best fitting parametric model from among the four examined, fit a model containing several sensible covariables, both categorical and continuous. Do a Wald test for whether each factor in the model has an association with survival time, and a likelihood ratio test for the simultaneous contribution of all predictors. For classification factors having more than two levels, be sure that the Wald test has the appropriate degrees of freedom. For continuous factors, verify or relax linearity assumptions. If using a Weibull model, test whether a simpler exponential model would be appropriate. Interpret all estimated coefficients in the model. Write the full survival model in mathematical form. Generate a predicted survival curve for a patient with a given set of characteristics.

See [361] for an analysis of this dataset using linear splines in time and in the covariables.

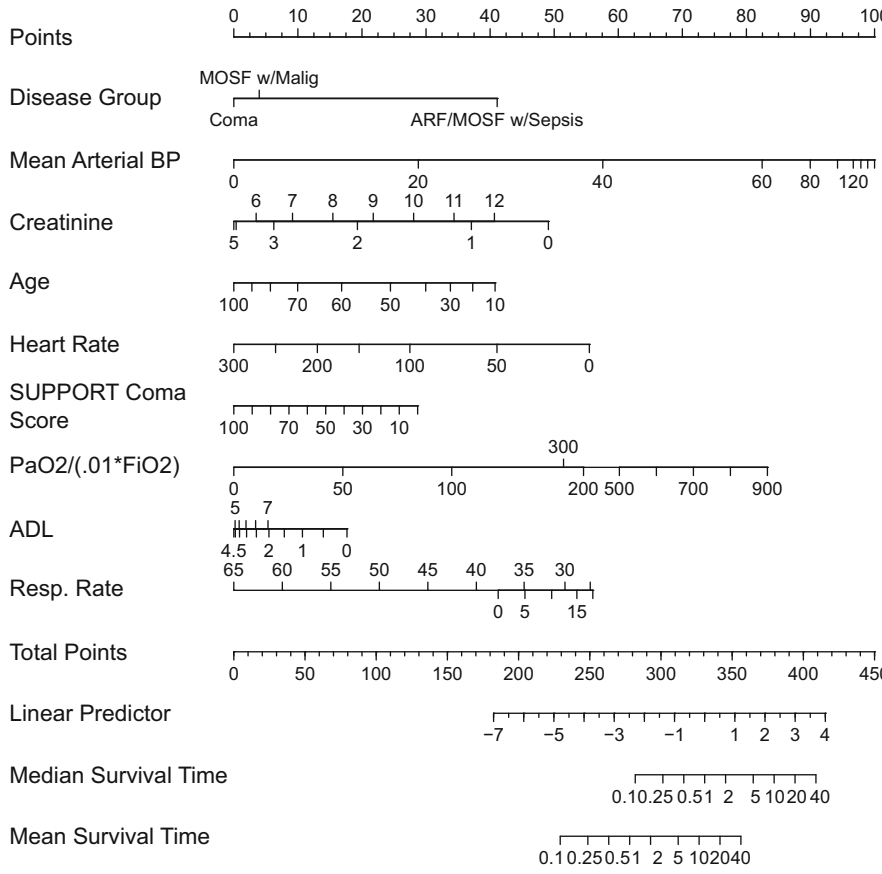


Fig. 19.12 Nomogram for predicting median and mean survival time, based on approximation of full model