Extensions to the Cox Model

Time Dependent Covariates

David M. Rocke

November 5, 2024

Bone Marrow Transplant Data

- Copelan et al. (1991) study of allogenic bone marrow transplant therapy for acute myeloid leukemia (AML) and acute lymphoblastic leukemia (ALL).
- Possible intermediate events are graft vs. host disease (GVHD), an immunological rejection response to the transplant, and platelet recovery, a return of platelet count to normal levels. One or the other, both in either order, or neither may occur.
- End point events are relapse of the disease or death.
- Any or all of these events may be censored.

KMsurv bmt data

```
The bmt data frame has 137 rows and 22 columns.

This data frame contains the following columns:

group   Disease Group 1-ALL, 2-AML Low Risk, 3-AML High Risk
t1      Time To Death Or On Study Time
t2      Disease Free Survival Time (Time To Relapse, Death Or End Of Study)
d1      Death Indicator 1-Dead 0-Alive
d2      Relapse Indicator 1-Relapsed, 0-Disease Free
d3      Disease Free Survival Indicator 1-Dead Or Relapsed, 0-Alive Disease Free)
ta      Time To Acute Graft-Versus-Host Disease
da      Acute GVHD Indicator 1-Developed Acute GVHD 0-Never Developed Acute GVHD)
tc      Time To Chronic Graft-Versus-Host Disease
dc      Chronic GVHD Indicator 1-Developed Chronic GVHD
          0-Never Developed Chronic GVHD
tp      Time To Platelet Recovery
dp      Platelet Recovery Indicator 1-Platelets Returned To Normal,
          0-Platelets Never Returned to Normal
```

KMsurv bmt data

```
z1      Patient Age In Years
z2      Donor Age In Years
z3      Patient Sex: 1-Male, 0-Female
z4      Donor Sex: 1-Male, 0-Female
z5      Patient CMV Status: 1-CMV Positive, 0-CMV Negative
z6      Donor CMV Status: 1-CMV Positive, 0-CMV Negative
z7      Waiting Time to Transplant In Days
z8      FAB: 1-FAB Grade 4 Or 5 and AML, 0-Otherwise
z9      Hospital: 1-The Ohio State University, 2-Alferd , 3-St. Vincent,
          4-Hahnemann
z10     MTX Used as a Graft-Versus-Host- Prophylactic: 1-Yes 0-No
```

Bone Marrow Transplant Example

- The main endpoint is disease-free survival (`t2` and `d3`) for the three risk groups, ALL, AML Low Risk, and AML High Risk.
- We are also interested in possibly using the covariates z1–z10 to adjust for other factors. We can do this with stepwise regression or hand examination of the results of adding or removing variables.
- In addition, the time-varying covariates for acute GVHD, chronic GVHD, and platelet recovery may be useful.

Time-Dependent Covariates

- A *time-dependent covariate* is one that changes value in the course of the study.
- For variables like age that change in a linear manner with time, we can just use the value at the start.
- But it may be plausible that when and if GVHD occurs, the risk of relapse or death increases, and when and if platelet recovery occurs, the risk decreases.

Formulation in R

- We form a variable `precovery` which is $= 0$ before platelet recovery and is $= 1$ after platelet recovery, if it occurs.
- For each subject where platelet recovery occurs, we set up multiple records (lines in the data frame); for example one from $t = 0$ to the time of platelet recovery, and one from that time to relapse, or death, or end of study.
- We do the same for acute GVHD and chronic GVHD.
- For each record, the covariates are constant.

```
id group   t1   t2 d1 d2 d3 ta da  tc dc tp dp
 1   ALL 2081 2081  0  0  0 67  1 121  1 13  1


times are

t  = 0     time of transplant
tp = 13    platelet recovery
ta = 67    acute GVHD onset
tc = 121   chronic GVHD onset
t2 = 2081  end of study, patient not relapsed or dead

id group   tstart tstop agvhd cgvhd precovery status
 1   ALL        0    13     0     0         0      0
 1   ALL       13    67     0     0         1      0
 1   ALL       67   121     1     0         1      0
 1   ALL      121  2081     1     1         1      0  #this status could be 1
```

- Let A, C, and P stand for the event occurs for that patient at some time. Each of the eight possible combinations of A or not-A, with C or not-C, with P or not-P occurs in this data set.
- A always occurs before C and P always occurs before C if both occur; this is for medical reasons.
- Thus there are ten kinds of patients in the data set: None, A, C, P, AC, AP, PA, PC, APC, and PAC.
- There could be as many as $1 + 3 + (3)(2) + 6 = 16$
- This is why a package to assist with this is helpful

Possible and Actual Event Sequences

Sequence	Occurs?	Sequence	Occurs?
None	Y	CP	—
A	Y	PC	Y
C	Y	ACP	—
P	Y	APC	Y
AC	Y	CAP	—
CA	—	CPA	—
AP	Y	PAC	Y
PA	Y	PCA	–

- Different subjects could have 1, 2, 3, or 4 intervals depending on which of acute GVHD, chronic GVHD, and/or platelet recovery occurred.
- The final interval for any subject has status $= 1$ if the subject relapsed or died at the end of that interval, otherwise the status is 0.
- Any earlier intervals have status $= 0$.
- Even though there might be multiple lines in the data frame, there is never more than one event, so no alterations need be made in the estimation procedures or in the interpretation of the output.
- The function `tmerge` in the `survival package` eases the process of constructing the new data frame.

KMsurv bmt data

```
z1      Patient Age In Years
z2      Donor Age In Years
z3      Patient Sex: 1-Male, 0-Female
z4      Donor Sex: 1-Male, 0-Female
z5      Patient CMV Status: 1-CMV Positive, 0-CMV Negative
z6      Donor CMV Status: 1-CMV Positive, 0-CMV Negative
z7      Waiting Time to Transplant In Days
z8      FAB: 1-FAB Grade 4 Or 5 and AML, 0-Otherwise
z9      Hospital: 1-The Ohio State University, 2-Alferd , 3-St. Vincent,
          4-Hahnemann
z10     MTX Used as a Graft-Versus-Host- Prophylactic: 1-Yes 0-No
```

Starting with all these covariates, we eliminated sequentially Patient and Donor Sex, Patient and Donor CMV Status, Waiting time, and MTX.

Fixed Covariates for the bmt Data

```
require(KMsurv)
require(survival)
data(bmt)
nsubj <- dim(bmt)[1]
id <- 1:nsubj
bmt1 <- data.frame(id,bmt)    #to identify the subject across multiple lines
bmt1$group <- factor(bmt1$group,labels=c("ALL","AML-Low","AML-High"))
bmt1$z9 <- factor(bmt1$z9) #hospital factor
bmt1.surv <- with(bmt1,Surv(t2,d3))

> drop1(coxph(bmt1.surv~group+z1*z2+z8+z9,data=bmt1),test="Chisq")
Single term deletions

Model:
bmt1.surv ~ group + z1 * z2 + z8 + z9
       Df    AIC     LRT  Pr(>Chi)
<none>    719.58
group   2 721.76  6.1738 0.0456426 *       #ALL, AML-High, AML-Low
z8      1 726.43  8.8504 0.0029303 **      #1-FAB Grade 4 Or 5 and AML, 0-Else
z9      3 725.79 12.2066 0.0067079 **      #Hospital
z1:z2   1 729.23 11.6537 0.0006407 ***     #Patient Age by Donor Age interaction
```

```
> summary(coxph(bmt1.surv~group+z1*z2+z8+z9,data=bmt1))
Call:
coxph(formula = bmt1.surv ~ group + z1 * z2 + z8 + z9, data = bmt1)

  n= 137, number of events= 83

                   coef  exp(coef)   se(coef)      z Pr(>|z|)
groupAML-Low  -0.7759558  0.4602636  0.3635689 -2.134 0.032820 *
groupAML-High -0.2379396  0.7882503  0.3577568 -0.665 0.505995
z1            -0.0982054  0.9064627  0.0378372 -2.595 0.009446 **
z2            -0.0823307  0.9209674  0.0301442 -2.731 0.006310 **
z8             0.8341968  2.3029635  0.2822471  2.956 0.003121 **
z92            0.7772511  2.1754838  0.3393736  2.290 0.022007 *
z93           -0.2766900  0.7582896  0.3365979 -0.822 0.411066
z94           -0.8881221  0.4114276  0.4204024 -2.113 0.034639 *
z1:z2          0.0035154  1.0035216  0.0009591  3.665 0.000247 ***
```

We will use the two age variables and FAB score in the following.
We omit the hospital effect since the significance test is possibly invalid (hospital-level effect, not patient effect).

```
> summary(coxph(bmt1.surv~group,data=bmt1))
                 coef exp(coef) se(coef)      z Pr(>|z|)
groupAML-Low  -0.5742    0.5632   0.2873 -1.999   0.0457 *
groupAML-High  0.3834    1.4673   0.2674  1.434   0.1516

> summary(coxph(bmt1.surv~group+z8,data=bmt1))
                  coef exp(coef) se(coef)      z Pr(>|z|)
groupAML-Low  -0.90450   0.40475  0.32031 -2.824  0.00475 **
groupAML-High -0.05195   0.94938  0.32060 -0.162  0.87128
z8             0.76950   2.15868  0.27032  2.847  0.00442 **
```

With group alone, AML-High is riskier than ALL and AML-Low is less risky. The FAB variable z8, which is 1 only for AML, 1/3 of the AML-Low cases and 60% of the AML-High cases, this absorbs some of the risk of the riskiest AML cases, so that the group effect shows both AML groups as less risky than ALL.

```
> newgroup <- unclass(bmt1$group)+bmt1$z8*3   #five different numerical values
> with(bmt1,table(unclass(group)+z8*3))

 1  2  3  5  6
38 36 18 18 27
> with(bmt1,table(group,z8))
          z8
group       0  1
  ALL      38  0
  AML-Low  36 18
  AML-High 18 27
> newgroup <- factor(newgroup,
   labels=c("ALL","AML-Low","AML-High","AML-Low+FAB","AML-High+FAB"))
> summary(coxph(bmt1.surv~newgroup,data=bmt1))

                        coef exp(coef) se(coef)      z Pr(>|z|)
newgroupAML-Low      -0.7759    0.4603   0.3384 -2.293  0.02185 *
newgroupAML-High     -0.2144    0.8070   0.3791 -0.566  0.57172
newgroupAML-Low+FAB  -0.2829    0.7536   0.3653 -0.774  0.43868
newgroupAML-High+FAB  0.7935    2.2112   0.2903  2.734  0.00626 **

> AIC(coxph(bmt1.surv~newgroup,data=bmt1))
[1] 731.9691
> AIC(coxph(bmt1.surv~group+z8,data=bmt1))
[1] 730.8491    #Lower AIC, so we use group and FAB Score separately
```

Construction of TDC Data Set

Using tmerge we set up the time-dependent covariates data set.

```
bmt2 <- tmerge(bmt1,bmt1,id=id,tstop=t2)           #sets up new data set
bmt2 <- tmerge(bmt2,bmt1,id=id,agvhd=tdc(ta))      #adds aghvd as tdc
bmt2 <- tmerge(bmt2,bmt1,id=id,cgvhd=tdc(tc))      #adds cghvd as tdc
bmt2 <- tmerge(bmt2,bmt1,id=id,precovery=tdc(tp)) #adds platelet recovery as tdc

status <- as.integer(with(bmt2,(tstop==t2 & d3)))

# status only = 1 if at end of t2 and not censored

bmt2 <- data.frame(bmt2,status)

bmt2.surv <- with(bmt2,Surv(time=tstart,time2=tstop,event=status,type="counting"))

#counting process formulation of Surv
```

	id	group	t1	t2	d1	d2	d3	ta	da	tc	dc	tp	dp	z1	z2	z8	tstart	tstop	agvhd	cgvhd	precovery	status
1	1	ALL	2081	2081	0	0	0	67	1	121	1	13	1	26	33	0	0	13	0	0	0	0
2	1	ALL	2081	2081	0	0	0	67	1	121	1	13	1	26	33	0	13	67	0	0	1	0
3	1	ALL	2081	2081	0	0	0	67	1	121	1	13	1	26	33	0	67	121	1	0	1	0
4	1	ALL	2081	2081	0	0	0	67	1	121	1	13	1	26	33	0	121	2081	1	1	1	0
5	2	ALL	1602	1602	0	0	0	1602	0	139	1	18	1	21	37	0	0	18	0	0	0	0
6	2	ALL	1602	1602	0	0	0	1602	0	139	1	18	1	21	37	0	18	139	0	0	1	0
7	2	ALL	1602	1602	0	0	0	1602	0	139	1	18	1	21	37	0	139	1602	0	1	1	0
8	3	ALL	1496	1496	0	0	0	1496	0	307	1	12	1	26	35	0	0	12	0	0	0	0
9	3	ALL	1496	1496	0	0	0	1496	0	307	1	12	1	26	35	0	12	307	0	0	1	0
10	3	ALL	1496	1496	0	0	0	1496	0	307	1	12	1	26	35	0	307	1496	0	1	1	0
11	4	ALL	1462	1462	0	0	0	70	1	95	1	13	1	17	21	0	0	13	0	0	0	0
12	4	ALL	1462	1462	0	0	0	70	1	95	1	13	1	17	21	0	13	70	0	0	1	0
13	4	ALL	1462	1462	0	0	0	70	1	95	1	13	1	17	21	0	70	95	1	0	1	0
14	4	ALL	1462	1462	0	0	0	70	1	95	1	13	1	17	21	0	95	1462	1	1	1	0
...																						
42	14	ALL	1167	1167	0	0	0	39	1	487	1	1167	0	27	22	0	0	39	0	0	0	0
43	14	ALL	1167	1167	0	0	0	39	1	487	1	1167	0	27	22	0	39	487	1	0	0	0
44	14	ALL	1167	1167	0	0	0	39	1	487	1	1167	0	27	22	0	487	1167	1	1	0	0
45	15	ALL	418	418	1	0	1	418	0	220	1	21	1	18	14	0	0	21	0	0	0	0
46	15	ALL	418	418	1	0	1	418	0	220	1	21	1	18	14	0	21	220	0	0	1	0
47	15	ALL	418	418	1	0	1	418	0	220	1	21	1	18	14	0	220	418	0	1	1	1
48	16	ALL	417	383	1	1	1	417	0	417	0	16	1	15	20	0	0	16	0	0	0	0
49	16	ALL	417	383	1	1	1	417	0	417	0	16	1	15	20	0	16	383	0	0	1	1

Add Time-Dependent Covariates

```
> summary(coxph(bmt2.surv~group+z1*z2+z8+agvhd+cgvhd+precovery,data=bmt2))

  n= 341, number of events= 83

                    coef  exp(coef)   se(coef)      z Pr(>|z|)
groupAML-Low  -1.0385144  0.3539802  0.3582204 -2.899  0.00374 **
groupAML-High -0.3804809  0.6835326  0.3748670 -1.015  0.31012
z1            -0.0733511  0.9292745  0.0359557 -2.040  0.04135 *
z2            -0.0764062  0.9264398  0.0301965 -2.530  0.01140 *
z8             0.8057002  2.2382632  0.2842726  2.834  0.00459 **
agvhd          0.1505649  1.1624908  0.3068484  0.491  0.62365
cgvhd         -0.1161359  0.8903542  0.2890463 -0.402  0.68784
precovery     -0.9411227  0.3901895  0.3478611 -2.705  0.00682 **
z1:z2          0.0028946  1.0028988  0.0009435  3.068  0.00216 **
```

Neither acute GVHD nor chronic GVHD has a statistically significant effect here or in a model with the other one removed. Platelet recovery is highly significant.

```
> summary(coxph(bmt2.surv~group+z1*z2+z8+precovery,data=bmt2))

  n= 341, number of events= 83

                    coef  exp(coef)   se(coef)      z Pr(>|z|)
groupAML-Low  -1.0325200  0.3561084  0.3532019 -2.923  0.00346 **
groupAML-High -0.4138881  0.6610749  0.3652095 -1.133  0.25709
z1            -0.0709647  0.9314948  0.0354533 -2.002  0.04532 *
z2            -0.0760524  0.9267677  0.0300071 -2.534  0.01126 *
z8             0.8119262  2.2522421  0.2832310  2.867  0.00415 **
precovery     -0.9835053  0.3739978  0.3379970 -2.910  0.00362 **
z1:z2          0.0028716  1.0028758  0.0009355  3.070  0.00214 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

              exp(coef) exp(-coef) lower .95 upper .95
groupAML-Low     0.3561     2.8081    0.1782    0.7116
groupAML-High    0.6611     1.5127    0.3231    1.3524
z1               0.9315     1.0735    0.8690    0.9985
z2               0.9268     1.0790    0.8738    0.9829
z8               2.2522     0.4440    1.2928    3.9238
precovery        0.3740     2.6738    0.1928    0.7254
z1:z2            1.0029     0.9971    1.0010    1.0047
```

Model Checking

We can use all the same tools for model checking in data sets with time dependent covariates as we do with data sets with only fixed covariates. This includes

- Schoenfeld residuals correlated with "time" to test for proportionality of hazards.
- Martingale residuals plotted vs numeric covariates to check for functional form.
- Martingale residuals and deviance residuals plotted vs the linear predictor to identify possible outliers.
- Columns of dfbeta to identify possible influential points: points whose removal changes the fit importantly.

We won't use the Cox-Snell residuals since this plot has low capacity to detect problems.

Model Checking

The original data set is 137 rows and 22 columns, corresponding to 137 patients with a number of events that depends on the type of event:

Number of Events of Various Types		
d1	death	81
d2	relapse	42
d3	disease-free survival	83
da	acute gvhd	26
dc	chronic gvhd	61
dp	platelet recovery	120

Model checking when using the original data set is as we have seen before.

Model Checking

Number of Events of Various Types	
death without relapse	41
relapse then death	40
relapse only	2
neither death nor relapse	54
death without platelet recovery	16
platelet recovery then death	65
platelet recovery without death	55
neither death nor platelet recovery	1

$$55/120 = 45.8\% \text{ Survival rate with precovery}$$
$$1/17 = 5.9\% \text{ Survival rate without precovery}$$

Number of Residuals

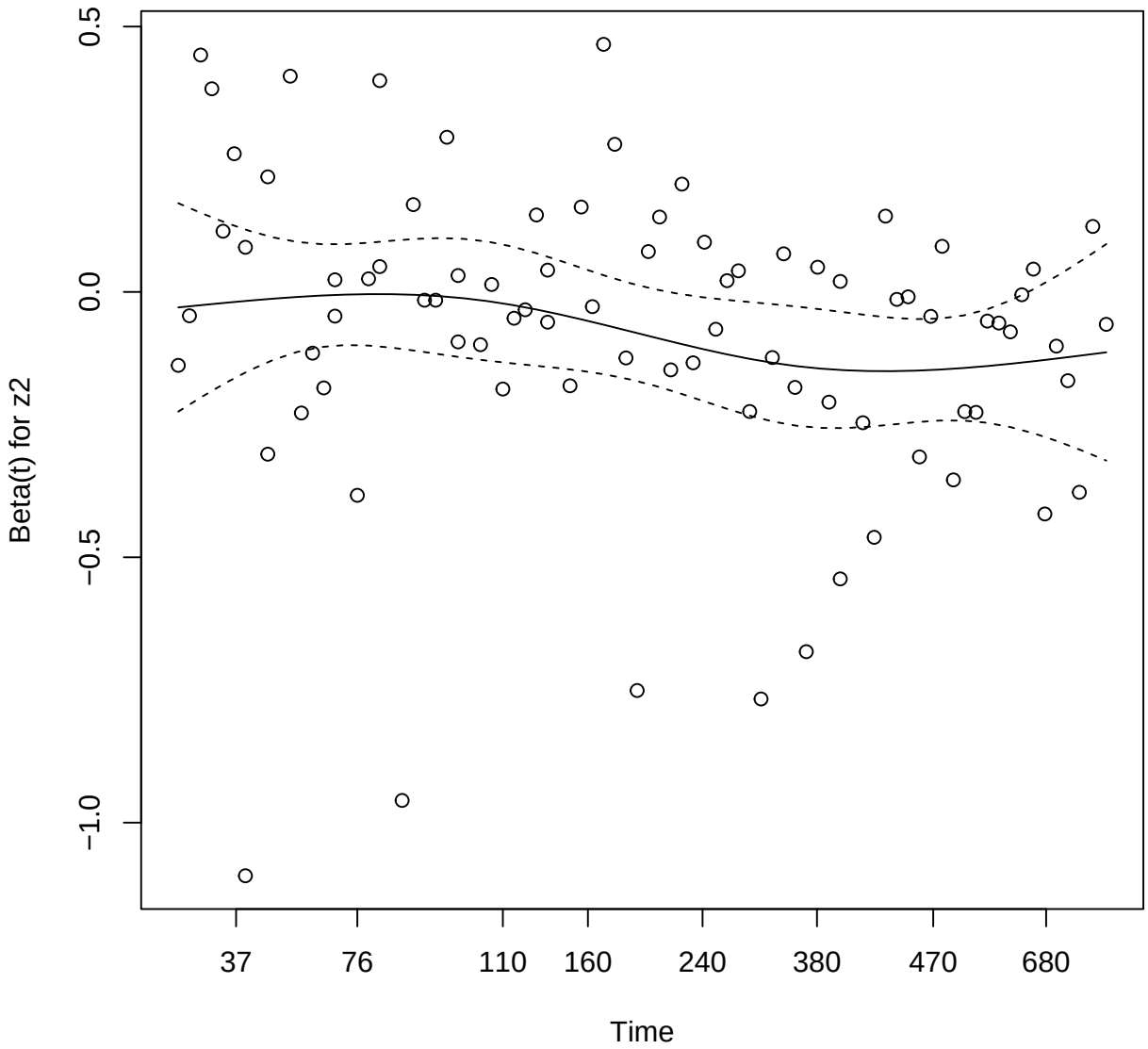
The original data set is 137 rows and 22 columns, corresponding to 137 patients. The data set for time-dependent analysis is 341 rows by 29 columns. This means that there are 341 different patient by time-dependent covariate intervals, about an average of 2.5 intervals per patient. The first extra column is `id` one unique value per patient, and the others are `tstart`, `tstop`, delimiting the intervals, `agvhd`, `cgvhd`, `precovery`, stating which events have already occurred before that interval, and `status` indicating whether the interval terminates with recurrence or death.

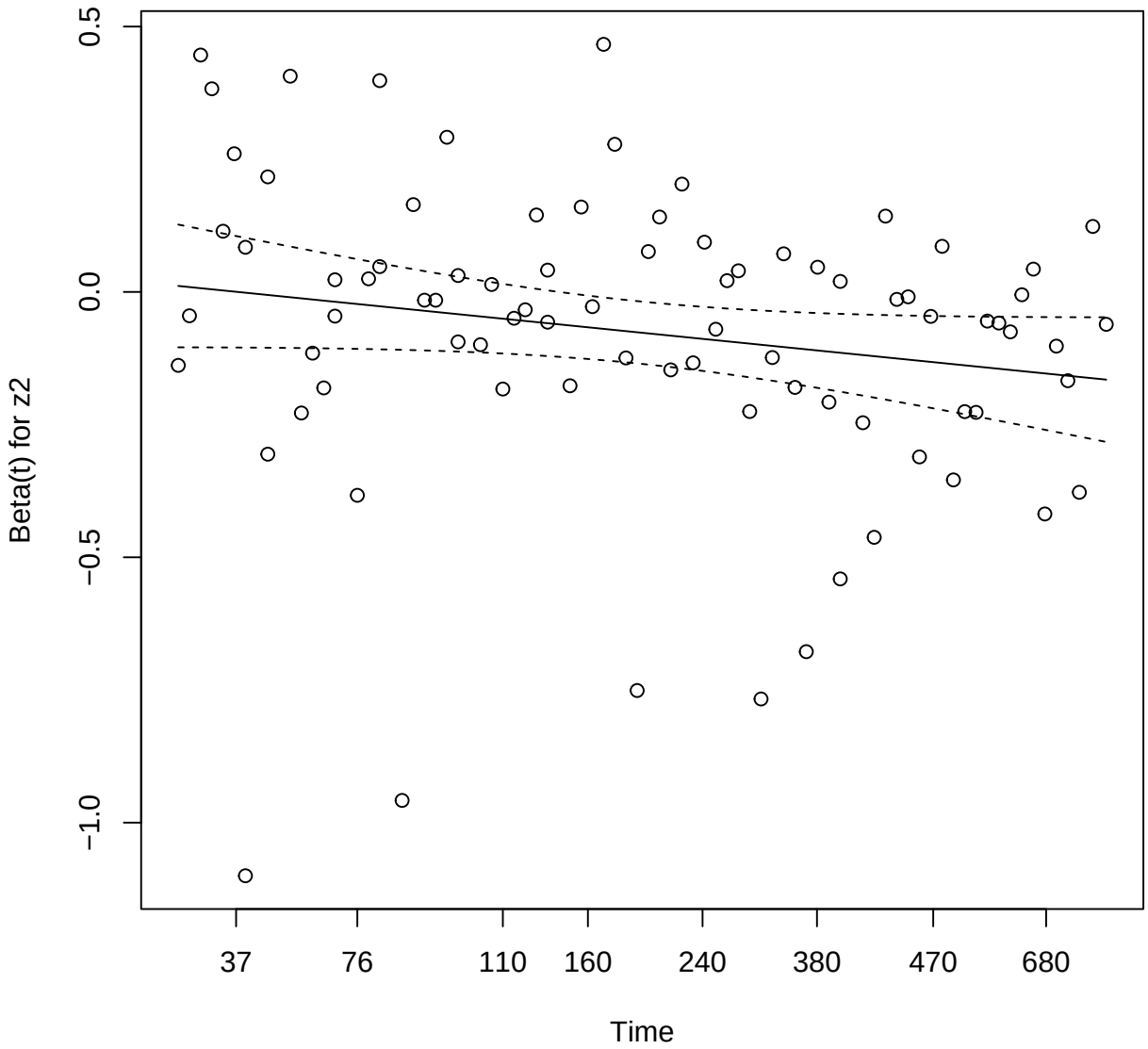
Number of Residuals

An argument to the `residual` command is `collapse` which has the default value `collapse = F = FALSE` which gives us 341 residuals or `collapse = id` which combines all the residuals for each patient, resulting in 137 residuals. Both approaches can be useful. The first gives us one residual per patient per values of the time-dependent covariates and the second has one residual per patient. If plotted vs. something in the data set it has to be from `bmt2` in the first case and bmt1 in the second, even though the residual vector is derived from the model using the data set `bmt2`.

Schoenfeld Residuals

```
bmt2.cox <- coxph(bmt2.surv~group+z1*z2+z8+precovery,data=bmt2)
bmt2.zph <- cox.zph(bmt2.cox)
print(bmt2.zph)
plot.zph <- function(i,df=4){        #df = 4 is the default degree of the spline
  plot(bmt2.zph[i],df=df)            #df = 2 uses linear splines
}
            chisq df    p
group      1.0458  2 0.59       #Disease
z1         0.6625  1 0.42       #Patient Age
z2         2.3980  1 0.12       #Donor Age
z8         0.3216  1 0.57       #FAB Score
precovery 0.0721  1 0.79       #Platelet Recovery
z1:z2      0.9210  1 0.34       #Age Interaction
GLOBAL     6.3820  7 0.50       #No major signs of non-proportionality

pdf("Schoenfeld3.pdf")       #These are for z2 = donor age
plot.zph(3)                  #This is column 3/7 of the scaled schoenfeld resids
dev.off()
pdf("Schoenfeld3a.pdf")
plot.zph(3,df=2)
dev.off()
```


Beta(t) for z2
0.5
0.0
−0.5
−1.0
37
76
110
160
240
380
470
680
Time

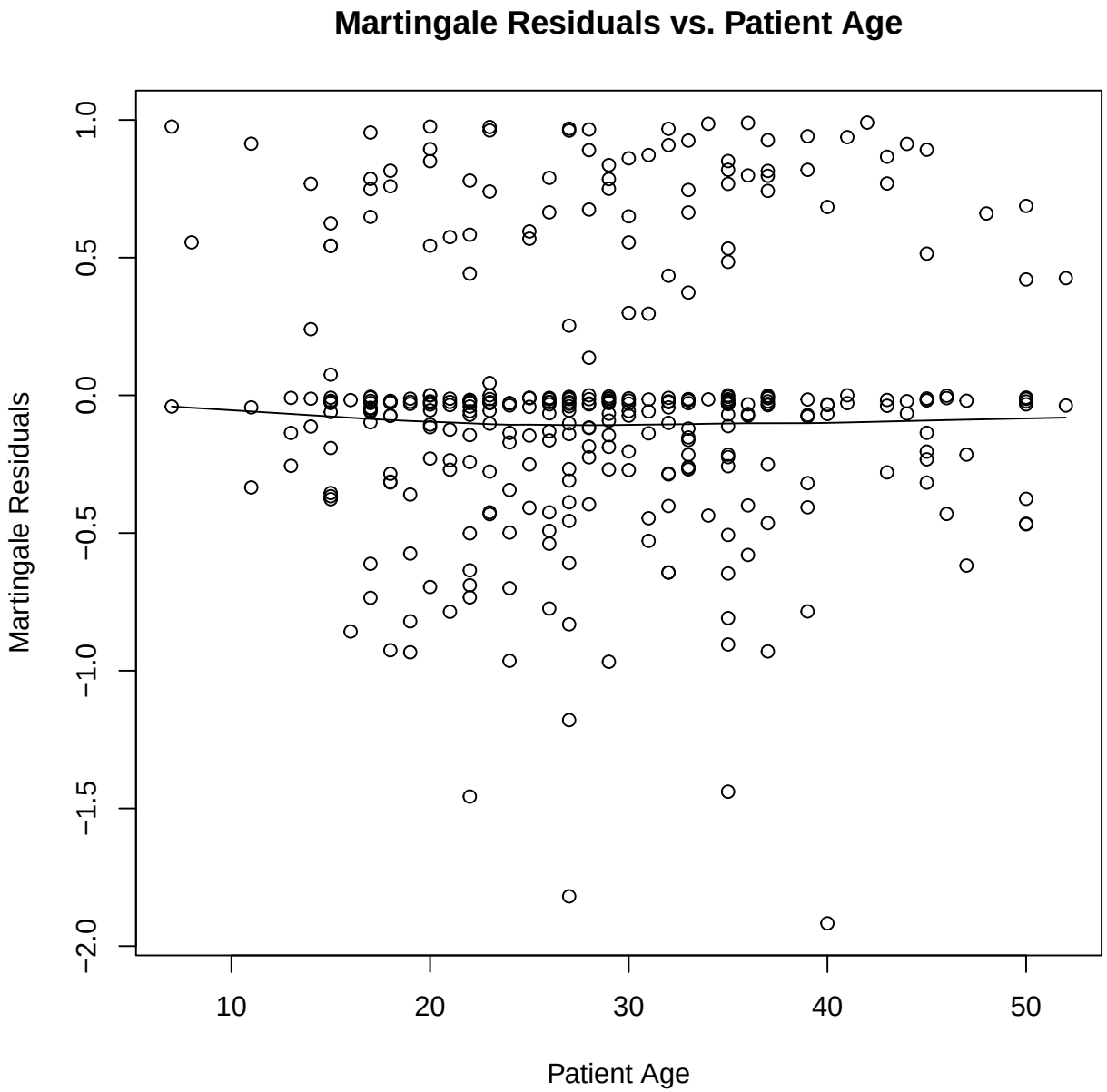

Beta(t) for z2
0.5
0.0
−0.5
−1.0
37
76
110
160
240
380
470
680
Time

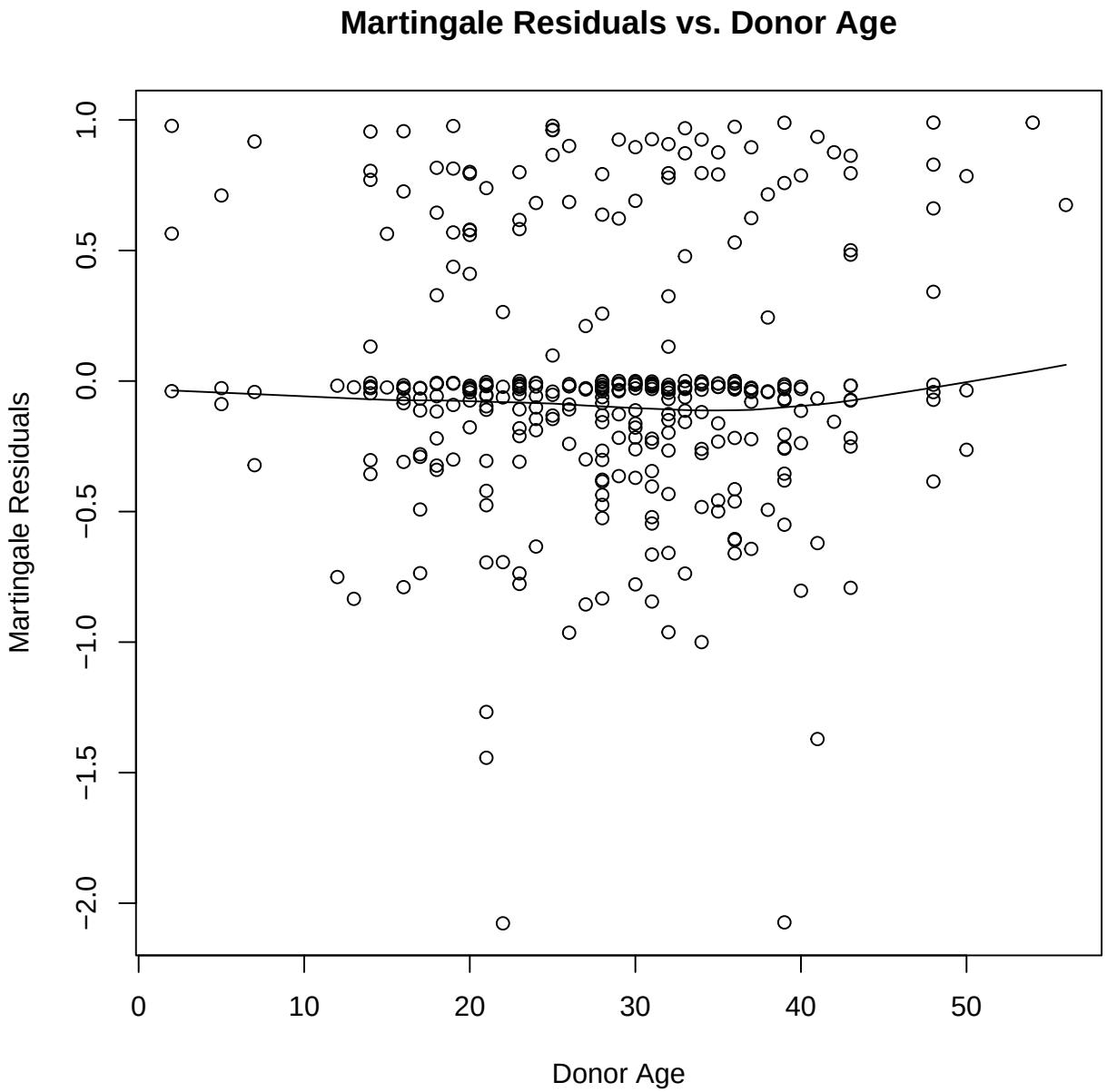
Martingale Residuals

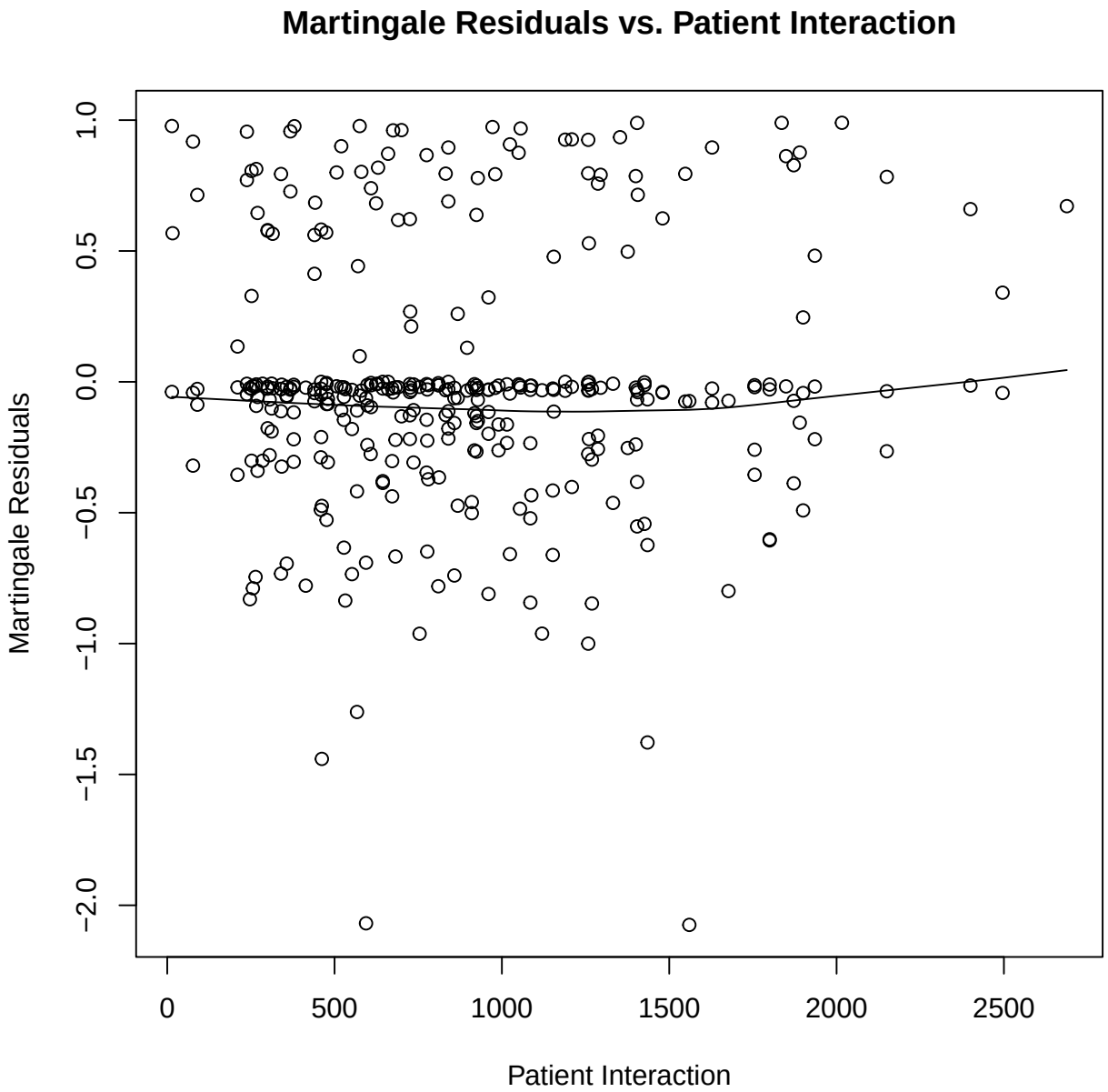
```
plot.mres.z1 <- function(){
  mres <- residuals(coxph(bmt2.surv~group+z2+z8+precovery,data=bmt2),
      type="martingale")
  plot(bmt2$z1,mres,xlab="Patient Age",ylab="Martingale Residuals")
  lines(lowess(bmt2$z1,mres))
  title("Martingale Residuals vs. Patient Age")
}

plot.mres.z2 <- function(){
  mres <- residuals(coxph(bmt2.surv~group+z1+z8+precovery,data=bmt2),
      type="martingale")
  plot(bmt2$z2,mres,xlab="Donor Age",ylab="Martingale Residuals")
  lines(lowess(bmt2$z2,mres))
  title("Martingale Residuals vs. Donor Age")
}
```

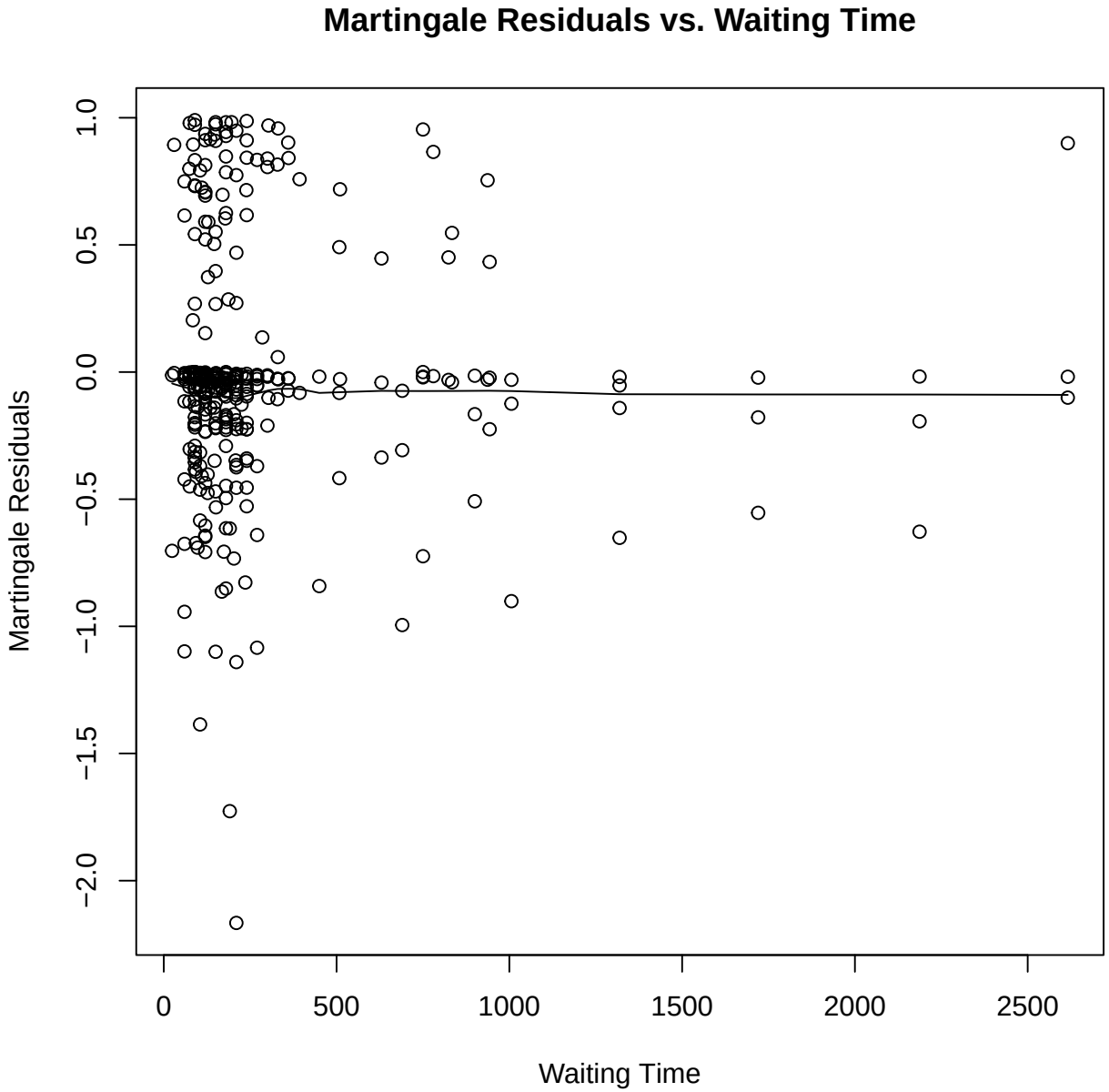
Martingale Residuals

```
plot.mres.z12 <- function(){
  mres <- residuals(coxph(bmt2.surv~group+z1+z2+z8+precovery,data=bmt2),
      type="martingale")
  plot(bmt2$z1*bmt2$z2,mres,xlab="Patient Interaction",
     ylab="Martingale Residuals")
  lines(lowess(bmt2$z1*bmt2$z2,mres))
  title("Martingale Residuals vs. Patient Interaction")
}

plot.mres.z7 <- function(){
  mres <- residuals(coxph(bmt2.surv~group+z1*z2+z8+precovery,data=bmt2),
      type="martingale")
  plot(bmt2$z7,mres,xlab="Waiting Time",ylab="Martingale Residuals")
  lines(lowess(bmt2$z7,mres))
  title("Martingale Residuals vs. Waiting Time")
}
```


Martingale Residuals vs. Patient Age
Martingale Residuals
1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
10
20
30
40
50
Patient Age

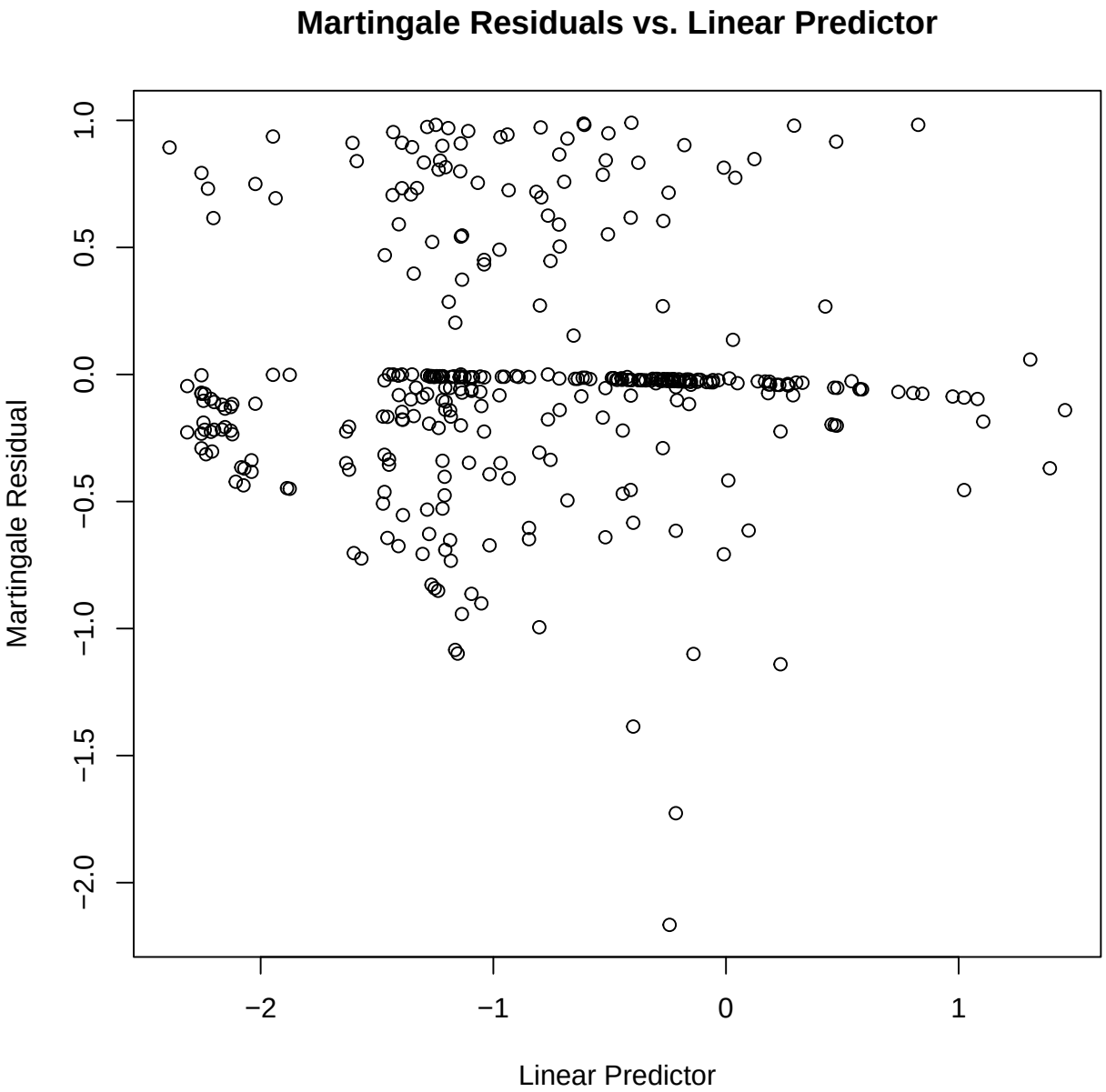

Martingale Residuals vs. Donor Age
Martingale Residuals
1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
0
10
20
30
40
50
Donor Age


Martingale Residuals vs. Patient Interaction
Martingale Residuals
1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
0
500
1000
1500
2000
2500
Patient Interaction


Martingale Residuals vs. Waiting Time
Martingale Residuals
1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
0
500
1000
1500
2000
2500
Waiting Time

Martingale and Deviance Residuals

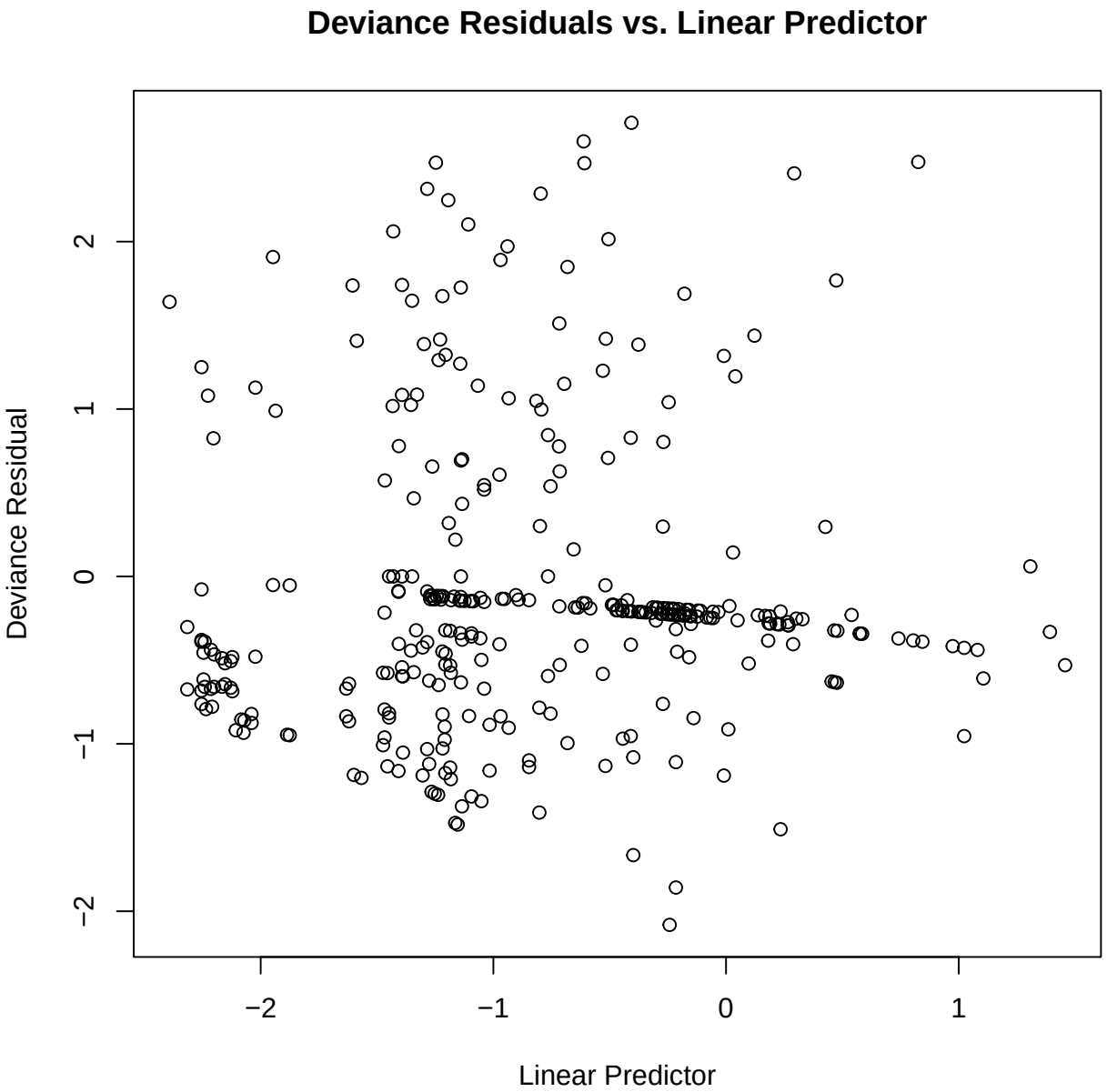
```
bmt2.mart <- residuals(bmt2.cox,type="martingale")
bmt2.dev <- residuals(bmt2.cox,type="deviance")
bmt2.dfb <- residuals(bmt2.cox,type="dfbeta")
bmt2.preds <- predict(bmt2.cox)

plotr.mart <- function(){
  plot(bmt2.preds,bmt2.mart,xlab="Linear Predictor",ylab="Martingale Residual")
  title("Martingale Residuals vs. Linear Predictor")
}

plotr.dev <- function(){
  plot(bmt2.preds,bmt2.dev,xlab="Linear Predictor",ylab="Deviance Residual")
  title("Deviance Residuals vs. Linear Predictor")
}
```



Three smallest martingale residuals are from patient id's 14, 100, and 103.

```
> bmt1[c(14,100,103),imp.vars1]
     id    group   t1   t2 d1 d2 d3   ta da  tc dc   tp dp z1 z2 z8
14   14      ALL 1167 1167  0  0  0   39  1 487  1 1167  0 27 22  0
100 100 AML-High 2024 2024  0  0  0 2024  0 180  1   16  1 35 41  1
103 103 AML-High  845  845  0  0  0  845  0 845  0   20  1 40 39  1
```

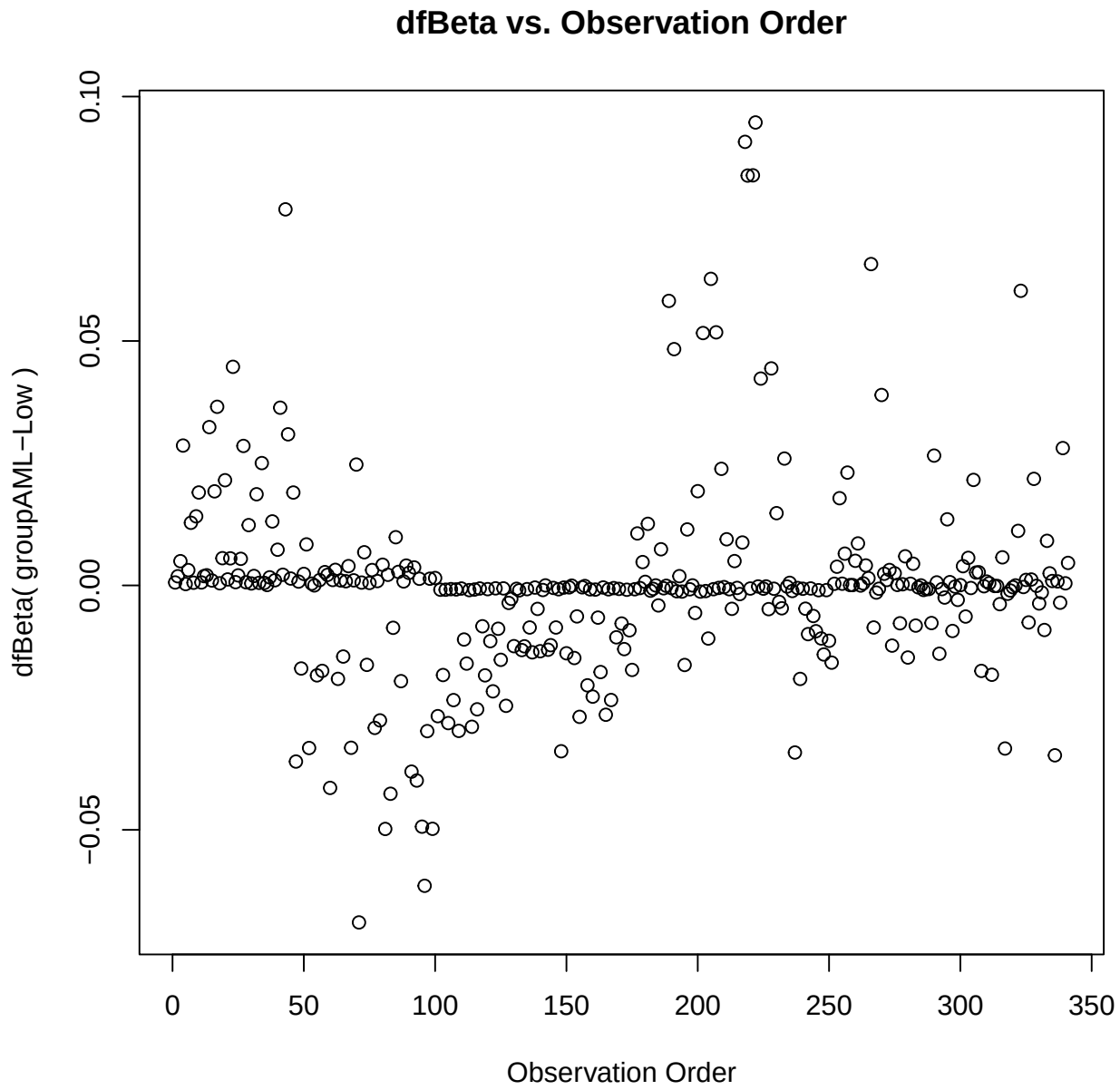
Patient 14 is in the medium-risk group, had a long survival time (censored), but early AGVHD and CGVHD, and no platelet recovery. Patients 100 and 103 are in the highest risk-group, had long survival times (censored), and early platelet recovery.

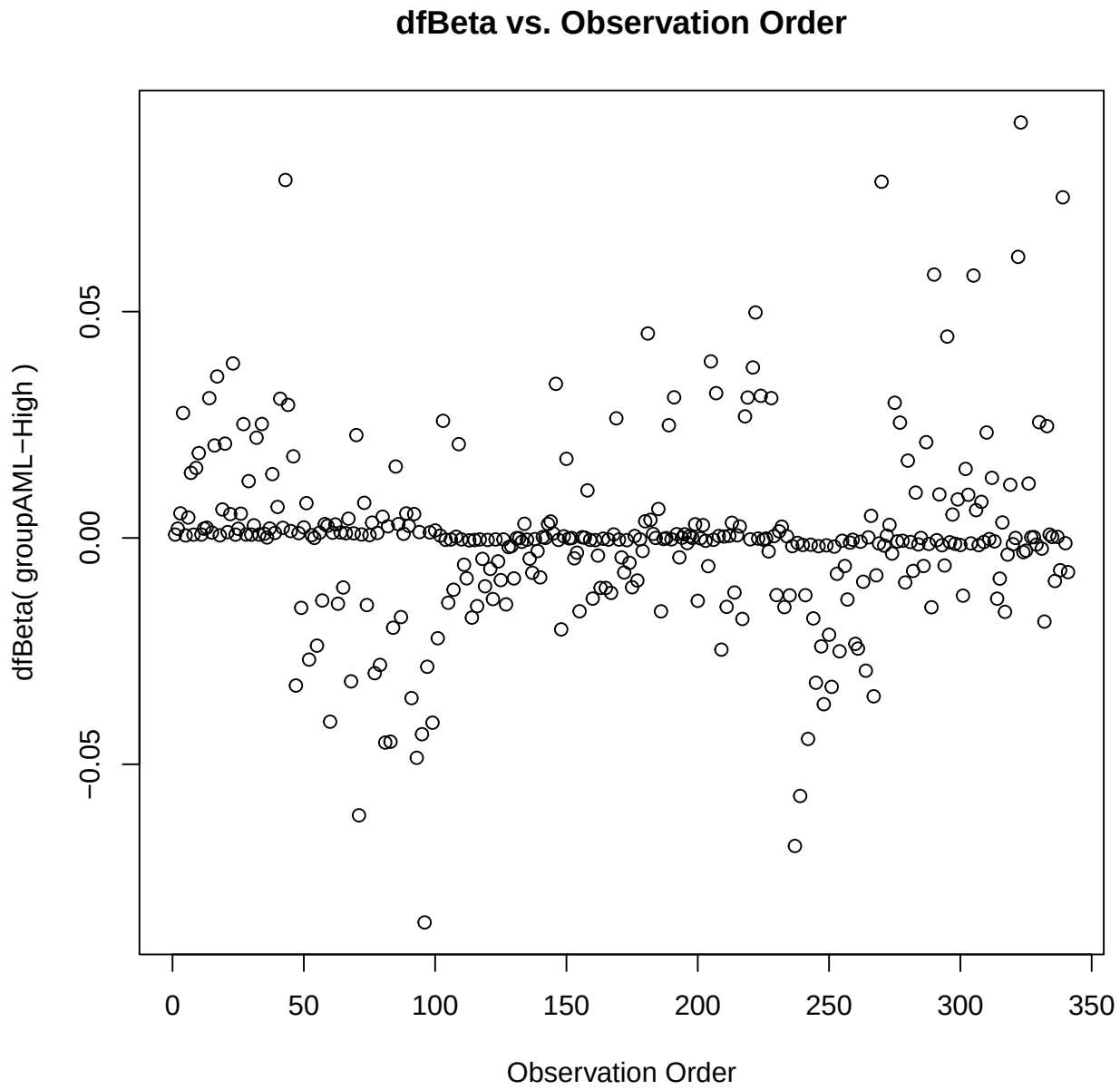


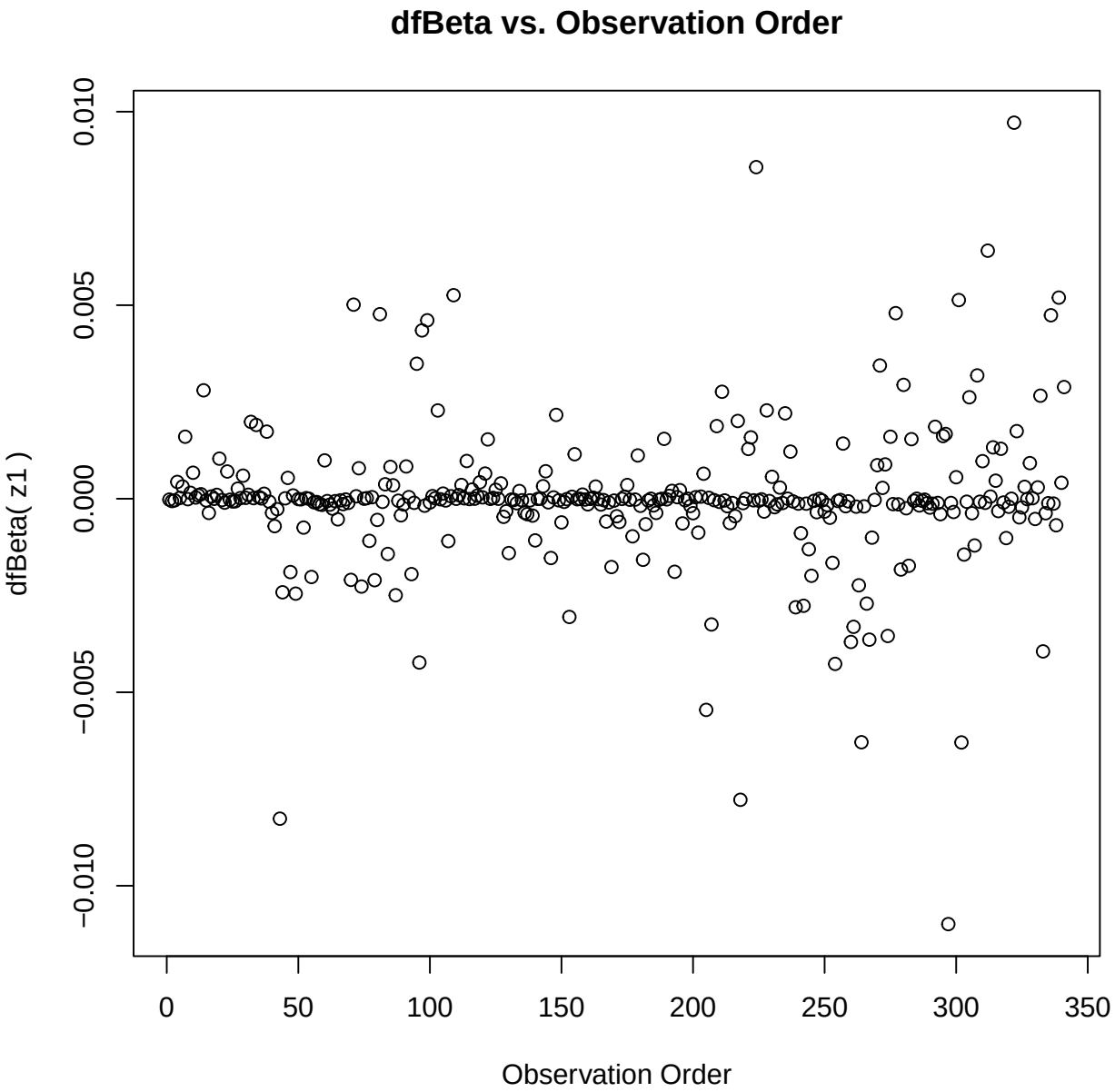
No unusualy low or high deviance residuals.

DFBETA

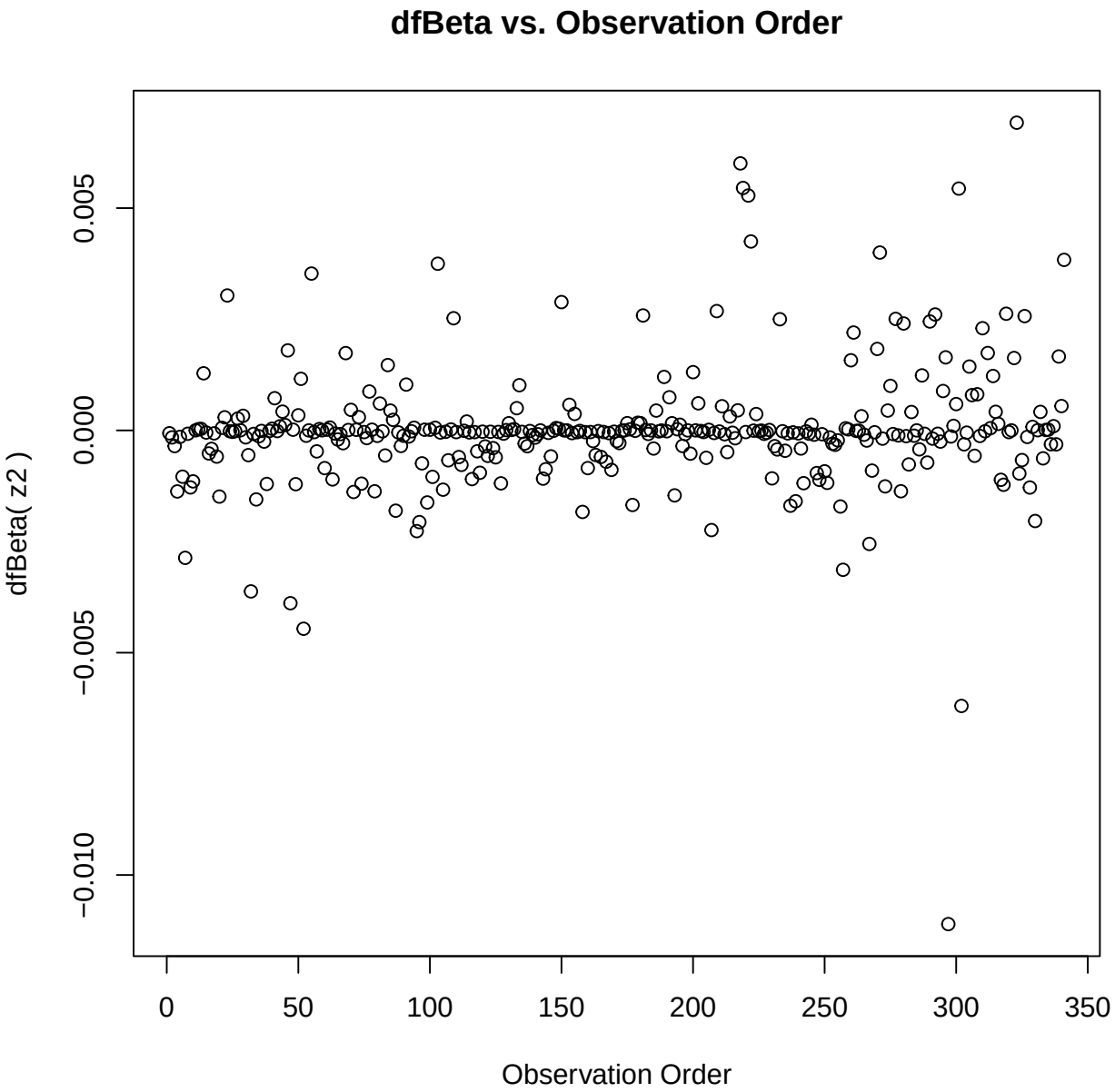
The `residuals = dfbeta` matrix is 341 by 7 with rows corresponding with patient×intervals and columns corresponding to the coefficients `groupAML-Low`, `groupAML-High`, `z1`, `z2`, `z8`, `precovery`, `z1:z2`.


dfBeta vs. Observation Order
dfBeta(groupAML−Low)
0.10
0.05
0.00
−0.05
0
50
100
150
200
250
300
350
Observation Order

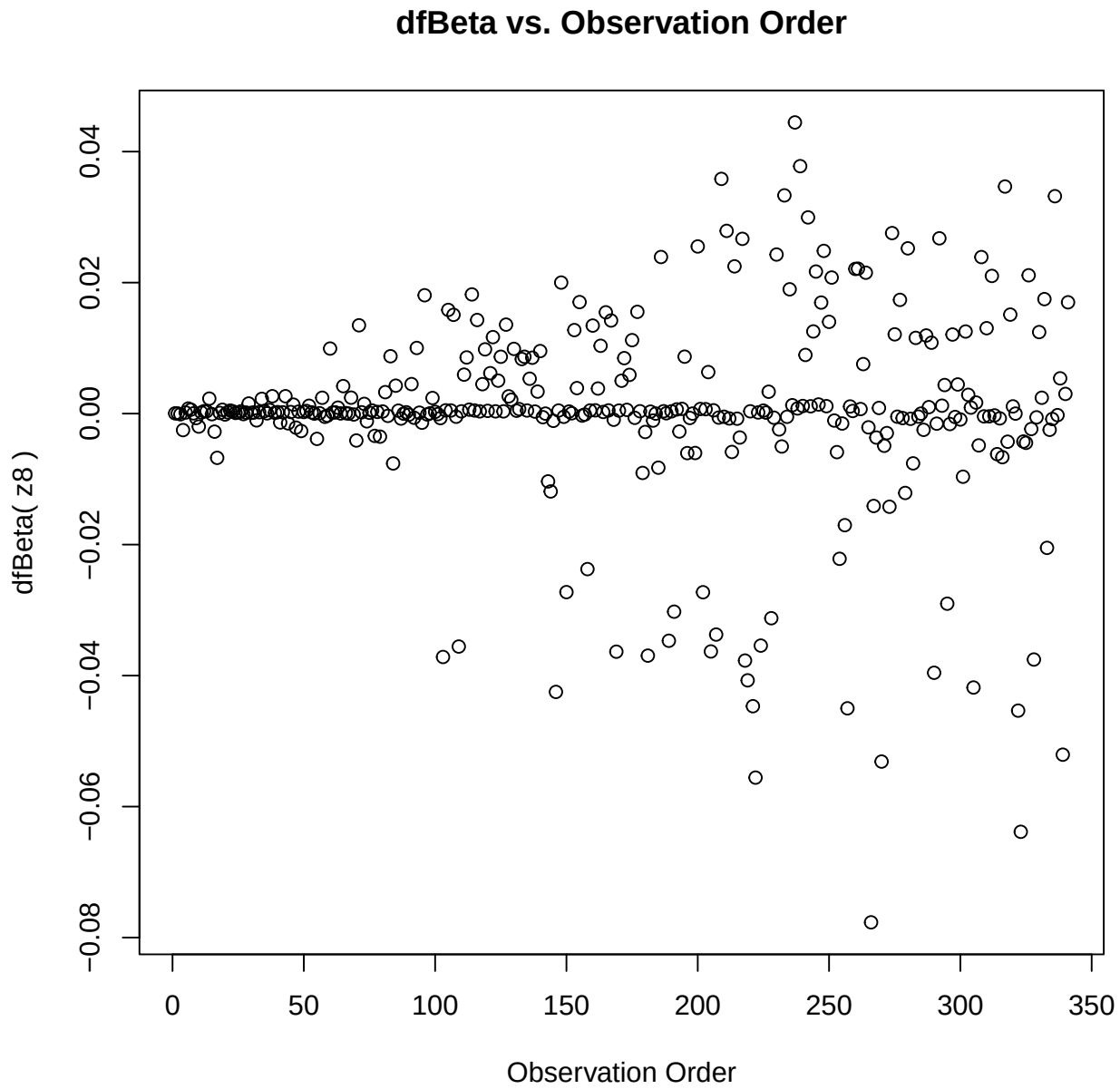

dfBeta vs. Observation Order
dfBeta(groupAML−High)
0.05
0.00
−0.05
0
50
100
150
200
250
300
350
Observation Order

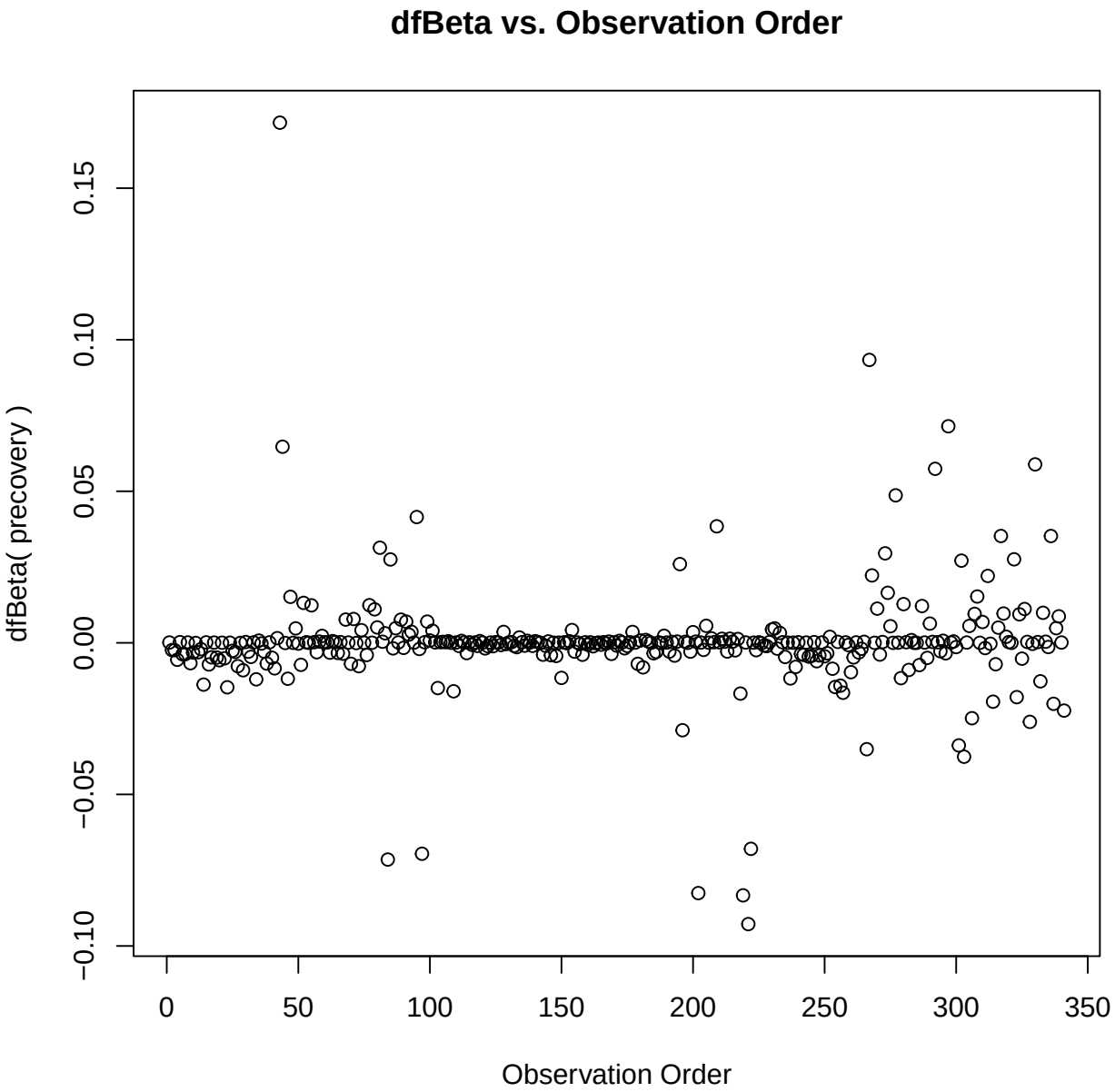


Observations 88 and 128 high and 14, 84, and 116 low.

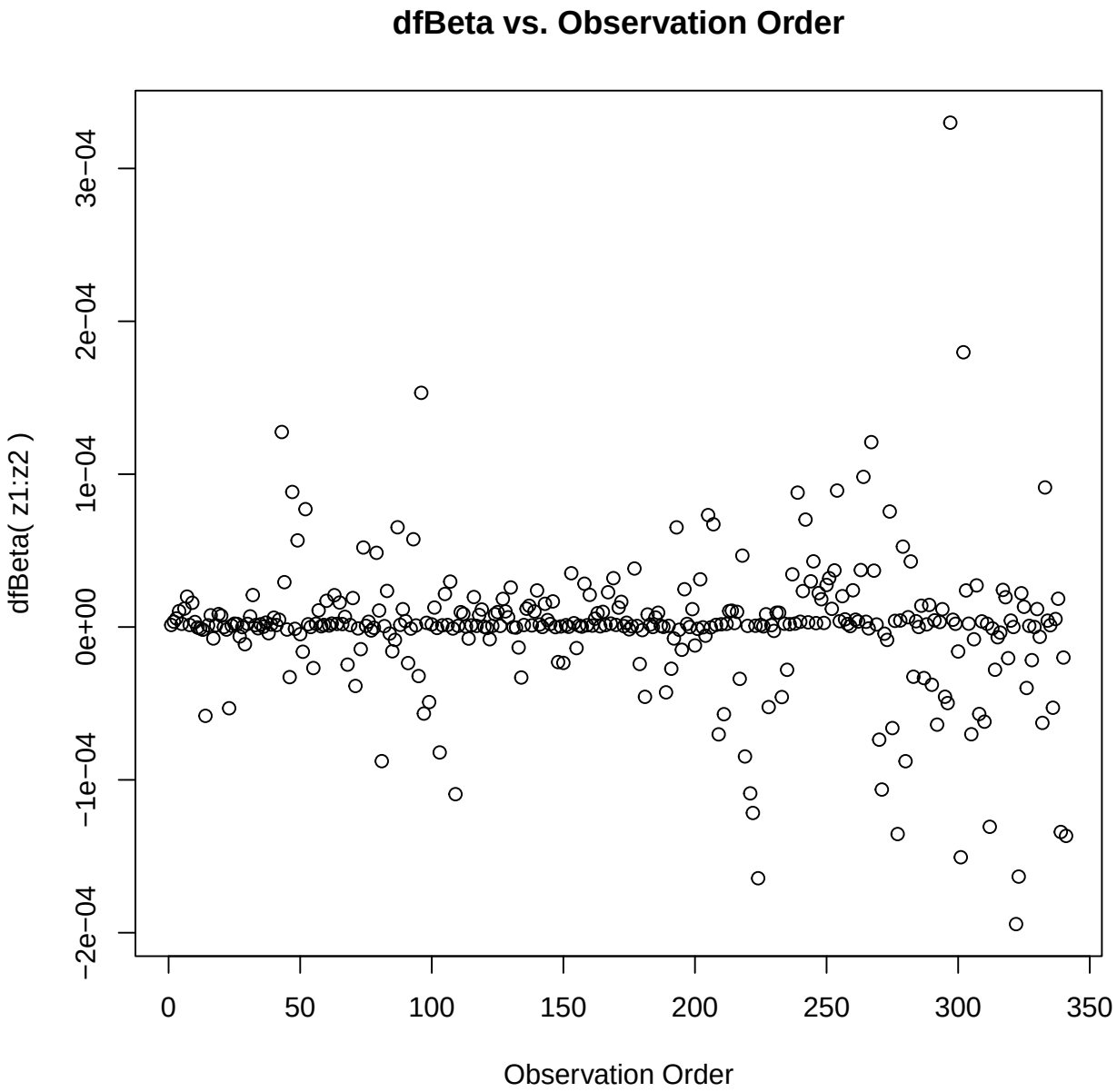


Observations 84 and 129 high and 116 and 118 low.


dfBeta vs. Observation Order
dfBeta(z8)
0.04
0.02
0.00
−0.02
−0.04
−0.06
−0.08
0
50
100
150
200
250
300
350
Observation Order



Observation 14 high and 30, 36, 77, 85, 86, 87 low.



Observation 116 high.

```
> bmt1[c(116,118,128,129,84:88),imp.vars1]
     id    group   t1  t2 d1 d2 d3   ta da   tc dc tp dp z1 z2 z8
116 116 AML-High   93  47  1  1  1   93  0   93  0 28  1  7  2  1
118 118 AML-High  183 183  1  0  1  183  0  130  1 21  1 11  7  1
128 128 AML-High   74  74  1  0  1   29  1   74  0 24  1 41 29  0
129 129 AML-High   16  16  1  0  1   16  0   16  0 16  0 27 36  0
84   84  AML-Low   10  10  1  0  1   10  0   10  0 10  0 34 54  0
85   85  AML-Low   53  53  1  0  1   53  0   53  0 53  0 33 41  0
86   86  AML-Low   80  80  1  0  1   10  1   80  0 80  0 30 35  0
87   87  AML-Low   35  35  1  0  1   35  0   35  0 35  0 23 25  0
88   88  AML-Low 1499 248  0  1  1 1499  0 1499  0  9  1 35 18  0
```

Observations 116 and 118 have very young patient/donor combinations. These are extreme in the linear function of age and especially in the product. Observations 128 and 129 are in AML-High but no z8 FAB extra risk and have very early deaths. Observations 84–87 have the lowest risk group, AML-Low + no extra FAB risk, but early deaths. Observation 88 is a low risk (of progression-free survival) with early platelet recovery but relapsed at a long interval.

This kind of analysis can identify errors. It can identify problems like use of linear age. Some outliers are explicable from unusual predictive values. The plots we use can identify these unusual combinations much more easily than just staring at the data.

This kind of analysis is even more important in early stages of the project because it can identify specious observations as well as influential ones.

Recurrent Events

- Sometimes an appropriate analysis requires consideration of recurrent events.
- A patient with arthritis may have more than one flareup. The same is true of many recurring-remitting diseases.
- In this case, we have more than one line in the dataframe, but each line may have an event.
- We have to use a "robust" variance estimator to account for correlation of time-to-events within a patient.

Bladder Cancer Data Set

The bladder cancer dataset from Kleinbaum and Klein contains recurrent event outcome information for eighty-six cancer patients followed for the recurrence of bladder cancer tumor after transurethral surgical excision (Byar and Green 1980). The exposure of interest is the effect of the drug treatment of thiotepa. Control variables are the initial number and initial size of tumors. The data layout is suitable for a counting processes approach.
This drug is still a possible choice for some patients. Another therapeutic choice is Bacillus Calmette-Guerin (BCG), a live bacterium related to cow tuberculosis.

Bladder Cancer Data Set

Variable	Definition
id	Patient unique ID
status	for each time interval
	1 = recurred
	2 = censored
interval	1 = first recurrence, etc.
intime	tstop − tstart (all times in months)
tstart	start of interval
tstop	end of interval
tx	treatment code, 1 = thiotepa
num	number of initial tumors
size	size of initial tumors (cm)

- There are 85 patients and 190 lines in the dataframe, meaning that many patients have more than one line.
- Patient 1 with 0 observation time was removed.
- Of the 85 patients, 47 had at least one recurrence and 38 had none.
- 18 patients had exactly one recurrence.
- There were up to 4 recurrences in a patient.
- Of the 190 intervals, 112 terminated with a recurrence and 78 were censored.

- Different intervals for the same patient are correlated.
- Of the 85 patients, 47 had at least one recurrence and 38 had none.
- Of the 190 intervals, 112 terminated with a recurrence and 78 were censored.
- Is the effective sample size 47 or 112? This might narrow confidence intervals by as much as a factor of $\sqrt{112/47} = 1.54$
- What happens if I have 5 treatment and 5 control values and want to do a t-test and I then duplicate the 10 values as if the sample size was 20? This falsely narrows confidence intervals by a factor of $\sqrt{2} = 1.41$.

	id	status	interval	intime	tstart	tstop	tx	num	size
2	2	0	1	1	0	1	0	1	3
3	3	0	1	4	0	4	0	2	1
...									
6	6	1	1	6	0	6	0	4	1
7	6	0	2	4	6	10	0	4	1
...									
10	9	1	1	5	0	5	0	1	3
11	9	0	2	13	5	18	0	1	3
...									
12	10	1	1	12	0	12	0	1	1
13	10	1	2	4	12	16	0	1	1
14	10	0	3	2	16	18	0	1	1
...									
22	14	1	1	3	0	3	0	3	1
23	14	1	2	6	3	9	0	3	1
24	14	1	3	12	9	21	0	3	1
25	14	0	4	2	21	23	0	3	1
...									
26	15	1	1	7	0	7	0	2	3
27	15	1	2	3	7	10	0	2	3
28	15	1	3	6	10	16	0	2	3
29	15	1	4	8	16	24	0	2	3

```
require(survival)
vars <- c("id","status","interval","intime","tstart","tstop","tx","num","size")
bladder <- read.table("bladder.dat",header=F,col.names=vars)
bladder <- bladder[-1,]  #remove subject with 0 observation time

#bladder.dat from Kleinbaum and Klein with lines before and after data removed

bladder.surv <- with(bladder,Surv(time=tstart,time2=tstop,event=status,
                  type="counting"))

bladder.cox1 <- coxph(bladder.surv~tx+num+size,data=bladder)
#biased variance co-variance matrix

bladder.cox2 <- coxph(bladder.surv~tx+num+size+cluster(id),data=bladder)
#unbiased though this reduces power

bladder.cox3 <- coxph(bladder.surv~tx+num+cluster(id),data=bladder)
#remove non-significant size variable
```

```
> summary(bladder.cox1)
Call:
coxph(formula = bladder.surv ~ tx + num + size, data = bladder)

  n= 190, number of events= 112

         coef exp(coef) se(coef)      z Pr(>|z|)
tx   -0.41164   0.66256  0.19989 -2.059 0.039466 *
num   0.16367   1.17782  0.04777  3.426 0.000611 ***
size -0.04108   0.95975  0.07029 -0.584 0.558967

> summary(bladder.cox2)
Call:
coxph(formula = bladder.surv ~ tx + num + size + cluster(id),
    data = bladder)

  n= 190, number of events= 112

         coef exp(coef) se(coef) robust se      z Pr(>|z|)
tx   -0.41164   0.66256  0.19989   0.24876 -1.655  0.09798 .
num   0.16367   1.17782  0.04777   0.05842  2.801  0.00509 **
size -0.04108   0.95975  0.07029   0.07421 -0.554  0.57991
```

```
> summary(bladder.cox1)
     exp(coef) exp(-coef) lower .95 upper .95
tx      0.6626      1.509    0.4478    0.9803
num     1.1778      0.849    1.0726    1.2934
size    0.9598      1.042    0.8362    1.1015
```

```
> summary(bladder.cox2)
     exp(coef) exp(-coef) lower .95 upper .95
tx      0.6626      1.509    0.4069     1.079
num     1.1778      0.849    1.0504     1.321
size    0.9598      1.042    0.8298     1.110
```

```
> summary(bladder.cox1)

Concordance= 0.624  (se = 0.03 )
Rsquare= 0.074   (max possible= 0.992 )
Likelihood ratio test= 14.66  on 3 df,   p=0.002127
Wald test            = 15.9   on 3 df,   p=0.001187
Score (logrank) test = 16.18  on 3 df,   p=0.001042

> summary(bladder.cox2)

Concordance= 0.624  (se = 0.03 )
Rsquare= 0.074   (max possible= 0.992 )
Likelihood ratio test= 14.66  on 3 df,   p=0.002127
Wald test            = 11.19  on 3 df,   p=0.01073
Score (logrank) test = 16.18  on 3 df,   p=0.001042,   Robust = 10.84  p=0.01263

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

```
> round(bladder.cox2$naive.var,4)
        [,1]    [,2]   [,3]
[1,]  0.0400 -0.0014 0.0000
[2,] -0.0014  0.0023 0.0007
[3,]  0.0000  0.0007 0.0049

> round(bladder.cox2$var,4)
        [,1]    [,2]    [,3]
[1,]  0.0619 -0.0026 -0.0004
[2,] -0.0026  0.0034  0.0013
[3,] -0.0004  0.0013  0.0055

> sqrt(with(bladder.cox2,diag(var)/diag(naive.var)))
[1] 1.244492 1.223092 1.055761
```

These are the ratios of correct confidence intervals to naive ones.

```
> summary(bladder.cox3)
Call:
coxph(formula = bladder.surv ~ tx + num + cluster(id), data = bladder)

  n= 190, number of events= 112

        coef exp(coef) se(coef) robust se      z Pr(>|z|)
tx  -0.41172   0.66251  0.20029   0.25153 -1.637  0.10166
num  0.17001   1.18531  0.04646   0.05636  3.016  0.00256 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

    exp(coef) exp(-coef) lower .95 upper .95
tx     0.6625     1.5094    0.4047     1.085
num    1.1853     0.8437    1.0613     1.324

Concordance= 0.623  (se = 0.029 )
Rsquare= 0.073   (max possible= 0.992 )
Likelihood ratio test= 14.31  on 2 df,   p=0.0007799
Wald test            = 10.24  on 2 df,   p=0.005969
Score (logrank) test = 15.81  on 2 df,   p=0.0003696,   Robust = 10.6  p=0.005001

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