NSWPCN Predictor Training

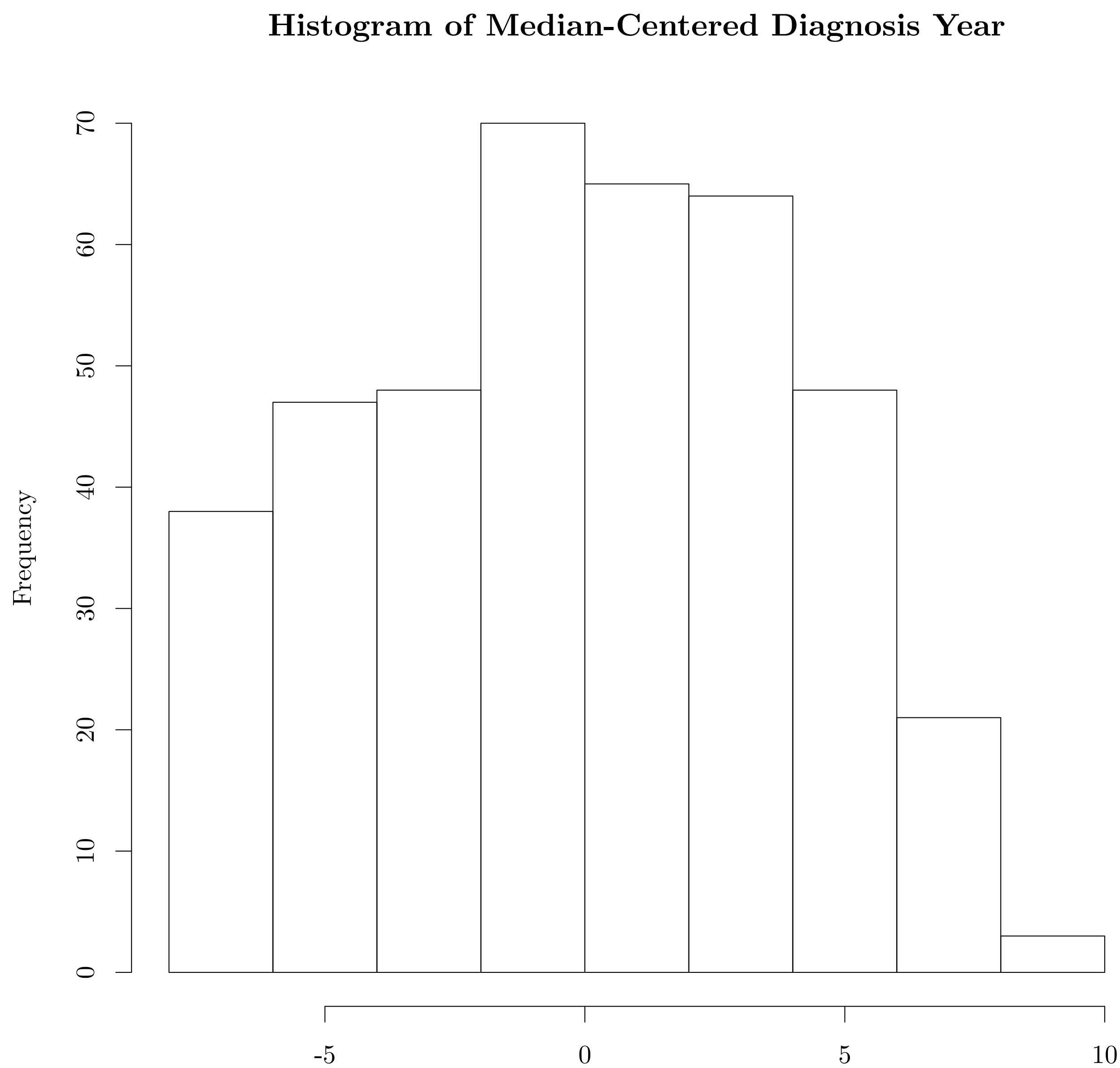
February 9, 2015

1 Preparation

```
library(survival)

## Loading required package:  splines

library(glmulti)

## Loading required package:  rJava
## Loading required package:  methods

library(flexsurv)
library(randomForestSRC)

## Loading required package:  parallel
##
##  randomForestSRC 1.5.5
##
##  Type rfsrc.news() to see new features, changes, and bug fixes.
##

library(reshape2)
library(plyr)
library(ggplot2)

library(MASS)
library(boot)

##
## Attaching package:  'boot'
##
## The following object is masked from 'package:survival':
##
##     aml

library(timeROC)

## Loading required package:  pec
## Loading required package:  mvtnorm
## Loading required package:  timereg

load("03_NSWPCN_subset.rda")

library(RColorBrewer)
pal = brewer.pal(5, "Dark2")
names(pal) = c("GG", "GG2", "CPH", "RSF", "KM0")
```

2 Cohort selection and transformation

```
x = data[,c("History.Diagnosis.Date", "Patient.Sex", "History.Diagnosis.AgeAt.Cent", "Path.LocationBody'
colnames(x) = c("DiagYearCent", "SexM", "AgeCent", "LocBody", "SizeCent", "Ca199", "A2", "A4")
x$SexM = x$Sex == "M"
x$Ca199 = x$Ca199 > 100
median(x$DiagYearCent)

## [1] "2002-01-13"

x$DiagYearCent = as.numeric((x$DiagYearCent - median(x$DiagYearCent)) / 365.25)
hist(x$DiagYearCent, main = "Histogram of Median-Centered Diagnosis Year", xlab = "")
```




```
y = Surv(as.numeric(data$History.Death.Date - data$History.Diagnosis.Date), data$History.DSDeath.Event)
# Note no surgery dates, though for almost all pts there were only a few days difference.
```

```r
temp = NA
temp = ls()
rm(list = temp[!(temp %in% c("x", "y", "pal"))])

sel = !is.na(y[,1]) & !is.na(y[,2]) & !is.na(x$A2) & !is.na(x$A4) & !is.na(x$LocBody)
x = x[sel,]
y = y[sel,]
rm(sel)

# Remove CA-19-9 measurements as they're mostly missing
x = x[,colnames(x) != "Ca199"]

data = as.data.frame(cbind(Time = y[,1], DSD = y[,2], x))
rm(x, y)
data$DSD = data$DSD == 1

# Remove long-survivors.  These are very likely to be misdiagnoses, or LTF.
nrow(data)

## [1] 256

data = data[data$Time < 3000,]
nrow(data)

## [1] 249
```

3 Data splitting

There's going to be an awful lot of model manipulation and black magic going on. Create a holdout validation set for final model comparison and selection.

```r
set.seed(20150201)
sel.val = sample.int(nrow(data), floor(nrow(data)/5))
sel.val = 1:nrow(data) %in% sel.val
mean(sel.val)

## [1] 0.1968

data.val = data[sel.val,,drop = FALSE]
data = data[!sel.val,,drop = FALSE]
nrow(data)

## [1] 200

nrow(data.val)

## [1] 49
```

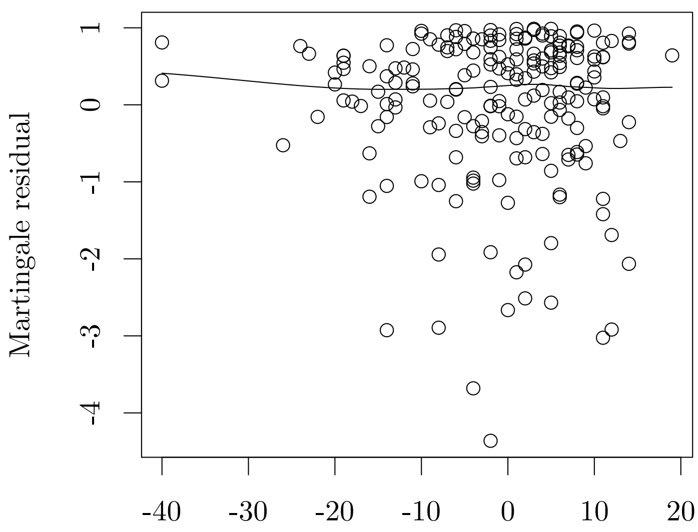
4 EDA

Use the CPH model as a convenient framework for EDA.

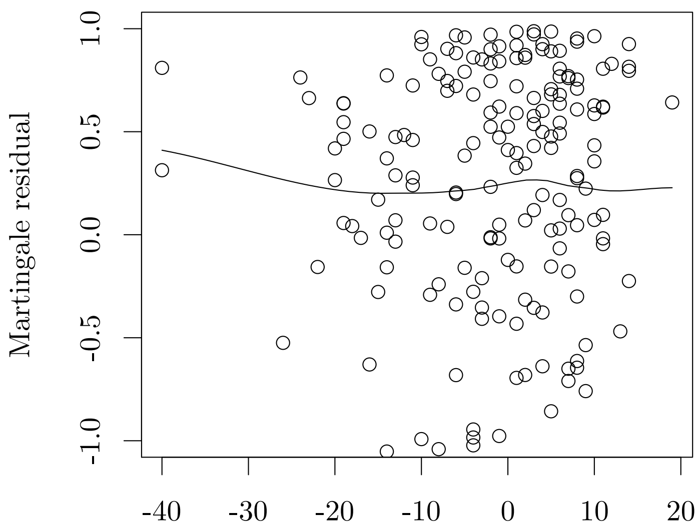
4.1 Functional form

Investigate functional form with martingale residuals.

```
fit.cph.NoAge = coxph(Surv(Time, DSD) ~ DiagYearCent + SexM + LocBody + SizeCent + A2 + A4, data = data)
scatter.smooth(data$AgeCent, resid(fit.cph.NoAge, type = "martingale"), xlab = "", ylab = "Martingale re
```

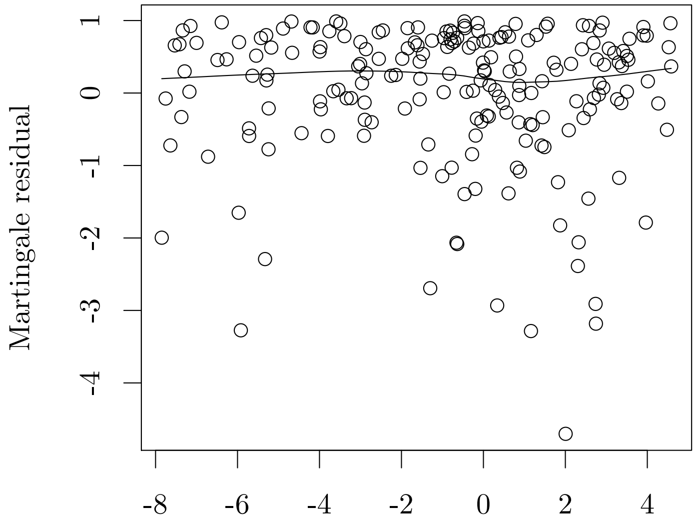



```
scatter.smooth(data$AgeCent, resid(fit.cph.NoAge, type = "martingale"), xlab = "", ylab = "Martingale re
```

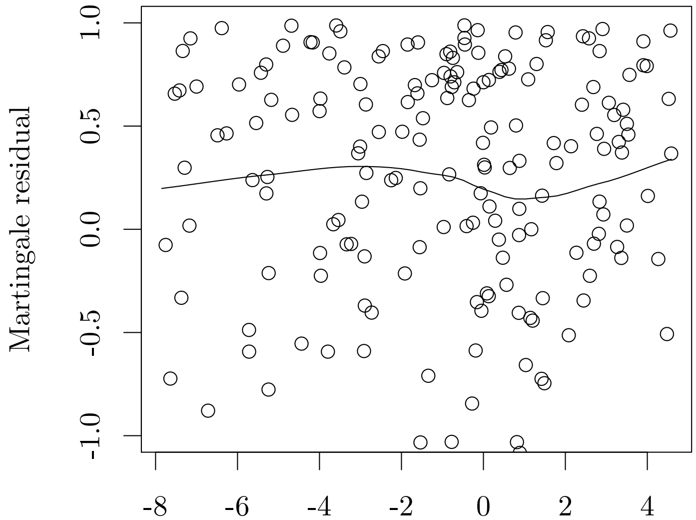


Close enough to linear.

```
fit.cph.NoDate = coxph(Surv(Time, DSD) ~ SexM + AgeCent + LocBody + SizeCent + A2 + A4, data = data)
scatter.smooth(data$DiagYearCent, resid(fit.cph.NoDate, type = "martingale"), xlab = "", ylab = "Marting
```

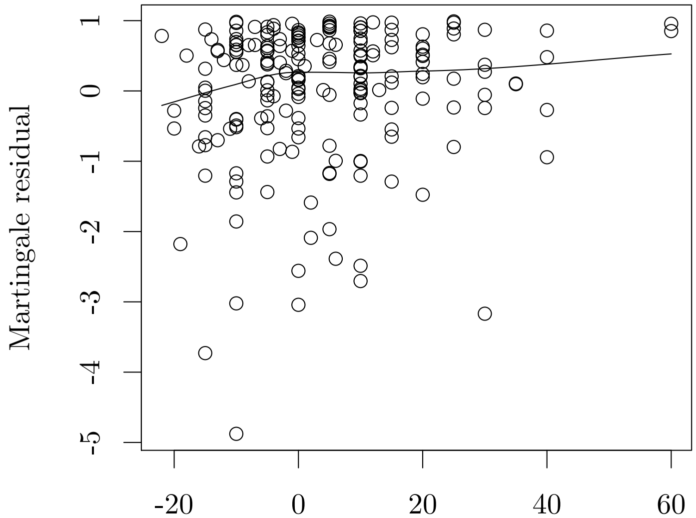



```
scatter.smooth(data$DiagYearCent, resid(fit.cph.NoDate, type = "martingale"), xlab = "", ylab = "Marting
```

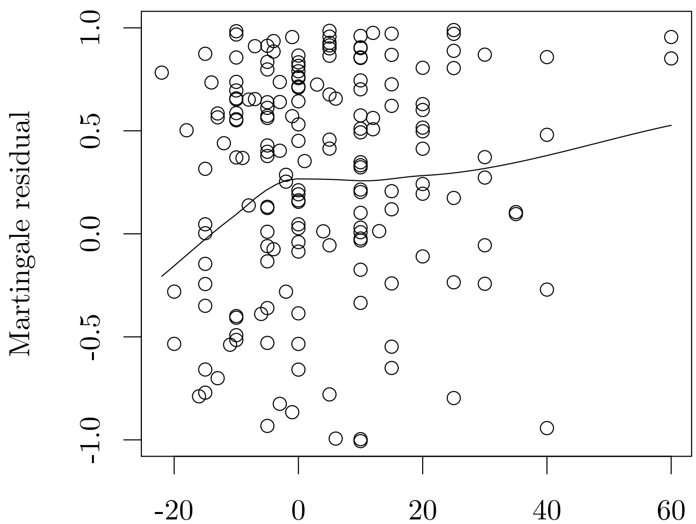


Doesn't appear to have much of an effect.

```
fit.cph.NoSize = coxph(Surv(Time, DSD) ~ DiagYearCent + SexM + AgeCent + LocBody + A2 + A4, data = data)
scatter.smooth(data$SizeCent, resid(fit.cph.NoSize, type = "martingale"), xlab = "", ylab = "Martingale
```




```
scatter.smooth(data$SizeCent, resid(fit.cph.NoSize, type = "martingale"), xlab = "", ylab = "Martingale
```



The size relationship appears to have a knee, close to size == 0, around which the relationship is approximately linear.

Model size as: $SizeCent + SizeCentI(SizeCent > 0) \equiv SizeCent + SizeCent_+$

```
data$SizePlus = pmax(data$SizeCent, 0)
data.val$SizePlus = pmax(data.val$SizeCent, 0)
```

4.2 PH assumption: full model

```
fit.cph = coxph(Surv(Time, DSD) ~ SexM + AgeCent + LocBody + SizeCent + SizePlus + A2 + A4, data = data)
cox.zph(fit.cph)

##                  rho    chisq      p
## SexMTRUE     0.17964  6.56115 0.0104
## AgeCent     -0.10574  2.40668 0.1208
## LocBodyTRUE -0.04856  0.37895 0.5382
## SizeCent     0.00231  0.00106 0.9740
```

```
## SizePlus    -0.01130  0.02666 0.8703
## A2TRUE      -0.03995  0.29907 0.5845
## A4TRUE      -0.08343  1.33308 0.2483
## GLOBAL            NA 13.17267 0.0680
```

```
fit.cph = coxph(Surv(Time, DSD) ~ strata(SexM) + AgeCent + LocBody + SizeCent + SizePlus + A2 + A4, data
cox.zph(fit.cph)
```

```
##                  rho   chisq      p
## AgeCent     -0.11339 2.78186 0.0953
## LocBodyTRUE -0.04618 0.34177 0.5588
## SizeCent     0.00662 0.00857 0.9262
## SizePlus    -0.01329 0.03588 0.8498
## A2TRUE      -0.04361 0.35772 0.5498
## A4TRUE      -0.07985 1.25354 0.2629
## GLOBAL            NA 6.03352 0.4194
```

Using a threshold of 0.1 for the CPH tests, sex is stuffing things up. Stratification by sex makes good sense, given known variation in survival between the sexes. It would have been possible to model this with a Sex:Age term in an AFT model, but given this is CPH, a baseline change is needed.

4.3 Date of diagnosis test

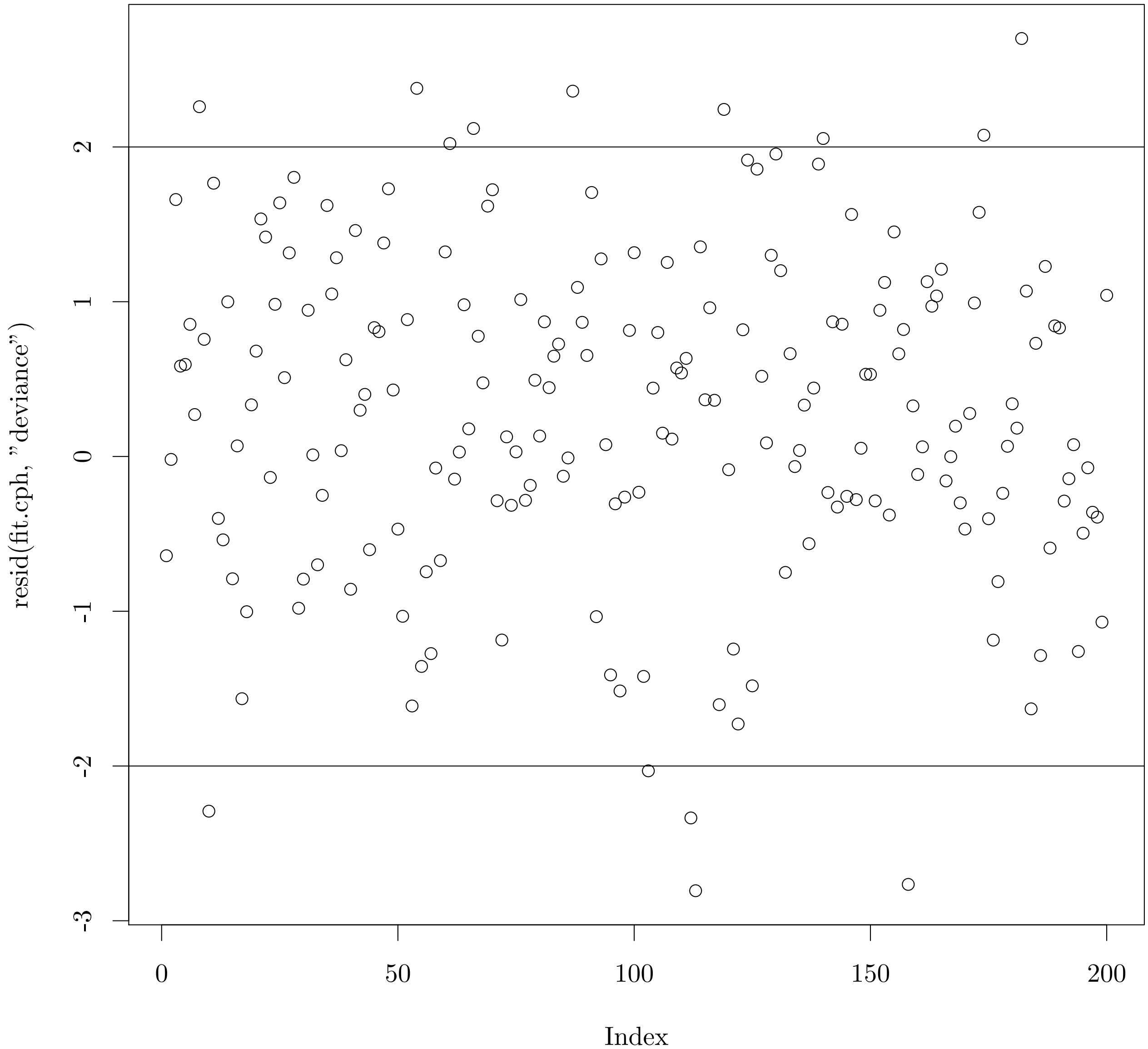
```
temp1 = coxph(Surv(Time, DSD) ~ strata(SexM) + AgeCent + LocBody + SizeCent + SizePlus + A2 + A4, data =
temp2 = coxph(Surv(Time, DSD) ~ strata(SexM) + AgeCent + LocBody + SizeCent + SizePlus + A2 + A4 + DiagY
anova(temp1, temp2)
```

```
## Analysis of Deviance Table
##  Cox model: response is  Surv(Time, DSD)
##  Model 1: ~ strata(SexM) + AgeCent + LocBody + SizeCent + SizePlus + A2 + A4
##  Model 2: ~ strata(SexM) + AgeCent + LocBody + SizeCent + SizePlus + A2 + A4 + DiagYearCent
##   loglik Chisq Df P(>|Chi|)
## 1   -682
## 2   -682  0.86  1      0.35
```

Not significant; good.

4.4 Outliers

```
plot(resid(fit.cph, "deviance"))
abline(h = c(-2, 2))
```




```
data$devresid = resid(fit.cph, type = "deviance")
temp = data[abs(data$devresid) >= 2,]
temp[order(temp$Time),]
```

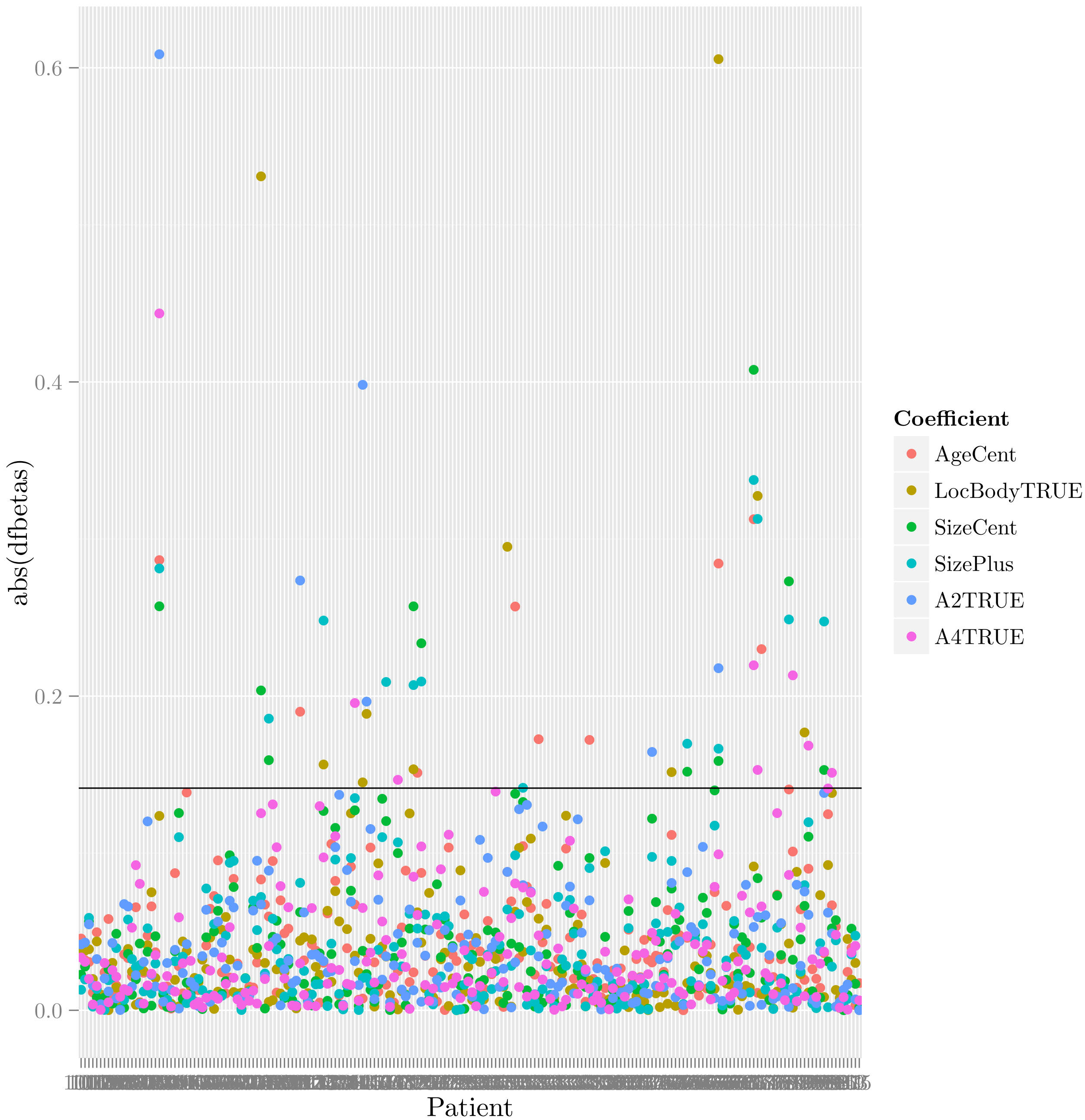
```
##              Time   DSD DiagYearCent  SexM AgeCent LocBody SizeCent    A2
## NSWPCN_315     26  TRUE      -0.4627  TRUE       3    TRUE       25  TRUE
## NSWPCN_374     63  TRUE      -3.5921  TRUE       5   FALSE      -10  TRUE
## NSWPCN_1177    68  TRUE      -4.6845 FALSE       1   FALSE        5 FALSE
## NSWPCN_333     90  TRUE      -0.1396  TRUE      10    TRUE       25 FALSE
## NSWPCN_779     96  TRUE       2.9076 FALSE       3   FALSE       12 FALSE
## NSWPCN_1165    97  TRUE       4.5640  TRUE      -6   FALSE       15 FALSE
## NSWPCN_324    100  TRUE       1.5688 FALSE     -10   FALSE       10 FALSE
## NSWPCN_1017   100  TRUE      -3.4908 FALSE      -5   FALSE        5 FALSE
## NSWPCN_125    103  TRUE      -6.3847 FALSE      -2   FALSE      -10 FALSE
## NSWPCN_133   1304 FALSE      -5.3279  TRUE       5   FALSE        6 FALSE
## NSWPCN_655   1723  TRUE       0.3368  TRUE      11    TRUE       10 FALSE
## NSWPCN_1095  1836 FALSE       2.7461  TRUE      -8   FALSE        0  TRUE
```

```
## NSWPCN_668  2106 FALSE       2.0068 FALSE      -2    FALSE       30 FALSE
## NSWPCN_667  2415 FALSE       1.1608 FALSE     -14    FALSE      -15 FALSE
##                A4 SizePlus devresid
## NSWPCN_315   TRUE       25    2.379
## NSWPCN_374   TRUE        0    2.361
## NSWPCN_1177  TRUE        5    2.701
## NSWPCN_333   TRUE       25    2.119
## NSWPCN_779   TRUE       12    2.243
## NSWPCN_1165  TRUE       15    2.076
## NSWPCN_324   TRUE       10    2.022
## NSWPCN_1017  TRUE        5    2.055
## NSWPCN_125  FALSE        0    2.260
## NSWPCN_133   TRUE        6   -2.292
## NSWPCN_655   TRUE       10   -2.031
## NSWPCN_1095 FALSE        0   -2.765
## NSWPCN_668   TRUE       30   -2.806
## NSWPCN_667   TRUE        0   -2.336
```

```r
temp = resid(fit.cph, type = "dfbetas")
colnames(temp) = names(fit.cph$coefficients)
temp = melt(temp)
colnames(temp) = c("Patient", "Coefficient", "dfbetas")
temp$Patient = gsub("NSWPCN_", "", temp$Patient)
2/sqrt(nrow(data))              # The classic threshold for concern is 2/sqrt(n).
```

```
## [1] 0.1414
```

```r
ggplot(temp, aes(y = abs(dfbetas), x = Patient, col = Coefficient)) + geom_point() + geom_hline(yinterce
```




```r
sort(apply(abs(resid(fit.cph, type = "dfbetas")), 1, max), decreasing = TRUE)

## NSWPCN_1095  NSWPCN_655 NSWPCN_1182  NSWPCN_667  NSWPCN_144  NSWPCN_668
##    0.608593    0.605502    0.530888    0.407716    0.398158    0.327457
##  NSWPCN_310 NSWPCN_1212  NSWPCN_784  NSWPCN_192  NSWPCN_313 NSWPCN_1253
##    0.295058    0.273608    0.273058    0.257139    0.257021    0.248100
##  NSWPCN_799  NSWPCN_195  NSWPCN_674  NSWPCN_788  NSWPCN_154  NSWPCN_145
##    0.247494    0.233634    0.229905    0.213230    0.209001    0.196503
##  NSWPCN_142 NSWPCN_1186  NSWPCN_794  NSWPCN_320  NSWPCN_349  NSWPCN_639
##    0.195590    0.185655    0.176847    0.172566    0.172120    0.169714
##  NSWPCN_795  NSWPCN_374  NSWPCN_445  NSWPCN_802  NSWPCN_194  NSWPCN_163
##    0.168470    0.164443    0.151616    0.151183    0.151178    0.146712
##  NSWPCN_316  NSWPCN_801  NSWPCN_654  NSWPCN_307 NSWPCN_1147  NSWPCN_135
```

```
##    0.141622    0.141212    0.139981    0.139227    0.138642    0.137147
##  NSWPCN_152 NSWPCN_1187  NSWPCN_317  NSWPCN_125  NSWPCN_315 NSWPCN_1145
##    0.134638    0.131048    0.130750    0.129984    0.127999    0.125567
##  NSWPCN_777  NSWPCN_141  NSWPCN_182  NSWPCN_333  NSWPCN_337 NSWPCN_1083
##    0.125524    0.125370    0.125309    0.123862    0.121565    0.120246
##  NSWPCN_321  NSWPCN_133 NSWPCN_1453  NSWPCN_269  NSWPCN_318  NSWPCN_296
##    0.116968    0.116092    0.115398    0.111797    0.109297    0.108507
##  NSWPCN_335  NSWPCN_132  NSWPCN_647 NSWPCN_1188  NSWPCN_354 NSWPCN_1168
##    0.107926    0.105932    0.104040    0.103771    0.101211    0.098668
##  NSWPCN_305 NSWPCN_1160 NSWPCN_1179 NSWPCN_1169  NSWPCN_151 NSWPCN_1071
##    0.096994    0.095425    0.095199    0.095118    0.093520    0.092355
##  NSWPCN_331  NSWPCN_267  NSWPCN_138  NSWPCN_276   NSWPCN_17  NSWPCN_789
##    0.091974    0.089754    0.089265    0.089051    0.088914    0.088141
## NSWPCN_1141 NSWPCN_1072  NSWPCN_257  NSWPCN_664 NSWPCN_1189 NSWPCN_1155
##    0.087304    0.080513    0.080219    0.079865    0.079050    0.077529
##  NSWPCN_303 NSWPCN_1088  NSWPCN_200  NSWPCN_798  NSWPCN_663 NSWPCN_1158
##    0.075390    0.075103    0.074720    0.073281    0.072910    0.072736
##  NSWPCN_364 NSWPCN_1177  NSWPCN_312  NSWPCN_375  NSWPCN_324 NSWPCN_1028
##    0.070600    0.069807    0.069320    0.068858    0.067746    0.067618
## NSWPCN_1183  NSWPCN_657 NSWPCN_1031 NSWPCN_1198  NSWPCN_637 NSWPCN_1222
##    0.067251    0.066525    0.066068    0.065549    0.065423    0.065069
##  NSWPCN_336 NSWPCN_1157  NSWPCN_790    NSWPCN_4   NSWPCN_13  NSWPCN_665
##    0.064764    0.064604    0.064267    0.063796    0.063341    0.062318
## NSWPCN_1213  NSWPCN_648  NSWPCN_636  NSWPCN_281   NSWPCN_20  NSWPCN_347
##    0.062295    0.062010    0.061429    0.061039    0.060975    0.060948
##  NSWPCN_769  NSWPCN_268 NSWPCN_1167 NSWPCN_1017 NSWPCN_1022  NSWPCN_804
##    0.060504    0.059817    0.059415    0.058835    0.058110    0.057379
##  NSWPCN_661  NSWPCN_779  NSWPCN_640  NSWPCN_164 NSWPCN_1066  NSWPCN_813
##    0.056829    0.055459    0.052953    0.052737    0.052580    0.051966
## NSWPCN_1165  NSWPCN_643 NSWPCN_1019 NSWPCN_1190  NSWPCN_376 NSWPCN_1026
##    0.051250    0.049904    0.049674    0.049013    0.048787    0.048713
##  NSWPCN_282  NSWPCN_815 NSWPCN_1089  NSWPCN_308  NSWPCN_284   NSWPCN_10
##    0.048235    0.047547    0.047147    0.046789    0.046011    0.045675
##  NSWPCN_811  NSWPCN_372 NSWPCN_1146 NSWPCN_1016 NSWPCN_1023  NSWPCN_653
##    0.045397    0.044821    0.043795    0.043085    0.042856    0.041802
##  NSWPCN_283  NSWPCN_270  NSWPCN_162  NSWPCN_306  NSWPCN_662  NSWPCN_363
##    0.041402    0.040900    0.040385    0.040045    0.039252    0.038086
## NSWPCN_1227  NSWPCN_369  NSWPCN_770 NSWPCN_1153  NSWPCN_796  NSWPCN_332
##    0.038003    0.035265    0.034983    0.034467    0.032432    0.032422
##  NSWPCN_373 NSWPCN_1148  NSWPCN_272 NSWPCN_1075  NSWPCN_149 NSWPCN_1139
##    0.032184    0.031468    0.031391    0.030774    0.030526    0.030104
## NSWPCN_1021  NSWPCN_352 NSWPCN_1171  NSWPCN_325  NSWPCN_362  NSWPCN_360
##    0.030095    0.030008    0.029728    0.027767    0.026652    0.026516
##  NSWPCN_309  NSWPCN_646  NSWPCN_348   NSWPCN_36  NSWPCN_384  NSWPCN_256
##    0.026444    0.026071    0.025947    0.025334    0.024514    0.024298
##  NSWPCN_143  NSWPCN_319  NSWPCN_366  NSWPCN_358  NSWPCN_775  NSWPCN_370
##    0.023931    0.022670    0.022058    0.021289    0.021157    0.021082
## NSWPCN_1150 NSWPCN_1175  NSWPCN_797 NSWPCN_1152  NSWPCN_350 NSWPCN_1018
##    0.020619    0.020474    0.020089    0.019960    0.019915    0.019881
## NSWPCN_1211  NSWPCN_656  NSWPCN_781    NSWPCN_9 NSWPCN_1207  NSWPCN_658
##    0.018856    0.018714    0.018620    0.017311    0.017230    0.016422
## NSWPCN_1176 NSWPCN_1136  NSWPCN_807 NSWPCN_1140 NSWPCN_1027 NSWPCN_1173
##    0.015073    0.014981    0.013112    0.012942    0.011384    0.011157
## NSWPCN_1215  NSWPCN_638 NSWPCN_1020  NSWPCN_330  NSWPCN_157  NSWPCN_353
```

```
##    0.010917    0.010175    0.010018    0.009944    0.009637    0.008259
##  NSWPCN_806  NSWPCN_136
##    0.007888    0.003489
```

```r
sum(apply(abs(resid(fit.cph, type = "dfbetas")), 1, max) > 2/sqrt(nrow(data)))
```

```
## [1] 31
```

```r
temp = resid(fit.cph, type = "dfbetas")
data$DFBETAS_max = apply(abs(temp), 1, max)
data$DFBETAS_vars = apply(abs(temp), 1, function(x) paste(attr(fit.cph$terms, "term.labels")[x > 2/sqrt(
temp = data[data$DFBETAS_max >= 2/sqrt(nrow(data)) | abs(data$devresid) >= 2,]
temp[order(temp$DFBETAS_max),]
```

```
##              Time   DSD DiagYearCent  SexM AgeCent LocBody SizeCent    A2
## NSWPCN_1165    97  TRUE       4.5640  TRUE      -6   FALSE       15 FALSE
## NSWPCN_779     96  TRUE       2.9076 FALSE       3   FALSE       12 FALSE
## NSWPCN_1017   100  TRUE      -3.4908 FALSE      -5   FALSE        5 FALSE
## NSWPCN_324    100  TRUE       1.5688 FALSE     -10   FALSE       10 FALSE
## NSWPCN_1177    68  TRUE      -4.6845 FALSE       1   FALSE        5 FALSE
## NSWPCN_133   1304 FALSE      -5.3279  TRUE       5   FALSE        6 FALSE
## NSWPCN_333     90  TRUE      -0.1396  TRUE      10    TRUE       25 FALSE
## NSWPCN_315     26  TRUE      -0.4627  TRUE       3    TRUE       25  TRUE
## NSWPCN_125    103  TRUE      -6.3847 FALSE      -2   FALSE      -10 FALSE
## NSWPCN_316   1698  TRUE      -0.6379 FALSE      -8   FALSE        5 FALSE
## NSWPCN_163    285  TRUE      -2.5544  TRUE      -2   FALSE        0 FALSE
## NSWPCN_194   1123  TRUE      -0.4600 FALSE     -16   FALSE       10 FALSE
## NSWPCN_802   1072  TRUE      -1.0048 FALSE     -14    TRUE       25 FALSE
## NSWPCN_445    114  TRUE       2.5763 FALSE      14    TRUE        5 FALSE
## NSWPCN_374     63  TRUE      -3.5921  TRUE       5   FALSE      -10  TRUE
## NSWPCN_795    128  TRUE       0.7858 FALSE       8   FALSE       -1 FALSE
## NSWPCN_639   1990  TRUE       2.3053 FALSE       1   FALSE        2 FALSE
## NSWPCN_349   1412  TRUE       0.6133  TRUE      11   FALSE        5 FALSE
## NSWPCN_320    804  TRUE      -1.3443 FALSE     -26   FALSE       15 FALSE
## NSWPCN_794    498 FALSE       0.8186 FALSE      -4    TRUE       10 FALSE
## NSWPCN_1186 1892  TRUE      -7.8494 FALSE       2   FALSE        0 FALSE
## NSWPCN_142   1691  TRUE       1.0349  TRUE       4   FALSE        0 FALSE
## NSWPCN_145    599  TRUE       2.5626  TRUE      -6    TRUE       15  TRUE
## NSWPCN_154    163  TRUE      -0.8022  TRUE      -2    TRUE       60 FALSE
## NSWPCN_788   2155 FALSE       1.8727 FALSE       5   FALSE      -10 FALSE
## NSWPCN_674    345  TRUE       3.5647  TRUE     -40   FALSE      -10 FALSE
## NSWPCN_195   1969  TRUE      -0.1889  TRUE       8   FALSE      -16 FALSE
## NSWPCN_799     70  TRUE      -0.4627 FALSE       4    TRUE       60  TRUE
## NSWPCN_1253 1044  TRUE      -5.9713  TRUE      -2   FALSE       40 FALSE
## NSWPCN_313   2521  TRUE      -1.2977 FALSE      12   FALSE      -10 FALSE
## NSWPCN_192    221  TRUE      -0.9008  TRUE      -3    TRUE      -22 FALSE
## NSWPCN_784   2701  TRUE       2.3244  TRUE      14   FALSE      -19 FALSE
## NSWPCN_1212 1053  TRUE       3.9644  TRUE      12   FALSE        2  TRUE
## NSWPCN_310   1093  TRUE       1.8207  TRUE       6    TRUE       -5 FALSE
## NSWPCN_668   2106 FALSE       2.0068 FALSE      -2   FALSE       30 FALSE
## NSWPCN_144   1206  TRUE       2.7379 FALSE       0   FALSE       10  TRUE
## NSWPCN_667   2415 FALSE       1.1608 FALSE     -14   FALSE      -15 FALSE
## NSWPCN_1182 2178  TRUE      -5.9192 FALSE      -4    TRUE      -10 FALSE
## NSWPCN_655   1723  TRUE       0.3368  TRUE      11    TRUE       10 FALSE
## NSWPCN_1095 1836 FALSE       2.7461  TRUE      -8   FALSE        0  TRUE
```

```
##                A4 SizePlus devresid DFBETAS_max
## NSWPCN_1165  TRUE       15   2.0759     0.05125
## NSWPCN_779   TRUE       12   2.2425     0.05546
## NSWPCN_1017  TRUE        5   2.0546     0.05884
## NSWPCN_324   TRUE       10   2.0218     0.06775
## NSWPCN_1177  TRUE        5   2.7006     0.06981
## NSWPCN_133   TRUE        6  -2.2919     0.11609
## NSWPCN_333   TRUE       25   2.1193     0.12386
## NSWPCN_315   TRUE       25   2.3787     0.12800
## NSWPCN_125  FALSE        0   2.2600     0.12998
## NSWPCN_316   TRUE        5  -1.3567     0.14162
## NSWPCN_163  FALSE        0   1.6387     0.14671
## NSWPCN_194   TRUE       10  -0.9807     0.15118
## NSWPCN_802  FALSE       25  -0.7492     0.15118
## NSWPCN_445   TRUE        5   1.7057     0.15162
## NSWPCN_374   TRUE        0   2.3607     0.16444
## NSWPCN_795  FALSE        0   1.8566     0.16847
## NSWPCN_639   TRUE        2  -1.4115     0.16971
## NSWPCN_349   TRUE        5  -1.1860     0.17212
## NSWPCN_320   TRUE       15  -0.6732     0.17257
## NSWPCN_794   TRUE       10  -1.4822     0.17685
## NSWPCN_1186  TRUE        0  -1.2863     0.18566
## NSWPCN_142  FALSE        0  -0.7903     0.19559
## NSWPCN_145   TRUE       15  -1.0032     0.19650
## NSWPCN_154   TRUE       60   1.4178     0.20900
## NSWPCN_788  FALSE        0  -1.7288     0.21323
## NSWPCN_674  FALSE        0   1.3545     0.22991
## NSWPCN_195  FALSE        0  -0.7925     0.23363
## NSWPCN_799   TRUE       60   1.9544     0.24749
## NSWPCN_1253  TRUE       40  -1.0703     0.24810
## NSWPCN_313   TRUE        0  -1.6122     0.25702
## NSWPCN_192   TRUE        0   1.8036     0.25714
## NSWPCN_784  FALSE        0  -1.2444     0.27306
## NSWPCN_1212  TRUE        2  -1.2600     0.27361
## NSWPCN_310   TRUE        0  -1.0326     0.29506
## NSWPCN_668   TRUE       30  -2.8060     0.32746
## NSWPCN_144   TRUE       10  -1.5652     0.39816
## NSWPCN_667   TRUE        0  -2.3357     0.40772
## NSWPCN_1182  TRUE        0  -1.6309     0.53089
## NSWPCN_655   TRUE       10  -2.0313     0.60550
## NSWPCN_1095 FALSE        0  -2.7651     0.60859
##                                                        DFBETAS_vars
## NSWPCN_1165
## NSWPCN_779
## NSWPCN_1017
## NSWPCN_324
## NSWPCN_1177
## NSWPCN_133
## NSWPCN_333
## NSWPCN_315
## NSWPCN_125
## NSWPCN_316                                                 SizeCent
## NSWPCN_163                                                       A2
## NSWPCN_194                                         strata(SexM),A4
```

```
## NSWPCN_802                                                   A2
## NSWPCN_445                                              AgeCent
## NSWPCN_374                                             SizePlus
## NSWPCN_795                                                   A2
## NSWPCN_639                                     LocBody,SizeCent
## NSWPCN_349                                      strata(SexM),A4
## NSWPCN_320                                      strata(SexM),A4
## NSWPCN_794                                              AgeCent
## NSWPCN_1186                                    LocBody,SizeCent
## NSWPCN_142                                                   A2
## NSWPCN_145                                     AgeCent,SizePlus
## NSWPCN_154                                             SizeCent
## NSWPCN_788                                                   A2
## NSWPCN_674                                      strata(SexM),A4
## NSWPCN_195                                     LocBody,SizeCent
## NSWPCN_799                                     LocBody,SizeCent
## NSWPCN_1253                                    AgeCent,SizeCent
## NSWPCN_313                                      strata(SexM),A4
## NSWPCN_192                             AgeCent,LocBody,SizeCent
## NSWPCN_784                                     LocBody,SizeCent
## NSWPCN_1212                            strata(SexM),SizePlus,A4
## NSWPCN_310                                              AgeCent
## NSWPCN_668                                  AgeCent,SizeCent,A2
## NSWPCN_144                                     AgeCent,SizePlus
## NSWPCN_667                   strata(SexM),LocBody,SizeCent,A2,A4
## NSWPCN_1182                                     AgeCent,LocBody
## NSWPCN_655  strata(SexM),AgeCent,LocBody,SizeCent,SizePlus,A4
## NSWPCN_1095      strata(SexM),LocBody,SizeCent,SizePlus,A2,A4
```

Remove points with deviance residuals ¿ 2.5, or DFBETAS ¿ 0.3.

```
data = data[data$DFBETAS_max <= 0.3 & abs(data$devresid) <= 2.5,]
fit.cph = coxph(Surv(Time, DSD) ~ strata(SexM) + AgeCent + LocBody + SizeCent + SizePlus + A2 + A4, data
```

4.5 EDA: Variable selection

```
nobs.coxph <<- function(obj, ...) sum(obj$y[,2])
set.seed(20150208)
fit.cph.as.bic = glmulti(Surv(Time, DSD) ~ strata(SexM) + AgeCent + LocBody + SizeCent + SizePlus + A2 +

## Initialization...
## TASK: Genetic algorithm in the candidate set.
## Initialization...
## Algorithm started...
##
## After 10 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+AgeCent+LocBody+SizePlus+A2+A4+SizePlus:LocBody
## Crit= 1330.74207056839
## Mean crit= 1371.32076533565
## Change in best IC: -8669.25792943161 / Change in mean IC: -8628.67923466435
##
## After 20 generations:
```

```
## Best model: Surv(Time,DSD)~1+strata(SexM)+AgeCent+LocBody+SizePlus+A2+A4
## Crit= 1325.88603606481
## Mean crit= 1366.36912555198
## Change in best IC: -4.85603450358053 / Change in mean IC: -4.95163978367555
##
## After 30 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+LocBody+SizePlus+A2+A4
## Crit= 1322.93145268634
## Mean crit= 1362.88668812839
## Change in best IC: -2.95458337846776 / Change in mean IC: -3.48243742359045
##
## After 40 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+SizePlus+A2+A4
## Crit= 1320.34394231168
## Mean crit= 1361.35541857129
## Change in best IC: -2.5875103746539 / Change in mean IC: -1.5312695570924
##
## After 50 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+SizePlus+A2+A4
## Crit= 1320.34394231168
## Mean crit= 1360.43151115989
## Change in best IC: 0 / Change in mean IC: -0.923907411405025
##
## After 60 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+SizePlus+A2+A4
## Crit= 1320.34394231168
## Mean crit= 1358.93358477098
## Change in best IC: 0 / Change in mean IC: -1.49792638890881
##
## After 70 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+SizePlus+A2+A4
## Crit= 1320.34394231168
## Mean crit= 1358.18268381061
## Change in best IC: 0 / Change in mean IC: -0.750900960367289
##
## After 80 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+SizePlus+A2+A4
## Crit= 1320.34394231168
## Mean crit= 1357.4704165225
## Change in best IC: 0 / Change in mean IC: -0.712267288116209
##
## After 90 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+SizePlus+A2+A4
## Crit= 1320.34394231168
## Mean crit= 1356.30395902444
## Change in best IC: 0 / Change in mean IC: -1.16645749806071
##
## After 100 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+SizePlus+A2+A4
## Crit= 1320.34394231168
## Mean crit= 1354.62428938053
## Change in best IC: 0 / Change in mean IC: -1.67966964390826
##
```

```
## After 110 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+SizePlus+A2+A4
## Crit= 1320.34394231168
## Mean crit= 1354.19848400635
## Change in best IC: 0 / Change in mean IC: -0.42580537417507
##
## After 120 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+SizePlus+A2+A4
## Crit= 1320.34394231168
## Mean crit= 1353.0086120984
## Change in best IC: 0 / Change in mean IC: -1.1898719079486
##
## After 130 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+SizePlus+A2+A4
## Crit= 1320.34394231168
## Mean crit= 1352.69883347707
## Change in best IC: 0 / Change in mean IC: -0.309778621338637
##
## After 140 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+SizePlus+A2+A4
## Crit= 1320.34394231168
## Mean crit= 1352.10200701269
## Change in best IC: 0 / Change in mean IC: -0.596826464376818
##
## After 150 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+SizePlus+A2+A4
## Crit= 1320.34394231168
## Mean crit= 1351.26232706556
## Change in best IC: 0 / Change in mean IC: -0.839679947126342
##
## After 160 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+SizePlus+A2+A4
## Crit= 1320.34394231168
## Mean crit= 1350.70620552145
## Change in best IC: 0 / Change in mean IC: -0.55612154411574
##
## After 170 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+SizePlus+A2+A4
## Crit= 1320.34394231168
## Mean crit= 1349.63847243742
## Change in best IC: 0 / Change in mean IC: -1.06773308402876
##
## After 180 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+SizePlus+A2+A4
## Crit= 1320.34394231168
## Mean crit= 1349.32765297463
## Change in best IC: 0 / Change in mean IC: -0.310819462789368
##
## After 190 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+SizePlus+A2+A4
## Crit= 1320.34394231168
## Mean crit= 1349.1123095406
## Change in best IC: 0 / Change in mean IC: -0.215343434027773
##
```

```
## After 200 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+SizePlus+A2+A4
## Crit= 1320.34394231168
## Mean crit= 1348.48239849382
## Change in best IC: 0 / Change in mean IC: -0.629911046780308
##
## After 210 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+SizePlus+A2+A4
## Crit= 1320.34394231168
## Mean crit= 1348.09675034221
## Change in best IC: 0 / Change in mean IC: -0.385648151610212
##
## After 220 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+SizePlus+A2+A4
## Crit= 1320.34394231168
## Mean crit= 1347.89881181048
## Change in best IC: 0 / Change in mean IC: -0.197938531734735
##
## After 230 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+SizePlus+A2+A4
## Crit= 1320.34394231168
## Mean crit= 1347.70964954015
## Change in best IC: 0 / Change in mean IC: -0.189162270326733
##
## After 240 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+SizePlus+A2+A4
## Crit= 1320.34394231168
## Mean crit= 1347.52376089243
## Change in best IC: 0 / Change in mean IC: -0.185888647723232
##
## After 250 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+SizePlus+A2+A4
## Crit= 1320.34394231168
## Mean crit= 1347.16933463019
## Change in best IC: 0 / Change in mean IC: -0.354426262238576
##
## After 260 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+SizePlus+A2+A4
## Crit= 1320.34394231168
## Mean crit= 1347.06409531869
## Change in best IC: 0 / Change in mean IC: -0.105239311493733
##
## After 270 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+SizePlus+A2+A4
## Crit= 1320.34394231168
## Mean crit= 1346.68513179375
## Change in best IC: 0 / Change in mean IC: -0.378963524947949
##
## After 280 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+SizePlus+A2+A4
## Crit= 1320.34394231168
## Mean crit= 1346.5079796517
## Change in best IC: 0 / Change in mean IC: -0.177152142049408
```

```
## Warning in fitter(X, Y, strats, offset, init, control, weights = weights, :  Loglik converged
before variable  11 ; beta may be infinite.
```

```
##
## After 290 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+SizePlus+A2+A4
## Crit= 1320.34394231168
## Mean crit= 1346.2390760551
## Change in best IC: 0 / Change in mean IC: -0.268903596595464
```

```
## Warning in fitter(X, Y, strats, offset, init, control, weights = weights, :  Loglik converged
before variable  13 ; beta may be infinite.
```

```
##
## After 300 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+SizePlus+A2+A4
## Crit= 1320.34394231168
## Mean crit= 1346.12370292987
## Change in best IC: 0 / Change in mean IC: -0.115373125235465
##
## After 310 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+SizePlus+A2+A4
## Crit= 1320.34394231168
## Mean crit= 1346.06901583365
## Change in best IC: 0 / Change in mean IC: -0.0546870962198227
##
## After 320 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+SizePlus+A2+A4
## Crit= 1320.34394231168
## Mean crit= 1345.88432093358
## Change in best IC: 0 / Change in mean IC: -0.184694900065324
##
## After 330 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+SizePlus+A2+A4
## Crit= 1320.34394231168
## Mean crit= 1345.8052620194
## Change in best IC: 0 / Change in mean IC: -0.0790589141829514
##
## After 340 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+SizePlus+A2+A4
## Crit= 1320.34394231168
## Mean crit= 1345.63375320605
## Change in best IC: 0 / Change in mean IC: -0.171508813342825
##
## After 350 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+SizePlus+A2+A4
## Crit= 1320.34394231168
## Mean crit= 1345.491678869
## Change in best IC: 0 / Change in mean IC: -0.142074337058375
##
## After 360 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+SizePlus+A2+A4
## Crit= 1320.34394231168
## Mean crit= 1345.39303926655
## Change in best IC: 0 / Change in mean IC: -0.0986396024438818
```

```
##
## After 370 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+SizePlus+A2+A4
## Crit= 1320.34394231168
## Mean crit= 1344.89485979836
## Change in best IC: 0 / Change in mean IC: -0.4981794681878
##
## After 380 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+SizePlus+A2+A4
## Crit= 1320.34394231168
## Mean crit= 1344.89485979836
## Change in best IC: 0 / Change in mean IC: 0
##
## After 390 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+SizePlus+A2+A4
## Crit= 1320.34394231168
## Mean crit= 1344.88633992423
## Change in best IC: 0 / Change in mean IC: -0.00851987413284405
##
## After 400 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+SizePlus+A2+A4
## Crit= 1320.34394231168
## Mean crit= 1344.76943861035
## Change in best IC: 0 / Change in mean IC: -0.116901313884
```

```
## Warning in fitter(X, Y, strats, offset, init, control, weights = weights, :  Loglik converged
before variable  11 ; beta may be infinite.
```

```
##
## After 410 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+SizePlus+A2+A4
## Crit= 1320.34394231168
## Mean crit= 1344.75613241449
## Change in best IC: 0 / Change in mean IC: -0.0133061958606504
```

```
## Warning in fitter(X, Y, strats, offset, init, control, weights = weights, :  Loglik converged
before variable  6 ; beta may be infinite.
```

```
##
## After 420 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+SizePlus+A2+A4
## Crit= 1320.34394231168
## Mean crit= 1344.72891317935
## Change in best IC: 0 / Change in mean IC: -0.0272192351376361
##
## After 430 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+SizePlus+A2+A4
## Crit= 1320.34394231168
## Mean crit= 1344.48179978082
## Change in best IC: 0 / Change in mean IC: -0.247113398531837
##
## After 440 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+SizePlus+A2+A4
## Crit= 1320.34394231168
## Mean crit= 1344.3654892015
```

```
## Change in best IC: 0 / Change in mean IC: -0.116310579320498
##
## After 450 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+SizePlus+A2+A4
## Crit= 1320.34394231168
## Mean crit= 1344.23789775616
## Change in best IC: 0 / Change in mean IC: -0.127591445334019
##
## After 460 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+SizePlus+A2+A4
## Crit= 1320.34394231168
## Mean crit= 1344.16223922723
## Change in best IC: 0 / Change in mean IC: -0.075658528930262
##
## After 470 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+SizePlus+A2+A4
## Crit= 1320.34394231168
## Mean crit= 1344.08041976008
## Change in best IC: 0 / Change in mean IC: -0.0818194671489891
##
## After 480 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+SizePlus+A2+A4
## Crit= 1320.34394231168
## Mean crit= 1343.98156317992
## Change in best IC: 0 / Change in mean IC: -0.0988565801608274
##
## After 490 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+SizePlus+A2+A4
## Crit= 1320.34394231168
## Mean crit= 1343.88916457382
## Change in best IC: 0 / Change in mean IC: -0.0923986061065989
##
## After 500 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+SizePlus+A2+A4
## Crit= 1320.34394231168
## Mean crit= 1343.84100789474
## Change in best IC: 0 / Change in mean IC: -0.0481566790765555
##
## After 510 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+SizePlus+A2+A4
## Crit= 1320.34394231168
## Mean crit= 1343.84100789474
## Change in best IC: 0 / Change in mean IC: 0
##
## After 520 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+SizePlus+A2+A4
## Crit= 1320.34394231168
## Mean crit= 1343.67859501471
## Change in best IC: 0 / Change in mean IC: -0.162412880032207
##
## After 530 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+SizePlus+A2+A4
## Crit= 1320.34394231168
```

```
## Mean crit= 1343.67859501471
## Change in best IC: 0 / Change in mean IC: 0
##
## After 540 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+SizePlus+A2+A4
## Crit= 1320.34394231168
## Mean crit= 1343.63117826687
## Change in best IC: 0 / Change in mean IC: -0.047416747836678

## Warning in fitter(X, Y, strats, offset, init, control, weights = weights, :  Loglik converged
before variable  2 ; beta may be infinite.

##
## After 550 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+SizePlus+A2+A4
## Crit= 1320.34394231168
## Mean crit= 1343.63117826687
## Change in best IC: 0 / Change in mean IC: 0
##
## After 560 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+SizePlus+A2+A4
## Crit= 1320.34394231168
## Mean crit= 1343.61386596246
## Change in best IC: 0 / Change in mean IC: -0.0173123044091881
##
## After 570 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+SizePlus+A2+A4
## Crit= 1320.34394231168
## Mean crit= 1343.56450517327
## Improvements in best and average IC have bebingo en below the specified goals.
## Algorithm is declared to have converged.
## Completed.

fit.cph.as.bic1 = glmulti(Surv(Time, DSD) ~ strata(SexM) + AgeCent + LocBody + SizeCent + SizePlus + A2

## Initialization...
## TASK: Exhaustive screening of candidate set.
## Fitting...
##
## After 50 models:
## Best model: Surv(Time,DSD)~1+A2+A4
## Crit= 1569.99720157408
## Mean crit= 1579.04206453807
##
## After 100 models:
## Best model: Surv(Time,DSD)~1+strata(SexM)+SizeCent+A2
## Crit= 1322.28966392719
## Mean crit= 1493.81514417481
##
## After 150 models:
## Best model: Surv(Time,DSD)~1+strata(SexM)+SizeCent+A2+A4
## Crit= 1319.12027767861
## Mean crit= 1416.9645603344
## Completed.
```

```
set.seed(20150208)
fit.cph.as.aicc = glmulti(Surv(Time, DSD) ~ strata(SexM) + AgeCent + LocBody + SizeCent + SizePlus + A2

## Initialization...
## TASK: Genetic algorithm in the candidate set.
## Initialization...
## Algorithm started...
##
## After 10 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+AgeCent+LocBody+SizeCent+SizePlus+A2+A4+SizeCent:AgeCent+Si
## Crit= 1313.81105306946
## Mean crit= 1325.17057373288
## Change in best IC: -8686.18894693054 / Change in mean IC: -8674.82942626712

## Warning in fitter(X, Y, strats, offset, init, control, weights = weights, :  Loglik converged
before variable  19 ; beta may be infinite.

##
## After 20 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+AgeCent+LocBody+SizeCent+SizePlus+A2+A4+SizeCent:AgeCent+A2
## Crit= 1308.19951621935
## Mean crit= 1320.61035775383
## Change in best IC: -5.611536850113 / Change in mean IC: -4.56021597904987

## Warning in fitter(X, Y, strats, offset, init, control, weights = weights, :  Loglik converged
before variable  7 ; beta may be infinite.

##
## After 30 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+AgeCent+LocBody+SizeCent+SizePlus+A2+A4+SizeCent:AgeCent+st
## Crit= 1307.12156816122
## Mean crit= 1319.82661240257
## Change in best IC: -1.07794805812773 / Change in mean IC: -0.783745351258631
##
## After 40 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+AgeCent+LocBody+SizeCent+SizePlus+A2+A4+SizeCent:AgeCent+st
## Crit= 1307.12156816122
## Mean crit= 1319.43379198762
## Change in best IC: 0 / Change in mean IC: -0.392820414949028

## Warning in fitter(X, Y, strats, offset, init, control, weights = weights, :  Loglik converged
before variable  8 ; beta may be infinite.

##
## After 50 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+AgeCent+LocBody+SizeCent+SizePlus+A2+A4+SizeCent:AgeCent+st
## Crit= 1307.12156816122
## Mean crit= 1318.98095408489
## Change in best IC: 0 / Change in mean IC: -0.45283790273561
##
## After 60 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+AgeCent+LocBody+SizeCent+SizePlus+A2+A4+SizeCent:AgeCent+st
## Crit= 1307.12156816122
## Mean crit= 1318.61910175701
## Change in best IC: 0 / Change in mean IC: -0.361852327874885
##
```

```
## After 70 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+AgeCent+LocBody+SizeCent+A2+A4+SizeCent:AgeCent+strata(SexM
## Crit= 1305.31222910484
## Mean crit= 1318.18499444666
## Change in best IC: -1.80933905638744 / Change in mean IC: -0.434107310358286
##
## After 80 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+AgeCent+LocBody+SizeCent+A2+A4+SizeCent:AgeCent+strata(SexM
## Crit= 1305.31222910484
## Mean crit= 1317.59981513137
## Change in best IC: 0 / Change in mean IC: -0.585179315288087
##
## After 90 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+AgeCent+LocBody+SizeCent+A2+A4+SizeCent:AgeCent+strata(SexM
## Crit= 1305.31222910484
## Mean crit= 1317.10893961997
## Change in best IC: 0 / Change in mean IC: -0.490875511393142
##
## After 100 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+AgeCent+LocBody+SizeCent+A2+A4+SizeCent:AgeCent+strata(SexM
## Crit= 1305.31222910484
## Mean crit= 1316.85335637546
## Change in best IC: 0 / Change in mean IC: -0.255583244515265
##
## After 110 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+AgeCent+LocBody+SizeCent+A2+A4+SizeCent:AgeCent+strata(SexM
## Crit= 1305.31222910484
## Mean crit= 1316.23010399055
## Change in best IC: 0 / Change in mean IC: -0.623252384906664
##
## After 120 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+AgeCent+LocBody+SizeCent+A2+A4+SizeCent:AgeCent+strata(SexM
## Crit= 1305.31222910484
## Mean crit= 1316.03826318198
## Change in best IC: 0 / Change in mean IC: -0.191840808574625
##
## After 130 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+AgeCent+LocBody+SizeCent+A2+A4+SizeCent:AgeCent+strata(SexM
## Crit= 1305.31222910484
## Mean crit= 1315.66672272578
## Change in best IC: 0 / Change in mean IC: -0.371540456202638
##
## After 140 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+AgeCent+LocBody+SizeCent+A2+A4+SizeCent:AgeCent+strata(SexM
## Crit= 1305.31222910484
## Mean crit= 1315.505968916
## Change in best IC: 0 / Change in mean IC: -0.160753809770995
```

```
## Warning in fitter(X, Y, strats, offset, init, control, weights = weights, :  Loglik converged
before variable  2 ; beta may be infinite.
```

```
##
## After 150 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+AgeCent+LocBody+SizeCent+A2+A4+SizeCent:AgeCent+strata(SexM
## Crit= 1305.31222910484
```

```
## Mean crit= 1315.49452986765
## Change in best IC: 0 / Change in mean IC: -0.011439048359307
##
## After 160 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+AgeCent+LocBody+SizeCent+A2+A4+SizeCent:AgeCent+strata(SexM
## Crit= 1305.31222910484
## Mean crit= 1315.32517136395
## Change in best IC: 0 / Change in mean IC: -0.169358503697822
##
## After 170 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+AgeCent+LocBody+SizeCent+A2+A4+SizeCent:AgeCent+strata(SexM
## Crit= 1305.31222910484
## Mean crit= 1315.31273510007
## Change in best IC: 0 / Change in mean IC: -0.0124362638737239
```

```
## Warning in fitter(X, Y, strats, offset, init, control, weights = weights, :  Loglik converged
before variable  13 ; beta may be infinite.
```

```
##
## After 180 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+AgeCent+LocBody+SizeCent+A2+A4+SizeCent:AgeCent+strata(SexM
## Crit= 1305.31222910484
## Mean crit= 1315.17172689421
## Change in best IC: 0 / Change in mean IC: -0.141008205865319
##
## After 190 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+AgeCent+LocBody+SizeCent+A2+A4+SizeCent:AgeCent+strata(SexM
## Crit= 1305.31222910484
## Mean crit= 1314.84371970646
## Change in best IC: 0 / Change in mean IC: -0.328007187751837
##
## After 200 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+AgeCent+LocBody+SizeCent+A2+A4+SizeCent:AgeCent+strata(SexM
## Crit= 1305.31222910484
## Mean crit= 1314.64107849843
## Change in best IC: 0 / Change in mean IC: -0.202641208030172
##
## After 210 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+AgeCent+LocBody+SizeCent+A2+A4+SizeCent:AgeCent+strata(SexM
## Crit= 1305.31222910484
## Mean crit= 1314.64107849843
## Change in best IC: 0 / Change in mean IC: 0
```

```
## Warning in fitter(X, Y, strats, offset, init, control, weights = weights, :  Loglik converged
before variable  14 ; beta may be infinite.
```

```
##
## After 220 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+AgeCent+LocBody+SizeCent+A2+A4+SizeCent:AgeCent+strata(SexM
## Crit= 1305.31222910484
## Mean crit= 1314.35056497869
## Change in best IC: 0 / Change in mean IC: -0.290513519736805
##
## After 230 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+AgeCent+LocBody+SizeCent+A2+A4+SizeCent:AgeCent+strata(SexM
```

```
## Crit= 1305.31222910484
## Mean crit= 1314.2362623868
## Change in best IC: 0 / Change in mean IC: -0.114302591889782
##
## After 240 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+AgeCent+LocBody+SizeCent+A2+A4+SizeCent:AgeCent+strata(SexM
## Crit= 1305.31222910484
## Mean crit= 1314.19582313689
## Change in best IC: 0 / Change in mean IC: -0.0404392499053756
##
## After 250 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+AgeCent+LocBody+SizeCent+A2+A4+SizeCent:AgeCent+strata(SexM
## Crit= 1305.31222910484
## Mean crit= 1314.15373535262
## Change in best IC: 0 / Change in mean IC: -0.04208778427369
##
## After 260 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+AgeCent+LocBody+SizeCent+A2+A4+SizeCent:AgeCent+strata(SexM
## Crit= 1305.31222910484
## Mean crit= 1314.08143718906
## Change in best IC: 0 / Change in mean IC: -0.0722981635649376
##
## After 270 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+AgeCent+LocBody+SizeCent+A2+A4+SizeCent:AgeCent+strata(SexM
## Crit= 1305.31222910484
## Mean crit= 1313.95506323665
## Change in best IC: 0 / Change in mean IC: -0.126373952406311

## Warning in fitter(X, Y, strats, offset, init, control, weights = weights, :  Loglik converged
before variable  2 ; beta may be infinite.

##
## After 280 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+AgeCent+LocBody+SizeCent+A2+A4+SizeCent:AgeCent+strata(SexM
## Crit= 1305.31222910484
## Mean crit= 1313.94861918451
## Change in best IC: 0 / Change in mean IC: -0.00644405213938626
##
## After 290 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+AgeCent+LocBody+SizeCent+A2+A4+SizeCent:AgeCent+strata(SexM
## Crit= 1305.31222910484
## Mean crit= 1313.74143092418
## Change in best IC: 0 / Change in mean IC: -0.207188260326802
##
## After 300 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+AgeCent+LocBody+SizeCent+A2+A4+SizeCent:AgeCent+strata(SexM
## Crit= 1305.31222910484
## Mean crit= 1313.65811981253
## Change in best IC: 0 / Change in mean IC: -0.0833111116537566
##
## After 310 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+AgeCent+LocBody+SizeCent+A2+A4+SizeCent:AgeCent+strata(SexM
## Crit= 1305.31222910484
## Mean crit= 1313.62936901721
## Change in best IC: 0 / Change in mean IC: -0.0287507953225941
```

```
##
## After 320 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+AgeCent+LocBody+SizeCent+A2+A4+SizeCent:AgeCent+strata(SexM
## Crit= 1305.31222910484
## Mean crit= 1313.60873430713
## Change in best IC: 0 / Change in mean IC: -0.0206347100800031
##
## After 330 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+AgeCent+LocBody+SizeCent+A2+A4+SizeCent:AgeCent+strata(SexM
## Crit= 1305.31222910484
## Mean crit= 1313.56773455186
## Change in best IC: 0 / Change in mean IC: -0.0409997552667392
##
## After 340 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+AgeCent+LocBody+SizeCent+A2+A4+SizeCent:AgeCent+strata(SexM
## Crit= 1305.31222910484
## Mean crit= 1313.44352373585
## Change in best IC: 0 / Change in mean IC: -0.124210816013147
##
## After 350 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+AgeCent+LocBody+SizeCent+A2+A4+SizeCent:AgeCent+strata(SexM
## Crit= 1305.31222910484
## Mean crit= 1313.43473888021
## Change in best IC: 0 / Change in mean IC: -0.00878485563521281
##
## After 360 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+AgeCent+LocBody+SizeCent+A2+A4+SizeCent:AgeCent+strata(SexM
## Crit= 1305.31222910484
## Mean crit= 1313.42990089756
## Change in best IC: 0 / Change in mean IC: -0.00483798264917823

## Warning in fitter(X, Y, strats, offset, init, control, weights = weights, :  Loglik converged
before variable  11 ; beta may be infinite.

##
## After 370 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+AgeCent+LocBody+SizeCent+A2+A4+SizeCent:AgeCent+strata(SexM
## Crit= 1305.31222910484
## Mean crit= 1313.42990089756
## Change in best IC: 0 / Change in mean IC: 0
##
## After 380 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+AgeCent+LocBody+SizeCent+A2+A4+SizeCent:AgeCent+strata(SexM
## Crit= 1305.31222910484
## Mean crit= 1313.3788732276
## Change in best IC: 0 / Change in mean IC: -0.0510276699585575
##
## After 390 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+AgeCent+LocBody+SizeCent+A2+A4+SizeCent:AgeCent+strata(SexM
## Crit= 1305.31222910484
## Mean crit= 1313.3788732276
## Change in best IC: 0 / Change in mean IC: 0
##
## After 400 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+AgeCent+LocBody+SizeCent+A2+A4+SizeCent:AgeCent+strata(SexM
```

```
## Crit= 1305.31222910484
## Mean crit= 1313.25933302457
## Change in best IC: 0 / Change in mean IC: -0.119540203037559
##
## After 410 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+AgeCent+LocBody+SizeCent+A2+A4+SizeCent:AgeCent+strata(SexM
## Crit= 1305.31222910484
## Mean crit= 1313.21297633323
## Change in best IC: 0 / Change in mean IC: -0.0463566913335853
##
## After 420 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+AgeCent+LocBody+SizeCent+A2+A4+SizeCent:AgeCent+strata(SexM
## Crit= 1305.31222910484
## Mean crit= 1313.21183234165
## Change in best IC: 0 / Change in mean IC: -0.00114399158724154
##
## After 430 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+AgeCent+LocBody+SizeCent+A2+A4+SizeCent:AgeCent+strata(SexM
## Crit= 1305.31222910484
## Mean crit= 1313.21128427043
## Change in best IC: 0 / Change in mean IC: -0.000548071215462187
##
## After 440 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+AgeCent+LocBody+SizeCent+A2+A4+SizeCent:AgeCent+strata(SexM
## Crit= 1305.31222910484
## Mean crit= 1313.18111206023
## Change in best IC: 0 / Change in mean IC: -0.0301722101953601
##
## After 450 generations:
## Best model: Surv(Time,DSD)~1+strata(SexM)+AgeCent+LocBody+SizeCent+A2+A4+SizeCent:AgeCent+strata(SexM
## Crit= 1305.31222910484
## Mean crit= 1313.16785543812
## Improvements in best and average IC have bebingo en below the specified goals.
## Algorithm is declared to have converged.
## Completed.

fit.cph.as.aicc1 = glmulti(Surv(Time, DSD) ~ strata(SexM) + AgeCent + LocBody + SizeCent + SizePlus + A2

## Initialization...
## TASK: Exhaustive screening of candidate set.
## Fitting...
##
## After 50 models:
## Best model: Surv(Time,DSD)~1+LocBody+SizeCent+A4
## Crit= 1562.92910743338
## Mean crit= 1570.63396981566
##
## After 100 models:
## Best model: Surv(Time,DSD)~1+strata(SexM)+LocBody+SizeCent+A2
## Crit= 1315.8613218026
## Mean crit= 1484.90325895394
##
## After 150 models:
## Best model: Surv(Time,DSD)~1+strata(SexM)+LocBody+SizeCent+A2+A4
```

```
## Crit= 1309.03451494962
## Mean crit= 1406.96604818801
## Completed.
```

```r
rm(nobs.coxph)
summary(fit.cph.as.bic)$bestmodel
```

```
## [1] "Surv(Time, DSD) ~ 1 + strata(SexM) + SizePlus + A2 + A4"
```

```r
summary(fit.cph.as.aicc)$bestmodel
```

```
## [1] "Surv(Time, DSD) ~ 1 + strata(SexM) + AgeCent + LocBody + SizeCent + "
## [2] "    A2 + A4 + SizeCent:AgeCent + strata(SexM):SizeCent"
```

```r
summary(fit.cph.as.bic1)$bestmodel
```

```
## [1] "Surv(Time, DSD) ~ 1 + strata(SexM) + SizeCent + A2 + A4"
```

```r
summary(fit.cph.as.aicc1)$bestmodel
```

```
## [1] "Surv(Time, DSD) ~ 1 + strata(SexM) + LocBody + SizeCent + A2 + "
## [2] "    A4"
```

Also run BIC stepwise, because we can.

```r
stepAIC(fit.cph, k = log(nrow(data)))
```

```
## Start:  AIC=1330
## Surv(Time, DSD) ~ strata(SexM) + AgeCent + LocBody + SizeCent +
##     SizePlus + A2 + A4
##
##            Df  AIC
## - SizePlus  1 1325
## - SizeCent  1 1326
## - AgeCent   1 1327
## - LocBody   1 1328
## <none>        1330
## - A4        1 1333
## - A2        1 1334
##
## Step:  AIC=1325
## Surv(Time, DSD) ~ strata(SexM) + AgeCent + LocBody + SizeCent +
##     A2 + A4
##
##            Df  AIC
## - AgeCent   1 1322
## - LocBody   1 1322
## - SizeCent  1 1324
## <none>        1325
## - A2        1 1329
## - A4        1 1330
##
## Step:  AIC=1322
## Surv(Time, DSD) ~ strata(SexM) + LocBody + SizeCent + A2 + A4
##
##            Df  AIC
```

```
## - LocBody   1 1319
## - SizeCent  1 1321
## <none>        1322
## - A2        1 1325
## - A4        1 1326
##
## Step:  AIC=1319
## Surv(Time, DSD) ~ strata(SexM) + SizeCent + A2 + A4
##
##            Df  AIC
## <none>        1319
## - SizeCent  1 1322
## - A4        1 1322
## - A2        1 1324
## Call:
## coxph(formula = Surv(Time, DSD) ~ strata(SexM) + SizeCent + A2 +
##     A4, data = data)
##
##
##            coef exp(coef) se(coef)    z      p
## SizeCent 0.0159      1.02  0.00543 2.92 0.0035
## A2TRUE   0.7003      2.01  0.20650 3.39 0.0007
## A4TRUE   0.5154      1.67  0.18497 2.79 0.0053
##
## Likelihood ratio test=34.1  on 3 df, p=1.92e-07  n= 193, number of events= 184
```

```r
stepAIC(fit.cph, k = 2)
```

```
## Start:  AIC=1311
## Surv(Time, DSD) ~ strata(SexM) + AgeCent + LocBody + SizeCent +
##     SizePlus + A2 + A4
##
##            Df  AIC
## - SizePlus  1 1309
## - SizeCent  1 1310
## - AgeCent   1 1311
## <none>        1311
## - LocBody   1 1311
## - A4        1 1317
## - A2        1 1318
##
## Step:  AIC=1309
## Surv(Time, DSD) ~ strata(SexM) + AgeCent + LocBody + SizeCent +
##     A2 + A4
##
##            Df  AIC
## - AgeCent   1 1309
## <none>        1309
## - LocBody   1 1309
## - SizeCent  1 1311
## - A2        1 1316
## - A4        1 1317
##
## Step:  AIC=1309
```

```
## Surv(Time, DSD) ~ strata(SexM) + LocBody + SizeCent + A2 + A4
##
##            Df  AIC
## <none>        1309
## - LocBody   1 1309
## - SizeCent  1 1311
## - A2        1 1315
## - A4        1 1316
## Call:
## coxph(formula = Surv(Time, DSD) ~ strata(SexM) + LocBody + SizeCent +
##     A2 + A4, data = data)
##
##
##                coef exp(coef) se(coef)    z      p
## LocBodyTRUE 0.3806      1.46   0.2267 1.68 0.0930
## SizeCent    0.0126      1.01   0.0058 2.18 0.0290
## A2TRUE      0.6301      1.88   0.2120 2.97 0.0030
## A4TRUE      0.5312      1.70   0.1850 2.87 0.0041
##
## Likelihood ratio test=36.7  on 4 df, p=2.04e-07  n= 193, number of events= 184
```

4.6 Final Fits

```
fit.cph.as.bic = coxph(Surv(Time, DSD) ~ strata(SexM) + SizePlus + A2 + A4, data = data)
cox.zph(fit.cph.as.bic)
```

```
##              rho  chisq     p
## SizePlus  0.0212 0.0876 0.767
## A2TRUE    0.0340 0.2136 0.644
## A4TRUE   -0.0808 1.1972 0.274
## GLOBAL        NA 1.3865 0.709
```

```
fit.cph.as.aicc = coxph(Surv(Time, DSD) ~ strata(SexM)+AgeCent+LocBody+SizeCent+A2+A4+SizeCent:AgeCent+s
cox.zph(fit.cph.as.aicc)
```

```
##                                    rho   chisq      p
## AgeCent                       -0.16098 5.43356 0.0198
## LocBodyTRUE                    0.03967 0.30863 0.5785
## SizeCent                       0.00379 0.00275 0.9581
## A2TRUE                         0.04060 0.34304 0.5581
## A4TRUE                        -0.06803 0.84941 0.3567
## AgeCent:SizeCent               0.03856 0.28388 0.5942
## strata(SexM)SexM=TRUE:SizeCent  0.00853 0.01322 0.9085
## GLOBAL                              NA 7.49932 0.3788
```

```
fit.cph.sw.bic = coxph(Surv(Time, DSD) ~ strata(SexM) + SizeCent + A2 + A4, data = data)
cox.zph(fit.cph.sw.bic)
```

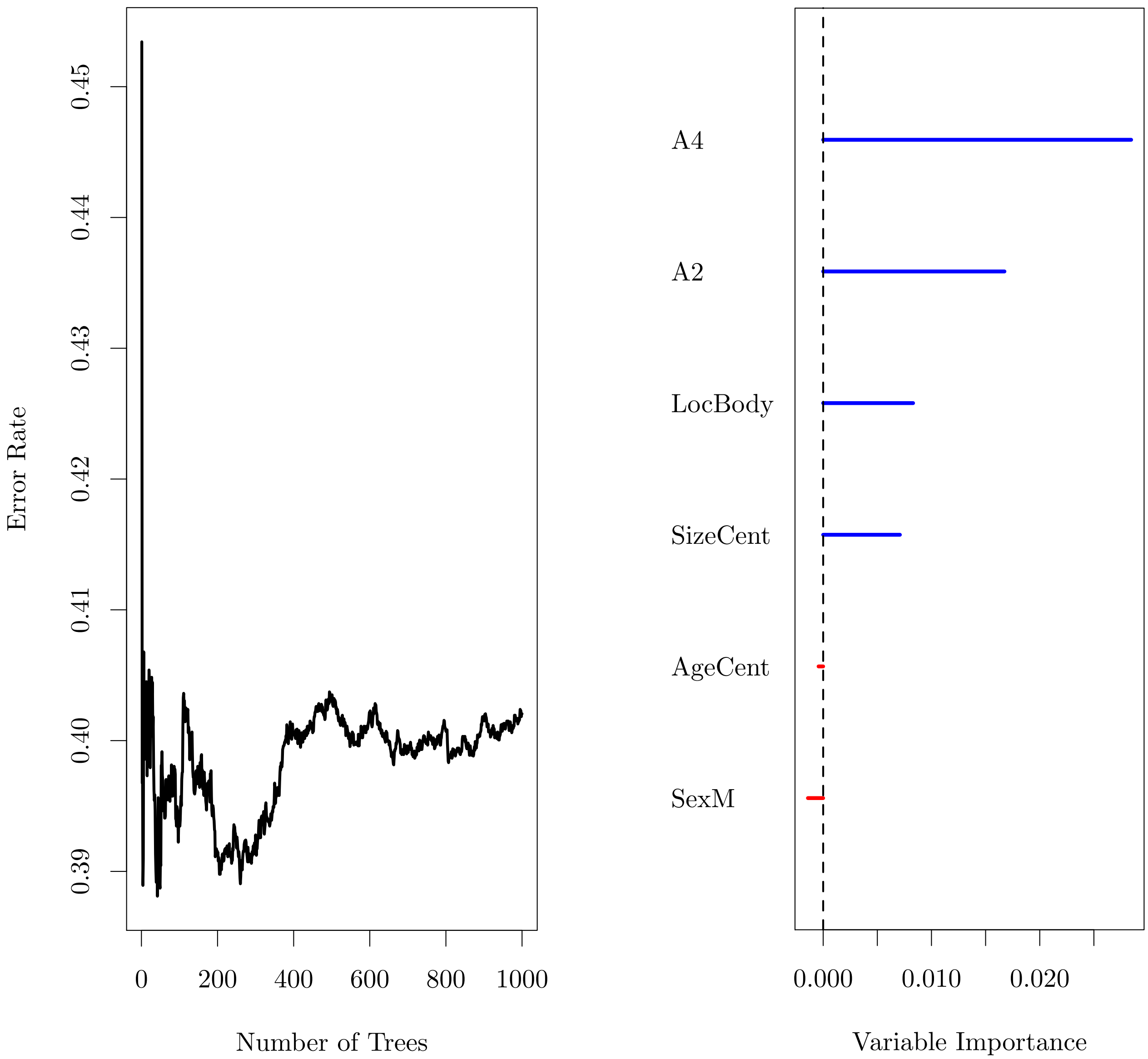
```
##              rho  chisq     p
## SizeCent  0.0162 0.0507 0.822
## A2TRUE    0.0312 0.1797 0.672
## A4TRUE   -0.0874 1.4015 0.236
## GLOBAL        NA 1.4878 0.685
```

```r
fit.cph.sw.aic = coxph(Surv(Time, DSD) ~ strata(SexM) + LocBody + SizeCent + A2 + A4, data = data)
cox.zph(fit.cph.sw.aic)
```

```
##                  rho  chisq     p
## LocBodyTRUE   0.0180 0.0592 0.808
## SizeCent      0.0280 0.1465 0.702
## A2TRUE        0.0292 0.1636 0.686
## A4TRUE       -0.0839 1.2904 0.256
## GLOBAL            NA 1.6815 0.794
```

```r
fit.cph = fit.cph.sw.aic
```

```r
set.seed(20150208)
fit.rsf = rfsrc(Surv(Time, DSD) ~ SexM + AgeCent + LocBody + SizeCent + A2 + A4, data = data, mtry = 1,
plot(fit.rsf)
```



```
##
##            Importance   Relative Imp
## A4             0.0284         1.0000
## A2             0.0167         0.5887
## LocBody        0.0083         0.2920
## SizeCent       0.0071         0.2492
## AgeCent       -0.0004        -0.0149
## SexM          -0.0014        -0.0494
```

```
fit.gg = flexsurvreg(Surv(Time, DSD) ~ SexM + LocBody + SizeCent + A2 + A4,
        anc = list(
                sigma = ~ SexM,
                Q = ~ SexM),
        data = data, dist = "gengamma")

fit.gg2 = flexsurvreg(Surv(Time, DSD) ~ SexM+AgeCent+LocBody+SizeCent+A2+A4+SizeCent:AgeCent+SexM:SizeCe
        anc = list(
                sigma = ~ SexM,
                Q = ~ SexM),
        data = data, dist = "gengamma")

fit.gg$loglik

## [1] -1325

fit.gg2$loglik

## [1] -1321

pchisq(2*(fit.gg2$loglik - fit.gg$loglik), 3, lower.tail = FALSE)

## [1] 0.04837

AIC(fit.gg)

## [1] 2669

AIC(fit.gg2)

## [1] 2668

fit.gg

##
## Call:
## flexsurvreg(formula = Surv(Time, DSD) ~ SexM + LocBody + SizeCent +     A2 + A4, anc = list(sigma = 
##
## Estimates:
##                    data mean  est       L95%      U95%      se
## mu                       NA    6.53611   6.19247   6.87976   0.17533
## sigma                    NA    0.78047   0.67245   0.90585   0.05932
## Q                        NA    0.11827  -0.49632   0.73287   0.31357
## SexMTRUE            0.51813    0.28181  -0.07256   0.63619   0.18081
## LocBodyTRUE         0.17098   -0.20952  -0.50577   0.08673   0.15115
## SizeCent            3.65285   -0.00879  -0.01600  -0.00158   0.00368
```

```
## A2TRUE            0.16580   -0.38962  -0.65941  -0.11983   0.13765
## A4TRUE            0.75130   -0.39725  -0.62687  -0.16763   0.11716
## sigma(SexMTRUE)   0.51813   -0.26267  -0.49374  -0.03159   0.11790
## Q(SexMTRUE)       0.51813    0.48452  -0.32987   1.29891   0.41551
##                  exp(est)  L95%      U95%
## mu                     NA        NA        NA
## sigma                  NA        NA        NA
## Q                      NA        NA        NA
## SexMTRUE          1.32553   0.93001   1.88927
## LocBodyTRUE       0.81097   0.60304   1.09060
## SizeCent          0.99124   0.98412   0.99842
## A2TRUE            0.67731   0.51715   0.88707
## A4TRUE            0.67217   0.53426   0.84567
## sigma(SexMTRUE)   0.76900   0.61034   0.96890
## Q(SexMTRUE)       1.62340   0.71902   3.66531
##
## N = 193,  Events: 184,  Censored: 9
## Total time at risk: 114833
## Log-likelihood = -1325, df = 10
## AIC = 2669
```

```
fit.gg2
```

```
##
## Call:
## flexsurvreg(formula = Surv(Time, DSD) ~ SexM + AgeCent + LocBody +     SizeCent + A2 + A4 + SizeCent
##
## Estimates:
##                    data mean  est        L95%       U95%       se
## mu                        NA   6.530218   6.184887   6.875549   0.176192
## sigma                     NA   0.771216   0.660311   0.900749   0.061092
## Q                         NA   0.228786  -0.410815   0.868387   0.326333
## SexMTRUE            0.518135   0.322116  -0.039753   0.683986   0.184631
## AgeCent            -1.067358   0.010352   0.000170   0.020534   0.005195
## LocBodyTRUE         0.170984  -0.271326  -0.558764   0.016113   0.146655
## SizeCent            3.652850  -0.004245  -0.015597   0.007107   0.005792
## A2TRUE              0.165803  -0.358631  -0.618603  -0.098660   0.132641
## A4TRUE              0.751295  -0.354054  -0.574822  -0.133287   0.112639
## AgeCent:SizeCent   -8.896373  -0.000855  -0.001550  -0.000160   0.000354
## SexMTRUE:SizeCent   1.772021  -0.006910  -0.020503   0.006684   0.006936
## sigma(SexMTRUE)     0.518135  -0.334045  -0.602093  -0.065998   0.136762
## Q(SexMTRUE)         0.518135   0.550014  -0.328860   1.428889   0.448414
##                    exp(est)   L95%       U95%
## mu                        NA         NA         NA
## sigma                     NA         NA         NA
## Q                         NA         NA         NA
## SexMTRUE            1.380045   0.961027   1.981761
## AgeCent             1.010406   1.000170   1.020746
## LocBodyTRUE         0.762368   0.571915   1.016243
## SizeCent            0.995764   0.984524   1.007133
## A2TRUE              0.698632   0.538697   0.906051
## A4TRUE              0.701837   0.562805   0.875214
## AgeCent:SizeCent    0.999145   0.998452   0.999840
## SexMTRUE:SizeCent   0.993114   0.979706   1.006706
```

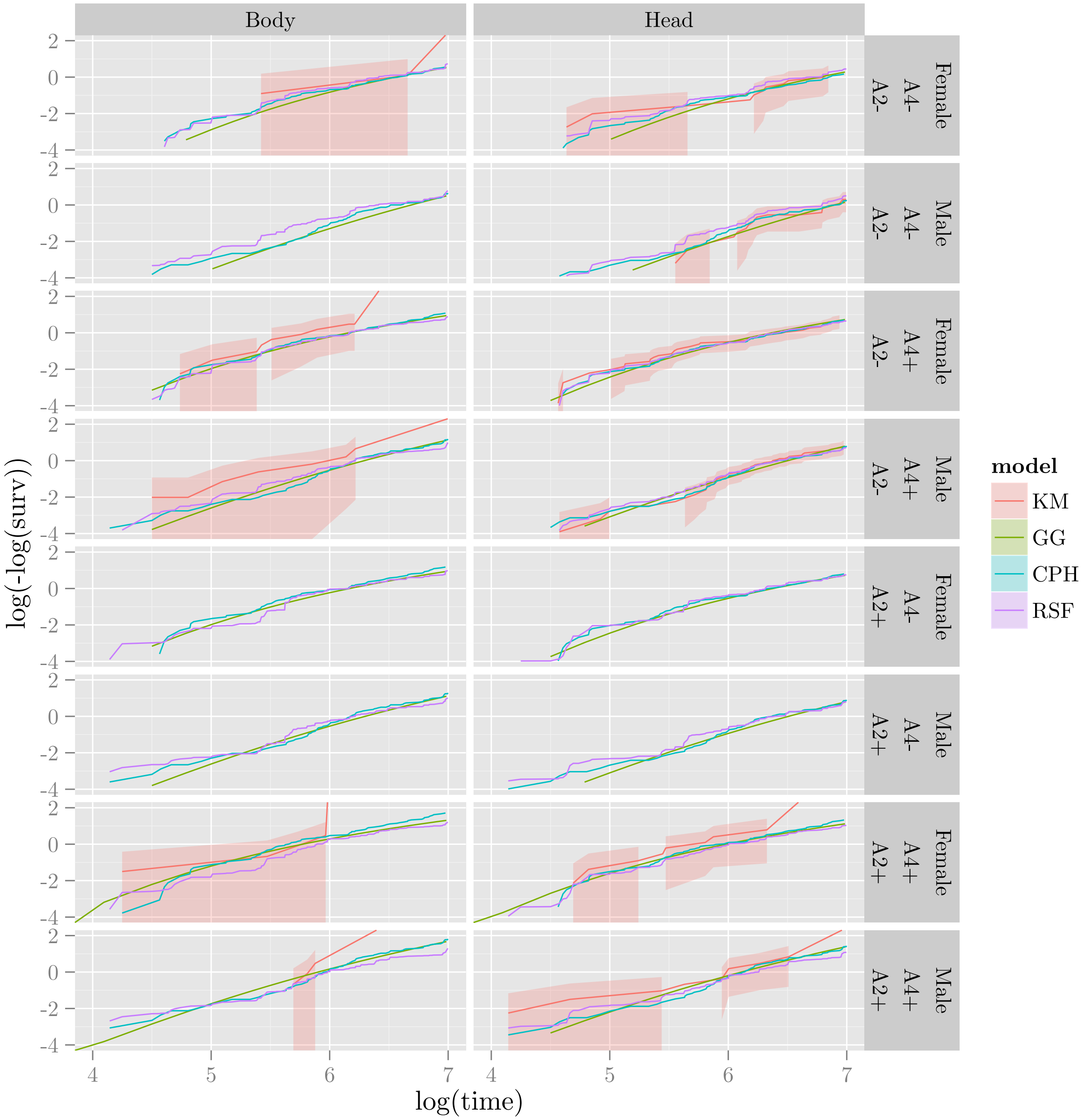
```
## sigma(SexMTRUE)     0.716021   0.547664   0.936133
## Q(SexMTRUE)         1.733278   0.719744   4.174059
##
## N = 193,  Events: 184,  Censored: 9
## Total time at risk: 114833
## Log-likelihood = -1321, df = 13
## AIC = 2668
```

5 Fit assessment

Plot fit stratified by sex, separate curves for A2, A4 status, at median (approx.) Size.

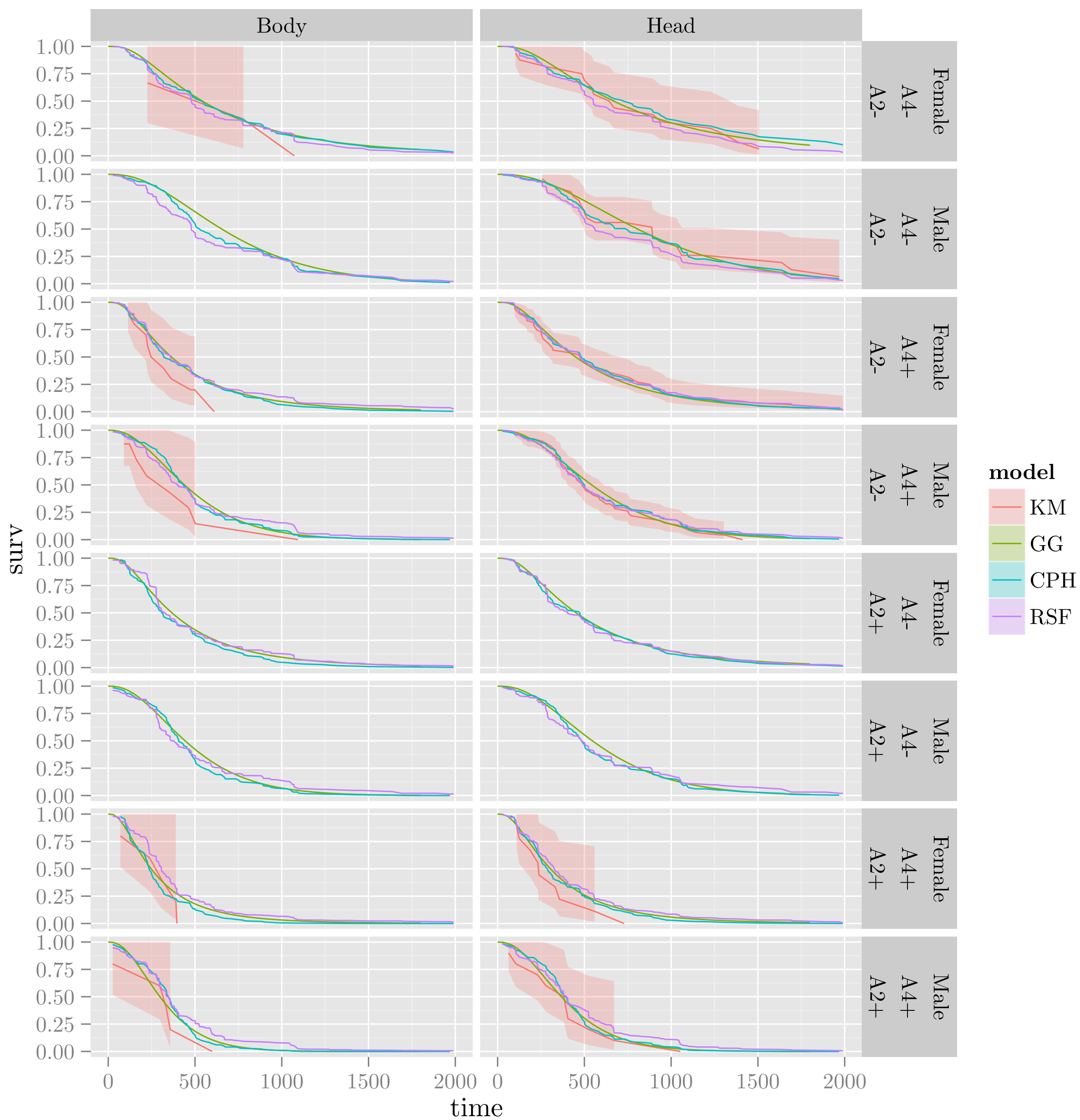
```
temp.grid = expand.grid(A4 = c(FALSE, TRUE), A2 = c(FALSE, TRUE), SexM = c(FALSE, TRUE), SizeCent = 0, L
temp.grid$ID = sprintf("SexM=%s, A2=% -5s, A4=% -5s, LocBody=%s", temp.grid$SexM, temp.grid$A2, temp.gri
temp.preds = summary(fit.gg, newdata = temp.grid, type = "survival", t = seq(0, 365*5, 30))
temp.preds2 = do.call(rbind, temp.preds)
temp.preds2$group = rep(gsub(".*ID=", "", names(temp.preds)), each = nrow(temp.preds[[1]]))
temp.preds.cox = survfit(fit.cph, newdata = temp.grid)
temp.preds.rsf = predict(fit.rsf, newdata = temp.grid)

temp.survfit = survfit(Surv(Time, DSD) ~ SexM + A2 + A4 + LocBody, data)
temp.data = data.frame(time = temp.survfit$time, surv = temp.survfit$surv, upper = temp.survfit$lower, l
temp.data = rbind(temp.data, data.frame(time = temp.preds2$time, surv = temp.preds2$est, upper = temp.pr
temp.data = rbind(temp.data, data.frame(time = temp.preds.cox$time, surv = temp.preds.cox$surv, upper =
temp.data = rbind(temp.data, data.frame(time = rep(temp.preds.rsf$time.interest, each = nrow(temp.preds.

temp.data$Sex = c("Male", "Female")[grepl("SexM=FALSE", temp.data$group)+1]
temp.data$A2 = c("A2-", "A2+")[grepl("A2=TRUE", temp.data$group)+1]
temp.data$A4 = c("A4-", "A4+")[grepl("A4=TRUE", temp.data$group)+1]
temp.data$Location = c("Head", "Body")[grepl("LocBody=TRUE", temp.data$group)+1]

temp.data$lower[temp.data$model != "KM"] = NA
temp.data$upper[temp.data$model != "KM"] = NA
ggplot(temp.data, aes(x = log(time), y = log(-log(surv)), ymin = log(-log(lower)), ymax = log(-log(upper
        geom_ribbon(alpha = 0.25, colour = NA) +
        geom_line() +
        xlim(4, 7) + ylim(-4, 2) +
        facet_grid(A2 ~ A4 ~ Sex ~ Location)

## Warning:  Removed 64 rows containing missing values (geom_path).
## Warning:  Removed 70 rows containing missing values (geom_path).
## Warning:  Removed 59 rows containing missing values (geom_path).
## Warning:  Removed 69 rows containing missing values (geom_path).
## Warning:  Removed 60 rows containing missing values (geom_path).
## Warning:  Removed 70 rows containing missing values (geom_path).
## Warning:  Removed 57 rows containing missing values (geom_path).
## Warning:  Removed 66 rows containing missing values (geom_path).
## Warning:  Removed 58 rows containing missing values (geom_path).
## Warning:  Removed 59 rows containing missing values (geom_path).
## Warning:  Removed 56 rows containing missing values (geom_path).
## Warning:  Removed 56 rows containing missing values (geom_path).
## Warning:  Removed 57 rows containing missing values (geom_path).
```

```
## Warning:  Removed 58 rows containing missing values (geom_path).
## Warning:  Removed 57 rows containing missing values (geom_path).
## Warning:  Removed 56 rows containing missing values (geom_path).
```

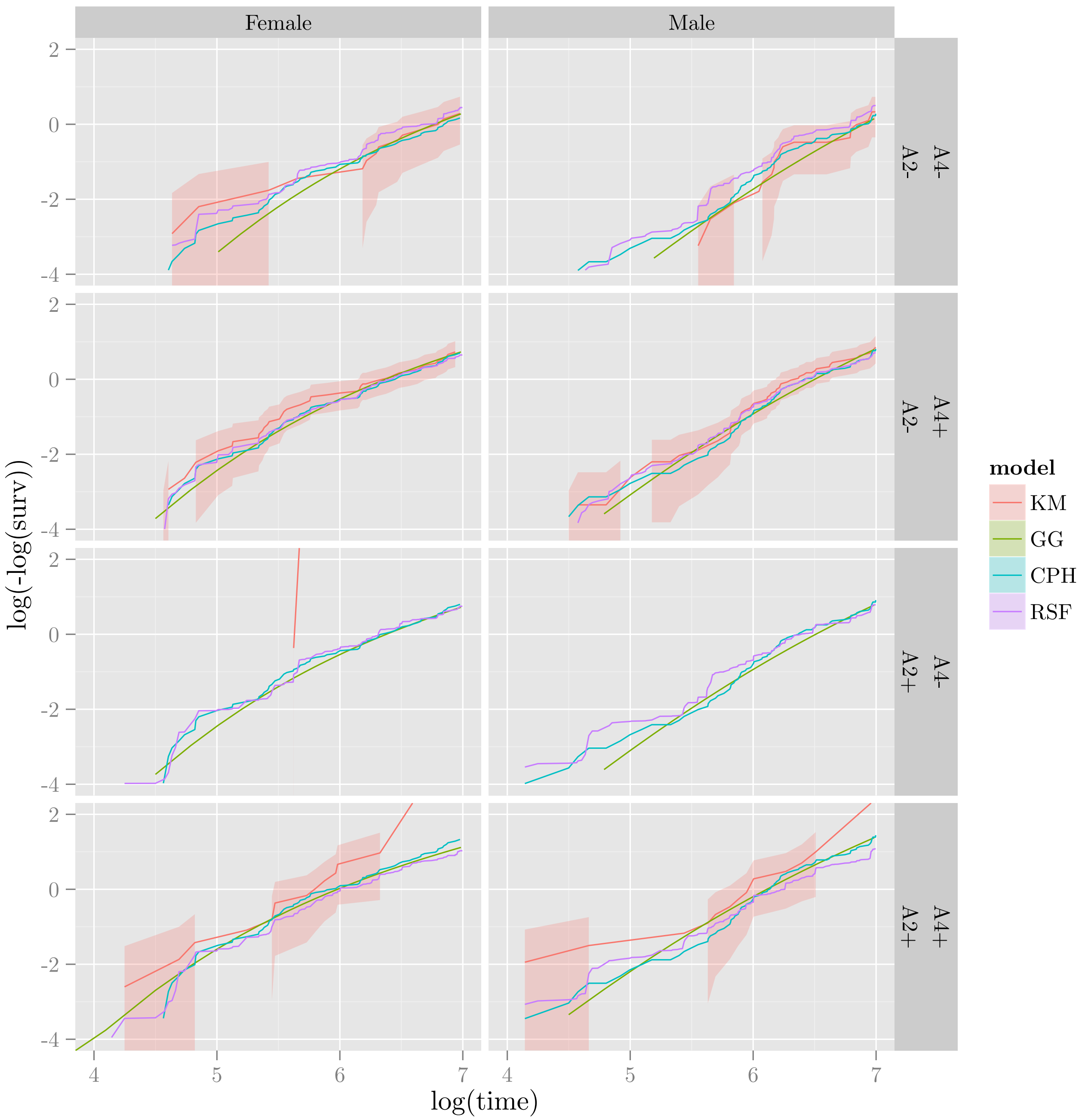



```
ggplot(temp.data, aes(x = time, y = surv, ymin = lower, ymax = upper, colour = model, fill = model)) +
        geom_ribbon(alpha = 0.25, colour = NA) +
        geom_line() + xlim(0, 2000) + ylim(0, 1) +
        facet_grid(A2 ~ A4 ~ Sex ~ Location)

## Warning:  Removed 4 rows containing missing values (geom_path).
## Warning:  Removed 5 rows containing missing values (geom_path).
## Warning:  Removed 3 rows containing missing values (geom_path).
```

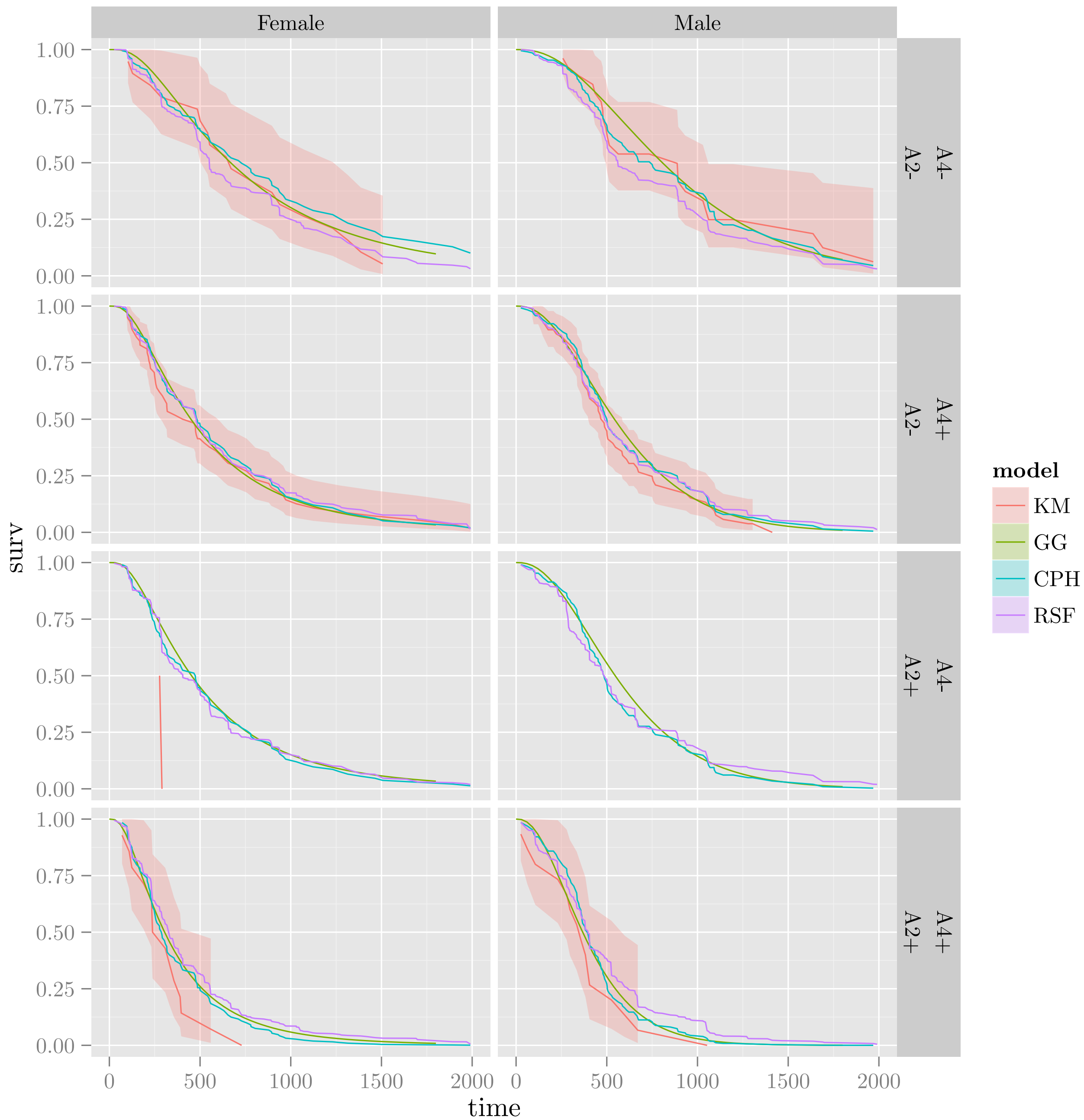
```
## Warning:  Removed 4 rows containing missing values (geom_path).
## Warning:  Removed 4 rows containing missing values (geom_path).
## Warning:  Removed 5 rows containing missing values (geom_path).
## Warning:  Removed 3 rows containing missing values (geom_path).
## Warning:  Removed 3 rows containing missing values (geom_path).
## Warning:  Removed 4 rows containing missing values (geom_path).
## Warning:  Removed 4 rows containing missing values (geom_path).
## Warning:  Removed 3 rows containing missing values (geom_path).
## Warning:  Removed 3 rows containing missing values (geom_path).
## Warning:  Removed 4 rows containing missing values (geom_path).
## Warning:  Removed 4 rows containing missing values (geom_path).
## Warning:  Removed 3 rows containing missing values (geom_path).
## Warning:  Removed 3 rows containing missing values (geom_path).
```



```
temp.grid = expand.grid(A4 = c(FALSE, TRUE), A2 = c(FALSE, TRUE), SexM = c(FALSE, TRUE), SizeCent = 0, A
temp.grid$ID = sprintf("SexM=%s, A2=% -5s, A4=% -5s, LocBody=%s", temp.grid$SexM, temp.grid$A2, temp.gri
temp.preds = summary(fit.gg, newdata = temp.grid, type = "survival", t = seq(0, 365*5, 30))
temp.preds2 = do.call(rbind, temp.preds)
temp.preds2$group = rep(gsub(".*ID=", "", names(temp.preds)), each = nrow(temp.preds[[1]]))
temp.preds.cox = survfit(fit.cph, newdata = temp.grid)
temp.preds.rsf = predict(fit.rsf, newdata = temp.grid)

temp.survfit = survfit(Surv(Time, DSD) ~ SexM + A2 + A4, data)
temp.data = data.frame(time = temp.survfit$time, surv = temp.survfit$surv, upper = temp.survfit$lower, l
temp.data = rbind(temp.data, data.frame(time = temp.preds2$time, surv = temp.preds2$est, upper = temp.pr
temp.data = rbind(temp.data, data.frame(time = temp.preds.cox$time, surv = temp.preds.cox$surv, upper =
temp.data = rbind(temp.data, data.frame(time = rep(temp.preds.rsf$time.interest, each = nrow(temp.preds.

temp.data$Sex = c("Male", "Female")[grepl("SexM=FALSE", temp.data$group)+1]
temp.data$A2 = c("A2-", "A2+")[grepl("A2=TRUE", temp.data$group)+1]
temp.data$A4 = c("A4-", "A4+")[grepl("A4=TRUE", temp.data$group)+1]

temp.data$lower[temp.data$model != "KM"] = NA
temp.data$upper[temp.data$model != "KM"] = NA
ggplot(temp.data, aes(x = log(time), y = log(-log(surv)), ymin = log(-log(lower)), ymax = log(-log(upper
        geom_ribbon(alpha = 0.25, colour = NA) +
        geom_line() +
        xlim(4, 7) + ylim(-4, 2) +
        facet_grid(A2 ~ A4 ~ Sex)

## Warning:  Removed 70 rows containing missing values (geom_path).
## Warning:  Removed 69 rows containing missing values (geom_path).
## Warning:  Removed 71 rows containing missing values (geom_path).
## Warning:  Removed 67 rows containing missing values (geom_path).
## Warning:  Removed 59 rows containing missing values (geom_path).
## Warning:  Removed 56 rows containing missing values (geom_path).
## Warning:  Removed 58 rows containing missing values (geom_path).
## Warning:  Removed 57 rows containing missing values (geom_path).
```




```
ggplot(temp.data, aes(x = time, y = surv, ymin = lower, ymax = upper, colour = model, fill = model)) +
        geom_ribbon(alpha = 0.25, colour = NA) +
        geom_line() + xlim(0, 2000) + ylim(0, 1) +
        facet_grid(A2 ~ A4 ~ Sex)

## Warning:  Removed 5 rows containing missing values (geom_path).
## Warning:  Removed 4 rows containing missing values (geom_path).
## Warning:  Removed 5 rows containing missing values (geom_path).
## Warning:  Removed 3 rows containing missing values (geom_path).
## Warning:  Removed 4 rows containing missing values (geom_path).
## Warning:  Removed 3 rows containing missing values (geom_path).
## Warning:  Removed 4 rows containing missing values (geom_path).
## Warning:  Removed 3 rows containing missing values (geom_path).
```



6 Model selection

It looks like that's as far as we can go with tweaking the fits. Time to put the different models against each other on the holdout data, and choose a winner.

DIY IBS, wooo.

```
calcIBS = function(surv, pred, pred_times, max_time, min_time = 0)
{
        stopifnot(nrow(surv) == nrow(pred) && length(pred_times) == ncol(pred))

        n = nrow(surv)
        marg_survfit = survfit(surv ~ 1)
```

```
        marg_censfit = survfit(Surv(surv[,1], !surv[,2]) ~ 1)
        marg_surv_func = approxfun(marg_survfit$time, marg_survfit$surv, method = "constant", yleft = 1,
        marg_cens_func = approxfun(marg_censfit$time, marg_censfit$surv, method = "constant", yleft = 1,

        pred_funcs = apply(pred, 1, function(pat_preds) approxfun(pred_times, pat_preds, yleft = 1, yrig

        indiv_patient_bsc = function(pat_i, tstars)
        {
                observed_time = surv[pat_i, 1]
                observed_event = surv[pat_i, 2]
                pred_func = pred_funcs[[pat_i]]
                category = 1*(observed_time <= tstars & observed_event) + 2*(observed_time > tstars) + 3
                bsc = rep(NA, length(tstars))
                bsc[category == 1] = pred_func(tstars[category == 1])^2 / marg_cens_func(observed_time)
                bsc[category == 2] = (1 - pred_func(tstars[category == 2]))^2 / marg_cens_func(tstars[ca
                bsc[category == 3] = 0
                bsc
        }

        bsc_func = function(tstars) { rowMeans(sapply(1:n, function(pat_i) indiv_patient_bsc(pat_i, tsta

        weight_func = function(tstars) { (1 - marg_surv_func(tstars)) / (1 - marg_surv_func(max_time)) }

        # Be slack and do trapezoidal int. with a fine grid.  It should be possible
        # to calulate the int. exactly but I cbfed.
        int_grid = seq(min_time, max_time, length.out = 1e3)
        bsc_vals = bsc_func(int_grid)
        weight_vals = weight_func(int_grid)
        int_vals = bsc_vals * weight_vals
        ibsc = (2*sum(int_vals) - int_vals[1] - int_vals[length(int_vals)]) * (diff(range(int_grid))) /

        return(list(bsc = bsc_vals, weights = weight_vals, eval_times = int_grid, ibsc = ibsc))
}
```

Calculate survival probability predictions for each of the models, on the validation data.

```
ibs_times = sort(unique(data.val$Time))
ibs_preds_gg = as.matrix(t(sapply(summary(fit.gg, newdata = data.val, type = "survival", t = ibs_times),
ibs_preds_gg2 = as.matrix(t(sapply(summary(fit.gg2, newdata = data.val, type = "survival", t = ibs_times
temp_cox_preds = survfit(fit.cph, newdata = data.val)
ibs_preds_cph = simplify2array(tapply(1:length(temp_cox_preds$time), rep(names(temp_cox_preds$strata), t
        approx(x = temp_cox_preds$time[strat_i], y = temp_cox_preds$surv[strat_i], xout = ibs_times, met
ibs_preds_cph = t(ibs_preds_cph[,rownames(data.val)])
temp_rsf_preds = predict(fit.rsf, newdata = data.val)
ibs_preds_rsf = t(apply(temp_rsf_preds$survival, 1, function(survs) approx(temp_rsf_preds$time.interest,
# Patients (from data.val) are in rows, times (from ibs_times) in columns.

# Add a no-information KM predictor
temp_km0 = survfit(Surv(Time, DSD) ~ 1, data)
ibs_preds_km0 = t(matrix(rep(approx(temp_km0$time, temp_km0$surv, xout = ibs_times, method = "constant",
ibs_preds_all = list(gg = ibs_preds_gg, gg2 = ibs_preds_gg2, cph = ibs_preds_cph, rsf = ibs_preds_rsf, k
```

```
val.prob.times = seq(0, max(data.val$Time), 1)

temp.coefs = coef(fit.gg)
val.linpred.gg = sapply(1:length(temp.coefs), function(coef_i) {
    if (names(temp.coefs)[coef_i] %in% colnames(data.val)) {
        temp.coefs[coef_i] * data.val[,names(temp.coefs)[coef_i]]
    } else if (gsub("TRUE$", "", names(temp.coefs)[coef_i]) %in% colnames(data.val)) {
        temp.coefs[coef_i] * data.val[,gsub("TRUE$", "", names(temp.coefs)[coef_i])]
    } else {
        rep(0, nrow(data.val))
    } })
val.linpred.gg = -rowSums(val.linpred.gg)   # Negate to bring into concordance with the direction of Co
temp = summary(fit.gg, newdata = data.val, ci = FALSE)
val.prob.gg = sapply(temp, function(x) approx(x[,1], x[,2], xout = val.prob.times, yleft = 1, yright = (
colnames(val.prob.gg) = rownames(data.val)

temp.coefs = coef(fit.gg2)
val.linpred.gg2 = sapply(1:length(temp.coefs), function(coef_i) {
    if (names(temp.coefs)[coef_i] %in% colnames(data.val)) {
        temp.coefs[coef_i] * data.val[,names(temp.coefs)[coef_i]]
    } else if (gsub("TRUE$", "", names(temp.coefs)[coef_i]) %in% colnames(data.val)) {
        temp.coefs[coef_i] * data.val[,gsub("TRUE$", "", names(temp.coefs)[coef_i])]
    } else {
        rep(0, nrow(data.val))
    } })
val.linpred.gg2 = -rowSums(val.linpred.gg2)   # Negate to bring into concordance with the direction of (
temp = summary(fit.gg2, newdata = data.val, ci = FALSE)
val.prob.gg2 = sapply(temp, function(x) approx(x[,1], x[,2], xout = val.prob.times, yleft = 1, yright =
colnames(val.prob.gg2) = rownames(data.val)

val.linpred.cph = predict(fit.cph, newdata = data.val)
temp = survfit(fit.cph, newdata = data.val)
val.prob.cph = simplify2array(tapply(1:length(temp$surv), rep(names(temp$strata), temp$strata), functio

temp = predict(fit.rsf, newdata = data.val)
# val.linpred.rsf = temp£predicted
# Median survival time:
val.linpred.rsf = apply(temp$survival, 1, function(s1) {
    sfunc = approxfun(temp$time.interest, s1, yleft = 1, yright = 0, rule = 2)
    med = uniroot(function(x) sfunc(x) - 0.5, lower = min(temp$time.interest), upper = max(temp$time.int
    med
})
val.linpred.rsf = -val.linpred.rsf
val.prob.rsf = apply(temp$survival, 1, function(s1) approx(temp$time.interest, s1, xout = val.prob.times
colnames(val.prob.rsf) = rownames(data.val)

summary(coxph(Surv(Time, DSD) ~ val.linpred.gg, data.val))

## Call:
## coxph(formula = Surv(Time, DSD) ~ val.linpred.gg, data = data.val)
##
##   n= 49, number of events= 49
##
##                 coef exp(coef) se(coef)    z Pr(>|z|)
```

```
## val.linpred.gg 1.54      4.68     0.45 3.43    6e-04
##
##                 exp(coef) exp(-coef) lower .95 upper .95
## val.linpred.gg      4.68      0.214      1.94      11.3
##
## Concordance= 0.673  (se = 0.05 )
## Rsquare= 0.216   (max possible= 0.997 )
## Likelihood ratio test= 11.9  on 1 df,   p=0.000554
## Wald test            = 11.8  on 1 df,   p=0.000599
## Score (logrank) test = 12.2  on 1 df,   p=0.000485
```

```
summary(coxph(Surv(Time, DSD) ~ val.linpred.gg2, data.val))
```

```
## Call:
## coxph(formula = Surv(Time, DSD) ~ val.linpred.gg2, data = data.val)
##
##   n= 49, number of events= 49
##
##                  coef exp(coef) se(coef)    z Pr(>|z|)
## val.linpred.gg2 1.78      5.93     0.51 3.49  0.00048
##
##                  exp(coef) exp(-coef) lower .95 upper .95
## val.linpred.gg2      5.93      0.169      2.18      16.1
##
## Concordance= 0.668  (se = 0.05 )
## Rsquare= 0.216   (max possible= 0.997 )
## Likelihood ratio test= 11.9  on 1 df,   p=0.000563
## Wald test            = 12.2  on 1 df,   p=0.000483
## Score (logrank) test = 12.5  on 1 df,   p=0.00041
```

```
summary(coxph(Surv(Time, DSD) ~ val.linpred.cph, data.val))
```

```
## Call:
## coxph(formula = Surv(Time, DSD) ~ val.linpred.cph, data = data.val)
##
##   n= 49, number of events= 49
##
##                   coef exp(coef) se(coef)    z Pr(>|z|)
## val.linpred.cph 1.139     3.123    0.311 3.66  0.00025
##
##                  exp(coef) exp(-coef) lower .95 upper .95
## val.linpred.cph      3.12       0.32       1.7      5.75
##
## Concordance= 0.65  (se = 0.05 )
## Rsquare= 0.236   (max possible= 0.997 )
## Likelihood ratio test= 13.2  on 1 df,   p=0.000284
## Wald test            = 13.4  on 1 df,   p=0.000252
## Score (logrank) test = 13.9  on 1 df,   p=0.000192
```

```
summary(coxph(Surv(Time, DSD) ~ val.linpred.rsf, data.val))
```

```
## Call:
## coxph(formula = Surv(Time, DSD) ~ val.linpred.rsf, data = data.val)
##
##   n= 49, number of events= 49
```

```
##
##                     coef exp(coef) se(coef)    z Pr(>|z|)
## val.linpred.rsf 0.00811   1.00814  0.00209 3.87  0.00011
##
##                  exp(coef) exp(-coef) lower .95 upper .95
## val.linpred.rsf      1.01      0.992         1      1.01
##
## Concordance= 0.663  (se = 0.05 )
## Rsquare= 0.258   (max possible= 0.997 )
## Likelihood ratio test= 14.6  on 1 df,   p=0.000133
## Wald test            = 15  on 1 df,   p=0.000107
## Score (logrank) test = 15.5  on 1 df,   p=8.4e-05

anova(coxph(Surv(Time, DSD) ~ offset(val.linpred.gg) + val.linpred.gg, data.val))

## Analysis of Deviance Table
##  Cox model: response is Surv(Time, DSD)
## Terms added sequentially (first to last)
##
##                 loglik Chisq Df Pr(>|Chi|)
## NULL              -139
## val.linpred.gg    -139  1.47  1       0.23

anova(coxph(Surv(Time, DSD) ~ offset(val.linpred.gg2) + val.linpred.gg2, data.val))

## Analysis of Deviance Table
##  Cox model: response is Surv(Time, DSD)
## Terms added sequentially (first to last)
##
##                  loglik Chisq Df Pr(>|Chi|)
## NULL               -140
## val.linpred.gg2    -139  2.32  1       0.13

anova(coxph(Surv(Time, DSD) ~ offset(val.linpred.cph) + val.linpred.cph, data.val))

## Analysis of Deviance Table
##  Cox model: response is Surv(Time, DSD)
## Terms added sequentially (first to last)
##
##                  loglik Chisq Df Pr(>|Chi|)
## NULL               -138
## val.linpred.cph    -138   0.2  1       0.66

anova(coxph(Surv(Time, DSD) ~ offset(val.linpred.rsf) + val.linpred.rsf, data.val))

## Warning in fitter(X, Y, strats, offset, init, control, weights = weights, :  Ran out of
iterations and did not converge
## Error in fitter(X, Y, strats, offset, init, control, weights = weights, :  NA/NaN/Inf in
foreign function call (arg 6)

summary(coxph(Surv(Time, DSD) ~ offset(val.linpred.gg) + SexM + AgeCent + LocBody + SizeCent + A2 + A4,

## Call:
## coxph(formula = Surv(Time, DSD) ~ offset(val.linpred.gg) + SexM +
##     AgeCent + LocBody + SizeCent + A2 + A4, data = data.val)
##
```

```
##   n= 49, number of events= 49
##
##                 coef exp(coef) se(coef)     z Pr(>|z|)
## SexMTRUE     0.10665   1.11255  0.37675  0.28     0.78
## AgeCent     -0.00735   0.99268  0.02276 -0.32     0.75
## LocBodyTRUE  0.29902   1.34854  0.37945  0.79     0.43
## SizeCent     0.00391   1.00392  0.01002  0.39     0.70
## A2TRUE       0.30761   1.36017  0.49719  0.62     0.54
## A4TRUE       0.27581   1.31760  0.39889  0.69     0.49
##
##             exp(coef) exp(-coef) lower .95 upper .95
## SexMTRUE        1.113      0.899     0.532      2.33
## AgeCent         0.993      1.007     0.949      1.04
## LocBodyTRUE     1.349      0.742     0.641      2.84
## SizeCent        1.004      0.996     0.984      1.02
## A2TRUE          1.360      0.735     0.513      3.60
## A4TRUE          1.318      0.759     0.603      2.88
##
## Concordance= 0.672  (se = 0.05 )
## Rsquare= 0.064   (max possible= 0.997 )
## Likelihood ratio test= 3.25  on 6 df,   p=0.777
## Wald test            = 3.3  on 6 df,   p=0.77
## Score (logrank) test = 3.36  on 6 df,   p=0.763
```

```r
summary(coxph(Surv(Time, DSD) ~ offset(val.linpred.gg2) + SexM + AgeCent + LocBody + SizeCent + A2 + A4,
```

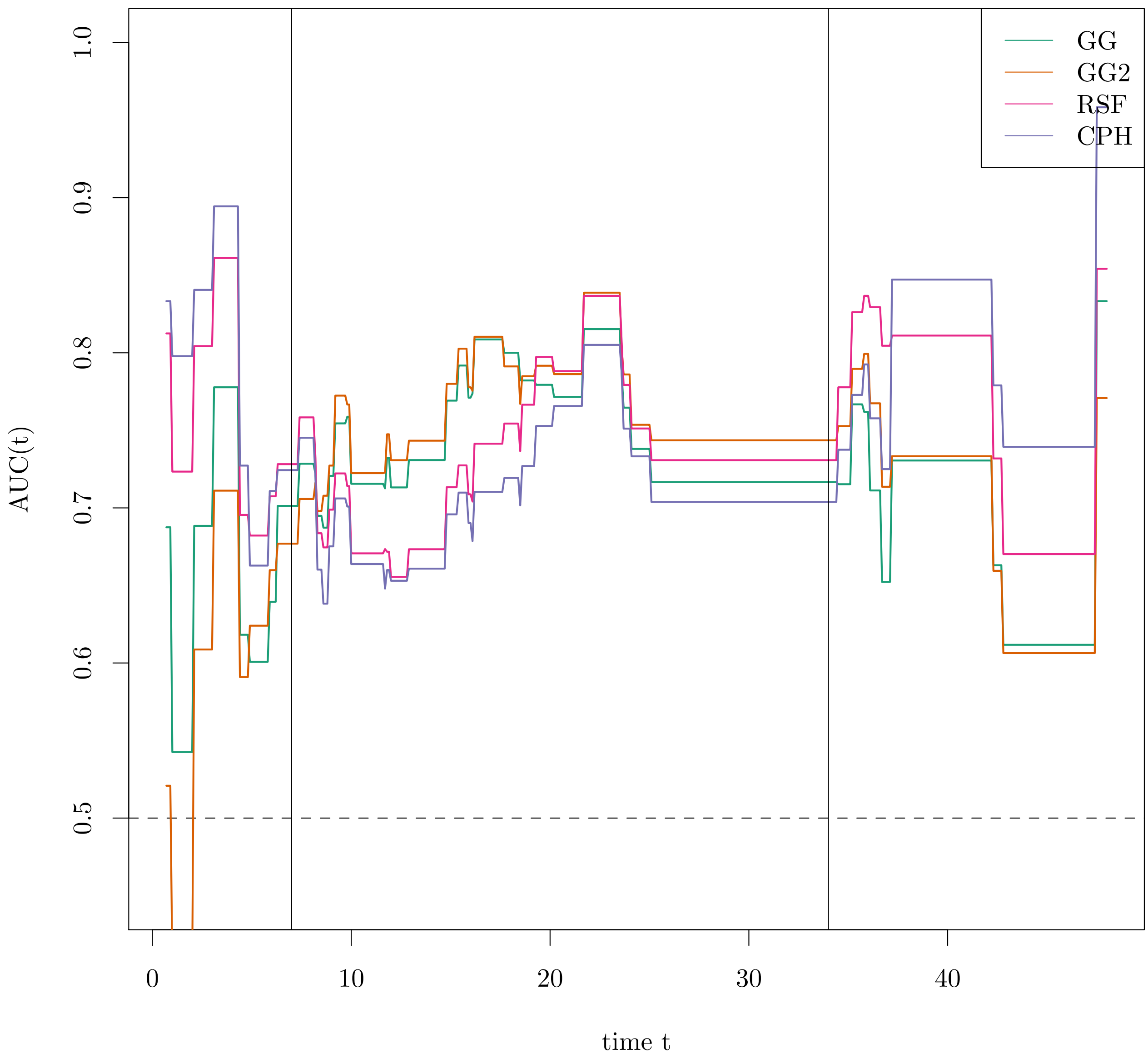
```
## Call:
## coxph(formula = Surv(Time, DSD) ~ offset(val.linpred.gg2) + SexM +
##     AgeCent + LocBody + SizeCent + A2 + A4, data = data.val)
##
##   n= 49, number of events= 49
##
##                coef exp(coef) se(coef)    z Pr(>|z|)
## SexMTRUE    0.14695   1.15830  0.37675 0.39     0.70
## AgeCent     0.00300   1.00301  0.02276 0.13     0.90
## LocBodyTRUE 0.23722   1.26772  0.37945 0.63     0.53
## SizeCent    0.00846   1.00849  0.01002 0.84     0.40
## A2TRUE      0.33860   1.40298  0.49719 0.68     0.50
## A4TRUE      0.31901   1.37576  0.39889 0.80     0.42
##
##             exp(coef) exp(-coef) lower .95 upper .95
## SexMTRUE         1.16      0.863     0.554      2.42
## AgeCent          1.00      0.997     0.959      1.05
## LocBodyTRUE      1.27      0.789     0.603      2.67
## SizeCent         1.01      0.992     0.989      1.03
## A2TRUE           1.40      0.713     0.529      3.72
## A4TRUE           1.38      0.727     0.630      3.01
##
## Concordance= 0.672  (se = 0.05 )
## Rsquare= 0.081   (max possible= 0.997 )
## Likelihood ratio test= 4.13  on 6 df,   p=0.659
## Wald test            = 4.14  on 6 df,   p=0.658
## Score (logrank) test = 4.23  on 6 df,   p=0.646
```

```r
summary(coxph(Surv(Time, DSD) ~ offset(val.linpred.cph) + SexM + AgeCent + LocBody + SizeCent + A2 + A4,
```

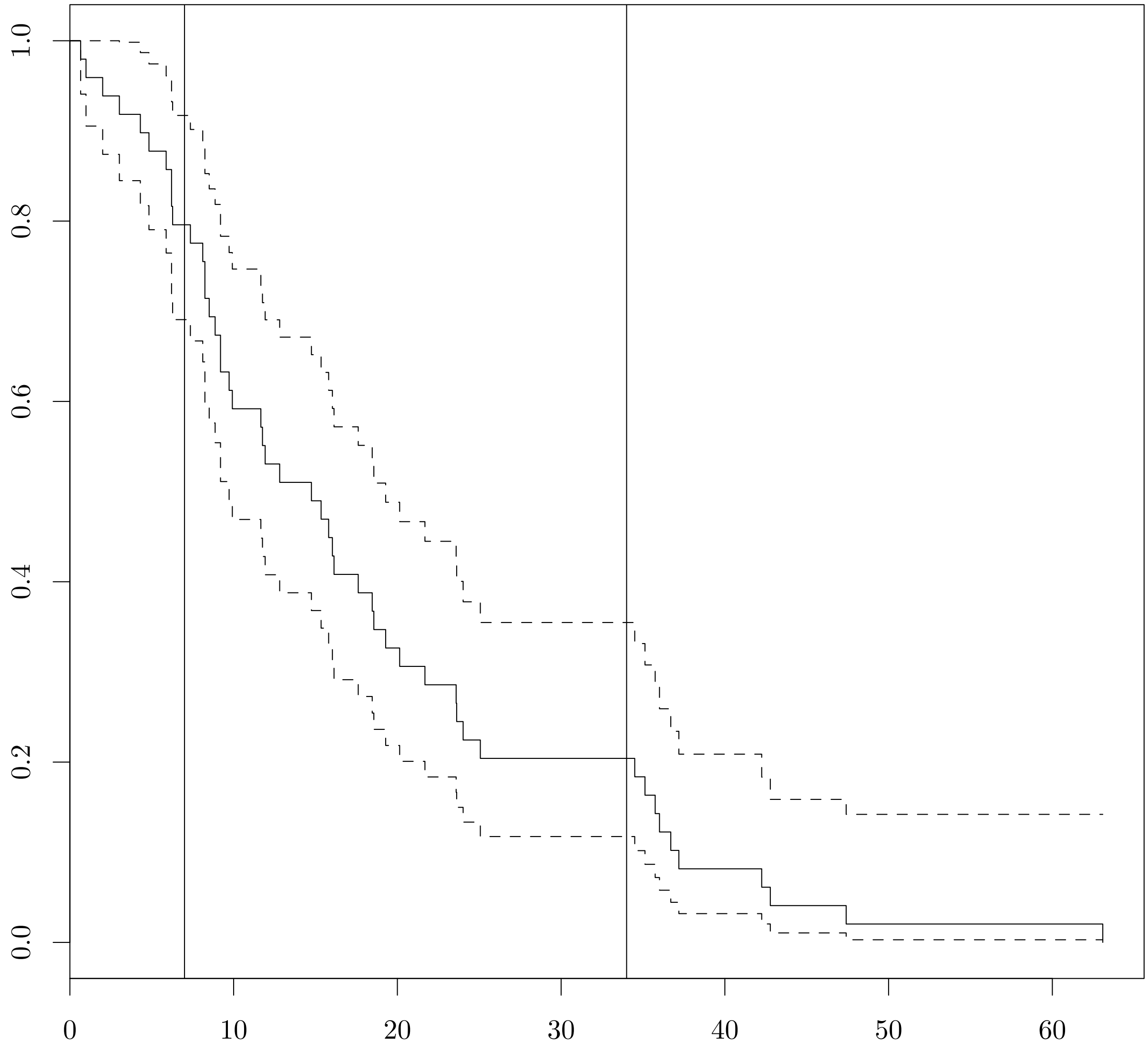
```
## Call:
## coxph(formula = Surv(Time, DSD) ~ offset(val.linpred.cph) + SexM +
##     AgeCent + LocBody + SizeCent + A2 + A4, data = data.val)
##
##   n= 49, number of events= 49
##
##                   coef exp(coef)  se(coef)     z Pr(>|z|)
## SexMTRUE    -2.37e-01  7.89e-01  3.77e-01 -0.63     0.53
## AgeCent     -7.35e-03  9.93e-01  2.28e-02 -0.32     0.75
## LocBodyTRUE  1.28e-01  1.14e+00  3.79e-01  0.34     0.74
## SizeCent     5.99e-05  1.00e+00  1.00e-02  0.01     1.00
## A2TRUE       6.71e-02  1.07e+00  4.97e-01  0.13     0.89
## A4TRUE       1.42e-01  1.15e+00  3.99e-01  0.36     0.72
##
##             exp(coef) exp(-coef) lower .95 upper .95
## SexMTRUE        0.789      1.267     0.377      1.65
## AgeCent         0.993      1.007     0.949      1.04
## LocBodyTRUE     1.137      0.880     0.540      2.39
## SizeCent        1.000      1.000     0.981      1.02
## A2TRUE          1.069      0.935     0.404      2.83
## A4TRUE          1.152      0.868     0.527      2.52
##
## Concordance= 0.672  (se = 0.05 )
## Rsquare= 0.015   (max possible= 0.996 )
## Likelihood ratio test= 0.73  on 6 df,   p=0.994
## Wald test            = 0.72  on 6 df,   p=0.994
## Score (logrank) test = 0.72  on 6 df,   p=0.994

summary(coxph(Surv(Time, DSD) ~ offset(val.linpred.rsf) + SexM + AgeCent + LocBody + SizeCent + A2 + A4,

## Warning in fitter(X, Y, strats, offset, init, control, weights = weights, :  Ran out of
iterations and did not converge
## Error in fitter(X, Y, strats, offset, init, control, weights = weights, :  NA/NaN/Inf in
foreign function call (arg 6)
```

TD-ROC AUC

```
temp.times = seq(0.1, 48, 0.1)
temp.gg = timeROC(T = data.val$Time/365.25*12, delta = data.val$DSD*1, marker = val.linpred.gg, cause =
temp.gg2 = timeROC(T = data.val$Time/365.25*12, delta = data.val$DSD*1, marker = val.linpred.gg2, cause
temp.rsf = timeROC(T = data.val$Time/365.25*12, delta = data.val$DSD*1, marker = val.linpred.rsf, cause
temp.cph = timeROC(T = data.val$Time/365.25*12, delta = data.val$DSD*1, marker = val.linpred.cph, cause
plotAUCcurve(temp.gg, conf.int = FALSE, add = FALSE, col = pal["GG"])
plotAUCcurve(temp.gg2, conf.int = FALSE, add = TRUE, col = pal["GG2"])
plotAUCcurve(temp.rsf, conf.int = FALSE, add = TRUE, col = pal["RSF"])
plotAUCcurve(temp.cph, conf.int = FALSE, add = TRUE, col = pal["CPH"])
legend("topright", legend = c("GG", "GG2", "RSF", "CPH"), col = pal[c("GG", "GG2", "RSF", "CPH")], lty =
abline(v = c(7, 34))
```

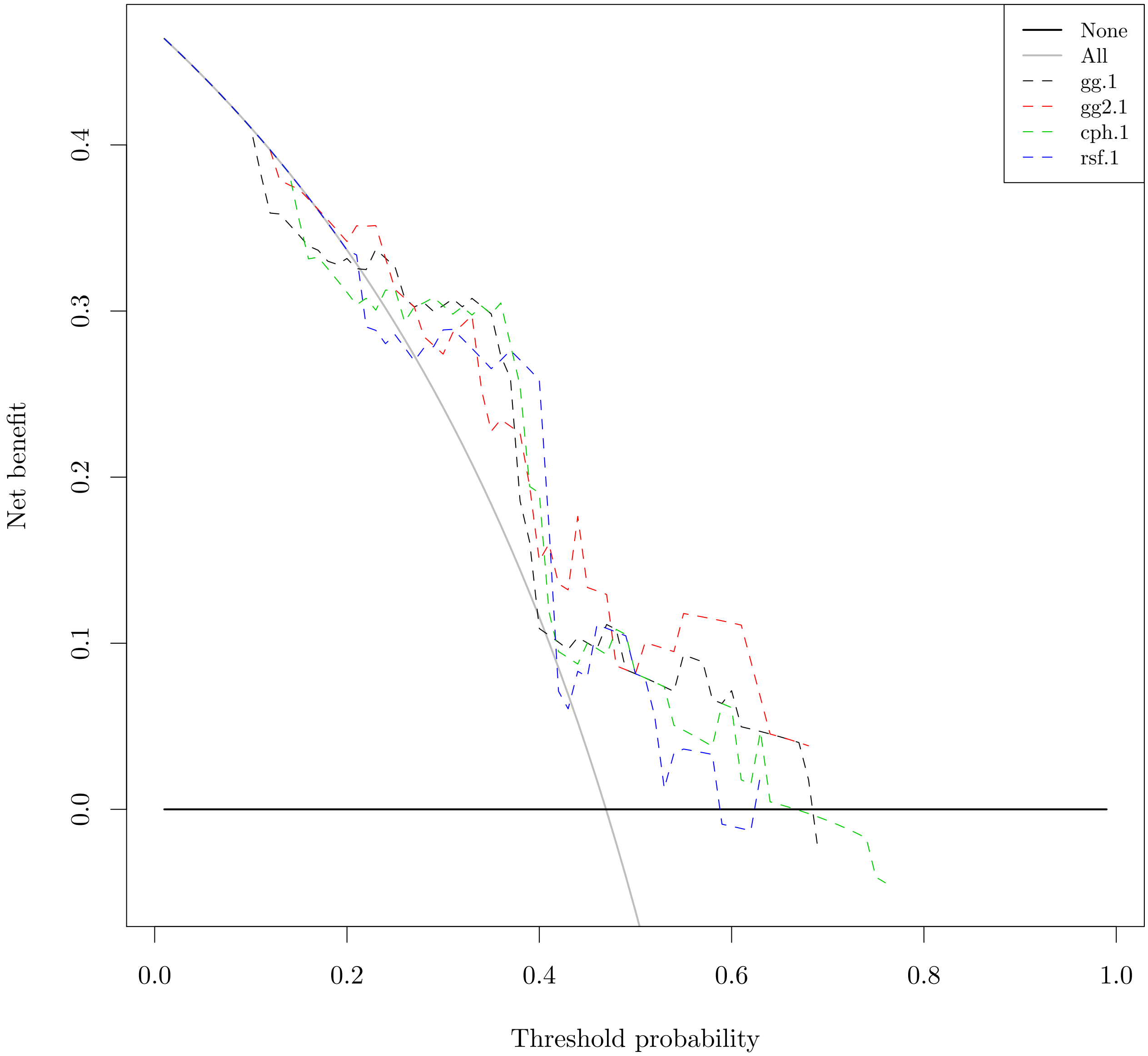



```
plot(survfit(Surv(data.val$Time/365.25*12, data.val$DSD) ~ 1))
abline(v = c(7, 34))
```

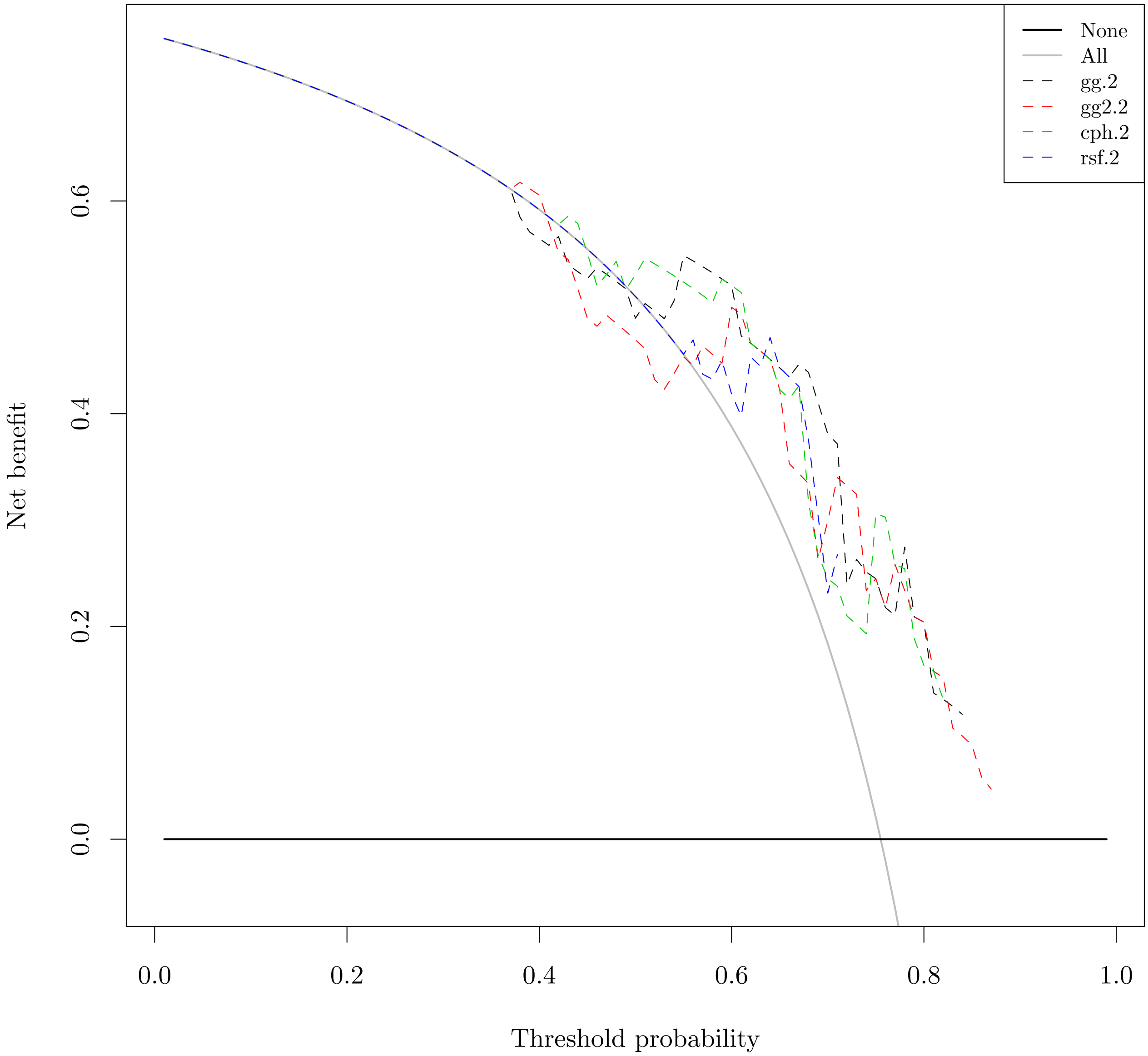


Decision curve analysis.

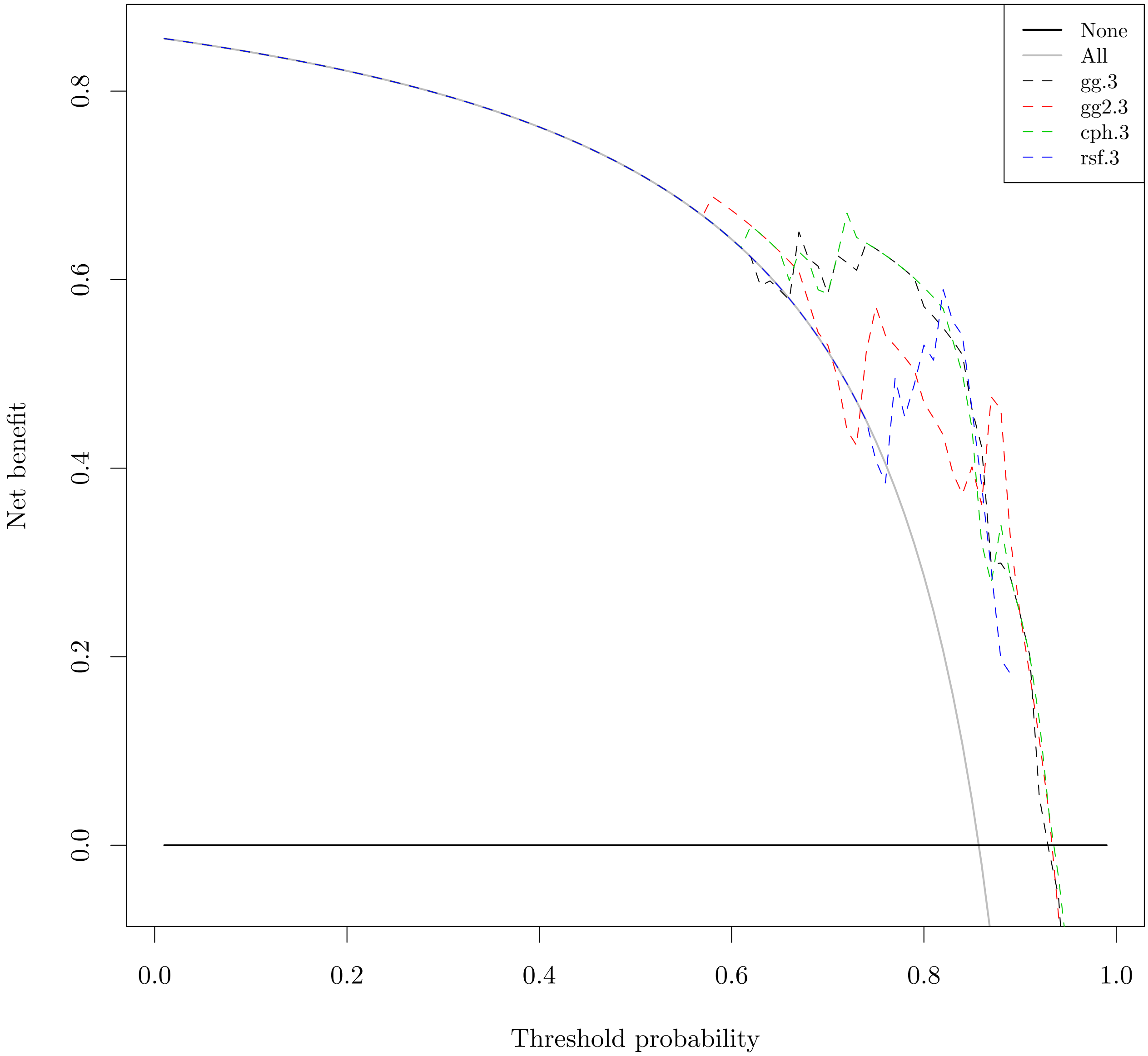
```
source("stdca.R")
temp.data = data.frame(Time = data.val$Time, DSD = data.val$DSD*1,
    gg.1 = 1-val.prob.gg[val.prob.times == 365,], gg.2 = 1-val.prob.gg[val.prob.times == 365*2,], gg.3 =
    gg2.1 = 1-val.prob.gg2[val.prob.times == 365,], gg2.2 = 1-val.prob.gg2[val.prob.times == 365*2,], gg
    cph.1 = 1-val.prob.cph[val.prob.times == 365,], cph.2 = 1-val.prob.cph[val.prob.times == 365*2,], cp
    rsf.1 = 1-val.prob.rsf[val.prob.times == 365,], rsf.2 = 1-val.prob.rsf[val.prob.times == 365*2,], rs
invisible(stdca(data = temp.data, outcome = "DSD", ttoutcome = "Time", predictors = c("gg.1", "gg2.1", "

## [1] "gg.1: No observations with risk greater than 70% that have followup through the timepoint select
## [2] "gg2.1: No observations with risk greater than 69% that have followup through the timepoint selec
## [3] "cph.1: No observations with risk greater than 77% that have followup through the timepoint selec
## [4] "rsf.1: No observations with risk greater than 64%, and therefore net benefit not calculable in t
```




```
invisible(stdca(data = temp.data, outcome = "DSD", ttoutcome = "Time", predictors = c("gg.2", "gg2.2", "

## [1] "gg.2: No observations with risk greater than 85% that have followup through the timepoint select
## [2] "gg2.2: No observations with risk greater than 88% that have followup through the timepoint selec
## [3] "cph.2: No observations with risk greater than 83% that have followup through the timepoint selec
## [4] "rsf.2: No observations with risk greater than 72% that have followup through the timepoint selec
```




```
invisible(stdca(data = temp.data, outcome = "DSD", ttoutcome = "Time", predictors = c("gg.3", "gg2.3", "

## [1] "gg.3: No observations with risk greater than 97% that have followup through the timepoint select
## [2] "gg2.3: No observations with risk greater than 99% that have followup through the timepoint selec
## [3] "cph.3: No observations with risk greater than 97% that have followup through the timepoint selec
## [4] "rsf.3: No observations with risk greater than 90% that have followup through the timepoint selec
```

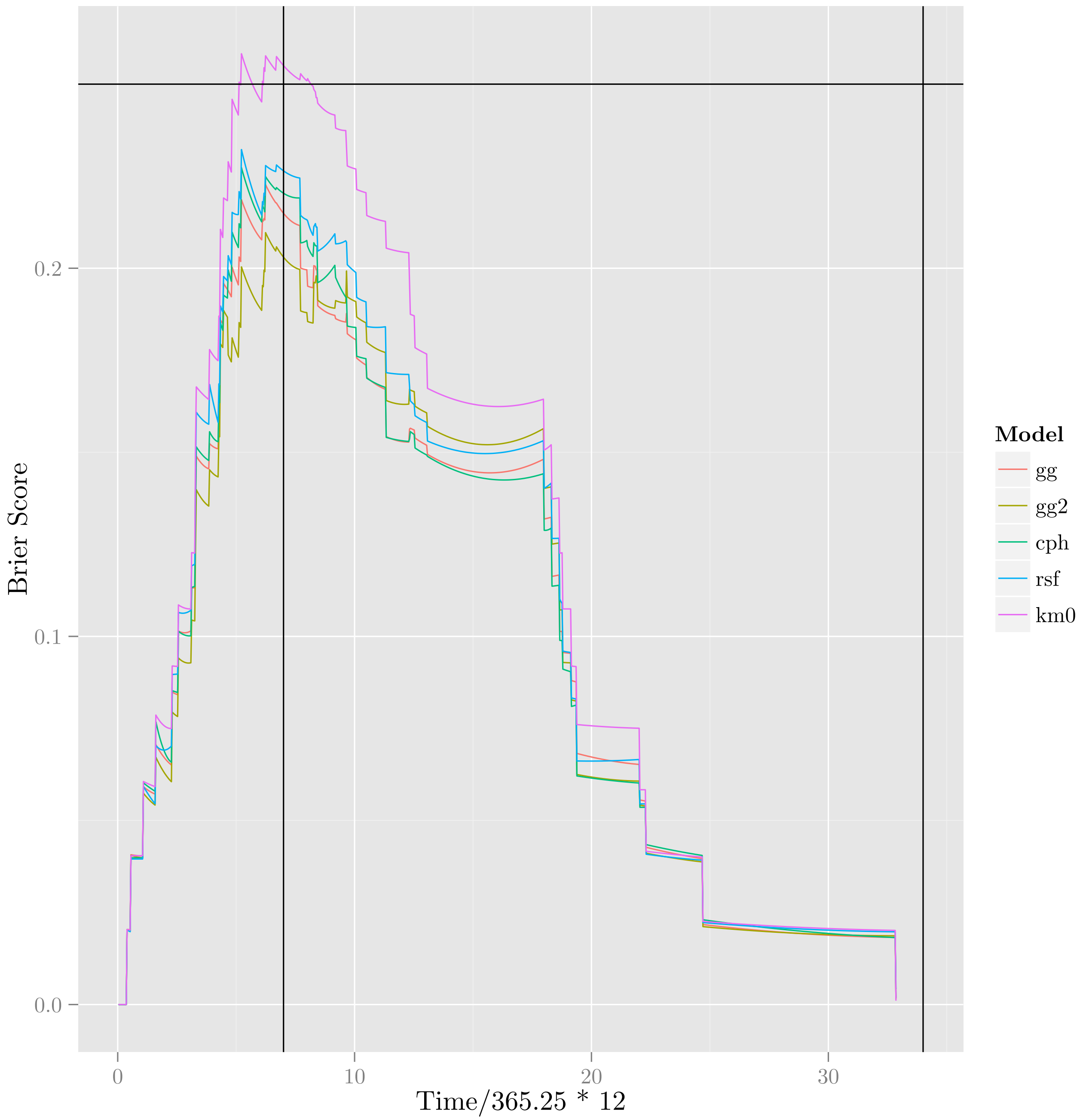


Evaluate IBS point estimates. BS paths over time on bootstrap samples of the holdout set.

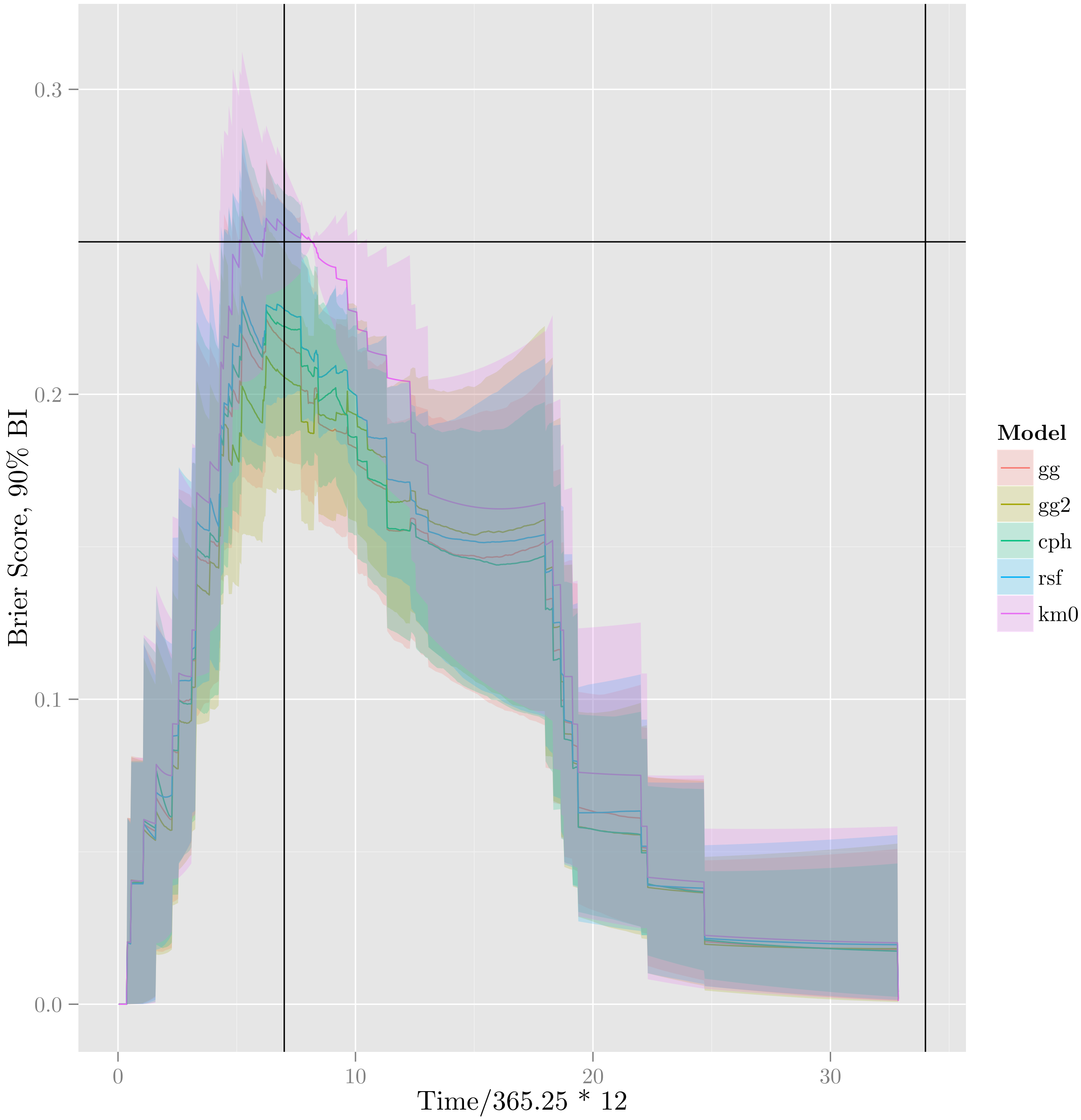
```
set.seed(20150208)
ibs_eval_times = calcIBS(Surv(data.val$Time, data.val$DSD), ibs_preds_gg, ibs_times, max(data.val$Time))
bsc_boots = laply(1:500, function(i) {
        if (i %% 50 == 0)       { message(i) }
        boot_samp = sample.int(nrow(data.val), replace = TRUE)
        gg = calcIBS(Surv(data.val$Time, data.val$DSD)[boot_samp,], ibs_preds_gg[boot_samp,], ibs_times,
        gg2 = calcIBS(Surv(data.val$Time, data.val$DSD)[boot_samp,], ibs_preds_gg2[boot_samp,], ibs_time
        cph = calcIBS(Surv(data.val$Time, data.val$DSD)[boot_samp,], ibs_preds_cph[boot_samp,], ibs_time
        rsf = calcIBS(Surv(data.val$Time, data.val$DSD)[boot_samp,], ibs_preds_rsf[boot_samp,], ibs_time
        km0 = calcIBS(Surv(data.val$Time, data.val$DSD)[boot_samp,], ibs_preds_km0[boot_samp,], ibs_time
        rbind(gg, gg2, cph, rsf, km0)
})

## 50
## 100
## 150
```

```
## 200
## 250
## 300
## 350
## 400
## 450
## 500
```

```
temp = sapply(list(gg = ibs_preds_gg, gg2 = ibs_preds_gg2, cph = ibs_preds_cph, rsf = ibs_preds_rsf, km(
temp = melt(temp)
colnames(temp) = c("Time", "Model", "BS")
ggplot(temp, aes(x = Time/365.25*12, y = BS, colour = Model)) + geom_line() + ylab("Brier Score") + geom
```



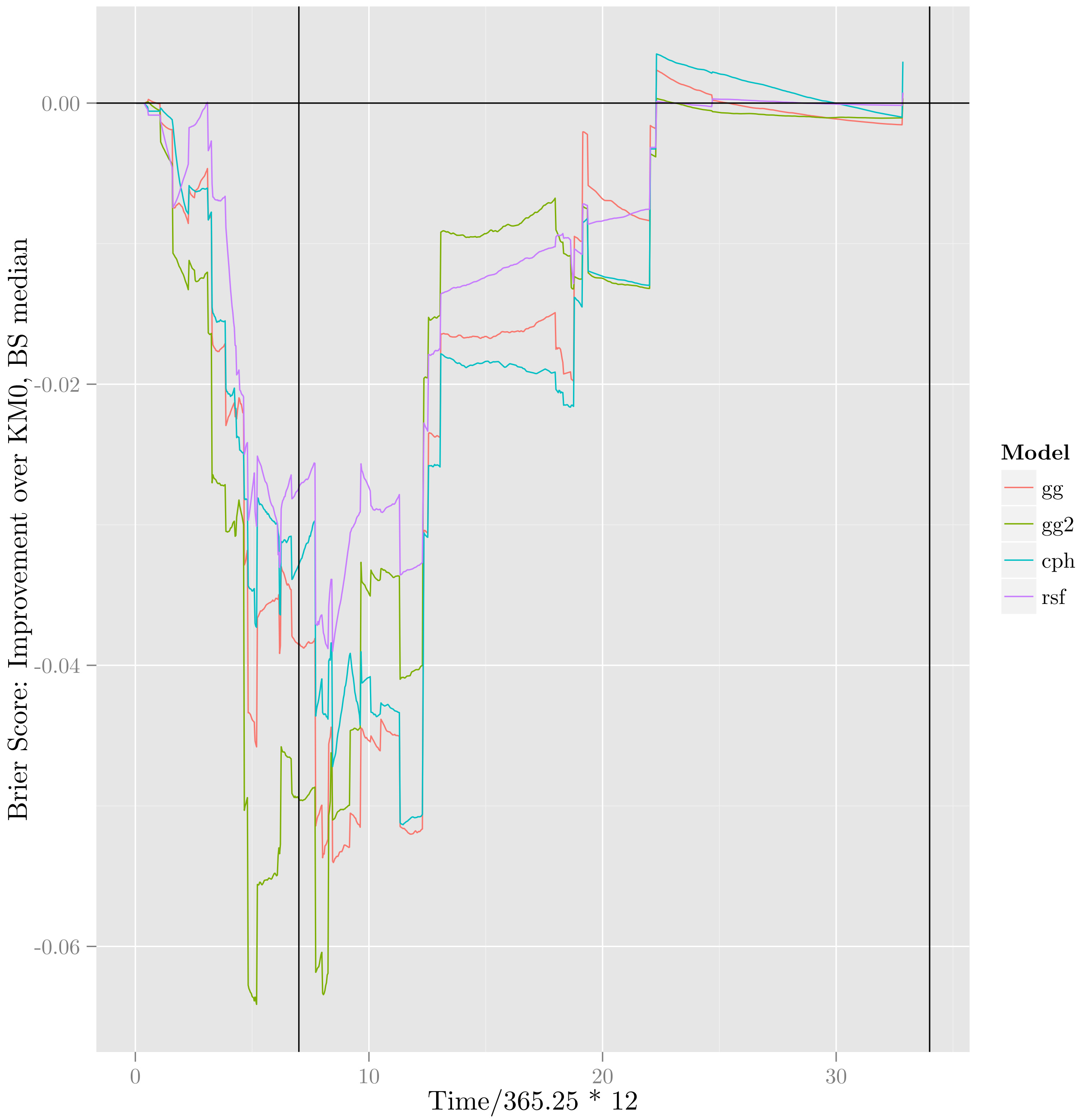
```
temp = melt(aaply(bsc_boots, 2:3, quantile, probs = c(0.05, 0.5, 0.95)))
colnames(temp) = c("Model", "Time", "Quantile", "Value")
temp$Quantile = paste("Q", gsub("%", "", temp$Quantile), sep = "")
temp = dcast(temp, Model + Time ~ Quantile, value.var = "Value")
ggplot(temp, aes(x = Time/365.25*12, y = Q50, ymin = Q5, ymax = Q95, colour = Model, fill = Model)) + ge
```



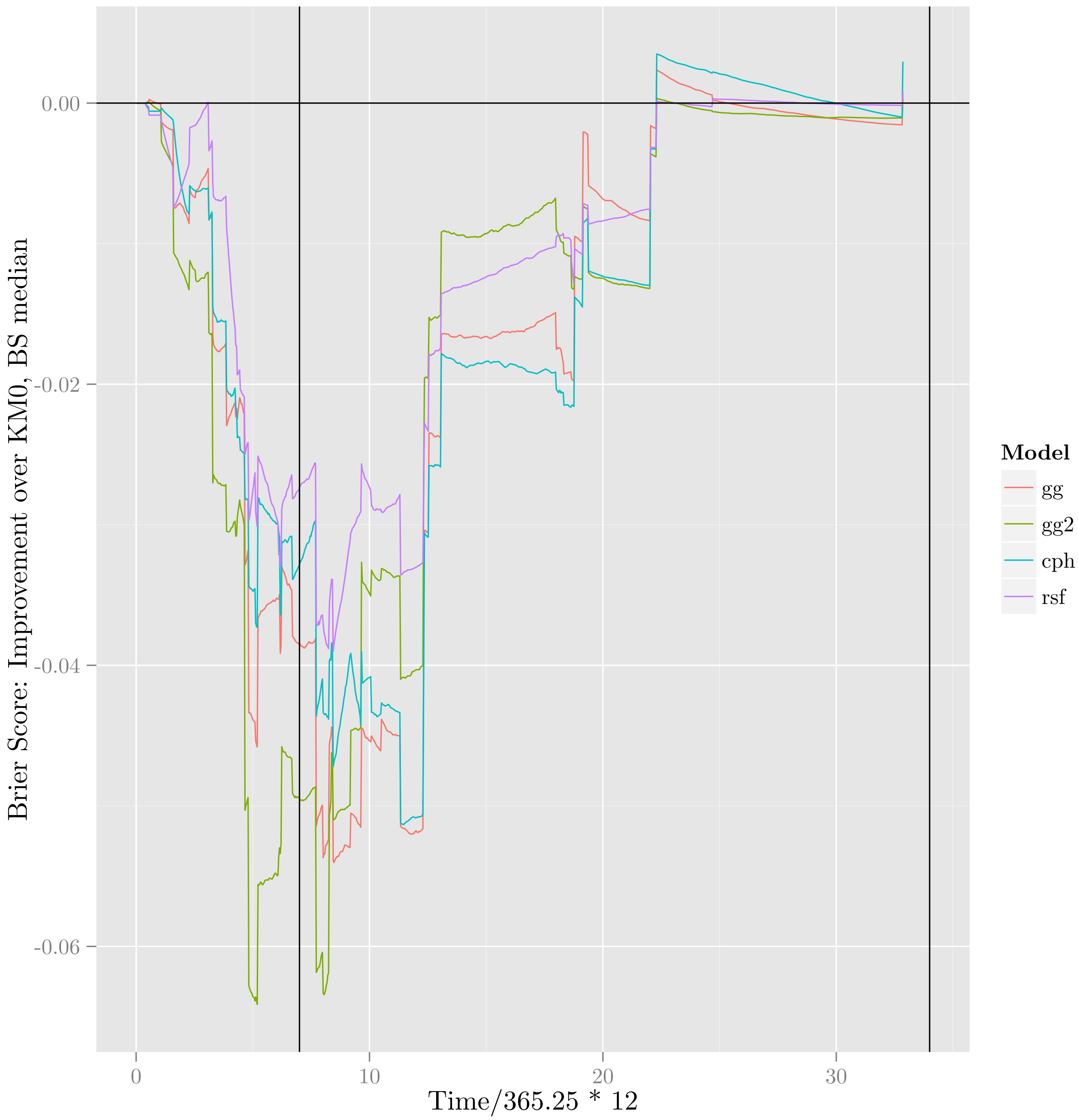

```
bsc_boots_diff = aaply(bsc_boots, 2, function(x) x - bsc_boots[,5,])[1:4,,]
temp = melt(aaply(bsc_boots_diff, c(1,3), quantile, probs = c(0.05, 0.5, 0.95)))
colnames(temp) = c("Model", "Time", "Quantile", "Value")
temp$Quantile = paste("Q", gsub("%", "", temp$Quantile), sep = "")
temp = dcast(temp, Model + Time ~ Quantile, value.var = "Value")
ggplot(temp, aes(x = Time/365.25*12, y = Q50, ymin = Q5, ymax = Q95, colour = Model, fill = Model)) + ge
```


```
ggplot(temp, aes(x = Time/365.25*12, y = Q50, ymin = Q5, ymax = Q95, colour = Model, fill = Model)) + ge
```


```
ggplot(temp, aes(x = Time/365.25*12, y = Q50, colour = Model)) + geom_line() + ylab("Brier Score: Improv
```




```
ggplot(temp, aes(x = Time/365.25*12, y = Q50, colour = Model)) + geom_line() + ylab("Brier Score: Improv
```

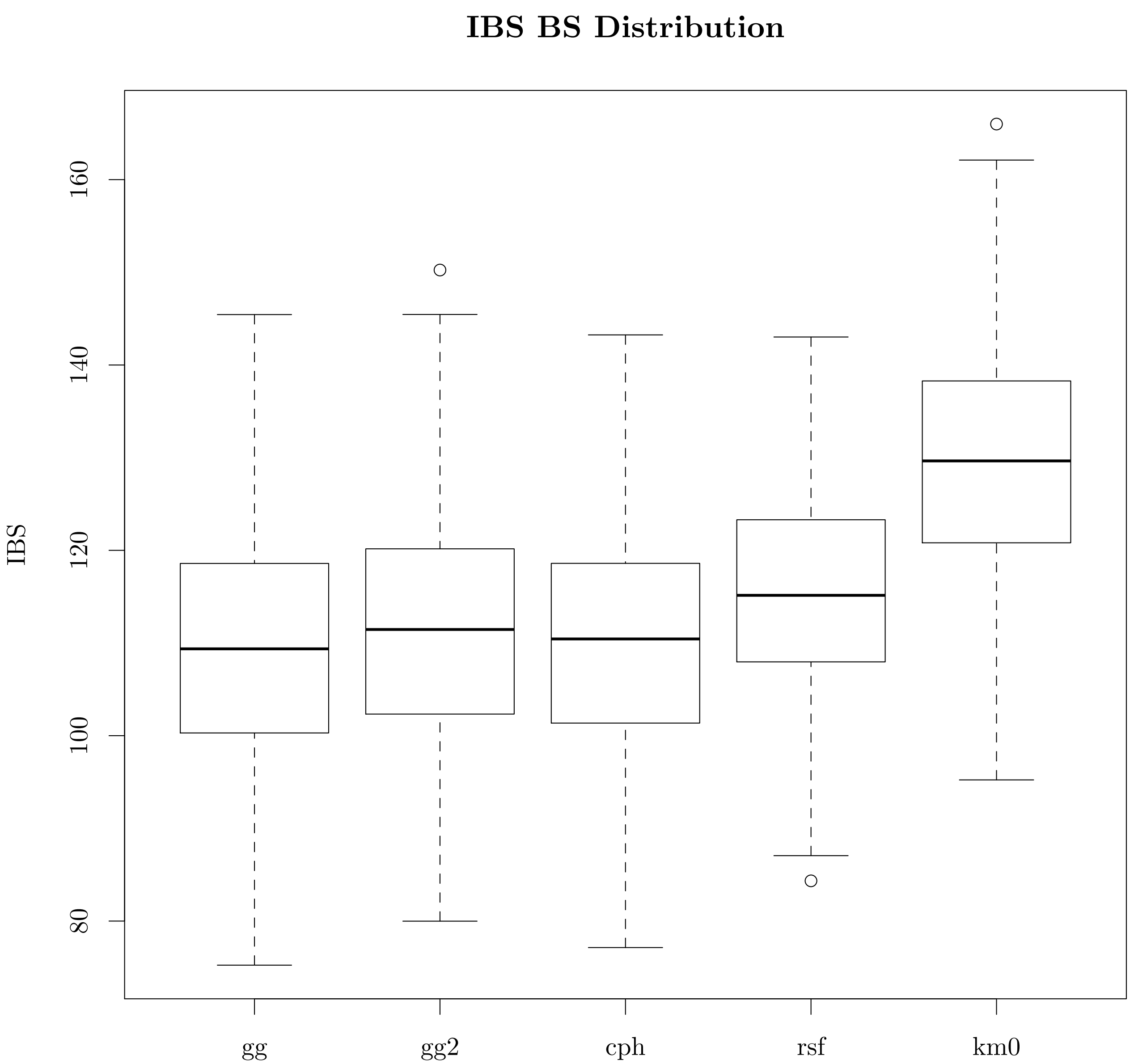


IBS comparisons.

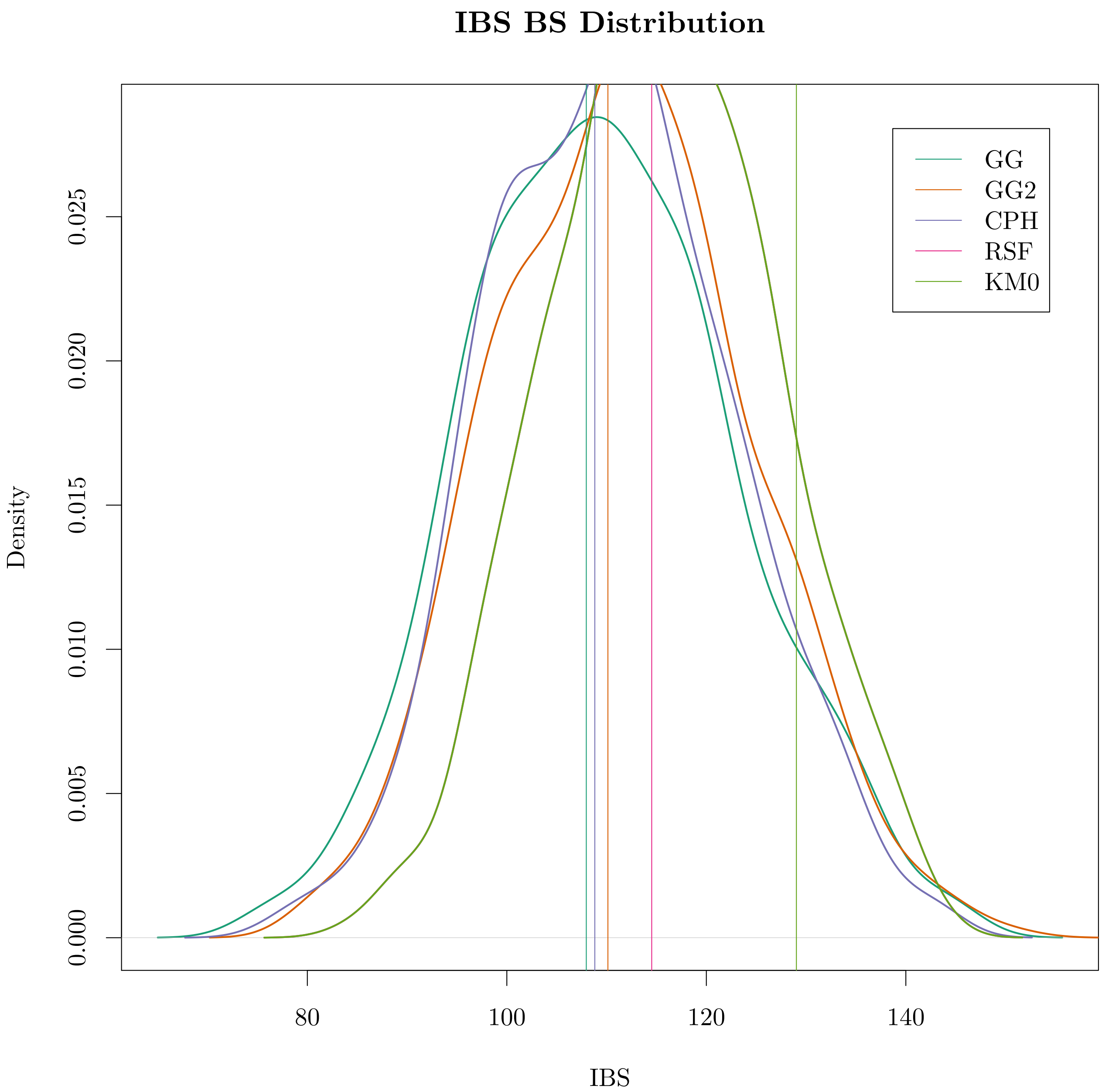
```
set.seed(20150208)
ibsc_boots = t(sapply(1:5e2, function(i) {
        if (i %% 5e1 == 0)      { message(i) }
        boot_samp = sample.int(nrow(data.val), replace = TRUE)
        gg = calcIBS(Surv(data.val$Time, data.val$DSD)[boot_samp,], ibs_preds_gg[boot_samp,], ibs_times,
        gg2 = calcIBS(Surv(data.val$Time, data.val$DSD)[boot_samp,], ibs_preds_gg2[boot_samp,], ibs_time
        cph = calcIBS(Surv(data.val$Time, data.val$DSD)[boot_samp,], ibs_preds_cph[boot_samp,], ibs_time
        rsf = calcIBS(Surv(data.val$Time, data.val$DSD)[boot_samp,], ibs_preds_rsf[boot_samp,], ibs_time
        km0 = calcIBS(Surv(data.val$Time, data.val$DSD)[boot_samp,], ibs_preds_km0[boot_samp,], ibs_time
        c(gg, gg2, cph, rsf, km0)
}))
```

```
## 50
## 100
## 150
## 200
## 250
## 300
## 350
## 400
## 450
## 500
colnames(ibsc_boots) = c("gg", "gg2", "cph", "rsf", "km0")
```

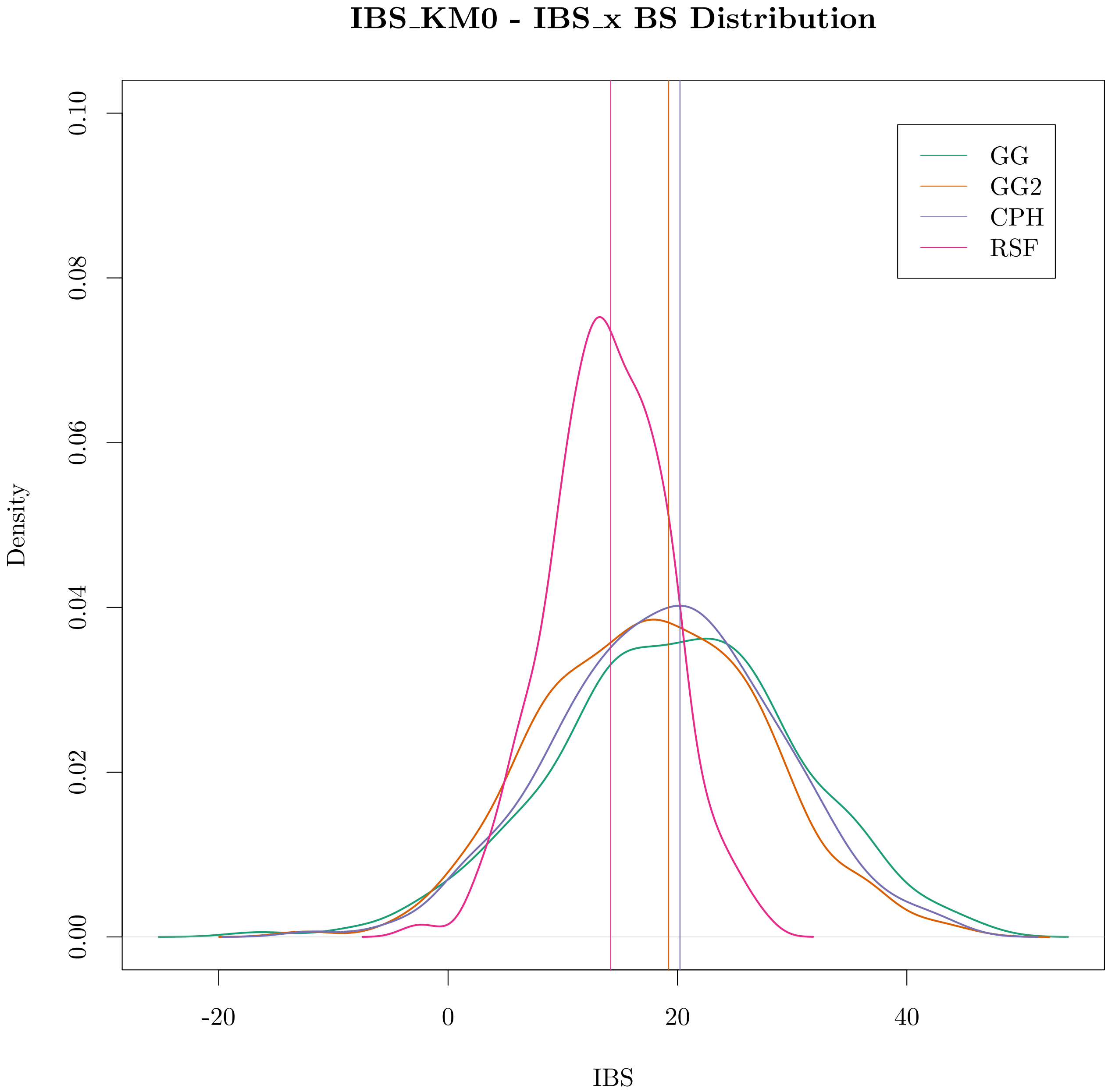
```
calcIBS(Surv(data.val$Time, data.val$DSD), ibs_preds_gg, ibs_times, 34*365.25/12, 7*365.25/12)$ibs

## [1] 108

calcIBS(Surv(data.val$Time, data.val$DSD), ibs_preds_gg2, ibs_times, 34*365.25/12, 7*365.25/12)$ibs

## [1] 110.1

calcIBS(Surv(data.val$Time, data.val$DSD), ibs_preds_cph, ibs_times, 34*365.25/12, 7*365.25/12)$ibs

## [1] 108.8

calcIBS(Surv(data.val$Time, data.val$DSD), ibs_preds_rsf, ibs_times, 34*365.25/12, 7*365.25/12)$ibs

## [1] 114.5

calcIBS(Surv(data.val$Time, data.val$DSD), ibs_preds_km0, ibs_times, 34*365.25/12, 7*365.25/12)$ibs

## [1] 129

boxplot(ibsc_boots, main = "IBS BS Distribution", ylab = "IBS")
```




```
plot(density(ibsc_boots[,1]), col = pal["GG"], lwd = 2, main = "IBS BS Distribution", xlab = "IBS")
lines(density(ibsc_boots[,2]), col = pal["GG2"], lwd = 2)
lines(density(ibsc_boots[,3]), col = pal["CPH"], lwd = 2)
lines(density(ibsc_boots[,4]), col = pal["RSF"], lwd = 2)
lines(density(ibsc_boots[,4]), col = pal["KM0"], lwd = 2)
abline(v = calcIBS(Surv(data.val$Time, data.val$DSD), ibs_preds_gg, ibs_times, 34*365.25/12, 7*365.25/12
abline(v = calcIBS(Surv(data.val$Time, data.val$DSD), ibs_preds_gg2, ibs_times, 34*365.25/12, 7*365.25/1
abline(v = calcIBS(Surv(data.val$Time, data.val$DSD), ibs_preds_cph, ibs_times, 34*365.25/12, 7*365.25/1
abline(v = calcIBS(Surv(data.val$Time, data.val$DSD), ibs_preds_rsf, ibs_times, 34*365.25/12, 7*365.25/1
abline(v = calcIBS(Surv(data.val$Time, data.val$DSD), ibs_preds_km0, ibs_times, 34*365.25/12, 7*365.25/1
legend("topright", legend = c("GG", "GG2", "CPH", "RSF", "KM0"), col = pal[c("GG", "GG2", "CPH", "RSF",
```




```r
plot(density(ibsc_boots[,5] - ibsc_boots[,1]), col = pal["GG"], lwd = 2, main = "IBS\\_KM0 - IBS\\_x BS
lines(density(ibsc_boots[,5] - ibsc_boots[,2]), col = pal["GG2"], lwd = 2)
lines(density(ibsc_boots[,5] - ibsc_boots[,3]), col = pal["CPH"], lwd = 2)
lines(density(ibsc_boots[,5] - ibsc_boots[,4]), col = pal["RSF"], lwd = 2)
abline(v = (calcIBS(Surv(data.val$Time, data.val$DSD), ibs_preds_km0, ibs_times, max(data.val$Time))$ibs
abline(v = (calcIBS(Surv(data.val$Time, data.val$DSD), ibs_preds_km0, ibs_times, max(data.val$Time))$ibs
abline(v = (calcIBS(Surv(data.val$Time, data.val$DSD), ibs_preds_km0, ibs_times, max(data.val$Time))$ibs
abline(v = (calcIBS(Surv(data.val$Time, data.val$DSD), ibs_preds_km0, ibs_times, max(data.val$Time))$ibs
legend("topright", legend = c("GG", "GG2", "CPH", "RSF"), col = pal[c("GG", "GG2", "CPH", "RSF")], lty =
```



Do some proper BCA bootstrapping on the differences, just as a double-check test.

```
set.seed(20150208)
ibsc_boots2 = boot(data.val, statistic = function(d, i) {
        gg = calcIBS(Surv(d$Time, d$DSD)[i,], ibs_preds_gg[i,], ibs_times, 34*365.25/12, 7*365.25/12)$ib
        gg2 = calcIBS(Surv(d$Time, d$DSD)[i,], ibs_preds_gg2[i,], ibs_times, 34*365.25/12, 7*365.25/12)$
        cph = calcIBS(Surv(d$Time, d$DSD)[i,], ibs_preds_cph[i,], ibs_times, 34*365.25/12, 7*365.25/12)$
        rsf = calcIBS(Surv(d$Time, d$DSD)[i,], ibs_preds_rsf[i,], ibs_times, 34*365.25/12, 7*365.25/12)$
        km0 = calcIBS(Surv(d$Time, d$DSD)[i,], ibs_preds_km0[i,], ibs_times, 34*365.25/12, 7*365.25/12)$
        c(gg - km0, gg2 - km0, cph - km0, rsf - km0, gg - rsf, gg2 - rsf, cph - rsf, gg - cph, gg2 - cph
}, R = 500)
ibsc_boots2_ci = t(sapply(1:length(ibsc_boots2$t0), function(i) boot.ci(ibsc_boots2, index = i, type = '
rownames(ibsc_boots2_ci) = c("gg-km0", "gg2-km0", "cph-km0", "rsf-km0", "gg-rsf", "gg2-rsf", "cph-rsf",
colnames(ibsc_boots2_ci) = c("level", "orderi1", "orderi2", "lci", "uci")
ibsc_boots2
```

```
##
## ORDINARY NONPARAMETRIC BOOTSTRAP
##
##
## Call:
## boot(data = data.val, statistic = function(d, i) {
##     gg = calcIBS(Surv(d$Time, d$DSD)[i, ], ibs_preds_gg[i, ],
##         ibs_times, 34 * 365.25/12, 7 * 365.25/12)$ibs
##     gg2 = calcIBS(Surv(d$Time, d$DSD)[i, ], ibs_preds_gg2[i,
##         ], ibs_times, 34 * 365.25/12, 7 * 365.25/12)$ibs
##     cph = calcIBS(Surv(d$Time, d$DSD)[i, ], ibs_preds_cph[i,
##         ], ibs_times, 34 * 365.25/12, 7 * 365.25/12)$ibs
##     rsf = calcIBS(Surv(d$Time, d$DSD)[i, ], ibs_preds_rsf[i,
##         ], ibs_times, 34 * 365.25/12, 7 * 365.25/12)$ibs
##     km0 = calcIBS(Surv(d$Time, d$DSD)[i, ], ibs_preds_km0[i,
##         ], ibs_times, 34 * 365.25/12, 7 * 365.25/12)$ibs
##     c(gg - km0, gg2 - km0, cph - km0, rsf - km0, gg - rsf, gg2 -
##         rsf, cph - rsf, gg - cph, gg2 - cph, gg - gg2)
## }, R = 500)
##
##
## Bootstrap Statistics :
##      original    bias    std. error
## t1*   -21.062  1.16153      10.132
## t2*   -18.895  1.10126       9.319
## t3*   -20.209  1.14810       9.399
## t4*   -14.505  0.54879       5.156
## t5*    -6.557  0.61274       5.820
## t6*    -4.390  0.55247       5.219
## t7*    -5.704  0.59930       4.888
## t8*    -0.853  0.01344       2.103
## t9*     1.314 -0.04683       4.374
## t10*   -2.167  0.06027       5.015
```

```
ibsc_boots2_ci
```

```
##          level orderi1 orderi2     lci    uci
## gg-km0    0.95    8.84   483.4 -42.087 -2.195
## gg2-km0   0.95    6.04   477.0 -39.834 -2.575
## cph-km0   0.95    4.95   473.6 -41.625 -4.555
## rsf-km0   0.95    3.86   469.0 -26.894 -6.234
## gg-rsf    0.95    9.64   484.5 -17.396  4.837
## gg2-rsf   0.95   14.55   490.4 -13.874  6.460
## cph-rsf   0.95    7.02   479.6 -15.948  2.835
## gg-cph    0.95   19.34   493.6  -4.524  4.448
## gg2-cph   0.95   13.62   489.5  -6.992 10.142
## gg-gg2    0.95   16.42   491.8 -11.578  9.249
```

All models perform equivalently on the validation set. Select the simplest: gg.

Final model fitting:

```
data.all = rbind(data[colnames(data.val)], data.val)
head(data.all)
```

```
##           Time  DSD DiagYearCent  SexM AgeCent LocBody SizeCent    A2   A4
```

```
## NSWPCN_4   937 TRUE       -5.717  TRUE     -16   FALSE       -1 FALSE TRUE
## NSWPCN_9   587 TRUE       -7.173  TRUE       5   FALSE       10 FALSE TRUE
## NSWPCN_10  177 TRUE       -7.337  TRUE      -9   FALSE       10 FALSE TRUE
## NSWPCN_13  247 TRUE       -7.532 FALSE     -19    TRUE       20 FALSE TRUE
## NSWPCN_17  316 TRUE       -7.417 FALSE     -23   FALSE       -5 FALSE TRUE
## NSWPCN_20  256 TRUE       -3.392 FALSE      -8   FALSE        0 FALSE TRUE
##            SizePlus
## NSWPCN_4          0
## NSWPCN_9         10
## NSWPCN_10        10
## NSWPCN_13        20
## NSWPCN_17         0
## NSWPCN_20         0
```

```
#fit.final.gg = flexsurvreg(Surv(Time, DSD) ~ SexM + AgeCent + LocBody + SizeCent + A2 + A4 + SizeCent:
fit.final.gg = flexsurvreg(Surv(Time, DSD) ~ SexM + LocBody + SizeCent + A2 + A4,
        anc = list(
                sigma = ~ SexM,
                Q = ~ SexM),
        data = data.all, dist = "gengamma")
# fit.final.cph = coxph(Surv(Time, DSD) ~ strata(SexM)+AgeCent+LocBody+SizeCent+A2+A4+SizeCent:AgeCent+
fit.final.cph = coxph(Surv(Time, DSD) ~ strata(SexM) + LocBody + SizeCent + A2 + A4, data = data.all, x
set.seed(20150208)
fit.final.rsf = rfsrc(Surv(Time, DSD) ~ SexM + AgeCent + LocBody + SizeCent + A2 + A4, data = data.all,
fit.final.km0 = survfit(Surv(Time, DSD) ~ 1, data.all)
saveRDS(list(gg = fit.final.gg, km0 = fit.final.km0, cph = fit.final.cph, rsf = fit.final.rsf, data.trai
```

```
save.image("05_train_NSWPCN_2.rda")
```

7 Session information

```
sessionInfo()
```

```
## R version 3.1.1 (2014-07-10)
## Platform: x86_64-unknown-linux-gnu (64-bit)
##
## locale:
##  [1] LC_CTYPE=en_AU.UTF-8          LC_NUMERIC=C
##  [3] LC_TIME=en_AU.UTF-8           LC_COLLATE=en_AU.UTF-8
##  [5] LC_MONETARY=en_AU.UTF-8       LC_MESSAGES=en_AU.UTF-8
##  [7] LC_PAPER=en_AU.UTF-8          LC_NAME=en_AU.UTF-8
##  [9] LC_ADDRESS=en_AU.UTF-8        LC_TELEPHONE=en_AU.UTF-8
## [11] LC_MEASUREMENT=en_AU.UTF-8    LC_IDENTIFICATION=en_AU.UTF-8
##
## attached base packages:
## [1] parallel  methods   splines   stats     graphics  grDevices utils
## [8] datasets  base
##
## other attached packages:
##  [1] RColorBrewer_1.0-5    timeROC_0.2          timereg_1.8.6
##  [4] mvtnorm_1.0-1         pec_2.4.4            boot_1.3-13
```

```
##  [7] MASS_7.3-35          ggplot2_1.0.0         plyr_1.8.1
## [10] reshape2_1.4          randomForestSRC_1.5.5 flexsurv_0.5
## [13] glmulti_1.0.7         rJava_0.9-6           survival_2.37-7
## [16] tikzDevice_0.8.1      knitr_1.8
##
## loaded via a namespace (and not attached):
##  [1] codetools_0.2-9  colorspace_1.2-4 deSolve_1.11     digest_0.6.4
##  [5] evaluate_0.5.5   filehash_2.2-2   foreach_1.4.2    formatR_1.0
##  [9] grid_3.1.1       gtable_0.1.2     highr_0.4        iterators_1.0.7
## [13] labeling_0.3     lava_1.3         muhaz_1.2.6      munsell_0.4.2
## [17] prodlim_1.5.1    proto_0.3-10     Rcpp_0.11.3      scales_0.2.4
## [21] stringr_0.6.2    tools_3.1.1
```