Exercises Topic 5 and 6

Lifetime Data Analysis - Autumn 2021

Alexander and Ulrik

28 desember, 2021

Exercise 1

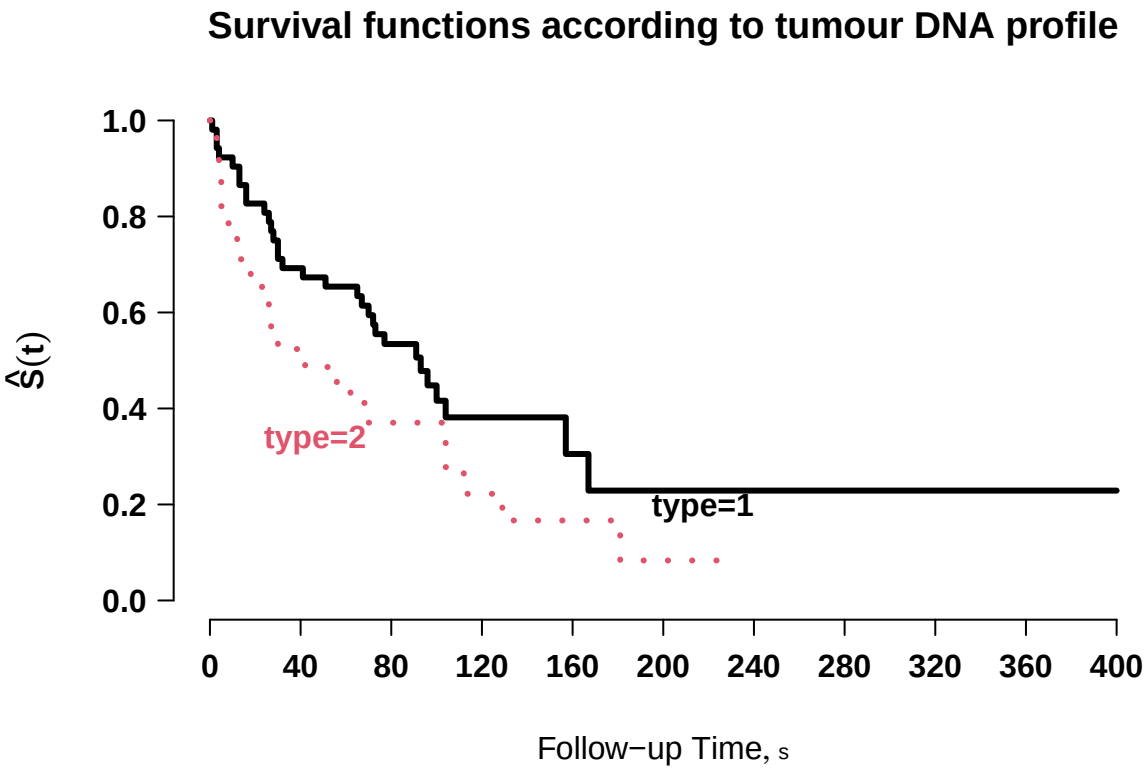
The data frame **tongue** of the R package **KMsurv** contains the survival times (in weeks) of 80 patients with oral cancer. The objective of this exercise is to study the possible relation of this cancer with the tumour DNA profile, which is either aneuploid (type 1) or diploid (type 2) .

```
data(tongue)
tongue$type <- factor(tongue$type)
```

1.a)

The survival curves are plotted below. Observe that the survival of the aneuploid tumour is greater than the survival of the diploid tumour. As the survival curves do not go to zero there is right censoring in both groups.

```
s <- with(tongue, Surv(time, delta))
t <- npsurv(s ~ type, tongue)

par(font = 2, font.axis = 2, font.lab = 4, las = 1, mar = c(5, 5, 4, 2))
survplot(t, ylab = expression(bold(hat(S)(t))), col = 1:2, lwd = 3, conf = "none")
title("Survival functions according to tumour DNA profile")
```



1.b)

The logrank test is used to test the hypothesis that survival is not related to tumour type. The hypothesis that is tested can be formulated as

$$H_0 : S_1(t) = S_2(t) \text{ vs. } H_1 : S_1(t) \neq S_2(t),$$

where $S_1(t)$ and $S_2(t)$ refer to the survival curves of tumour type 1 and 2 respectively.

The result from the logrank test is a Z-value of 1.7 and a p-value of 0.0949, which means that we would conclude not to reject H_0 when choosing a significance level of (for example) 0.05 for the p-value. This means that, according to the logrank test, there is not enough evidence to conclude that the tumour DNA profile is related to the survival of the cancer patients, i.e. one does not conclude that one of the tumour profiles leads to more severe cancer.

```
s2 <- with(tongue, Surv(time, delta) ~ type)
FHtestrcc(s2)
```

```
##
##  Two-sample test for right-censored data
##
## Parameters: rho=0, lambda=0
## Distribution: counting process approach
##
## Data: Surv(time, delta) by type
##
##         N Observed Expected   O-E (O-E)^2/E (O-E)^2/V
## type=1 52       31     36.6 -5.55     0.843      2.79
## type=2 28       22     16.4  5.55     1.873      2.79
##
```

```
## Statistic Z= 1.7, p-value= 0.0949
## Alternative hypothesis: survival functions not equal
```

1.c)

The log-logistic regression model is fitted with the single covariate 'Tumour type'. This model is used to test the same hypothesis as in **b)**. In this case, the null and alternative hypothesis can be formulated as

$$H_0 : \gamma_1 = 0 \text{ vs. } H_1 : \gamma_1 \neq 0,$$

where γ_1 is the covariate coefficent for tumour DNA profile in the model. The regression gives $\hat{\gamma}_1 = -0.79$ with p-value 0.051, which gives significant evidence to reject the null hypothesis with a level of $\alpha = 0.05$. Thus, the test suggests that DNA profile has an impact on the survival time of cancer patients. The negative value of the parameter estimate suggests that the diploid tumour profile is non-protective, i.e. that it lowers the survival of a person. This will be more closely examined in the next part of the problem.

```
loglogistic <- survreg(s ~ type, data = tongue, dist = "loglogistic")
summary(loglogistic)
```

```
##
## Call:
## survreg(formula = s ~ type, data = tongue, dist = "loglogistic")
##               Value Std. Error     z      p
## (Intercept)  4.4695     0.2424 18.44 <2e-16
## type2       -0.7906     0.4042 -1.96   0.05
## Log(scale)  -0.0413     0.1161 -0.36   0.72
##
## Scale= 0.96
##
## Log logistic distribution
## Loglik(model)= -298.7   Loglik(intercept only)= -300.6
##  Chisq= 3.79 on 1 degrees of freedom, p= 0.051
## Number of Newton-Raphson Iterations: 4
## n= 80
```

1.d)

Since the log-logistic model can be viewed as a proportional odds model, the odds ratio $\exp\{-\boldsymbol{\beta}'\boldsymbol{Z}\} = \exp\{\boldsymbol{\gamma}'\boldsymbol{Z}/\sigma\}$ indicates how the odds to survive changes with respect to covariates $\mathbf{Z}$ compared to $\mathbf{Z} = \mathbf{0}$. For this model a patient with DNA profile diploid has a survival odds 0.44 times the survival of a patient with DNA profile aneuploid. This means that the odds of surviving with the diploid profile is 44% of the odds of surviving with the aneuploid profile.

Similarly, the acceleration factor $\exp\{-\boldsymbol{\gamma}'\boldsymbol{Z}\}$ describes the change in time scale compared to the baseline $\mathbf{Z} = \mathbf{0}$. The value of the acceleration factor in this case is 2.205. This means that the death of a person with a diploid profile is accelerated by 2.205 compared to the death of a person with a aneuploid profile. For example, considering medians, this means that the median of a person with an aneuploid profile is estimated to being 2.205 larger than the median of a person with a diploid profile, which fits nicely with the survival curves we have seen earlier.

```
with(loglogistic, exp(coefficients[2] / scale))  # Odds ratio.
```

```
##     type2
## 0.4387053
```

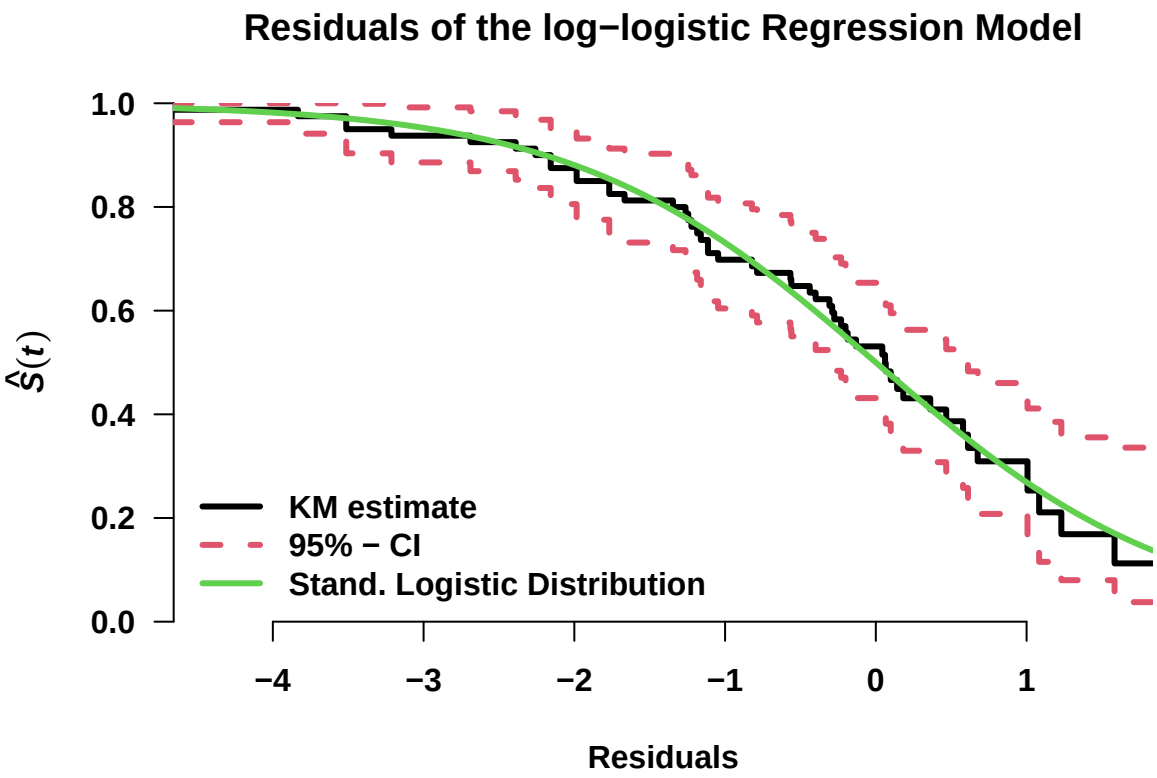
```
with(loglogistic, exp(-coefficients[2]))  # Acceleration factor.
```

```
##    type2
## 2.204704
```

1.e)

The residuals of the log-logistic regression model are plotted below. As one can see, the standard logistic distribution seems to fit the residual KM-estimate well, which can be used as an indication that the log-logistic model is appropriate for this data. This is because the standard logistic distribution seems a reasonable choice for the error W.

```
loglo.pred <- predict(loglogistic, type = "linear")
resids.loglo <- (log(tongue$time) - loglo.pred) / loglogistic$scale

par(font = 2, font.axis = 2, font.lab = 2, las = 1, mar = c(5, 5, 4, 2))
plot(survfit(Surv(resids.loglo, tongue$delta) ~ 1), col = c(1,2,2), xlab = "Residuals",
     ylab = expression(bolditalic(hat(S)(t))),
     lty = 1, lwd = 3, yaxs = "i", xaxs = "i", bty = "n")
title("Residuals of the log-logistic Regression Model")
curve(plogis(x, lower.tail = F), from = min(resids.loglo), to = max(resids.loglo), col = 3, lwd = 3,
      add = TRUE)
legend("bottomleft", c("KM estimate", "95% - CI", "Stand. Logistic Distribution"),
       col = c(1, 2, 3), lty = c(1, 2, 1), lwd = 3, bty = "n")
```



Exercise 2

Following, we will generate survival times from a log-normal distribution and check whether the Weibull or the log-logistic distribution fit better to the data.

2.a)

300 survival times from a log-normal distribution with parameters $\mu = 2$ and $\sigma = 1$ are generated below.

```
set.seed(1)
mu <- 2
sigma <- 1
location <- log(mu^2 / sqrt(sigma^2 + mu^2))
shape <- sqrt(log(1 + (sigma^2 / mu^2)))
RVT <- rlnorm(300, meanlog = location, sdlog = shape)
# https://en.wikipedia.org/wiki/Log-normal_distribution#Arithmetic_moments
# https://msalganik.wordpress.com/2017/01/21/making-sense-of-the-rlnorm-function-in-r/
```

2.b)

300 censoring times from an exponential distribution with mean 20 are generated below.

```
RVC <- rexp(300, rate = 1/20)
```

2.c)

The variables $Y = \min(T, C)$ and $\delta = \mathbf{1}_{\{T \leq C\}}$ are created below. The table below shows the counts of exact failure times and censoring times.

```
obs <- pmin(RVT, RVC)
cens <- as.numeric(RVT <= RVC)
(tab <- table(cens))
```

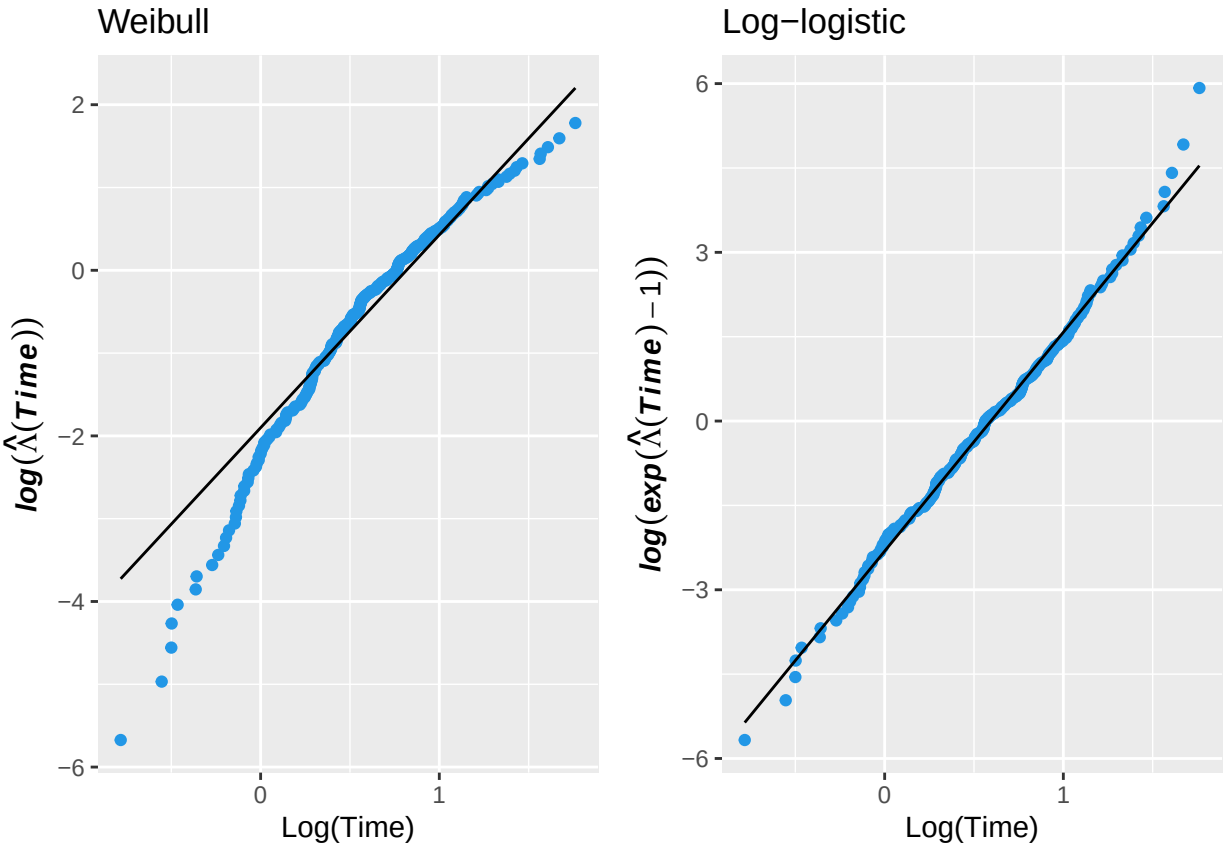
```
## cens
##   0   1
##  37 263
```

The proportion of right-censored survival times is thus 0.123.

2.d)

The cumulative hazard plots for the Weibull and the log-logistic distributions are plotted below. It looks like the log-logistic distribution fits the data better than the Weibull model, since the latter is clearly a bad fit.

```
cumhazPlot(obs, cens, col = 4, distr = c("wei", "loglo"), ggplo = T)
```



Exercise 3

Assume that the model assumptions for the Cox proportional hazards model holds for a continuous survival time T with covariates $\mathbf{Z} = (Z_1, \ldots, Z_p)'$, i.e., that,

$$\lambda(t; \mathbf{z}) = \lambda_0(t) \exp(\boldsymbol{\beta}' \boldsymbol{z}),$$

but that the survival times are grouped in the intervals $[0 = a_0, a_1), \ldots, [a_{g-1}, a_g)$. The corresponding hazards are defined as

$$\lambda_j(\mathbf{z}) = P(T < a_j | T \geq a_{j-1}; \mathbf{z}), \ j = 1, \ldots, g.$$

Then we can write

$$\begin{aligned} 1 - \lambda_j(\mathbf{z}) &= P(T \geq a_j | T \geq a_{j-1}; \mathbf{z}) \\ &= \frac{P(T \geq a_j, T \geq a_{j-1}; \mathbf{z})}{P(T \geq a_{j-1}; \mathbf{z})} \\ &= \frac{P(T \geq a_j; \mathbf{z})}{P(T \geq a_{j-1}; \mathbf{z})}. \end{aligned}$$

First we observe that

$$
\begin{aligned}
P(T \geq a_j; \mathbf{z}) &= S_{\mathbf{z}}(a_j) \\
&= \exp(-\Lambda_{\mathbf{z}}(a_j)) \\
&= \exp(-(\int_{a_0}^{a_1} \lambda_0(t)\exp(\boldsymbol{\beta'z})\mathrm{d}t + \ldots + \int_{a_{j-1}}^{a_j} \lambda_0(t)\exp(\boldsymbol{\beta'z})\mathrm{d}t)) \\
&= \exp(-(\int_{a_0}^{a_1} \lambda_0(t)\mathrm{d}t + \ldots + \int_{a_{j-1}}^{a_j} \lambda_0(t)\mathrm{d}t)\exp(\boldsymbol{\beta'z}) \\
&= (\exp(-\Lambda_{\mathbf{0}}(a_j)))^{\exp(\boldsymbol{\beta'z})} \\
&= (P(T \geq a_j; \mathbf{0}))^{\exp(\boldsymbol{\beta'z})}.
\end{aligned}
$$

The exact same argument for $P(T \geq a_{j-1}; \mathbf{z})$ gives us

$$
\begin{aligned}
1 - \lambda_j(\mathbf{z}) &= \left(\frac{P(T \geq a_j; \mathbf{0})}{P(T \geq a_{j-1}; \mathbf{0})}\right)^{\exp(\boldsymbol{\beta'z})} \\
&= (1 - \lambda_j(\mathbf{0}))^{\exp(\boldsymbol{\beta'z})}
\end{aligned}
$$

or equivalently

$$
\log(1 - \lambda_j(\mathbf{z})) = \log(1 - \lambda_j(\mathbf{0}))\exp(\boldsymbol{\beta'z}).
$$

Exercise 4

The data frame **hodg** of the **KMsurv** package contains the times until relapse or death of 43 lymphoma patients that underwent a bone marrow transplant.

4.a)

The variables **gtype** and **dtype** are converted into factors.

```
data(hodg)
str(hodg)
```

```
## 'data.frame':    43 obs. of  6 variables:
##  $ gtype: int  1 1 1 1 1 1 1 1 1 1 ...
##  $ dtype: int  1 1 1 1 1 1 1 1 1 1 ...
##  $ time : int  28 32 49 84 357 933 1078 1183 1560 2114 ...
##  $ delta: int  1 1 1 1 1 0 0 0 0 0 ...
##  $ score: int  90 30 40 60 70 90 100 90 80 80 ...
##  $ wtime: int  24 7 8 10 42 9 16 16 20 27 ...
```

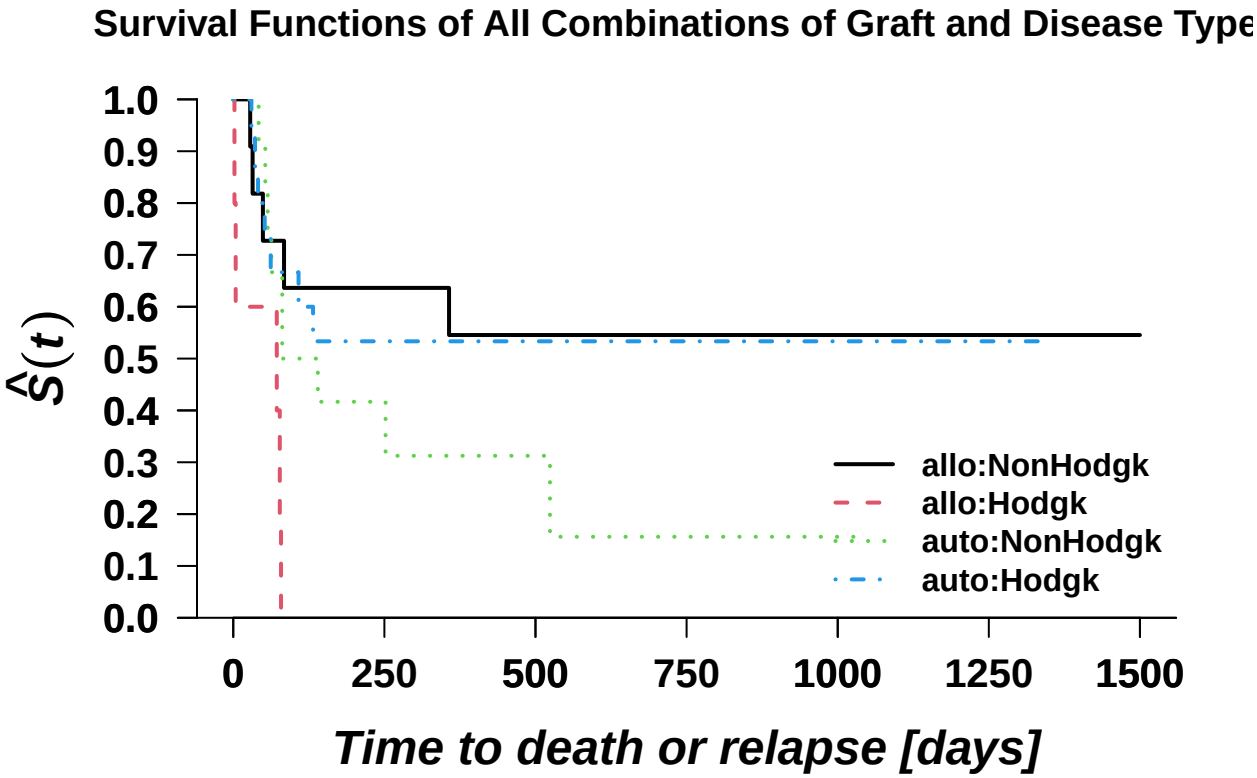
```
hodg$gtype <- factor(hodg$gtype, labels = c("allo", "auto"))
hodg$dtype <- factor(hodg$dtype, labels = c("NonHodgk", "Hodgk"))
str(hodg)
```

```
## 'data.frame':    43 obs. of  6 variables:
##  $ gtype: Factor w/ 2 levels "allo","auto": 1 1 1 1 1 1 1 1 1 1 ...
##  $ dtype: Factor w/ 2 levels "NonHodgk","Hodgk": 1 1 1 1 1 1 1 1 1 1 ...
##  $ time : int  28 32 49 84 357 933 1078 1183 1560 2114 ...
##  $ delta: int  1 1 1 1 1 0 0 0 0 0 ...
##  $ score: int  90 30 40 60 70 90 100 90 80 80 ...
##  $ wtime: int  24 7 8 10 42 9 16 16 20 27 ...
```

4.b

Survival curves corresponding to the four combinations of graft and disease types are plotted below.

```
srem <- with(hodg, Surv(time, delta))
svf <- survfit(srem ~ gtype + dtype, data = hodg)

par(las = 1, font = 2, font.axis = 2, font.lab = 4, xaxs = "i", yaxs = "i",
    mar = c(5, 5, 4, 2), bty = "l", cex.lab = 1.5, cex.axis = 1.25)
plot(svf, conf.int = F, lwd = 2, col = 1:4, lty = 1:4, xlab = "Time to death or relapse [days]",
     ylab = expression(bolditalic(hat(S)(t))), ylim = c(0,1), xlim = c(0, 1500))
axis(1, at = seq(0, 2000, 250))
axis(2, at = seq(0, 1, 0.1))
title("Survival Functions of All Combinations of Graft and Disease Types")
legend("bottomright", legend = levels(hodg$gtype:hodg$dtype),
       col = 1:4, lty = 1:4, lwd = 2, bty = "n")
```



The leftmost number in the legend is **gtype** and the rightmost number is **dtype**. The x-axis is limited to 1500 in order to see more clearly what is happening in the beginning.

It looks like the different combinations of graft and disease types yield significantly different survival times. Considering longevity, the combination of allogenic graft type and Non Hodgkin lymphoma yields the largest survival, closely followed by autologous graft type and Hodgkins disease. The combination of allogenic graft type and Hodgkins disease seams to yield very low survival. Note that all combinations have a right-censored last time in the data set, except the combination of allogenic graft type and Hodgkins disease, which has a survival that ends in zero, because the last time is a failure time.

4.c)

Fit the proportional hazards model that includes graft type, disease type, the interaction of both and the Karnofsky index.

```r
proph <- coxph(srem ~ gtype + dtype + gtype:dtype + score, data = hodg)
summary(proph)
```

```
## Call:
## coxph(formula = srem ~ gtype + dtype + gtype:dtype + score, data = hodg)
##
##   n= 43, number of events= 26
##
##                           coef exp(coef) se(coef)      z Pr(>|z|)
## gtypeauto              0.53270   1.70353  0.58231  0.915   0.3603
## dtypeHodgk             1.68314   5.38244  0.69463  2.423   0.0154 *
## score                 -0.05471   0.94676  0.01226 -4.463  8.1e-06 ***
## gtypeauto:dtypeHodgk -1.65262   0.19155  0.91614 -1.804   0.0712 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##                      exp(coef) exp(-coef) lower .95 upper .95
## gtypeauto               1.7035     0.5870    0.5441    5.3335
## dtypeHodgk              5.3824     0.1858    1.3795   21.0015
## score                   0.9468     1.0562    0.9243    0.9698
## gtypeauto:dtypeHodgk    0.1915     5.2207    0.0318    1.1537
##
## Concordance= 0.753  (se = 0.053 )
## Likelihood ratio test= 28.87  on 4 df,   p=8e-06
## Wald test            = 25.82  on 4 df,   p=3e-05
## Score (logrank) test = 34.66  on 4 df,   p=5e-07
```

As is seen from the output above, all three tests shows that the model is significantly better than the null model The Karnofsky index (`score`) is significant. Moreover, when choosing a significance level of $\alpha = 0.05$, the parameter estimate comparing Hodgkins disease to non Hodgkins disease is significant.

Since the parameter estimation for the Karnofsky index is negative, the model is predicting that an increase in the Karnofsky score, while all other covariates are kept static, decreases the instantaneous risk of death or relapse. More precisely, for a unitary increase in the Karnofsky index, an individual with the same profile (except the change in the score) will have an instantaneous risk that is $\approx e^{-0.05441} \approx 0.95$ times the individual before the increase. On the contrary, having Hodgkins disease is predicted as being non-protective compared to not having it, when considering the other covariates as static. More precisely, having the disease will yield an instantaneous risk of $\approx e^{1.68314} \approx 5.38$ times the instantaneous risk when not having the disease, for an individual with the same profile (the remaining covariates). Similar interpretations can be done with the two other covariates, but this is excluded since they are not statistically significant.

4.d)

The hazard ratio measures the comparative instantaneous risk of death. Since there exists interaction between the graft type and the disease type, two hazard ratios are computed, one for patients with non Hodgkin lymphoma, and one for patients with Hodgkins disease. The hazard ratios are 1.704 and 0.326 respectively. This means that patients receiving an autologous graft have 1.704 times higher instantaneous risk of dying compared to patients with allogenic graft type, given that the disease type is non Hodgkin lymphoma and the Karnofsky index is the same. Similarly, patients receiving an autologous graft have 0.326 times lower instantaneous risk of dying compared to patients with allogenic graft type, given that the disease type is Hodgkins disease and the Karnofsky index is the same.

```r
library(Epi)
ctmat <- matrix(c(1,0,0,0,1,0,0,1), byrow = TRUE, nr = 2)
HRmat <- round(ci.lin(proph, ctr.mat = ctmat, Exp = TRUE), 3)[, c(1, 5:7)]
```

```
rownames(HRmat) <- c("HR| Non Hodgkin lymphoma", "HR| Hodgkins disease")
colnames(HRmat) <- c("logHR", "HR", "Lower 95%", "Upper 95%")
HRmat
```

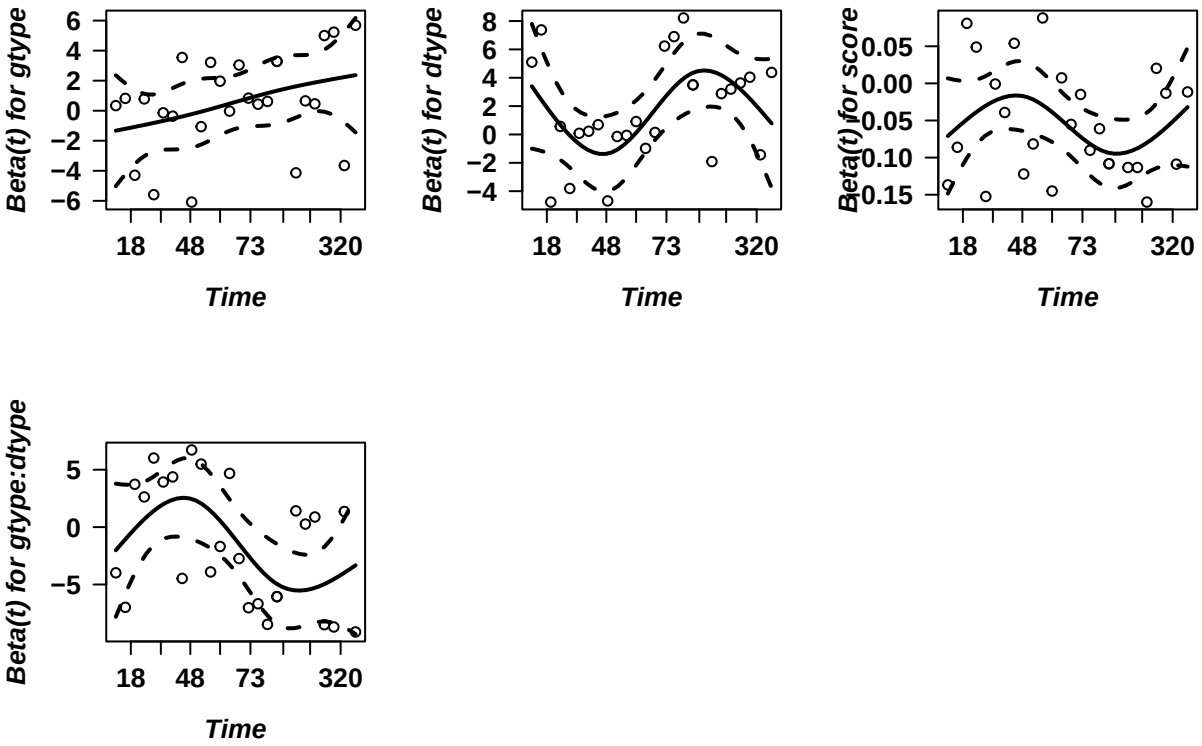
```
##                            logHR    HR Lower 95% Upper 95%
## HR| Non Hodgkin lymphoma  0.533 1.704     0.544     5.334
## HR| Hodgkins disease     -1.120 0.326     0.090     1.179
```

4.e)

The proportional hazards assumption is checked, using the Schoenfeld residuals.

```
schres <- residuals(proph, "schoenfeld")

prop.haz.test <- cox.zph(proph)
```

```
par(mfrow = c(2, 3), font = 2, font.lab = 4, font.axis = 2, las = 1,
    cex.lab = 1.3, cex.axis = 1.2)
plot(prop.haz.test, lwd = 2)
```



The Schoenfeld residuals are plotted above. None of the covariates look like they have a line with slope zero, i.e. it looks like they all exhibit a systematic pattern. This means that, based on the plot, the proportional hazards assumption does not hold for any of the covariates.

```
prop.haz.test
```

```
##                chisq df     p
## gtype          0.358  1 0.550
## dtype          2.428  1 0.119
## score          3.691  1 0.055
## gtype:dtype    4.630  1 0.031
## GLOBAL        12.623  4 0.013
```

The p-values in the table above are p-values from a two-sided test of slope $= 0$ in the Schoenfeld residuals plotted earlier. To a significance level of $\alpha = 0.05$, the only null hypothesis to be rejected is the one concerning the interaction term.

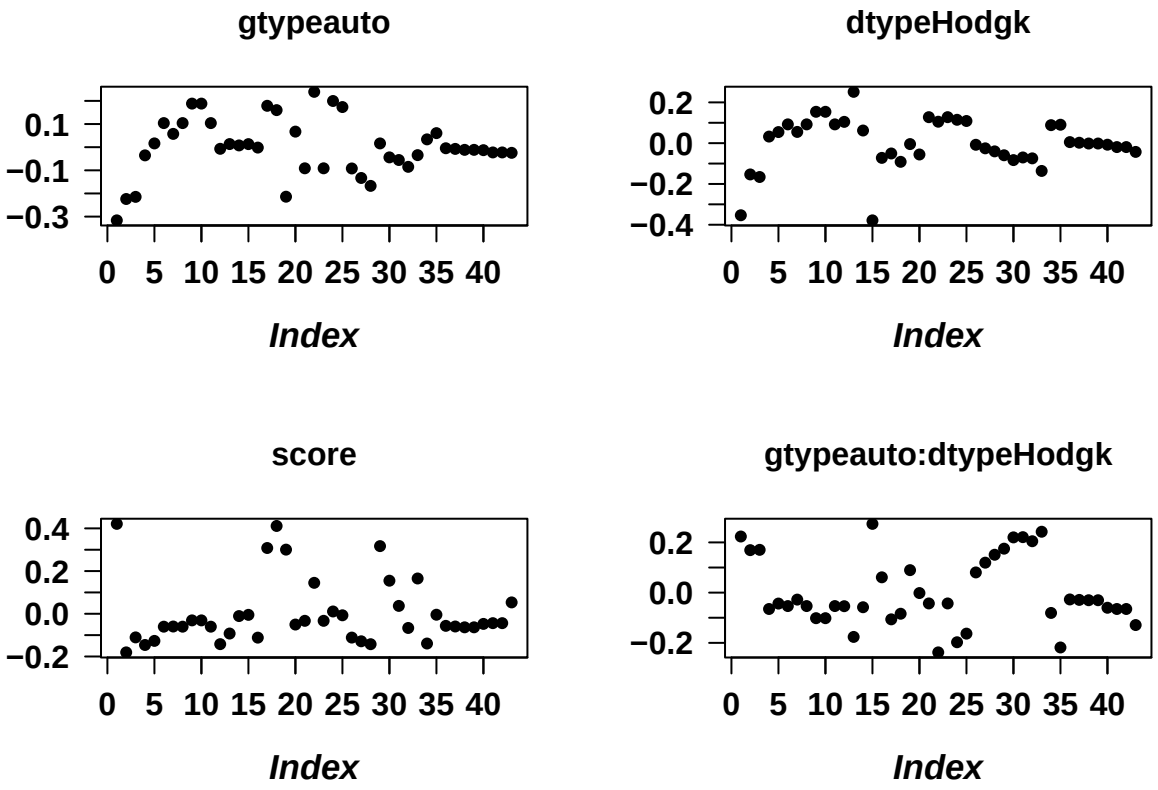
Despite the results from the tests above, we would still conclude that the proportional hazards assumption does not hold for any of the covariates, which means that the model is not suitable in this case.

4.f)

Concerning the estimation of the four model parameters, are there any influential observations?

A transformation of the score residuals for each of the four coefficients is plotted below. More precisely, each residual that is plotted is the approximate change in the coefficient vector if the observation in question is dropped, scaled by the standard error of the coefficients.

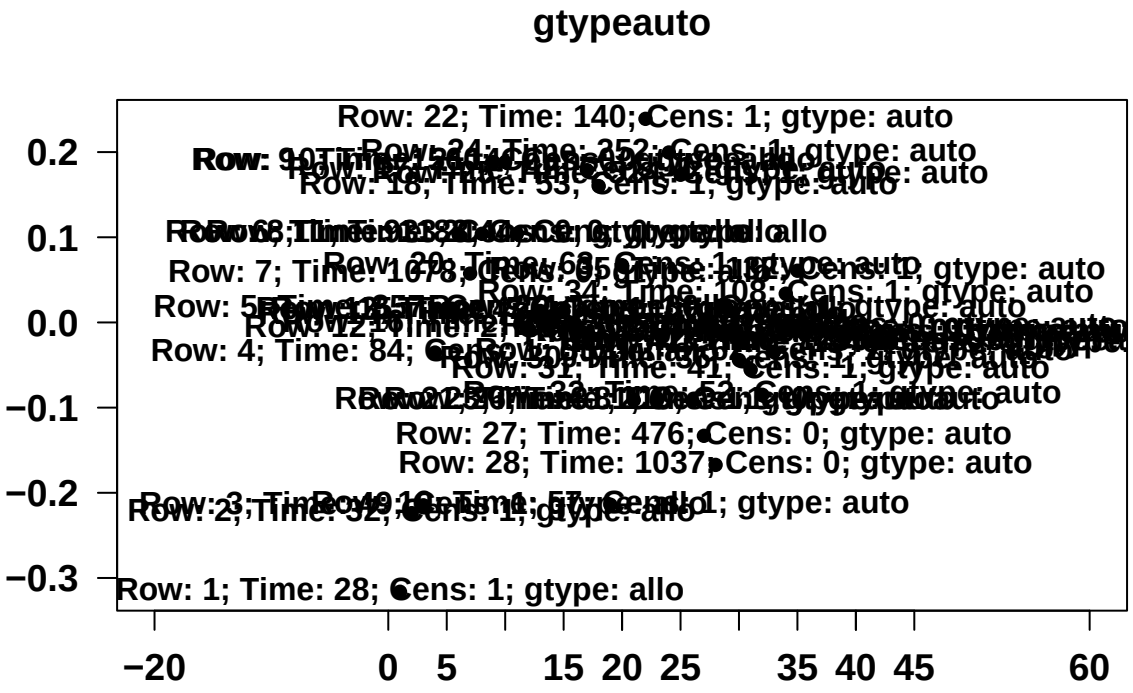
```
dfbet <- residuals(proph, type = "dfbetas")

par(mfrow = c(2, 2), font = 2, font.lab = 4, font.axis = 2, las = 1,
    cex.lab = 1.3, cex.axis = 1.2)
for (i in 1:4) {
  plot(dfbet[, i], pch = 16, ylab = "")
  title(names(coef(proph))[i])
  axis(1, at = seq(5, 45, 5))
}
```



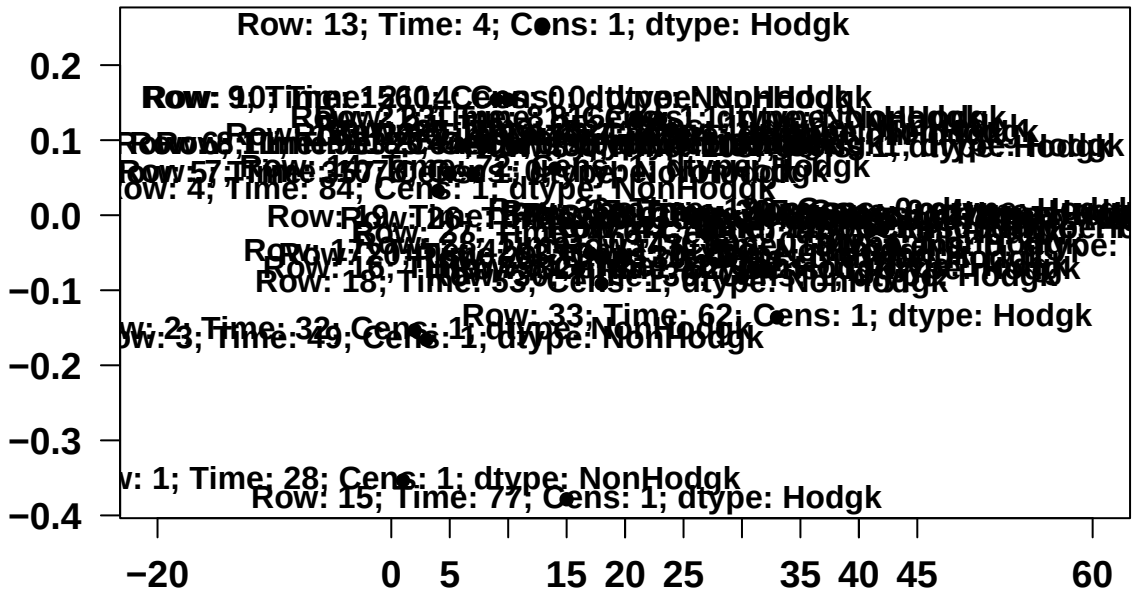
In order to identify which observations are influential, text describing each residual is plotted alongside the points. They can also be identified manually by clicking on the plots using the `identify` function in R.

```
par(mfrow = c(1,1), font = 2, font.lab = 4, font.axis = 2, las = 1, cex.lab = 1.3,
    cex.axis = 1.2)
plot(dfbet[, 1], pch = 16, ylab = "",xlim = c(-20, 60))
title(names(coef(proph))[1])
```

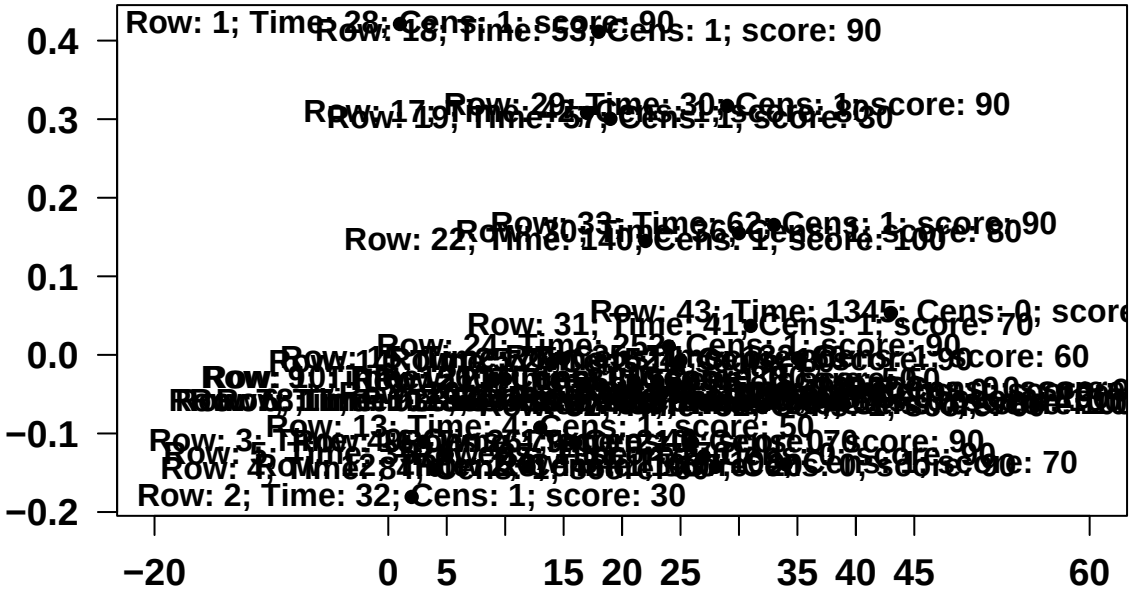
```
axis(1, at = seq(5, 45, 5))
text(dfbet[, 1],
         labels = paste0("Row: ", rownames(hodg), "; Time: ", hodg$time,
                         "; Cens: ", hodg$delta, "; gtype: ", hodg$gtype))
```



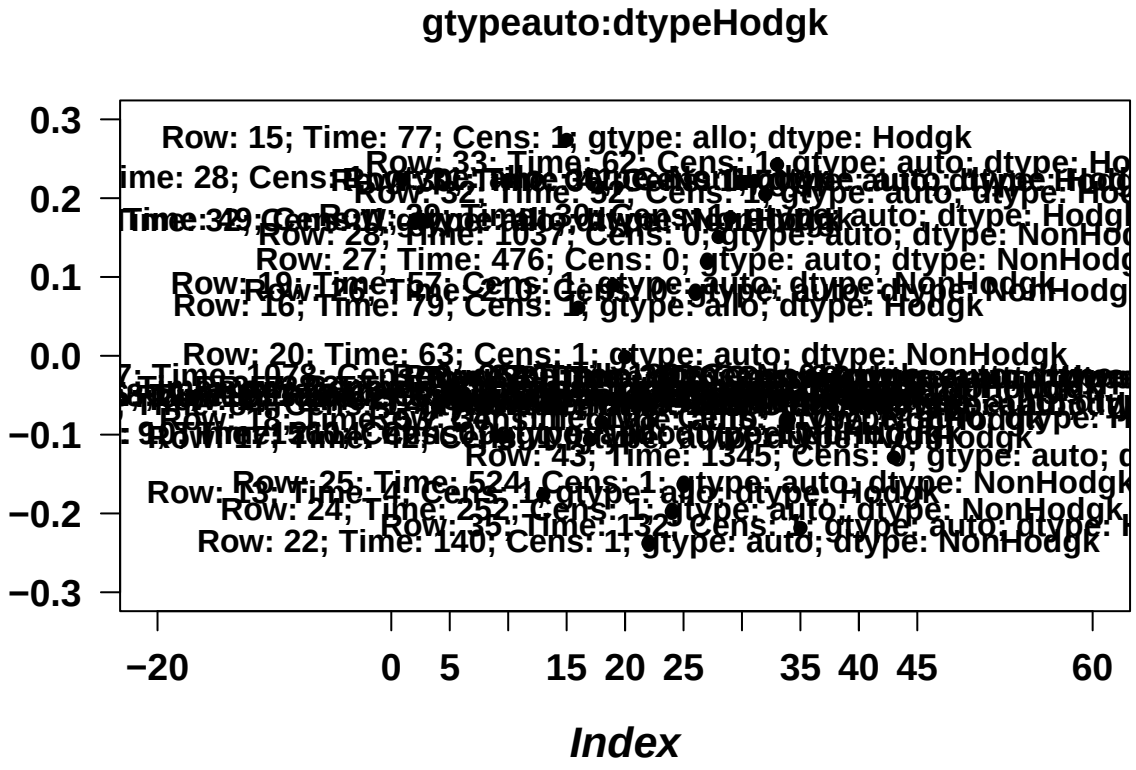

```
par(mfrow = c(1,1), font = 2, font.lab = 4, font.axis = 2, las = 1, cex.lab = 1.3,
    cex.axis = 1.2)
plot(dfbet[, 2], pch = 16, ylab = "", xlim = c(-20, 60))
title(names(coef(proph))[2])
axis(1, at = seq(5, 45, 5))
text(dfbet[, 2],
         labels = paste0("Row: ", rownames(hodg), "; Time: ", hodg$time,
                         "; Cens: ", hodg$delta, "; dtype: ", hodg$dtype))
```




```
par(mfrow = c(1,1), font = 2, font.lab = 4, font.axis = 2, las = 1, cex.lab = 1.3,
    cex.axis = 1.2)
plot(dfbet[, 3], pch = 16, ylab = "", xlim = c(-20, 60))
title(names(coef(proph))[3])
axis(1, at = seq(5, 45, 5))
text(dfbet[, 3],
         labels = paste0("Row: ", rownames(hodg), "; Time: ", hodg$time,
                         "; Cens: ", hodg$delta, "; score: ", hodg$score))
```




```
par(mfrow = c(1,1), font = 2, font.lab = 4, font.axis = 2, las = 1, cex.lab = 1.3,
    cex.axis = 1.2)
plot(dfbet[, 4], pch = 16, ylab = "", xlim = c(-20, 60), ylim = c(-0.3, 0.3))
title(names(coef(proph))[4])
axis(1, at = seq(5, 45, 5))
text(dfbet[, 4],
         labels = paste0("Row: ", rownames(hodg), "; Time: ", hodg$time,
                         "; Cens: ", hodg$delta, "; gtype: ",
                         hodg$gtype, "; dtype: ", hodg$dtype))
```



There are some influential observations. They can be found for large absolute values that deviate from zero in the plots above. Examples like row 1 and row 15 seem to be influential for several of the covariates.

For example, row 1

```
hodg[1,]
```

```
##   gtype    dtype time delta score wtime
## 1  allo NonHodgk   28     1    90    24
```

has a large score and looks to be influential for both `gtypescore` and `dtypeHodgk`. Estimating the Cox model without the first row gives

```
summary(coxph(Surv(time, delta) ~ gtype + dtype + gtype:dtype + score, data = hodg[-1,]))
```

```
## Call:
## coxph(formula = Surv(time, delta) ~ gtype + dtype + gtype:dtype +
##     score, data = hodg[-1, ])
##
##   n= 42, number of events= 25
##
##                           coef exp(coef) se(coef)      z Pr(>|z|)
## gtypeauto              0.73956   2.09502  0.63065  1.173  0.24092
## dtypeHodgk             1.95950   7.09578  0.74342  2.636  0.00839 **
## score                 -0.06029   0.94150  0.01295 -4.656 3.22e-06 ***
## gtypeauto:dtypeHodgk -1.88821   0.15134  0.96640 -1.954  0.05072 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##                      exp(coef) exp(-coef) lower .95 upper .95
## gtypeauto               2.0950     0.4773   0.60867    7.2110
## dtypeHodgk              7.0958     0.1409   1.65272   30.4650
```

```
## score                   0.9415     1.0621   0.91791    0.9657
## gtypeauto:dtypeHodgk    0.1513     6.6075   0.02277    1.0059
##
## Concordance= 0.801  (se = 0.042 )
## Likelihood ratio test= 32.83  on 4 df,   p=1e-06
## Wald test            = 27.98  on 4 df,   p=1e-05
## Score (logrank) test = 39.26  on 4 df,   p=6e-08
```

where we see that the hazard ratios for the two mentioned covariates change quite a bit.