

Opgavebesvarelse vedr. overlevelsesanalyse

Opgaven går ud på at vurdere effekten af azathioprine på overlevelsen hos 216 patienter med primær biliær cirrhose (PBC), ref. Christensen et al. (1985).

Data stammer fra en 'clinical trial', hvor de 216 patienter er randomiseret til behandling med azathioprine eller placebo, hvorefter de følges over tid indtil død eller censurering.

Filen `pbc.txt` indeholder følgende variable:

- **dead**: Indikator for død (0: Censurering, 1: død)
- **months**: Antal måneder fra randomisering til udgang (død eller censurering)
- **azathiop**: Behandling (0: Placebo, 1: Azathioprine)
- **bilirubin**: Serum bilirubin måling ved baseline

Data indlæses direkte fra hjemmesiden og vi tager et hurtigt kig på de første seks linier:

```
wr <- read.table( "http://publicifsv.sund.ku.dk/~lts/basal/data/pbc.txt", header=T)
head(wr)
```

	dead	months	azathiop	bilirubin
1	1	0.16	1	294
2	0	0.26	1	48
3	1	0.30	1	159
4	1	0.30	0	61
5	1	0.62	0	423
6	1	1.25	1	322

Vi kan vælge at lave azathiop til en faktor med labels `ja` for de som har fået azathiop, `nej` for de som har fået placebo:

```
wr$azathiop <- factor( wr$azathiop, labels = c('nej','ja'))
```

Vi vil benytte fire pakker som installeres med:

og derefter skal indholdet af disse pakker gøres tilgængeligt med

```
library('survival')
library('prodlim')
library('timereg')
library('car')
```

Loading required package: carData

Nogle af de funktioner vi benytter nedenfor er beskrevet i denne tutorial om overlevelsesanalyse i R.

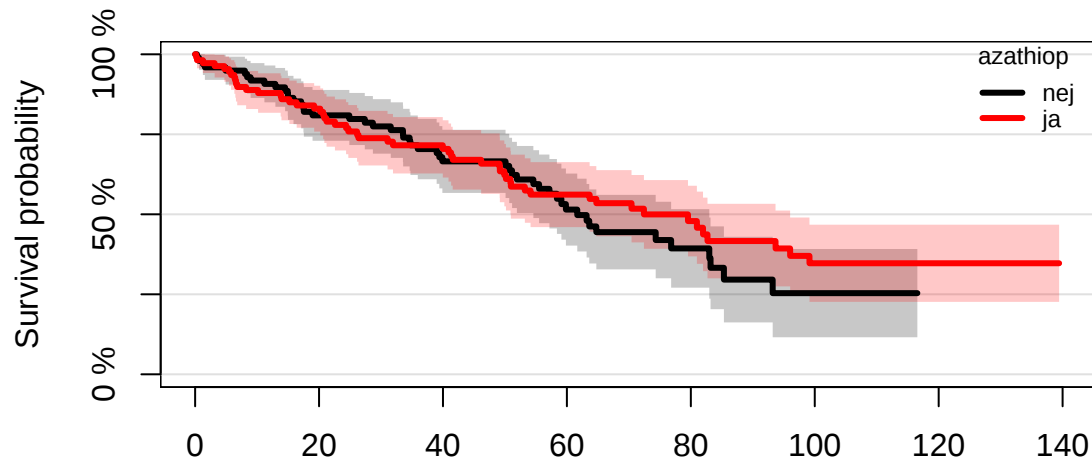
1

Da der er tale om et randomiseret studie, vurderes behandlingseffekten først uden inddragelse af diverse kovariater.

a) Effekten af azathioprine ud fra Kaplan-Meier kurver

Kaplan-Meier kurver er nonparametriske estimater for overlevelseskurverne (nonparametrisk = ingen antagelser om fordelingen af dødstidspunkterne). Et plot med skravering af konfidensintervallerne kan man få ved brug af funktionen `prodlim()` fra `prodlim`-pakken:

```
km1 <- prodlim( Hist(months,dead) ~ azathiop, data=wr )
plot(km1, legend.cex=.75, legend.inset=.01)
```



	Time										
azathiop											
nej:	98	83	68	49	39	21	11	5	2	0	0
ja:	109	89	69	59	44	32	19	16	11	6	1

De er her tegnet med punktvisse konfidensgrænser, fordi det oftest er nødvendigt at have konfidensgrænserne med for at kunne vurdere, om der er klar forskel på grupperne...

Vi ser, at der er adskillige overkrydsninger mellem de to overlevelseskurver og ingen iøjnefaldende forskelle undtagen, at de måske divergerer lidt mod slut.

Tag eventuelt et kig på den tutorial jeg linker til ovenfor for at se, hvordan man kan tilpasse plottet yderligere med farver, titel, legend mm.

b) Et passende test

Vi skal se på tid til død, under hensyntagen til, at der forekommer censureringer ($dead=0$), og vi skal sammenligne to grupper. Det relevante test er et *log-Rank test*. Vi finder nedenstående, der ikke viser nogen signifikant forskel på de to grupper ($P=0.49$).

```
survdifff( Surv(months,dead) ~ azathiop, data=wr )
```

Call:

```
survdifff(formula = Surv(months, dead) ~ azathiop, data = wr)
```

	N	Observed	Expected	$(O-E)^2/E$	$(O-E)^2/V$
azathiop=nej	98	50	46.6	0.253	0.469
azathiop=ja	109	53	56.4	0.209	0.469

Chisq= 0.5 on 1 degrees of freedom, p= 0.5

c) et estimat for hazard ratioen for azathiop-behandlede vs. placebo-behandlede med konfidensinterval

```
cox1 <- coxph( Surv(months,dead) ~ azathiop, data=wr)
summary(cox1)
```

Call:

```
coxph(formula = Surv(months, dead) ~ azathiop, data = wr)
```

n= 207, number of events= 103

	coef	exp(coef)	se(coef)	z	Pr(> z)
azathiopja	-0.1358	0.8730	0.1984	-0.684	0.494

	exp(coef)	exp(-coef)	lower .95	upper .95
azathiopja	0.873	1.145	0.5917	1.288

Concordance= 0.503 (se = 0.027)
Rsquare= 0.002 (max possible= 0.991)
Likelihood ratio test= 0.47 on 1 df, p=0.5
Wald test = 0.47 on 1 df, p=0.5
Score (logrank) test = 0.47 on 1 df, p=0.5

Idet R automatisk vælger det laveste niveau af den forklarende variabel som reference (her `azathiop=0 (=nej)`), fik vi hazard ratio for de azathioprine-behandlede vs de placebo-behandlede direkte skrevet ud i outputtet ovenfor som 0.873 (under `exp(coef)`). Konfidensgrænser finder vi til (0.592, 1.288).

Bemærk at vi i vores summary ovenfor også får HR for de placebo-behandlede vs de azathiop-behandlede (under `exp(-coef)`). Vi kan dog også ændre referencegruppen til `ja` i selve analysen:

```
cox2 <- coxph( Surv(months,dead) ~ relevel( azathiop, 'ja' ) , data=wr)  
summary(cox2)
```

Call:

```
coxph(formula = Surv(months, dead) ~ relevel(azathiop, "ja"),  
      data = wr)
```

n= 207, number of events= 103

	coef	exp(coef)	se(coef)	z	Pr(> z)
relevel(azathiop, "ja")nej	0.1358	1.1454	0.1984	0.684	0.494

	exp(coef)	exp(-coef)	lower .95	upper .95
relevel(azathiop, "ja")nej	1.145	0.873	0.7764	1.69

Concordance= 0.503 (se = 0.027)
Rsquare= 0.002 (max possible= 0.991)
Likelihood ratio test= 0.47 on 1 df, p=0.5
Wald test = 0.47 on 1 df, p=0.5
Score (logrank) test = 0.47 on 1 df, p=0.5

Vi ser, at konfidensgrænserne bliver (0.776, 1.690).

I begge tilfælde er der tale om konfidensgrænser, der indeholder 1, svarende til, at der *ikke* er signifikant forskel på dødeligheden i de to grupper.

2

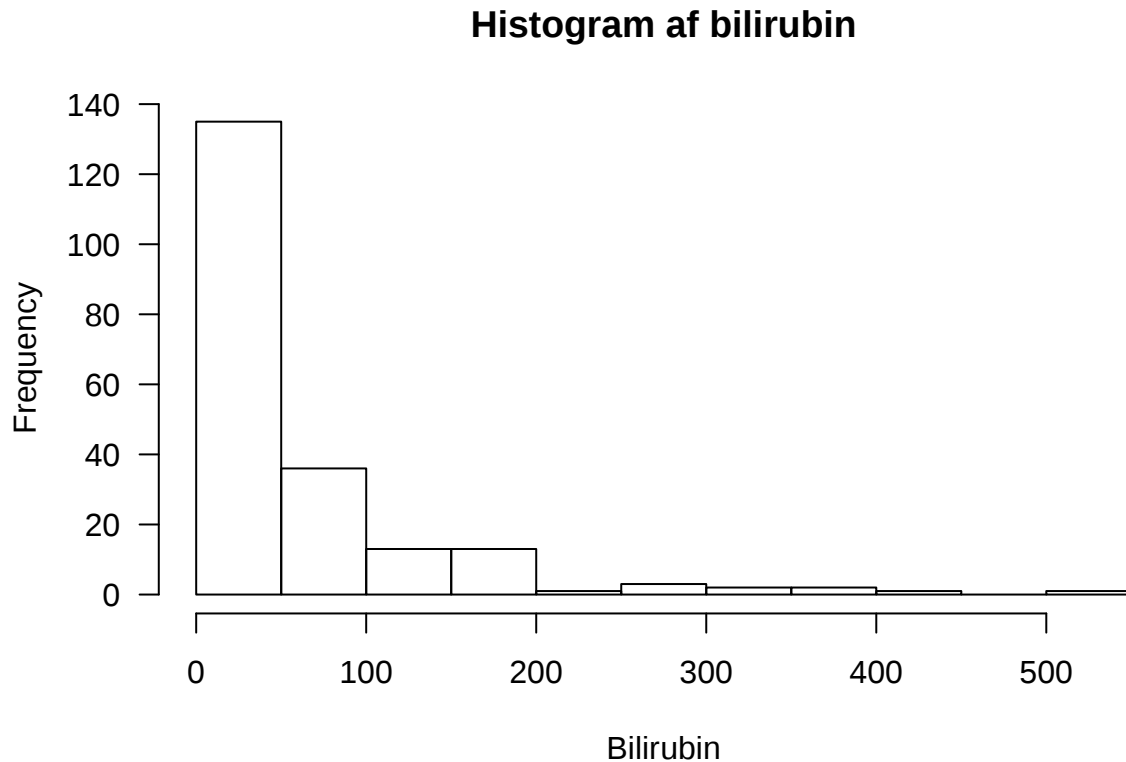
Serum bilirubin niveauet er generelt en meget stærk prognostisk faktor for patienter med leversygdom. Det er derfor relevant at undersøge betydningen af baseline serum bilirubin for prognosen hos patienter med primær biliær cirrhose.

Vurdér først ud fra nedenstående kriterier, om serum bilirubin bør transformeres:

- a) *ved at se på fordelingen af serum bilirubin – er der ekstreme observationer?*

Et histogram over alle værdierne set for bilirubin viser tydeligt, at der er tale om en meget skæv fordeling:

```
par( las=1 )  
hist( wr$bilirubin, main = 'Histogram af bilirubin', xlab = 'Bilirubin' )
```



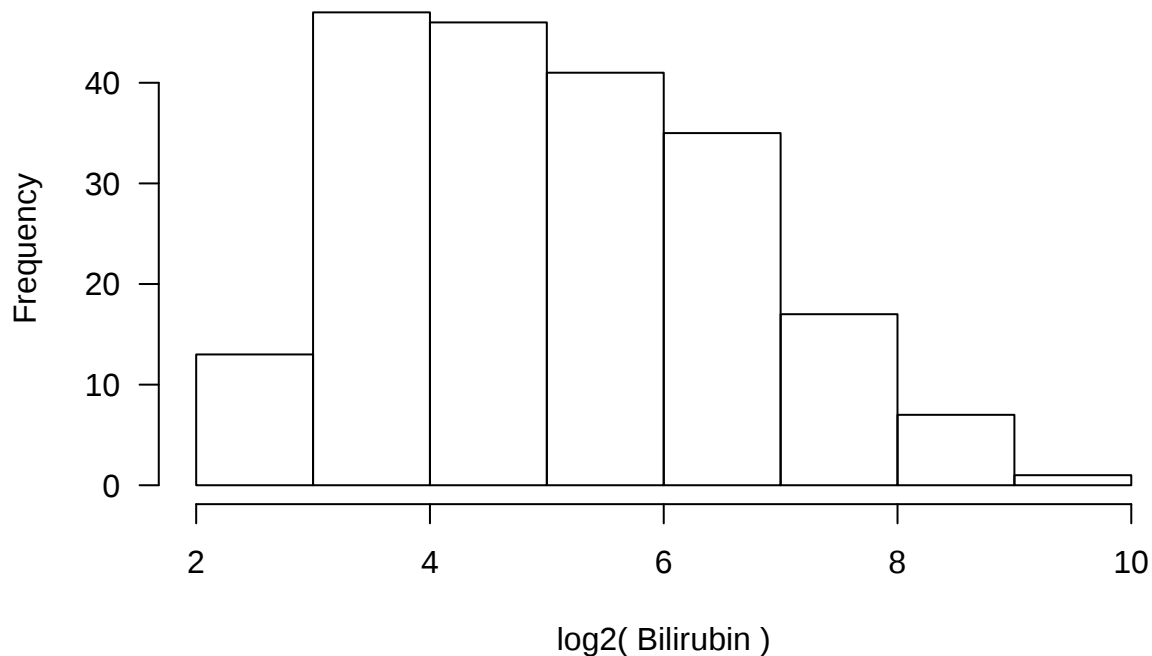
Bemærk at jeg her satte grafik-option `par( las=1 )` som sikrer, at tallene på y-aksen står horisontalt i stedet for vertikalt. Kommandoen skal bare køres én gang, så gælder den for alle plots vi laver fra nu af. Den kan rettes tilbage ved at køre `par( las=0 )` (`las` er en skør forkortelse af 'Axis Label orientation').

Skævheden er ikke noget problem i sig selv, da der ikke er noget som helst krav til fordelingen af en kovariat. *Men:* I en så skæv fordeling kan der være meget indflydelsesrige observationer blandt de få, meget høje værdier, og det kan derfor være meget ustabil at bibeholde denne skala. Logaritmetransformation kan bruges til at 'trykke' ekstremt høje værdier ind mod midten.

Vi ser derfor også på logaritmetransformeret bilirubin. Her er valgt logaritmen med grundtal 2:

```
wr$log2bili <- log2( wr$bilirubin )  
hist( wr$log2bili, main = 'Histogram af logaritmetransformeret bilirubin',  
      xlab = 'log2( Bilirubin )' )
```

Histogram af logaritmetransformeret bilirubin



Her ser vi en meget mere jævn fordeling uden meget ekstreme værdier. Det er derfor relevant at undersøge, om utransformeret bilirubin eller logaritmetransformeret bilirubin er mest relevant som prognostisk variabel – for det vigtigste er jo, at det er en passende model for sammenhængen mellem bilirubin og dødsraten.

b) *ved at inkludere serum bilirubin både utransformeret og passende transformeret på samme tid*

Dette kan gøres enten med eller uden behandlingen som kovariat, og vi ser på begge muligheder nedenfor. Først uden behandlingseffekt:

```
cox2 <- coxph( Surv(months,dead) ~ bilirubin + log2bili, data=wr)
summary(cox2)
```

Call:

```
coxph(formula = Surv(months, dead) ~ bilirubin + log2bili, data = wr)
```

n= 207, number of events= 103

	coef	exp(coef)	se(coef)	z	Pr(> z)
bilirubin	-0.001839	0.998163	0.001951	-0.943	0.346
log2bili	0.806816	2.240761	0.139753	5.773	7.78e-09 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

	exp(coef)	exp(-coef)	lower .95	upper .95
bilirubin	0.9982	1.0018	0.9944	1.002
log2bili	2.2408	0.4463	1.7039	2.947

Concordance= 0.776 (se = 0.022)

Rsquare= 0.357 (max possible= 0.991)

Likelihood ratio test= 91.39 on 2 df, p=<2e-16

Wald test = 83.4 on 2 df, p=<2e-16

Score (logrank) test = 115.5 on 2 df, p=<2e-16

Ovenstående output viser klart, at

- den logaritmetransformerede version af bilirubin er den med stærkest statistisk signifikans, mens den utransformerede kan udelades
- sammenhængen flader en smule *mere* ud end logaritme-transformationen (koefficienten til bilirubin er negativ).

Og så til analysen *med* behandlingseffekt:

```
cox3 <- coxph( Surv(months,dead) ~ azathiop + bilirubin + log2bili, data=wr)
summary(cox3)
```

Call:

```
coxph(formula = Surv(months, dead) ~ azathiop + bilirubin + log2bili,
      data = wr)
```

n= 207, number of events= 103

	coef	exp(coef)	se(coef)	z	Pr(> z)
azathiopja	-0.509680	0.600688	0.205912	-2.475	0.0133 *
bilirubin	-0.001615	0.998387	0.001964	-0.822	0.4109
log2bili	0.841057	2.318818	0.142092	5.919	3.24e-09 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

	exp(coef)	exp(-coef)	lower .95	upper .95
azathiopja	0.6007	1.6648	0.4012	0.8993
bilirubin	0.9984	1.0016	0.9946	1.0022
log2bili	2.3188	0.4313	1.7552	3.0635

Concordance= 0.779 (se = 0.022)
Rsquare= 0.376 (max possible= 0.991)
Likelihood ratio test= 97.46 on 3 df, p=<2e-16
Wald test = 87.05 on 3 df, p=<2e-16
Score (logrank) test = 120.1 on 3 df, p=<2e-16

der ses at vise stort set det samme, for så vidt angår effekten af bilirubin.

Vi noterer os dog, at behandlingseffekten nu er blevet signifikant - hvilket vi skal komme tilbage til nedenfor.

c) *ved at modellere effekten af serum bilirubin ved hjælp af en lineær spline*

Dette er svært... først skal vi have fundet nogle fornuftige steder til opdeling af splinen, altså nogle steder (værdier af bilirubin), hvor vi tillader kurven at knække. I mangel af bedre viden om forløbet, kan man argumentere for at opdele efter kvartiler af bilirubin, men for at skaffe ca lige mange dødsfald i hver gruppe, vil vi benytte kvartilerne udregnet udelukkende blandt de døde.

Vi definerer derfor et sub-datasæt bestående af de døde og bestemmer kvartilerne:

```
wr1 <- subset( wr, dead==1 )
quantile( wr1$bilirubin, probs=c(.25,.5,.75) )
```

25%	50%	75%
31.0	54.0	119.5

Vi vælger altså tærskelværdierne 31, 54 og 119.5 for bilirubin og tilføjer 3 splinevariable baseret på disse værdier til datasættet:

```
wr$bili25 = (wr$bilirubin-31)*(wr$bilirubin>31);
wr$bili50 = (wr$bilirubin-54)*(wr$bilirubin>54);
wr$bili75 = (wr$bilirubin-119.5)*(wr$bilirubin>119.5);
```

Vi er dermed klar til at benytte disse ekstra 3 variable sammen med selve bilirubin. Vi laver samtidigt et test for om vi kan undvære alle 3 spline variable ved at teste den lineære hypotese, at alle splinevariablene er 0:

```
cox4 <- coxph( Surv(months,dead) ~ azathiop + bilirubin + bili25+bili50+bili75, data=wr)

linearHypothesis( cox4, c('bili25=0','bili50=0','bili75=0'))
```

Linear hypothesis test

```
Hypothesis:
bili25 = 0
bili50 = 0
bili75 = 0
```

Model 1: restricted model

Model 2: Surv(months, dead) ~ azathiop + bilirubin + bili25 + bili50 +
bili75

```
Res.Df Df Chisq Pr(>Chisq)
1      205
2      202  3 34.18  1.816e-07 ***
---
```

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Output viser, at der *ikke* er en lineær sammenhæng med baseline serum bilirubin! Testet for linearitet giver $P<.0001$.

Det numeriske test bør dog aldrig stå alene – man skal altid tegne den lineære spline. Hvis P -værdien er lav, men det er sammen med en meget savtakket kurve omkring en generelt lineær trend, kan en lineær sammenhæng stadig være den mest relevante model at bruge.

Når vi skal tegne den lineære spline bestemmer vi først de prædikterede værdier på log-hazard-skala. Hertil definerer vi først et nyt datasæt svarende til de behandlede (`azathiop='ja'`) og med bilirubin værdier som observeret. Desuden tilføjes spline variablene til dette fiktive datasæt (vi skal have præcis samme variable som benyttet i vores model `cox4`). Tilsvarende prædikterer vi for det datasæt svarende til de ubehandlede (`azathiop='nej'`):

```
range( wr$bilirubin )

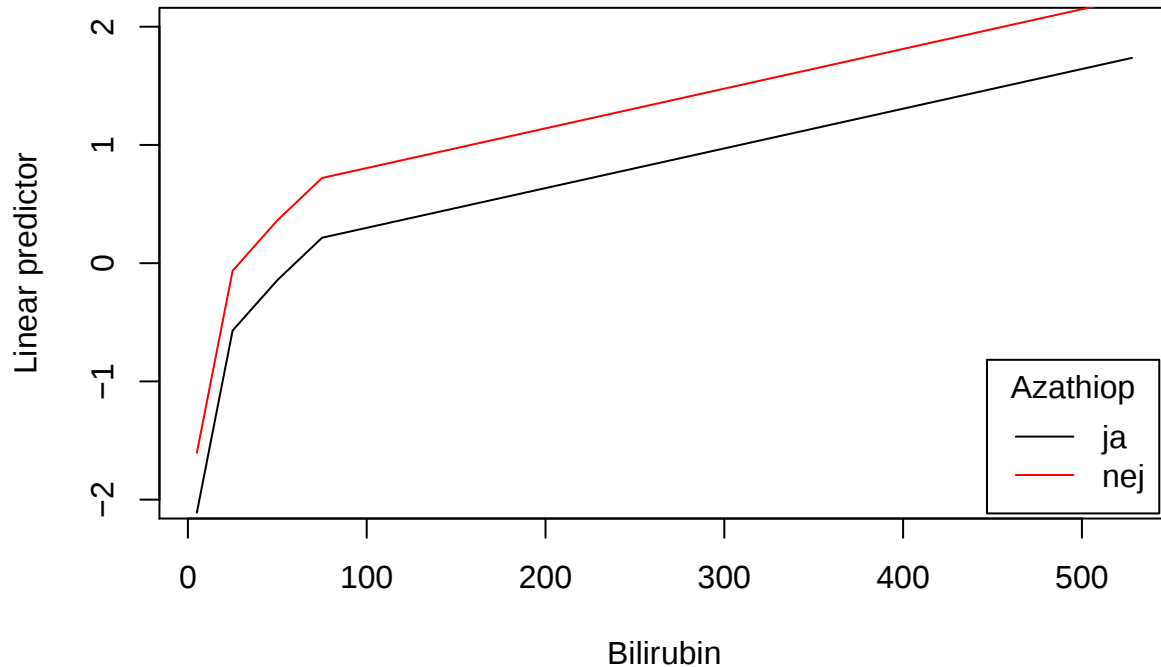
[1]    5 528
b <- 5:528

newDja <- data.frame(azathiop='ja',bilirubin=b,
                     bili25=(b-25)*(b>25),
                     bili50=(b-50)*(b>50),
                     bili75=(b-75)*(b>75))
lp.ja <- predict( cox4, newdata= newDja, type='lp' )

newDnej <- newDja; newDnej$azathiop<-'nej'
lp.nej <- predict( cox4, newdata= newDnej, type='lp' )
```

Vi plotter de prædikterede værdier som funktion af bilirubin for hver behandling:

```
plot( b, lp.ja, type='l', ylim=c(-2,2),
      xlab="Bilirubin", ylab="Linear predictor")
lines(b, lp.nej, col='red')
legend('bottomright',c('ja','nej'),col=1:2,inset=.01,lty=1,title='Azathiop')
```



Vi ser her en påfaldende lighed med en logaritme-funktion. Hvis man kørte hele proceduren med en lineær spline igennem med `log2bili` som variabel i stedet for `bilirubin`, ville man faktisk finde en næsten perfekt lineær sammenhæng.

3

Estimer og fortolk baseline serum bilirubins betydning for dødsraten.

Alt under spørgsmål 2 viser, at vi bør se på logaritmetransformeret bilirubin. Vi vælger $\log_2$ -transformeret bilirubin, fordi så får vi i outputtet vist effekten af en fordobling af serum bilirubin direkte som effekttestimatet.

```
cox6 <- coxph( Surv(months,dead) ~ azathiop + log2bili , data=wr)
summary(cox6)
```

Call:

```
coxph(formula = Surv(months, dead) ~ azathiop + log2bili, data = wr)
```

n= 207, number of events= 103

	coef	exp(coef)	se(coef)	z	Pr(> z)
azathiopja	-0.5201	0.5945	0.2061	-2.524	0.0116 *
log2bili	0.7425	2.1012	0.0773	9.605	<2e-16 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

	exp(coef)	exp(-coef)	lower .95	upper .95
azathiopja	0.5945	1.6822	0.3969	0.8903

log2bili 2.1011 0.4759 1.8058 2.4449

Concordance= 0.778 (se = 0.022)
Rsquare= 0.373 (max possible= 0.991)
Likelihood ratio test= 96.74 on 2 df, p=<2e-16
Wald test = 92.26 on 2 df, p=<2e-16
Score (logrank) test = 103.6 on 2 df, p=<2e-16

En dobbelt så høj baseline serum bilirubin giver en 2.10 (95% konfidensinterval: 1.81 til 2.44) gange større dødsrate, altså ca. en fordobling af dødsraten.

4

Estimer og fortolk effekten af azathioprine, justeret for effekten af bilirubin.

Vi ser nu på det andet aspekt af analysen ovenfor, nemlig effekten af behandling med azathioprin. Vi konstaterede allerede tidligere, at denne var blevet signifikant efter justering med bilirubin, og *P*-værdien i analysen ovenfor er 0.012. Vi skal nu kvantificere effekten.

Konklusionen ses at være, at den estimerede effekt af behandling med azathioprine er en dødsrate på kun 59.5% (95% konfidensinterval fra 39.7% til 89.0%) af dødsraten blandt placebo-behandlede med samme baseline serum bilirubin.

Det var lidt af en forskel fra den ujusterede analyse...!

5

Hvordan ville du afrapportere resultaterne i en artikel?

Her bør man tage i betragtning,

- at Cochran centeret har vist, at justerede analyser i gennemsnit overestimerer de faktiske behandlingseffekter (sikkert fordi folk tror på behandlingseffekten og derfor tror, at nu har de fundet den "rigtige" model, når den estimerede effekt tilfældigvis bliver stærkere).
- at randomisering ikke sikrer, at man får helt ens fordelinger af confoundere, kun at der ikke er nogen systematiske forskelle til fordel for en bestemt behandlingsgruppe
- hvad man selv skrev i protokollen (så man undgår fisketure)

6

Lav modelkontrol for Cox modellen, primært for effekten af den randomiserede behandling, men hvis der er tid, også for effekten af bilirubin. Modelkontrollen indbefatter

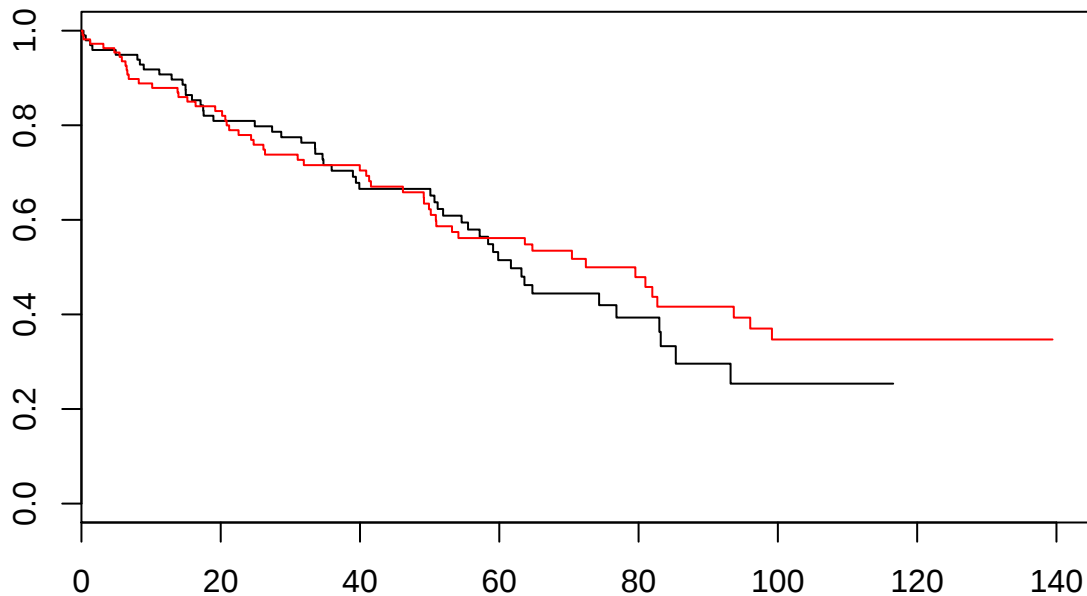
- Grafisk modelkontrol af proportionale rater for behandling*

Vi skal her se på antagelsen om proportionale intensiteter, hvilket gøres ved at tegne 'log minus log' af overlevelseskurverne. Hvis der er proportionale intensiteter, skal disse kurver være parallelle.

Vi ser først på behandlingseffekten, som er den nemmeste, fordi der kun er to niveauer (azathioprin eller placebo).

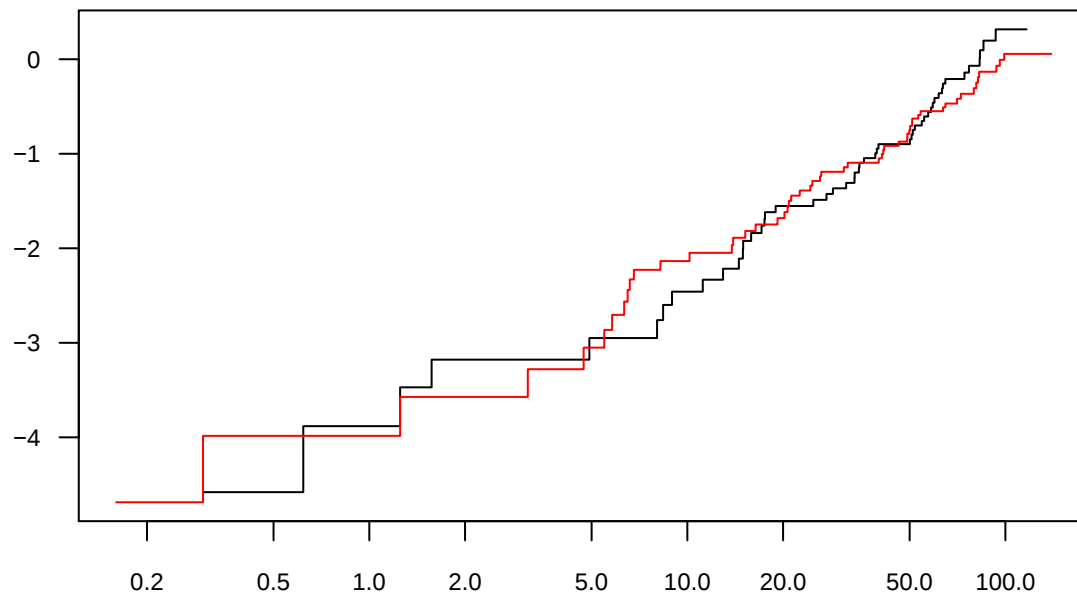
Overlevelseskurverne fandt vi i opgave 1 vha `prodlm` men når vi skal bestemme log-minus-log må vi gøre det vha `survfit()`. Denne er standardfunktionen til at tegne Kaplan-Meier plot i `survival` pakken, de bliver bare ikke så pæne som i `prodlm`:

```
km1 <- survfit( Surv(months,dead) ~ azathiop, data=wr)
plot( km1, col=c(1,2) )
```



Til gengæld er det nemt at tegne log-minus-log-kurverne:

```
plot( km1, col=c(1,2), fun='cloglog', cex.axis=.75)
```



Bemærk her, at R vælger at benytte en logaritmisk tidsakse. Ofte kan det være næsten umuligt at bedømme den lodrette afstand mellem kurverne i den første del af perioden, fordi raterne ofte er særligt høje i begyndelsen, så kurverne er meget stejle. Det afhjælper man med den logaritmiske tidsakse.

De to kurver ses at krydse over hinanden flere gange frem til 5 år (60 måneder). Derefter er der lidt varierende lodret afstand mellem kurverne, hvorefter de muligvis bliver lidt mere forskellige.

b) Grafisk modelkontrol af proportionale rater for bilirubin

For at checke antagelsen om proportionale intensiteter for bilirubin, er vi nødt til at foretage en inddeling i et par intervaller (vi kan jo ikke checke proportionalitet af kurver svarende til hver enkelt forekommende værdi af bilirubin). Vi vælger her at opdele i 3 intervaller, svarende til fraktilerne 33.33% og 66.67%, igen kun for de døde. Disse findes ud fra datasættet `wr1` bestående af de døde benyttet ovenfor:

```
quantile( wr1$bilirubin, probs=c( .3333, .6667) )
```

```
33.33% 66.67%  
37.0000 92.0136
```

Vi vil altså opdele efter bilirubin-værdierne 37 og 92 og får brug for en ny variabel svarende til denne inddeling. Hertil vil vi benytte funktionen `cut()`, som skærer en kvantitativ variabel i bidder. Her skal man dog også angive en øvre og nedre grænse for ydergrupperne og vælger man den nedre/øvre grænse for højt/lavt i forhold de værdier man har observeret, sættes værdierne uden for disse grænser til manglende. Vi checker derfor først hvilke værdier bilirubin antager:

```
range( wr$bilirubin )
```

```
[1] 5 528
```

og prøver at bruge funktionen og laver samtidigt et check af, at inddelingen ser fornuftig ud (som man altid bør gøre når man laver nye variable):

```
wr$biliGrp <- cut( wr$bilirubin, c( 5,37,92,528 ))  
table( wr$biliGrp, useNA='ifany' )
```

```
(5,37] (37,92] (92,528] <NA>  
113      55      38      1
```

Vi ser, at vi her har en manglende værdi. Det skyldes at det nederste skæringspunkt ikke indgår i intervallet (til at illustrere dette benytter R '(' i definitionen af grupperne, hvorimod '[' betyder at værdien er med i intervallet). Vi må derfor modificere vores variabel ved at sætte den laveste værdi f.eks 1 lavere end vi har observeret:

```
wr$biliGrp <- cut( wr$bilirubin, c( 5-1,37,92,528 ))  
table( wr$biliGrp, useNA='ifany' )
```

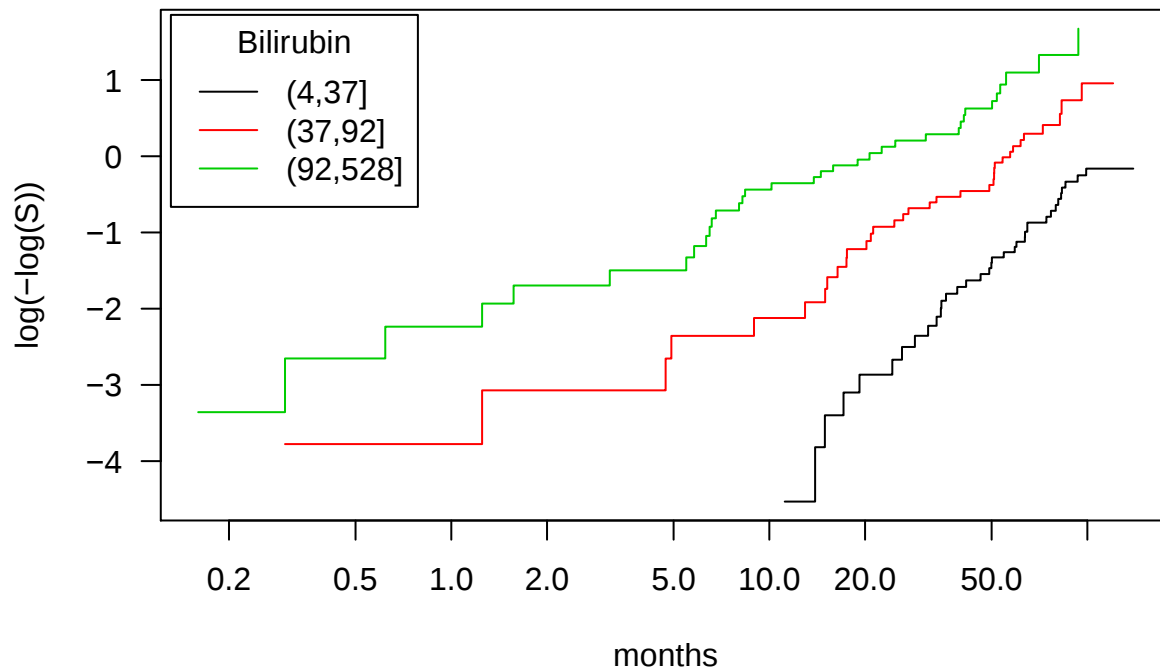
```
(4,37] (37,92] (92,528]  
114      55      38
```

Vi udfører nu Cox-regressionsmodellen, hvor vi stratificerer på bilirubin i grupper frem for at have den som en kvantitativ variabel:

```
cox7 <- coxph( Surv(months,dead) ~ azathiop + strata( biliGrp ), data=wr )
```

Til modelkontrol vil vi igen benytte log-minus-log kurverne. Disse kan vi få frem ved at benytte funktionen `survfit()` og derefter lave plot på cloglog-skala:

```
newD=data.frame( azathiop=0) # vælg vilkårlig værdi for azathiop  
survfit7=survfit( cox7, newdata=newD )  
  
plot( survfit7, fun="cloglog", ylab="log(-log(S))", col=1:3,  
      mark.time=F, xlab="months")  
legend("topleft",levels(wr$biliGrp),lty=1,col=1:3,title="Bilirubin",inset=.01)
```



Proportionaliteten synes at være vældig god for bilirubin.

De næste 3 punkter tager vi under et:

- c) *Numerisk test af proportionale rater for behandling*
- d) *Numerisk test af proportionale rater for bilirubin*
- e) *Check af linearitetsantagelsen for bilirubin (på den valgte skala)*

I R kan disse tests laves ved hjælp af `cox.aalen()`-funktionen i `timereg`-pakken. først skal Cox modellen køres endnu en gang ved brug af `cox.aalen()`, og her er man nødt til at fortælle at vi ønsker at arbejde med Cox modellen hvor alle kovariater har proportionale effekter (`cox.aalen()` kan håndtere modeller, hvor effekterne ikke er proportionale). Det gør man ved at skrive `prop()` (PROPortional) omkring hver kovariat. Da metoden er baseret på simulation, er det en god idé at vælge et seed som udgangspunkt for beregningerne, så man får præcis samme p-værdi hver gang man laver beregningen:

```
set.seed( 1234 ) # vælg et hvilket som helst tal
ca1 <- cox.aalen( Surv(months,dead) ~ prop( azathiop ) + prop( log2bili ),
                 data=wr)
summary( ca1 )
```

Cox-Aalen Model

Test for Aalen terms

Test not computed, sim=0

Proportional Cox terms :

	Coef.	SE Robust	SE D2log(L) ⁻¹	z	P-val
prop(azathiop)ja	-0.520	0.2070	0.2040	0.2060	-2.55 0.0107
prop(log2bili)	0.743	0.0708	0.0661	0.0773	11.20 0.0000
	lower2.5%	upper97.5%			
prop(azathiop)ja	-0.926	-0.114			
prop(log2bili)	0.604	0.882			

Test of Proportionality

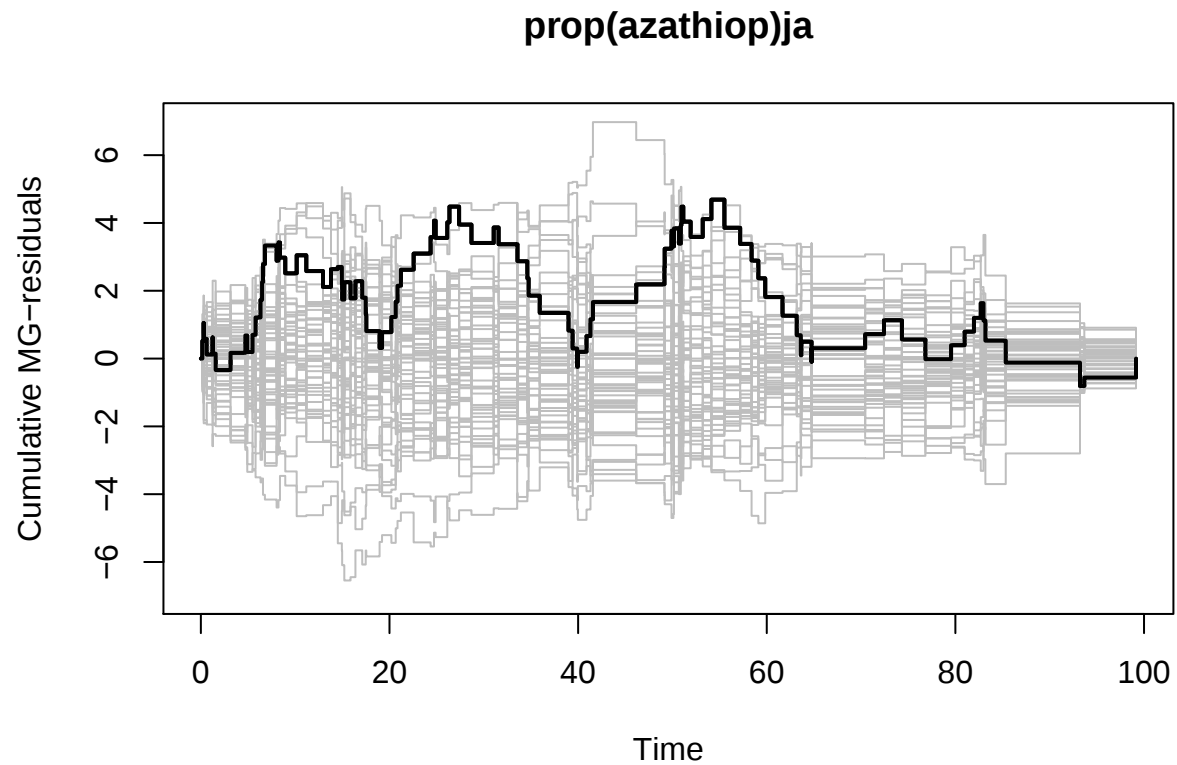
	supl	hat U(t)	p-value	H_0
prop(azathiop)ja		4.69	0.292	

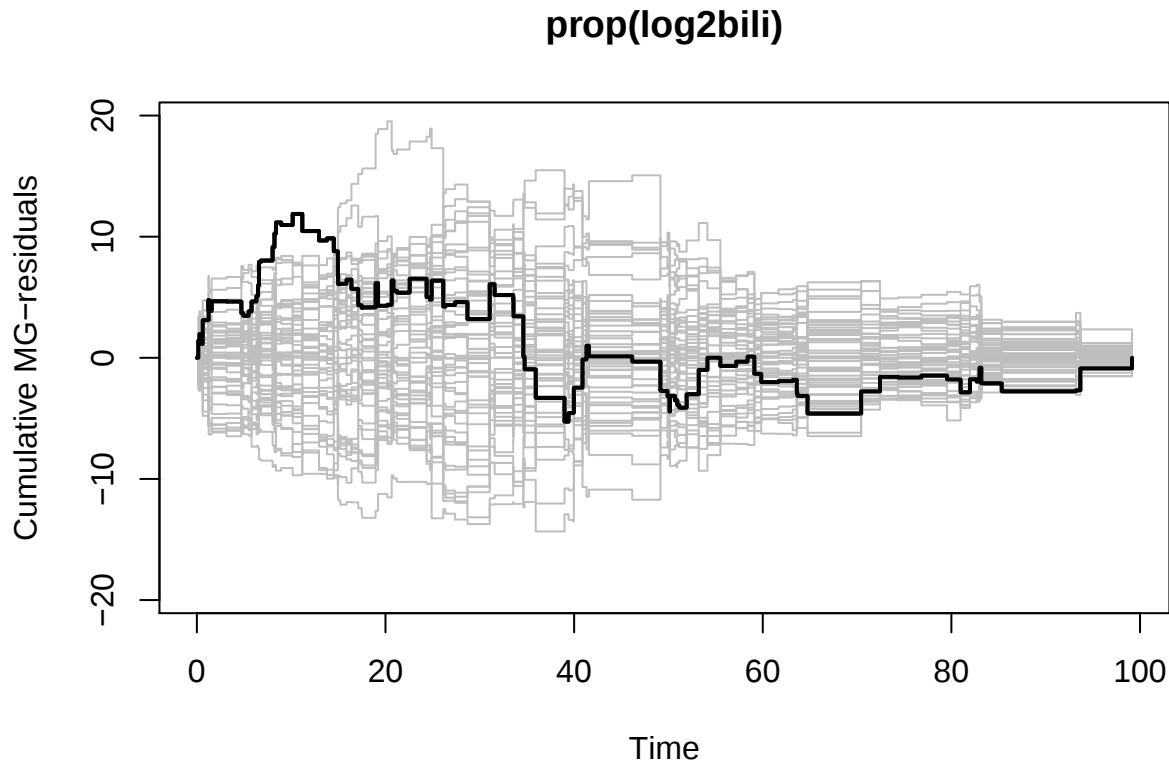
```
prop(log2bili)          11.90          0.280
```

P-værdien for test af proportionale rater aflæses under **p-value** H_0 . Her ses ingen tegn på problemer med antagelsen om proportionale rater ($P=0.292$ hhv $P=0.28$ for azathiop hhv bilirubin)

Man kan i tilgift få figurer, som kan give fingerpeg om, hvordan afvigelserne fra antagelserne evt kunne se ud:

```
plot( ca1, score=T )
```





Den øverste figur, for behandlingseffekten, viser, at personer med “høje værdier” af azathiop1 (dvs. de behandlede) dør lidt for hurtigt til at starte med (det var jo også her, de kumulerede rater var nærmest ens). Det kan jo være, at behandlingseffekten først sætter ind efter et stykke tid...

For bilirubin ser vi ligeledes, at dem med høj værdi af bilirubin dør “for hurtigt” til at starte med, hvilket kunne være en effekt af, at det drejer sig om en værdi målt ved baseline, og denne har formentlig større prognostisk værdi i starten (da den kan ændre sig hen ad vejen).

Vi undersøger dernæst linearitetsantagelsen for log2bili ved et formelt test. Igen kan vi benytte `cox.aalen()` dog med lidt andre specifikationer:

```
set.seed( 2345 ) # vælg et hvilket som helst tal
ca2 <- cox.aalen( Surv(months,dead) ~ prop( azathiop ) + prop( log2bili ),
                 residuals=1, n.sim=0, data=wr )
resids <- cum.residuals( ca2, cum.resid=1, data=wr )
summary(resids)
```

Test for cumulative MG-residuals

Grouped cumulative residuals not computed, you must provide
modelmatrix to get these (see help)

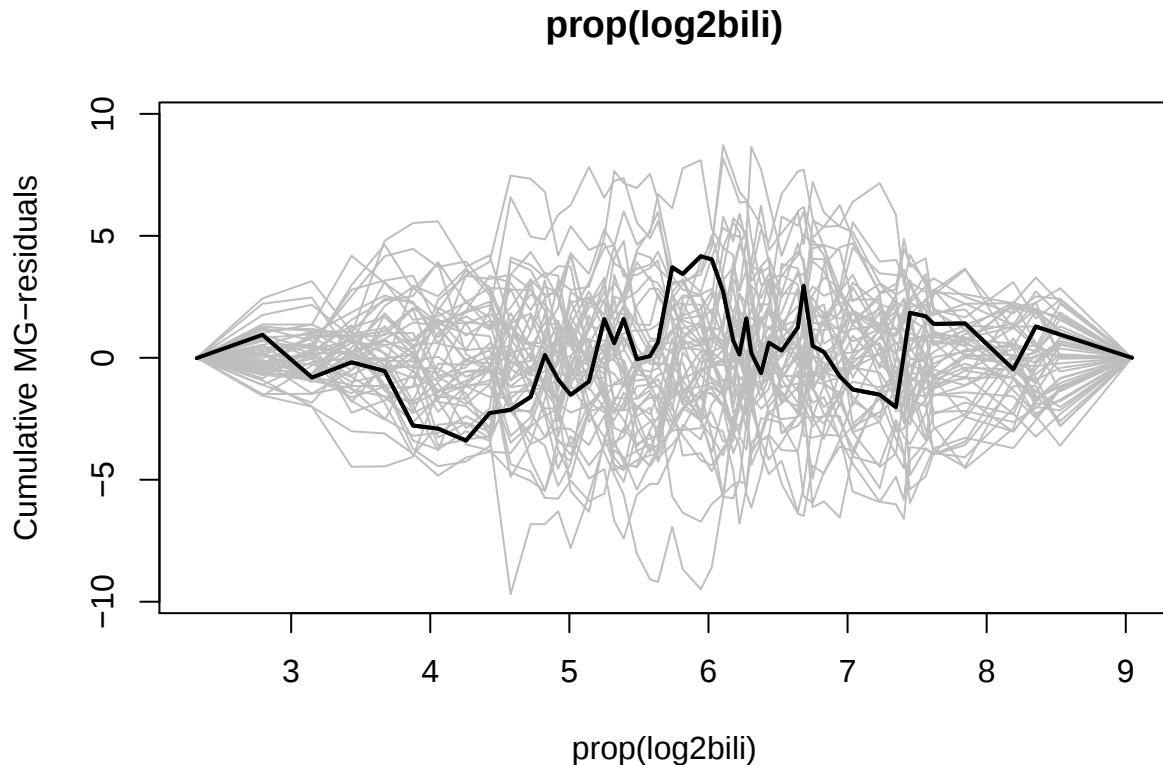
Residual versus covariates consistent with model

	sup hat B(t)	p-value	H_0: B(t)=0
prop(log2bili)	4.161	0.858	

Her får vi en P-værdi på 0.858 og derfor intet tegn på at der skulle være problemer med lineariteten.

Igen kan vi lave et plot:

```
plot( resids, score=2 )
```



Denne figur viser kun ukarakteristiske svingninger, svarende til en rigtig god linearitetsantagelse, som vi også fandt ud af i spørgsmål 3c, hvor vi modellerede effekten som en lineær spline.

Ovenstående er en nem metode til at lave modelkontrol, men man kan supplere med mere specifikke tests, i form af tidsvarierende effekter.

Man kan så vælge nogle passende tidspunkter, hvor proportionalitetsfaktoren kan få lov til at ændre sig. En god måde at vælge disse opdelingstidspunkter er at undersøge, om de kumulerede rater er nogenlunde lineære i nogle intervaller. Hvis de kumulerede rater er nogenlunde lineære, så er raterne nogenlunde konstante i de samme intervaller. Hvis raterne er konstante i begge grupper i et interval, er de selvfølgelig også proportionale i hele det interval.

Hvis vi f.eks. ikke antager proportionale intensiteter for behandling i hele tidsrummet, men tillader proportionalitetskonstanten at variere mellem f.eks. 3 tidsintervaller (her valgt med tærskelværdier 12 og 56 måneder, baseret på figuren fra spørgsmål 6a (ca der hvor der er overkrydsninger)), kan vi skrive:

```
cox8 <- coxph( Surv(months, dead) ~ tt(azathiop)+log2bili, data=wr,
  tt=function(x, t, ...) {
    cbind(az1=(x=='ja')*(t<=12), az2= (x=='ja')*(12<t&t<=56), az3= (x=='ja')*(t>56) )
  })
```

Funktionen `tt` (Time Transform) hjælper med at lave en dummy-variabel for hver tidsperiode svarende til at vi får estimeret effekten af azathiop for hver periode separat.

Vi får output

```
summary(cox8)
```

Call:

```
coxph(formula = Surv(months, dead) ~ tt(azathiop) + log2bili,
  data = wr, tt = function(x, t, ...) {
    cbind(az1 = (x == "ja") * (t <= 12), az2 = (x == "ja") *
      (12 < t & t <= 56), az3 = (x == "ja") * (t > 56))
```

```
})
```

```
n= 207, number of events= 103
```

```
              coef exp(coef) se(coef)      z Pr(>|z|)
tt(azathiop)az1 -0.04899   0.95219  0.43589 -0.112  0.91052
tt(azathiop)az2 -0.42944   0.65088  0.27472 -1.563  0.11801
tt(azathiop)az3 -1.15536   0.31495  0.41058 -2.814  0.00489 **
log2bili         0.74406   2.10446  0.07756  9.593 < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
              exp(coef) exp(-coef) lower .95 upper .95
tt(azathiop)az1    0.9522    1.0502    0.4052    2.2375
tt(azathiop)az2    0.6509    1.5364    0.3799    1.1152
tt(azathiop)az3    0.3149    3.1752    0.1408    0.7042
log2bili            2.1045    0.4752    1.8077    2.4500
```

```
Concordance= 0.781 (se = 0.022 )
Rsquare= 0.385 (max possible= 0.991 )
Likelihood ratio test= 100.5 on 4 df, p=<2e-16
Wald test              = 95.47 on 4 df, p=<2e-16
Score (logrank) test = 107.4 on 4 df, p=<2e-16
```

hvoraf vi ser, at vi har en hazardratio på 0.95 i det første år, 0.65 i anden periode og 0.31 i tredje. Vi kan undersøge ved et formelt test om disse tre hazardratioer faktisk er forskellige:

```
linearHypothesis( cox8, c("tt(azathiop)az1=tt(azathiop)az2",
                          "tt(azathiop)az2=tt(azathiop)az3"))
```

Linear hypothesis test

Hypothesis:

```
tt(azathiop)az1 - tt(azathiop)az2 = 0
```

```
tt(azathiop)az2 - tt(azathiop)az3 = 0
```

Model 1: restricted model

```
Model 2: Surv(months, dead) ~ tt(azathiop) + log2bili
```

```
      Res.Df Df    Chisq Pr(>Chisq)
1         205              0.155
2         203  2  3.7292      0.155
```

Her finder vi altså heller ikke nogen indikation af, at proportionalitetskonstanterne skulle afvige væsentligt fra hinanden.

I stedet for at basere opdelingen på det visuelle indtryk af stykkevis linearitet for den kumulerede rate, kan man eksempelvis vælge at opdele tidsintervallerne, så der bliver ca. lige mange døde i hvert tidsinterval, fordi der så er ca. lige stor statistisk styrke i hvert interval. Dette prøver vi for effekten af bilirubin:

```
quantile( wr1$months, probs=c(.3333,.6667) )
```

```
      33.33%   66.67%
18.94521 50.16180
```

Benytter vi således 19 og 50 måneder som tærskelværdier, skriver vi:

```
cox9 <- coxph( Surv(months, dead) ~ azathiop+tt(log2bili), data=wr,
  tt=function(x, t, ...) {
    cbind(lb1=x*(t<=19), lb2= x*(19<t&t<=50), lb3= x*(t>50) )
  })

linearHypothesis( cox9, c("tt(log2bili)lb1=tt(log2bili)lb2",
  "tt(log2bili)lb2=tt(log2bili)lb3"))
```

Linear hypothesis test

Hypothesis:

$tt(\log_2bili)lb1 - tt(\log_2bili)lb2 = 0$

$tt(\log_2bili)lb2 - tt(\log_2bili)lb3 = 0$

Model 1: restricted model

Model 2: $\text{Surv}(\text{months}, \text{dead}) \sim \text{azathiop} + tt(\log_2bili)$

	Res.Df	Df	Chisq	Pr(>Chisq)
1	205			
2	203	2	2.162	0.3392

Det ser således ud til, at proportionalitetsantagelsen (=konstant effekt) for serum bilirubin holder rimeligt godt henover hele perioden.

Reference:

Christensen E, Neuberger J, Crowe J, Altman DG, Popper H, Portmann B, et al. Beneficial effect of azathioprine and prediction of prognosis in primary biliary cirrhosis. Final results of an international trial. *Gastroenterology* 1985;89(5):1084-91.