

Faculty of Health Sciences

Basal statistik

Overlevelsesanalyse

Lene Theil Skovgaard & Susanne Rosthøj

11. maj 2020



Overlevelsesanalyse

- ▶ Levetider og censurerede observationer
- ▶ Kaplan-Meier kurver
- ▶ Log-rank test
- ▶ Cox regression
- ▶ Modelkontrol og udvidede modeller

Hjemmesider:

http://publicifsv.sund.ku.dk/~lts/basal19_1

E-mail: ltsk@sund.ku.dk



Forklarende variable = Kovariater

Outcome

Outcome	Dikotom	Kategorisk	Kvantitativ	Kategoriske og kvantitative
Dikotom parret	2*2-tabeller Mc Nemar	χ^2 -test svært, mixed models		Logistisk regression Mixed models
Kategorisk	Kontingenstabeller/ χ^2 -test		Generaliseret logistisk regression	
Ordinale	svært, f.eks. proportional odds modeller			
Kvantitativ parret	Mann-Whitney Wilcoxon signed rank	Kruskal-Wallis Friedman		Robust multipel regression
Normalfordelte residualer	T-test uparret/parret	Variansanalyse ensidet/tosidet		Kovariansanalyse Multipel regression
Censureret	Log-rank test		Cox regression	
Korrelerede kvantitative Nf. residualer	Varianskomponent-modeller Mixed models		Modeller for gentagne målinger	



Eksempel: Lungecancer blandt krigsveteraner

137 krigsveteraner med diagnosen inoperabel lungecancer, af **4 forskellige typer** (1:squamous, 2:smallcell, 3:adeno, 4:other) følges frem til død (vi glemmer lige et øjeblik, at 9 af dem faktisk endnu er i live ved tidspunktet for analysen)

De er randomiseret til enten

- ▶ standard behandling ($\text{treat}=0$, $n=69$)
- ▶ ny test behandling ($\text{treat}=1$, $n=68$)

Vi vil gerne sammenligne disse to behandlinger ...og så har vi nogle kovariater (udover behandling og type), som vi vender tilbage til



Valg af tidsskala

Der skal vælges en *tidsskala* for levetiden, dvs. et nulpunkt.

Typiske muligheder er:

- ▶ alder
- ▶ kalendertid
- ▶ tid siden diagnose
- ▶ **tid siden randomisering** (vælges her)

Man kan f.eks.

- ▶ få diagnosen som 40-årig, indgå i undersøgelsen fra dette tidspunkt og dø som 72-årig....
- ▶ få diagnosen som 50-årig, randomiseres til undersøgelsen som 55-årig og dø som 65-årig....



Valg af tidsskala, fortsat

Ofte er der flere muligheder, og **valget er ikke altid helt oplagt**:

Der skal være en grund til at tro, at den valgte tidsskala afspejler risikoen ens for alle individer (på nær diverse kovariater), altså at der som funktion af tiden ses samme *form* på risikoen.

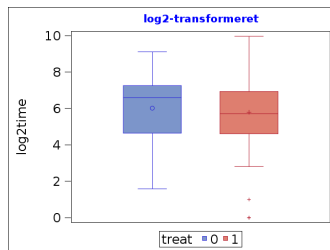
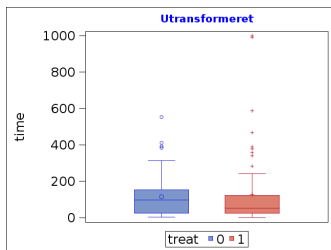
Eksempelvis kunne man vælge

- ▶ For kroniske sygdomme:
Alder: Fordi behandlingen formodentlig ikke ændrer radikalt ved prognosen
- ▶ For stærkt behandlingskrævende sygdomme:
Tid siden behandling: iden man kan formode, at en vellykket behandling på afgørende vis ændrer prognosen



Sammenligning af behandlinger

Hvis alle var fulgt indtil død, kunne vi lave T-test el.lign.



T-test på log-skala: $P=0.50$,

Mann-Whitney: $P=0.35$

men men men.....

Hvad er nyt?

Outcome: Tid fra et veldefineret nulpunkt og indtil en bestemt begivenhed (sædvanligvis død) indtræffer.

Her: Levetid i dage siden randomisering, angivet som y_i .

Det er her, der sker noget nyt, fordi det ikke er alle, der er døde endnu

Vi kalder det for censurerede observationer

Her er der kun 9 af sådanne, men ofte er der mange flere

Kovariater: Behandling, cancertype, alder, etc. etc.

Ikke noget *essentielt* nyt her....



Notation for outcome

t_i : Det tidspunkt (regnet fra nulpunktet),
der er tilknyttet det i 'te individ,
og som angiver *enten* død *eller* censureringstidspunkt.

d_i : Status ved tid t_i :

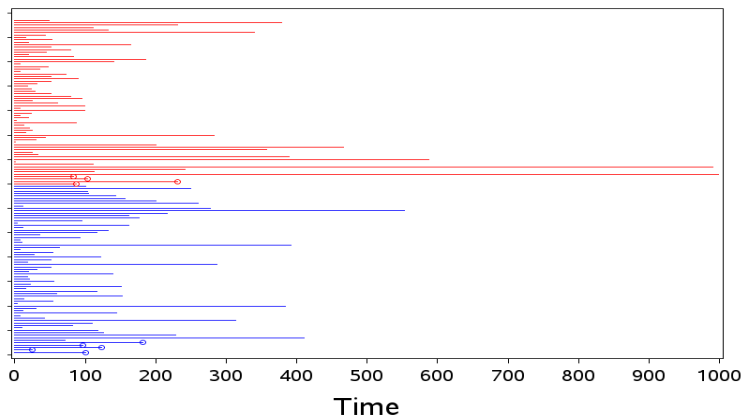
- ▶ 0: censureret, dvs. stadig i live
- ▶ 1: død

y_i : Tid til død for i 'te patient er så ikke observeret for alle:

- ▶ Hvis en patient dør til tid t_i , så er $y_i = t_i$ og $d_i = 1$
- ▶ Hvis man mister kontakten til en patient til tid t_i ,
så er $y_i > t_i$ og $d_i = 0$
- ▶ Hvis analysen laves til tid t_i , *inden* patient i dør,
så er $y_i > t_i$ og $d_i = 0$



Illustration af data

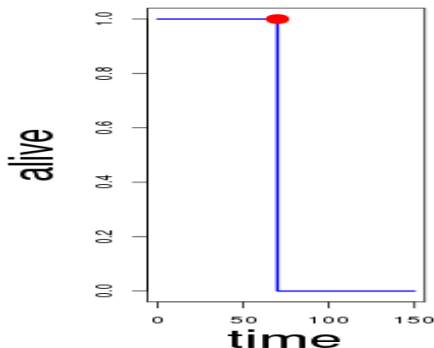


treat=0, treat=1

Cirklene angiver censurerede observationer

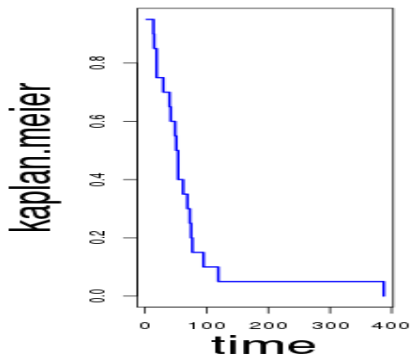
Hvad ved vi?

Vi vil opfatte tiden som *dynamisk*, i den forstand, at vi for hvert tidspunkt t ser på, hvilken information vi har om den enkelte, nemlig status for overlevelse: $S_i(t) = I(y_i > t)$



Kaplan-Meier: Overlevelseskurve

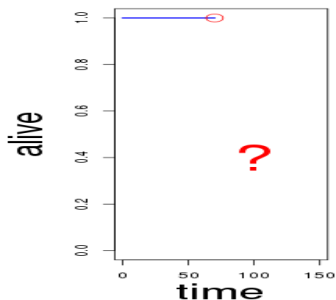
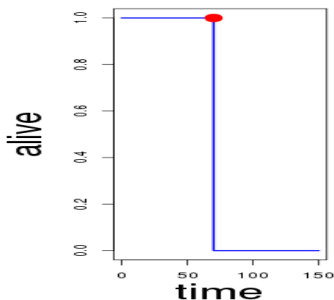
Hvis alle er observeret til død, er overlevelseskurven $S(t)$ blot gennemsnittet af disse, altså til hvert tidspunkt blot fraktionen, der stadig er i live. Den kaldes Kaplan-Meier kurven



Men når vi har censureringer?

så ved vi ikke, hvad der sker med funktionen $S_i(t) = I(y_i > t)$ efter det tidspunkt t_i , hvor vi sidst ser patienten i live.

Vi ved så kun, at deres levetid er større end t_i ,
men ikke *hvor meget* større.



Vigtigt - Uafhængige censureringer?

En **meget vigtig** antagelse om censureringer:

De skal være **uafhængige** af prognosen, dvs.

Det faktum, at en patient censureres ved tid t_i må på *ingen måde* være relateret til den fremtidige risiko for at dø

Censureringer, der muligvis *ikke* opfylder dette kriterium:

- ▶ Patienter, der overflyttes til et andet hospital:
De har formentlig en dårligere prognose
- ▶ Patienter, der flytter udenlands pga nyt job:
De har formentlig en bedre prognose

Og hvad, hvis

- ▶ patienten bliver kørt over?



Kaplan-Meier kurver, I

Når der er censureringer, kan vi ikke tage gennemsnittet af de individuelle S_i 'er, fordi vi ikke kender dem alle.

Men hvis censureringerne er uafhængige, kan vi estimere den del, vi ikke kender til blot at være "*ligesom dem, vi ser*", altså antage, at disse patienter dør i samme takt som de øvrige

Det er netop det, der ligger i uafhængig censurering.

Så hvis det, at man bliver kørt over, tyder på, at man har det dårligt (har fået recidiv), så har man nok på dette tidspunkt en værre prognose end de, der *ikke* blev kørt over....?



Håndtering af censureringer

Hvorfor kan vi ikke bare...

- ▶ Smide de censurerede individer helt væk?

Vi kender jo ikke deres levetid, så der er vel tale om *missing values*?

Nej, for vi ved *noget* om deres levetid, nemlig at den er mindst t_i , og hvis vi smider dem ud, laver vi derfor *bias* i overlevelseskurven

- ▶ Lade som om alle t_i 'er er dødsfald?

Fordi så ser det jo ud som om de dør alt for hurtigt, så her bliver også *bias*

Vi er nødt til at håndtere censureringer på en *dynamisk* måde, dvs. ved at se på situationen til et givet tidspunkt og et kort tidsrum herefter.



Kaplan-Meier kurver, II

når der er censureringer:

- ▶ Vi deler tidsaksen op ved hvert tidspunkt $t_1 < t_2 < t_3 \dots$ (her sker der noget, enten et dødsfald, eller en censurering)
- ▶ For hvert interval, f.eks. (t_3, t_4) ser vi på, hvor mange, der var *i risiko* i starten (n_3) og hvor stor en del af disse, der overlevede tidsintervallet, nemlig brøken

$$R_4 = \begin{cases} 1 & \text{hvis } t_4 \text{ er en censurering} \\ \frac{n_3-1}{n_3} & \text{hvis } t_4 \text{ er et dødsfald} \end{cases}$$

- ▶ Kaplan-Meier kurven fås nu ved successiv sammengangning af disse brøker:

$$1 \times R_1 \times R_2 \times R_3 \times R_4 \times \dots$$



Praktisk håndtering af censurering

Til programmering er outcome 2-delt:

- t_i Observationstiden for i 'te patient (time)
som kan være *enten* død (status=1), *eller* en
censurering (status=0).
- d_i Indikator for død (status), dvs. 1, hvis patienten
faktisk er død, og 0, hvis vedkommende er censureret

Begge disse variable skal angives som (todelt) outcome

I R gøres dette forskelligt afhængigt af hvilken pakke der benyttes:

`Surv(time,status)` i `survival` og `timereg`-pakkerne

`Hist(time,status)` i `prodlm`-pakken.



Kaplan-Meier kurver i R

kan tegnes vha survival-pakken med:

```
library( survival )  
km1 <- survfit( Surv(time,status) ~ treat, data=d )  
plot( km1 )
```

Hvis vi ønsker konfidensgrænser gøres det dog pænere vha prodlim-pakken:

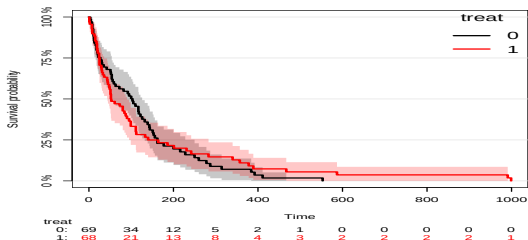
```
library(prodlim)  
km2 <- prodlim( Hist(time,status) ~ treat, data=d )  
plot( km2 )
```

Figuren ses s. 20



Kaplan-Meier kurver, $S(t)$

ved `prodlm` (se kode s. 19)

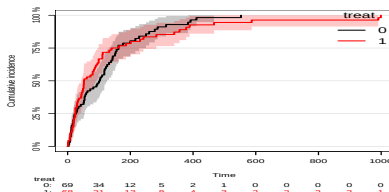


Til ethvert tidspunkt kan vi aflæse den forventede procentdel af patienter, der stadig er i live - for hver behandlingsgruppe

Kumulativ incidens

Til et hvert tidspunkt angiver dette sandsynligheden for at have fået et event, dvs. (i dette tilfælde) at være død. Denne er blot

$$CI(t) = 1 - S(t)$$



Figuren laves ved at plotte vores prodlim-objekt km2 med ekstra argument `type="cuminc"`.

Pas på i tilfælde af *competing risks* (se s. 82-83)

Er der forskel på de to behandlinger?

De to overlevelseskurver (s. 20) krydser lidt frem og tilbage over hinanden. De eksperimentelt behandlede dør hurtigst til at starte med, men efter ca. 100 dage ser det ud til at vende (bedømt på hældningen af kurverne).

Hvordan kan vi sammenligne?:

- ▶ **log-rank test:**
giver en P-værdi for test af identitet
- ▶ **Cox proportional hazard model:**
giver estimat for forskel (ratio) mellem dødsrater
– under *forudsætning* af proportionale rater
(kommer om lidt...)



Log-rank testet for H_0 : ens dødelighed

Et **non-parametrisk** test til sammenligning af overlevelseskurver **uden antagelser om formen af disse**

- ▶ Til hvert tidspunkt ser man på forskellen mellem observeret og forventet antal døde (under H_0) i de to grupper.
- ▶ Forskellen mellem disse, passende normeret, giver et χ^2 -test med $G - 1$ frihedsgrader (G er antallet af grupper, her $G = 2$)
- ▶ Det er ikke så stærkt, hvis forskellen på kurverne ændrer sig markant over tid.

Her giver testet $P = 0.93$ (se s. 24), altså **ingen påviselig forskel**

Testet udføres ved:

```
survdif( Surv(time,status) ~ treat, data=d ).
```



Output fra Log-rank test

```
> survdiff( Surv(time,status) ~ treat, data=d )  
Call:  
survdiff(formula = Surv(time, status) ~ treat, data = d)  
  
      N Observed Expected (O-E)^2/E (O-E)^2/V  
treat=0 69      64      64.5  0.00388  0.00823  
treat=1 68      64      63.5  0.00394  0.00823  
  
Chisq= 0 on 1 degrees of freedom, p= 0.928 <-- log-Rank test
```

Der er absolut ikke signifikant forskel -
men hvad med kvantificering af forskellen?



Rater = hazards = intensiteter

- ▶ Vi har set, at levetiden ikke dur som outcome, fordi en sådan ikke altid observeres.
- ▶ I stedet vil vi gerne modellere “hvor hurtigt overlevelseskurven går ned ad bakke”, altså en mere *dynamisk* beskrivelse, i form af den umiddelbare risiko for død (*rater*, *hazards* eller *intensiteter*):

Givet, at jeg er i live lige nu, hvad er så min risiko for at dø indenfor det næste år?

Eller næste måned? Næste dag/næste time....

- ▶ **Hazard funktion $h(t)$:** den *umiddelbare* risiko for at dø i det næste korte interval Δt er $h(t)\Delta t$

$h(t)$ er en rate, ikke en sandsynlighed



Hazard funktionen

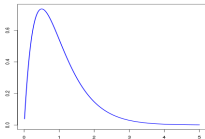
- ▶ er ikke så god at estimere, fordi den hopper op og ned (andel døde i hvert lille tidsrum er enten 0 eller et ret lille tal)
- ▶ estimer i stedet **kumuleret hazard, $H(t)$**
- ▶ Matematikken kan vise, at $H(t) = -\log(S(t))$
- ▶ Hældningen af $H(t)$ afspejler den umiddelbare risiko/rate/hazard $h(t)$ for at dø:
 - ▶ stejl hældning: stor risiko
 - ▶ lille hældning: lav risiko

Teknisk note: $H'(t) = -\frac{S'(t)}{S(t)}$, dvs. det er altså den *relative* ændring af overlevelseskurven, vi modellerer.

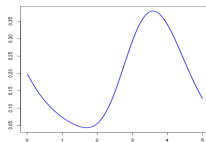


Eksempler på hazard-funktioner $h(t)$

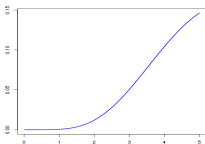
Umiddelbar farlig behandling:



Gavnlig behandling,
med senfølger:

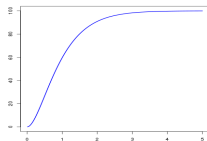


Alder som mest betydende:

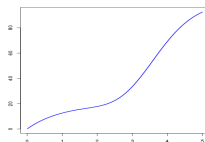


Kumulerede hazard-funktioner $H(t)$

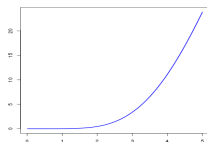
Umiddelbar farlig behandling:



Gavnlig behandling,
med senfølger:

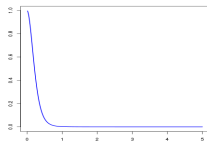


Alder som mest betydende:

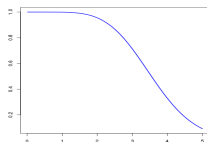


Overlevelseskurver $S(t)$

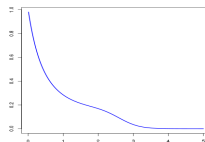
Umiddelbar farlig behandling:



Alder som mest betydende:



Gavnlig behandling,
med senfølger:

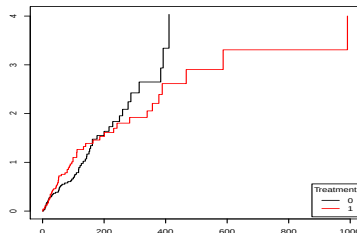


Kumulerede hazard funktioner

i vores eksempel om krigsveteraner plottes ved vores `survfit`-objekt `km1` med ekstra argument `fun`:

```
plot( km1, fun="cumhaz", col=1:2 )  
legend( "bottomright", c("0","1"),  
       inset=.01,  
       title="Treatment",  
       lty=1, col=1:2)
```

Hvis vi ønsker konfidensgrænser benyttes ekstra argument `conf.int=T` (giver dog rodet plot)



Er de ens?

Hvis de ikke er ens, er de så **proportionale?**

30 / 99 (de starter jo begge i 0)



Cox *proportional hazards* model

En såkaldt **Cox model** specificerer alle hazard-funktioner til at være proportionale.

Her har vi bare to grupper, så vi **antager**

$$h_1(t) = k \times h_0(t)$$

Men **baseline hazard funktionen** $h_0(t)$ specificeres ikke nærmere, dvs. vi antager ikke noget om afhængigheden af tiden

Vi vil gerne estimere proportionalitetskonstanten k ,
hvis modellen altså holder....., dvs. hvis der er additivitet på log-skala: $\log(h_1(t)) = \log(k) + \log(h_0(t))$



Estimat for behandlingseffekt

under antagelse af proportionale intensiteter, bestemmes ved coxph-funktionen (COX Proportional Hazards):

```
cox1 <- coxph( Surv(time,status) ~ factor(treat), data=d )
summary( cox1 )
```

med beskåret output

	coef	exp(coef)	se(coef)	z	Pr(> z)
factor(treat)1	0.01774	1.01790	0.18066	0.098	0.922

	exp(coef)	exp(-coef)	lower .95	upper .95
factor(treat)1	1.018	0.9824	0.7144	1.45

Her er coef differensen (treat=1 vs treat=0) mellem grupperne på log-hazard skala, og den skal tilbagetransformeres vha exponential funktionen til en faktor, kaldet **hazard ratio** eller *rate ratio*. Dvs $\text{hazard ratio} = \exp(\text{coef})$ med tilhørende konfidensgrænser lower .95 til upper .95.



Fortolkning af behandlingseffekten

Effekten af eksperimentel behandling ($\text{treat}=1$ vs. $\text{treat}=0$) er at øge hazard med faktoren 1.018, altså en forøgelse på 1.8%.

Dette er dog på ingen måde signifikant ($P=0.92$), idet konfidensintervallet er $CI=(0.71, 1.45)$, og altså indeholder effekter lige fra en forøgelse af hazard med 45% til en formindskelse af hazard med 29%.

Dette er den simplest tænkelige Cox-model med kun en enkelt kategorisk kovariat på 2 niveauer (2 grupper), *og raterne antages altså at være proportionale.*



*Proportionale rater/hazards

I tilfælde af to grupper har vi $h_1(t) = k \times h_0(t)$
og dermed parallellitet på log-skala, også for de *kumulerede* rater:

$$\log(H_1(t)) = \log(k) + \log(H_0(t))$$

Da der gælder $H(t) = -\log(S(t))$, svarer det til, at

$$\log(-\log(S_1(t))) = \log(k) + \log(-\log(S_0(t)))$$

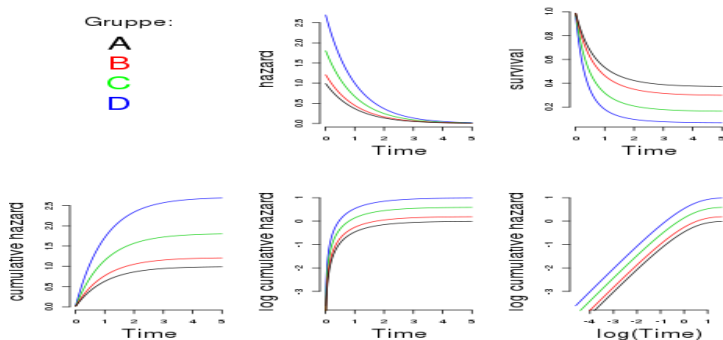
og man siger derfor, at der er parallellitet på loglogs-skala,
eller på *cloglog*-skala. **Dette kan kontrolleres**

- ▶ Grafisk: se s. 36 og 39
- ▶ Numerisk: se s. 39-41



Eksempel på proportionale rater/hazards

i 4 grupper (4 værdier af en kategorisk kovariat):

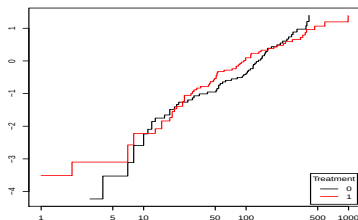


Figuren af $\log H(t)$ mod $\log(t)$ nederst til højre viser tydeligt parallelle kurver, dvs. **proportional hazards (PH)**

Har vi PH i vores eksempel?

Vi plotter $\log(H(t))$ ved at plotte vores survfit-objektet km1 med ekstra argument `fun="cloglog"`:

```
plot( km1, fun="cloglog", col=1:2)  
legend( "bottomright", c("0","1"), inset=.01, title="Treatment",  
       lty=1, col=1:2)
```



Er disse to parallelle?
Ikke synderligt....

Check af proportionale rater

Grafisk: Her er to muligheder:

1. Check parallellitet af $H(t)$ 'erne i grupperne, på loglogs-skala, som vist s. 36
2. Benyt de såkaldte *kumulerede score-residualer* via `cox.aalen`-funktionen i `timereg`-pakken, se s. 39

Numerisk: Her er også 2 muligheder:

1. Den ovenfor omtalte `cox.aalen` kan benyttes til et numerisk test, baseret på simulationer, se s. 40
2. **En sværere metode:** Inkluder en tidsafhængig effekt af kovariaten, se mere s. 73ff



*Score residualer (Schoenfeld)

Til hvert tidspunkt for event, f.eks. t_i for individ i ser vi på vedkommendes kovariat x_i og spørger:

Er det "den forventede" person, der dør?

$$x_i - \bar{x}(t_i, \hat{\beta})$$

Disse kumuleres over tid og standardiseres (*standardized score process*) og kan afsløre, om betydningen af x er den samme over tid (svarende til proportional hazards):

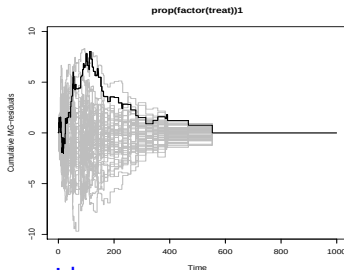
Hvis f.eks. folk med høje værdier af kovariaten dør "*for tidligt*" i starten, vil processen starte med at være positiv, for derefter at synke ned mod 0.



Grafisk check af proportionale rater

Benyt `cox.aalen` i `timereg`-pakken. Der sættes `prop()` (`PROPortional`) omkring de forklarende variable. Sæt gerne et *seed* (valgfrit tal), da vi ellers får lavet præcis samme simulationer / p-værdier hver gang analysen køres (forudsat at vi kører begge linier hver gang).

```
library( timereg )  
set.seed( 123 ) # valgfrit tal  
cal <- cox.aalen( Surv(time,status) ~ prop( factor(treat) ), data=d)  
plot( cal, score=T )
```



Bemærkninger næste side

Numerisk check af proportionale rater

P-værdi for test af proportionale rater får vi vha `summary()`:

```
> 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 lower2.5%
prop(factor(treat))1 0.015 0.182      0.176      0.181 0.0851 0.932      -0.342
              upper97.5%
prop(factor(treat))1      0.372
Test of Proportionality
```

```

              sup|  hat U(t) | p-value H_0
prop(factor(treat))1      8.03      0.016  <-- test for prop rater
```

Baseret på simulationer ser vores proces ud til at være ret aparte ($P=0.016$), så antagelsen om proportional hazards er ikke helt god.

Bemærkninger til figuren på næste side



Numerisk check af proportionale rater

Figuren s. 39 viser, at det i starten er patienter med høje værdier af variablen `treat1` (dvs. personer med 1-taller for indikatoren for `treat=1`, altså de eksperimentelt behandlede), der dør for tidligt.

Dette svarer også til, hvad vi har set på overlevelseskurven (s. 20) og kumuleret hazard funktion (s. 36).



Antagelsen om proportionale intensiteter er ikke opfyldt

Det betyder at:

- ▶ Forskellen på behandlingerne er *ikke* er konstant over tid.
- ▶ Hazard-funktionerne har ikke samme form for de to grupper

Hvad gør vi så?

- ▶ Overvejer, om vi har glemt nogle vigtige kovariater:
Fik vi svaret på det rigtige?
- ▶ Benytter tidsafhængige kovariater
Dette er svært, så vi udskyder det til s. 73ff



Har vi andre kovariater?

Jada, vi har adskillige:

celltype: 1:squamous, 2:smallcell, 3:adeno, 4:other

karno: Karnofsky performance score (0-100, 100 er godt):

- ▶ 10-30: hospitaliseret
- ▶ 40-60: delvist hospitaliseret
- ▶ 70-90: tager vare på sig selv

diagtime: Måneder fra diagnose til randomisering/behandling

age: Personens alder i år

prior: Er der givet tidligere behandling? 0:nej, 1:ja

Vi forsøger os med en model, der (ud over `treat`) tager højde for `celltype` og `karno`, dvs. en model med to kategoriske og en enkelt kvantitativ kovariat.



Regressionsmodeller i overlevelsesanalyse

i daglig tale: **Cox modeller:**

Da hazard-funktioner er positive, $h(t) > 0$, vil $\log(h(t))$ kunne antage alle værdier, og det er derfor naturligt at modellere effekten af kvantitative kovariater (x , f.eks. karno) på denne skala:

$$\log(h(t|x)) = \log(h_0(t)) + \beta x$$

således at effekten bliver *multiplikativ* på selve hazard-skalaen.

Her er $h_0(t)$ **baseline hazard**, svarende til $x = 0$

(altså en generaliseret form for intercept),

og β er regressionskoefficienten hørende til kovariaten x .

Bemærk, at β er uafhængig af tiden,

og at $\exp(\beta)$ netop er proportionalitetskonstanten k fra s. 31



Hazard-funktioner

For hver kombination af kovariater har vi til hvert tidspunkt en vis umiddelbar rate for død, som har det multiplikative udtryk:

$$h(t|x) = h_0(t) \cdot \exp(\beta x)$$

Hazard funktionen (raten) er altså en funktion af to (eller flere) variable: tiden og kovariaten (kovariaterne):

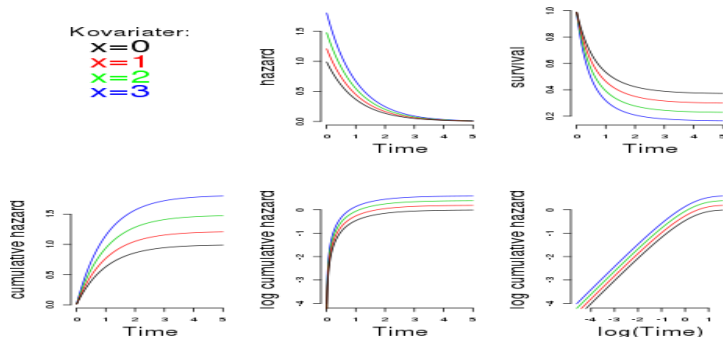
Proportional hazards: er en antagelse om afhængigheden af tiden, nemlig at denne afhængighed er den samme for alle kovariat-værdier, pånær en multiplikativ effekt (en faktor)

Linearitet: er en antagelse om afhængigheden af hver af de kvantitative kovariater, nemlig at der for hver givet tidspunkt er linearitet (på log-skala)



Eksempel på hazard-funktioner

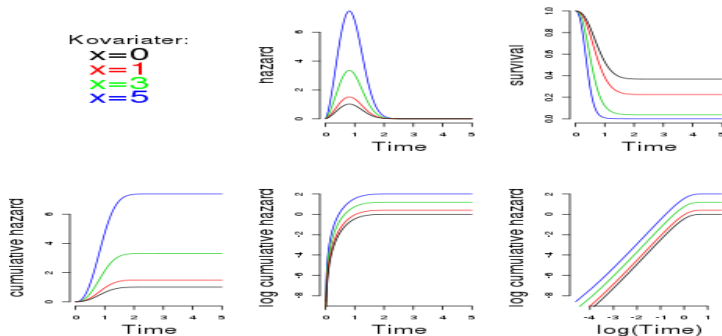
for 4 forskellige ækvidistante værdier (0, 1, 2 og 3) af en kvantitativ kovariat x :



Her er både **proportional hazards** og **linearitet** (det første ses ved parallellitet, og det sidste fordi afstanden mellem kurverne er ækvidistant)

Eksempel på hazard-funktioner

for 4 forskellige værdier af en kvantitativ kovariat x :

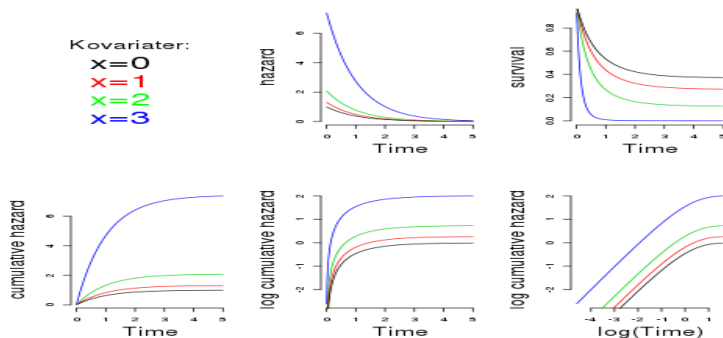


Her er både **proportional hazards** og **linearitet** (det første ses ved parallellitet, og det sidste fordi afstanden mellem kurverne afspejler de ikke-ækvivalente kovariat-værdier)



Eksempel på hazard-funktioner

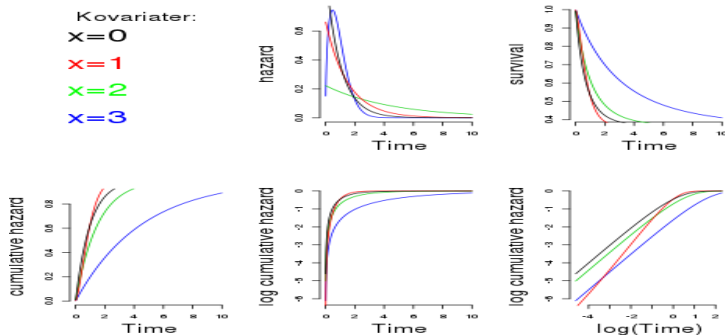
for 4 forskellige ækvidistante værdier (0, 1, 2 og 3) af en kvantitativ kovariat x :



Her er **proportional hazards**, men *ikke* linearitet (kurverne er parallelle, men *ikke* ækvidistante)

Eksempel på hazard-funktioner

for 4 forskellige ækvidistante værdier (0, 1, 2 og 3) af en kvantitativ kovariat x :



Her er *hverken* proportional hazards, *eller* linearitet

Model med 3 kovariater

- ▶ treat: Behandlingen
- ▶ celltype: De 4 cancertyper
- ▶ karno: kvantitativ Karnofsky score

celltype er en tekstvariabel og vi behøver derfor ikke at sætte `factor()` omkring denne i modelformlen:

```
cox2 <- coxph( Surv(time,status) ~ factor(treat) + celltype + karno,  
              data=d )
```



Output fra model med 3 kovariater

```
> summary( cox2 )
Call:
coxph(formula = Surv(time, status) ~ factor(treat) + celltype +
      karno, data = d)

n= 137, number of events= 128
```

	coef	exp(coef)	se(coef)	z	Pr(> z)
factor(treat)1	0.261744	1.299194	0.200923	1.303	0.193
celltypelarge	-0.759369	0.467962	0.298657	-2.543	0.011 *
celltypesmallcell	-0.329014	0.719633	0.269058	-1.223	0.221
celltypesquamous	-1.153994	0.315375	0.295038	-3.911	9.18e-05 ***
karno	-0.031271	0.969213	0.005165	-6.054	1.41e-09 ***

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

	exp(coef)	exp(-coef)	lower .95	upper .95
factor(treat)1	1.2992	0.7697	0.8763	1.9262
celltypelarge	0.4680	2.1369	0.2606	0.8403
celltypesmallcell	0.7196	1.3896	0.4247	1.2194
celltypesquamous	0.3154	3.1708	0.1769	0.5623
karno	0.9692	1.0318	0.9595	0.9791

Concordance= 0.737 (se = 0.03)
Rsquare= 0.36 (max possible= 0.999)
Likelihood ratio test= 61.07 on 5 df, p=7.307e-12
Wald test = 63.41 on 5 df, p=2.397e-12
Score (logrank) test = 66.55 on 5 df, p=5.347e-13



Output fra model med 3 kovariater

Overall test for association (type III) test får vi med `drop1()`:

```
> drop1( cox2, test="Chisq")  
Single term deletions
```

Model:

```
Surv(time, status) ~ factor(treat) + celltype + karno
```

	Df	AIC	LRT	Pr(>Chi)
<none>		959.83		
factor(treat)	1	959.53	1.697	0.192626
celltype	3	971.93	18.102	0.000419 ***
karno	1	993.04	35.212	2.957e-09 ***

```
Signif. codes:  0 "****" 0.001 "***" 0.01 "*" 0.05 "." 0.1 " " 1
```

```
>
```



Output fra model med 3 kovariater, fortsat

Tilbagetransformerede estimer (hazard ratio) har vi i den nederste del af output s. 51 under `exp(coef)` med tilhørende konfidensgrænser.

Se fortolkning s. 56-57



Output fra model med 3 kovariater, fortsat

For karno er $HR=0.969$, 95% CI 0.960-0.979, pr 1 enhed.

Hvis vi ønsker at rapportere effekten af karno svarende til 10 enheder, skal vi gange parameteren (coef) med 10 inden vi tager exponentialfunktion. Samme manøvre skal laves på CI:

```
# Parameter nr 5 beskriver effekten af karno:
coef( cox2 )
# Denne ganges med 10
10*coef( cox2 )[5]
# HR er antilog (exp()) til parameteren:
exp( 10* coef( cox2 )[5] )

# Konfidensinterval for alle parametrene:
confint( cox2 )
# For linie 5 (karno) ganges med 10 inden vi tager exp():
exp( 10* confint( cox2 )[5,] )
```

Man finder da at effekten af karno svarende til 10 enheder er
54 / 99 $HR=0.731$, 95% CI 0.661-0.809. Se fortolkning s. 56-57



Output fra model med 3 kovariater, fortsat

Hvis vi ønsker en anden referencegruppe for celltype kan vi ændre den med `relevel()`:

```
> d$ctS <- relevel( factor(d$celltype), "squamous")  
> cox3 <- coxph( Surv(time,status) ~ factor(treat) + ctS + karno,  
+               data=d )  
summary( cox3 )
```

hvorved vi (beskåret) får HR med squamous som reference:

	exp(coef)	exp(-coef)	lower .95	upper .95
factor(treat)1	1.2992	0.7697	0.8763	1.9262
ctSadeno	3.1708	0.3154	1.7784	5.6534
ctSlarge	1.4838	0.6739	0.8534	2.5801
ctSsmallcell	2.2818	0.4382	1.3471	3.8653
karno	0.9692	1.0318	0.9595	0.9791



Fortolkning af output fra de forrige sider

- ▶ Der er stærkt signifikant negativ effekt af karno-score ($P < 0.0001$)
- ▶ Der er stærkt signifikant forskel på cancertyperne, med adeno og smallcell som de farligste (de har de største positive afvigelser fra referencegruppen squamous)
- ▶ Behandlingseffekten er fortsat ikke signifikant, $P=0.19$

For at forstå selve estimerne, skal vi se på rate ratioerne (hazard ratios), som det ses s. 51 og 55.



Fortolkning af rate ratioer

- ▶ Effekten af 1 enhed på karno-skalaen er et fald i dødsraten på 3.1%, $CI=(2.1\%, 4.1\%)$. Effekten af en øgning på 10 enheder på karno-skalaen er et fald i dødsraten på 26.9%, $CI=(19.1\%, 33.9\%)$, se p. 54
- ▶ Umiddelbart er den eksperimentelle behandling dårlig, med en forøget rate på 30.0%, men CI viser, at det ligeså godt kunne være en formindsket rate (12.4% lavere), eller muligvis helt op til næsten dobbelt så stor rate (92.6% højere)
- ▶ adeno-typen estimeres til at være ca. 3 gange så "farlig" som referencen squamous, dog med betydelig usikkerhed, svarende til faktorer mellem 1.78 og 5.65.



Modelcheck

For kategoriske kovariater:

Her drejer det sig om `treat` og `celltype`:

Her skal antagelsen om *Proportional hazards* checkes:

- ▶ ved at sætte kovariaten i `strata` i stedet for i `model`-sætning, for derefter at tegne $\log H(t)$, se det tidligere check af `treat`, s. 36, og det nuværende på s. 59

For kovariaten `celltype` finder vi her figuren på s. 60

- ▶ benytte `cox.aalen` som vi gjorde s. 39, se hvordan for den multivariable model s. 61 og 62

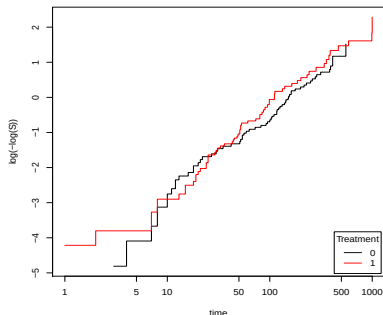


Modelkontroltegninger for treat

i denne nye model med 3 kovariater.

Da dette er en kategorisk variabel, sætter vi den i stedet som strata-variabel for at kunne tegne kumulerede rater på log-skala:

```
cox4 <- coxph( Surv(time,status) ~ strata(treat) + ctS + karno, data=d )  
  
newD=data.frame( ctS="squamous",karno=50) # vælg vilkaarlig vaerdi for ctS og karno  
survfit4=survfit( cox4, newdata=newD )  
plot( survfit4, fun="cloglog", ylab="log(-log(S))", col=1:2, mark.time=F, xlab="time")  
legend("bottomright",c("0","1"),lty=1,col=1:2,title="Treatment",inset=.01)
```



Modelkontroltegninger for celltype

Til brug for denne figur, er det celletypen ctS, der sættes som

strata-variabel:

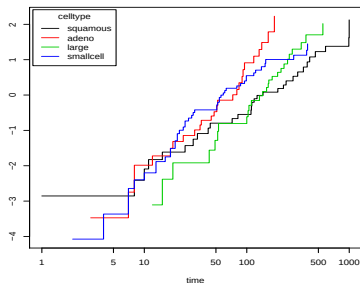
```
cox5 <- coxph( Surv(time,status) ~ treat + strata(ctS) + karno, data=d )
```

```
newD=data.frame( treat=0,karno=50)
```

```
survfit5=survfit( cox5, newdata=newD )
```

```
plot( survfit5, fun="cloglog", col=1:4, mark.time=F,xlab="time")
```

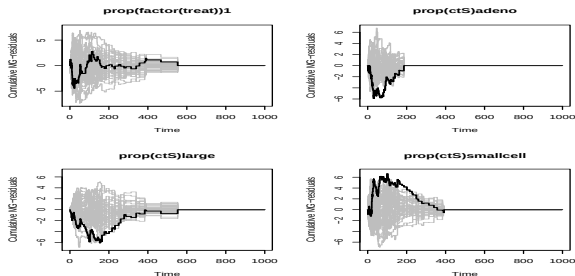
```
legend("topleft",levels(d$ctS),col=1:4,lty=1,title="celltype",inset=.01)
```



Det ses, at det netop er referencegruppen squamous, der opfører sig anderledes end de andre, med lavere hældning

Proportional hazards via cox.aalen, I

```
set.seed( 234 )  
ca2 <- cox.aalen( Surv(time,status) ~ prop(factor(treat))  
                + prop(ctS) + prop(karno), data=d)  
  
plot( ca2, score=T, specific.comps = 1:4 )  
# Med specific.comps vælger vi kun at plotte for de kvalitative  
# (=4 foerste parametre), dvs karno er ikke med her
```



Proportional hazards via cox.aalen, II

Numeriske tests af modeltilpasning:

```
> summary(ca2)
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(factor(treat))1	0.2580	0.219	0.17000	0.20100	1.52	1.30e-01
prop(ctS)adeno	1.1500	0.314	0.28400	0.29500	4.04	5.31e-05
prop(ctS)large	0.3900	0.292	0.24700	0.28200	1.57	1.15e-01
prop(ctS)smallcell	0.8230	0.291	0.30400	0.26900	2.70	6.85e-03
prop(karno)	-0.0312	0.005	0.00536	0.00517	-5.82	6.06e-09

lower2.5% upper97.5%

prop(factor(treat))1	-0.171	0.6870
prop(ctS)adeno	0.535	1.7700
prop(ctS)large	-0.182	0.9620
prop(ctS)smallcell	0.253	1.3900
prop(karno)	-0.041	-0.0214

Test of Proportionality

	supl	hat U(t)	p-value H ₀	<-- p-vaerdier for prop hazards her
prop(factor(treat))1		4.45	0.360	
prop(ctS)adeno		5.92	0.020	
prop(ctS)large		6.08	0.026	
prop(ctS)smallcell		6.63	0.066	
prop(karno)		390.00	0.002	



Bemærkning vedrørende Proportional hazards

Først for kovariaten `treat`:

Det numeriske test giver nu $P=0.36$, dvs.

vi har ikke mere anledning til at betvivle proportionaliteten

Bemærk, at vi tidligere (s. 39-40) havde $P=0.016$, og de proportionale intensiteter er altså opnået ved at tilføje vigtige ekstra kovariater



Bemærkning vedrørende Proportional hazards, II

Og så for kovariaten `celltype`:

Det numeriske test er her 3-delt, idet alle typer sammenlignes til reference-typen `squamous`. Det ses, at P-værdien hørende til `adeno` og `large` er signifikante, `smallcell` lige præcis ikke.

Her har vi altså et problem, og fra figuren s. 60 ses det, at netop celletypen `squamous` adskiller sig fra de øvrige ved at have et fladere forløb, svarende til en større dødelighed fra start, efterfulgt af lavere dødelighed senere.

Havde vi valgt en anden referencegruppe, havde vi rimeligvis kun set problemer i forhold til sammenligningen med `squamous`.



Modelcheck for kvantitative kovariater

Her drejer det sig kun om karno:

Her skal der checkes to antagelser:

1. *Linearitet* på log(hazard)-skala Her kan man benytte **Kumulerede Martingal residualer**:

For hver kovariatværdi x ser man på, i hvor høj grad personer med kovariatværdier $< x$ “levede for længe” (positive værdier) eller “døde for tidligt” (negative værdier).

Fås i R ved at benytte `cox.aalen` med lidt andre argumenter, se næste side. **Metoden er dog ikke særligt sensitiv**

2. *Proportional hazards*: checkes som for kategoriske kovariater, nemlig vha `cox.aalen` som beskrevet p 61.



Check af modelforudsætninger for karno

Test for linearitet ved kumulerede martingal residualer udføres ved at benytte `cox.aalen` med ekstra argumenter `residuals` og `n.sim` og derefter regne videre på residualerne:

```
set.seed( 345 )
ca3 <- cox.aalen( Surv(time,status) ~ prop(factor(treat)) + prop(ctS) + prop(karno),
                 data=d, residuals=1, n.sim=0 )
resids <- cum.residuals( ca3, cum.resid=1, data=d )
summary(resids)
```

med output

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

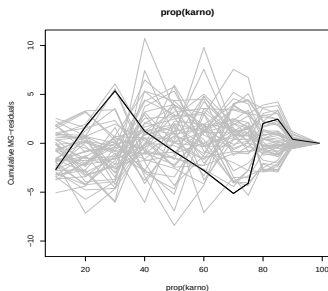
	sup	hat B(t)	p-value H ₀ : B(t)=0
prop(karno)		5.374	0.344



Check af modelforudsætninger for karno

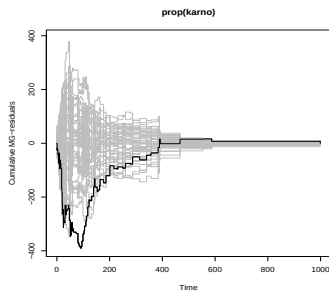
Lineariteten:

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



Proportional hazards:

```
plot( ca2, score=T, specific.comps=5 )
```



Linearitetsantagelsen for karno

- ▶ Figuren til venstre s. 67 og det tilhørende test s. 66 viser, at lineariteten er rimelig ($P=0.34$).
- ▶ Alternative checks af lineariteten kunne være:
 - ▶ Såvel karno som $\log(\text{karno})$ som kovariater
 - ▶ En lineær spline i karno, hvor vi f.eks. tillader knæk ved karno=40 og karno=80:

```
d$karno40 <- (d$karno>40)*(d$karno-40)
d$karno80 <- (d$karno>80)*(d$karno-80)
cox6 <- coxph( Surv(time,status) ~ factor(treat) + ctS + karno
              + karno40 + karno80, data=d )
```

Test af linearitet (begge spline-variable lig 0):

```
library(car)
linearHypothesis( cox6, c("karno40=0","karno80=0"))
```

Se næste side for output og illustration af effekten.



Effekt af karno som lineær spline

Test af *ingen knæk i spline'en*: (kode til grafik i App s. s. 95-97.)

```
> linearHypothesis( cox6, c("karno40=0","karno80=0"))
```

Linear hypothesis test

Hypothesis:

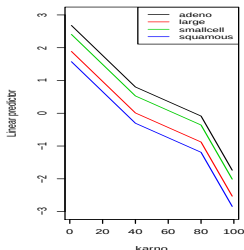
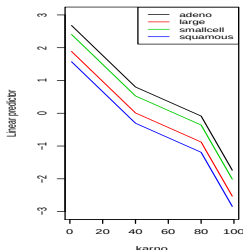
karno40 = 0

karno80 = 0

Model 1: restricted model

Model 2: `Surv(time, status) ~ factor(treat) + celltype + karno + karno40 + karno80`

	Res.Df	Df	Chisq	Pr(>Chisq)
1	132			
2	130	2	2.6318	0.2682



Check af proportional hazards for karno

Testet for proportionale intensiteter for karno (s. 62) viser, at denne antagelse er helt gal ($P = 0.002$).

Denne manglende proportionalitet ses også på figuren til højre på s. 67, hvor den standardiserede score-process (se s. 38) ses at afvige fra de simulerede.

Dette betyder:

- ▶ Effekten af karno er *ikke* er konstant over tid.
- ▶ Hazard-funktionerne har ikke samme form for alle værdier af karno

Hvad gør vi så??



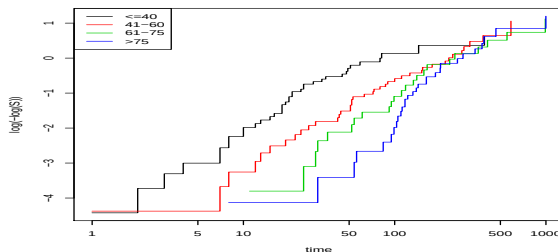
Vi prøver at forstå problemet

- ▶ På figuren s. 67 ses, at de med små værdier af karno i højere grad dør i starten af perioden i forhold til, hvad modellen siger
- ▶ Effekten af karno er i sig selv negativ, således at små værdier af karno altid vil formodes at dø først - de gør det bare endnu mere i starten
- ▶ Det betyder, at effekten af karno er størst i starten, men aftager efterhånden som tiden går



Hvis vi nu grupperer karno-kovariaten

og benytter den som strata-variabel (se kode s. 91):



Her ser vi ligeledes, at kurverne afviger en del i starten (svarende til stor effekt af karno), men efterhånden nærmer sig hinanden (svarende til *næsten ingen* effekt)

Tidsafhængige kovariater

Vi har altså set, at antagelsen om proportionale intensiteter **hazards** ikke holdt for kovariaten karno.

Det betyder, at den prognostiske værdi af karno ændrer sig med tiden, og vi fortolkede det som at effekten af karno er stærkest i starten.

Hvad gør vi så?

Vi kan opdele tiden i intervaller, og estimere en proportionalitetsfaktor for hvert interval for sig.

Vi kan f.eks. opdele tiden efter **kvartilerne for de døde**:
23, 62 og 147.5 dage



*Tidsafhængige kovariater i praksis

- ▶ For at få effekten af 10 karno-enheder direkte, skifter vi karno-variablen ud med $\text{karno10} = \text{karno}/10$, således at 1 enhed af karno10 svarer til 10 enheder af karno.
- ▶ Dernæst benytter vi *time-transform* funktionen `tt` til at specificere separate effekter af karno10 for hvert interval:

```
d$karno10 <- d$karno/10
cox8 <- coxph( Surv(time,status) ~ factor(treat) + ctS + tt(karno10),
              data=d,
              tt = function(k,t,...) {
                  cbind( k*(t<=23),
                        k*(t>23 & t<=62),
                        k*(t>62 & t<=147.5),
                        k*(t>147.5)) }
              )
```

Her er `tt` en funktion af `k` (karno) og `t` (time).



Output fra tidsafhængig effekt af karno (beskåret)

```
> summary(cox8)
n= 137, number of events= 128
```

	coef	exp(coef)	se(coef)	z	Pr(> z)	
factor(treat)1	0.08422	1.08787	0.20516	0.411	0.681429	
ctSadeno	1.09673	2.99435	0.30059	3.649	0.000264	***
ctSlarge	0.33496	1.39788	0.28392	1.180	0.238101	
ctSsmallcell	0.88668	2.42706	0.27687	3.202	0.001363	**
tt(karno10)1	-0.51792	0.59576	0.09281	-5.580	2.4e-08	***
tt(karno10)2	-0.36564	0.69376	0.09111	-4.013	6.0e-05	***
tt(karno10)3	-0.07695	0.92593	0.12182	-0.632	0.527563	
tt(karno10)4	-0.03099	0.96948	0.12836	-0.241	0.809199	

	exp(coef)	exp(-coef)	lower .95	upper .95
factor(treat)1	1.0879	0.9192	0.7277	1.6263
ctSadeno	2.9943	0.3340	1.6613	5.3971
ctSlarge	1.3979	0.7154	0.8013	2.4386
ctSsmallcell	2.4271	0.4120	1.4106	4.1760
tt(karno10)1	0.5958	1.6785	0.4967	0.7146
tt(karno10)2	0.6938	1.4414	0.5803	0.8294
tt(karno10)3	0.9259	1.0800	0.7293	1.1756
tt(karno10)4	0.9695	1.0315	0.7538	1.2468



Test af tidsafhængig effekt af karno

Vi kan teste om effekten af karno virkelig er forskellig i de 4 tidsperioder:

```
linearHypothesis( cox8, c("tt(karno10)1=tt(karno10)2",  
                          "tt(karno10)2=tt(karno10)3",  
                          "tt(karno10)3=tt(karno10)4") )
```

med output

Linear hypothesis test

Hypothesis:

$\text{tt(karno10)1} - \text{tt(karno10)2} = 0$

$\text{tt(karno10)2} - \text{tt(karno10)3} = 0$

$\text{tt(karno10)3} - \text{tt(karno10)4} = 0$

Model 1: restricted model

Model 2: $\text{Surv(time, status)} \sim \text{factor(treat)} + \text{ctS} + \text{tt(karno10)}$

	Res.Df	Df	Chisq	Pr(>Chisq)
1	132			
2	129	3	13.08	0.004467 **



Fortolkning af tidsafhængig effekt

Effekten af karno aftager med tiden:

Rate-ratioerne nærmer sig 1, ganske som vi fandt det tidligere på kontroltegningen s. 72.

Det kunne have den simple forklaring, at karno var målt ved indgang i studiet.

Vi får et alternativt test for konstant effekt af karno, med $P = 0.0045$, altså (igen) forkastelse.



Kan der være interaktioner med celltype?

Man kunne forestille sig en differentieret effekt af behandlingen for de 4 celletyper, altså interaktionen `treat*celltype`.

Interaktionen tilføjes modellen

```
cox9 <- coxph( Surv(time,status) ~ factor(treat) + ctS + factor(treat)*ctS
              + tt(karno10), data=d,
              tt = function(k,t,...) cbind( k*(t<=23), k*(t>23 & t<=62),
```

Når vi skal teste interaktionen dur `drop1()` og `anova()` desværre ikke, idet de bliver forvirrede over `tt`-funktionen. Vi må derfor selv finde navnene på interaktionsparametrene (feks vha `summary()`) og derefter selv specificere hypotesen:

```
summary(cox9)
linearHypothesis( cox9, c("factor(treat)1:ctSadeno=0",
                          "factor(treat)1:ctSlarge=0",
                          "factor(treat)1:ctSsmallcell=0") )
```

Se beskåret output næste side.



Kan der være interaktioner med cellype?

	exp(coef)	exp(-coef)	lower .95	upper .95
factor(treat)1	0.5482	1.8241	0.2471	1.2164
ctSadeno	2.6911	0.3716	1.1000	6.5838
ctSlarge	0.8239	1.2137	0.3786	1.7932
ctSsmallcell	1.4582	0.6858	0.7502	2.8341
tt(karno10)1	0.5993	1.6686	0.4983	0.7208
tt(karno10)2	0.6908	1.4476	0.5752	0.8296
tt(karno10)3	0.9263	1.0796	0.7237	1.1855
tt(karno10)4	0.9820	1.0184	0.7548	1.2775
factor(treat)1:ctSadeno	1.4162	0.7061	0.4367	4.5929
factor(treat)1:ctSlarge	2.6866	0.3722	0.8790	8.2119
factor(treat)1:ctSsmallcell	3.1940	0.3131	1.1571	8.8168

Linear hypothesis test

Hypothesis:

factor(treat)1:ctSadeno = 0

factor(treat)1:ctSlarge = 0

factor(treat)1:ctSsmallcell = 0

Model 1: restricted model

Model 2: `Surv(time, status) ~ factor(treat) + ctS + factor(treat) * ctS +
tt(karno10)`

	Res.Df	Df	Chisq	Pr(>Chisq)
1	129			
2	126	3	6.3038	0.09773

Der er ikke tilstrækkelig evidens for en interaktion (P=0.10)



Interaktioner med celltype, fortsat

Estimator for behandlingen, opdelt efter celletype:

```
cox10 <- coxph( Surv(time,status) ~ ctS + factor(treat):ctS + tt(karno10),
  data=d, tt = function(k,t,...) cbind( k*(t<=23), k*(t>23 & t<=62),
summary(cox10)
```

med beskåret output:

	exp(coef)	exp(-coef)	lower .95	upper .95
ctSadeno	2.6911	0.3716	1.1000	6.5838
ctSlarge	0.8239	1.2137	0.3786	1.7932
ctSsmallcell	1.4582	0.6858	0.7502	2.8341
tt(karno10)1	0.5993	1.6686	0.4983	0.7208
tt(karno10)2	0.6908	1.4476	0.5752	0.8296
tt(karno10)3	0.9263	1.0796	0.7237	1.1855
tt(karno10)4	0.9820	1.0184	0.7548	1.2775
ctSsquamous:factor(treat)1	0.5482	1.8241	0.2471	1.2164
ctSadeno:factor(treat)1	0.7764	1.2880	0.3342	1.8035
ctSlarge:factor(treat)1	1.4729	0.6789	0.6723	3.2268
ctSsmallcell:factor(treat)1	1.7510	0.5711	0.9194	3.3350



Problemstillinger, I: Forsinket indgang

Forsinket indgang = left-truncation

Når vi skal sammenligne behandlinger, kan patienterne naturligvis først tælle med, når de blev randomiseret til en af disse og derfor er en del af *risikogruppen*.

Hvis vi benytter en tidsskala, der hedder *tid siden diagnose*, kan vi have et problem, idet nogle kan være døde efter diagnose og inden randomisering, således at de, der indgår i studiet er selekterede, måske forskelligt i de to behandlingsgrupper. *Hvad gør vi så?*

Vi konstruerer en variabel (f.eks. indgang), der indeholder det tidspunkt, som patienten skal tælle med fra, og så skriver vi f.eks.

```
Surv(indgang,time,status)~ factor(treat)+celltype
```



Problemstillinger, II: Competing risks

Competing risks = konkurrerende dødsårsager/afgangsårsager:

Hvis man f.eks. studerer tid til tilbagefald efter behandling, har man det problem, at folk stadig kan dø, *inden* de får tilbagefald. **Hvad gør man så?**

1. Man analyserer det sammensatte endepunkt: tid til *enten* tilbagefald *eller* censurering, hvad der så end kommer først
2. Man opfatter død som censureringer

I første tilfælde er der intet problem, men i tilfælde 2 får man problemer:

Man kan *stadig* godt

- Udregne hazard-funktioner, og estimere i Cox-modeller for tilbagefald=recidiv, ved blot at opfatte død som en censurering



Competing risks, fortsat

men man kan *ikke*

- ▶ fortolke overlevelseskurver (Kaplan-Meier), se s. 20
fordi det ikke giver nogen mening at antage, at de censurerede patienter (som jo er døde) ville få recidiv i samme takt som de uensurerede, se s. 15.
- ▶ beregne kumulerede incidenser, altså sandsynligheder for at opleve recidiv (som vist s. 21)

I stedet kan man benytte de såkaldte **Fine and Gray modeller**, der direkte modellerer kumulative incidenser



Ofte stillede spørgsmål: FAQ

- ▶ *Hvorfor bruger vi ikke bare logistisk regression af f.eks. 1-års mortaliteten?*

Fordi vi mister en masse information, og fordi vi mister muligheden for at håndtere censureringer

- ▶ *Hvorfor bruger vi ikke bare normalfordelingsmodeller på tiden fra randomisering til død?*

Fordi fordelingen ofte er ganske anderledes end en normalfordeling, og igen fordi vi mister muligheden for at håndtere censureringer

Overlevelsesanalyse er den stærkeste måde at håndtere overlevelsesdata på!



APPENDIX

med R-programbidder svarende til nogle af slides

- ▶ Kaplan-Meier kurver, s. 86
- ▶ Kumulerede incidenser, s. 87
- ▶ Log-rank test, s. 88
- ▶ Kumulerede hazards, s. 89
- ▶ Check af proportional hazards, s. 91-92
- ▶ Check af linearitet, s. 93, 95-97
- ▶ Model med 3 kovariater og modelkontrol, s. 94-95
- ▶ Tidsafhængige kovariater, s. 98
- ▶ Interaktioner, s. 99

Vi bruger pakkerne `survival`, `prodlm` og `timereg` til at lave overlevelsesanalyser. Pakken `car` benyttes til at teste lineære hypoteser. NB: Sammenligner du med resultaterne i SAS-slides vil du se, at der for Cox-regression er små forskelle i tallene. Det skyldes at R bruger en anden default indstilling end SAS i det tilfælde hvor der er *ties*, dvs hvor nogle patienter dør samme tid.



Kaplan-Meier kurver i R

Slide 20

Figur af 2 Kaplan-Meier overlevelseskurver,
svarende til 2 værdier af variablen treat. Uden konfidensgrænser:

```
library(survival)
km1 <- survfit( Surv(time,status) ~ treat, data=d )
plot( km1 )
```

Med konfidensgrænser:

```
library(prodlim)
km2 <- prodlim( Hist(time,status) ~ treat, data=d )
plot( km2 )
```



Kumulerede incidenser, med figur

Slide 21

Kumuleret incidens kan vi kun få frem af prodlim:

```
km2 <- prodlim( Hist(time,status) ~ treat, data=d )  
plot(km2, type="cuminc")
```



Log-Rank test

Slide 23-24

```
survdif( Surv(time,status) ~ treat, data=d )
```



Figur af kumulerede hazards

Slide 30

Kumulerede hazards får vi frem ved brug af `survfit` og derefter ved brug af `plot()` med ekstra argument `fun="cumhaz"`:

```
km1 <- survfit( Surv(time,status) ~ treat, data=d )  
plot( km1, fun="cumhaz", col=1:2 )  
legend( "bottomright", c("0","1"), inset=.01,  
        title="Treatment", lty=1, col=1:2)
```



Check af Proportional hazards, figur

Slide 36

Log kumuleret rate ($\log(-\log(S(t)))$) får vi frem ved brug af `survfit` og derefter ved brug af `plot()` med ekstra argument `fun="cloglog"`:

```
km1 <- survfit( Surv(time,status) ~ treat, data=d )  
plot( km1, fun="cloglog", col=1:2)  
legend( "bottomright", c("0","1"), inset=.01, title="Treatment",  
       lty=1, col=1:2)
```



Check af Proportional hazards, figur kvantitativ kovariat

Slide 72

Log kumuleret rate ($\log(-\log(S(t)))$) får vi igen frem ved brug af `survfit` og derefter ved brug af `plot()` med ekstra argument `fun="cloglog"`. Forinden skærer vi den kvantitative variabel i bidder med `cut()`:

```
d$karnogrp <- cut( d$karno, breaks=c(0,40,60,75,100),
                  labels=c('<=40','41-60','61-75','>75') )
cox7 <- coxph( Surv(time,status) ~ factor(treat) + ctS + strata(karnogrp),
              data=d )

newD=data.frame( treat=0, ctS='squamous')
survfit7=survfit( cox7, newdata=newD )
plot( survfit7, fun="cloglog", ylab="log(-log(S))",
      col=1:4, mark.time=F,xlab="time")
legend( "topleft", levels(d$karnogrp), lty=1, col=1:4 )
```



Numerisk og grafisk check af proportionale rater

Slide 39-40

Hertil benyttes `cox.aalen` i `timereg`-pakken. Husk at sætte `prop()` omkring alle de forklarende variable i modelformlen:

```
library( timereg )  
set.seed(123)  
ca1 <- cox.aalen( Surv(time,status) ~ prop( factor(treat) ),  
                 data=d)  
summary( ca1 )           # Numerisk - aflæs p-vaerdi  
plot( ca1, score=T )     # Grafisk
```



Numerisk og grafisk check af linearitet

Slide 66-68

Igen benyttes `cox.aalen` med `prop()` om alle forklarende variable (her i den multivariable model). Der tilføjes ekstra argumenter og regnes videre på residualerne / plottes:

```
ca2 <- cox.aalen( Surv(time,status) ~ prop(factor(treat))  
                + prop(ctS) + prop(karno),  
                data=d,  
                residuals=1, n.sim=0 )
```

```
resids <- cum.residuals( ca2, cum.resid=1, data=d )  
summary(resids)          # Numerisk - aflæs p-vaerdi  
plot(resids, score=2)    # Grafisk
```



Model med 3 kovariater

Slide 50

```
cox2 <- coxph( Surv(time,status) ~ factor(treat) + ctS + karno,  
              data=d )  
summary( cox2 )
```

Overall test for association:

```
drop1( cox2, test="Chisq")
```

Hvordan man ændrer referencegruppen ses på side 55.

Hvordan man rapporterer effekten af karno svarende til 10 enheder fremgår af side 54. Alternativt kan det gøres ved at benytte $\text{karno10} = \text{karno}/10$ i analysen isf karno.



Kovariat modelleret som spline

Slide 68-69

Definitionerne af de to ekstra kovariater er

```
d$karno40 <- (d$karno>40)*(d$karno-40)
```

```
d$karno80 <- (d$karno>80)*(d$karno-80)
```

```
cox6 <- coxph( Surv(time,status) ~ factor(treat) + ctS + karno +  
               karno40 + karno80, data=d )
```

Test af linearitet:

```
library(car)
```

```
linearHypothesis( cox6, c("karno40=0","karno80=0"))
```

Illustration af effekten. Her skal vi prædiktere den lineære prædiktor. Hvordan fremgår af s. 96-97.



Plot af splines I

```
# Bestemmer vaerdier for karno og laver et datasæt d.k med mulige karno+spline vaerdier
```

```
range(d$karno)
k <- 1:99
d.k <- data.frame(karno=k, karno40=(k>40)*(k-40), karno80=(k>80)*(k-80))
```

```
# Laver nye datasæt svarende til alle vaerdier af celletypen ctS og treat=0
new0.a <- d.k; new0.a$treat=0; new0.a$ctS="adeno"
new0.l <- d.k; new0.l$treat=0; new0.l$ctS="large"
new0.sm <- d.k; new0.sm$treat=0; new0.sm$ctS="smallcell"
new0.sq <- d.k; new0.sq$treat=0; new0.sq$ctS="squamous"
```

```
# Linear predictor for alle vaerdier af ctS og treat=0
p0.a <- predict( cox6, newdata=new0.a, type="lp")
p0.l <- predict( cox6, newdata=new0.l, type="lp")
p0.sm <- predict( cox6, newdata=new0.sm, type="lp")
p0.sq <- predict( cox6, newdata=new0.sq, type="lp")
```

```
# Laver nye datasæt svarende til alle vaerdier af ctS og treat=1
new1.a <- new0.a; new1.a$treat <- 1
new1.l <- new0.l; new1.l$treat <- 1
new1.sm <- new0.sm; new1.sm$treat <- 1
new1.sq <- new0.sq; new1.sq$treat <- 1
```

```
# Linear predictor for alle vaerdier af ctS og treat=1
p1.a <- predict( cox6, newdata=new0.a, type="lp")
p1.l <- predict( cox6, newdata=new0.l, type="lp")
p1.sm <- predict( cox6, newdata=new0.sm, type="lp")
p1.sq <- predict( cox6, newdata=new0.sq, type="lp")
```



Plot af splines II

Vi har nu 8 vektorer som indeholder estimerer for den lineære prædiktør. Dem plotter vi nu side om side for hhv `treat=0` og `treat=1`:

```
par(mfrow=c(1,2)) # plot med 1 række, 2 søjler

# Plot for treat=0
plot( k, p0.a, col=1, type="l", ylim=c(-3,3), xlab='karno', ylab='Linear predictor' )
lines( k, p0.l, col=2 )
lines( k, p0.sm, col=3 )
lines( k, p0.sq, col=4 )
legend( "topright", c("adeno","large","smallcell","squamous"), lty=1, col=1:4, cex=.8 )

# Plot for treat=1
plot( k, p1.a, col=1, type="l", ylim=c(-3,3), xlab='karno', ylab='Linear predictor' )
lines( k, p1.l, col=2 )
lines( k, p1.sm, col=3 )
lines( k, p1.sq, col=4 )
legend( "topright", c("adeno","large","smallcell","squamous"), lty=1, col=1:4, cex=.8 )
```



Tidsafhængige kovariater

Slide 73ff

```
cox8 <- coxph( Surv(time,status) ~ factor(treat) + ctS + tt(karno10),  
              data=d,  
              tt = function(x,t,...) {  
                cbind( x*(t<=23), x*(t>23 & t<=62), x*(t>62 & t<=147.5), x*(t>  
              })  
summary(cox8)
```

For kort forklaring af brugen af tt, se s. 74.

Test af proportionale effekter

```
library(car)  
linearHypothesis( cox8, c("tt(karno10)1=tt(karno10)2",  
                          "tt(karno10)2=tt(karno10)3",  
                          "tt(karno10)3=tt(karno10)4" ) )
```



Interaktion

i model med tidsafhængige kovariater.

Slide 78

```
cox9 <- coxph( Surv(time,status) ~ factor(treat) + ctS + factor(treat)*ctS
              + tt(karno10), data=d,
              tt = function(k,t,...) cbind( k*(t<=23), k*(t>23 & t<=62),
```

Desværre virker ikke `drop1()` eller `anova()` ikke når vi har tidsafhængige effekter. Test for interaktion må vi derfor selv lave ved at teste interaktionsleddene lig 0:

```
summary(cox9) # herved findes navnene paa interaktionsleddene,
              # som sættes ind i linearHypothesis nedenfor
```

```
linearHypothesis( cox9, c("factor(treat)1:ctSlarge=0",
                          "factor(treat)1:ctSsmallcell=0",
                          "factor(treat)1:ctSsquamous=0") )
```

