

Survival And Reliability Final

Kendall Brown r0773111

5/27/2020

```
rm(list=ls())
#####
# Loading the dataset
#####
leukem<-read.table(file="C:/Users/kebro/OneDrive/KU_Leuven/Survivial & Reliability/Leuk.dat",na.strings=
colClasses=c(NA,"character",NA,NA,NA,NA,NA,NA,NA,NA,NA,NA,NA,"character","character",NA,NA,"character",
"Platelets","Hemoglobin","Eval","CR","NumbCR","DateCR","DateFollow","StatusFollow","StatusBone","DateBone"),
head(leukem)
```

##	ID	DateStudy	Treat	Sex	Age	FAB	Karnof	WhiteCells	Platelets	Hemoglobin	Eval	CR
## 1	1	072384	D	M	27	5	80	179.0	51	8.8	Y	N
## 2	2	071984	D	M	43	3	90	0.9	14	13.1	Y	N
## 3	3	082984	I	M	36	1	90	1.8	71	6.9	Y	N
## 4	4	090184	I	M	54	1	70	31.9	46	10.8	Y	Y
## 5	5	112984	D	F	49	4	60	24.4	23	10.2	Y	Y
## 6	6	120584	I	M	49	4	90	46.2	21	9.5	Y	Y

##	NumbCR	DateCR	DateFollow	StatusFollow	StatusBone	DateBone	Incl
## 1	NA	<NA>	072984		D	N	<NA> Y
## 2	NA	<NA>	082184		D	N	<NA> Y
## 3	NA	<NA>	082585		D	N	<NA> Y
## 4	1	100884	010286		D	N	<NA> Y
## 5	1	011185	111485		D	N	<NA> Y
## 6	1	122884	112686		D	N	<NA> Y

```
attach(leukem)

#####
# Creating time to event variables
#####
DatumStudy<-as.Date(DateStudy,"%m%d%y")
DatumCR<-as.Date(DateCR,"%m%d%y")
DatumFollow<-as.Date(DateFollow,"%m%d%y")
DatumBone<-as.Date(DateBone,"%m%d%y")

TimeCR<-difftime(DatumCR,DatumStudy)
TimeSurv<-difftime(DatumFollow,DatumStudy)
TimeBone<-difftime(DatumBone,DatumStudy)
TimeEvent<-data.frame(DatumStudy,DatumCR,DatumFollow,DatumBone,TimeCR,TimeBone)
head(TimeEvent)
```

##	DatumStudy	DatumCR	DatumFollow	DatumBone	TimeCR	TimeBone
----	------------	---------	-------------	-----------	--------	----------

```
## 1 1984-07-23      <NA> 1984-07-29      <NA> NA days  NA days
## 2 1984-07-19      <NA> 1984-08-21      <NA> NA days  NA days
## 3 1984-08-29      <NA> 1985-08-25      <NA> NA days  NA days
## 4 1984-09-01 1984-10-08 1986-01-02      <NA> 37 days  NA days
## 5 1984-11-29 1985-01-11 1985-11-14      <NA> 43 days  NA days
## 6 1984-12-05 1984-12-28 1986-11-26      <NA> 23 days  NA days
```

```
#####
# Creating the observed time to event variables
#####
IndCR<-1*I(CR=="Y")
TimetoCR<-TimeCR
TimetoCR[IndCR==0]<-TimeSurv[IndCR==0]
IndSurv<-1*I(StatusFollow=="D")
IndBone<-1*I(StatusBone=="Y")
TimetoBone<-TimeBone
TimetoBone[IndBone==0]<-TimeSurv[IndBone==0]
TimeObs<-data.frame(TimetoCR,IndCR,TimetoBone,IndBone,TimeSurv,IndSurv)
head(TimeObs)
```

```
##      TimetoCR IndCR TimetoBone IndBone TimeSurv IndSurv
## 1      6 days     0      6 days      0  6 days      1
## 2     33 days     0     33 days      0 33 days      1
## 3    361 days     0    361 days      0 361 days      1
## 4     37 days     1    488 days      0 488 days      1
## 5     43 days     1    350 days      0 350 days      1
## 6     23 days     1    721 days      0 721 days      1
```

```
Leukefinal<-data.frame(leukem,TimeEvent,TimeObs)
head(Leukefinal)
```

```
##      ID DateStudy Treat Sex Age FAB Karnof WhiteCells Platelets Hemoglobin Eval CR
## 1 1 072384 D M 27 5 80 179.0 51 8.8 Y N
## 2 2 071984 D M 43 3 90 0.9 14 13.1 Y N
## 3 3 082984 I M 36 1 90 1.8 71 6.9 Y N
## 4 4 090184 I M 54 1 70 31.9 46 10.8 Y Y
## 5 5 112984 D F 49 4 60 24.4 23 10.2 Y Y
## 6 6 120584 I M 49 4 90 46.2 21 9.5 Y Y
##      NumbCR DateCR DateFollow StatusFollow StatusBone DateBone Incl DatumStudy
## 1 NA <NA> 072984 D N <NA> Y 1984-07-23
## 2 NA <NA> 082184 D N <NA> Y 1984-07-19
## 3 NA <NA> 082585 D N <NA> Y 1984-08-29
## 4 1 100884 010286 D N <NA> Y 1984-09-01
## 5 1 011185 111485 D N <NA> Y 1984-11-29
## 6 1 122884 112686 D N <NA> Y 1984-12-05
##      DatumCR DatumFollow DatumBone TimeCR TimeBone TimetoCR IndCR TimetoBone
## 1 <NA> 1984-07-29 <NA> NA days NA days 6 days 0 6 days
## 2 <NA> 1984-08-21 <NA> NA days NA days 33 days 0 33 days
## 3 <NA> 1985-08-25 <NA> NA days NA days 361 days 0 361 days
## 4 1984-10-08 1986-01-02 <NA> 37 days NA days 37 days 1 488 days
## 5 1985-01-11 1985-11-14 <NA> 43 days NA days 43 days 1 350 days
## 6 1984-12-28 1986-11-26 <NA> 23 days NA days 23 days 1 721 days
##      IndBone TimeSurv IndSurv
```

```
## 1      0   6 days      1
## 2      0  33 days      1
## 3      0 361 days      1
## 4      0 488 days      1
## 5      0 350 days      1
## 6      0 721 days      1
```

```
library(ggplot2)
library(survival)
library(survminer)
```

```
## Loading required package: ggpubr
```

```
library(ggfortify)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##   filter, lag
```

```
## The following objects are masked from 'package:base':
##
##   intersect, setdiff, setequal, union
```

```
library(pec)
```

```
## Loading required package: prodlim
```

```
library(ciTools)
```

```
## Registered S3 methods overwritten by 'lme4':
##   method                                from
##   cooks.distance.influence.merMod      car
##   influence.merMod                     car
##   dfbeta.influence.merMod              car
##   dfbetas.influence.merMod             car
```

```
## ciTools version 0.5.1 (C) Institute for Defense Analyses
```

```
library(ldatools)
```

Question 1.1

Shown here is a plot of the survival function of patients diagnosed with leukemia. We are tracking their time till complete remission and have imposed the 95% “log-log” confidence interval around the expected probabilities. For added interpretability X-axis of the plot is scaled logarithmically.

```

Leukefinal$CRnum=rep(0,130)
for(i in 1:130){
  if(Leukefinal$CR[i]=="Y"){
    Leukefinal$CRnum[i]=1
  }
}

```

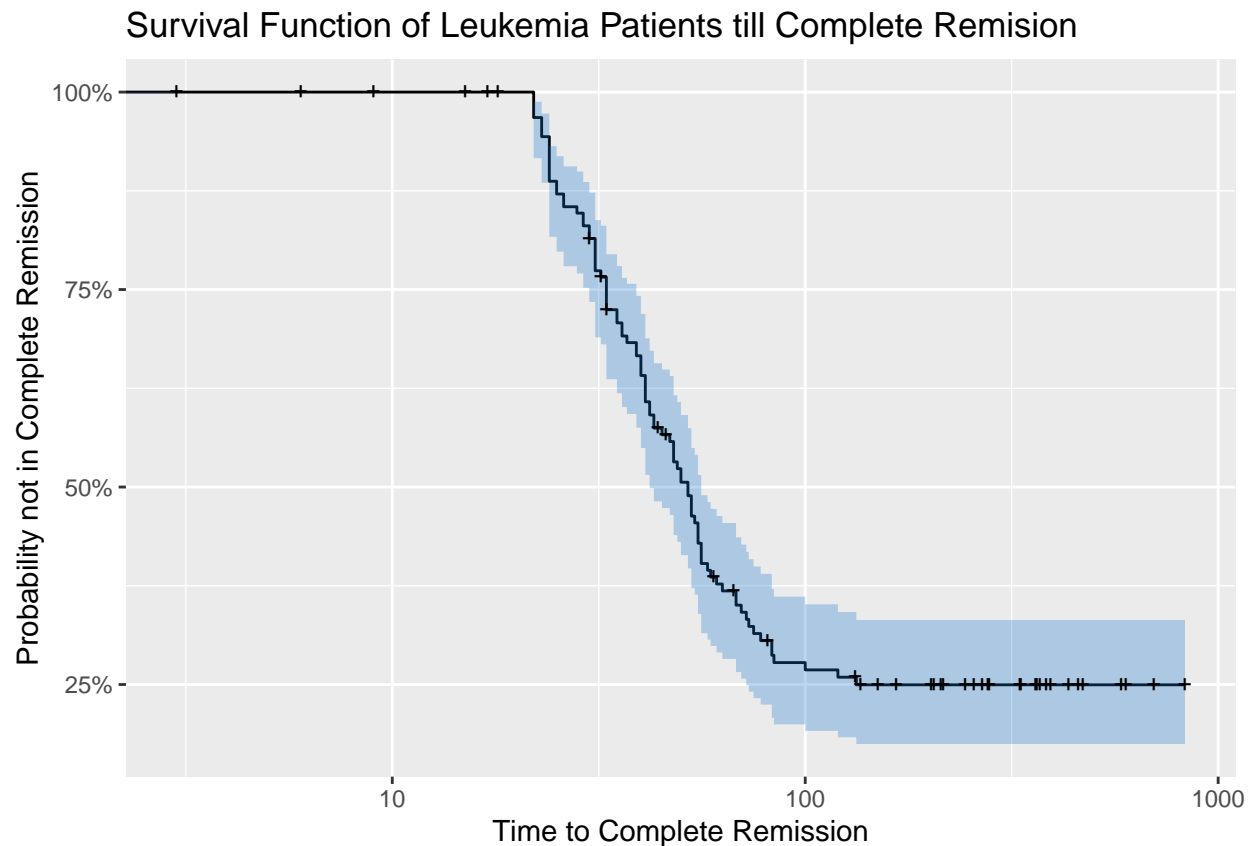
```

fit1.1 = survfit(Surv(as.numeric(TimetoCR),CRnum)~1, data = Leukefinal,conf.type="log-log")
autoplot(fit1.1,surv.colour = 'black',conf.int.fill = 'dodgerblue3')+
  scale_x_log10()+
  ggtitle("Survival Function of Leukemia Patients till Complete Remission")+
  xlab("Time to Complete Remission")+
  ylab("Probability not in Complete Remission")

```

```
## Warning: Transformation introduced infinite values in continuous x-axis
```

```
## Warning: Transformation introduced infinite values in continuous x-axis
```



From this function we can calculate that after about 75 days, 70% of patients are expected to be in complete remission.

```

sumfit1.1=summary(fit1.1)
#this function is exclusive to the desired quantile so have to add the additional .001 to get the 70% b
.3*124

```

```
## [1] 37.2
```

```
sumfit1.1
```

```
## Call: survfit(formula = Surv(as.numeric(TimetoCR), CRnum) ~ 1, data = Leukefinal,  
##      conf.type = "log-log")
```

```
##  
##   time n.risk n.event survival std.err lower 95% CI upper 95% CI  
##    22    124     4   0.968  0.0159    0.916    0.988  
##    23    120     3   0.944  0.0207    0.885    0.973  
##    24    117     7   0.887  0.0284    0.817    0.932  
##    25    110     2   0.871  0.0301    0.798    0.919  
##    26    108     2   0.855  0.0316    0.780    0.906  
##    28    106     1   0.847  0.0323    0.770    0.899  
##    29    105     2   0.831  0.0337    0.752    0.886  
##    30    103     2   0.815  0.0349    0.734    0.873  
##    31    100     5   0.774  0.0376    0.689    0.838  
##    32     95     1   0.766  0.0381    0.681    0.831  
##    33     93     5   0.724  0.0402    0.636    0.795  
##    35     87     2   0.708  0.0410    0.619    0.780  
##    36     85     2   0.691  0.0417    0.601    0.765  
##    37     83     1   0.683  0.0420    0.593    0.757  
##    39     82     2   0.666  0.0426    0.575    0.742  
##    40     80     3   0.641  0.0434    0.549    0.719  
##    41     77     4   0.608  0.0442    0.515    0.688  
##    42     73     2   0.591  0.0445    0.499    0.672  
##    43     71     2   0.575  0.0448    0.482    0.657  
##    45     68     1   0.566  0.0449    0.473    0.649  
##    47     66     1   0.558  0.0451    0.465    0.641  
##    48     65     3   0.532  0.0454    0.439    0.616  
##    49     62     1   0.523  0.0455    0.431    0.608  
##    50     61     2   0.506  0.0456    0.414    0.591  
##    52     59     2   0.489  0.0456    0.397    0.575  
##    53     57     3   0.463  0.0456    0.372    0.549  
##    54     54     1   0.455  0.0455    0.364    0.541  
##    55     53     3   0.429  0.0453    0.339    0.515  
##    56     50     3   0.403  0.0449    0.315    0.490  
##    58     47     1   0.395  0.0448    0.307    0.481  
##    59     46     1   0.386  0.0446    0.299    0.472  
##    61     44     1   0.377  0.0445    0.291    0.463  
##    63     43     1   0.368  0.0443    0.282    0.455  
##    68     41     2   0.350  0.0439    0.266    0.436  
##    70     39     1   0.341  0.0437    0.257    0.427  
##    72     38     1   0.333  0.0435    0.249    0.418  
##    73     37     1   0.324  0.0432    0.241    0.409  
##    75     36     1   0.315  0.0429    0.233    0.399  
##    78     35     1   0.306  0.0426    0.225    0.390  
##    83     33     2   0.287  0.0420    0.208    0.371  
##    84     31     1   0.278  0.0417    0.200    0.361  
##   100     30     1   0.269  0.0413    0.191    0.352  
##   120     29     1   0.259  0.0409    0.183    0.342  
##   133     27     1   0.250  0.0405    0.175    0.332
```

We may also calculate the probability a person takes longer than 3 months (90 days) to enter complete

remission. We calculate the mean probability for this to be true at 27.78% with a 95% confidence band of 19.96% and 36.12%.

```
fit1.1$surv[fit1.1$time==84]
```

```
## [1] 0.2777733
```

```
fit1.1$lower[fit1.1$time==84]
```

```
## [1] 0.1995904
```

```
fit1.1$upper[fit1.1$time==84]
```

```
## [1] 0.3612412
```

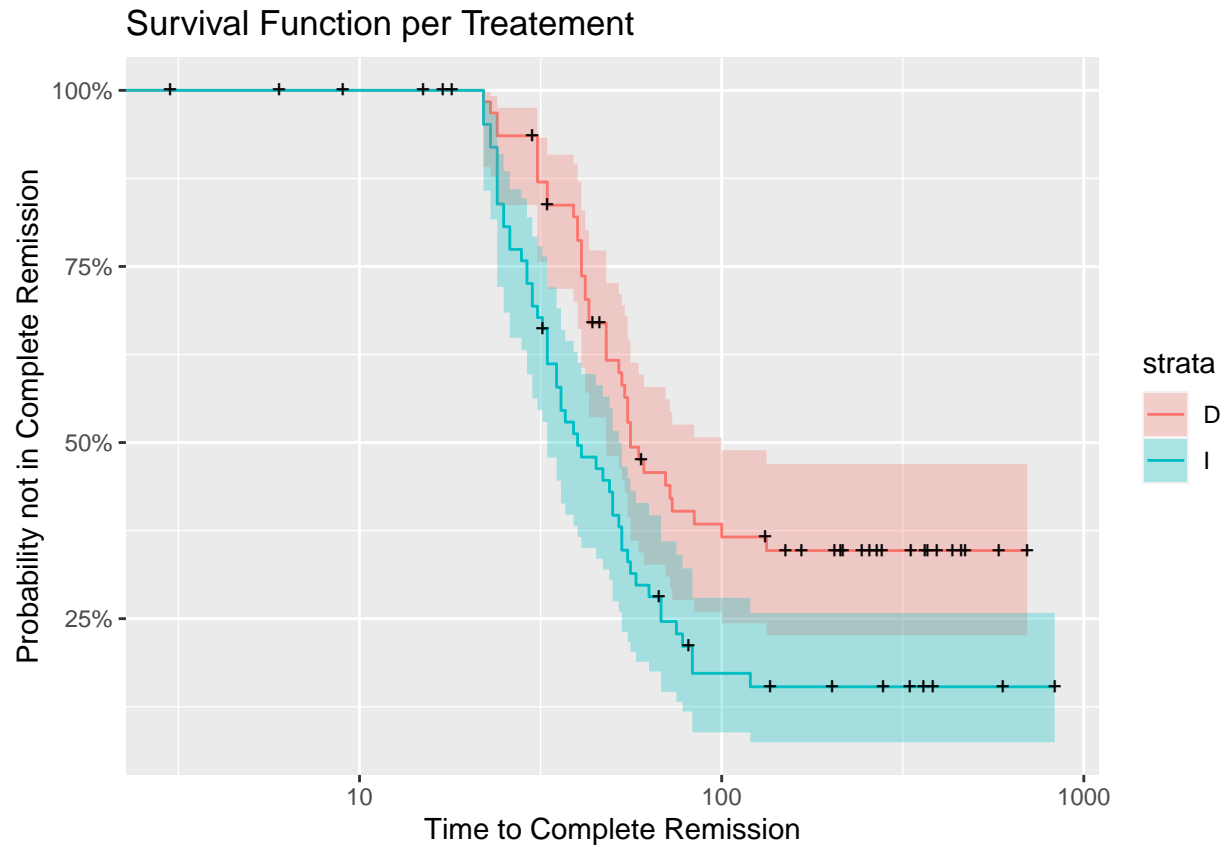
#Question 1.2

We now wish to determine the effect of treatment type on a patients time to enter complete remission. We are examining two treatment types, Daunorubicin and Idarubicin, and we wish to determine if there is a significant difference in the time till complete remission. Here we plot the survival function respective of both treatment types. Visual analysis shows a stark difference in performance in favor of Idarubicin.

Numerically we can examine the effect further. We do this by examining the median time to complete remission for both drugs. Upon doing this we find that 50% of patients treated with Idarubicin are expected to enter complete remission within 40 days. This is compared to expectation of 56 days for patients treated with Daunorubicin.

With this analysis we can claim that, in terms of reducing time to complete remission, Idarubicin is a superior drug to Daunorubicin.

```
fit1.2=survfit(Surv(as.numeric(TimetoCR),CRnum)~Treat, data = Leukefinal,conf.type="log-log")
treat.plot=autoplot(fit1.2)+
  scale_x_log10()+
  ggtitle("Survival Function per Treatement")+
  xlab("Time to Complete Remission")+
  ylab("Probability not in Complete Remission")
treat.plot
```



```
fit1.2
```

```
## Call: survfit(formula = Surv(as.numeric(TimetoCR), CRnum) ~ Treat,
##   data = Leukefinal, conf.type = "log-log")
##
##           n events median 0.95LCL 0.95UCL
## Treat=D 65      38      56      48      100
## Treat=I 65      51      40      33      52
```

```
sumfit1.2=summary(fit1.2)
sumfit1.2
```

```
## Call: survfit(formula = Surv(as.numeric(TimetoCR), CRnum) ~ Treat,
##   data = Leukefinal, conf.type = "log-log")
##
##           Treat=D
##   time n.risk n.event survival std.err lower 95% CI upper 95% CI
##    22     62      1   0.984  0.0160    0.891    0.998
##    23     61      1   0.968  0.0224    0.877    0.992
##    24     60      2   0.935  0.0312    0.837    0.975
##    31     57      4   0.870  0.0429    0.756    0.933
##    33     53      2   0.837  0.0472    0.718    0.909
##    39     50      1   0.820  0.0491    0.699    0.896
##    40     49      2   0.787  0.0525    0.661    0.870
##    41     47      3   0.737  0.0566    0.606    0.830
```

##	42	44	2	0.703	0.0588		0.571		0.802
##	43	42	2	0.670	0.0605		0.536		0.773
##	48	38	3	0.617	0.0630		0.481		0.727
##	52	35	1	0.599	0.0636		0.463		0.711
##	53	34	1	0.582	0.0641		0.446		0.695
##	54	33	1	0.564	0.0646		0.428		0.679
##	55	32	2	0.529	0.0652		0.394		0.646
##	56	30	2	0.493	0.0654		0.360		0.613
##	59	28	1	0.476	0.0654		0.344		0.596
##	61	26	1	0.457	0.0654		0.327		0.579
##	70	25	1	0.439	0.0653		0.310		0.561
##	72	24	1	0.421	0.0651		0.293		0.544
##	73	23	1	0.403	0.0648		0.276		0.526
##	84	22	1	0.384	0.0644		0.260		0.507
##	100	21	1	0.366	0.0638		0.244		0.489
##	133	19	1	0.347	0.0633		0.226		0.470
##									
##	Treat=I								
##	time	n.risk	n.event	survival	std.err	lower	95% CI	upper	95% CI
##	22	62	3	0.952	0.0273		0.8574		0.984
##	23	59	2	0.919	0.0346		0.8170		0.966
##	24	57	5	0.839	0.0467		0.7210		0.910
##	25	52	2	0.806	0.0502		0.6844		0.885
##	26	50	2	0.774	0.0531		0.6487		0.860
##	28	48	1	0.758	0.0544		0.6312		0.846
##	29	47	2	0.726	0.0567		0.5966		0.820
##	30	45	2	0.694	0.0585		0.5626		0.792
##	31	43	1	0.677	0.0594		0.5459		0.778
##	32	42	1	0.661	0.0601		0.5293		0.764
##	33	40	3	0.612	0.0620		0.4788		0.720
##	35	37	2	0.579	0.0629		0.4458		0.690
##	36	35	2	0.546	0.0635		0.4135		0.660
##	37	33	1	0.529	0.0637		0.3975		0.644
##	39	32	1	0.513	0.0638		0.3817		0.629
##	40	31	1	0.496	0.0639		0.3660		0.613
##	41	30	1	0.479	0.0639		0.3504		0.597
##	45	29	1	0.463	0.0638		0.3350		0.581
##	47	28	1	0.446	0.0636		0.3197		0.565
##	49	27	1	0.430	0.0634		0.3046		0.549
##	50	26	2	0.397	0.0626		0.2747		0.516
##	52	24	1	0.380	0.0622		0.2600		0.499
##	53	23	2	0.347	0.0610		0.2311		0.466
##	55	21	1	0.331	0.0603		0.2168		0.449
##	56	20	1	0.314	0.0595		0.2028		0.432
##	58	19	1	0.298	0.0586		0.1888		0.414
##	63	18	1	0.281	0.0577		0.1751		0.397
##	68	16	2	0.246	0.0555		0.1461		0.359
##	75	14	1	0.228	0.0543		0.1320		0.341
##	78	13	1	0.211	0.0529		0.1182		0.321
##	83	11	2	0.172	0.0497		0.0886		0.280
##	120	9	1	0.153	0.0477		0.0745		0.258

Now we wish to determine which drug performs best within a two month time limit. Using similar methods from the previous analysis we can determine that a patient undergoing treatment with Idarubicin will have a

probability to enter complete remission within two months of approximately 70.24% with a 95% confidence bound of 58.57% and 71.12%. We can also claim that a patient undergoing treatment with Daunorubicin will have a probability of 52% to enter complete remission within two months. This prediction carries a 95% confidence boundry of 40.35% and 65.62%. From these results we can extrapolate that Idarubicin is a better form of treatment within two months.

```
D2month=c(sumfit1.2$urv[sumfit1.2$time==59],
          sumfit1.2$lower[sumfit1.2$time==59],
          sumfit1.2$upper[sumfit1.2$time==59])
```

```
I2month=c(sumfit1.2$urv[sumfit1.2$time==58],
          sumfit1.2$lower[sumfit1.2$time==58],
          sumfit1.2$upper[sumfit1.2$time==58])
```

```
1-D2month
```

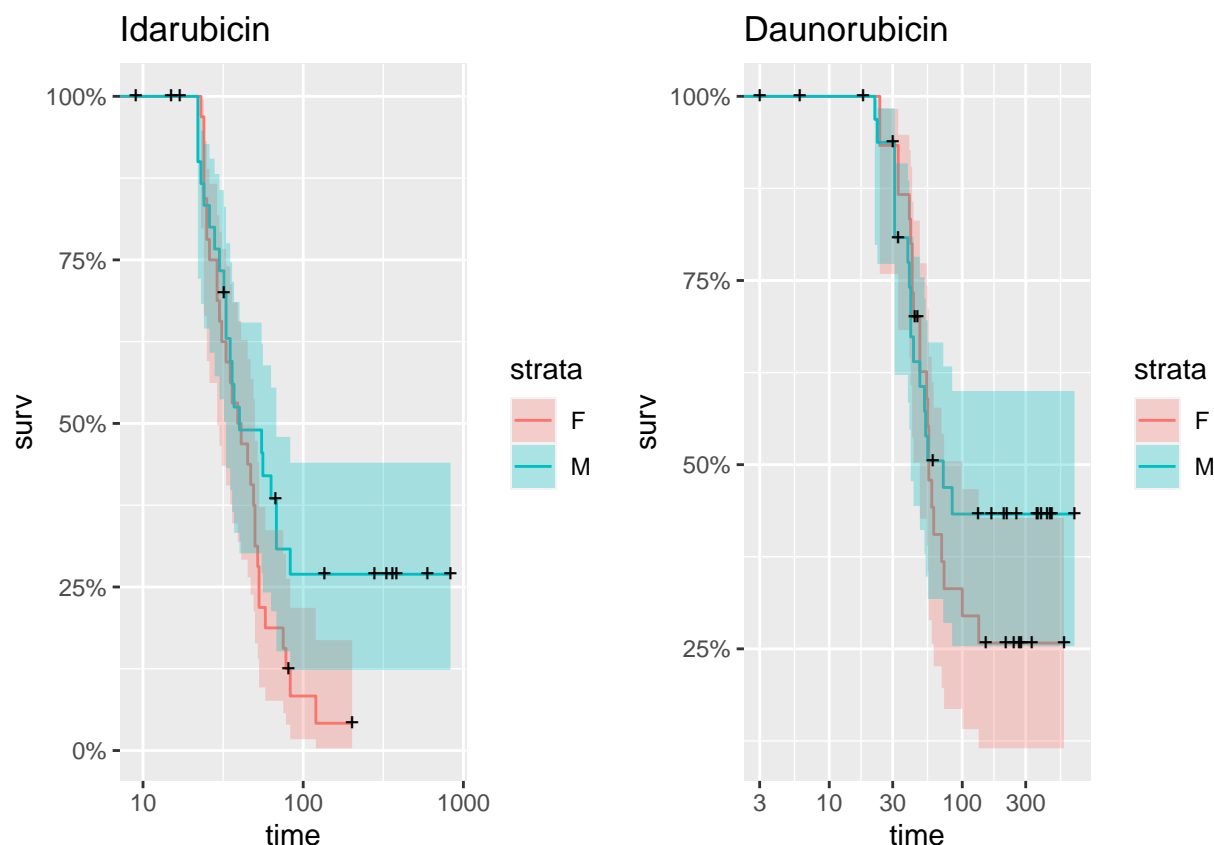
```
## [1] 0.5242248 0.6561648 0.4035932
```

```
1-I2month
```

```
## [1] 0.7024194 0.8111535 0.5857919
```

#Question 1.3 Now we wish to determine the effect of treatment respective to the sex of the patient. Firstly, we must determine the survival function for each sex separately. Plots of both sex's response to the treatments can be seen here. Initial Visual analysis does indicate patient sex influencing the performance of the drug.

```
fit1.3.I=survfit(Surv(as.numeric(TimetoCR),CRnum)~Sex, data = Leukefinal[Leukefinal$Treat=="I",],conf.t,
fit1.3.D=survfit(Surv(as.numeric(TimetoCR),CRnum)~Sex, data = Leukefinal[Leukefinal$Treat=="D",],conf.t,
I.plot=autoplot(fit1.3.I)+ggtitle("Idarubicin")+scale_x_log10()
D.plot=autoplot(fit1.3.D)+ggtitle("Daunorubicin")+scale_x_log10()
gridExtra::grid.arrange(I.plot,D.plot,ncol=2)
```



Further visual examination of each sex group and treatment type does not show signs of deviation from the expectation gathered from the original plot detailing treatment type vs remission time. We can numerically examine this relation further. We use the respective survival functions for each sex to compute the median time to complete remission. Doing so shows that 50% of men and women undergoing treatment with Daunorubicin are expected to enter complete remission before 72 and 56 days respectively. Whereas, 50% of men and women undergoing treatment with Idarubicin are expected to enter complete remission before 40 days. When we examine the confidence intervals of each groups median probabilities we arrive at a similar conclusion as the bounds for each treatment are quite large and generally encapsulate each other. From this we can say that the sex of the patient is not a confounding variable for their response to certain treatment. A comparison of the survival function per sex and treatment type in addition to a table comparing each distribution can be found [here](#).

```
md=rep(0,130)
md=md+1*(Leukefinal$Sex=="M"&Leukefinal$Treat=="D")

mi=rep(0,130)
mi=mi+1*(Leukefinal$Sex=="M"&Leukefinal$Treat=="I")

fd=rep(0,130)
fd=fd+1*(Leukefinal$Sex=="F"&Leukefinal$Treat=="D")

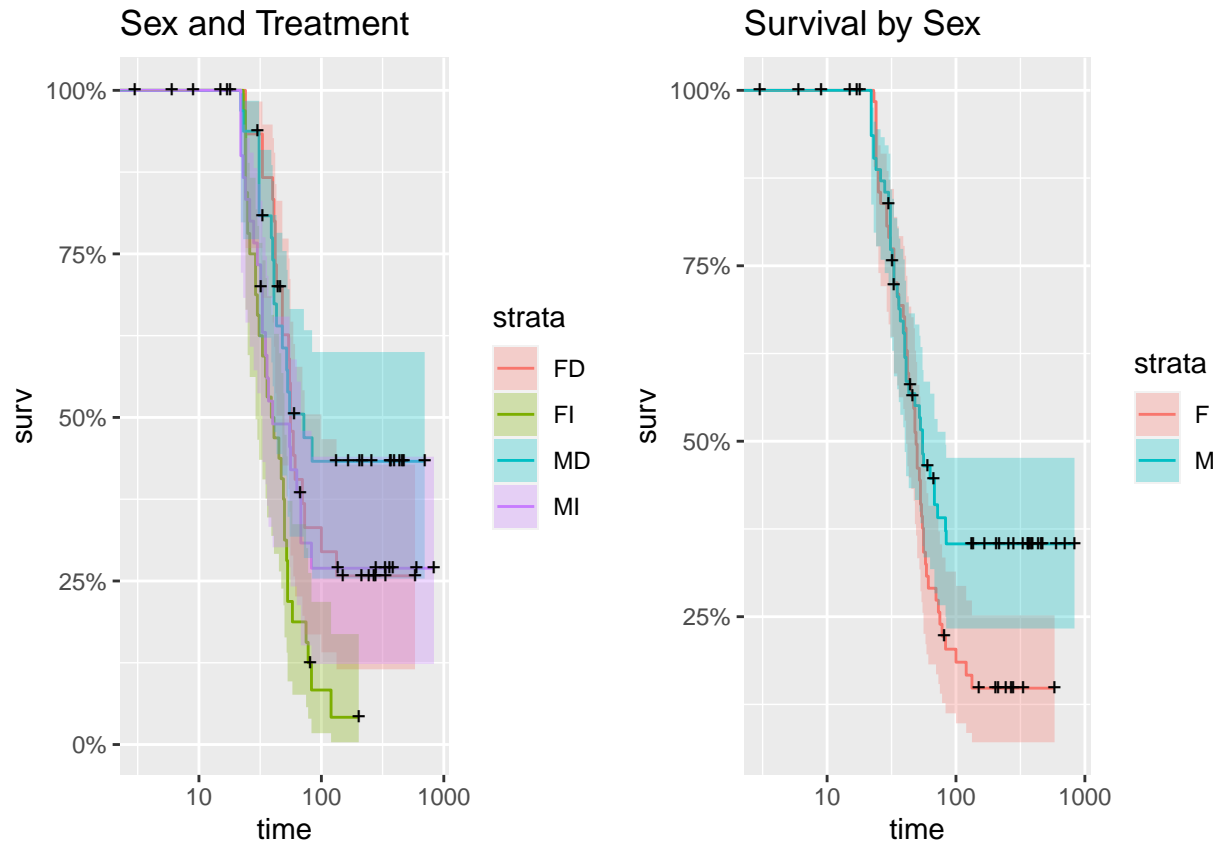
fi=rep(0,130)
fi=fi+1*(Leukefinal$Sex=="F"&Leukefinal$Treat=="I")
cr.treat.sex=tibble(CR=Leukefinal$TimetoCR)%>%
  mutate(CRnum=Leukefinal$CRnum)%>%
  mutate(MD=md)%>%
```

```

mutate(MI=mi)%>%
mutate(FD=fd)%>%
mutate(FI=fi)
cr.treat.sex$Treat=rep("",130)
for(i in 1:130){
  if(md[i]==1){
    cr.treat.sex$Treat[i]="MD"
  }
  if(mi[i]==1){
    cr.treat.sex$Treat[i]="MI"
  }
  if(fd[i]==1){
    cr.treat.sex$Treat[i]="FD"
  }
  if(fi[i]==1){
    cr.treat.sex$Treat[i]="FI"
  }
}

fit1.3.sex=survfit(Surv(as.numeric(TimetoCR),CRnum)~Sex, data = Leukefinal,conf.type="log-log")
sex.plot=autoplot(fit1.3.sex)+ggtitle("Survival by Sex")+scale_x_log10()
fit1.3.full=survfit(Surv(as.numeric(TimetoCR),CRnum)~Treat, data = cr.treat.sex,conf.type="log-log")
full.plot=autoplot(fit1.3.full)+ggtitle("Sex and Treatment")+scale_x_log10()
gridExtra::grid.arrange(full.plot,sex.plot,ncol=2)

```



```
fit1.3.full
```

```
## Call: survfit(formula = Surv(as.numeric(TimetoCR), CRnum) ~ Treat,
##   data = cr.treat.sex, conf.type = "log-log")
##
##           n events median 0.95LCL 0.95UCL
## Treat=FD 30      21      56      48     100
## Treat=FI 35      30      40      29      50
## Treat=MD 35      17      72      41      NA
## Treat=MI 30      21      40      33      68
```

#Question 2.1

Now we wish to determine a patients survival time. We will be focusing only on patients diagnosed with acute myelogenous leukemia (AML). To begin, we will calculate and plot the survival function detailing the time until death. From this survival function we can calculate the median survival time to be 458 days, with 25% of patients passing before 260 days and 75% of patients passing before 1424 days.

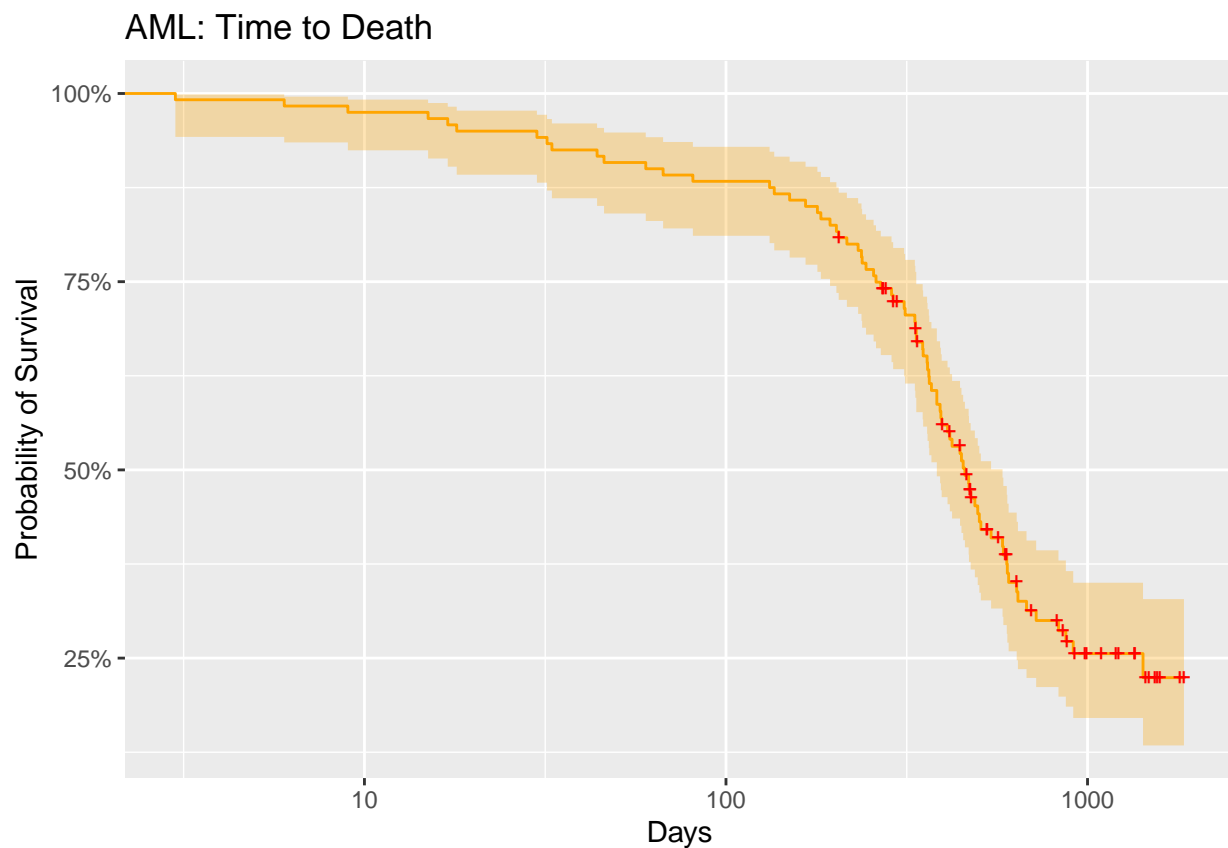
```
leukem.aml=Leukefinal[Leukefinal$Eval=="Y",]
leukem.aml$statusnum=rep(0,120)
for(i in 1:120){
  if(leukem.aml$StatusFollow[i]=="D"){
    leukem.aml$statusnum[i]=1
  }
}
```

```
leukem.aml.surv=with(leukem.aml,Surv(TimeSurv,statusnum))
leukem.aml$km=leukem.aml.surv
```

```
fit2.1=survfit(km~1,leukem.aml,conf.type="log-log")
autoplot(fit2.1, surv.colour = 'orange', censor.colour = 'red')+
  scale_x_log10()+
  ggtitle("AML: Time to Death")+
  xlab("Days")+
  ylab("Probability of Survival")
```

```
## Warning: Transformation introduced infinite values in continuous x-axis
```

```
## Warning: Transformation introduced infinite values in continuous x-axis
```



```
fit2.1
```

```
## Call: survfit(formula = km ~ 1, data = leukem.aml, conf.type = "log-log")
##
##      n  events  median 0.95LCL 0.95UCL
##    120     79    458    383    582
```

```
quantile(fit2.1,c(.25,.5,.75))
```

```
## $quantile
##   25   50   75
## 260 458 1424
##
## $lower
##   25   50   75
## 194 383 637
##
## $upper
##   25   50   75
## 336 582  NA
```

#Question 2.2

Now we wish to use a cox proportional hazard model to determine the effect of the patients age and their treatment (Daunorubicin and Idarubicin). From this model we can claim that a patients age and their treatment are significant factors when trying to predict their survival function. From this we can claim that for each unitary increase in age the probability of death increases by 1.83%. Patients that undergo treatment with Idarubicin have their probability of survival increase by a substantial 58.62%

```
cox2.2=coxph(km~Age+Treat,data=leukem.aml)
summary(cox2.2)
```

```
## Call:
## coxph(formula = km ~ Age + Treat, data = leukem.aml)
##
##      n= 120, number of events= 79
##
##              coef exp(coef)  se(coef)      z Pr(>|z|)
## Age          0.018499  1.018671  0.008772  2.109   0.0350 *
## TreatI     -0.461316  0.630453  0.230394 -2.002   0.0453 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##              exp(coef) exp(-coef) lower .95 upper .95
## Age              1.0187      0.9817   1.0013   1.0363
## TreatI           0.6305      1.5862   0.4014   0.9903
##
## Concordance= 0.587 (se = 0.036 )
## Likelihood ratio test= 9.15 on 2 df,  p=0.01
## Wald test               = 9.26 on 2 df,  p=0.01
## Score (logrank) test = 9.43 on 2 df,  p=0.009
```

#Question 2.3

Now we wish to determine the effect of going into complete remission or undergoing a bone marrow transfusion has on the survival rate. Upon the addition of these parameters the effect of the type of treatment has become insignificant. Similarly the status of their bone marrow transfusion is insignificant as well. Unsurprisingly, whether the patient has undergone complete remission is a very significant piece of knowledge.

```
cox2.3=coxph(km~Age+Treat+CR+StatusBone,data=leukem.aml)
summary(cox2.3)
```

```
## Call:
## coxph(formula = km ~ Age + Treat + CR + StatusBone, data = leukem.aml)
##
##      n= 120, number of events= 79
##
##              coef exp(coef)  se(coef)      z Pr(>|z|)
## Age           0.023842  1.024128  0.008794  2.711  0.0067 **
## TreatI        -0.224481  0.798931  0.235414 -0.954  0.3403
## CRY           -1.623896  0.197129  0.253821 -6.398 1.58e-10 ***
## StatusBoneY   -0.298732  0.741758  0.488430 -0.612  0.5408
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##              exp(coef) exp(-coef) lower .95 upper .95
## Age              1.0241    0.9764    1.0066    1.0419
## TreatI           0.7989    1.2517    0.5036    1.2673
## CRY              0.1971    5.0728    0.1199    0.3242
## StatusBoneY      0.7418    1.3481    0.2848    1.9320
##
## Concordance= 0.716 (se = 0.031 )
## Likelihood ratio test= 50.7 on 4 df,  p=3e-10
## Wald test              = 53.32 on 4 df,  p=7e-11
## Score (logrank) test = 64.27 on 4 df,  p=4e-13
```

#2.4

Now we wish to determine which covariates satisfy the assumptions of a cox proportional hazard model. To begin we can test the weighted residuals. We see that all covariates aside from age are significant or very close to significance at the 10% level. Age is largely insignificant. The cox-snell residuals show that the model does fit the data well. Residual plots reconfirm this claim, with the schoenfeld, martingale, dfbeta, and dfbetas (dfbeta/standardized residuals) showing no signs of stark deviance from the model assumptions. To assure that our continuous variable, Age, is in its correct form we examine the martingale residuals of various transforms (logarithmic and square-root). Unfortunately neither of these transforms appear to provide evidence of linearity potential.

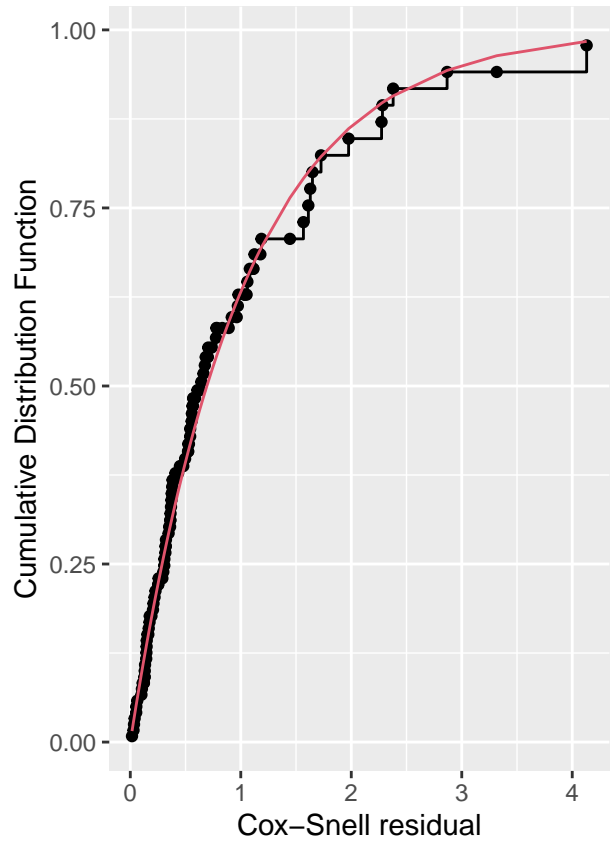
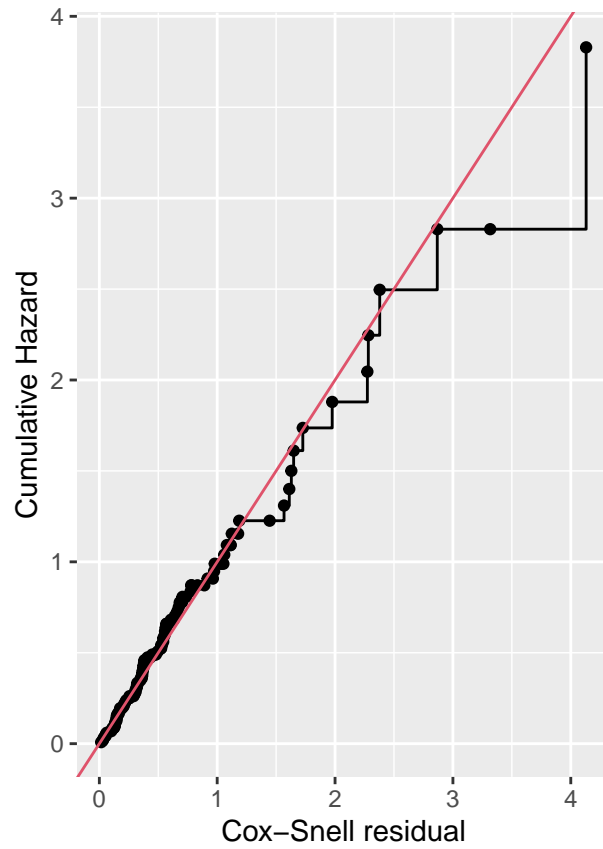
```
test2.4=cox.zph(cox2.3)
test2.4
```

```
##              chisq df      p
## Age           0.496  1 0.481
## Treat         3.642  1 0.056
## CR            4.335  1 0.037
## StatusBone    2.628  1 0.105
## GLOBAL        13.054  4 0.011
```

```
coxsnellplot=gg_coxsnell(cox2.3)+
  geom_abline(intercept=0, slope=1, col=2)

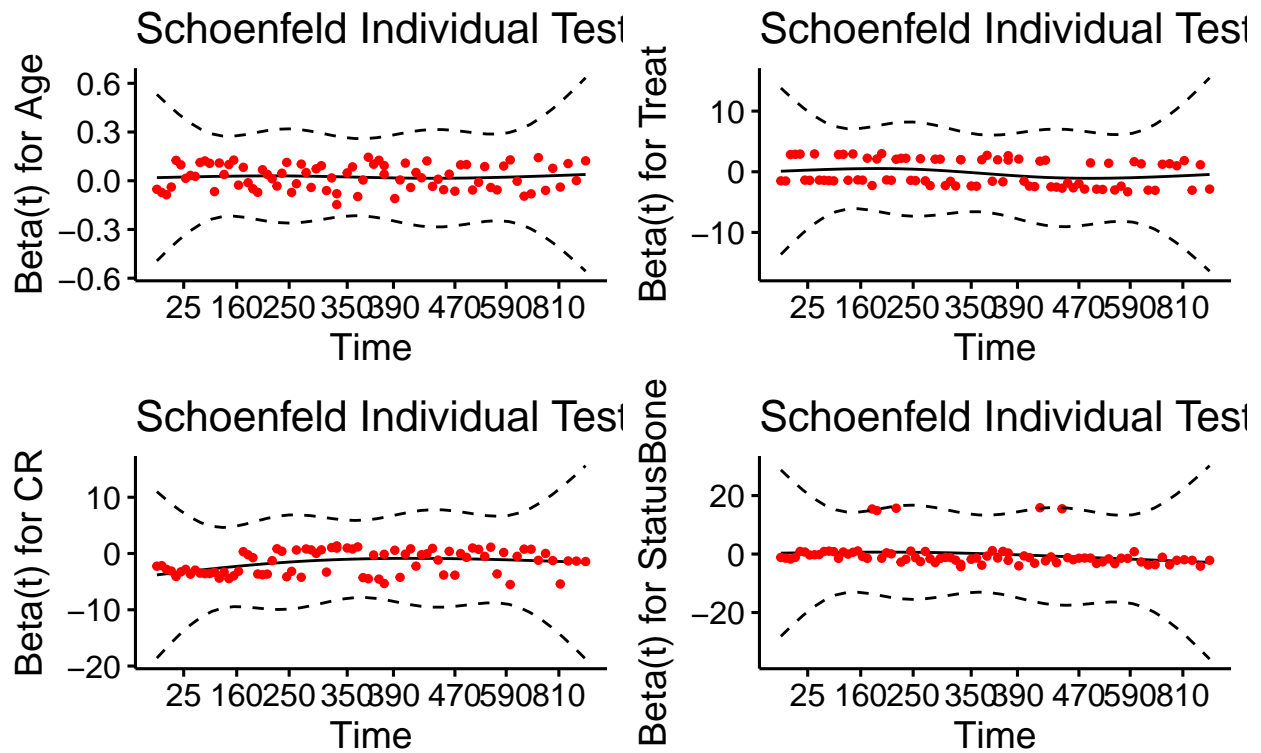
coxsnellplot.cdf=gg_coxsnell(cox2.3,type="cdf")+
  geom_line(aes(y=F), col=2)

gridExtra::grid.arrange(coxsnellplot,coxsnellplot.cdf,ncol=2)
```



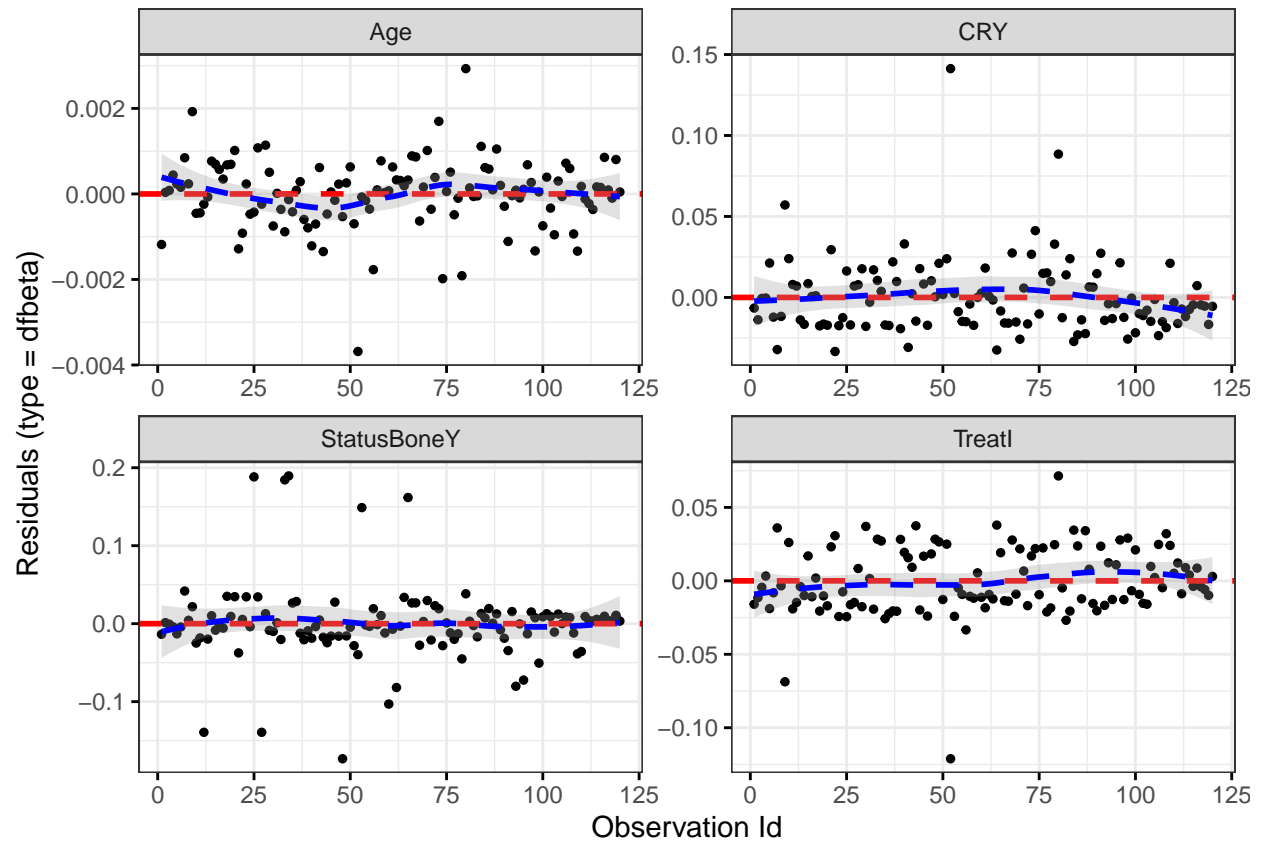
```
ggcoxzph(test2.4)
```


Global Schoenfeld Test p: 0.01102



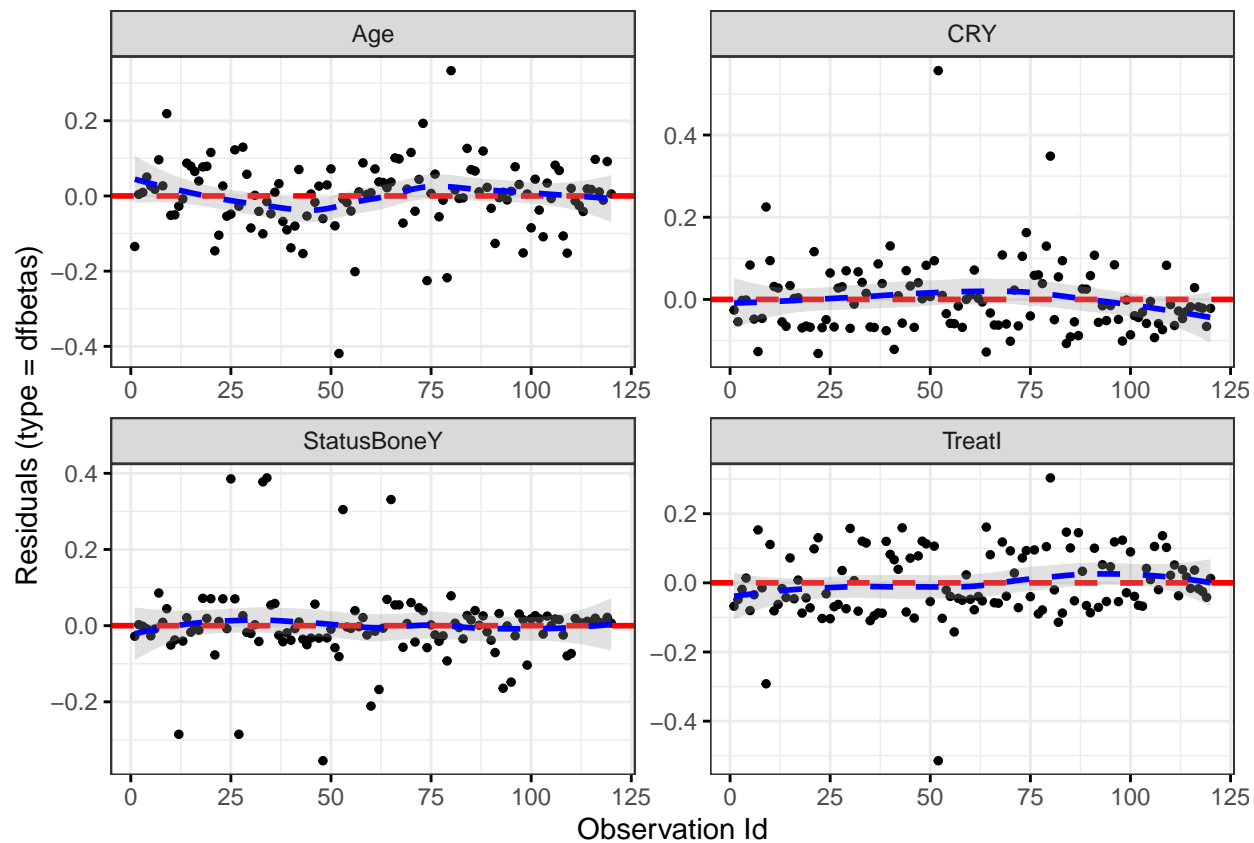
```
ggcoxdiagnostics(cox2.3, type = "dfbeta",  
  linear.predictions = F, ggtheme = theme_bw())
```

```
## 'geom_smooth()' using formula 'y ~ x'
```



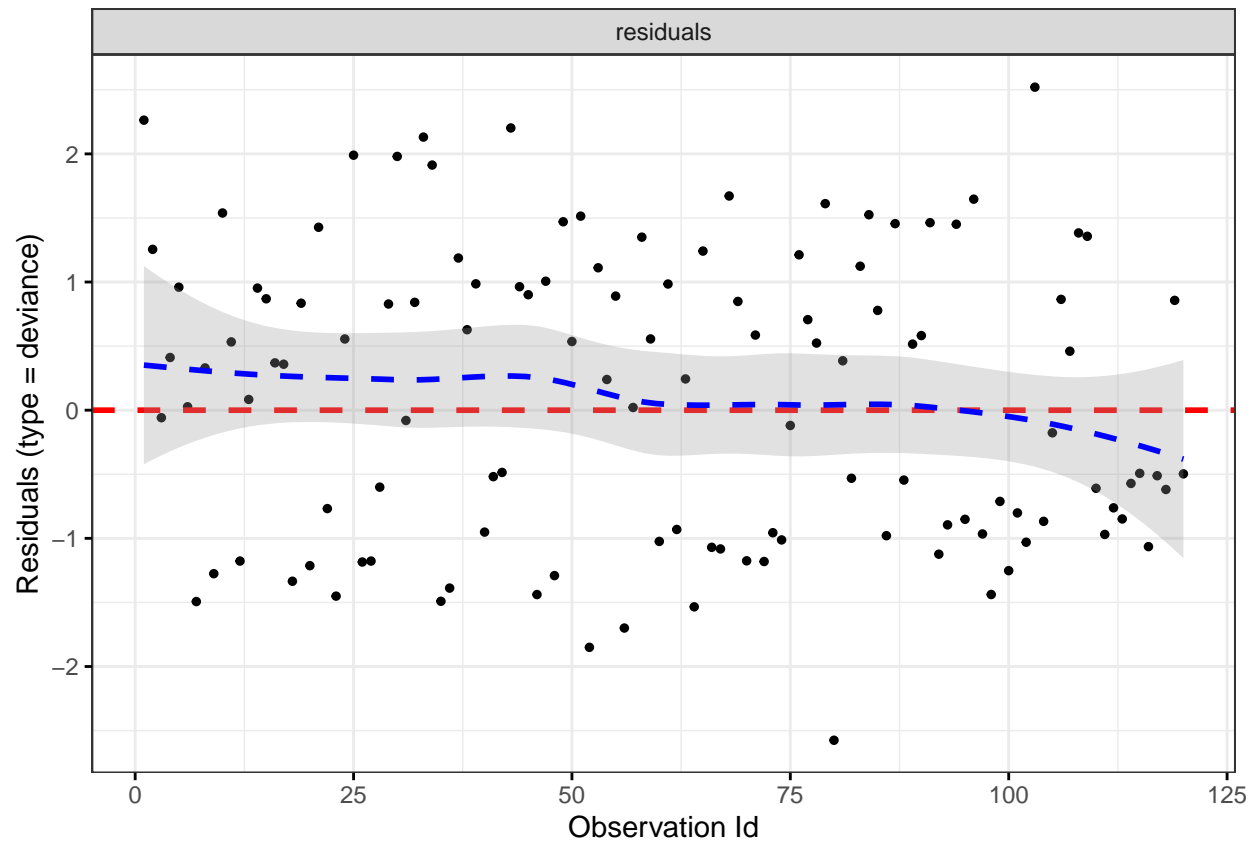
```
ggcoxdiagnostics(cox2.3, type = "dfbetas",
  linear.predictions = F, ggtheme = theme_bw())
```

```
## 'geom_smooth()' using formula 'y ~ x'
```



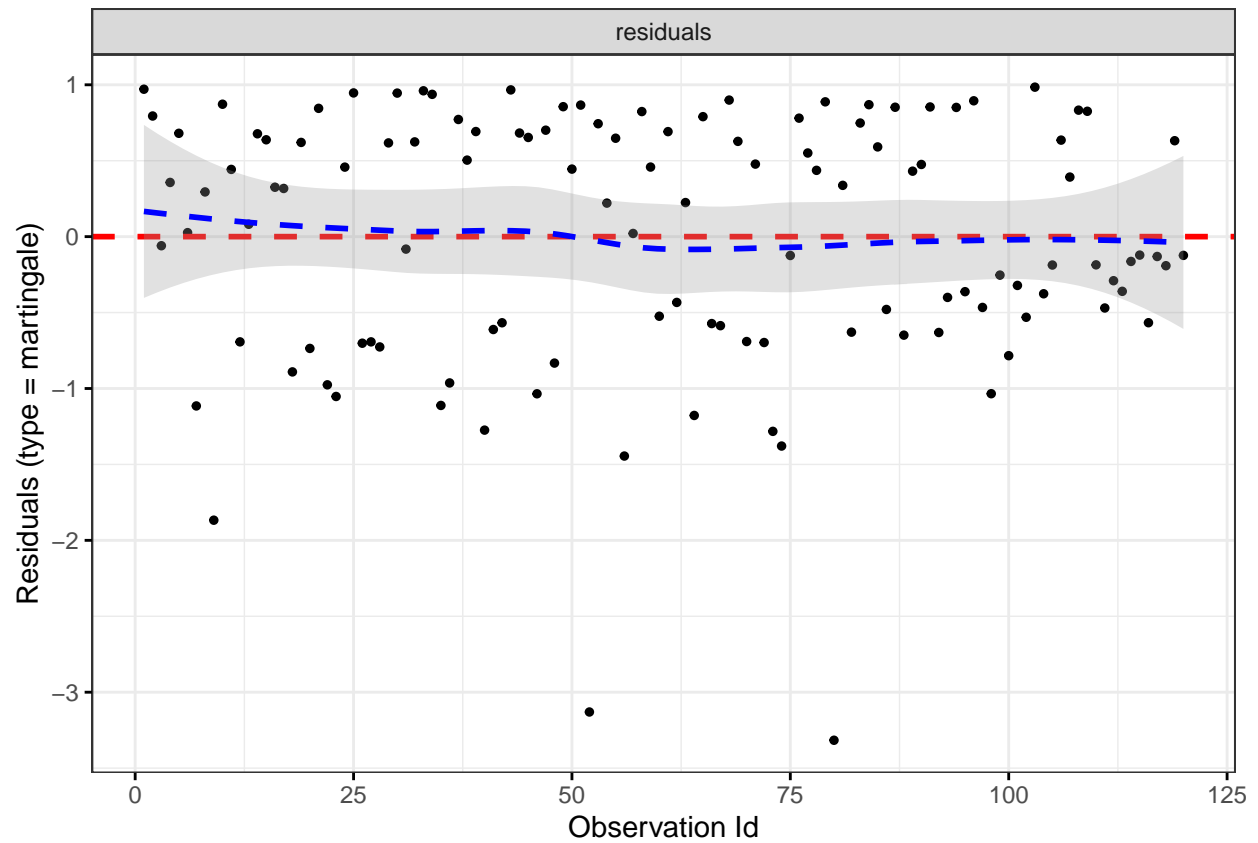
```
ggcoxdiagnostics(cox2.3, type = "deviance",
  linear.predictions = F, ggtheme = theme_bw())
```

```
## 'geom_smooth()' using formula 'y ~ x'
```

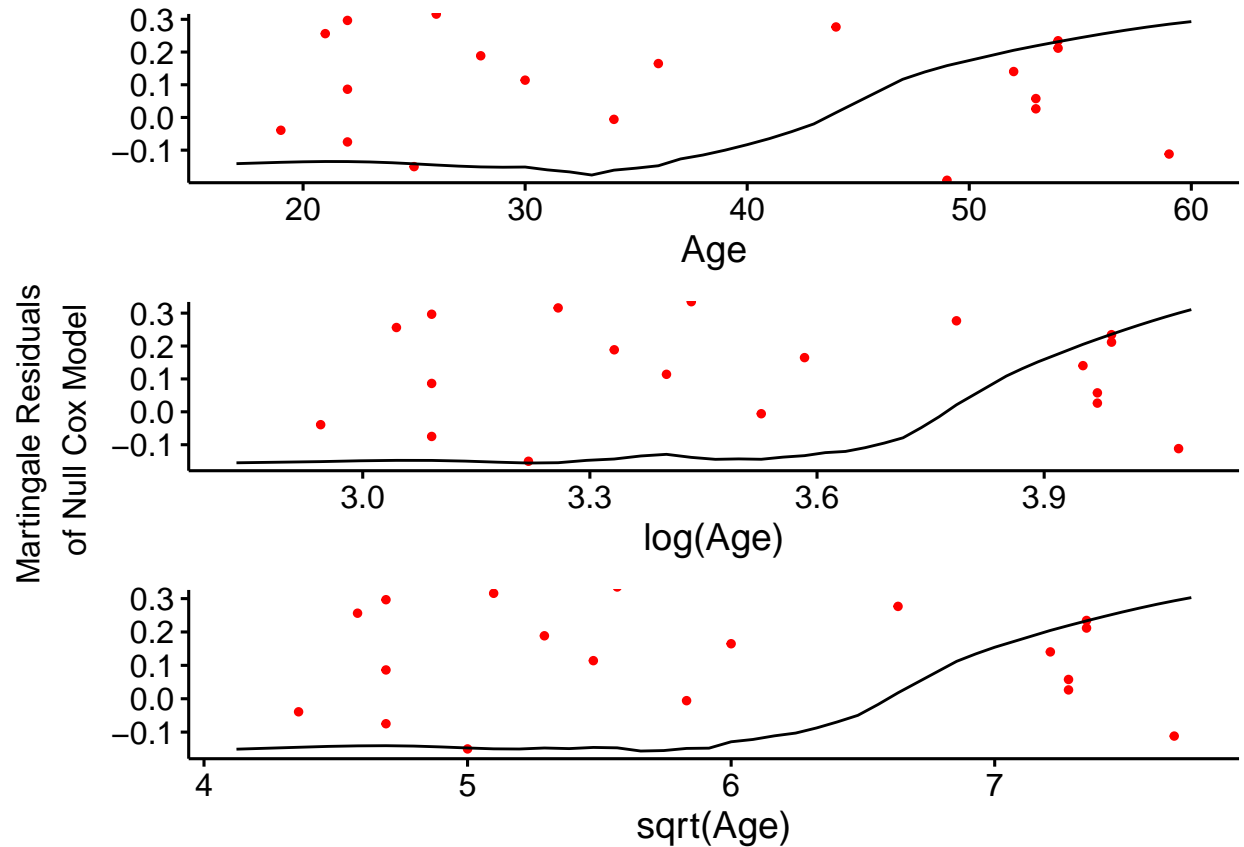


```
ggcoxdiagnostics(cox2.3, type = "martingale",
  linear.predictions = F, ggtheme = theme_bw())
```

```
## 'geom_smooth()' using formula 'y ~ x'
```



```
cox.age=coxph(km~Age+log(Age)+sqrt(Age),data=leukem.aml)
lin2.4=ggcoxfunctional(cox.age,data=leukem.aml)
lin2.4
```



#3.1

Now we wish to model the relationship between survival, age, and treatment as an accelerated failure time model.

```
fit3.1=survreg(km~Age+Treat,data=leukem.aml)
summary(fit3.1)
```

```
##
## Call:
## survreg(formula = km ~ Age + Treat, data = leukem.aml)
##              Value Std. Error      z      p
## (Intercept)  7.10024    0.41283 17.20 <2e-16
## Age         -0.01831    0.00913  -2.01  0.045
## TreatI       0.51223    0.23285   2.20  0.028
## Log(scale)   0.01578    0.09461   0.17  0.868
##
## Scale= 1.02
##
## Weibull distribution
## Loglik(model)= -599.1   Loglik(intercept only)= -604.1
##  Chisq= 9.92 on 2 degrees of freedom, p= 0.007
## Number of Newton-Raphson Iterations: 5
## n= 120
```

#3.2

#3.3

```
para=fit3.1$coefficients
lscale=log(fit3.1$scale)
V=fit3.1$var
para
```

```
## (Intercept)      Age      TreatI
##  7.10023549 -0.01830923  0.51223207
```

```
lscale
```

```
## [1] 0.01578274
```

```
V
```

```
##           [,1]      [,2]      [,3]      [,4]
## [1,]  0.170432253 -3.499067e-03 -0.0341777298  4.197719e-03
## [2,] -0.003499067  8.330842e-05  0.0002569706 -9.267087e-05
## [3,] -0.034177730  2.569706e-04  0.0542213815  1.785605e-03
## [4,]  0.004197719 -9.267087e-05  0.0017856052  8.951442e-03
```

```
lambda<-exp(-para[1]*exp(-lscale))
alpha<-exp(-lscale)
beta<-para[-1]*exp(-lscale)
x<-c(lambda,alpha,beta)
names(x)[1]<-"lambda"
names(x)[2]<-"alpha"
m<-length(para[-1])
G<-matrix(0,m+2,ncol=m+2)
G[1,1]<-exp(-para[1]*exp(-lscale))*exp(-lscale)
G[2:(m+1),3:(m+2)]<-diag(m)*(-exp(-lscale))
G[m+2,1]<-exp(-para[1]*exp(-lscale))*para[1]*exp(-lscale)
G[m+2,2]<-exp(-lscale)
G[m+2,3:(m+2)]<-para[-1]*exp(-lscale)
G
```

```
##           [,1]      [,2]      [,3]      [,4]
## [1,] -0.0009074821  0.00000000  0.00000000  0.00000000
## [2,]  0.0000000000  0.00000000 -0.98434115  0.00000000
## [3,]  0.0000000000  0.00000000  0.00000000 -0.9843412
## [4,]  0.0064433366 -0.9843412 -0.01802253  0.5042111
```

```
PrVar<-t(G)%*%V%*%G
PrVar
```

```
##           [,1]      [,2]      [,3]      [,4]
## [1,]  4.628984e-07 -5.302429e-05 -3.508693e-06 -1.469436e-05
## [2,] -5.302429e-05  8.673299e-03  6.901004e-05 -2.712620e-03
## [3,] -3.508693e-06  6.901004e-05  8.033933e-05  2.453139e-04
## [4,] -1.469436e-05 -2.712620e-03  2.453139e-04  5.303985e-02
```

```
PrStd<-sqrt(diag(PrVar))
PrStd
```

```
## [1] 0.0006803664 0.0931305471 0.0089632209 0.2303038312
```

```
PrChisq<-c(" ", " ", (x[3:(m+2)]/PrStd[3:(m+2)])^2)
PrPvalue<-c(" ", " ", pchisq((x[3:(m+2)]/PrStd[3:(m+2)])^2,1,lower.tail=F))
out<-data.frame(x,PrStd,PrChisq,PrPvalue)
names(out)<-c("Estimate", "StdError", "Chisq", "P-value")
out
```

```
##           Estimate      StdError          Chisq          P-value
## lambda  0.0009219183 0.0006803664
## alpha   0.9843411517 0.0931305471
## Age      0.0180225253 0.0089632209 4.04299391246284 0.044355066728441
## TreatI -0.5042111027 0.2303038312 4.79316615145063 0.0285728583696877
```

```
#3.4
```

```
pred.data=tibble(Age=44,Treat=as.factor("I"))

#mean response
predict(fit3.1,pred.data,type = "response")
```

```
##           1
## 904.0291
```

```
#median response
predict(fit3.1,pred.data,type = "quantile",p=.5)
```

```
##           1
## 622.9823
```

```
days.aft=predict(fit3.1,pred.data,type = "quantile",p=seq(0,1,length=1000))

upperbound=rep(0,1000)
lowerbound=rep(0,1000)
seq_0_1_1000=seq(0.00001,.999999,length=1000)
for(i in 1:1000){
  upperbound[i]=add_quantile(pred.data,fit3.1,p=seq_0_1_1000[i])[1,5:6][[2]]
  lowerbound[i]=add_quantile(pred.data,fit3.1,p=seq_0_1_1000[i])[1,5:6][[1]]
}
```

```
## Warning: 'as_data_frame()' is deprecated as of tibble 2.0.0.
## Please use 'as_tibble()' instead.
## The signature and semantics have changed, see '?as_tibble'.
## This warning is displayed once every 8 hours.
## Call 'lifecycle::last_warnings()' to see where this warning was generated.
```



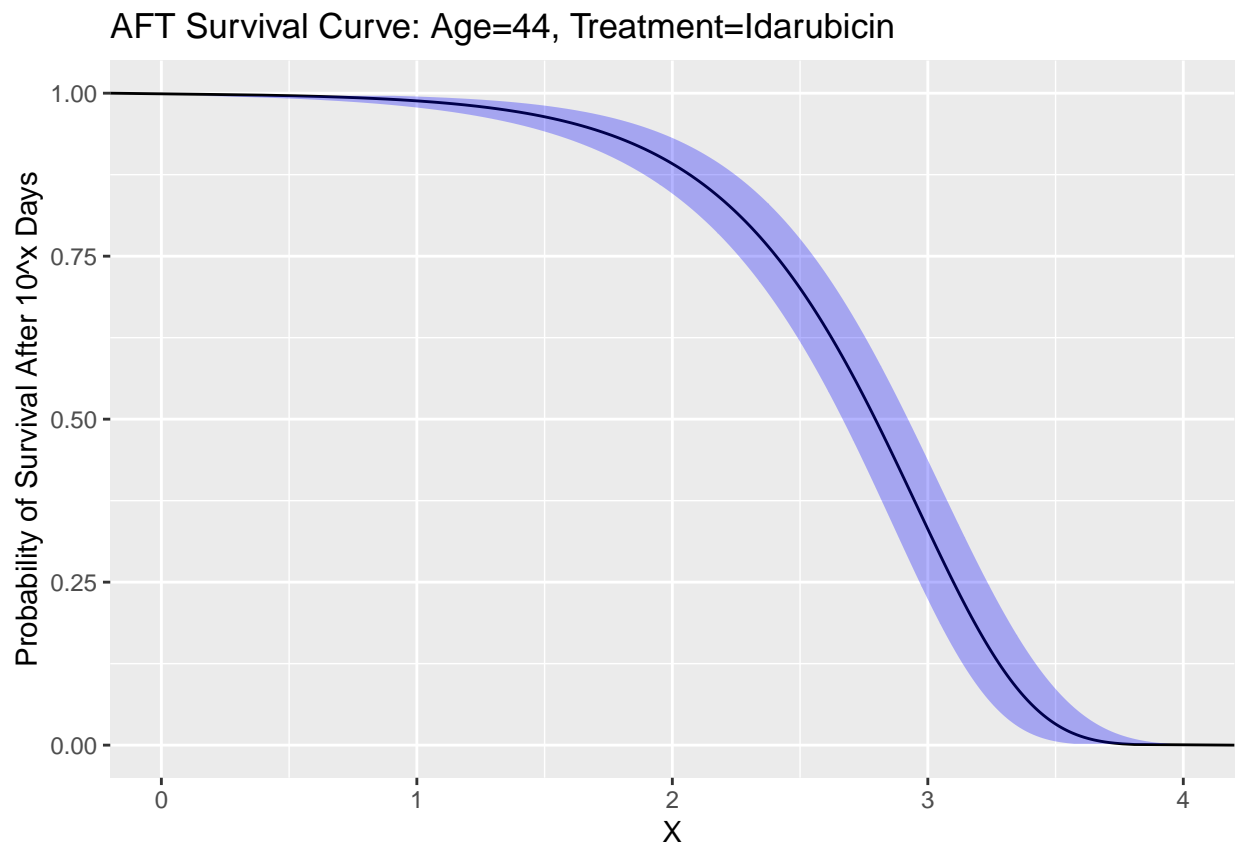
```

aft.preds=tibble(days=days.aft)%>%
  mutate(prob=seq(0,1,length=1000))%>%
  mutate(ul=upperbound)%>%
  mutate(ll=lowerbound)

ggplot(aft.preds,aes(x=log10(days),y=1-prob))+
  geom_line()+
  ggtitle("AFT Survival Curve: Age=44, Treatment=Idarubicin")+
  xlab("X")+
  ylab("Probability of Survival After 10^x Days")+
  geom_ribbon(data=aft.preds,aes(xmin=log10(ll),xmax=log10(ul)),alpha=0.3,fill="blue")+
  xlim(c(0,4))

```

Warning: Removed 1 row(s) containing missing values (geom_path).



```

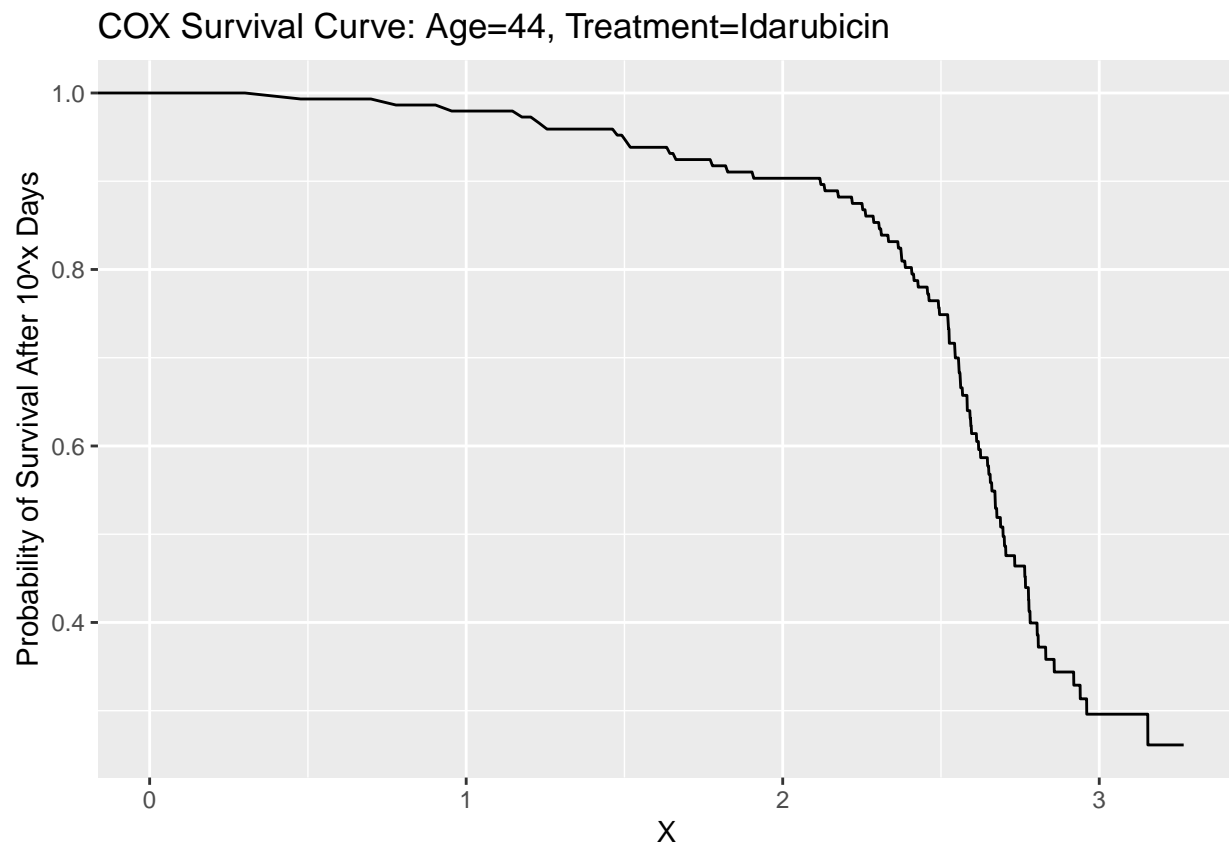
cox2.2=coxph(km~Age+Treat,data=leukem.aml,x=TRUE)
cox.pred=t(predictSurvProb(cox2.2,pred.data,times=0:1848))

cox.tibble=tibble(prob=(cox.pred))%>%
  mutate(days=0:1848)

ggplot(cox.tibble,aes(x=log10(days),y=prob))+
  geom_line()+
  ggtitle("COX Survival Curve: Age=44, Treatment=Idarubicin")+

```

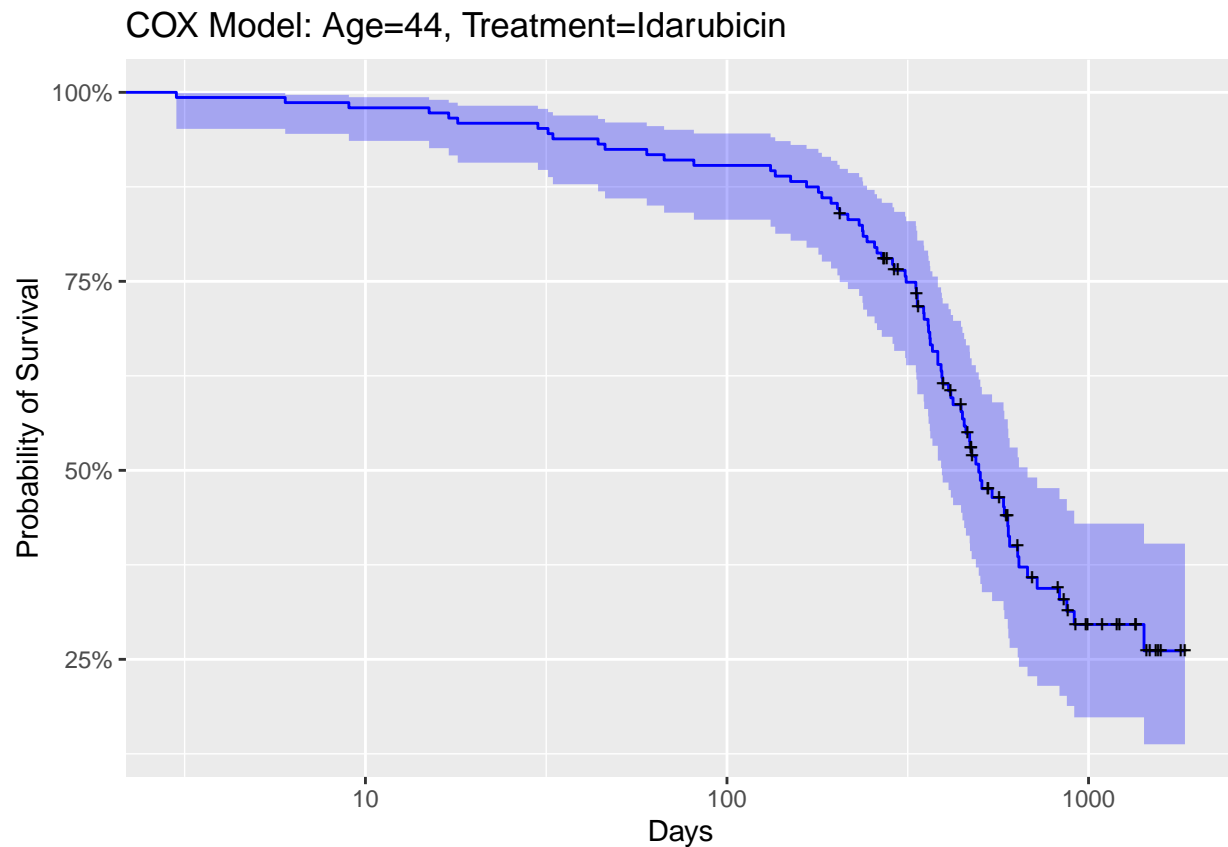
```
xlab("X")+
ylab("Probability of Survival After 10^x Days")
```



```
cox.pred.nolog=survfit(cox2.2,pred.data, conf.type = "log-log")
autoplot(cox.pred.nolog,surv.colour = "blue")+
  scale_x_log10()+
  ggtitle("COX Model: Age=44, Treatment=Idarubicin")+
  xlab("Days")+
  ylab("Probability of Survival")
```

```
## Warning: Transformation introduced infinite values in continuous x-axis
```

```
## Warning: Transformation introduced infinite values in continuous x-axis
```



```
aft.preds[500,1:4]
```

```
## # A tibble: 1 x 4
##   days  prob    ul    ll
##   <dbl> <dbl> <dbl> <dbl>
## 1  622.  0.499  835.  463.
```

```
cox.pred.nolog
```

```
## Call: survfit(formula = cox2.2, newdata = pred.data, conf.type = "log-log")
##
##      n  events  median 0.95LCL 0.95UCL
##    120     79   497    393    678
```