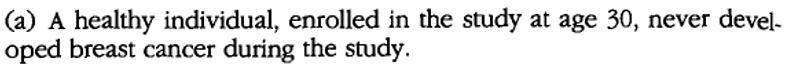
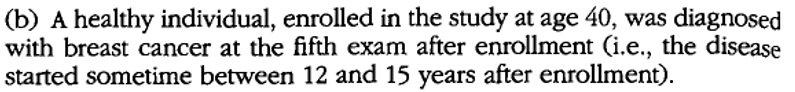
homework1

Question #1a, 3.2

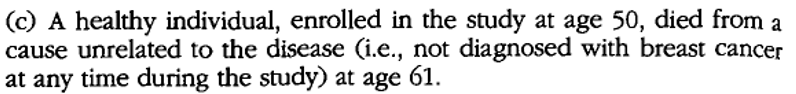
3.2 
A large number of diseae-free individuals were enrolled in a study 
beginning January 1, 1970, and were followed for 30 years to assess 
the age at which they developed breast cancer. Individuals had clinical 
exams every 3 years after enrollment. For four selected individuals 
described below, discuss in detail, the types Of censoring and truncation 
that are represented. 



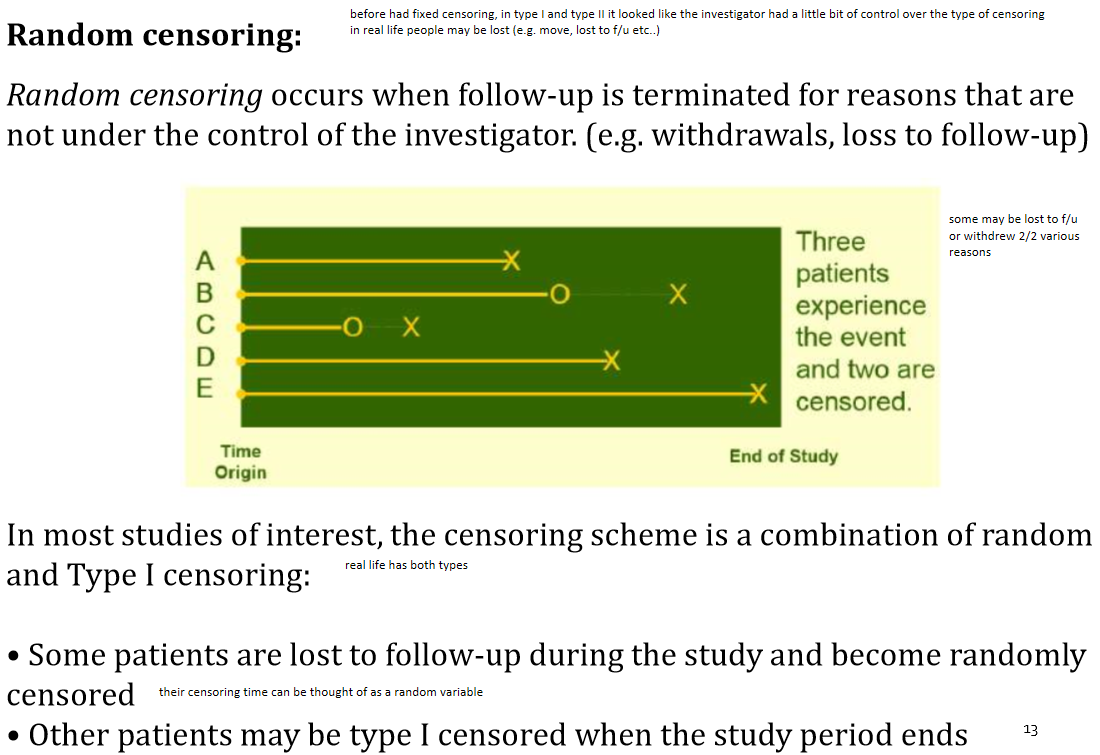
1. type I, right censored since event did not occur prior to some prespecified time

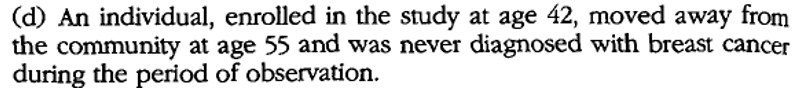


1. interval censoring since event time is only known to occur between some interval follow up times

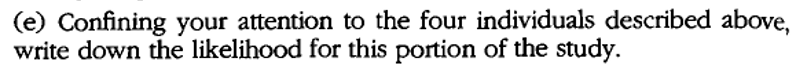


1. Here if we assume the death was unrelated to breast cancer then is likely random censoring since the time at which the patient would have died from breast cancer is unknown



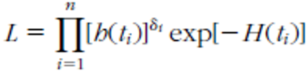


1. This again seems to be random censoring since patient dropped out of study with no apparent reasons related to the disease process of breast cancer.



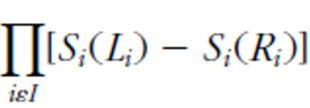
1. Likelihood for four observations would be the product of the four likelihoods using the constructed likelihood function

obs 1; i=1



**[h(t1)]exp[-H(t1)]**

obs 2; i=2



**1-[h(t2)]exp[-H(t2)]-[h(t2)]exp[-H(t2)]**

obs 3; i=3

Textbook section 3.5 example 3.10 seems to suggest that the formula for random censoring is:

Machine generated alternative text:

(35 6) 

which appears nearly identical to that for right censoring

Machine generated alternative text:



however the lecture notes mention that for right censoring we use fi not f

So for random censoring do we use

f(t3) or f3(t) or are they the same

Assuming the same then random censoring seems similar to right censoring for the calculation

**[h(t3)]exp[-H(t3)]**

This is further backed up by lecture notes

Machine generated alternative text:
Does Type of Censoring Affect Survival Methods? 
Standard methods of survival analysis do not distinguish among Type I, 
Type Il, or Random censoring 
theory behind this is explained in pages 75-77 of textbook 
With Type I or Type Il censoring, standard methods are not biased 
The analytic tools we will use assume that, if random censoring occurs, it 
is non-informative or unrelated to the reason for failure. 
make assumption in our class that random censoring may occur but it is not related to the cause of failure, if it does contribute then need more sophisticated form of analysis 

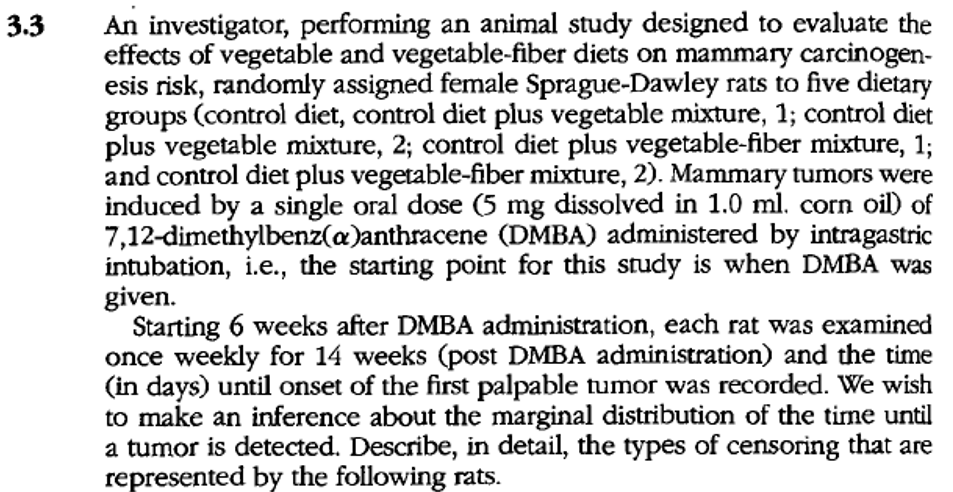
obs 4; i=4

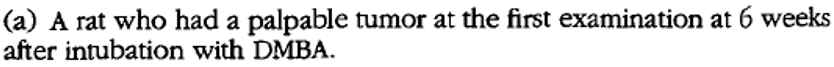
**[h(t4)]exp[-H(t4)]**

**Thus, for e the final likelihood is:**

**[h(t1)]exp[-H(t1)] \* 1-[h(t2)]exp[-H(t2)]-[h(t2)]exp[-H(t2)] \* [h(t3)]exp[-H(t3)] \* [h(t4)]exp[-H(t4)]**

Question #1b, 3.3



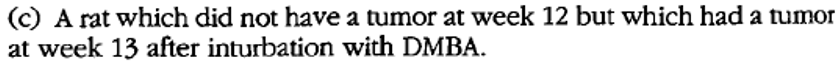


Left censoring since to have a palpable tumor carcinogenesis must have been occurring before the DBMA was administered.

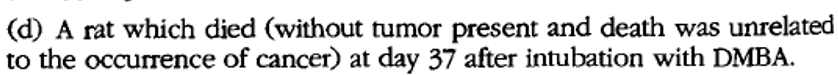
The definition of left censoring is that the event of interest occurred before the study started but it is unknown when the event occurred



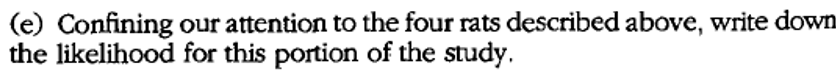
Right censored since we only know that the subject did not have the event during the observation period



Interval censoring since the event occurred between follow-up time points but it is not clear at what point between 12 and 13 weeks post DMBA administration when the tumor development occurred



Random censoring since the death of the subject is not though to be related to the disease process of tumor formation. The time that the subject would have developed the tumor is unknown

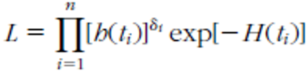


obs 1; i=1

(CD)tS—T) 

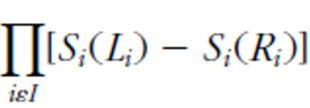
**1-[h(t1)]exp[-H(t1)]**

obs 2; i=2



**[h(t2)]exp[-H(t2)]**

obs 3; i=3



**1-[h(t3)]exp[-H(t3)]-[h(t3)]exp[-H(t3)]**

obs 4; i=4

As mentioned before, the textbook section 3.5 example 3.10 seems to suggest that the formula for random censoring is

Machine generated alternative text:

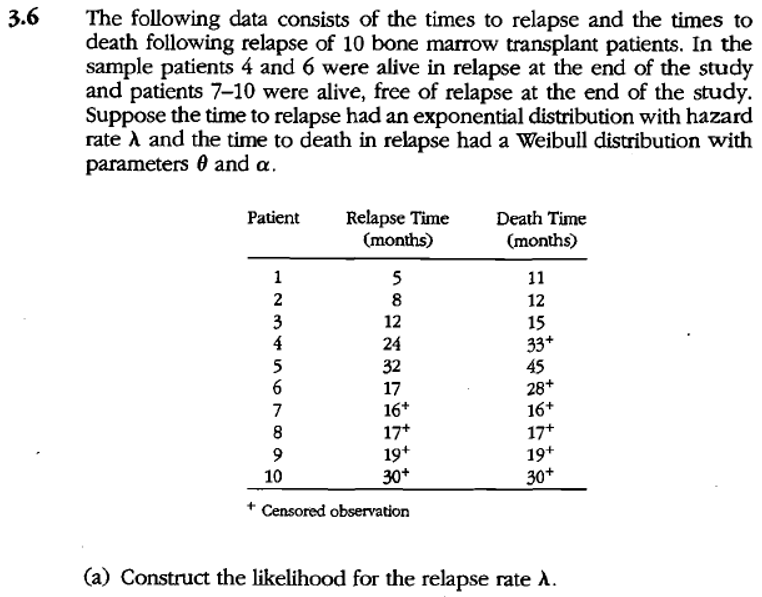
(35 6) 

**[h(t4)]exp[-H(t4)]**

**Thus for e the final likelihood is:**

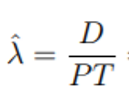
**1-[h(t1)]exp[-H(t1)]\*[h(t2)]exp[-H(t2)]\* 1-[h(t3)]exp[-H(t3)]-[h(t3)]exp[-H(t3)]\*[h(t4)]exp[-H(t4)]**

Question #1c, 3.6 a



D= # that relapsed during study period = 6   
PT = total time in months = 180

Likelihood for relapse rate



> sum(c(5,8,12,24,32,17,16,17,19,30))

[1] 180

**Thus lambda\_hat = D/PT = 6/180=0.0333/month**

Question #1d

Give examples of right censored, left censored, interval-censored, left-truncated data from your field of study. These examples should not be from the Klein & Moeschberger textbook or from the video lectures.

Agriculture is an industry which I grew up in and remain interested in so am providing examples from this context

**Right censored**

An example of right censored observations in agriculture could be time to hybrid seed germination where at the end of the study some seeds with specific hybrid genotypes have not germinated.

**Left censored**

Study at what age calves wean from their mothers with a bull present in the herd but find that at time of study initiation some calves have already weaned from their mother but don’t know when.

**Left truncated**

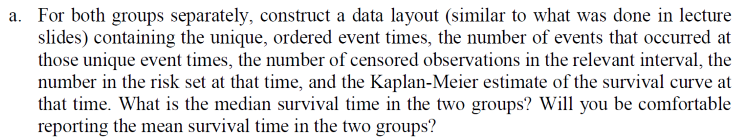
Study at what age vaccinated animals become infected where animals randomly enter the study at different ages and times.

**Interval censured**

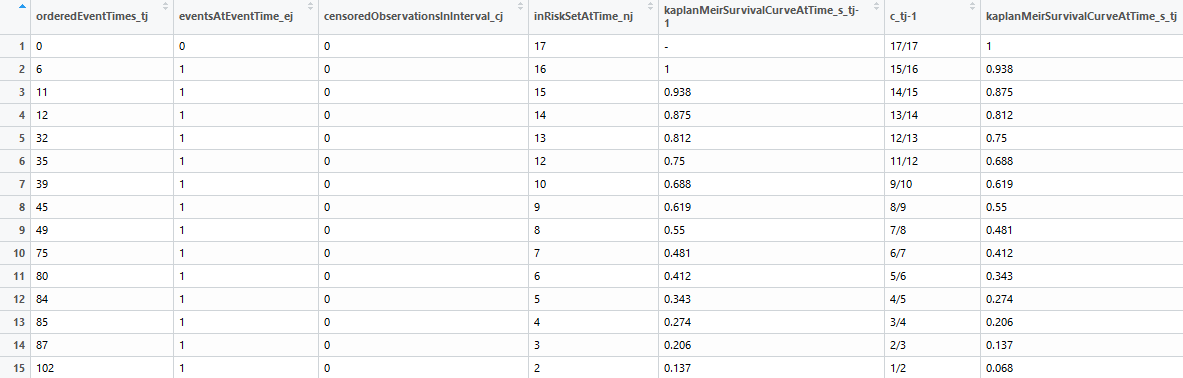
Visit orchards every two weeks to study at what age trees first produce fruit and find trees which produced between previous and current visit but do not know exactly when produced between those timepoints.

**Question #2**

Researchers wish to explore the efficacy of triple-drug combinations of antiretroviral therapy for 
treatment of HIV-infected patients. Because of limitations on potency and the continuing 
emergence of drug resistance seen with the use of currently available antiretroviral agents in 
monotherapy and two-drug regimens, triple combination regimens should represent a more 
promising approach to maximize antiviral activity, maintain long-tenn efficacy, and reduce the 
incidence of drug resistance. Towards this end, investigators performed a randomized study 
comparing AZT + zalcitalbine (ddC) versus AZT + zalcitabine (ddC) + saquinavir. The data, time 
from administration of treatment (in days) until the CD4 count reached a pre-specified level, is 
given below for the two groups: 
AZT + zalcitabine (ddC): 
4+, 6, 11, 12, 32, 35, 38+, 39, 45, 49, 75, 80, 84, 85, 87, 102, 180+ 
AZT + zalcitabine (ddC) + saquinivir: 
2, 3, 4, 12, 22, 48, 51+, 56+, 80, 85, 90, 94+, 160, 171, 180, 180+, 238 



1. **KM table using custom function in R for azt\_ddc group**



**b) KM table using built in function in R for azt\_ddc group**

## Call: survfit(formula = Surv(as.numeric(sub("+", "", azt\_ddc, fixed = TRUE)),   
## ifelse(grepl("+", azt\_ddc, fixed = TRUE), 0, 1)) ~ 1, conf.type = "log-log")  
##   
## time n.risk n.event survival std.err lower 95% CI upper 95% CI  
## 6 16 1 0.9375 0.0605 0.63235 0.991  
## 11 15 1 0.8750 0.0827 0.58598 0.967  
## 12 14 1 0.8125 0.0976 0.52460 0.935  
## 32 13 1 0.7500 0.1083 0.46343 0.898  
## 35 12 1 0.6875 0.1159 0.40460 0.856  
## 39 10 1 0.6188 0.1230 0.33929 0.808  
## 45 9 1 0.5500 0.1271 0.27933 0.756  
## 49 8 1 0.4813 0.1285 0.22410 0.699  
## 75 7 1 0.4125 0.1272 0.17339 0.639  
## 80 6 1 0.3438 0.1232 0.12728 0.575  
## 84 5 1 0.2750 0.1162 0.08617 0.507  
## 85 4 1 0.2063 0.1055 0.05082 0.433  
## 87 3 1 0.1375 0.0900 0.02265 0.354  
## 102 2 1 0.0688 0.0662 0.00443 0.267

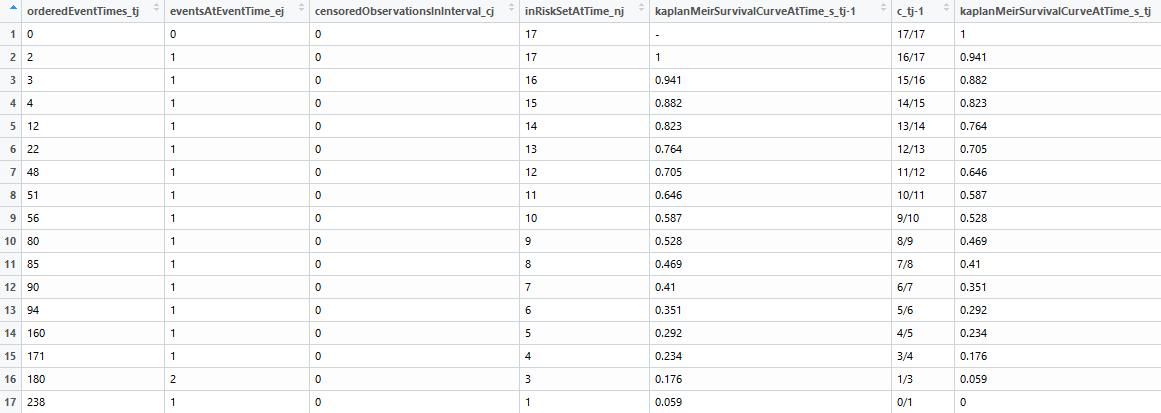
Call: survfit(formula = Surv(as.numeric(sub("+", "", azt\_ddc, fixed = TRUE)),   
 ifelse(grepl("+", azt\_ddc, fixed = TRUE), 0, 1)) ~ 1, conf.type = "log-log")

n events median 0.95LCL 0.95UCL   
 17 14 **49** 32 85

**median of 49**

The median survival time is defined as the time at t such that S(t) = 1/2. For AZT + DDC it is **49.**

**a) KM table using custom function in R for azt\_ddc\_saq group**

****

**b) KM table using built in function in R for azt\_ddc\_saq**

## time n.risk n.event survival std.err lower 95% CI upper 95% CI  
## 2 17 1 0.9412 0.0571 0.65018 0.991  
## 3 16 1 0.8824 0.0781 0.60598 0.969  
## 4 15 1 0.8235 0.0925 0.54713 0.939  
## 12 14 1 0.7647 0.1029 0.48828 0.904  
## 22 13 1 0.7059 0.1105 0.43148 0.866  
## 48 12 1 0.6471 0.1159 0.37715 0.823  
## 51 11 1 0.5882 0.1194 0.32537 0.778  
## 56 10 1 0.5294 0.1211 0.27617 0.730  
## 80 9 1 0.4706 0.1211 0.22960 0.680  
## 85 8 1 0.4118 0.1194 0.18576 0.626  
## 90 7 1 0.3529 0.1159 0.14483 0.570  
## 94 6 1 0.2941 0.1105 0.10712 0.511  
## 160 5 1 0.2353 0.1029 0.07308 0.449  
## 171 4 1 0.1765 0.0925 0.04348 0.383  
## 180 3 2 0.0588 0.0571 0.00391 0.235  
## 238 1 1 0.0000 NaN NA NA

n events median 0.95LCL 0.95UCL   
 17 17 **80** 12 160

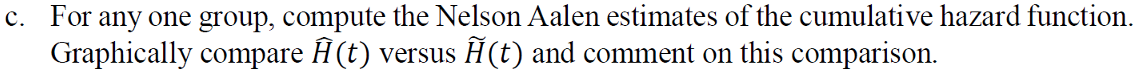
**median of 80**

**If the survival curve is not continuous at 1/2 (if the survival function is a step function, for example), then the median is taken to be the smallest t such that S(t)<=1/2. Thus for AZT + DDC + SAQ the median survival would be 80.**

The **mean** survival time is only defined if the survival curve goes to zero that is S(inf)=0. Thus, if the last subject is in the study is censored then the survival function will not go to 0 and the area under the curve cannot be calculated. In other words, and in theory, the mean survival cannot be computed when the Kaplan-Meier survival curve does not reach zero. Therefore, I would not feel comfortable reporting the mean for the azt\_ddc group since the last observation is censored. I would feel more comfortable reporting the mean survival time in the azt\_ddc\_saq group since the last observation is not censored and the survival function does go to zero.

Machine generated alternative text:
azt ddc KM survival curve 
100 
Time 
150 

Machine generated alternative text:
azt ddc_saq_KM survival curve 
100 
Time 
150 
200 



**cumulative hazard times using Nelson-Aalen technique for azt\_ddc group**

|  |  |  |  |
| --- | --- | --- | --- |
| orderedEventTimes\_tj | eventsAtEventTime\_ej | inRiskSetAtTime\_nj | cumulativeHazardRate\_ht |
| 0 | 0 | 17 | 0 |
| 6 | 1 | 16 | 0.062 |
| 11 | 1 | 15 | 0.129 |
| 12 | 1 | 14 | 0.2 |
| 32 | 1 | 13 | 0.277 |
| 35 | 1 | 12 | 0.36 |
| 39 | 1 | 10 | 0.46 |
| 45 | 1 | 9 | 0.571 |
| 49 | 1 | 8 | 0.696 |
| 75 | 1 | 7 | 0.839 |
| 80 | 1 | 6 | 1.006 |
| 84 | 1 | 5 | 1.206 |
| 85 | 1 | 4 | 1.456 |
| 87 | 1 | 3 | 1.789 |
| 102 | 1 | 2 | 2.289 |

azt ddc NA vs. KM cumulative hazard curves 
100 
Time 
150 

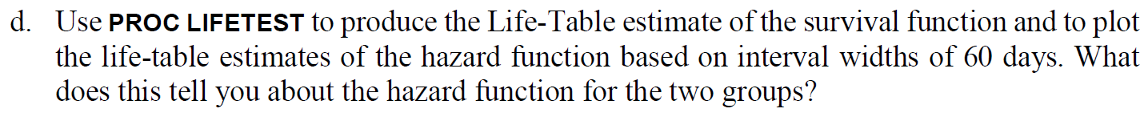
**As described in the lecture and text the Nelson-Aalen technique underestimates hazard and overestimates survival compared to the Kaplan-Meier technique**

**Cumulative hazard times using Nelson-Aalen technique for azt\_ddc\_saq group**

|  |  |  |  |
| --- | --- | --- | --- |
| orderedEventTimes\_tj | eventsAtEventTime\_ej | inRiskSetAtTime\_nj | cumulativeHazardRate\_ht |
| 2 | 1 | 17 | 0.059 |
| 3 | 1 | 16 | 0.122 |
| 4 | 1 | 15 | 0.189 |
| 12 | 1 | 14 | 0.26 |
| 22 | 1 | 13 | 0.337 |
| 48 | 1 | 12 | 0.42 |
| 51 | 1 | 11 | 0.511 |
| 56 | 1 | 10 | 0.611 |
| 80 | 1 | 9 | 0.722 |
| 85 | 1 | 8 | 0.847 |
| 90 | 1 | 7 | 0.99 |
| 94 | 1 | 6 | 1.157 |
| 160 | 1 | 5 | 1.357 |
| 171 | 1 | 4 | 1.607 |
| 180 | 2 | 3 | 2.274 |
| 238 | 1 | 1 | 3.274 |

azt_ddc_saq NA vs. KM cumulative hazard curves 
100 
Time 
150 

**Again as described in the lecture and text, the Nelson-Aalen technique underestimates hazard and overestimates survival compared to the Kaplan-Meier technique**



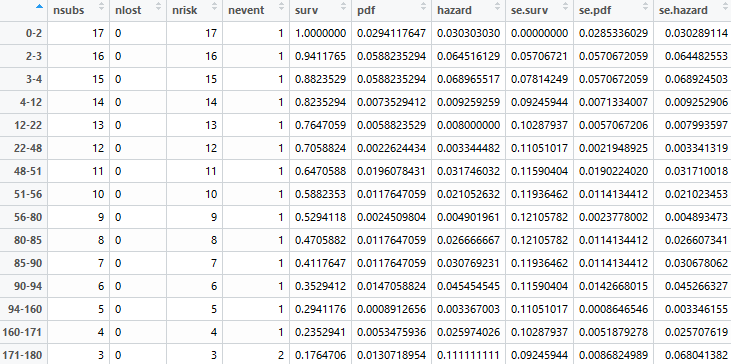
Used R function *“lifetab”* to create table and graphs below instead of PROC LIFETEST

**life-table azt\_ddc**

Machine generated alternative text:
6-11 
11-12 
12-32 
45-49 
49-75 
87-102 
16.5 
16.0 
15.0 
14.0 
13.0 
12.0 
10.5 
sur v 
00000 
00000 
0.93750 
0.87500 
0.81250 
0.75000 
0.68750 
0.68750 
0.61875 
0.55000 
0.48125 
0.41250 
0.34375 
0.27500 
0.20625 
0.13750 
pdf 
0.000000000 
0.031250000 
0.012500000 
0.062500000 
0.003125000 
0.020833333 
0.000000000 
0.068750000 
0.011458333 
0.017187500 
0.002644231 
0.013750000 
0.017187500 
0.068750000 
0.034375000 
0.004583333 
0.00000000 
0.03225806 
0.01379310 
0.07407407 
0.00400000 
0.02898551 
0.00000000 
0.10526316 
0.01960784 
0.03333333 
0.00591716 
0.03636364 
0.05555556 
0.28571429 
0.20000000 
O 04444444 

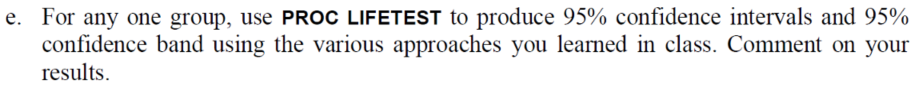
azt_ddc Life Table Hazard Function in 60 day windows 
120 
180 

**life-table azt\_ddc\_saq**



azt_ddc_saq Life Table Hazard Function in 60 day windows 
120 
180 
238 

**The life table plots demonstrate that the addition of the anti-retroviral medication Saquinavir reduces the hazard function of death when compared to the group with only two medications azt and ddc**

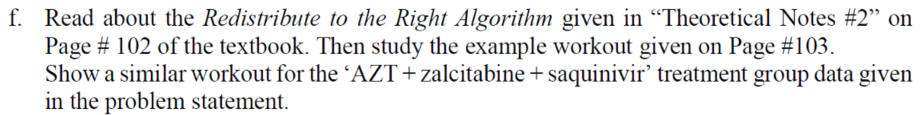


Used R functions *“survfit”* and *“confband”* to generate values below

Machine generated alternative text:



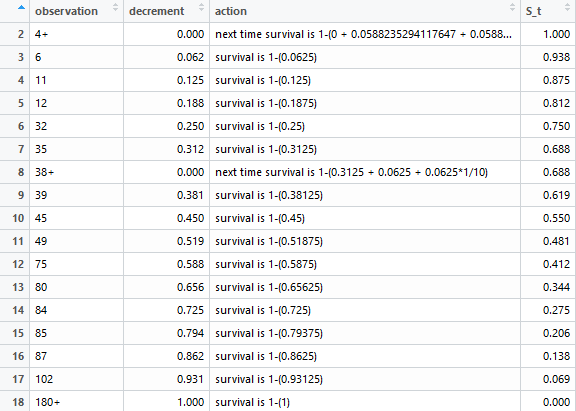
**As expected the lower and upper CIs (confidence intervals) are in general tighter than the CB (confidence band)**



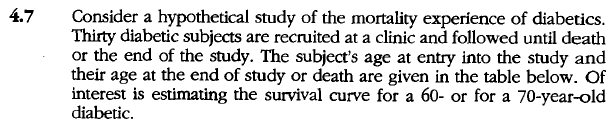
Act\_ddc\_saq

Machine generated alternative text:
10 
12 
13 
14 
15 
16 
17 
observation 
12 
51 
80 
90 
94 
160 
171 
180 
238 
0.059 
0.118 
0.176 
0.235 
0.294 
0.353 
0.412 
0.471 
0.529 
0.588 
0.647 
0.706 
0.765 
0.824 
0.882 
0.941 
survival is 1 -(0.0588235294117647) 
survival is 1-(0.117647058823529) 
survival is 1-(0.176470588235294 
survival is 1-(0.235294117647059) 
survival is 1-(0.294117647058824 
survival is 1-(0.352941176470588) 
survival is 1-(0.411764705882353) 
survival is 1-(0.470588235294118) 
survival is 1-(0.529411764705882) 
survival is 1-(0.588235294117647) 
survival is 1-(0.647058823529412) 
survival is 1-(0.705882352941177) 
survival is 1-(0.764705882352941) 
survival is 1-(0.823529411764706) 
survival is 1-(0.882352941176471) 
survival is 1-(0.941176470588235) 
0.941 
0.882 
0.824 
0.765 
0.706 
0.647 
0.471 
0.412 
0.353 
0.294 
0.235 
0.176 
0.118 
0.059 

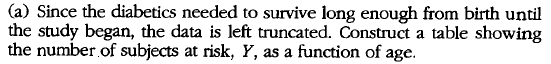
In table above there were no censored values so the survival function was simply reduced by 1/n for each observation (azt\_ddc\_saq)



For completeness this is the output for azt\_ddc without saq



USING LEFT TRUNCATION AND RIGHT CENSORING



Left truncated and right censored approach:

1. Machine generated alternative text:
   10 
   11 
   12 
   13 
   14 
   15 
   tj 
   70 
   71 
   72 
   73 
   74 
   76 
   c_tj- 
   11/13 
   12/14 
   11/13 
   12/14 
   12/14 
   12/13 
   11/12 
   s_tj 
   0.7111 
   0.512 
   0.3342 
   0.2424 
   0.2078 
   0.1918 
   0.1758 
   0.1438 

where nj is the # at risk Y as a function of age tj. For example at age 62 there are 9 at risk



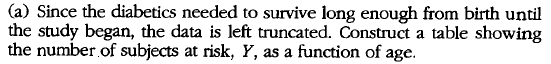
Using above table conditional survival for the age of death for a diabetic patient who has survived to age 60 is 0.8



Using above table conditional survival for the age of death for a diabetic patient who has survived to age 70 is 0.283



USING RIGHT CENSORING WHLE IGNORING LEFT CENSORING



Right censored approach while ignoring left truncation:

1. Machine generated alternative text:

where inRiskSetAtTIme\_nj is the # at risk Y as a function of age tj. For example, at age 63 there are 28 at risk



Using above table conditional survival for the age of death for a diabetic patient who has survived to age 60 is 0.97



Using above table conditional survival for the age of death for a diabetic patient who has survived to age 70 is 0.6

**R CODE AND OUTPUT BELOW**

azt\_ddc=c("4+",6,11,12,32,35,"38+",39,45,49,75,80,84,85,87,102,"180+")  
azt\_ddc\_saq=c(2,3,4,12,22,48,51,56,80,85,90,94,160,171,180,180,238)  
  
kmTable=data.frame()  
getKMTable = function(censoredTimesVector,censorSymbol){  
 #get numeric representation of censor vector  
 censoredTimesVectorNumeric=as.numeric(sub(censorSymbol,'',censoredTimesVector,fixed=TRUE))  
 #count number of actual rows in KM table  
 cnt\_n=length(censoredTimesVectorNumeric)  
 #create first row of KM table  
 kmTable=setNames(data.frame(matrix(nrow=1,c(0,0,0,cnt\_n,as.character("-"),as.character(paste0(cnt\_n,"/",cnt\_n)),1)),stringsAsFactors=FALSE),c("orderedEventTimes\_tj","eventsAtEventTime\_ej",  
 "censoredObservationsInInterval\_cj","inRiskSetAtTime\_nj","kaplanMeirSurvivalCurveAtTime\_s\_tj-1","c\_tj-1","kaplanMeirSurvivalCurveAtTime\_s\_tj"))  
 censoredTimesVectorNumeric=sort(censoredTimesVectorNumeric)  
 for (i in 1:max(censoredTimesVectorNumeric)){  
 if(i %in% censoredTimesVectorNumeric){  
 #create empty row to fill in  
 kmTableRow=setNames(data.frame(matrix(NA,nrow=1,ncol=length(names(kmTable)))),names(kmTable))  
 kmTableRow$orderedEventTimes\_tj=i  
 #count how many events at time  
 kmTableRow$eventsAtEventTime\_ej=length(which(censoredTimesVector==i))  
 #count how many censured at time  
 kmTableRow$censoredObservationsInInterval\_cj=length(which(censoredTimesVector==paste0(i,censorSymbol)))  
 kmTableRow$inRiskSetAtTime\_nj=cnt\_n  
 #sum events and number censored at time  
 loss=kmTableRow$eventsAtEventTime\_ej+kmTableRow$censoredObservationsInInterval\_cj  
 prevSurv=kmTable[dim(kmTable)[1],c("kaplanMeirSurvivalCurveAtTime\_s\_tj")]  
 kmTableRow[c("kaplanMeirSurvivalCurveAtTime\_s\_tj-1")]=prevSurv  
 kmTableRow[c("c\_tj-1")]=paste0((cnt\_n-loss),"/",cnt\_n)  
 #kmTableRow$kaplanMeirSurvivalCurveAtTime\_s\_tj=round((cnt\_n-loss)/length(censoredTimesVectorNumeric),3)  
 kmTableRow$kaplanMeirSurvivalCurveAtTime\_s\_tj=round((cnt\_n-loss)/cnt\_n\*as.numeric(prevSurv),3)  
 #update count  
 cnt\_n=cnt\_n-loss  
 if(kmTableRow$censoredObservationsInInterval\_cj==0){  
 #add row to kmtable  
 kmTable=rbind(kmTable,kmTableRow)  
 }  
 }  
 }  
 kmTable  
}  
azt\_ddc\_KM=getKMTable(azt\_ddc,"+")  
azt\_ddc\_saq\_KM=getKMTable(azt\_ddc\_saq,"+")  
show(azt\_ddc\_KM)

## orderedEventTimes\_tj eventsAtEventTime\_ej  
## 1 0 0  
## 2 6 1  
## 3 11 1  
## 4 12 1  
## 5 32 1  
## 6 35 1  
## 7 39 1  
## 8 45 1  
## 9 49 1  
## 10 75 1  
## 11 80 1  
## 12 84 1  
## 13 85 1  
## 14 87 1  
## 15 102 1  
## censoredObservationsInInterval\_cj inRiskSetAtTime\_nj  
## 1 0 17  
## 2 0 16  
## 3 0 15  
## 4 0 14  
## 5 0 13  
## 6 0 12  
## 7 0 10  
## 8 0 9  
## 9 0 8  
## 10 0 7  
## 11 0 6  
## 12 0 5  
## 13 0 4  
## 14 0 3  
## 15 0 2  
## kaplanMeirSurvivalCurveAtTime\_s\_tj-1 c\_tj-1  
## 1 - 17/17  
## 2 1 15/16  
## 3 0.938 14/15  
## 4 0.875 13/14  
## 5 0.812 12/13  
## 6 0.75 11/12  
## 7 0.688 9/10  
## 8 0.619 8/9  
## 9 0.55 7/8  
## 10 0.481 6/7  
## 11 0.412 5/6  
## 12 0.343 4/5  
## 13 0.274 3/4  
## 14 0.206 2/3  
## 15 0.137 1/2  
## kaplanMeirSurvivalCurveAtTime\_s\_tj  
## 1 1  
## 2 0.938  
## 3 0.875  
## 4 0.812  
## 5 0.75  
## 6 0.688  
## 7 0.619  
## 8 0.55  
## 9 0.481  
## 10 0.412  
## 11 0.343  
## 12 0.274  
## 13 0.206  
## 14 0.137  
## 15 0.068

show(azt\_ddc\_saq\_KM)

## orderedEventTimes\_tj eventsAtEventTime\_ej  
## 1 0 0  
## 2 2 1  
## 3 3 1  
## 4 4 1  
## 5 12 1  
## 6 22 1  
## 7 48 1  
## 8 51 1  
## 9 56 1  
## 10 80 1  
## 11 85 1  
## 12 90 1  
## 13 94 1  
## 14 160 1  
## 15 171 1  
## 16 180 2  
## 17 238 1  
## censoredObservationsInInterval\_cj inRiskSetAtTime\_nj  
## 1 0 17  
## 2 0 17  
## 3 0 16  
## 4 0 15  
## 5 0 14  
## 6 0 13  
## 7 0 12  
## 8 0 11  
## 9 0 10  
## 10 0 9  
## 11 0 8  
## 12 0 7  
## 13 0 6  
## 14 0 5  
## 15 0 4  
## 16 0 3  
## 17 0 1  
## kaplanMeirSurvivalCurveAtTime\_s\_tj-1 c\_tj-1  
## 1 - 17/17  
## 2 1 16/17  
## 3 0.941 15/16  
## 4 0.882 14/15  
## 5 0.823 13/14  
## 6 0.764 12/13  
## 7 0.705 11/12  
## 8 0.646 10/11  
## 9 0.587 9/10  
## 10 0.528 8/9  
## 11 0.469 7/8  
## 12 0.41 6/7  
## 13 0.351 5/6  
## 14 0.292 4/5  
## 15 0.234 3/4  
## 16 0.176 1/3  
## 17 0.059 0/1  
## kaplanMeirSurvivalCurveAtTime\_s\_tj  
## 1 1  
## 2 0.941  
## 3 0.882  
## 4 0.823  
## 5 0.764  
## 6 0.705  
## 7 0.646  
## 8 0.587  
## 9 0.528  
## 10 0.469  
## 11 0.41  
## 12 0.351  
## 13 0.292  
## 14 0.234  
## 15 0.176  
## 16 0.059  
## 17 0

library(survival)

## Warning: package 'survival' was built under R version 3.5.2

#numeric times and censor list (0 for not censored 1 for censored)  
Surv(as.numeric(sub("+","",azt\_ddc,fixed=TRUE)),ifelse(grepl("+",azt\_ddc,fixed=TRUE),0,1))

## [1] 4+ 6 11 12 32 35 38+ 39 45 49 75 80 84 85   
## [15] 87 102 180+

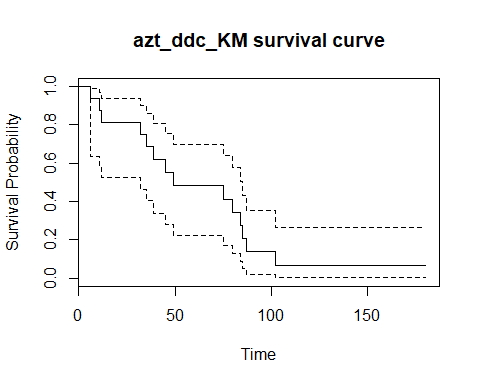
azt\_ddc\_KM\_R=survfit(Surv(as.numeric(sub("+","",azt\_ddc,fixed=TRUE)),ifelse(grepl("+",azt\_ddc,fixed=TRUE),0,1))~1,conf.type="log-log")  
summary(azt\_ddc\_KM\_R)

## Call: survfit(formula = Surv(as.numeric(sub("+", "", azt\_ddc, fixed = TRUE)),   
## ifelse(grepl("+", azt\_ddc, fixed = TRUE), 0, 1)) ~ 1, conf.type = "log-log")  
##   
## time n.risk n.event survival std.err lower 95% CI upper 95% CI  
## 6 16 1 0.9375 0.0605 0.63235 0.991  
## 11 15 1 0.8750 0.0827 0.58598 0.967  
## 12 14 1 0.8125 0.0976 0.52460 0.935  
## 32 13 1 0.7500 0.1083 0.46343 0.898  
## 35 12 1 0.6875 0.1159 0.40460 0.856  
## 39 10 1 0.6188 0.1230 0.33929 0.808  
## 45 9 1 0.5500 0.1271 0.27933 0.756  
## 49 8 1 0.4813 0.1285 0.22410 0.699  
## 75 7 1 0.4125 0.1272 0.17339 0.639  
## 80 6 1 0.3438 0.1232 0.12728 0.575  
## 84 5 1 0.2750 0.1162 0.08617 0.507  
## 85 4 1 0.2063 0.1055 0.05082 0.433  
## 87 3 1 0.1375 0.0900 0.02265 0.354  
## 102 2 1 0.0688 0.0662 0.00443 0.267

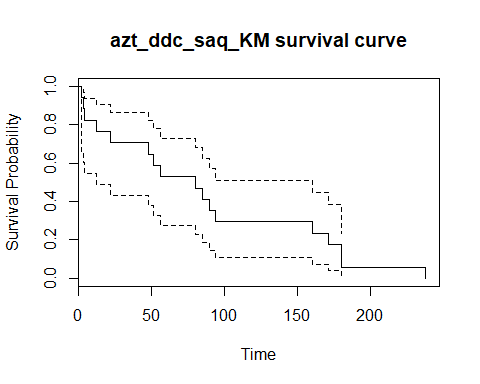
azt\_ddc\_saq\_KM\_R = survfit(Surv(as.numeric(sub("+","",azt\_ddc\_saq,fixed=TRUE)),ifelse(grepl("+",azt\_ddc\_saq,fixed=TRUE),0,1))~1,conf.type="log-log")  
summary(azt\_ddc\_saq\_KM\_R)

## Call: survfit(formula = Surv(as.numeric(sub("+", "", azt\_ddc\_saq, fixed = TRUE)),   
## ifelse(grepl("+", azt\_ddc\_saq, fixed = TRUE), 0, 1)) ~ 1,   
## conf.type = "log-log")  
##   
## time n.risk n.event survival std.err lower 95% CI upper 95% CI  
## 2 17 1 0.9412 0.0571 0.65018 0.991  
## 3 16 1 0.8824 0.0781 0.60598 0.969  
## 4 15 1 0.8235 0.0925 0.54713 0.939  
## 12 14 1 0.7647 0.1029 0.48828 0.904  
## 22 13 1 0.7059 0.1105 0.43148 0.866  
## 48 12 1 0.6471 0.1159 0.37715 0.823  
## 51 11 1 0.5882 0.1194 0.32537 0.778  
## 56 10 1 0.5294 0.1211 0.27617 0.730  
## 80 9 1 0.4706 0.1211 0.22960 0.680  
## 85 8 1 0.4118 0.1194 0.18576 0.626  
## 90 7 1 0.3529 0.1159 0.14483 0.570  
## 94 6 1 0.2941 0.1105 0.10712 0.511  
## 160 5 1 0.2353 0.1029 0.07308 0.449  
## 171 4 1 0.1765 0.0925 0.04348 0.383  
## 180 3 2 0.0588 0.0571 0.00391 0.235  
## 238 1 1 0.0000 NaN NA NA

plot(azt\_ddc\_KM\_R,xlab="Time",ylab="Survival Probability",main="azt\_ddc\_KM survival curve")



plot(azt\_ddc\_saq\_KM\_R,xlab="Time",ylab="Survival Probability",main="azt\_ddc\_saq\_KM survival curve")

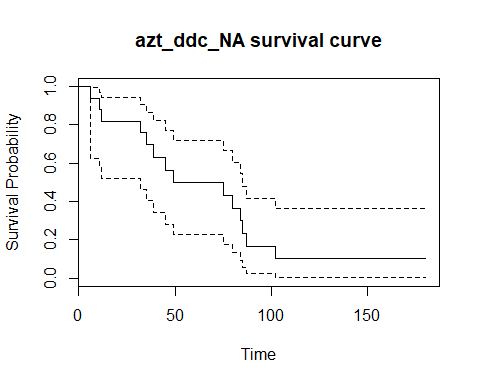


For both groups separately, construct a data layout (similar to what was done in lecture slides) containing the unique, ordered event times, the number of events that occurred at those unique event times, the number of censored observations in the relevant interval, the number in the risk set at that time, and the Kaplan-Meier estimate of the survival curve at that time. What is the median survival time in the two groups? Will you be comfortable reporting the mean survival time in the two groups?

#6MP as test case  
leukemia\_6MP = c(10,7,"32+",23,22,6,16,"34+","32+","25+","11+","20+","19+",6,"17+","35+",6,13,"9+","6+","10+")  
getNATable = function(censoredTimesVector,censorSymbol){  
 #get numeric representation of censor vector  
 censoredTimesVectorNumeric=as.numeric(sub(censorSymbol,'',censoredTimesVector,fixed=TRUE))  
 #count number of actual rows in NA table  
 cnt\_n=length(censoredTimesVectorNumeric)  
 #create first row of NA table  
 naTable=setNames(data.frame(matrix(nrow=1,c(0,0,cnt\_n,0,0,as.character(paste0(cnt\_n,"/",cnt\_n)),0,0)),stringsAsFactors=FALSE),c("orderedEventTimes\_tj","eventsAtEventTime\_ej","inRiskSetAtTime\_nj","censoredObservationsInInterval\_cj","cumulativeHazardRate\_ht","d\_Y\_ratio","cumulativeHazardEstimatedVariance\_vt","nelsonAalenSurvivalCurveAtTime\_s\_tj"))  
 censoredTimesVectorNumeric=sort(censoredTimesVectorNumeric)  
 sumCensoredInInterval=0  
 for (i in 1:max(censoredTimesVectorNumeric)){  
 if(i %in% censoredTimesVectorNumeric){  
 #create empty row to fill in  
 naTableRow=setNames(data.frame(matrix(NA,nrow=1,ncol=length(names(naTable)))),names(naTable))  
 naTableRow$orderedEventTimes\_tj=i  
 #count how many events at time  
 naTableRow$eventsAtEventTime\_ej=length(which(censoredTimesVector==i))  
 #running total of censured between censured time intervals  
 naTableRow$censoredObservationsInInterval\_cj=length(which(censoredTimesVector==paste0(i,censorSymbol)))  
 sumCensoredInInterval=sumCensoredInInterval+naTableRow$censoredObservationsInInterval\_cj  
 naTableRow$inRiskSetAtTime\_nj=cnt\_n  
 naTableRow$d\_Y\_ratio=paste0((naTableRow$eventsAtEventTime\_ej),"/",cnt\_n)  
 naTableRow$cumulativeHazardRate\_ht=round(as.numeric(naTable[dim(naTable)[1],c("cumulativeHazardRate\_ht")])+naTableRow$eventsAtEventTime\_ej/cnt\_n,3)  
 naTableRow$cumulativeHazardEstimatedVariance\_vt=round(as.numeric(naTable[dim(naTable)[1],c("cumulativeHazardEstimatedVariance\_vt")])+naTableRow$eventsAtEventTime\_ej/(cnt\_n)^2,3)  
 naTableRow$nelsonAalenSurvivalCurveAtTime\_s\_tj=round(exp(-naTableRow$cumulativeHazardRate\_ht),3)  
 #sum events and number censored at time  
 loss=naTableRow$eventsAtEventTime\_ej+naTableRow$censoredObservationsInInterval\_cj  
 #update count  
 cnt\_n=cnt\_n-loss  
 #add row to na table if at least one uncensored variable  
 if (i %in% censoredTimesVector){  
 naTableRow$censoredObservationsInInterval\_cj=sumCensoredInInterval  
 naTable=rbind(naTable,naTableRow)  
 sumCensoredInInterval=0  
 }  
 }  
 }  
 naTable  
}  
  
#adapted from http://sas-and-r.blogspot.com/2010/05/example-739-nelson-aalen-estimate-of.html  
getCumulativeHazardNA = function(time, event) {  
 na.fit = survfit(coxph(Surv(time,event)~1), type="aalen")  
 jumps = c(0, na.fit$time, max(time))  
 # need to be careful at the beginning and end  
 surv = c(1, na.fit$surv, na.fit$surv[length(na.fit$surv)])  
 # apply appropriate transformation  
 neglogsurv = -log(surv)   
 # create placeholder of correct length  
 naest = numeric(length(time))   
 for (i in 2:length(jumps)) {  
 naest[which(time>=jumps[i-1] & time<=jumps[i])] =   
 neglogsurv[i-1] # snag the appropriate value  
 }  
 return(sort(unique(naest)))  
}  
  
#TEST SET VALIDATED BY TABLE 4.2 in text page 95  
# leukemia\_6MP\_NA=getNATable(leukemia\_6MP,"+")  
# leukemia\_6MP\_NA\_R=survfit(Surv(as.numeric(sub("+","",leukemia\_6MP,fixed=TRUE)),ifelse(grepl("+",leukemia\_6MP,fixed=TRUE),0,1))~1,conf.type="log-log",type="fh")  
# summary(leukemia\_6MP\_NA\_R)  
# plot(leukemia\_6MP\_NA\_R,xlab="Time",ylab="Survival Probability",main="leukemia\_6MP\_NA\_R survival curve")  
# leukemia\_6MP\_NA\_CH\_R=getCumulativeHazardNA(as.numeric(sub("+","",leukemia\_6MP,fixed=TRUE)),ifelse(grepl("+",leukemia\_6MP,fixed=TRUE),0,1))  
  
azt\_ddc\_NA=getNATable(azt\_ddc,"+")  
azt\_ddc\_NA\_R=survfit(Surv(as.numeric(sub("+","",azt\_ddc,fixed=TRUE)),ifelse(grepl("+",azt\_ddc,fixed=TRUE),0,1))~1,conf.type="log-log",type="fh")  
summary(azt\_ddc\_NA\_R)

## Call: survfit(formula = Surv(as.numeric(sub("+", "", azt\_ddc, fixed = TRUE)),   
## ifelse(grepl("+", azt\_ddc, fixed = TRUE), 0, 1)) ~ 1, conf.type = "log-log",   
## type = "fh")  
##   
## time n.risk n.event survival std.err lower 95% CI upper 95% CI  
## 6 16 1 0.939 0.0606 0.62304 0.992  
## 11 15 1 0.879 0.0830 0.58170 0.970  
## 12 14 1 0.818 0.0983 0.52281 0.940  
## 32 13 1 0.758 0.1094 0.46341 0.905  
## 35 12 1 0.697 0.1175 0.40600 0.865  
## 39 10 1 0.631 0.1254 0.34186 0.820  
## 45 9 1 0.564 0.1304 0.28289 0.772  
## 49 8 1 0.498 0.1330 0.22843 0.720  
## 75 7 1 0.432 0.1331 0.17823 0.664  
## 80 6 1 0.366 0.1310 0.13234 0.606  
## 84 5 1 0.299 0.1264 0.09106 0.545  
## 85 4 1 0.233 0.1192 0.05506 0.481  
## 87 3 1 0.167 0.1093 0.02560 0.417  
## 102 2 1 0.101 0.0976 0.00539 0.367

plot(azt\_ddc\_NA\_R,xlab="Time",ylab="Survival Probability",main="azt\_ddc\_NA survival curve")



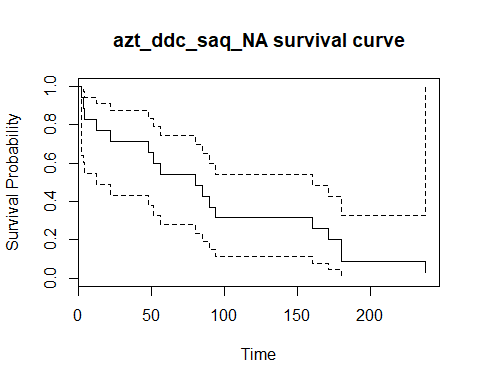
#cumulative hazard, confirmation of results  
azt\_ddc\_NA\_CH\_R=getCumulativeHazardNA(as.numeric(sub("+","",azt\_ddc,fixed=TRUE)),ifelse(grepl("+",azt\_ddc,fixed=TRUE),0,1))  
show(azt\_ddc\_NA)

## orderedEventTimes\_tj eventsAtEventTime\_ej inRiskSetAtTime\_nj  
## 1 0 0 17  
## 2 6 1 16  
## 3 11 1 15  
## 4 12 1 14  
## 5 32 1 13  
## 6 35 1 12  
## 7 39 1 10  
## 8 45 1 9  
## 9 49 1 8  
## 10 75 1 7  
## 11 80 1 6  
## 12 84 1 5  
## 13 85 1 4  
## 14 87 1 3  
## 15 102 1 2  
## censoredObservationsInInterval\_cj cumulativeHazardRate\_ht d\_Y\_ratio  
## 1 0 0 17/17  
## 2 1 0.062 1/16  
## 3 0 0.129 1/15  
## 4 0 0.2 1/14  
## 5 0 0.277 1/13  
## 6 0 0.36 1/12  
## 7 1 0.46 1/10  
## 8 0 0.571 1/9  
## 9 0 0.696 1/8  
## 10 0 0.839 1/7  
## 11 0 1.006 1/6  
## 12 0 1.206 1/5  
## 13 0 1.456 1/4  
## 14 0 1.789 1/3  
## 15 0 2.289 1/2  
## cumulativeHazardEstimatedVariance\_vt  
## 1 0  
## 2 0.004  
## 3 0.008  
## 4 0.013  
## 5 0.019  
## 6 0.026  
## 7 0.036  
## 8 0.048  
## 9 0.064  
## 10 0.084  
## 11 0.112  
## 12 0.152  
## 13 0.214  
## 14 0.325  
## 15 0.575  
## nelsonAalenSurvivalCurveAtTime\_s\_tj  
## 1 0  
## 2 0.94  
## 3 0.879  
## 4 0.819  
## 5 0.758  
## 6 0.698  
## 7 0.631  
## 8 0.565  
## 9 0.499  
## 10 0.432  
## 11 0.366  
## 12 0.299  
## 13 0.233  
## 14 0.167  
## 15 0.101

azt\_ddc\_saq\_NA=getNATable(azt\_ddc\_saq,"+")  
azt\_ddc\_saq\_NA\_R = survfit(Surv(as.numeric(sub("+","",azt\_ddc\_saq,fixed=TRUE)),ifelse(grepl("+",azt\_ddc\_saq,fixed=TRUE),0,1))~1,conf.type="log-log",type="fh")  
summary(azt\_ddc\_saq\_NA\_R)

## Call: survfit(formula = Surv(as.numeric(sub("+", "", azt\_ddc\_saq, fixed = TRUE)),   
## ifelse(grepl("+", azt\_ddc\_saq, fixed = TRUE), 0, 1)) ~ 1,   
## conf.type = "log-log", type = "fh")  
##   
## time n.risk n.event survival std.err lower 95% CI upper 95% CI  
## 2 17 1 0.9429 0.0572 0.6417 0.992  
## 3 16 1 0.8857 0.0784 0.6021 0.971  
## 4 15 1 0.8286 0.0930 0.5455 0.943  
## 12 14 1 0.7715 0.1038 0.4883 0.910  
## 22 13 1 0.7144 0.1118 0.4328 0.874  
## 48 12 1 0.6573 0.1177 0.3795 0.834  
## 51 11 1 0.6001 0.1218 0.3287 0.791  
## 56 10 1 0.5430 0.1242 0.2802 0.746  
## 80 9 1 0.4859 0.1250 0.2343 0.698  
## 85 8 1 0.4288 0.1243 0.1908 0.649  
## 90 7 1 0.3717 0.1221 0.1501 0.597  
## 94 6 1 0.3147 0.1182 0.1124 0.543  
## 160 5 1 0.2576 0.1126 0.0780 0.486  
## 171 4 1 0.2006 0.1051 0.0476 0.428  
## 180 3 2 0.0872 0.0846 0.0049 0.327  
## 238 1 1 0.0321 Inf NaN 1.000

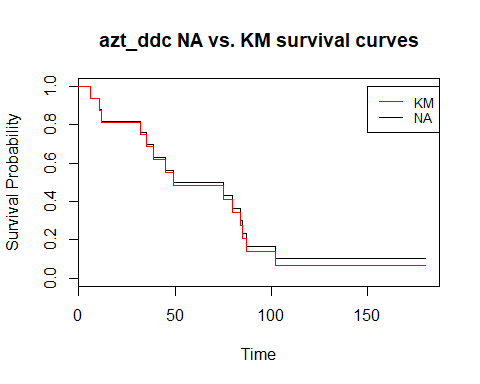
plot(azt\_ddc\_saq\_NA\_R,xlab="Time",ylab="Survival Probability",main="azt\_ddc\_saq\_NA survival curve")



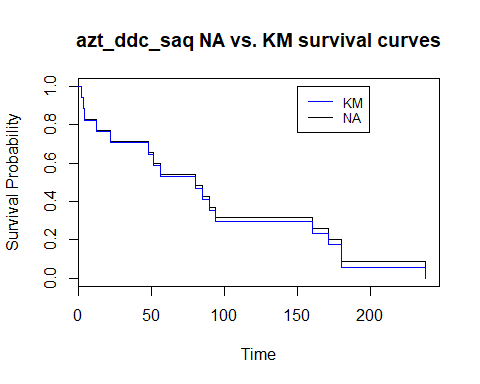
#cumulative hazard, confirmation of results  
azt\_ddc\_saq\_NA\_CH\_R=getCumulativeHazardNA(as.numeric(sub("+","",azt\_ddc\_saq,fixed=TRUE)),ifelse(grepl("+",azt\_ddc\_saq,fixed=TRUE),0,1))  
show(azt\_ddc\_saq\_NA)

## orderedEventTimes\_tj eventsAtEventTime\_ej inRiskSetAtTime\_nj  
## 1 0 0 17  
## 2 2 1 17  
## 3 3 1 16  
## 4 4 1 15  
## 5 12 1 14  
## 6 22 1 13  
## 7 48 1 12  
## 8 51 1 11  
## 9 56 1 10  
## 10 80 1 9  
## 11 85 1 8  
## 12 90 1 7  
## 13 94 1 6  
## 14 160 1 5  
## 15 171 1 4  
## 16 180 2 3  
## 17 238 1 1  
## censoredObservationsInInterval\_cj cumulativeHazardRate\_ht d\_Y\_ratio  
## 1 0 0 17/17  
## 2 0 0.059 1/17  
## 3 0 0.122 1/16  
## 4 0 0.189 1/15  
## 5 0 0.26 1/14  
## 6 0 0.337 1/13  
## 7 0 0.42 1/12  
## 8 0 0.511 1/11  
## 9 0 0.611 1/10  
## 10 0 0.722 1/9  
## 11 0 0.847 1/8  
## 12 0 0.99 1/7  
## 13 0 1.157 1/6  
## 14 0 1.357 1/5  
## 15 0 1.607 1/4  
## 16 0 2.274 2/3  
## 17 0 3.274 1/1  
## cumulativeHazardEstimatedVariance\_vt  
## 1 0  
## 2 0.003  
## 3 0.007  
## 4 0.011  
## 5 0.016  
## 6 0.022  
## 7 0.029  
## 8 0.037  
## 9 0.047  
## 10 0.059  
## 11 0.075  
## 12 0.095  
## 13 0.123  
## 14 0.163  
## 15 0.226  
## 16 0.448  
## 17 1.448  
## nelsonAalenSurvivalCurveAtTime\_s\_tj  
## 1 0  
## 2 0.943  
## 3 0.885  
## 4 0.828  
## 5 0.771  
## 6 0.714  
## 7 0.657  
## 8 0.6  
## 9 0.543  
## 10 0.486  
## 11 0.429  
## 12 0.372  
## 13 0.314  
## 14 0.257  
## 15 0.2  
## 16 0.103  
## 17 0.038

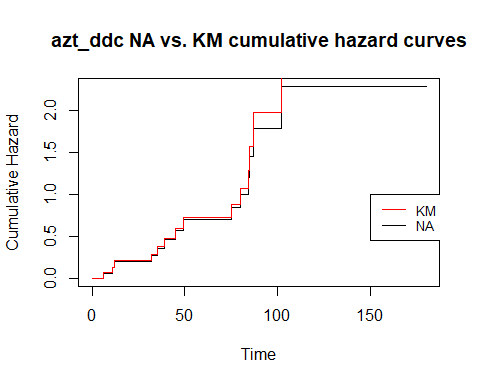
#compare NA to KM survival curve graphically  
#azt\_ddc  
plot(survfit(Surv(as.numeric(sub("+","",azt\_ddc,fixed=TRUE)),ifelse(grepl("+",azt\_ddc,fixed=TRUE),0,1))~1,conf.type="none",type="fh"),xlab="Time",ylab="Survival Probability",main="azt\_ddc NA vs. KM survival curves")  
lines(survfit(Surv(as.numeric(sub("+","",azt\_ddc,fixed=TRUE)),ifelse(grepl("+",azt\_ddc,fixed=TRUE),0,1))~1,conf.type="none",type="kaplan-meier"),col="red")  
legend(150, 1, legend=c("KM", "NA"),  
 col=c("red", "black"), lty=1,cex=0.8)



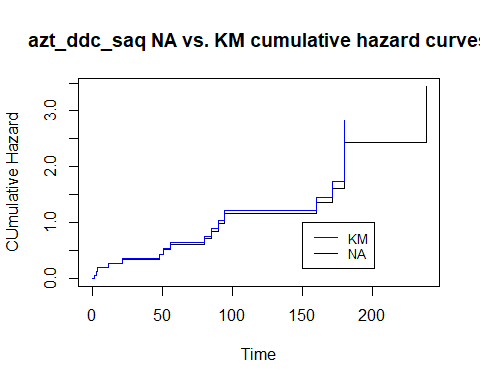
#azt\_ddc\_saq  
plot(survfit(Surv(as.numeric(sub("+","",azt\_ddc\_saq,fixed=TRUE)),ifelse(grepl("+",azt\_ddc\_saq,fixed=TRUE),0,1))~1,conf.type="none",type="fh"),xlab="Time",ylab="Survival Probability",main="azt\_ddc\_saq NA vs. KM survival curves")  
lines(survfit(Surv(as.numeric(sub("+","",azt\_ddc\_saq,fixed=TRUE)),ifelse(grepl("+",azt\_ddc\_saq,fixed=TRUE),0,1))~1,conf.type="none",type="kaplan-meier"),col="blue")  
legend(150, 1, legend=c("KM", "NA"),  
 col=c("blue", "black"), lty=1,cex=0.8)



#compare NA to KM cumulative hazard curve graphically  
plot(survfit(Surv(as.numeric(sub("+","",azt\_ddc,fixed=TRUE)),ifelse(grepl("+",azt\_ddc,fixed=TRUE),0,1))~1,conf.type="none",type="fh"),xlab="Time",ylab="Cumulative Hazard",main="azt\_ddc NA vs. KM cumulative hazard curves",fun="cumhaz")  
lines(survfit(Surv(as.numeric(sub("+","",azt\_ddc,fixed=TRUE)),ifelse(grepl("+",azt\_ddc,fixed=TRUE),0,1))~1,conf.type="none",type="kaplan-meier"),col="red",fun="cumhaz")  
legend(150, 1, legend=c("KM", "NA"),  
 col=c("red", "black"), lty=1,cex=0.8)



#azt\_ddc\_saq  
plot(survfit(Surv(as.numeric(sub("+","",azt\_ddc\_saq,fixed=TRUE)),ifelse(grepl("+",azt\_ddc\_saq,fixed=TRUE),0,1))~1,conf.type="none",type="fh"),xlab="Time",ylab="CUmulative Hazard",main="azt\_ddc\_saq NA vs. KM cumulative hazard curves",fun="cumhaz")  
lines(survfit(Surv(as.numeric(sub("+","",azt\_ddc\_saq,fixed=TRUE)),ifelse(grepl("+",azt\_ddc\_saq,fixed=TRUE),0,1))~1,conf.type="none",type="kaplan-meier"),col="blue",fun="cumhaz")  
legend(150, 1, legend=c("KM", "NA"),  
 col=c("blue", "black"), lty=1,cex=0.8)



library(KMsurv)

## Warning: package 'KMsurv' was built under R version 3.5.2

library(tidyverse)

## -- Attaching packages ----------------------------------------------------------------------------------------------------------------------- tidyverse 1.2.1 --

## v ggplot2 3.1.0 v purrr 0.2.5  
## v tibble 1.4.2 v dplyr 0.7.6  
## v tidyr 0.8.1 v stringr 1.3.1  
## v readr 1.2.1 v forcats 0.3.0

## -- Conflicts -------------------------------------------------------------------------------------------------------------------------- tidyverse\_conflicts() --  
## x dplyr::filter() masks stats::filter()  
## x dplyr::lag() masks stats::lag()

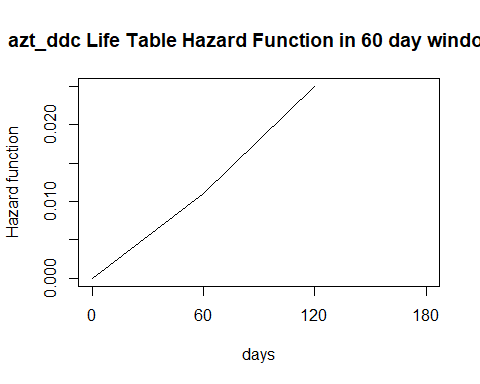
getLifeTableInput = function(censoredTimesVector,censorSymbol){  
 #get numeric representation of censor vector  
 censoredTimesVectorNumeric=as.numeric(sub(censorSymbol,'',censoredTimesVector,fixed=TRUE))  
 #count number of actual rows in KM table  
 cnt\_n=length(censoredTimesVectorNumeric)  
 #create first row of KM table  
 lifeTableInputTable=setNames(data.frame(matrix(nrow=1,c(NA,NA,NA)),stringsAsFactors=FALSE),c("time","nlost","nevent"))  
 censoredTimesVectorNumeric=sort(censoredTimesVectorNumeric)  
 for (i in 1:max(censoredTimesVectorNumeric)){  
 if(i %in% censoredTimesVectorNumeric){  
 #create empty row to fill in  
 lifeTableInputRow=setNames(data.frame(matrix(NA,nrow=1,ncol=length(names(lifeTableInputTable)))),names(lifeTableInputTable))  
 lifeTableInputRow$time=i  
 #count how many events at time  
 lifeTableInputRow$nevent=length(which(censoredTimesVector==i))  
 #count how many censured at time  
 lifeTableInputRow$nlost=length(which(censoredTimesVector==paste0(i,censorSymbol)))  
 lifeTableInputTable=rbind(lifeTableInputTable,lifeTableInputRow)  
 }  
 }  
 na.omit(lifeTableInputTable)  
}  
azt\_ddc\_numeric=as.numeric(gsub("+","",azt\_ddc,fixed=TRUE))  
cuts\_ad=seq(0,max(azt\_ddc\_numeric),60)  
#ensure no loss of upper bound when incrementing  
if(max(azt\_ddc\_numeric)>cuts\_ad[length(cuts\_ad)]){cuts\_ad=cuts\_ad=c(cuts\_ad,(max(azt\_ddc\_numeric)))}  
azt\_ddc\_lt\_raw=getLifeTableInput(azt\_ddc,"+")  
lifetab\_dat=mutate(azt\_ddc\_lt\_raw,time\_cat = cut(time, cuts\_ad)) %>% group\_by(time\_cat) %>% summarize(ilost=sum(nlost),ievent=sum(nevent))  
azt\_ddc\_lt=lifetab(tis = c(0,azt\_ddc\_lt\_raw$time), ninit = length(azt\_ddc), nlost = azt\_ddc\_lt\_raw$nlost, nevent = azt\_ddc\_lt\_raw$nevent) %>% drop\_na(hazard)  
show(azt\_ddc\_lt)

## nsubs nlost nrisk nevent surv pdf hazard se.surv  
## 0-4 17 1 16.5 0 1.00000 0.000000000 0.00000000 0.00000000  
## 4-6 16 0 16.0 1 1.00000 0.031250000 0.03225806 0.00000000  
## 6-11 15 0 15.0 1 0.93750 0.012500000 0.01379310 0.06051536  
## 11-12 14 0 14.0 1 0.87500 0.062500000 0.07407407 0.08267973  
## 12-32 13 0 13.0 1 0.81250 0.003125000 0.00400000 0.09757809  
## 32-35 12 0 12.0 1 0.75000 0.020833333 0.02898551 0.10825318  
## 35-38 11 1 10.5 0 0.68750 0.000000000 0.00000000 0.11587810  
## 38-39 10 0 10.0 1 0.68750 0.068750000 0.10526316 0.11587810  
## 39-45 9 0 9.0 1 0.61875 0.011458333 0.01960784 0.12300557  
## 45-49 8 0 8.0 1 0.55000 0.017187500 0.03333333 0.12710724  
## 49-75 7 0 7.0 1 0.48125 0.002644231 0.00591716 0.12847323  
## 75-80 6 0 6.0 1 0.41250 0.013750000 0.03636364 0.12719172  
## 80-84 5 0 5.0 1 0.34375 0.017187500 0.05555556 0.12318011  
## 84-85 4 0 4.0 1 0.27500 0.068750000 0.28571429 0.11615588  
## 85-87 3 0 3.0 1 0.20625 0.034375000 0.20000000 0.10551909  
## 87-102 2 0 2.0 1 0.13750 0.004583333 0.04444444 0.08999783  
## se.pdf se.hazard  
## 0-4 NaN NaN  
## 4-6 0.030257682 0.032241277  
## 6-11 0.012103073 0.013784901  
## 11-12 0.060515365 0.074023251  
## 12-32 0.003025768 0.003996799  
## 32-35 0.020171788 0.028958098  
## 35-38 NaN NaN  
## 38-39 0.066243366 0.105117263  
## 39-45 0.011040561 0.019573890  
## 45-49 0.016560842 0.033259177  
## 49-75 0.002547822 0.005899627  
## 75-80 0.013248673 0.036213062  
## 80-84 0.016560842 0.055211555  
## 84-85 0.066243366 0.282783805  
## 85-87 0.033121683 0.195959179  
## 87-102 0.004416224 0.041902624

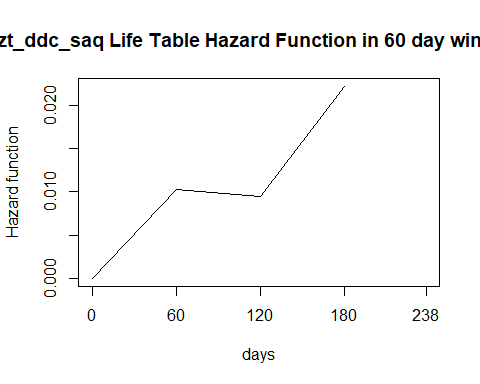
azt\_ddc\_lt\_60=lifetab(tis = cuts\_ad, ninit = length(azt\_ddc), nlost = lifetab\_dat$ilost, nevent = lifetab\_dat$ievent)   
azt\_ddc\_saq\_numeric=as.numeric(gsub("+","",azt\_ddc\_saq,fixed=TRUE))  
cuts\_ads=seq(0,max(azt\_ddc\_saq\_numeric),by=60)  
#ensure no loss of upper bound when incrementing  
if(max(azt\_ddc\_saq\_numeric)>cuts\_ads[length(cuts\_ads)]){cuts\_ads=c(cuts\_ads,(max(azt\_ddc\_saq\_numeric)))}  
azt\_ddc\_saq\_lt\_raw=getLifeTableInput(azt\_ddc\_saq,"+")  
lifetab\_dat=mutate(azt\_ddc\_saq\_lt\_raw,time\_cat = cut(time, cuts\_ads)) %>% group\_by(time\_cat) %>% summarize(ilost=sum(nlost),ievent=sum(nevent))  
  
azt\_ddc\_saq\_lt=lifetab(tis = c(0,azt\_ddc\_saq\_lt\_raw$time), ninit = length(azt\_ddc\_saq), nlost = azt\_ddc\_saq\_lt\_raw$nlost, nevent = azt\_ddc\_saq\_lt\_raw$nevent) %>% drop\_na(hazard)  
show(azt\_ddc\_saq\_lt)

## nsubs nlost nrisk nevent surv pdf hazard  
## 0-2 17 0 17 1 1.0000000 0.0294117647 0.030303030  
## 2-3 16 0 16 1 0.9411765 0.0588235294 0.064516129  
## 3-4 15 0 15 1 0.8823529 0.0588235294 0.068965517  
## 4-12 14 0 14 1 0.8235294 0.0073529412 0.009259259  
## 12-22 13 0 13 1 0.7647059 0.0058823529 0.008000000  
## 22-48 12 0 12 1 0.7058824 0.0022624434 0.003344482  
## 48-51 11 0 11 1 0.6470588 0.0196078431 0.031746032  
## 51-56 10 0 10 1 0.5882353 0.0117647059 0.021052632  
## 56-80 9 0 9 1 0.5294118 0.0024509804 0.004901961  
## 80-85 8 0 8 1 0.4705882 0.0117647059 0.026666667  
## 85-90 7 0 7 1 0.4117647 0.0117647059 0.030769231  
## 90-94 6 0 6 1 0.3529412 0.0147058824 0.045454545  
## 94-160 5 0 5 1 0.2941176 0.0008912656 0.003367003  
## 160-171 4 0 4 1 0.2352941 0.0053475936 0.025974026  
## 171-180 3 0 3 2 0.1764706 0.0130718954 0.111111111  
## se.surv se.pdf se.hazard  
## 0-2 0.00000000 0.0285336029 0.030289114  
## 2-3 0.05706721 0.0570672059 0.064482553  
## 3-4 0.07814249 0.0570672059 0.068924503  
## 4-12 0.09245944 0.0071334007 0.009252906  
## 12-22 0.10287937 0.0057067206 0.007993597  
## 22-48 0.11051017 0.0021948925 0.003341319  
## 48-51 0.11590404 0.0190224020 0.031710018  
## 51-56 0.11936462 0.0114134412 0.021023453  
## 56-80 0.12105782 0.0023778002 0.004893473  
## 80-85 0.12105782 0.0114134412 0.026607341  
## 85-90 0.11936462 0.0114134412 0.030678062  
## 90-94 0.11590404 0.0142668015 0.045266327  
## 94-160 0.11051017 0.0008646546 0.003346155  
## 160-171 0.10287937 0.0051879278 0.025707619  
## 171-180 0.09245944 0.0086824989 0.068041382

azt\_ddc\_saq\_lt\_60=lifetab(tis = cuts\_ads, ninit = length(azt\_ddc\_saq), nlost = lifetab\_dat$ilost, nevent = lifetab\_dat$ievent)   
  
#plot azt\_ddc  
plot(cuts\_ad,c(0,azt\_ddc\_lt\_60$hazard),type='l',ylab="Hazard function",xlab="days",xaxt="n",main="azt\_ddc Life Table Hazard Function in 60 day windows")  
axis(1, at = cuts\_ad, las=1)



#plot azt\_ddc\_saq  
plot(cuts\_ads,c(0,azt\_ddc\_saq\_lt\_60$hazard),type='l',ylab="Hazard function",xlab="days",xaxt="n",main="azt\_ddc\_saq Life Table Hazard Function in 60 day windows")  
axis(1, at = cuts\_ads, las=1)



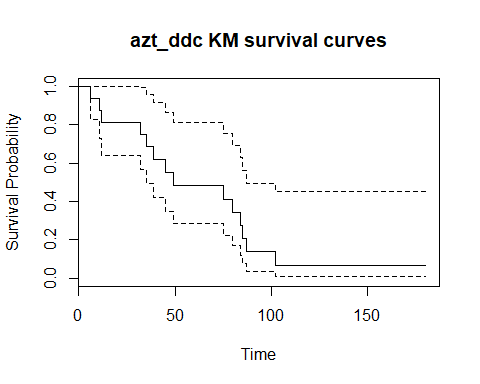
library(kmconfband)

## Warning: package 'kmconfband' was built under R version 3.5.2

#survival function confidence intervals for azt\_ddc  
azt\_ddc\_s=survfit(Surv(as.numeric(sub("+","",azt\_ddc,fixed=TRUE)),ifelse(grepl("+",azt\_ddc,fixed=TRUE),0,1))~1)  
azt\_ddc\_s\_ci=summary(azt\_ddc\_s)  
azt\_ddc\_s\_ci\_df=data.frame(azt\_ddc\_s\_ci$time,azt\_ddc\_s\_ci$n.risk,azt\_ddc\_s\_ci$n.event,azt\_ddc\_s\_ci$surv,azt\_ddc\_s\_ci$std.err,azt\_ddc\_s\_ci$lower,azt\_ddc\_s\_ci$upper)  
#lower and upper intervals and bounds  
azt\_ddc\_s\_ci\_cb=setNames(cbind(azt\_ddc\_s\_ci\_df,confband(azt\_ddc\_s)[1:dim(azt\_ddc\_s\_ci\_df)[1],]),c("time","n.risk","n.event","survival","std.err","lower 95% CI" ,"upper 95% CI","lower 95% CB" ,"upper 95% CB"))

## The critical value required is 0.4404776

plot(azt\_ddc\_s,xlab="Time",ylab="Survival Probability",main="azt\_ddc KM survival curves")



rtr\_example=c(3,4,"5+",6,"6+","8+",11,14,15,"16+")  
decrement=0  
#redistribute to right value  
getRedistributeToRightTable = function(censoredTimesVector,censorSymbol){  
 #get numeric representation of censor vector  
 censoredTimesVectorNumeric=as.numeric(sub(censorSymbol,'',censoredTimesVector,fixed=TRUE))  
 #sort to ensure when determine step# omit correct last element  
 censoredTimesVector=censoredTimesVector[order(censoredTimesVectorNumeric)]  
 #count number of actual rows in RTR table  
 cnt\_n=length(censoredTimesVectorNumeric)  
 steps=length(which(grepl(censorSymbol,censoredTimesVector[1:length(censoredTimesVector)-1],fixed=TRUE)))  
 #create first row of RTR table  
 rtrTable=setNames(data.frame(matrix(nrow=1,c(0,(1/cnt\_n),"",1)),stringsAsFactors=FALSE),c("observation","decrement","action","S\_t"))  
 censoredTimesVectorNumeric=sort(censoredTimesVectorNumeric)  
 decrement = 0  
 base=1/cnt\_n  
 i=0  
 uniqueCensoredTimesVector=unique(censoredTimesVector)  
 for (dataPoint in uniqueCensoredTimesVector){  
 i=i+1  
 #create empty row to fill in  
 rtrTableRow=setNames(data.frame(matrix(NA,nrow=1,ncol=length(names(rtrTable)))),names(rtrTable))  
 rtrTableRow$observation=dataPoint  
 #count how many events or censures at unique dataPoint  
 numerator=length(which(censoredTimesVector==dataPoint))  
 #censured at time?  
 if(length(which(censoredTimesVector==dataPoint & grep(censorSymbol,dataPoint,fixed=TRUE)))>0){  
 #no change in survival function  
 rtrTableRow$S\_t=rtrTable[dim(rtrTable)[1],c("S\_t")]  
 #display decrement of 0  
 rtrTableRow$decrement=0  
 #update denominator   
 denominator=length(censoredTimesVector)-max(which(dataPoint == censoredTimesVector))  
 if(i==length(uniqueCensoredTimesVector)){  
 rtrTableRow$action=paste0("survival is 1-(1)")  
 rtrTableRow$S\_t=0   
 rtrTableRow$decrement=1  
 }  
 else{  
 rtrTableRow$action=paste0("next time survival is 1-(",decrement," + ",base," + ",base,"\*",numerator,"/",denominator,")")  
 }  
 if(length(uniqueCensoredTimesVector)>=(i+1) && (!grepl(censorSymbol,uniqueCensoredTimesVector[i+1],fixed=TRUE))){  
 #update decrement only if not proceeded by a censored observation  
 decrement = decrement + base+(base\*numerator/denominator)  
 }  
 #update base  
 base=base+(base\*numerator/denominator)  
 }  
 else{  
 #update decrement and show action  
 if (as.numeric(rtrTable[dim(rtrTable)[1],c("decrement")])==0)  
 {  
 rtrTableRow$action=paste0("survival is 1-(",decrement,")")  
 decrement = decrement  
 }  
 else{  
 decrement = decrement + base  
 rtrTableRow$action=paste0("survival is 1-(",decrement,")")  
 }  
 rtrTableRow$S\_t=1-decrement  
 #update decrement display  
 rtrTableRow$decrement=decrement  
 }  
 #add row to rtrtable  
 rtrTable=rbind(rtrTable,rtrTableRow)  
 }  
 #eleminate initial bogus row  
 rtrTable$decrement=round(as.numeric(rtrTable$decrement),3)  
 rtrTable$S\_t=round(as.numeric(rtrTable$S\_t),3)  
 rtrTable[-1,]  
}  
  
azt\_ddc\_saq\_RTR=getRedistributeToRightTable(azt\_ddc\_saq,"+")  
azt\_ddc\_RTR=getRedistributeToRightTable(azt\_ddc,"+")  
show(azt\_ddc\_saq\_RTR)

## observation decrement action S\_t  
## 2 2 0.059 survival is 1-(0.0588235294117647) 0.941  
## 3 3 0.118 survival is 1-(0.117647058823529) 0.882  
## 4 4 0.176 survival is 1-(0.176470588235294) 0.824  
## 5 12 0.235 survival is 1-(0.235294117647059) 0.765  
## 6 22 0.294 survival is 1-(0.294117647058824) 0.706  
## 7 48 0.353 survival is 1-(0.352941176470588) 0.647  
## 8 51 0.412 survival is 1-(0.411764705882353) 0.588  
## 9 56 0.471 survival is 1-(0.470588235294118) 0.529  
## 10 80 0.529 survival is 1-(0.529411764705882) 0.471  
## 11 85 0.588 survival is 1-(0.588235294117647) 0.412  
## 12 90 0.647 survival is 1-(0.647058823529412) 0.353  
## 13 94 0.706 survival is 1-(0.705882352941177) 0.294  
## 14 160 0.765 survival is 1-(0.764705882352941) 0.235  
## 15 171 0.824 survival is 1-(0.823529411764706) 0.176  
## 16 180 0.882 survival is 1-(0.882352941176471) 0.118  
## 17 238 0.941 survival is 1-(0.941176470588235) 0.059

show(azt\_ddc\_RTR)

## observation decrement  
## 2 4+ 0.000  
## 3 6 0.062  
## 4 11 0.125  
## 5 12 0.188  
## 6 32 0.250  
## 7 35 0.312  
## 8 38+ 0.000  
## 9 39 0.381  
## 10 45 0.450  
## 11 49 0.519  
## 12 75 0.588  
## 13 80 0.656  
## 14 84 0.725  
## 15 85 0.794  
## 16 87 0.862  
## 17 102 0.931  
## 18 180+ 1.000  
## action  
## 2 next time survival is 1-(0 + 0.0588235294117647 + 0.0588235294117647\*1/16)  
## 3 survival is 1-(0.0625)  
## 4 survival is 1-(0.125)  
## 5 survival is 1-(0.1875)  
## 6 survival is 1-(0.25)  
## 7 survival is 1-(0.3125)  
## 8 next time survival is 1-(0.3125 + 0.0625 + 0.0625\*1/10)  
## 9 survival is 1-(0.38125)  
## 10 survival is 1-(0.45)  
## 11 survival is 1-(0.51875)  
## 12 survival is 1-(0.5875)  
## 13 survival is 1-(0.65625)  
## 14 survival is 1-(0.725)  
## 15 survival is 1-(0.79375)  
## 16 survival is 1-(0.8625)  
## 17 survival is 1-(0.93125)  
## 18 survival is 1-(1)  
## S\_t  
## 2 1.000  
## 3 0.938  
## 4 0.875  
## 5 0.812  
## 6 0.750  
## 7 0.688  
## 8 0.688  
## 9 0.619  
## 10 0.550  
## 11 0.481  
## 12 0.412  
## 13 0.344  
## 14 0.275  
## 15 0.206  
## 16 0.138  
## 17 0.069  
## 18 0.000

rtr\_example\_RTR = getRedistributeToRightTable(rtr\_example,"+")

#column1 of book page 137 problem 4.7  
entry\_c1 = c(58,58,59,60,60,61,61,62,62,62,63,63,64,66,66)  
exit\_c1=c(60,63,69,62,65,72,69,73,66,65,68,74,71,68,69)  
death\_c1 = c(1,1,0,1,1,0,0,0,1,1,1,0,1,1,1)  
#column1 of book page 137 problem 4.7  
entry\_c2=c(67,67,67,68,69,69,69,70,70,70,71,72,72,73,73)  
exit\_c2=c(70,77,69,72,79,72,70,76,71,78,79,76,73,80,74)  
death\_c2=c(1,1,1,1,0,1,1,0,1,0,0,1,1,0,1)  
df2\_lec = data.frame(cbind(entry\_c1,exit\_c1,death\_c1))  
df2\_prob = setNames(data.frame(cbind(c(entry\_c1,entry\_c2),c(exit\_c1,exit\_c2),c(death\_c1,death\_c2))),c("entry","exit","death"))  
  
df2\_lec\_censored\_noLT=c(60,63,"69+",62,65,"72+","69+","73+",66,65,68,"74+",71,68,69)  
df2\_prob\_censored\_noLT=c(60,63,"69+",62,65,"72+","69+","73+",66,65,68,"74+",71,68,69,70,77,69,72,"79+",72,70,"76+",71,"78+","79+",76,73,"80+",74)  
  
getKM\_LT\_Table = function(entryExitDeathVector,deathSymbol){  
 #create first row of KM\_LT table  
 km\_ltTable=setNames(data.frame(matrix(nrow=1,c(0,0,0,length(entryExitDeathVector),"",1)),stringsAsFactors=FALSE),c("tj","ej","cj","nj","c\_tj-1","s\_tj"))  
 #sort by exit  
 orderedIndices=order(entryExitDeathVector$exit)  
 entryExitDeathVector=entryExitDeathVector[orderedIndices,]  
 for (time in unique(entryExitDeathVector$exit)){  
 total\_at\_risk=length(which(entryExitDeathVector$entry<=time))  
 gone=length(which(entryExitDeathVector$exit<time & entryExitDeathVector$death==deathSymbol))  
 events=length(which(entryExitDeathVector$exit==time & entryExitDeathVector$death==deathSymbol))  
 truncated\_at\_risk=total\_at\_risk-gone-as.numeric(km\_ltTable$cj[length(km\_ltTable$cj)])  
 censored=length(which(entryExitDeathVector$exit==time & entryExitDeathVector$death!=deathSymbol))  
 numerator=truncated\_at\_risk-events  
 denominator=truncated\_at\_risk  
 #print(paste0(time,": ",numerator,"/",denominator))  
 #create empty row to fill in  
 km\_ltTableRow=setNames(data.frame(matrix(NA,nrow=1,ncol=length(names(km\_ltTable)))),names(km\_ltTable))  
 km\_ltTableRow$tj=time  
 #count how many events at time  
 km\_ltTableRow$ej=events  
 #count how many censured at time  
 km\_ltTableRow$cj=censored  
 km\_ltTableRow$nj=truncated\_at\_risk  
 #sum events and number censored at time  
 km\_ltTableRow[c("c\_tj-1")]=paste0(numerator,"/",denominator)  
 km\_ltTableRow$s\_tj=round((numerator/denominator)\*as.numeric(km\_ltTable[dim(km\_ltTable)[1],c("s\_tj")]),4)  
 if (km\_ltTableRow$ej>0 | km\_ltTableRow$s\_tj==1){  
 #add row to km\_lttable  
 km\_ltTable=rbind(km\_ltTable,km\_ltTableRow)  
 }  
 }  
 km\_ltTable  
}  
df2\_lec\_LT=getKM\_LT\_Table(df2\_lec,1)  
show(df2\_lec\_LT)

## tj ej cj nj c\_tj-1 s\_tj  
## 1 0 0 0 3 1  
## 2 60 1 0 5 4/5 0.8  
## 3 62 1 0 9 8/9 0.7111  
## 4 63 1 0 10 9/10 0.64  
## 5 65 2 0 10 8/10 0.512  
## 6 66 1 0 10 9/10 0.4608  
## 7 68 2 0 9 7/9 0.3584  
## 8 69 1 2 7 6/7 0.3072  
## 9 71 1 0 4 3/4 0.2304

getKMTableNoCensorRemoval = function(censoredTimesVector,censorSymbol){  
 #get numeric representation of censor vector  
 censoredTimesVectorNumeric=as.numeric(sub(censorSymbol,'',censoredTimesVector,fixed=TRUE))  
 #count number of actual rows in KM table  
 cnt\_n=length(censoredTimesVectorNumeric)  
 #create first row of KM table  
 kmTable=setNames(data.frame(matrix(nrow=1,c(0,0,0,cnt\_n,as.character("-"),as.character(paste0(cnt\_n,"/",cnt\_n)),1)),stringsAsFactors=FALSE),c("orderedEventTimes\_tj","eventsAtEventTime\_ej",  
 "censoredObservationsInInterval\_cj","inRiskSetAtTime\_nj","kaplanMeirSurvivalCurveAtTime\_s\_tj-1","c\_tj-1","kaplanMeirSurvivalCurveAtTime\_s\_tj"))  
 # orderedIndices=order(censoredTimesVectorNumeric)  
 # censoredTimesVectorNumeric=censoredTimesVectorNumeric[orderedIndices]  
 # censoredTimesVector=censoredTimesVector[orderedIndices]  
 censoredTimesVectorNumeric=sort(censoredTimesVectorNumeric)  
 for (i in 1:max(censoredTimesVectorNumeric)){  
 if(i %in% censoredTimesVectorNumeric){  
 #create empty row to fill in  
 kmTableRow=setNames(data.frame(matrix(NA,nrow=1,ncol=length(names(kmTable)))),names(kmTable))  
 kmTableRow$orderedEventTimes\_tj=i  
 #count how many events at time  
 kmTableRow$eventsAtEventTime\_ej=length(which(censoredTimesVector==i))  
 #count how many censured at time  
 kmTableRow$censoredObservationsInInterval\_cj=length(which(censoredTimesVector==paste0(i,censorSymbol)))  
 kmTableRow$inRiskSetAtTime\_nj=cnt\_n  
 #sum events and censored  
 loss=kmTableRow$eventsAtEventTime\_ej+kmTableRow$censoredObservationsInInterval\_cj  
 kmTableRow[c("kaplanMeirSurvivalCurveAtTime\_s\_tj-1")]=kmTable[dim(kmTable)[1],c("kaplanMeirSurvivalCurveAtTime\_s\_tj")]  
 #TOOK LAZY WAY OUT AND JUST ADDED BACK IN THE CENSORED OBS - WILL DO CORRECT WAY LATER I Hope  
 numerator=(cnt\_n-loss+kmTableRow$censoredObservationsInInterval\_cj)  
 denominator=cnt\_n  
 kmTableRow[c("c\_tj-1")]=paste0(numerator,"/",denominator)  
 kmTableRow$kaplanMeirSurvivalCurveAtTime\_s\_tj=round(numerator/denominator\*as.numeric(kmTable[dim(kmTable)[1],c("kaplanMeirSurvivalCurveAtTime\_s\_tj")]),2)  
 #update count  
 cnt\_n=cnt\_n-loss  
 #don't add a row when no events 0 should put this at top but no time :0  
 if (kmTableRow$eventsAtEventTime\_ej>0 | cnt\_n==length(censoredTimesVectorNumeric)){  
 #add row to kmtable  
 kmTable=rbind(kmTable,kmTableRow)  
 }  
 }  
 }  
 kmTable  
}  
df2\_lec\_censored\_noLT\_KM=getKMTableNoCensorRemoval(df2\_lec\_censored\_noLT,"+")  
show(df2\_lec\_censored\_noLT\_KM)

## orderedEventTimes\_tj eventsAtEventTime\_ej  
## 1 0 0  
## 2 60 1  
## 3 62 1  
## 4 63 1  
## 5 65 2  
## 6 66 1  
## 7 68 2  
## 8 69 1  
## 9 71 1  
## censoredObservationsInInterval\_cj inRiskSetAtTime\_nj  
## 1 0 15  
## 2 0 15  
## 3 0 14  
## 4 0 13  
## 5 0 12  
## 6 0 10  
## 7 0 9  
## 8 2 7  
## 9 0 4  
## kaplanMeirSurvivalCurveAtTime\_s\_tj-1 c\_tj-1  
## 1 - 15/15  
## 2 1 14/15  
## 3 0.93 13/14  
## 4 0.86 12/13  
## 5 0.79 10/12  
## 6 0.66 9/10  
## 7 0.59 7/9  
## 8 0.46 6/7  
## 9 0.39 3/4  
## kaplanMeirSurvivalCurveAtTime\_s\_tj  
## 1 1  
## 2 0.93  
## 3 0.86  
## 4 0.79  
## 5 0.66  
## 6 0.59  
## 7 0.46  
## 8 0.39  
## 9 0.29

4.7(a) Since the diabetics needed to survive long enough from birth until the study began, the data is left truncated. Construct a table showing the number of subjects at risk, Y, as a function of age.

#above code and data match output in lecture slide 33 output table - now try problem data  
df2\_prob\_LT=getKM\_LT\_Table(df2\_prob,1)  
show(df2\_prob\_LT)

## tj ej cj nj c\_tj-1 s\_tj  
## 1 0 0 0 3 1  
## 2 60 1 0 5 4/5 0.8  
## 3 62 1 0 9 8/9 0.7111  
## 4 63 1 0 10 9/10 0.64  
## 5 65 2 0 10 8/10 0.512  
## 6 66 1 0 10 9/10 0.4608  
## 7 68 2 0 13 11/13 0.3899  
## 8 69 2 2 14 12/14 0.3342  
## 9 70 2 0 13 11/13 0.2828  
## 10 71 2 0 14 12/14 0.2424  
## 11 72 2 1 14 12/14 0.2078  
## 12 73 1 1 13 12/13 0.1918  
## 13 74 1 1 12 11/12 0.1758  
## 14 76 1 1 11 10/11 0.1598  
## 15 77 1 0 10 9/10 0.1438

df2\_prob\_censored\_noLT\_KM=getKMTableNoCensorRemoval(df2\_prob\_censored\_noLT,"+")  
show(df2\_prob\_censored\_noLT\_KM)

## orderedEventTimes\_tj eventsAtEventTime\_ej  
## 1 0 0  
## 2 60 1  
## 3 62 1  
## 4 63 1  
## 5 65 2  
## 6 66 1  
## 7 68 2  
## 8 69 2  
## 9 70 2  
## 10 71 2  
## 11 72 2  
## 12 73 1  
## 13 74 1  
## 14 76 1  
## 15 77 1  
## censoredObservationsInInterval\_cj inRiskSetAtTime\_nj  
## 1 0 30  
## 2 0 30  
## 3 0 29  
## 4 0 28  
## 5 0 27  
## 6 0 25  
## 7 0 24  
## 8 2 22  
## 9 0 18  
## 10 0 16  
## 11 1 14  
## 12 1 11  
## 13 1 9  
## 14 1 7  
## 15 0 5  
## kaplanMeirSurvivalCurveAtTime\_s\_tj-1 c\_tj-1  
## 1 - 30/30  
## 2 1 29/30  
## 3 0.97 28/29  
## 4 0.94 27/28  
## 5 0.91 25/27  
## 6 0.84 24/25  
## 7 0.81 22/24  
## 8 0.74 20/22  
## 9 0.67 16/18  
## 10 0.6 14/16  
## 11 0.52 12/14  
## 12 0.45 10/11  
## 13 0.41 8/9  
## 14 0.36 6/7  
## 15 0.31 4/5  
## kaplanMeirSurvivalCurveAtTime\_s\_tj  
## 1 1  
## 2 0.97  
## 3 0.94  
## 4 0.91  
## 5 0.84  
## 6 0.81  
## 7 0.74  
## 8 0.67  
## 9 0.6  
## 10 0.52  
## 11 0.45  
## 12 0.41  
## 13 0.36  
## 14 0.31  
## 15 0.25