



저작자표시-비영리-변경금지 2.0 대한민국

이용자는 아래의 조건을 따르는 경우에 한하여 자유롭게

- 이 저작물을 복제, 배포, 전송, 전시, 공연 및 방송할 수 있습니다.

다음과 같은 조건을 따라야 합니다:



저작자표시. 귀하는 원저작자를 표시하여야 합니다.



비영리. 귀하는 이 저작물을 영리 목적으로 이용할 수 없습니다.



변경금지. 귀하는 이 저작물을 개작, 변형 또는 가공할 수 없습니다.

- 귀하는, 이 저작물의 재이용이나 배포의 경우, 이 저작물에 적용된 이용허락조건을 명확하게 나타내어야 합니다.
- 저작권자로부터 별도의 허가를 받으면 이러한 조건들은 적용되지 않습니다.

저작권법에 따른 이용자의 권리는 위의 내용에 의하여 영향을 받지 않습니다.

이것은 [이용허락규약\(Legal Code\)](#)을 이해하기 쉽게 요약한 것입니다.

[Disclaimer](#)

공학석사 학위논문

교차위험 자료에 대한  
가속화위험 모델링 접근법



2021년 2월

부경대학교 대학원

인공지능융합학과

고낙경

공학석사 학위논문

교차위험 자료에 대한  
가속화위험 모델링 접근법

지도교수 하 일 도

이 논문을 공학석사 학위논문으로 제출함.

2021년 2월

부경대학교 대학원

인공지능융합학과

고낙경

고낙경의 공학석사 학위논문을 인준함.

2021년 2월 19일

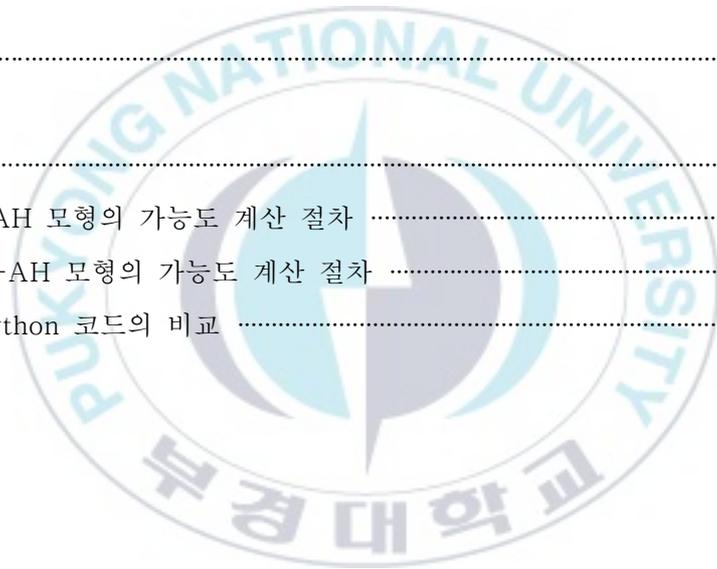


위원장	이학박사	박인호	(인)
위원	이학박사	장대흥	(인)
위원	이학박사	하일도	(인)

# 목 차

표 차례	iii
그림 차례	iv
제 1장 서론 .....	1
제 2장 가속화위험 회귀모형 및 가능도 추론 .....	3
2.1 비례위험 회귀모형 .....	3
2.2 가속화 위험 회귀모형 .....	4
2.2.1 로그-로지스틱 분포에 기반한 AH 모형 .....	7
2.2.2 AH 회귀모형의 가능도 추정 .....	9
2.2.3 시간 효과를 허용하는 AH 회귀모형의 가능도 추정 .....	11
2.3 제한된 평균 생존시간 .....	14
제 3장 예증 .....	16
3.1 신장 감염 자료 .....	16
3.1.1 모형 적합 .....	19
3.1.2 유의성 검토 .....	22
3.1.3 모형 검토 .....	23
3.1.4 모형 선택 .....	23
3.1.5 제한된 평균 생존시간 .....	24
3.2 위암 자료 .....	26
3.2.1 모형 적합 .....	28
3.2.2 유의성 검토 .....	31
3.2.3 모형 검토 .....	31
3.2.4 모형 선택 .....	32

3.2.5 제한된 평균 생존시간 .....	33
제 4장 AH 모형의 확장 .....	35
4.1 혼합치료 AH 모형 .....	35
4.2 혼합치료 AH 모형의 가능도 추정법 .....	37
4.3 SEER 자료 .....	38
4.3.1 자료 설명 .....	38
4.3.2 혼합치료 AH 모형: 모수 추정 및 모형의 적합 .....	40
제 5장 결론 .....	43
참고문헌 .....	44
부록 A LLG-AH 모형의 가능도 계산 절차 .....	46
부록 B LLGT-AH 모형의 가능도 계산 절차 .....	48
부록 C R과 Python 코드의 비교 .....	51



## 표 차례

[표 2.1] 세 모형 (PH, AFT, AH 모형)의 회귀계수 의미의 비교 .....	5
[표 2.2] 세 모형 (PH, AFT, AH 모형)의 위험함수의 비교 .....	6
[표 3.1] 신장 감염 자료: 세 모형(AH, PH, AFT 모형)을 이용한 분석 .....	20
[표 3.2] 신장 감염 자료: 유의성 검토를 위한 통계량과 p-값 .....	22
[표 3.3] 신장 감염 자료: 각 모형에 대한 모형 선택 .....	24
[표 3.4] 신장 감염 자료: RMST와 그룹 간 차이 분석 .....	25
[표 3.5] 위암 자료: 세 모형 (AH, PH, AFT 모형)을 이용한 분석 .....	29
[표 3.6] 위암 자료: 유의성 검토를 위한 통계량과 p-값 .....	31
[표 3.7] 위암 자료: 각 모형에 대한 모형 선택 .....	33
[표 3.8] 위암 자료: RMST와 그룹 간 차이 분석 .....	34
[표 4.1] SEER 자료에 대한 변수설명 .....	38
[표 4.2] SEER 자료: M1의 모수추정 .....	40

## 그림 차례

[그림 2.1] 위험함수: 와이블 분포와 로그-로지스틱 분포 .....	8
[그림 3.1] 신장 감염 자료: 카플란-마이어 생존곡선 .....	17
[그림 3.2] 신장 감염 자료: 비례위험 모형에 대한 잔차도 .....	18
[그림 3.3] 신장 감염 자료: 평활 및 AH 모형에 근거한 위험률 곡선 .....	21
[그림 3.4] 신장 감염 자료: LLG-AH 모형과 LLGT-AH 모형의 적합 .....	23
[그림 3.5] 위암 자료: 카플란-마이어 생존곡선 .....	27
[그림 3.6] 위암 자료: 비례위험 모형에 대한 잔차도 .....	28
[그림 3.7] 위암 자료: 평활 및 AH 모형에 근거한 위험률 곡선 .....	30
[그림 3.8] 위암 자료: LLG-AH 모형과 LLGT-AH 모형의 적합 .....	32
[그림 4.1] SEER 자료: 카플란-마이어 생존곡선 .....	39
[그림 4.2] SEER 자료의 모형적합 .....	42

Accelerated hazards modelling approaches for crossing hazards data

Nak Gyeong Ko

Department of Artificial Intelligence Convergence, The Graduate School,  
Pukyong National University

### Abstract

Proportional hazards model and accelerated failure time model have been widely used in survival analysis. However, the two models can not capture time-scaled effects such as crossing hazard or the gradual effect of treatment over time. For the purpose of this detection, accelerated hazards (AH) model (Chen and Wang, 2000a) has been studied. In this thesis, we propose the use of log-logistic distribution for the baseline hazard function of the AH model. In particular, the log-logistic distribution gives various forms (e.g. non-monotone) of hazard function and also an easy implementation for likelihood-based model fitting due to an explicit form of hazard and survival functions. The likelihood-based estimation procedures of the AH models are derived. The proposed method is extended to AH model with time-effect between groups and also to mixture cure model. Our method is demonstrated with two practical data sets with model checking.

keywords : Crossing hazards data, Accelerated hazards model, Log-logistic distribution, Time effect, Mixture cure rate model

## 제 1장 서론

생존 자료의 분석에 대해 자주 사용되는 두 가지 형태의 모형으로 콕스의 비례 위험 모형 (Cox's proportional hazards model; PH model)과 가속화 실패시간 모형 (accelerated failure time model; AFT model)이 있다. 이러한 모형은 해석이 쉬운 장점이 있지만 두 그룹의 생존을 비교에서 생존함수 (survival function) 또는 위험함수 (hazard function)가 교차되어 있는 자료는 잘 다루지 못한다는 한계가 있다 (Co, 2010). 실제 생존 자료나 임상시험 (clinical trial) 자료에서 생존함수 또는 위험함수가 비례하지 않은 경우가 종종 있다. 이러한 상황에서 사용할 수 있는 모형으로 가속화 위험 모형 (accelerated hazards model; AH model)이 자주 사용되어 왔다 (Chen과 Wang, 2000a,b).

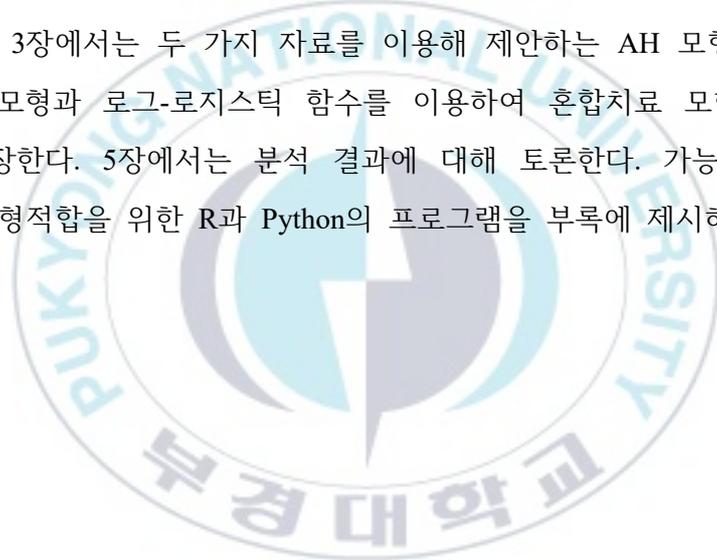
AH 모형은 기저 위험함수 (baseline hazard function)에 기저 (known)의 분포를 지정하는 모수적 방법을 사용할 경우 어떤 특정 분포를 지정하는가에 따라 분석 결과가 달라질 수 있다. 생존시간을  $T$ 로 표현하면  $\log(T)$ 의 분포가 로지스틱 분포 (logistic distribution)일 때  $T$ 의 분포는 로그-로지스틱 분포 (log-logistic distribution; LLG)를 따른다. 로그-로지스틱 분포는 양의 값을 가지는 자료를 다루며 분포의 꼬리가 두터운 형태로 극단값 (extreme value)의 자료를 잘 설명한다. 생존함수와 위험함수가 명확한 형태 (closed form)로 표현되기 때문에 로그-정규 (log-normal)분포보다 중도 절단 (censoring)된 자료를 잘 다루고 (Akhtar와 Khan, 2014) 계산 속도가 빠른 장점이 있다. 특히 기존의 와이블 분포와 달리 비-단조 (non-monotonic) 형태의 위험률 (hazard rate)을 제공해준다.

따라서 본 논문에서는 기저 위험함수로 로그-로지스틱 분포 및 시간 효과 (time effect)를 허용하는 AH 모형을 제안한다. 이때 시간 효과란 그룹 간 효과에 시간을 반

영한 것을 의미한다. 비교를 위해 이 두 방법을 적용한 모형과 생존분석에서 자주 사용되는 와이블 (Weibull) 분포를 적용한 모형을 사용하였다. 예증을 위해 신장 감염 자료 (Klein와 Moeschberger, 1997)와 위암 자료 (Gastrointestinal Tumor Study Group, 1982)를 사용한다. 특히 제안된 방법의 타당성은 두 자료에 근거한 모형 검토를 통해 입증한다. 모형 적합을 위해 R과 Python을 사용하였다.

나아가 로그-로지스틱 분포에 기초한 AH 모형을 혼합 치료율 모형 (mixture cure model)으로 확장하여 실제 임상자료에 적용한 후 적절한 분석 결과를 제시하였다.

본 논문의 구성은 다음과 같다. 2장에서는 AH 모형의 연구방법 및 가능도 추론 방법을 제시한다. 3장에서는 두 가지 자료를 이용해 제안하는 AH 모형을 예증한다. 4장에서는 AH 모형과 로그-로지스틱 함수를 이용하여 혼합치료 모형 (mixture cure model)으로 확장한다. 5장에서는 분석 결과에 대해 토론한다. 가능도 추론을 위한 계산 절차와 모형적합을 위한 R과 Python의 프로그램을 부록에 제시하였다.



## 제 2장 가속화위험 회귀모형 및 가능도 추론

### 2.1. 비례위험 회귀모형

콕스의 비례위험 회귀모형 (Cox's proportional hazards (PH) regression model)은 생존 자료에 대한 위험률 회귀모형으로 생존분석에서 자주 사용되는 모형이다. 이 모형은 생존시간  $T$ 의 위험률과 공변량 (covariate) 간의 관계를 모형화하여 위험률에 유의한 영향을 주는 위험인자를 도출하고자 한다. 이 모형의 형태는 식 (2.1)과 같이 표현된다.

$$\lambda(t;x) = \lambda_0(t)e^{x^T\beta} \quad (2.1)$$

여기서  $\lambda_0(\cdot)$ 는  $x=0$ 일 때 미지의 기저 위험함수 (unknown baseline hazard function) 이고  $\beta = (\beta_1, \dots, \beta_p)^T$ 는 공변량들의 벡터  $x = (x_1, \dots, x_p)^T$ 에 대응하는  $p \times 1$  차원의 회귀계수 벡터이다. PH 모형은 준모수적 (semi-parametric) 모형이며 절편항이 없는 것 (no intercept term)이 특징이다. 그리고 기본적으로 서로 다른 특성을 갖는 개개인들의 위험률은 시간에 관계없이 서로 비례적인 관계를 가짐을 가정한다. 이는 서로 다른 두 개인의 공변량 벡터  $x$ 가 0 (대조군) 또는 1 (처리군)인 이진형인 경우 (binary case) 위험비 (hazard ratio; HR)가 다음과 같이 시간에 의존하지 않는 것을 의미한다.

$$HR(t;x_1,x_2) = \frac{\lambda(t;x_1=1)}{\lambda(t;x_2=0)} = e^\beta \quad (2.2)$$

이러한 위험비  $e^\beta$ 에 대한 해석은  $\beta > 0$ 일 때 처리군은 대조군에 비해  $e^\beta$ 만큼 위험이 증가함을 의미하고  $\beta < 0$ 일 때 처리군은 대조군에 비해  $e^\beta$ 만큼 위험이 감소함을 의미한다.

## 2.2. 가속화 위험 회귀모형

가속화 위험 회귀모형 (accelerated hazards regression model; AH model)은 두 집단의 생존함수가 교차되어있는 자료에 유용한 모형이며 (Chen과 Wang, 2000a) (2.3)과 같이 표현된다.

$$\lambda(t;x) = \lambda_0(t e^{x^T \beta})$$

$$\text{즉, } S(t;x) = S_0(t e^{x^T \beta}) e^{-x^T \beta} \quad (2.3)$$

여기서  $\lambda_0(\cdot)$ 는 기저 위험함수이고  $S_0(\cdot)$ 는 기저 생존함수이다.  $\beta = (\beta_1, \dots, \beta_p)^T$ 는 공변량 (covariate) 벡터  $x = (x_1, \dots, x_p)^T$ 에 대응하는  $p \times 1$  차원의 회귀계수 벡터이다. AH 모형은 대조군 (control group)에 비해 처리군의 위험률 (hazard rate)이 시간에 따라 얼마나 가속화 (acceleration) 또는 감속화 (deceleration) 하는지를 설명할 수 있다. 다시 말하면  $\beta > 0$ 일 때 시간에 대한 위험함수가 증가형태라면 처리군의 위험률이 빠르게 증가하고 (harmful), 위험함수가 감소형태라면 위험률이 빠르게 감소함 (beneficial)을 의미한다. 반대로  $\beta < 0$ 일 때 시간에 대한 위험함수가 증가형태라면 처리군의 위험률이 느리게 증가하고 (beneficial), 위험함수가 감소형태라면 위험률이 느리게 감소함 (harmful)을 의미한다 (Chen과 Wang, 2000b). 즉,  $\beta > 0$  [ $\beta < 0$ ]이면 생존

시간  $T$ 에 대한 위험률을 가속화 [감속화] 한다. PH 모형, AFT 모형 그리고 AH 모형에서  $\beta$ 의 의미는 표 2.1과 같다.

표 2.1. 세 모형 (PH, AFT, AH 모형)의 회귀계수 의미의 비교

Model	Effect
PH	$\beta_{PH} > 0$ Treatment proportionally increases risk/hazard by a factor of $e^{\beta_{PH}}$
	$\beta_{PH} < 0$ Treatment proportionally decreases risk/hazard by a factor of $e^{\beta_{PH}}$
AFT	$\beta_{AFT} > 0$ Treatment decelerates failure time of the survival function by a factor of $e^{\beta_{AFT}}$
	$\beta_{AFT} < 0$ Treatment accelerates failure time of the survival function by a factor of $e^{\beta_{AFT}}$
AH	$\beta_{AH} > 0$ Treatment accelerates the risk/hazard by a factor of $e^{\beta_{AH}}$
	$\beta_{AH} < 0$ Treatment decelerates the risk/hazard by a factor of $e^{\beta_{AH}}$
All	$\beta_{PH} = \beta_{AFT} = \beta_{AH} = 0$ Treatment does not have an effect

PH 모형과 AFT 모형은 즉각적인 (immediate) 치료효과를 가정한다. 이는  $t = 0$ 일 때 두 그룹의 위험률이 다름을 의미하고 (Co, 2010) 아래의 표 2.2에서 확인 가능하다.

표 2.2. 세 모형 (PH, AFT, AH 모형)의 위험함수의 비교

Model	PH	AFT	AH
Hazards function $\lambda(t; x)$	$\lambda_0(t)e^{xT\beta}$	$\lambda_0(te^{-xT\beta})e^{-xT\beta}$	$\lambda_0(te^{xT\beta})$

즉, 두 모형에서  $x=0$  또는 1인 이진형인 경우,  $\lambda_1(0) \neq \lambda_2(0)$ 가 된다. 하지만 잘 무작위 배정된 임상설계라면 일반적으로 임상시험에서 시험을 시작할 때 두 그룹의 위험률은 동일하며 시간이 흐르면서 치료효과가 서서히 나타난다. AH 모형은 이러한 즉각적인 치료효과를 가정하지 않으므로 PH 모형과 AFT 모형이 가진 단점을 극복할 수 있다. 이는 AH 모형에서  $t=0$ 이면 항상  $\lambda(t; x)=0$ 인 특성에 의해 설명 가능하다. 즉,  $x=0$  또는 1인 이진형 자료인 경우,  $\lambda_1(0)=\lambda_2(0)$ 이 된다 (Chen과 Wang, 2000b).

하지만 AH 모형도 교차된 위험자료만을 다루는 것은 아니다. 예를 들어, 생존시간  $T$ 가 와이블 분포 즉, 기저위험함수  $\lambda_0(s)=\theta_0\phi s^{\phi-1}$  (여기서  $\theta_0$ : 척도모수,  $\phi$ : 형상모수)를 고려하자. 그러면 AH 모형의 위험함수는  $\lambda(t; x)=\theta_0\phi t^{\phi-1}e^{xT\beta(\phi-1)}$  이고  $x$ 가 0 또는 1인 이진형일 때 위험비는 (2.4)와 같이 얻어진다.

$$HR(t; x_1, x_2) = \frac{\lambda(t; x_1 = 1)}{\lambda(t; x_1 = 0)} = e^{\beta(\phi-1)} \quad (2.4)$$

위험비가  $t$ 에 의존하지 않으므로 비례위험 모형이 된다. 뿐만 아니라  $T$ 가 와이블 분포일 때 PH, AFT 그리고 AH 모형의 치료효과  $\beta$ 들은 (2.5)과 같은 관계가 성립한다. 즉, AH 모형은 비례위험 모형이 된다.

$$\beta_{PH} = -\phi\beta_{AFT} = (\phi-1)\beta_{AH} \quad (2.5)$$

여기서  $\phi$ 는 와이블의 형상모수 (shape parameter)이다.

### 2.2.1 로그-로지스틱 분포에 기반한 AH 모형

본 논문에서 사용하는 로그-로지스틱 분포의 기저 위험함수, 기저 생존함수 그리고 기저 확률분포함수는 각각 (2.6)과 같다.

$$\lambda_0(t) = \frac{\theta_0 \phi t^{\phi-1}}{1 + \theta_0 t^\phi}, \quad S_0(t) = \frac{1}{1 + \theta_0 t^\phi}, \quad f_0(t) = \frac{\theta_0 \phi t^{\phi-1}}{(1 + \theta_0 t^\phi)^2} \quad (t > 0) \quad (2.6)$$

여기서  $\theta_0$ 와  $\phi$ 는 각각 척도모수 (scale parameter)와 형상모수 (shape parameter)로 모두 양의 실수 값을 갖는다.  $t$ 가 작을 때 위험함수는 확률밀도함수와 유사하다 (Clark와 El-Taha, 2015). 생존분석에서 주로 사용되는 와이블 분포와 달리 로그-로지스틱 분포는 비-단조 위험 함수 (non-monotonic hazard function)이다. 이는 두 분포의 척도모수의 값을 1로 고정한 후 형상모수의 값의 변화에 따른 형태에서 확인 가능하다 (그림 2.1).

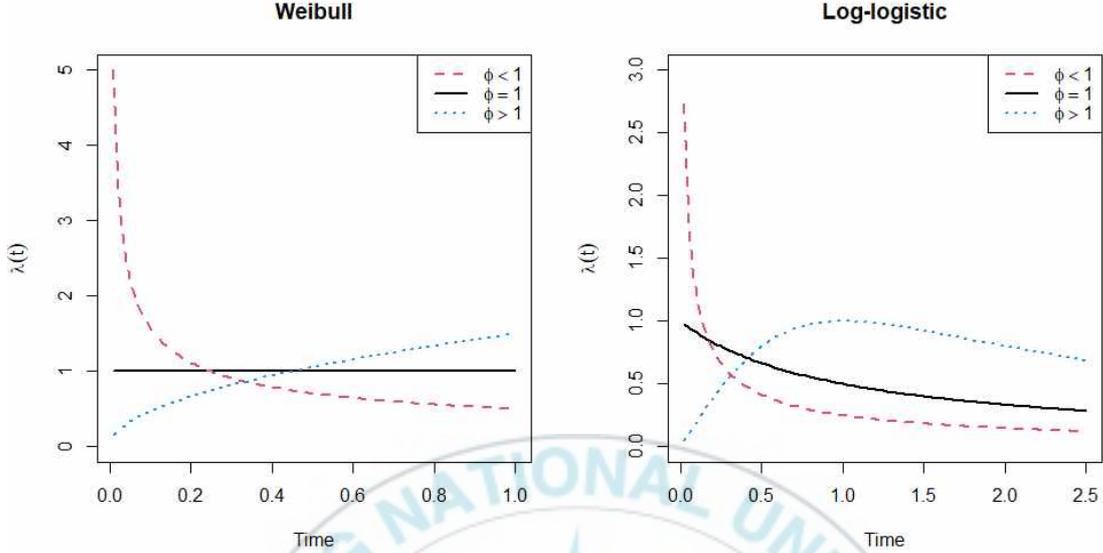


그림 2.1. 위험함수: 와이블 분포 (좌)와 로그-로지스틱 분포 (우)

그림 2.1에서 와이블 분포의 경우 형상모수  $\phi$ 의 변화에 관계없이 항상 단조적인 형태를 나타내고 있다. 하지만 로그-로지스틱 분포의 경우  $\phi$ 가 1보다 클 때 봉우리가 하나이고 (unimodal),  $\phi$ 가 1보다 작으면 단조롭게 감소 (monotone decreasing)하는 것이 특징이다 (Kartsonaki, 2016). 그리고  $T$ 가 로그-로지스틱 분포를 가질 때 대응하는 AH 모형의 위험함수와 위험비는 각각 (2.7), (2.8)과 같다.

$$\lambda(t;x) = \frac{\theta_0 \phi t^{\phi-1} e^{x^T \beta (\phi-1)}}{1 + \theta_0 t^\phi e^{x^T \beta \phi}} \quad (2.7)$$

$$HR(t; x_1, x_2) = \frac{\lambda(t; x_1 = 1)}{\lambda(t; x_2 = 0)} = \frac{(1 + \theta_0 t^\phi) e^{\beta(\phi-1)}}{1 + \theta_0 t^\phi e^{\beta \phi}} \quad (2.8)$$

위험비가  $t$ 에 의존하므로 위험함수는 비비례 (non-PH) 형태가 됨을 알 수 있다. 즉,

생존함수 또는 위험함수가 교차된 자료를 다룰 수 있음을 의미한다. 따라서 본 논문에서는 교차된 자료를 다루기 위해 기저 위험함수로서 로그-로지스틱 분포를 사용한 AH 모형으로 분석하고자한다.

## 2.2.2 AH 회귀모형의 가능도 추정

$T_i(i=1, \dots, n)$ 를  $i$ 번째 관측 값에 대한 생존시간 (survival time)이라 하고,  $C_i$ 는 각  $T_i$ 에 대응하는 중도절단 시간 (censoring time)이라고 하면 관측 가능한 확률변수 (observable random variables)는 다음과 같다.

$$Y_i = \min(T_i, C_i) \text{ and } \delta_i = I(T_i \leq C_i)$$

여기서  $I(\cdot)$ 는 지시함수 (indicator function)이며,  $\delta_i$ 는 중도절단 지시함수 (censoring indicator)이다. 본 연구에서는 모형추론을 위해 공변량  $x_i$ 가 주어졌을 때  $C_i(i=1, \dots, n)$ 에 대한 다음과 같은 통상적인 두 가지 조건부 가정을 허용한다 (Lawless, 2003):

가정 1: 순서쌍  $(T_i, C_i)$ 가 조건부 독립이고,  $T_i$ 와  $C_i$  또한 조건부 독립이다.

가정 2:  $C_i$ 는  $T_i$ 에 대하여 조건부 무정보적 (conditionally non-informative)이다.

여기서 무정보적이란 중도절단 시간의 분포가 생존시간 분포의 관심 모수에 의존하지 않음을 의미한다 (Klein과 Moeschberger, 2006).

AH 모형의 기저함수로 LLG 분포를 적용한 모형을 LLG-AH 모형이라고 하자.  $(y_i, \delta_i, x_i)(i=1, \dots, n)$ 에 근거한 모수  $\theta = (\beta^T, \theta_0, \phi)^T$ 에 대한 AH 모형 (2.7)의 로그 가능

도 함수  $l(\beta, \theta_0, \phi)$ 는 (2.9)와 같다.

$$l(\beta, \theta_0, \phi; y, \delta, x) = \sum_{i=1}^n \delta_i \log(\theta_0) + \sum_{i=1}^n \delta_i \log(\phi) + \sum_{i=1}^n \delta_i (\phi - 1) \log(y_i) + \sum_{i=1}^n \delta_i x_i^T \beta (\phi - 1) - \sum_{i=1}^n \delta_i \log(1 + \theta_0 y_i^\phi e^{x_i^T \beta \phi}) - \sum_{i=1}^n \log(1 + \theta_0 y_i^\phi e^{x_i^T \beta \phi}) e^{-x_i^T \beta} \quad (2.9)$$

위 식에 대한 이 로그-가능도 함수를 이용한 최대 가능도 (maximum likelihood; ML) 추정 방정식은 다음과 같다.

$$\begin{aligned} \frac{\partial l}{\partial \beta} &= \sum_i \delta_i x_i (\phi - 1) - \sum_i \delta_i \frac{\theta_0 t_i^\phi e^{x_i^T \beta \phi} x_i \phi}{1 + \theta_0 t_i^\phi e^{x_i^T \beta \phi}} - \sum_i \frac{\theta_0 t_i^\phi e^{x_i^T \beta (\phi - 1)} x_i \phi}{1 + \theta_0 t_i^\phi e^{x_i^T \beta \phi}} + \sum_i \log(1 + \theta_0 t_i^\phi e^{x_i^T \beta \phi}) e^{-x_i^T \beta} x_i = 0 \\ \frac{\partial l}{\partial \theta_0} &= \sum_i \frac{\delta_i}{\theta_0} - \sum_i (\delta_i + e^{-x_i^T \beta}) \frac{t_i^\phi e^{x_i^T \beta \phi}}{1 + \theta_0 t_i^\phi e^{x_i^T \beta \phi}} = 0 \\ \frac{\partial l}{\partial \phi} &= \sum_i \frac{\delta_i}{\phi} + \sum_i \delta_i \log t_i + \sum_i \delta_i x_i^T \beta - \sum_i \delta_i \frac{\theta_0 t_i^\phi e^{x_i^T \beta \phi} (\log t_i + x_i^T \beta)}{1 + \theta_0 t_i^\phi e^{x_i^T \beta \phi}} - \sum_i e^{-x_i \beta} \frac{\theta_0 t_i^\phi e^{x_i^T \beta \phi} (\log t_i + x_i^T \beta)}{1 + \theta_0 t_i^\phi e^{x_i^T \beta \phi}} = 0 \end{aligned}$$

$(\hat{\beta}, \hat{\theta}_0, \hat{\phi})$ 의 분산-공분산을 구하기 위한  $\theta = (\beta^T, \theta_0, \phi)^T$ 의 음의 헤시안 행렬 (negative Hessian matrix)은 아래와 같으며 상세한 식은 부록 A에 있다.

$$H = -\frac{\partial^2 l}{\partial \theta \partial \theta^T} = \begin{pmatrix} -\frac{\partial^2 l}{\partial \beta \partial \beta^T} & -\frac{\partial^2 l}{\partial \beta \partial \theta_0} & -\frac{\partial^2 l}{\partial \beta \partial \phi} \\ -\frac{\partial^2 l}{\partial \theta_0 \partial \beta^T} & -\frac{\partial^2 l}{\partial \theta_0^2} & -\frac{\partial^2 l}{\partial \theta_0 \partial \phi} \\ -\frac{\partial^2 l}{\partial \phi \partial \beta^T} & -\frac{\partial^2 l}{\partial \phi \partial \theta_0} & -\frac{\partial^2 l}{\partial \phi^2} \end{pmatrix}$$

모수를 추정을 위해 R 프로그램에서 repeat()문과 Python 프로그램에서 while과 if문을 사용하였으며 로그 가능도 함수를 미분한 식을 이용한 뉴턴-랩슨 방법 (Newton-Raphson method)을 이용하였다.

### 2.2.3 시간 효과를 허용하는 AH 회귀모형의 가능도 추정

시간 효과 (time effect)는 시간에 따라 그룹 간 효과가 다를 수 있음을 의미한다 (Calsavara 등, 2019). 본 연구에서는 AH 모형에서 로그-로지스틱 분포의 형상모수  $\phi$ 에  $x = I(\text{group} = 1)$ 인 그룹 효과를 부여하였다.

$$\begin{aligned} \phi &= \alpha_0 I(\text{group} = 0) + \alpha_1 I(\text{group} = 1) \\ &= \alpha_0(1 - x) + \alpha_1 x \end{aligned} \quad (2.10)$$

그러면 시간 효과는 (2.11)과 같이 표현된다.

$$\begin{aligned} t^\phi &= t^{\alpha_0(1-x) + \alpha_1 x} \\ &= \begin{cases} t^{\alpha_0} & x = 0 \\ t^{\alpha_1} & x = 1 \end{cases} \end{aligned} \quad (2.11)$$

즉,  $t^\phi$ 는 그룹의 범주가 0일 때  $t^{\alpha_0}$ 값을 가지고 범주가 1일 때는  $t^{\alpha_1}$ 값을 가진다. 이때 시간 효과를 허용하는 로그-로지스틱 분포의 기저 위험함수, 기저 생존함수와 기저 확률밀도함수는 각각 아래와 같다 (2.12).

$$\begin{aligned}\lambda_0(t; x) &= \frac{\theta_0 [\alpha_0(1-x) + \alpha_1 x] t^{\alpha_0(1-x) + \alpha_1 x - 1}}{1 + \theta_0 t^{\alpha_0(1-x) + \alpha_1 x}}, \\ S_0(t; x) &= \frac{1}{1 + \theta_0 t^{\alpha_0(1-x) + \alpha_1 x}}, \\ f_0(t; x) &= \frac{\theta_0 [\alpha_0(1-x) + \alpha_1 x] t^{\alpha_0(1-x) + \alpha_1 x - 1}}{(1 + \theta_0 t^{\alpha_0(1-x) + \alpha_1 x})^2}.\end{aligned}\tag{2.12}$$

AH 모형의 기저 위험함수로 시간 효과를 허용하는 LLG 분포를 사용한 모형을 LLGT-AH 모형이라고 하자. 이를 이용한  $(y_i, \delta_i, x_i) (i = 1, \dots, n)$ 에 근거하는 모수  $\theta = (\beta^T, \theta_0, \alpha_0, \alpha_1)^T$ 에 대한 위험함수  $\lambda(t; x)$ 는 (2.13)과 같다.

$$\lambda(t; x) = \frac{\theta_0 [\alpha_0(1-x) + \alpha_1 x] (te^{x^T \beta})^{\alpha_0(1-x) + \alpha_1 x - 1}}{1 + \theta_0 (te^{x^T \beta})^{\alpha_0(1-x) + \alpha_1 x}}\tag{2.13}$$

AH 모형 (2.13)의 로그 가능도 함수  $l(\beta, \theta_0, \alpha_0, \alpha_1)$ 는 (2.14)와 같이 얻어진다.

$$\begin{aligned}
l(\beta, \theta_0, \alpha_0, \alpha_1; y, \delta) &= \sum_{i=1}^n \delta_i \log(\theta_0) + \sum_{i=1}^n \delta_i \log[\alpha_0(1-x_i) + \alpha_1 x_i] \\
&+ \sum_{i=1}^n \delta_i [\alpha_0(1-x_i) + \alpha_1 x_i - 1] \log(y_i) + \sum_{i=1}^n \delta_i x_i^T \beta [\alpha_0(1-x_i) + \alpha_1 x_i - 1] \\
&- \sum_{i=1}^n (\delta_i + e^{-x_i^T \beta}) \log(1 + \theta_0 t_i^{[\alpha_0(1-x_i) + \alpha_1 x_i]} e^{x_i^T \beta [\alpha_0(1-x_i) + \alpha_1 x_i]}) \quad (2.14)
\end{aligned}$$

이러한 로그-가능도 함수를 이용한 ML 추정 방정식은 다음과 같다.

$$\begin{aligned}
\frac{\partial l}{\partial \beta} &= \sum_i \delta_i x_i [\alpha_0(1-x_i) + \alpha_1 x_i - 1] - \sum_i \delta_i \frac{\theta_0 t_i^{[\alpha_0(1-x_i) + \alpha_1 x_i]} e^{x_i^T \beta [\alpha_0(1-x_i) + \alpha_1 x_i]} x_i [\alpha_0(1-x_i) + \alpha_1 x_i]}{1 + \theta_0 t_i^{[\alpha_0(1-x_i) + \alpha_1 x_i]} e^{x_i^T \beta [\alpha_0(1-x_i) + \alpha_1 x_i]}} \\
&- \sum_i \frac{\theta_0 t_i^{[\alpha_0(1-x_i) + \alpha_1 x_i]} e^{x_i^T \beta [\alpha_0(1-x_i) + \alpha_1 x_i - 1]} x_i [\alpha_0(1-x_i) + \alpha_1 x_i]}{1 + \theta_0 t_i^{[\alpha_0(1-x_i) + \alpha_1 x_i]} e^{x_i^T \beta [\alpha_0(1-x_i) + \alpha_1 x_i]}} \\
&+ \sum_i \log(1 + \theta_0 t_i^{[\alpha_0(1-x_i) + \alpha_1 x_i]} e^{x_i^T \beta [\alpha_0(1-x_i) + \alpha_1 x_i]}) e^{-x_i^T \beta} x_i = 0 \\
\frac{\partial l}{\partial \theta_0} &= \sum_i \frac{\delta_i}{\theta_0} - \sum_i (\delta_i + e^{-x_i^T \beta}) \frac{t_i^{[\alpha_0(1-x_i) + \alpha_1 x_i]} e^{x_i^T \beta [\alpha_0(1-x_i) + \alpha_1 x_i]}}{1 + \theta_0 t_i^{[\alpha_0(1-x_i) + \alpha_1 x_i]} e^{x_i^T \beta [\alpha_0(1-x_i) + \alpha_1 x_i]}} = 0 \\
\frac{\partial l}{\partial \alpha_0} &= \sum_i \frac{\delta_i (1-x_i)}{\alpha_0(1-x_i) + \alpha_1 x_i} + \sum_i \delta_i \log t_i (1-x_i) + \sum_i \delta_i x_i^T \beta (1-x_i) \\
&- \sum_i (\delta_i + e^{-x_i^T \beta}) \frac{\theta_0 t_i^{[\alpha_0(1-x_i) + \alpha_1 x_i]} e^{x_i^T \beta [\alpha_0(1-x_i) + \alpha_1 x_i]} (1-x_i) (\log t_i + x_i^T \beta)}{1 + \theta_0 t_i^{[\alpha_0(1-x_i) + \alpha_1 x_i]} e^{x_i^T \beta [\alpha_0(1-x_i) + \alpha_1 x_i]}} = 0 \\
\frac{\partial l}{\partial \alpha_1} &= \sum_i \frac{\delta_i x_i}{\alpha_0(1-x_i) + \alpha_1 x_i} + \sum_i \delta_i x_i \log t_i + \sum_i \delta_i x_i^2 \beta \\
&- \sum_i (\delta_i + e^{-x_i^T \beta}) \frac{\theta_0 t_i^{[\alpha_0(1-x_i) + \alpha_1 x_i]} e^{x_i^T \beta [\alpha_0(1-x_i) + \alpha_1 x_i]} x_i (\log t_i + x_i^T \beta)}{1 + \theta_0 t_i^{[\alpha_0(1-x_i) + \alpha_1 x_i]} e^{x_i^T \beta [\alpha_0(1-x_i) + \alpha_1 x_i]}} = 0
\end{aligned}$$

그리고 관심모수  $(\hat{\beta}, \hat{\theta}_0, \hat{\alpha}_0, \hat{\alpha}_1)$ 의 MLE의 분산-공분산을 구하기 위한  $\theta = (\beta^T, \theta_0, \alpha_0, \alpha_1)^T$ 의 음의 헤시안 행렬은 아래와 같으며 그에 해당하는 상세한 식은 부록 B에 주어진다.

$$H = -\frac{\partial^2 l}{\partial \theta \partial \theta^T} = \begin{pmatrix} -\frac{\partial^2 l}{\partial \beta \partial \beta^T} & -\frac{\partial^2 l}{\partial \beta \partial \theta_0} & -\frac{\partial^2 l}{\partial \beta \partial \alpha_0} & -\frac{\partial^2 l}{\partial \beta \partial \alpha_1} \\ -\frac{\partial^2 l}{\partial \theta_0 \partial \beta^T} & -\frac{\partial^2 l}{\partial \theta_0^2} & -\frac{\partial^2 l}{\partial \theta_0 \partial \alpha_0} & -\frac{\partial^2 l}{\partial \theta_0 \partial \alpha_1} \\ -\frac{\partial^2 l}{\partial \alpha_0 \partial \beta^T} & -\frac{\partial^2 l}{\partial \alpha_0 \partial \theta_0} & -\frac{\partial^2 l}{\partial \alpha_0^2} & -\frac{\partial^2 l}{\partial \alpha_0 \partial \alpha_1} \\ -\frac{\partial^2 l}{\partial \alpha_1 \partial \beta^T} & -\frac{\partial^2 l}{\partial \alpha_1 \partial \theta_0} & -\frac{\partial^2 l}{\partial \alpha_1 \partial \alpha_0} & -\frac{\partial^2 l}{\partial \alpha_1^2} \end{pmatrix}$$

### 2.3. 제한된 평균 생존시간

생존자료 분석에서 두 그룹 간 생존함수의 비교를 위해 로그-순위 검정 (log-rank test)을 자주 사용한다. 하지만 이 로그-순위 검정은 비례위험 가정이 만족될 때 사용하는 검정 방법으로 비례위험을 위반하는 자료에서는 잘못된 분석 결과를 초래하는 단점이 있다. 다시 말하면 실제로 두 그룹 간 유의한 차이가 있으나 차이가 없다는 결론을 줄 수 있다. 잘못된 추론 결과를 초래할 수 있는 자료 형태의 한 예로 생존함수 또는 위험함수가 교차된 자료가 있다. 이에 대한 대안으로 제한된 평균 생존시간을 사용할 수 있다. 제한된 평균 생존시간 (restricted mean survival time; RMST)은 제한된 시간 내의 평균 생존시간을 의미한다. 다시 말해 시험 시작 시점부터 관심이 있는 시점  $\tau$ 까지 추적했을 때 사건이 발생할 때까지 평균적으로 걸린 시간을 의미하며 (2.15)와 같이 표현된다.

$$\mu(\tau) = E(T^*) = \int_0^{\tau} S(t)dt \quad (2.15)$$

여기서  $T^* = \min(T, \tau)$ 이라 할 때  $\tau$ 는 무한대보다 작은 상수이다 ( $\tau < \infty$ ). RMST는 다음과 같은 장점이 있다.

- 1) 관심이 있는 기간 내의 평균 수명을 알 수 있어 임상학적 측면에서 매우 유용하며 해석 또한 쉽다 (Lee, 2020).
- 2) 평균 생존시간이나 중위수 생존시간은 중도 절단된 자료가 많은 경우 (즉, heavy censoring) 추정이 불가능하여 분석에 제한이 있다. 하지만 RMST는 중도 절단 비율이 높아도 추정이 가능한 장점이 있다 (Tian 등, 2014).
- 3) 비례위험 가정을 위반했을 때에도 여전히 유효하며 비례위험이 만족되지 않는 자료에 대해 로그-순위 검정보다 더 높은 검정력을 가진다 (Tian 등, 2018).

RMST의 분석을 위한 R 패키지로 “survRM2”를 이용할 수 있다.

RMST를 통해 제한된 평균 손실 시간 (restricted mean time lost; RMTL)도 계산가능하다. RMTL은 RMST의 반대 의미로 제한된 시간 내의 평균 손실 시간을 의미한다. 즉, 관심 있는  $\tau$ 시점까지 평균적으로 얼마만큼의 수명을 손실하였는지를 나타내며 (2.16)과 같다.

$$L(\tau) = \tau - \mu(\tau) = \int_0^{\tau} [1 - S(t)]dt \quad (2.16)$$

## 제 3장 예증

### 3.1. 신장 감염 자료

신장 감염 자료 (Klein와 Moeschberger, 1997)는 총 119명 ( $n=119$ )을 대상으로 신장 감염이 발생하는데 걸린 시간을 나타내는 생존자료로 생존시간, 감염 발생 여부, 그룹 변수로 구성되어 있다. 생존시간은 감염 발생까지 걸린 시간을 개월 (month) 단위로 표현하였다. 감염 발생 여부는 신장 감염이 발생했다면 1, 발생하지 않았다면 0으로 범주화하였고 연구 종료 시점 (end of study in date)에 감염되지 않은 대상자들을 중도절단으로 하였다. 119명의 대상자 중 93명이 감염되지 않았으므로 중도절단 비율은 78.2%이다. 그룹 변수는 주관심 공변량 ( $x$ )으로 도뇨관 (catheter)을 설치한 방법에 따라 두 그룹으로 나누었으며, 처리군 (treatment group;  $x=1$ )은 도뇨관을 피부를 통하여 경피적으로 (percutaneously) 설치한 그룹이고 대조군 (control group;  $x=0$ )은 도뇨관을 외과적으로 (surgically) 설치한 그룹이다. 이 자료의 카플란-마이어 (Kaplan와 Meier, 1958) 생존곡선은 그림 3.1과 같다.

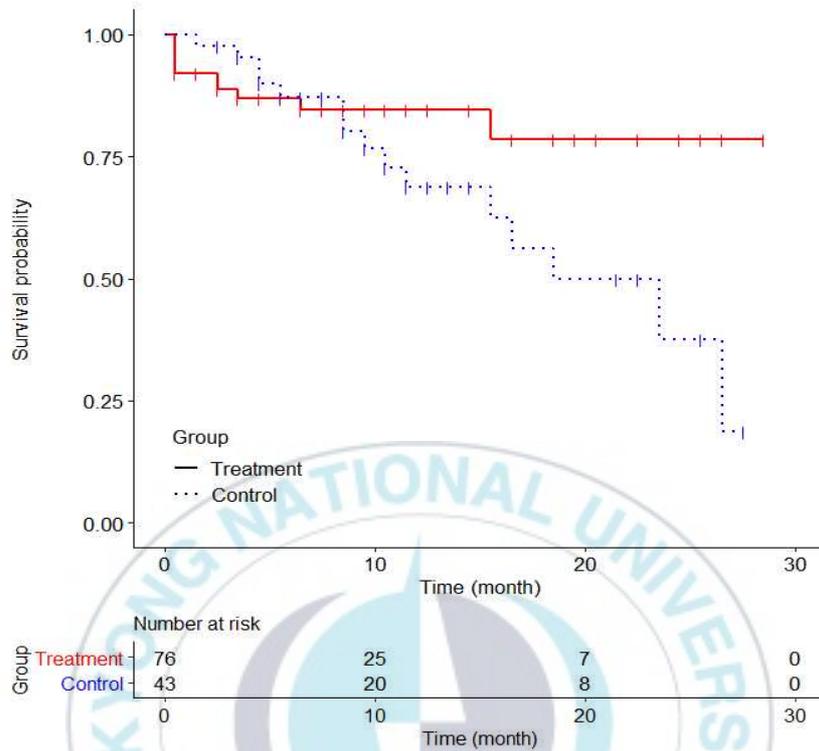


그림 3.1. 신장 감염 자료: 카플란-마이어 생존곡선

먼저 그림 3.1 하단의 위험 대상자 수에 대한 표를 통해 연구시작부터 10개월 동안 감염사건 발생으로 인해 위험에 처한 대상자의 수가 절반이상으로 감소하여 가장 많이 감소하는 것을 볼 수 있으며 연구 종료시점에는 위험에 처한 사람이 아무도 존재하지 않음을 확인할 수 있다. 카플란-마이어 생존곡선에서 두 그룹의 생존함수가 이른 시기에 교차되는 것을 볼 수 있다. 약 5개월 이전까지는 처리군이 대조군보다 생존율이 낮지만 그 이후에는 처리군이 더 높은 생존율을 보인다. 생존시간의 중앙값은 처리군에는 존재하지 않았으며 대조군은 18.5개월이다. 평균 생존시간의 경우 처리군은 23.3개월이고 대조군은 18.6개월이다. 즉, 대조군의 경우에 신장 감염이 더 빠르게 일어난다고 할 수 있다. 그리고 두 그룹의 생존곡선이 비례하지 않음

을 보이기 위해 스콘필드 잔차 (Schoenfeld residual)에 근거한 비례성 검정을 실시하였다. 그 결과는 자유도가 1인 카이제곱 검정통계량은 8.62, p-value는 0.003으로 두 그룹이 비례하지 않음을 알 수 있으며, Cox's PH 모형에 대한 잔차도 (residual plot)는 그림 3.2와 같다. 시간 경과에 따라 곡선이 평평하지 않고 기울어져있는 형태를 통해 신장 감염 자료가 비례위험을 따르지 않음을 확인할 수 있다.

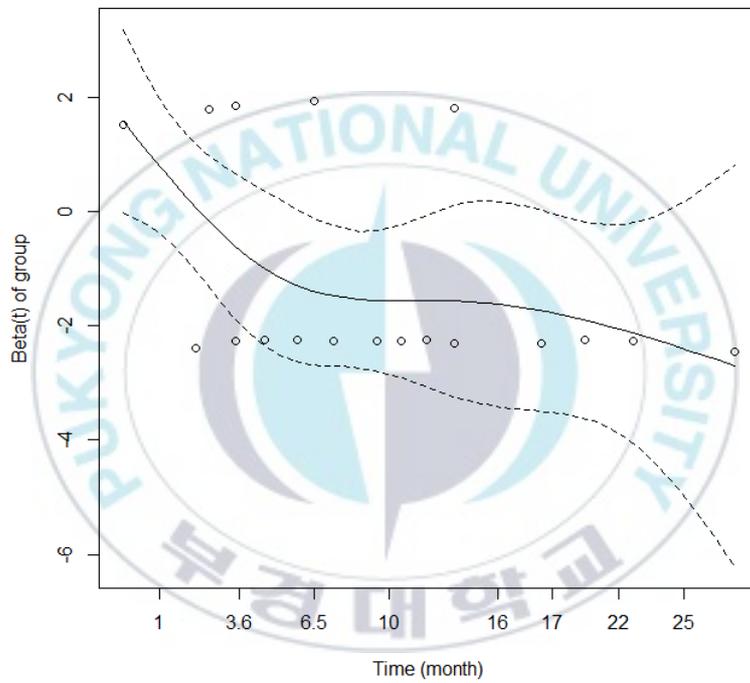


그림 3.2. 신장 감염 자료: 비례위험 모형에 대한 잔차도

### 3.1.1. 모형 적합

3.1절의 신장 감염 자료의 분석을 위해 고려한 다섯 가지 위험모형은 다음과 같다.

1) LLG-AH 모형

$$\lambda(t;x) = \frac{\theta_0 \phi (te^{xT\beta})^{\phi-1}}{1 + \theta_0 (te^{xT\beta})^{\phi}}$$

2) LLGT-AH 모형

$$\lambda(t;x) = \frac{\theta_0 \{ \alpha_0 (1-x) + \alpha_1 x \} (te^{xT\beta})^{\alpha_0(1-x) + \alpha_1 x - 1}}{1 + \theta_0 (te^{xT\beta})^{\alpha_0(1-x) + \alpha_1 x}}$$

3) Wei-AH 모형

$$\lambda(t;x) = \theta_0 \phi (te^{xT\beta})^{\phi-1}$$

4) Cox's PH 모형

$$\lambda(t;x) = \lambda_0(t) e^{xT\beta}$$

5) AFT 모형

$$\lambda(t;x) = \lambda_0 (te^{-xT\beta}) e^{-xT\beta}$$

대응하는 추정 결과는 표 3.1과 같다. 여기서 AH 모형의 모수는 2.2절의 가능도 추정법을 이용하였다. 세 가지의 AH 모형 (LLG-AH, LLGT-AH, Wei-AH 모형)에 있는 회귀모수  $\beta$ 의 최대가능도 (maximum likelihood) 추정값을 구하기 위해 뉴턴-랩슨 방법을 사용하였다. 이러한 방법의 알고리즘 구현을 위해 본 논문에서는 R 프로그램에서 repeat()문과 Python 프로그램에서 while과 if문을 사용하였다. 참고적으로 Cox's PH 모형에서는 Breslow 방법에 근거한 R의 coxph() 함수를, AFT 모형에서는 비모수적 방

법인 붓스트랩 (Bootstrap) 방법에 근거한 R의 `aftgee()` 함수 (Chiou 등, 2014)를 이용하였다. 2.2절의 가능도 추정법을 이용하여 모수를 추정한 결과는 표 3.1과 같다.

표 3.1. 신장 감염 자료: 세 모형 (AH, PH, AFT 모형)을 이용한 분석

Model	Est (SE)				
	$\beta$ : Treatment ( $x = 1$ )	$\theta_0$	$\phi$	$\alpha_0$	$\alpha_1$
LLG-AH	3.731 (0.365)	0.002 (0.002)	2.147 (0.381)	—	—
LLGT-AH	3.670 (0.346)	0.003 (0.003)	—	1.953 (0.389)	2.294 (0.404)
Wei-AH	0.075 (0.524)	0.002 (0.0005)	1.850 (0.139)	—	—
Cox's PH	-0.618 (0.398)	—	—	—	—
AFT	-0.261 (0.394)	—	—	—	—

(Note) Est = estimate; SE = estimated standard error;  $\beta$ : regression parameter;  $\theta_0$ : scale parameter;  $\phi$ : shape parameter;  $\alpha_0$ : time effect of control group;  $\alpha_1$ : time effect of treatment group

표 3.1에서 Wei-AH는 AH 모형의 기저위험함수로 와이블 분포를 사용한 것으로 그룹효과인  $\beta$ 가 Wald 검정 ( $H_0: \beta = 0$ ,  $Z = \hat{\beta}/SE(\hat{\beta})$ )에 의해 유의수준 5%에서 유의하지 않은 결과를 보인다. 그리고 PH 모형과 100번 반복의 붓스트랩 방법에 근거한 AFT 모형 또한 유의하지 않은 결과를 보인다. LLG-AH 모형과 LLGT-AH 모형 모두 그룹 간 효과인  $\beta$ 가 유의한 것으로 나타나며 각각의  $\hat{\beta}$ 은 모두 양수임을 확인할 수 있다.

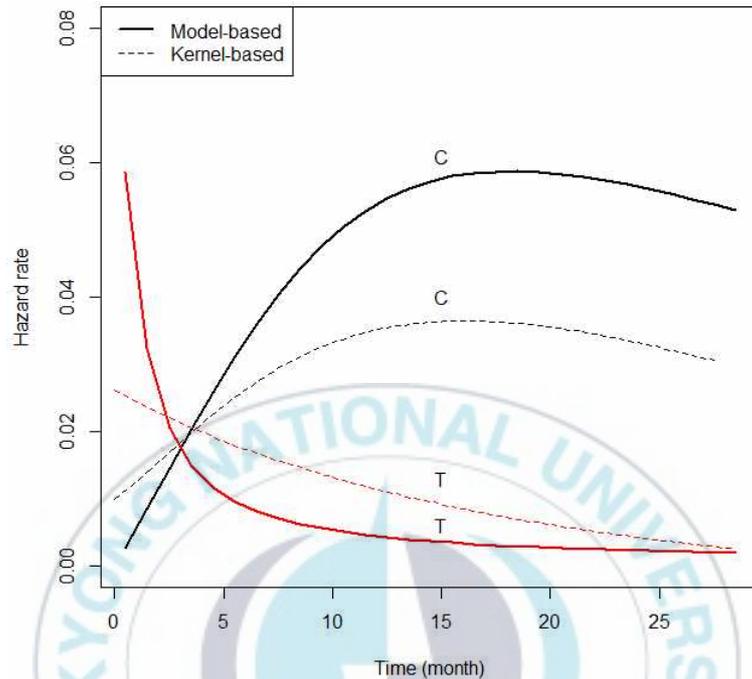


그림 3.3. 신장 감염 자료: 평활 및 AH 모형에 근거한 위험률 곡선  
(Note) T: 처리군; C: 대조군

그림 3.3에서 굵은 실선은 본 논문에서 제안하는 LLG-AH 모형에 근거한 위험률 곡선이고 얇은 점선은 평활 위험률 곡선 (smooth hazard function)이다. 평활 위험률 곡선을 위해 R의 `muhaz()` 함수를 이용하였다. 두 곡선이 유사한 형태를 보이며 처리군의 곡선이 감소형태임을 볼 수 있다. 따라서 처리군은 시간이 지날수록 위험을 빠르게 감소시킨다는 결론을 내릴 수 있다. 반면 대조군은 증가형태이므로 시간이 지날수록 위험이 증가함을 의미한다. 따라서 제안된 AH 모형은 적절한 모형으로 기대된다.

### 3.1.2. 유의성 검토

두 그룹 간 차이의 유의성 검토를 전반적으로 비교하기 위하여 여덟 가지 검정을 실시하였다. 먼저 두 그룹 간 생존율의 동등성 여부를 검정하기 위해 두 그룹의 생존함수가 비례하는 경우에 많이 사용되는 Log Rank 검정, Log Rank 검정에 가중치를 주어 두 그룹의 생존함수가 비례하지 않는 경우에 사용되는 Gehan 검정과 Tarone-Ware 검정을 실시하였다. 세 가지 AH 모형 (LLG-AH, LLGT-AH, Wei-AH 모형), Cox's PH 모형 그리고 AFT 모형은 추정된 모수를 이용하여 Wald 검정을 실시하였다. 유의수준 5%에서 검정을 실시한 결과 표 3.1에서 확인한 바와 같이 LLG-AH 모형의 검정과 LLGT-AH 모형의 검정에서  $\beta$ 가 유의하게 판정되었다. 반면, 고전적인 세 가지 검정법 (Log-rank, Gehan, Tarone-Ware 방법)과 표준적 모형 (Wei-AH, Cox's PH and AFT)은 5%에서 유의하지 않게 판정되었다.

표 3.2. 신장 감염 자료: 유의성 검토를 위한 통계량과 p-값

Method	Statistics (p-value)
Log-rank	2.530 (0.112)
Gehan	1.387 (0.239)
Tarone-Ware	1.912 (0.167)
LLG-AH	10.73 (<0.001)
LLGT-AH	10.79 (<0.001)
Wei-AH	0.142 (0.887)
Cox's PH	-1.553 (0.120)
AFT	-0.664 (0.507)

### 3.1.3. 모형 검토

본 자료에 대한 카플란-마이어 생존곡선에 LLG-AH 모형과 LLGT-AH 모형을 적합한 그래프는 그림 3.4와 같다. LLG-AH 모형에서는 두 그룹의 생존함수가 맞닿은 부분을 교차지점이라고 잘못 표현하였지만 LLGT-AH 모형은 두 생존함수가 교차된 부분을 확실하게 표현하였다. 그리고 LLGT-AH 모형이 교차 이후의 시점부분에서도 보다 자료를 잘 따라가고 있음을 알 수 있다. 따라서 LLGT-AH 모형이 본 자료를 더 잘 적합한다고 볼 수 있다.

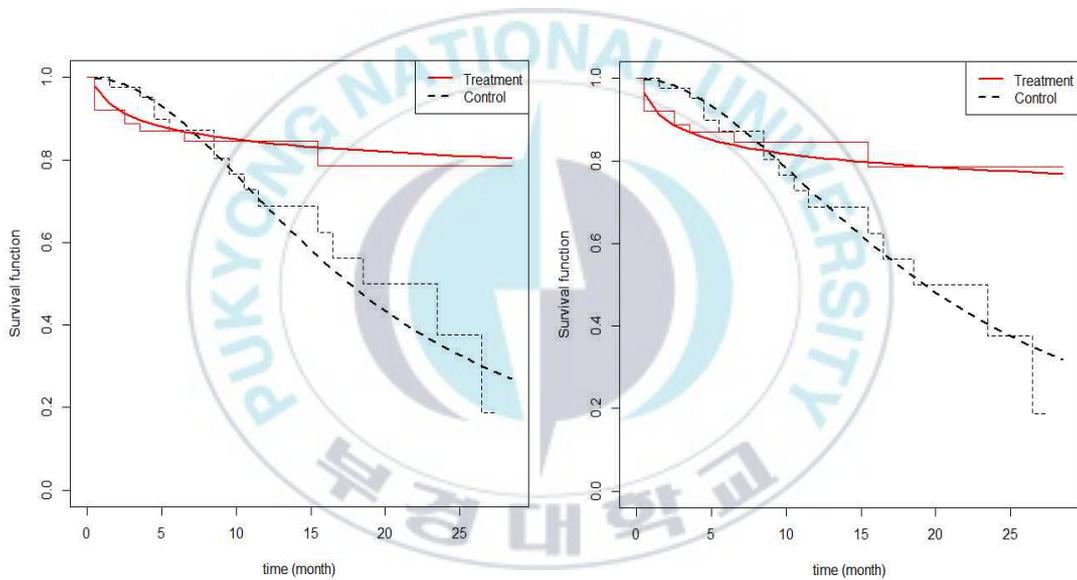


그림 3.4. 신장 감염 자료: LLG-AH (좌) 모형과 LLGT-AH (우) 모형의 적합

### 3.1.4. 모형 선택

신장 감염 자료를 대상으로 세 가지 모형 (LLG-AH, LLGT-AH, Wei-AH 모형)을 대상으로 하여 모형 선택을 시도해보았다. 모형 선택을 위해 일반적으로 사용되는 Akaike's information criterion (AIC)와 Bayesian information criterion (BIC)를 사용하였다.

- 1)  $AIC = -2l + 2k$
- 2)  $BIC = -2l + k \log(n)$

여기서  $l$ 은 로그 가능도 함수,  $k$ 는 적합된 모수의 개수,  $n$ 은 자료의 개수이다. 이 두 가지 모형 선택 기준은 적합된 모수의 개수가 증가할수록 벌점을 더 부여하는 방식이고 값이 작을수록 더욱 적절한 모형으로 판단한다. 통상적으로 두 모형 간 값이 2 정도의 차이가 날 때 모형 간 선택을 명확히 할 수 있다 (Burnham와 Anderson, 2002). 표 3.3을 보면 AIC는 LLG-AH 모형에서, BIC는 LLGT-AH 모형에서 그 값이 작음을 알 수 있다. 두 결과가 동일하게 나타나지 않는 것은 그림 3.4의 카플란마이어 생존곡선의 결과에서 보는 바와 같이 두 모형 (LLG-AH, LLGT-AH)의 적합이 시간에 따라 다소 차이가 있음을 반영한 것 같다. 하지만 두 모형의 AIC와 BIC 모두 그 값이 2보다 크지 않으므로 모형의 간결성 (parsimony) 측면에서 본 논문에서는 LLG-AH 모형을 선택한다.

표 3.3. 신장 감염 자료: 각 모형에 대한 모형 선택

Model	Log-likelihood	AIC	BIC
LLG-AH	-114.084	<b>234.167</b>	237.726
LLGT-AH	-113.422	234.843	<b>236.402</b>
Wei-AH	-136.985	279.970	283.529

### 3.1.5. 제한된 평균 생존시간

신장 감염 자료의 RMST와 두 그룹 간 RMST의 차이 값을 추정하였으며 그 추정 결과는 표 3.4와 같다. 먼저 두 그룹의 생존함수가 교차되기 전인  $\tau=4$ (개월)를 고려

하였다. 4개월 동안 지켜본 결과 처리군의 RMST에 근거한 평균 생존시간은 3.665개월이고 대조군은 3.930개월이다. 이때 유의수준 5%에서 두 그룹 간 차이에 대한 p-value는 0.036으로 두 그룹 간 유의한 차이를 보였다. 그리고 생존함수가 교차된 이후인  $\tau=6.5$ (개월)을 고려하였으며 처리군의 평균 생존시간은 5.840개월이고 대조군은 6.177개월이다. 이때 두 그룹 간 차이에 대한 p-value는 0.189로 유의수준 5%에서 유의하지 않으므로 두 그룹 간 생존함수에 유의한 차이가 없음을 나타내었다. 처리군의 평균 생존시간이 대조군의 평균 생존시간보다 더 길어지는 시점인  $\tau=12.5$ (개월)일 때이다. 즉, 12.5개월 동안 지켜보았을 때 처리군의 평균 생존시간은 10.911개월이고 대조군의 평균 생존시간은 10.905개월로 유사함을 알 수 있다. 이때 두 그룹 간 차이에 대한 p-value는 0.992로 두 그룹 간 차이가 존재하지 않으므로, RMST 측면에서 두 그룹 간 유의한 차이는 연구 초반에만 나타난다고 할 수 있다.

표 3.4. 신장 감염 자료: RMST와 그룹 간 차이 분석

$\tau$	Group	RMST	group1-group0
		Est (SE)	Est (p-value)
4	Treatment	3.665 (0.112)	-0.264 (0.036)
	Control	3.930 (0.058)	
6.5	Treatment	5.840 (0.208)	-0.337 (0.189)
	Control	6.177 (0.151)	
12.5	Treatment	10.911 (0.466)	0.007 (0.992)
	Control	10.905 (0.481)	

(Note) Est = estimate; SE = estimated standard error

### 3.2. 위암 자료

본 예증을 위해 두 번째로 사용한 자료는 위암 자료 (Gastrointestinal Tumor Study Group, 1982)이다. 이 자료는 총 90명 ( $n=90$ )의 위암환자를 대상으로 사망까지 걸린 시간을 나타내는 생존자료로 생존시간, 사망여부, 그룹변수로 이루어져있다. 시간 변수는 실험 시작으로부터 사망까지의 시간을 년 (year) 단위로 표현하였다. 사망 여부는 환자가 사망하였다면 1, 사망하지 않았다면 0으로 표시하였다. 연구 종료 시점에 사망하지 않은 환자들을 중도절단으로 하였고 90명의 환자들 중 8명이 사망하지 않았으므로 중도절단 비율은 8.9%이다. 그룹변수는 이진형 공변량 ( $x$ )으로 치료방법에 따라 두 그룹으로 나누었으며, 처리군 ( $x=1$ )은 화학치료 (chemotherapy)와 방사선치료 (radiation therapy)를 모두 실시한 그룹이고, 대조군 ( $x=0$ )은 화학치료만 실시한 그룹이다. 이 자료의 카플란마이어 생존곡선은 그림 3.5와 같다.

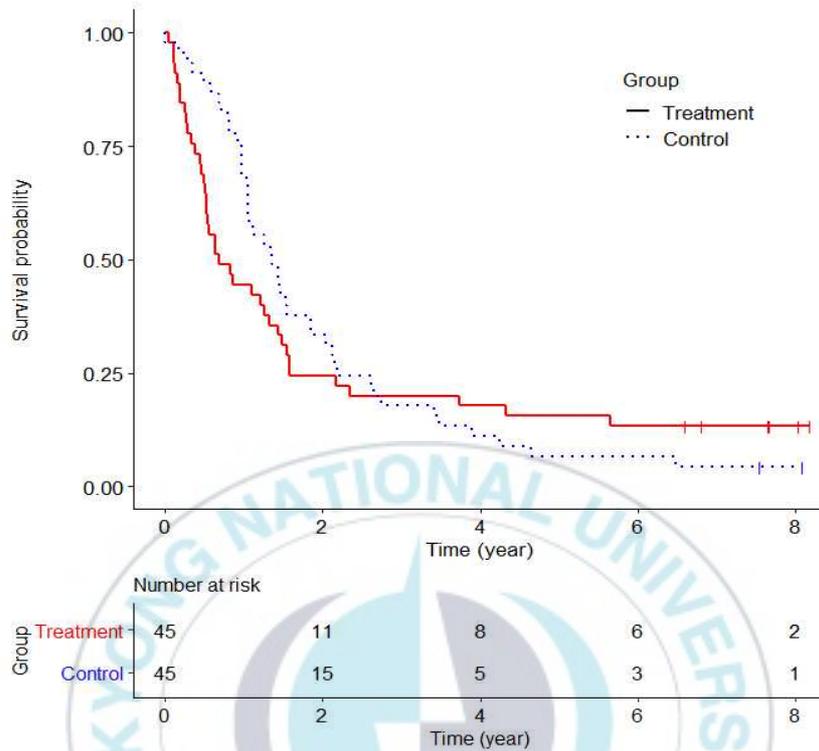


그림 3.5. 위암 자료: 카플란-마이어 생존곡선

먼저 그림 3.5의 하단의 위험 대상자 수에 대한 표를 통해 연구시작부터 2년 동안 많은 사망 발생으로 인해 위험 대상자의 수가 가장 많이 감소하고 연구종료 시점에는 위험에 처한 대상자가 3명 존재하는 것을 볼 수 있으며 이는 중도절단 수 (즉 생존자 수)를 의미한다. 카플란-마이어 생존곡선에서 두 그룹의 생존확률의 교차는 약 2.7년 정도에 발생하여 이는 전체 실험의 중간 시점 정도에 해당한다. 시험의 초기시점에는 대조군의 생존확률이 더 높지만 일정 시간이 지나면 처리군이 대조군보다 생존확률이 더 높게 나타난다. 평균 생존시간은 처리군은 2년이고 대조군은 1.98년으로 처리군이 더 긴 생존시간을 보인다. 중앙 생존시간은 처리군은 0.70년이고 대조군은 1.37년으로 처리군이 더 짧다. 위암 자료에서도 비례성 검정을 실시한 결과 자유도가

1인 카이제곱 검정통계량은 13.2이고 p-value는 0.0003으로 두 그룹의 위험률이 비례하지 않음을 알 수 있다. Cox's PH 모형에 대한 그림 3.6의 스콘필드 잔차도 (residual plot)에 의하면 3.1절의 신장 감염 자료보다 더 기울어져있는 형태를 통해 위암 자료의 두 그룹 간 위험률이 비례하지 않음을 알 수 있다.

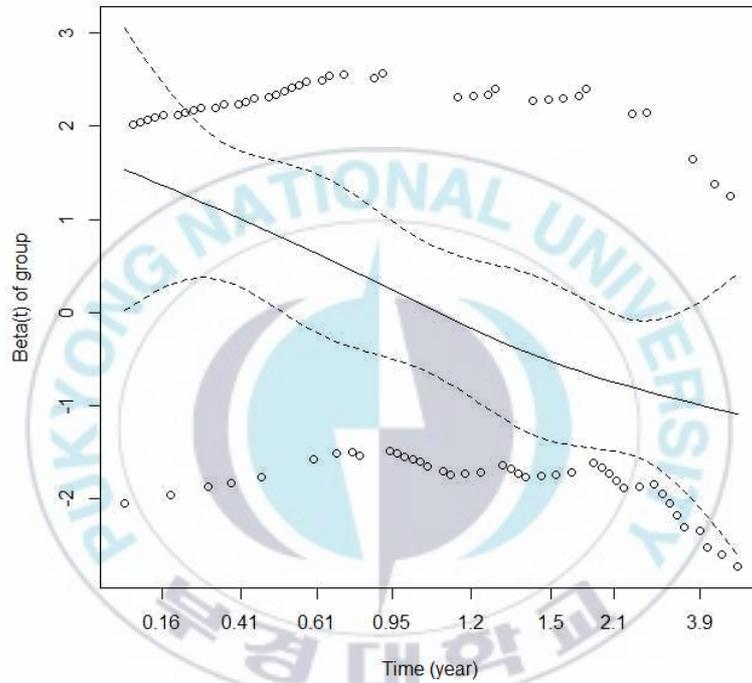


그림 3.6. 위암 자료: 비례위험 모형에 대한 잔차도

### 3.2.1. 모형 적합

위암 자료에서도 R 프로그램에서 `repeat()`문과 Python 프로그램에서 `while`과 `if`문으로 뉴턴-랩슨 방법을 이용하여 고려된 다섯 가지 모형의 모수를 추정하였으며 그 결과는 표 3.5와 같다.

표 3.5. 위암 자료: 세 모형 (AH, PH, AFT 모형)을 이용한 분석

Model	Est (SE)				
	$\beta$ : Treatment ( $x = 1$ )	$\theta_0$	$\phi$	$\alpha_0$	$\alpha_1$
LLG-AH	1.092 (0.306)	0.6978 (0.165)	1.861 (0.232)	—	—
LLGT-AH	1.554 (0.340)	0.603 (0.154)	—	1.745 (0.229)	2.652 (0.597)
Wei-AH	0.209 (1.528)	0.559 (0.093)	0.854 (0.074)	—	—
Cox's PH	0.106 (0.223)	—	—	—	—
AFT	-0.378 (0.326)	—	—	—	—

(Note) Est = estimate; SE = estimated standard error;  $\beta$ : regression parameter;  $\theta_0$ : scale parameter;  $\phi$ : shape parameter;  $\alpha_0$ : time effect of control group;  $\alpha_1$ : time effect of treatment group

신장 감염 자료와 동일하게 위암 자료에서도 Wei-AH 모형, Cox's PH 모형 그리고 AFT 모형에서 그룹 간 효과인  $\beta$ 가 유의수준 5%에서 유의하지 않은 결과를 보인다. 그리고 LLG-AH 모형과 LLGT-AH 모형 모두  $\beta$ 가 유의함을 알 수 있으며  $\hat{\beta} > 0$ 이므로 처리군이 위험을 가속화한다고 할 수 있다.

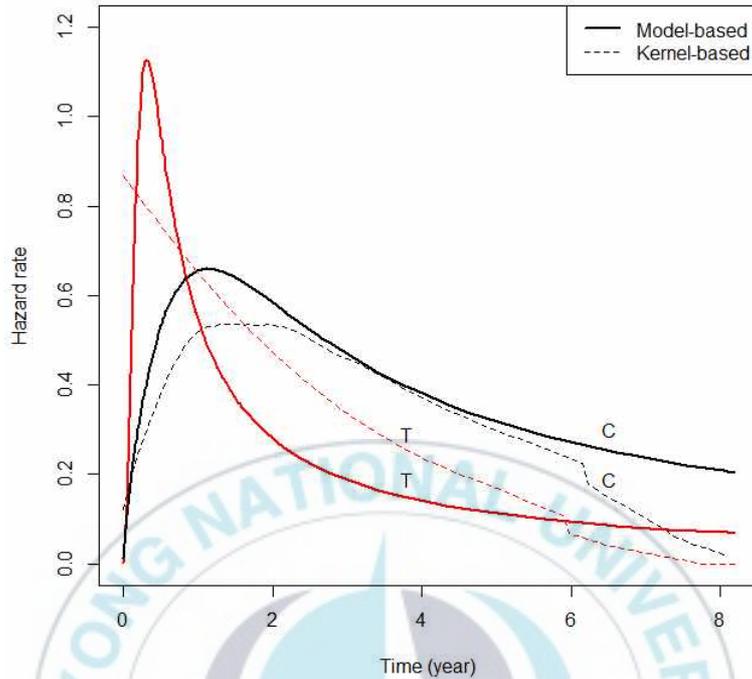


그림 3.7. 위암 자료: 평활 및 AH 모형에 근거한 위험률 곡선  
(Note) T: 처리군; C: 대조군

그림 3.7에서도 제안하는 모형에 근거한 위험률 곡선과 평활 위험률 곡선이 매우 유사한 형태를 보인다. 두 그룹 모두 시간이 지나면서 감소하는 형태를 보이지만 처리군의 위험률이 더 낮게 감소한다. 따라서 처리군의 위험은 시간이 흐르면 빠른 속도로 감소함을 의미한다. 대조군도 위험률이 낮아지기는 하지만 처리군 보다는 위험률이 높은 것으로 보인다. 표 3.5의 LLG-AH 모형에서  $\hat{\phi} = 1.861$ 으로 1보다 큰 값으로 봉우리가 하나인 형태를 의미한다. 이 결과와 동일하게 그림 3.7에서 봉우리가 하나인 위험함수 형태를 볼 수 있다.

### 3.2.2. 유의성 검토

이 자료 역시 두 생존곡선이 비례하지 않고 교차된 형태이므로 Wei-AH 모형, Log-rank 검정과 Cox's PH 모형에서 그룹 간 차이가 가장 유의하지 않음을 보인다. 그리고 신장 감염 자료에 대한 유의성 검토의 결과와 동일하게 그룹 간 효과인  $\beta$ 는 LLG-AH 모형의 검정과 LLGT-AH 모형의 검정에서 유의수준 5%에서 유의하였다. 추가적으로 Gehan 검정에서도 유의하였지만 p-value에서 LLG-AH 모형과 LLGT-AH 모형의 값에 비해 큰 값을 나타낸다. 즉, LLG-AH 모형과 LLGT-AH 모형에서 더욱 유의함을 의미한다.

표 3.6. 위암 자료: 유의성 검토를 위한 통계량과 p-값

Method	Statistics (p-value)
Log-rank	0.225 (0.635)
Gehan	3.964 ( <b>0.046</b> )
Tarone-Ware	1.903 (0.168)
LLG-AH	3.560 ( <b>&lt;0.001</b> )
LLGT-AH	4.570 ( <b>&lt;0.001</b> )
Wei-AH	0.136 (0.891)
Cox's PH	0.474 (0.635)
AFT	-1.159 (0.247)

### 3.2.3. 모형 검토

위암 자료를 이용한 카플란-마이어에 LLG-AH 모형과 LLGT-AH 모형을 적합한 그래프는 그림 3.8과 같다. LLG-AH 모형보다 LLGT-AH 모형이 자료를 보다 잘 설명하고 있음을 명확하게 알 수 있다. 특히 교차 이전의 시점에서 자료를 확실하게 반영하

였고 LLG-AH 모형보다 교차된 지점에 더 가까워졌다. 따라서 LLGT-AH 모형이 위암 자료를 더 잘 적합함을 알 수 있다.

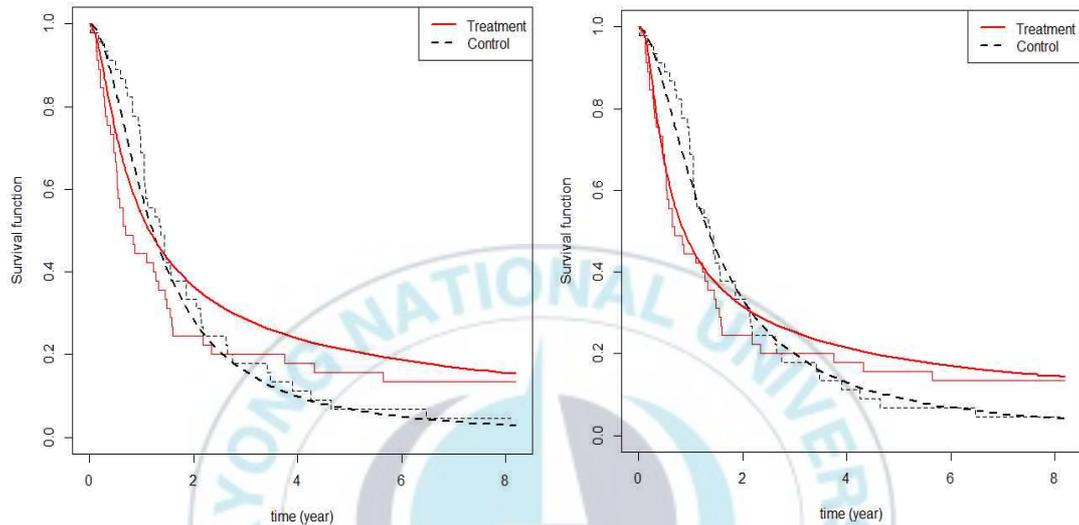


그림 3.8 위암 자료: LLG-AH (좌) 모형과 LLGT-AH (우) 모형의 적합

### 3.2.4. 모형 선택

위암 자료에서도 모형 선택을 실시하였다. AIC와 BIC 모두 LLGT-AH 모형의 값이 가장 작고 그 다음으로 LLG-AH 모형, Wei-AH 모형 순이다. 특히, LLGT-AH 모형의 AIC와 BIC의 값들 모두 LLG-AH 모형보다 약 2.6정도가 작다. 따라서 LLGT-AH 모형이 더 적절한 모형임을 보여준다.

표 3.7. 위암 자료: 각 모형에 대한 모형 선택

Model	Log-likelihood	AIC	BIC
LLG-AH	-131.832	269.664	272.664
LLGT-AH	-130.547	<b>267.095</b>	<b>270.094</b>
Wei-AH	-142.182	290.365	293.364

### 3.2.5. 제한된 평균 생존시간

위암 자료를 이용하여 제한된 평균 생존시간 (RMST)에 대한 추정과 이에 근거한 두 그룹 간 차이의 유의성을 검정을 실시하였으며 표 3.8과 같다. 먼저 그림 3.5로부터 두 그룹의 생존함수가 교차되기 전의 시점인  $\tau=2$ (년)일 때를 고려하였다. 처리군의 평균 생존시간은 0.998년이고 대조군은 1.341년으로 대조군의 평균 생존시간이 더 길다. 두 그룹 간의 차이에 대한 p-value는 0.013으로 유의수준 5%에서 매우 유의한 차이를 보인다. 반면 두 그룹의 생존함수가 교차되는 시점인  $\tau=2.7$ (년)일 때는 두 그룹에 대한 p-value가 0.047로 유의하였지만 그 차이가 많이 감소하였음을 볼 수 있다. 처리군의 평균 생존시간이 대조군의 평균 생존시간보다 훨씬 더 길어지는 시점은  $\tau=8$ (년)일 때로 처리군의 생존시간은 1.981년이고 대조군의 생존시간은 1.978년이다. 이때 두 그룹 간 차이에 대한 p-value는 0.995로 두 군간 차이는 유의하지 않게 된다. 따라서 위암 자료는 연구의 중반정도 시점까지 두 그룹 간의 유의한 차이를 보인다.

표 3.8. 위암 자료: RMST와 그룹 간 차이 분석

$\tau$	Group	Est (SE)	Est (p-value)
		RMST	group1-group0
2	Treatment	0.998 (0.105)	-0.343 (0.013)
	Control	1.341 (0.089)	
2.7	Treatment	1.149 (0.140)	-0.371 (0.047)
	Control	1.520 (0.123)	
8	Treatment	1.981 (0.388)	0.003 (0.995)
	Control	1.978 (0.271)	

(Note) Est = estimate; SE = estimated standard error



## 제 4장 AH 모형의 확장

### 4.1. 혼합치료 AH 모형

추적기간 (follow-up period)이 길 때 생존시간  $T$ 에 대한 생존함수  $S(t)$ 는 0으로 수렴하는 것이 일반적이다 (즉,  $\lim_{t \rightarrow \infty} S(t) = 0$ ). 하지만 암 연구와 같은 임상연구에서는 어떤 개인들은 관심사건 (예: 수술 후 재발)을 경험하지 않아서 생존함수가 0으로 수렴하지 않고 특정 상수로 수렴하는 경우도 있다. 이때 관심사건을 경험하지 않은 개인들을 완치된 (cured) 것으로 판단하며, 이때 수렴되는  $S(t)$ 의 0이 아닌 특정 상수를 치료율 (cure rate)이라 부른다 (Amico와 Keilegom, 2018). 이러한 생존함수가 특정 상수로 수렴되는 경우를 다루기 위한 모형으로 혼합치료 모형 (mixture cure model)이 있다.  $U$ 를 관측되지 않는 치료상태 (unobserved 또는 latent cure status)를 나타내는 확률 변수라 하자. 만약 한 개인이 관심사건에 대해 치료되었다면 (cured 또는 non-susceptible)  $U=0$ 이고 치료되지 않았다면 (uncured 또는 susceptible)  $U=1$ 이라 하자. 그리고  $S_u(t) = P(T > t | U=1)$ 는 치료되지 않은 개인들의 생존함수로  $\lim_{t \rightarrow \infty} S_u(t) = 0$ 을 만족하는 통상적인 생존함수 (proper survival function)라 하자.  $\pi = P(U=1)$ 는 치료되지 않는 확률이라 하자. 혼합치료 모형은 치료된 그룹과 치료되지 않은 그룹의 결합 형태로 (4.1)과 같이 표현된다.

$$\begin{aligned} S(t) &= P(U=0)P(T > t | U=0) + P(U=1)P(T > t | U=1) \\ &= (1 - \pi) + \pi S_u(t) \end{aligned} \tag{4.1}$$

여기서  $P(T > t | U = 0) = 1$ 은 치료된 개체들의 생존함수로서 퇴화 생존함수 (degenerate survival function)라 부른다. 특히  $\lim_{t \rightarrow \infty} S_u(t) = 0$ 이므로  $\lim_{t \rightarrow \infty} S(t) = 1 - \pi$ 이 되고 이때  $1 - \pi$ 를 치료율 (cure rate 또는 cured probability)이라 부른다. 혼합치료 모형에서  $\pi$ 에 영향을 주는 공변량 (covariate) 벡터  $z = (z_1, \dots, z_q)^T$ 와  $S_u(t)$ 에 영향을 주는 공변량 벡터  $x = (x_1, \dots, x_p)^T$ 를 도입하면 (4.1)은 (4.2)와 같이 표현된다.

$$S(t|x, z) = [1 - \pi(z)] + \pi(z)S_u(t|x) \quad (4.2)$$

여기서  $\pi(z) = P(U=1|z)$ 는 이진형으로 모형화되며 로지스틱 (logistic), 프로빗 (probit) 또는 보 로그-로그 (complementary log-log) 등의 연결함수 (link function)가 사용될 수 있다. 본 연구에서는  $\pi(z)$ 에 대해서는 로지스틱 함수를 사용하여 (4.3)과 같이 표현된다.

$$\pi(z) = \frac{e^{z^T \gamma}}{1 + e^{z^T \gamma}} \quad (4.3)$$

여기서  $\gamma$ 는 절편을 포함한 로지스틱 회귀 계수들의 벡터이다. 특히  $S_u(t|x)$ 는 비례위험 (PH) 모형, 가속화 실패시간 (AFT) 모형 또는 가속화 위험 (AH) 모형 등이 사용될 수 있다. 본 연구에서는  $S_u(t|x)$ 에 대해서는 AH 모형을 사용한다. 따라서  $S_u(t|x)$ 와 그에 대응하는  $\Lambda_u(t|x)$ 는 (4.4)와 같이 표현된다.

$$\begin{aligned} S_u(t|x) &= S_{u0}(te^{x^T \beta})^{\exp(-x^T \beta)} \\ \Lambda_u(t|x) &= \Lambda_{u0}(te^{x^T \beta})e^{-x^T \beta} \end{aligned} \quad (4.4)$$

여기서  $\beta$ 는 절편이 없는 비례위험 회귀계수들의 벡터이며  $S_{0u}(\cdot)$ 와  $A_{0u}(\cdot)$ 는 각각 기저 생존함수와 기저 누적 위험함수이다. 본 연구에서는 (4.3)과 (4.4)를 (4.2)에 대입한 모형을 로지스틱-AH 혼합치료 모형이라 부른다. 이 로지스틱-AH 혼합치료 모형은 AH 모형의 특성에 따라 치료되지 않은 환자들의 시간에 대해 즉각적인 치료효과를 허용한다는 장점이 있다 (Zhang과 Peng, 2009). 그리고  $S_{0u}(\cdot)$ 와  $A_{0u}(\cdot)$ 에 대해 다양한 모수적 분포의 적용이 가능하다. 본 연구에서는 앞서 많은 장점을 언급해 온 로그-로지스틱 분포를 고려하였다. 로그-로지스틱 분포를 고려한 최종 혼합치료 모형을 “Logistic-LLGAH” 모형이라 부르며 이에 대한 가능도 추정법은 4.2절에서 다룬다.

## 4.2. 혼합치료 AH 모형의 가능도 추정법

본 절에서는 (4.4)식의  $A_{0u}(\cdot)$ 에 로그-로지스틱 함수를 적용한 Logistic-LLGAH 모형의 최대 가능도 추정법을 제시한다. 이를 위한 로그-가능도 함수는 (4.5)와 같다.

$$\begin{aligned}
 l &= \sum_{i=1}^n [\delta_i \log f(y_i | x_i, z_i) + (1 - \delta_i) \log S(y_i | x_i, z_i)] \\
 &= \sum_{i=1}^n \delta_i \log [\pi(z_i) f_u(y_i | x_i)] + \sum_{i=1}^n (1 - \delta_i) \log [1 - \pi(z_i) + \pi(z_i) S_u(y_i | x_i)] \\
 &= \sum_{i=1}^n \delta_i \log [\pi(z_i)] - \sum_{i=1}^n \delta_i A_{u0}(y_i e^{x_i^T \beta}) e^{-x_i^T \beta} + \sum_{i=1}^n \delta_i \log [\lambda_{u0}(y_i e^{x_i^T \beta})] \\
 &\quad + \sum_{i=1}^n (1 - \delta_i) \log [1 - \pi(z_i) + \pi(z_i) S_{u0}(y_i e^{x_i^T \beta}) \exp(-x_i^T \beta)] \tag{4.5}
 \end{aligned}$$

여기서  $S_{0u}(te^{x^T \beta})$ 와  $A_{0u}(te^{x^T \beta})$ 는 (4.6)으로 표현된다.

$$S_{0u}(te^{xT\beta}) = \frac{1}{1 + \theta_0(te^{xT\beta})^\phi}$$

$$A_{0u}(te^{xT\beta}) = \log[1 + \theta_0(te^{xT\beta})^\phi] \quad (4.6)$$

고려된 모형 모수의 ML 추정량을 얻기 위하여 R의 optim() 함수를 이용하였다.

### 4.3. SEER 자료

#### 4.3.1 자료 설명

예증을 위하여 R의 “RecurRisk” 패키지에서 제공하는 “data.individual” 데이터를 이용하였다. 이 자료는 다양한 암 관련 자료를 제공하는 SEER\*Stat software 홈페이지 (<https://seer.cancer.gov/seerstat/>)에서 제공받았다 (SEER: Surveillance, Epidemiology, and End Results). 본 자료는 5개의 공변량에 대해 401,410개의 대량의 관측값을 가지고 있다. 관심사건은 암으로 인한 사망이며 연구종료까지 생존으로 인한 중도절단 비율은 86.96%이다.

표 4.1. SEER 자료에 대한 변수설명

변수	코딩 및 설명
Time	Follow-up 기간 (year)
Status	암에 의한 사망여부 (1: 사망, 0: 생존)
Stage	암의 전이 정도 (0: Localized, 1: Regional, 2: Distant)
Yeargroup	연도에 따른 그룹화 (0:1985-89, 1: 1990-94, 2: 1995-99, 3: 2000-13)
Agegroup	나이에 따른 그룹화 (0: 15-59세, 1: 60-69세, 2: 70-84세)

표 4.1의 공변량 중 하나의 변수에 대한 치료율 모형을 적합시키기 위해 암의 Stage 변수만 고려하여 분석하였으며 대응하는 카플란-마이어 생존곡선은 그림 4.1과 같다.

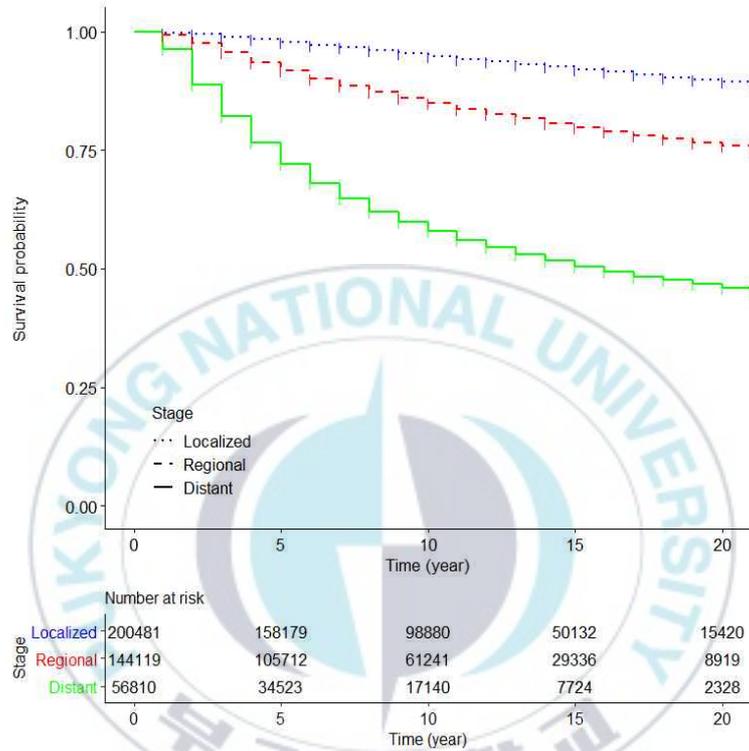


그림 4.1. SEER 자료: 카플란-마이어 생존곡선

그림 4.1에 의하면 충분한 시간이 흐른 뒤에도 세 그룹에 대한 생존함수 모두 0으로 수렴하지 않고 특정 상수로 수렴하므로 치료율 모형의 형태를 보인다.

### 4.3.2 혼합치료 AH 모형: 모수 추정 및 모형의 적합

SEER 자료의 적합을 위해 다음과 같은 네 가지 혼합치료 모형을 고려한다.

M1: Logistic-LLGAH

M2: Logistic-LLGPH

M3: Logistic-WeiPH

M4: Logistic-CoxPH

M1의 최대 가능도 기반 모수추정을 위해 R의 `optim()` 함수를 이용하였으며 추정된 모형적합 결과는 표 4.2와 같다.

표 4.2. SEER 자료: M1의 모수추정

Model	Parameters	Est	(SE)
Logistic	$\gamma_0$ : Intercept	-1.835	(0.015)
	$\gamma_1$ : Regional group	1.091	(0.015)
	$\gamma_2$ : Distant group	3.789	(0.063)
Survival	$\beta_1$ : Regional group	0.715	(0.017)
	$\beta_2$ : Distant group	1.441	(0.015)
	$\theta_0$ : Scale	0.005	(0.0001)
	$\phi$ : Shape	2.140	(0.009)

(Note) Est = estimate; SE = estimated standard error

표 4.2의 로지스틱 모형에서 모든 추정값들이 유의수준 5%에서 유의하게 나타난다. 추정된  $\gamma_1$ 와  $\gamma_2$ 의 값이 모두 양수이고  $\gamma_1$ 보다  $\gamma_2$ 의 값이 더 크므로 암이 전이된 정도가 넓을수록 사망의 위험률이 더 증가함을 나타낸다. 또,  $\beta_1$ 보다  $\beta_2$ 의 값이 더 크므로 전이가 넓을수록 생존율에 또한 많은 영향을 미침을 알 수 있다. 추정된 모수들을 이

용하여 SEER 자료에 M1을 적합한 그래프는 그림 4.2의 (a)와 같다. M1의 타당성을 보이기 위하여 M2, M3 그리고 M4의 적합을 통해 비교하였다. M2는 자료를 완전히 따라가지 못하고 M3는 일부분이 따라가지 못한다. M4는 대체로 잘 따라가는 편이지만 곡선의 꼬리부분에서 M1만큼 잘 따라가는 것처럼 보이지는 않는다. 따라서 네 가지의 모형들 중 M1이 세 그룹의 카플란-마이어 생존함수를 가장 잘 따라감으로써 모형의 타당성을 보인다.



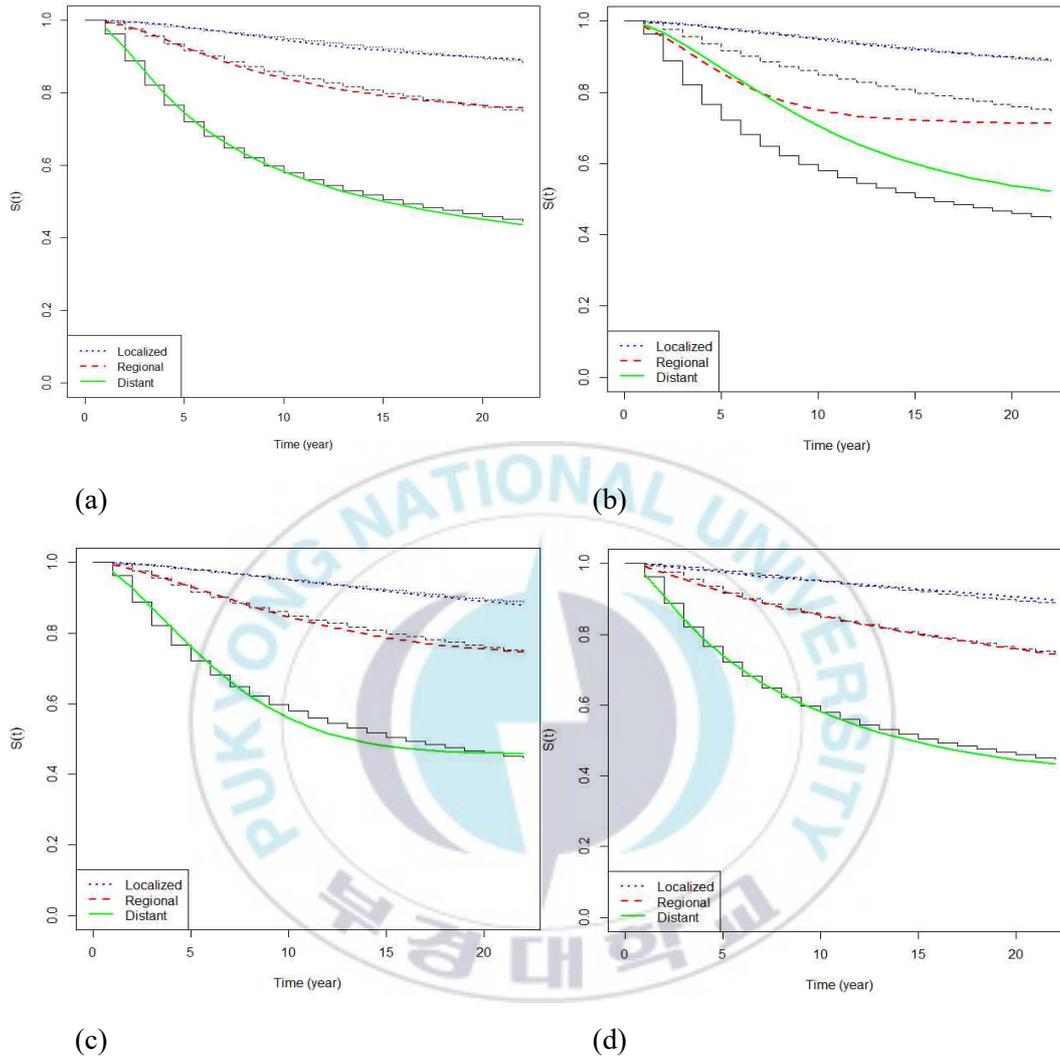


그림 4.2. SEER 자료의 모형적합: (a) M1; (b) M2; (c) M3; (d) M4

## 제 5장 결론

본 연구에서는 시간에 따라 두 그룹의 생존함수 또는 위험함수가 교차하는 생존자료를 분석하기 위해 기저 위험함수로 로그-로지스틱 분포를 사용하는 가속화 위험 모형인 LLG-AH 모형과 시간효과를 허용하는 LLGT-AH 모형을 제안하였다. 예증을 위해 임상시험 자료인 신장 감염 자료와 위암 자료를 분석하였다. 모형의 타당성을 보이기 위하여 모형적합, 유의성 검토, 모형선택 결과들을 제시하였다. 분석결과 LLG-AH 모형보다 LLGT-AH 모형이 교차되는 부분을 보다 명확하게 인지하므로 시간효과가 두 그룹 간의 차이를 잘 반영함을 확인하였다. 특히, 생존곡선이 교차하는 경우 AH 모형의 기저 위험함수로 와이블 분포를 사용하는 것은 부적절함을 확인하였다.

따라서 생존함수 또는 위험함수가 교차되는 경우 교차지점을 확실하게 인지해주는 LLGT-AH 모형을 사용하는 것을 제안한다. 나아가 로그-로지스틱 분포에 기초한 AH 모형을 치료율 모형으로 확장하여 SEER 자료에 적용한 후 적절한 분석 결과를 제시하였다. 제안된 모형 접근법의 타당성을 보다 명확히 제시하기 위해 모의실험 및 변수선택 연구 (Chen 등, 2018)가 추후 과제가 될 것이다.

첨가적으로, 중도절단된 생존데이터 학습 (learning) 및 예측오차 (prediction error)에 기반하여 제안된 AH 모형의 예측 성능을 평가하는 방법론 개발도 흥미 있는 추후 연구과제가 될 것으로 사료된다.

## 참고문헌

- [1] Akhtar, M. T. and Khan, A. A. (2014). Log-logistic distribution as a reliability model: A Bayesian analysis, *American Journal of Mathematics and Statistics*, **4**, 162-170.
- [2] Calsavara, V. F., Milani, E. A., Bertolli, E. and Tomazella, V. (2019). Long-term frailty modeling using a non-proportional hazards model: Application with a melanoma dataset, *Statistical methods in medical research*, 0962280219883905.
- [3] Chen, Y., Zhang, J. and Xu, Y. (2018). Adaptive lasso for accelerated hazards models, *Journal of Statistical Computation and Simulation*, **88**, 2948-2960.
- [4] Chen, Y. Q. and Wang, M. C. (2000a). Analysis of accelerated hazards models, *Journal of the American Statistical Association*, **95**, 608-618.
- [5] Chen, Y. Q. and Wang, M. C. (2000b). Estimating a treatment effect with the accelerated hazards models, *Controlled clinical trials*, **21**, 369-380.
- [6] Chiou, S., Kang, S. and Yan, J. (2014). Fitting Accelerated Failure Time Model in Routine Survival Analysis with R Package Aftgee, *Journal of Statistical Software*, **61**, 1-23.
- [7] Clark, D. E. and El-Taha, M. (2015). Some useful properties of log-logistic random variables for health care simulations, *International Journal of Statistics in Medical Research*, **4**, 79-86.
- [8] Co, C. A. T. (2010). Investigating the use of the accelerated hazards model for survival analysis, Master thesis, Simon Fraser University.
- [9] Burnham, K. P. and Anderson, D. R. (2002). A practical information-theoretic

approach. Model selection and multimodel inference, 2nd ed. *Springer*, New York, 2.

- [10] Kartsonaki, C. (2016). Survival analysis, *Diagnostic Histopathology*, **22**, 263-270.
- [11] Klein, J. P. and Moeschberger, M. L. (2006). Survival analysis: techniques for censored and truncated data, *Springer Science & Business Media*.
- [12] Lee, M. J. (2020). A study on the restricted mean survival time. *The Korean Data and Information Science Society*, **31**, 341-352.
- [13] Lin, X. and Wang, H. (2004). A new testing approach for comparing the overall homogeneity of survival curves, *Biometrical Journal: Journal of Mathematical Methods in Biosciences*, **46**, 489-496.
- [14] MacKenzie, G. and Ha, I. D. (2007). Modelling survival data with crossing hazards, proceedings of *IWSM*.
- [15] Tian, L., Zhao, L. and Wei, L. J. (2014). Predicting the restricted mean event time with the subject's baseline covariates in survival analysis, *Biostatistics*, **15**, 222-233.
- [16] Yang, D. G., Ha, I. D. and Cho, G. H. (2018). Survival analysis for contract maintenance period using life insurance data, *Korean Journal of Applied Statistics*, **31**, 771-783.
- [17] Zhang, J. and Peng, Y. (2009). Crossing hazard functions in common survival models, *Statistics & probability letters*, **79**, 2124-2130.
- [18] Zhang, J. and Peng, Y. (2009). Accelerated hazards mixture cure model. *Lifetime data analysis*, **15**, 455.

## 부록A

### LLG-AH 모형의 가능도 계산 절차

LLG-AH 모형에서 최대 가능도 추정량  $\hat{\theta} = (\hat{\beta}, \hat{\theta}_0, \hat{\phi})$ 의 분산-공분산을 구하기 위한  $\theta$ 의 음의 헤시안 행렬에 해당하는 식들은 다음과 같다.

$$\begin{aligned} \frac{\partial^2 l}{\partial \beta \partial \beta^T} &= \sum_i \frac{\delta_i \theta_0 t_i^\phi e^{x_i^T \beta \phi} x_i^2 \phi^2}{(1 + \theta_0 t_i^\phi e^{x_i^T \beta \phi})^2} - \sum_i \frac{\theta_0 t_i^\phi e^{x_i^T \beta \phi} x_i^2 \phi (\phi - \theta_0 t_i^\phi e^{x_i^T \beta \phi} - 1)}{(1 + \theta_0 t_i^\phi e^{x_i^T \beta \phi})^2} \\ &\quad + \sum_i \frac{\theta_0 t_i^\phi e^{x_i^T \beta (\phi-1)} x_i^2 \phi}{1 + \theta_0 t_i^\phi e^{x_i^T \beta \phi}} - \sum_i x_i^2 \log(1 + \theta_0 t_i^\phi e^{x_i^T \beta \phi}) e^{-x_i^T \beta \phi} \\ \frac{\partial^2 l}{\partial \theta_0^2} &= - \sum_i \frac{\delta_i}{\phi^2} + \sum_i (\delta_i + e^{-x_i^T \beta}) \frac{(t_i^\phi e^{x_i^T \beta \phi})^2}{(1 + \theta_0 t_i^\phi e^{x_i^T \beta \phi})^2} \\ \frac{\partial^2 l}{\partial \phi^2} &= - \sum_i \frac{\delta_i}{\phi^2} - \sum_i \delta_i \theta_0 t_i^\phi e^{x_i^T \beta \phi} (\log t_i + x_i^T \beta)^2 \left[ \frac{1}{1 + \theta_0 t_i^\phi e^{x_i^T \beta \phi}} - \frac{\theta_0 t_i^\phi e^{x_i^T \beta \phi}}{(1 + \theta_0 t_i^\phi e^{x_i^T \beta \phi})^2} \right] \\ &\quad - \sum_i \theta_0 t_i^\phi e^{x_i^T \beta (\phi-1)} (\log t_i + x_i^T \beta)^2 \left[ \frac{1}{1 + \theta_0 t_i^\phi e^{x_i^T \beta \phi}} - \frac{\theta_0 t_i^\phi e^{x_i^T \beta \phi}}{(1 + \theta_0 t_i^\phi e^{x_i^T \beta \phi})^2} \right] \\ \frac{\partial^2 l}{\partial \beta \partial \theta_0} &= - \sum_i \frac{\delta_i t_i^\phi e^{x_i^T \beta \phi} x_i \phi}{(1 + \theta_0 t_i^\phi e^{x_i^T \beta \phi})^2} - \sum_i t_i^\phi e^{x_i^T \beta (\phi-1)} x_i \left[ \frac{\phi - \theta_0 t_i^\phi e^{x_i^T \beta \phi} - 1}{(1 + \theta_0 t_i^\phi e^{x_i^T \beta \phi})^2} \right] \\ \frac{\partial^2 l}{\partial \beta \partial \phi} &= \sum_i \delta_i x_i - \sum_i \delta_i \theta_0 x_i t_i^\phi e^{x_i^T \beta \phi} \left[ \frac{1 + \phi \log t_i + \phi x_i^T \beta}{1 + \theta_0 t_i^\phi e^{x_i^T \beta \phi}} - \frac{\theta_0 \phi t_i^\phi e^{x_i^T \beta \phi} (\log t_i + x_i^T \beta)}{(1 + \theta_0 t_i^\phi e^{x_i^T \beta \phi})^2} \right] \end{aligned}$$

$$\begin{aligned}
& - \sum_i \theta_0 x_i t_i^\phi e^{x_i^T \beta (\phi-1)} \left[ \frac{1 + \phi \log t_i + \phi x_i^T \beta}{1 + \theta_0 t_i^\phi e^{x_i^T \beta \phi}} - \frac{\theta_0 \phi t_i^\phi e^{x_i^T \beta \phi} (\log t_i + x_i^T \beta)}{(1 + \theta_0 t_i^\phi e^{x_i^T \beta \phi})^2} \right] \\
& + \sum_i \theta_0 x_i t_i^\phi e^{x_i^T \beta (\phi-1)} \frac{\log t_i + x_i^T \beta}{1 + \theta_0 t_i^\phi e^{x_i^T \beta \phi}} \\
\frac{\partial^2 l}{\partial \theta_0 \partial \phi} &= - \sum_i \frac{\delta t_i^\phi e^{x_i^T \beta \phi} (\log t_i + x_i^T \beta)}{(1 + \theta_0 t_i^\phi e^{x_i^T \beta \phi})^2} - \sum_i \frac{t_i^\phi e^{x_i^T \beta (\phi-1)} (\log t_i + x_i^T \beta)}{(1 + \theta_0 t_i^\phi e^{x_i^T \beta \phi})^2}
\end{aligned}$$



## 부록B

### LLGT-AH 모형의 가능도 계산 절차

LLGT-AH 모형에서 최대 가능도 추정량  $\hat{\theta} = (\hat{\beta}, \hat{\theta}_0, \hat{\alpha}_0, \hat{\alpha}_1)$ 의 분산-공분산을 구하기 위한  $\theta$ 의 음의 헤시안 행렬에 해당하는 식들은 다음과 같다.

$$\begin{aligned}
 \frac{\partial^2 l}{\partial \beta \partial \beta^T} &= \sum_i \frac{\delta_i \theta_0 t_i^{[\alpha_0(1-x_i) + \alpha_1 x_i]} e^{x_i^T \beta [\alpha_0(1-x_i) + \alpha_1 x_i]} x_i^2 [\alpha_0(1-x_i) + \alpha_1 x_i]^2}{(1 + \theta_0 t_i^{[\alpha_0(1-x_i) + \alpha_1 x_i]} e^{x_i^T \beta [\alpha_0(1-x_i) + \alpha_1 x_i]})^2} \\
 &- \sum_i \frac{\theta_0 t_i^{[\alpha_0(1-x_i) + \alpha_1 x_i]} e^{x_i^T \beta [\alpha_0(1-x_i) + \alpha_1 x_i]} x_i^2 [\alpha_0(1-x_i) + \alpha_1 x_i] [\alpha_0(1-x_i) + \alpha_1 x_i - \theta_0 t_i^{[\alpha_0(1-x_i) + \alpha_1 x_i]} e^{x_i^T \beta [\alpha_0(1-x_i) + \alpha_1 x_i]} - 1]}{(1 + \theta_0 t_i^{[\alpha_0(1-x_i) + \alpha_1 x_i]} e^{x_i^T \beta [\alpha_0(1-x_i) + \alpha_1 x_i]})^2} \\
 &+ \sum_i \frac{\theta_0 t_i^{[\alpha_0(1-x_i) + \alpha_1 x_i]} e^{x_i^T \beta [\alpha_0(1-x_i) + \alpha_1 x_i - 1]} x_i^2 [\alpha_0(1-x_i) + \alpha_1 x_i]}{1 + \theta_0 t_i^{[\alpha_0(1-x_i) + \alpha_1 x_i]} e^{x_i^T \beta [\alpha_0(1-x_i) + \alpha_1 x_i]}} - \sum_i x_i^2 \log(1 + \theta_0 t_i^\phi e^{x_i^T \beta \phi}) e^{-x_i^T \beta \phi} \\
 &- \sum_i x_i^2 \log(1 + \theta_0 t_i^{[\alpha_0(1-x_i) + \alpha_1 x_i]} e^{x_i^T \beta [\alpha_0(1-x_i) + \alpha_1 x_i]}) e^{-x_i^T \beta [\alpha_0(1-x_i) + \alpha_1 x_i]} \\
 \frac{\partial^2 l}{\partial \theta_0^2} &= - \sum_i \frac{\delta_i}{\phi^2} + \sum_i (\delta_i + e^{-x_i^T \beta}) \frac{(t_i^{[\alpha_0(1-x_i) + \alpha_1 x_i]} e^{x_i^T \beta [\alpha_0(1-x_i) + \alpha_1 x_i]})^2}{(1 + \theta_0 t_i^{[\alpha_0(1-x_i) + \alpha_1 x_i]} e^{x_i^T \beta [\alpha_0(1-x_i) + \alpha_1 x_i]})^2} \\
 \frac{\partial^2 l}{\partial \alpha_0^2} &= - \sum_i \frac{\delta_i (1-x_i)^2}{[\alpha_0(1-x_i) + \alpha_1 x_i]^2} \\
 &- \sum_i (\delta_i + e^{-x_i^T \beta}) (1-x_i)^2 (\log t_i + x_i^T \beta)^2 \frac{\theta_0 t_i^{[\alpha_0(1-x_i) + \alpha_1 x_i]} e^{x_i^T \beta [\alpha_0(1-x_i) + \alpha_1 x_i]}}{(1 + \theta_0 t_i^{[\alpha_0(1-x_i) + \alpha_1 x_i]} e^{x_i^T \beta [\alpha_0(1-x_i) + \alpha_1 x_i]})^2}
 \end{aligned}$$

$$\begin{aligned}
\frac{\partial^2 l}{\partial \alpha_1^2} &= - \sum_i \frac{\delta_i x_i^2}{[\alpha_0(1-x_i) + \alpha_1 x_i]^2} \\
&\quad - \sum_i (\delta_i + e^{-x_i^T \beta}) x_i^2 (\log t_i + x_i^T \beta)^2 \frac{\theta_0 t_i^{[\alpha_0(1-x_i) + \alpha_1 x_i]} e^{x_i^T \beta [\alpha_0(1-x_i) + \alpha_1 x_i]}}{(1 + \theta_0 t_i^{[\alpha_0(1-x_i) + \alpha_1 x_i]} e^{x_i^T \beta [\alpha_0(1-x_i) + \alpha_1 x_i]})^2} \\
\frac{\partial^2 l}{\partial \beta \partial \theta_0} &= - \sum_i \frac{\delta_i t_i^{[\alpha_0(1-x_i) + \alpha_1 x_i]} e^{x_i^T \beta [\alpha_0(1-x_i) + \alpha_1 x_i]} x_i [\alpha_0(1-x_i) + \alpha_1 x_i]}{(1 + \theta_0 t_i^{[\alpha_0(1-x_i) + \alpha_1 x_i]} e^{x_i^T \beta [\alpha_0(1-x_i) + \alpha_1 x_i]})^2} \\
&\quad - \sum_i t_i^\phi e^{x_i \beta [\alpha_0(1-x_i) + \alpha_1 x_i] - 1} x_i \left[ \frac{\alpha_0(1-x_i) + \alpha_1 x_i - \theta_0 t_i^{[\alpha_0(1-x_i) + \alpha_1 x_i]} e^{x_i \beta [\alpha_0(1-x_i) + \alpha_1 x_i]} - 1}{(1 + \theta_0 t_i^{[\alpha_0(1-x_i) + \alpha_1 x_i]} e^{x_i \beta [\alpha_0(1-x_i) + \alpha_1 x_i]})^2} \right] \\
\frac{\partial^2 l}{\partial \beta \partial \alpha_0} &= - \sum_i \delta_i x_i (1-x_i) - \sum_i (\delta_i + e^{-x_i^T \beta}) \theta_0 x_i t_i^{[\alpha_0(1-x_i) + \alpha_1 x_i]} e^{x_i^T \beta [\alpha_0(1-x_i) + \alpha_1 x_i]} (1-x_i) \times \\
&\quad \left[ \frac{1 + [\alpha_0(1-x_i) + \alpha_1 x_i] (\log t_i + x_i^T \beta)}{1 + \theta_0 t_i^{[\alpha_0(1-x_i) + \alpha_1 x_i]} e^{x_i^T \beta [\alpha_0(1-x_i) + \alpha_1 x_i]}} - \frac{[\alpha_0(1-x_i) + \alpha_1 x_i] \theta_0 t_i^{[\alpha_0(1-x_i) + \alpha_1 x_i]} e^{x_i^T \beta [\alpha_0(1-x_i) + \alpha_1 x_i]} (\log t_i + x_i^T \beta)}{(1 + \theta_0 t_i^{[\alpha_0(1-x_i) + \alpha_1 x_i]} e^{x_i^T \beta [\alpha_0(1-x_i) + \alpha_1 x_i]})^2} \right] \\
&\quad + \sum_i e^{-x_i^T \beta} x_i (1-x_i) (\log t_i + x_i^T \beta) \frac{\theta_0 t_i^{[\alpha_0(1-x_i) + \alpha_1 x_i]} e^{x_i^T \beta [\alpha_0(1-x_i) + \alpha_1 x_i]}}{1 + \theta_0 t_i^{[\alpha_0(1-x_i) + \alpha_1 x_i]} e^{x_i^T \beta [\alpha_0(1-x_i) + \alpha_1 x_i]}} \\
\frac{\partial^2 l}{\partial \beta \partial \alpha_1} &= - \sum_i \delta_i x_i^2 - \sum_i (\delta_i + e^{-x_i^T \beta}) \theta_0 x_i^2 t_i^{[\alpha_0(1-x_i) + \alpha_1 x_i]} e^{x_i^T \beta [\alpha_0(1-x_i) + \alpha_1 x_i]} \times \\
&\quad \left[ \frac{1 + [\alpha_0(1-x_i) + \alpha_1 x_i] (\log t_i + x_i^T \beta)}{1 + \theta_0 t_i^{[\alpha_0(1-x_i) + \alpha_1 x_i]} e^{x_i^T \beta [\alpha_0(1-x_i) + \alpha_1 x_i]}} - \frac{[\alpha_0(1-x_i) + \alpha_1 x_i] \theta_0 t_i^{[\alpha_0(1-x_i) + \alpha_1 x_i]} e^{x_i^T \beta [\alpha_0(1-x_i) + \alpha_1 x_i]} (\log t_i + x_i^T \beta)}{(1 + \theta_0 t_i^{[\alpha_0(1-x_i) + \alpha_1 x_i]} e^{x_i^T \beta [\alpha_0(1-x_i) + \alpha_1 x_i]})^2} \right] \\
&\quad + \sum_i e^{-x_i^T \beta} x_i^2 (\log t_i + x_i^T \beta) \frac{\theta_0 t_i^{[\alpha_0(1-x_i) + \alpha_1 x_i]} e^{x_i^T \beta [\alpha_0(1-x_i) + \alpha_1 x_i]}}{1 + \theta_0 t_i^{[\alpha_0(1-x_i) + \alpha_1 x_i]} e^{x_i^T \beta [\alpha_0(1-x_i) + \alpha_1 x_i]}} \\
\frac{\partial^2 l}{\partial \theta_0 \partial \alpha_0} &= - \sum_i (\delta_i + e^{-x_i^T \beta}) \frac{t_i^{[\alpha_0(1-x_i) + \alpha_1 x_i]} e^{x_i^T \beta [\alpha_0(1-x_i) + \alpha_1 x_i]} (1-x_i) (\log t_i + x_i^T \beta)}{(1 + \theta_0 t_i^{[\alpha_0(1-x_i) + \alpha_1 x_i]} e^{x_i^T \beta [\alpha_0(1-x_i) + \alpha_1 x_i]})^2}
\end{aligned}$$

$$\frac{\partial^2 l}{\partial \theta_0 \partial \alpha_1} = - \sum_i (\delta_i + e^{-x_i^T \beta}) \frac{t_i^{[\alpha_0(1-x_i) + \alpha_1 x_i]} e^{x_i^T \beta [\alpha_0(1-x_i) + \alpha_1 x_i]} x_i (\log t_i + x_i^T \beta)}{(1 + \theta_0 t_i^{[\alpha_0(1-x_i) + \alpha_1 x_i]} e^{x_i^T \beta [\alpha_0(1-x_i) + \alpha_1 x_i]})^2}$$

$$\frac{\partial^2 l}{\partial \alpha_0 \partial \alpha_1} = - \sum_i \frac{\delta_i (1-x_i) x_i}{[\alpha_0(1-x_i) + \alpha_1 x_i]^2}$$

$$- \sum_i (\delta_i + e^{-x_i^T \beta}) (1-x_i) x_i (\log t_i + x_i^T \beta)^2 \frac{\theta_0 t_i^{[\alpha_0(1-x_i) + \alpha_1 x_i]} e^{x_i^T \beta [\alpha_0(1-x_i) + \alpha_1 x_i]}}{(1 + \theta_0 t_i^{[\alpha_0(1-x_i) + \alpha_1 x_i]} e^{x_i^T \beta [\alpha_0(1-x_i) + \alpha_1 x_i]})^2}$$



## 부록C

### R과 Python 코드의 비교

```
##### 1) Kidney 자료 불러오기 및 카플란-마이어 생존곡선 #####  
#----- R -----#  
install.packages("KMsurv")  
library(KMsurv)  
data(kidney, package = "KMsurv")  
head(kidney)  
kidney$type = kidney$type-1  
kidney_data <- kidney[order(kidney$time),]  
  
kidney_data$group = ifelse(kidney_data$type==1, 0, 1)  
  
install.packages("survminer")  
library("survminer")  
install.packages("survival")  
library(survival)  
fit <- survfit(Surv(time,delta) ~ group, data=kidney_data)  
aa = ggsurvplot(fit, xlab = "Time (month)", font.x = 13, font.y = 13, font.tickslab = 13,  
               legend = c(0.15, 0.15), legend.title = "Group", font.legend = 13,  
               legend.labs = c("Treatment", "Control"), conf.int = FALSE,  
               palette = c("red", "blue"), censor.shape = "|", censor.size = 2.8,  
               linetype = c("solid", "dotted"), size = 1, risk.table = TRUE,
```

```

        risk.table.y.text.col = TRUE, risk.table.height = 0.2)
aa$plot = aa$plot + guides(col = FALSE)
aa$table = aa$table + theme(plot.title = element_text(size=12),
        axis.title.x = element_text(size=12),
        axis.title.y = element_text(size=12))

aa

#----- Python -----#
import pandas as pd
kidney = pd.read_csv("K:/Accelerated Hazard model (AH model)/kidney data_KMsurv
package_early cross/kidney.csv")
kidney
T = kidney["time"]
E = kidney["delta"]

trt = (kidney["type"]==1)

# Kaplan-Meier function
from lifelines import KaplanMeierFitter
kmf = KaplanMeierFitter()

import matplotlib.pyplot as plt
plt.rcParams["figure.figsize"] = (10, 8)
ax = plt.subplot(111)
kmf.fit(T[trt], event_observed=E[trt], label="percutaneously")

```

```
kmf.plot(ax=ax, ci_show = False, c = 'r', style = '-', fontsize=14)
```

```
kmf.fit(T[~trt], event_observed=E[~trt], label="surgically")
```

```
kmf.plot(ax=ax, ci_show = False, c = 'k', style = '--')
```

```
plt.ylim(0, 1);
```

```
ax.set_xlabel('Time (month)', fontsize=14)
```

```
ax.set_ylabel('Survival', fontsize=14)
```

```
ax.legend(loc='upper right', fontsize=14)
```

```
##### 2) 위험률 곡선 #####
```

```
#----- R -----#
```

```
install.packages("muhaZ")
```

```
library(muhaZ)
```

```
# Model-based
```

```
kidney$type = kidney$type-1
```

```
head(kidney)
```

```
attach(kidney)
```

```
df <- data.frame(time, delta,type)
```

```
kidney <- df[order(df$time),]
```

```

y = kidney$time
delta = kidney$delta
group = kidney$type

theta0 = 0.002663746
beta = 3.672099812
phi = 2.067170793

lam0 = theta0*phi*y^(phi-1)/(1+theta0*(y^phi))
lam1 = theta0*phi*(y^(phi-1))*exp(beta*(phi-1))/(1+theta0*(y^phi)*exp(beta*phi))

plot(lam0 ~ y, lwd = 2, type = "l", xlab = "Time (month)", ylim = c(0, 0.08), ylab =
"Hazard rate", main = "hazard rate")
lines(lam1 ~ y, lwd = 2, lty = "solid", col = "red")

# Kernel - based
data(kidney, package = "KMsurv")
head(kidney)

treatds = kidney[kidney$type==2,]
controlds = kidney[kidney$type==1,]

haztreat = with(treatds, muhaz(time, delta, bw.grid=30, bw.method = "local", max.time =
max(time)))
hazcontrol = with(controlds, muhaz(time, delta, bw.grid=30, bw.method = "local",

```

```

max.time = max(time)))

lines(hazcontrol, lty=2, lwd=1, col = "black")
lines(haztreat, lty=2, lwd=1, col = "red")

text(15, 0.061,"C")
text(15, 0.04,"C")

text(15, 0.013,"T", col = "red")
text(15, 0.006,"T", col = "red")

legend("topleft", legend=c("Model-based", "Kernel-based"), lty=1:2, lwd=2:1)

#----- Python -----#
from lifelines import NelsonAalenFitter
naf = NelsonAalenFitter()

# Cumulative hazard function
naf.fit(T[trt], event_observed=E[trt])
ax = naf.plot_hazard(ci_show = False, c = 'r', style = '--', fontsize=14, bandwidth=20)

naf.fit(T[~trt], event_observed=E[~trt])
naf.plot_hazard(ax=ax, ci_show = False, c = 'k', style = '--', bandwidth=20)

ax.set_xlabel('Time (month)', fontsize=14)

```

```
ax.set_ylabel('Hazard rate', fontsize=14)
```

```
import numpy as np  
theta0 = 0.002663746  
beta = 3.672099812  
phi = 2.067170793
```

```
lam0 = theta0*phi*T**(phi-1)/(1+theta0*(T**phi))  
lam1 = theta0*phi*(T**(phi-1))*np.exp(beta*(phi-1))/(1+theta0*(T**phi)*np.exp(beta*phi))
```

```
ax.plot(T, lam0, 'k-', linewidth = 2)  
ax.plot(T, lam1, 'r-', linewidth = 2)
```

```
plt.text(15, 0.06, "C", fontsize=15)  
plt.text(15, 0.045, "C", fontsize=15)  
plt.text(15, 0.001, "T", c = "r", fontsize=15)  
plt.text(15, 0.008, "T", c = "r",fontsize=15)
```

```
##### 3) LLG-AH 모형 적합 #####
```

```
#----- R -----#
```

```
x <- model.matrix(~group + 0)  
n <- nrow(x)  
p <- ncol(x)  
X <- matrix(x, n, p)
```

```

time = matrix(time, n, 1)
delta = matrix(delta, n, 1)

# initial value
iter = 1
beta = 3.731337
theta0 = 0.002132
phi = 2.147429

repeat {
  theta0 = ifelse(theta0 < 0, 0.000001, theta0)
  phi = ifelse(phi < 0, 0.000001, phi)

  A = (time^(phi))*exp(X*beta*phi)
  B = theta0*A
  C = 1 + B
  D = (time^(phi))*exp(X*beta*(phi-1))

# first derivative
d.beta = sum(delta*X*(phi-1)) - sum(delta*B*X*phi/C) - sum(theta0*D*X*phi/C) +
  sum((log(C))*exp(-X*beta)*X)
d.theta0 = sum(delta/theta0) - sum((delta*A)/C) - sum((exp(-X*beta))*A/C)
d.phi = sum(delta/phi) + sum(delta*(log(time))) + sum(delta*X*beta) -
  sum(delta*B*(log(time)+X*beta)/C) - sum(exp(-X*beta)*B*((log(time))+X*beta)/C)

```

# second derivative

```

dd.beta = -sum(delta*B*(X^2)*(phi^2)/(C^2)) -
  sum(theta0*D*(X^2)*phi*((phi-1)*C-phi*B)/(C^2)) + sum(D*(X^2)*theta0*phi/C) -
  sum((X^2)*(log(C))*exp(-X*beta))
dd.theta0 = -sum(delta/(theta0)^2) + sum(delta*(A^2)/(C^2)) +
  sum(exp(-X*beta)*(A^2)/(C^2))
dd.phi = -sum(delta/(phi^2)) -
  sum(delta*theta0*(((log(time))+X*beta)^2)*A*((1/C)-(B/(C^2)))) -
  sum(theta0*(((log(time))+X*beta)^2)*D*((1/C)-(B/(C^2))))
dd.betatheta0 = -sum(delta*A*X*phi*((1/C)-(B/(C^2)))) -
  sum(D*X*(((phi-1)/C)-(phi*B/(C^2))))
dd.betaphi = sum(delta*X) -
  sum(delta*X*B*(((1+phi*log(time)+phi*X*beta)/C)-((phi*B*(log(time)+X*beta))/(C^2)))) -
  sum(X*theta0*D*(((1+phi*log(time)+phi*X*beta)/C)-((phi*B*(log(time)+X*beta))/(C^2)))) +
  sum(exp(-X*beta)*X*B*(log(time)+X*beta)/C)
dd.theta0phi = -sum(delta*((log(time))+X*beta)*A*((1/C)-(B/(C^2)))) -
  sum((exp(-X*beta))*((log(time))+X*beta)*A*((1/C)-(B/(C^2))))

```

# negative Hessian matrix

```

d.l = c(d.beta, d.theta0, d.phi)
onerow = cbind(dd.betatheta0, dd.beta, dd.betaphi)
tworow = cbind(dd.betatheta0, dd.theta0, dd.theta0phi)
threerow = cbind(dd.betaphi, dd.theta0phi, dd.phi)
dd.l = rbind(-onerow, -tworow, -threerow)

```

```

theta <- c(beta, theta0, phi)
theta_new = theta + solve(dd.l)%*%(d.l)
beta.new = theta_new[length(beta), 1]
theta0.new = theta_new[1:length(beta)+length(theta0), 1]
phi.new = theta_new[length(beta)+length(theta0)+length(phi), 1]

iter_est1 = c(iter, beta.new, theta0.new, phi.new)
names(iter_est1) = c("iter", "beta", "theta0", "phi")
print(iter_est1)

if((max(abs(theta_new-theta)))<0.000001|iter>100)(break)
theta <- theta_new
iter <- iter + 1

beta = theta_new[length(beta), 1]
theta0 = theta_new[1:length(beta)+length(theta0), 1]
phi = theta_new[length(beta)+length(theta0)+length(phi), 1]
}
print(iter)

# est
est <- matrix(round(theta_new, 6), ncol = 1)
rownames(est) <- c("beta", "theta0", "phi")

```

```

# se
SE <- matrix(round(sqrt(diag(solve(dd.1))),6),ncol = 1)
rownames(SE) <- c("beta", "theta", "phi")

# fitting
beta = 3.7313
theta0 = 0.0021
phi = 2.1474

st0 = 1/(1+(theta0*(time^(phi))))
st1 = (1/(1+(theta0*(time*exp(beta))^phi))^(exp(-beta))

lines(st0 ~ time, lwd = 2, lty = "dashed", col = "black")
lines(st1 ~ time, lwd = 2, lty = "solid", col = "red")

#----- Python -----#
import pandas as pd
kidney = pd.read_csv("K:/Accelerated Hazard model (AH model)/kidney data_KMsurv
package_early cross/kidney.csv")

time = kidney["time"]
delta = kidney["delta"]
X = kidney["type"]

n = len(X)

```

```

# initial value
theta0 = 0.0021
beta = 3.7313
phi = 2.1474

result = pd.DataFrame()

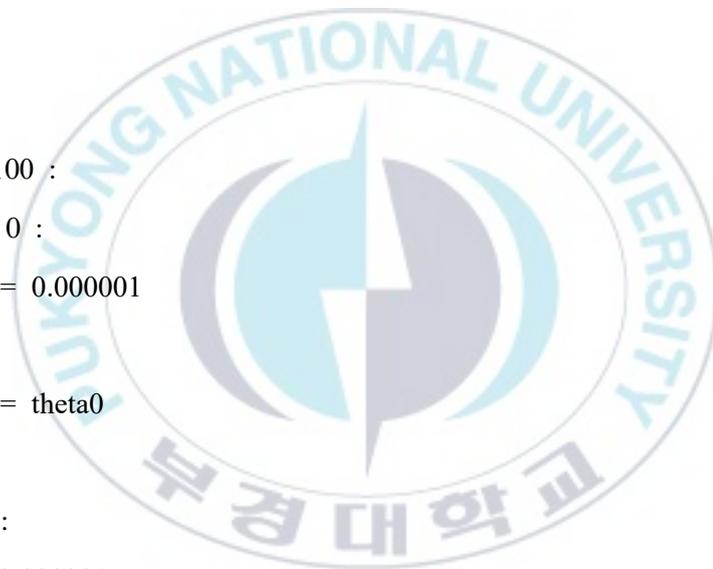
iters = 1

while iters <= 100 :
    if theta0 < 0 :
        theta0 = 0.000001
    else :
        theta0 = theta0

    if phi < 0 :
        phi = 0.000001
    else :
        phi = phi

    A = (time**(phi))*np.exp(X*beta*phi)
    B = theta0*A
    C = 1 + B
    D = (time**(phi))*np.exp(X*beta*(phi-1))

```



```

# first derivative
d_theta0 = sum(delta/theta0) - sum((delta*A)/C) - sum((np.exp(-X*beta))*A/C)
d_beta =sum(delta*X*(phi-1)) - sum(delta*B*X*phi/C) - sum(theta0*D*X*phi/C) +
sum((np.log(C))*np.exp(-X*beta)*X)
d_phi = sum(delta/phi) + sum(delta*(np.log(time))) + sum(delta*X*beta) -
sum(delta*B*(np.log(time)+X*beta)/C) -
sum(np.exp(-X*beta)*B*((np.log(time))+X*beta)/C)

# second derivative
dd_theta0 = -sum(delta/(theta0)**2) + sum(delta*(A**2)/(C**2)) +
sum(np.exp(-X*beta)*(A**2)/(C**2))
dd_beta = -sum(delta*B*(X**2)*(phi**2)/(C**2)) -
sum(theta0*D*(X**2)*phi*((phi-1)*C-phi*B)/(C**2)) +
sum(D*(X**2)*theta0*phi/C) - sum((X**2)*(np.log(C))*np.exp(-X*beta))
dd_phi = -sum(delta/(phi**2)) -
sum(delta*theta0*(((np.log(time))+X*beta)**2)*A*((1/C)-(B/(C**2)))) -
sum(theta0*(((np.log(time))+X*beta)**2)*D*((1/C)-(B/(C**2))))
dd_theta0beta = -sum(delta*A*X*phi*((1/C)-(B/(C**2)))) -
sum(D*X*(((phi-1)/C)-(phi*B/(C**2))))
dd_theta0phi = -sum(delta*((np.log(time))+X*beta)*A*((1/C)-(B/(C**2)))) -
sum((np.exp(-X*beta))*((np.log(time))+X*beta)*A*((1/C)-(B/(C**2))))
dd_betaphi = sum(delta*X) -
sum(delta*X*B*(((1+phi*np.log(time))+phi*X*beta)/C)-
((phi*B*(np.log(time)+X*beta))/(C**2)))) -

```

```

sum(X*theta0*D*(((1+phi*np.log(time)+phi*X*beta)/C)-
((phi*B*(np.log(time)+X*beta))/(C**2)))) +
sum(np.exp(-X*beta)*X*B*(np.log(time)+X*beta)/C)

# negative Hessian matrix
d_l = np.array([d_theta0, d_beta, d_phi])
dd_l = np.matrix([[ - dd_theta0, - dd_theta0beta, - dd_theta0phi],
                  [ - dd_theta0beta, - dd_beta, - dd_betaphi],
                  [ - dd_theta0phi, - dd_betaphi, - dd_phi]])
theta = np.array([theta0, beta, phi])

theta_new = theta + np.matmul(np.linalg.inv(dd_l) , d_l)
theta_new = np.array(theta_new)

theta0_new = theta_new[0][0]
beta_new = theta_new[0][1]
phi_new = theta_new[0][2]

diff = max(abs(theta0 - theta0_new), abs(beta - beta_new), abs(phi - phi_new))

iterest = pd.DataFrame([iters, theta0, theta0_new, beta, beta_new, phi, phi_new,
diff])

```

```

if diff < 0.00001 :
    break
else :
    iters = iters + 1
    theta0 = theta0_new
    beta = beta_new
    phi = phi_new ;

result.rename(columns= {0 : "iters", 1 : "theta0", 2 : "theta0_new", 3 : "beta",
                        4 : "beta_new", 5 : "phi", 6 : "phi_new", 7 : "diff"},
              inplace = True)

print("\n\n", np.round(result,6))

est = np.array([theta_new, beta_new, phi_new])
stderr = np.sqrt(np.diag(np.linalg.inv(dd_1)))

# fitting
trt = (kidney["type"]==1)

# Kaplan-Meier function
from lifelines import KaplanMeierFitter
kmf = KaplanMeierFitter()

```

```

import matplotlib.pyplot as plt
plt.rcParams["figure.figsize"] = (10, 8)
ax = plt.subplot(111)

kmf.fit(T[trt], event_observed=E[trt], label="percutaneously")
kmf.plot(ax=ax, ci_show = False, c = 'r', style = '-', fontsize=14)

kmf.fit(T[~trt], event_observed=E[~trt], label="surgically")
kmf.plot(ax=ax, ci_show = False, c = 'k', style = '--')

plt.ylim(0, 1);

ax.set_xlabel('Time (month)', fontsize=14)
ax.set_ylabel('Survival', fontsize=14)
ax.legend(loc='upper right', fontsize=14)

import numpy as np

# initial value
beta = 3.7313
theta0 = 0.0021
phi = 2.1474

cumhaz0 = np.log(1 + theta0*(T**phi))
cumhaz1 = np.log(1 + theta0*(T**phi)*np.exp(beta*phi)) * np.exp(-beta)

```

```
st0 = np.exp(-cumhaz0)
```

```
st1 = np.exp(-cumhaz1)
```

```
ax.plot(T, st0, 'k--', linewidth = 2)
```

```
ax.plot(T, st1, 'r-', linewidth = 2)
```

```
##### 4) LLGT-AH 모형 적합 #####
```

```
#----- R -----#
```

```
x <- model.matrix(~group + 0)
```

```
n <- nrow(x)
```

```
p <- ncol(x)
```

```
X <- matrix(x, n, p)
```

```
time = matrix(time, n, 1)
```

```
delta = matrix(delta, n, 1)
```

```
# initial value
```

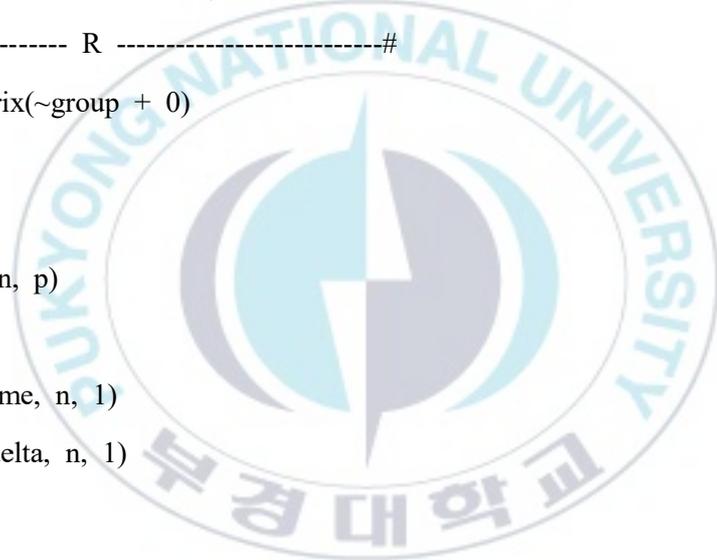
```
iter = 1
```

```
beta = 3.6699
```

```
theta0 = 0.0031
```

```
alpha0 = 1.9528
```

```
alpha1 = 2.2943
```



```

repeat {
  theta0 = ifelse(theta0 < 0, 0.000001, theta0)
  alpha0 = ifelse(alpha0 < 0, 0.000001, alpha0)
  alpha1 = ifelse(alpha1 < 0, 0.000001, alpha1)

  phi = alpha0*(1-X) + alpha1*X
  A = (time^(phi))*exp(X*beta*phi)
  B = theta0*A
  C = 1 + B
  D = (time^(phi))*exp(X*beta*(phi-1))

  # first derivative
  d.beta = sum(delta*X*(phi-1)) - sum(delta*B*X*phi/C) - sum(theta0*D*X*phi/C) +
    sum((log(C))*exp(-X*beta)*X)
  d.theta0 = sum(delta/theta0) - sum((delta*A)/C) - sum((exp(-X*beta))*A/C)
  d.alpha0 = sum(delta*(1-X)/phi) + sum(delta*log(time)*(1-X)) +
    sum(delta*X*beta*(1-X)) - sum((delta+exp(-X*beta))*B*(1-X)*(log(time)+X*beta)/C)
  d.alpha1 = sum(delta*X/phi) + sum(delta*log(time)*X) + sum(delta*X^2*beta) -
    sum((delta+exp(-X*beta))*B*X*(log(time)+X*beta)/C)

  # second derivative
  dd.beta = -sum(delta*B*(X^2)*(phi^2)/(C^2)) -
    sum(theta0*D*(X^2)*phi*((phi-1)*C-phi*B)/(C^2)) + sum(D*(X^2)*theta0*phi/C) -
    sum((X^2)*(log(C))*exp(-X*beta))
  dd.theta0 = -sum(delta/(theta0)^2) + sum(delta*(A^2)/(C^2)) +

```

```

sum(exp(-X*beta)*(A^2)/(C^2))
dd.alpha0 = -sum(delta*((1-X)^2)/(phi^2)) -
sum((delta+(exp(-X*beta))) * B * ((1-X)^2) * (((log(time))+X*beta)^2) * ((1/C)-(B/(C^2))))
dd.alpha1 = -sum(delta*(X^2)/(phi^2)) -
sum((delta+(exp(-X*beta))) * B * (X^2) * (((log(time))+X*beta)^2) * ((1/C)-(B/(C^2))))
dd.betatheta0 = -sum(delta*A*X*phi*((1/C)-(B/(C^2)))) -
sum(D*X*((phi-1)/C)-(phi*B/(C^2)))
dd.betaalpha0 = sum(delta*X*(1-X)) -
sum((delta+exp(-X*beta))*X*(1-X)*B*(((1+phi*((log(time))+X*beta))/C)-
((phi*B*((log(time))+X*beta))/(C^2)))) +
sum((exp(-X*beta))*X*(1-X)*B*((log(time))+X*beta)/C)
dd.betaalpha1 = sum(delta*(X^2)) -
sum((delta+exp(-X*beta))*(X^2)*B*(((1+phi*((log(time))+X*beta))/C)-
((phi*B*((log(time))+X*beta))/(C^2)))) +
sum((exp(-X*beta))*(X^2)*B*((log(time))+X*beta)/C)
dd.theta0alpha0 =
-sum((delta+(exp(-X*beta))) * A * (1-X) * ((log(time))+X*beta) * ((1/C)-(B/(C^2))))
dd.theta0alpha1 =
-sum((delta+(exp(-X*beta))) * A * X * ((log(time))+X*beta) * ((1/C)-(B/(C^2))))
dd.alpha0alpha1 = -sum(delta*(1-X)*X/(phi^2)) -
sum((delta+(exp(-X*beta))) * B * (1-X) * X * (((log(time))+X*beta)^2) * ((1/C)-(B/(C^2))))

# negative Hessian matrix
d.l = c(d.beta, d.theta0, d.alpha0, d.alpha1)
onerow = cbind(dd.beta, dd.betatheta0, dd.betaalpha0, dd.betaalpha1)

```

```

tworow = cbind(dd.betatheta0, dd.theta0, dd.theta0alpha0, dd.theta0alpha1)
threerow = cbind(dd.theta0alpha0, dd.theta0alpha0, dd.alpha0, dd.alpha0alpha1)
fourrow = cbind(dd.betaalpha1, dd.theta0alpha1, dd.alpha0alpha1, dd.alpha1)
dd.l = rbind(-onerow, -tworow, -threerow, -fourrow)

theta <- c(beta, theta0, alpha0, alpha1)
theta_new = theta + solve(dd.l)%*%(d.l)
beta.new = theta_new[length(beta), 1]
theta0.new = theta_new[1:length(beta)+length(theta0), 1]
alpha0.new = theta_new[length(beta)+length(theta0)+length(alpha0), 1]
alpha1.new = theta_new[length(beta)+length(theta0)+length(alpha0)+length(alpha1), 1]

iter_est1 = c(iter, beta.new, theta0.new, alpha0.new, alpha1.new)
names(iter_est1) = c("iter", "beta", "theta0", "alpha0", "alpha1")
print(iter_est1)

if((max(abs(theta_new-theta)))<0.00001|iter>200)(break)
theta <- theta_new
iter <- iter + 1

beta = theta_new[length(beta),1]
theta0 = theta_new[1:length(beta)+length(theta0),1]
alpha0 = theta_new[length(beta)+length(theta0)+length(alpha0),1]
alpha1 = theta_new[length(beta)+length(theta0)+length(alpha0)+length(alpha1),1]
}

```

```

# est
est <- matrix(round(theta_new, 6), ncol = 1)
rownames(est) <- c("beta", "theta0", "alpha0", "alpha1")

# se
SE <- matrix(round(sqrt(diag(solve(dd.1))),6),ncol = 1)
rownames(SE) <- c("beta", "theta0", "alpha0", "alpha1")

# fitting
beta = 3.6699
theta0 = 0.0031
alpha0 = 1.9528
alpha1 = 2.2943

st0 = 1/(1+(theta0*(time^(alpha0))))
st1 = (1/(1+(theta0*(time*exp(beta))^(alpha1))))^(exp(-beta))
lines(st0 ~ time, lwd = 2, lty = "dashed", col = "black")
lines(st1 ~ time, lwd = 2, lty = "solid", col = "red")

#----- Python -----#
import pandas as pd
kidney = pd.read_csv("K:/Accelerated Hazard model (AH model)/kidney data_KMsurv
package_early cross/kidney.csv")

```

```
time = kidney["time"]
delta = kidney["delta"]
X = kidney["type"]
n = len(X)

# initial value
theta0 = 0.0031
beta = 3.6699
alpha0 = 1.9528
alpha1 = 2.2943

result = pd.DataFrame()

iters = 1

while iters <= 100 :
    if theta0 < 0 :
        theta0 = 0.000001
    else :
        theta0 = theta0

    if alpha0 < 0 :
        alpha0 = 0.000001
    else :
        alpha0 = alpha0
```



```

if alpha1 < 0 :
    alpha1 = 0.000001
else :
    alpha1 = alpha1

phi = alpha0*(1-X) + alpha1*X
A = (time**(phi))*np.exp(X*beta*phi)
B = theta0*A
C = 1 + B
D = (time**(phi))*np.exp(X*beta*(phi-1))

# first derivative
d_theta0 = sum(delta/theta0) - sum((delta*A)/C) - sum((np.exp(-X*beta))*A/C)
d_beta = sum(delta*X*(phi-1)) - sum(delta*B*X*phi/C) - sum(theta0*D*X*phi/C) +
    sum((np.log(C))*np.exp(-X*beta)*X)
d_alpha0 = sum(delta*(1-X)/phi) + sum(delta*np.log(time)*(1-X)) +
    sum(delta*X*beta*(1-X)) -
    sum((delta+np.exp(-X*beta))*B*(1-X)*(np.log(time)+X*beta)/C)
d_alpha1 = sum(delta*X/phi) + sum(delta*np.log(time)*X) + sum(delta*X**2*beta) -
    sum((delta+np.exp(-X*beta))*B*X*(np.log(time)+X*beta)/C)

# second derivative
dd_theta0 = -sum(delta/(theta0)**2) + sum(delta*(A**2)/(C**2)) +
    sum(np.exp(-X*beta)*(A**2)/(C**2))

```

$$\begin{aligned}
dd\_beta &= -\sum(\delta * B * (X^{**2}) * (\phi^{**2}) / (C^{**2})) - \\
&\quad \sum(\theta_0 * D * (X^{**2}) * \phi * ((\phi - 1) * C - \phi * B) / (C^{**2})) + \\
&\quad \sum(D * (X^{**2}) * \theta_0 * \phi / C) - \sum((X^{**2}) * (\ln(C)) * \exp(-X * \beta)) \\
dd\_alpha0 &= -\sum(\delta * ((1 - X)^{**2}) / (\phi^{**2})) - \\
&\quad \sum(((\delta + (\exp(-X * \beta))) * B * ((1 - X)^{**2}) * (((\ln(\text{time})) + X * \beta)^{**2}) * ((1/C) - \\
&\quad (B / (C^{**2})))) \\
dd\_alpha1 &= -\sum(\delta * (X^{**2}) / (\phi^{**2})) - \\
&\quad \sum(((\delta + (\exp(-X * \beta))) * B * (X^{**2}) * (((\ln(\text{time})) + X * \beta)^{**2}) * ((1/C) - \\
&\quad (B / (C^{**2})))) \\
dd\_theta0beta &= -\sum(\delta * A * X * \phi * ((1/C) - (B / (C^{**2})))) - \\
&\quad \sum(D * X * (((\phi - 1) / C) - (\phi * B / (C^{**2})))) \\
dd\_theta0alpha0 &= \\
&\quad -\sum(((\delta + (\exp(-X * \beta))) * A * (1 - X) * (((\ln(\text{time})) + X * \beta) * ((1/C) - (B / (C^{**2})))))) \\
dd\_theta0alpha1 &= \\
&\quad -\sum(((\delta + (\exp(-X * \beta))) * A * X * (((\ln(\text{time})) + X * \beta) * ((1/C) - (B / (C^{**2})))))) \\
dd\_betaalpha0 &= \sum(\delta * X * (1 - X)) - \\
&\quad \sum((\delta + \exp(-X * \beta)) * X * (1 - X) * B * \\
&\quad (((1 + \phi * ((\ln(\text{time})) + X * \beta)) / C) - ((\phi * B * ((\ln(\text{time})) + X * \beta)) / (C^{**2})))) + \\
&\quad \sum((\exp(-X * \beta)) * X * (1 - X) * B * ((\ln(\text{time})) + X * \beta) / C) \\
dd\_betaalpha1 &= \sum(\delta * (X^{**2})) - \sum((\delta + \exp(-X * \beta)) * (X^{**2}) * B * \\
&\quad (((1 + \phi * ((\ln(\text{time})) + X * \beta)) / C) - ((\phi * B * ((\ln(\text{time})) + X * \beta)) / (C^{**2})))) + \\
&\quad \sum((\exp(-X * \beta)) * (X^{**2}) * B * ((\ln(\text{time})) + X * \beta) / C) \\
dd\_alpha0alpha1 &= -\sum(\delta * (1 - X) * X / (\phi^{**2})) - \\
&\quad \sum(((\delta + (\exp(-X * \beta))) * B * (1 - X) * X * (((\ln(\text{time})) + X * \beta)^{**2}) * \\
&\quad ((1/C) - (B / (C^{**2}))))
\end{aligned}$$

```

# negative Hessian matrix
d_1 = np.array([d_theta0, d_beta, d_alpha0, d_alpha1])
dd_1 = np.matrix([[ - dd_theta0, - dd_theta0beta, - dd_theta0alpha0, - dd_theta0alpha1],
                  [ - dd_theta0beta, - dd_beta, - dd_betaalpha0, - dd_betaalpha1],
                  [ - dd_theta0alpha0, - dd_betaalpha0, - dd_alpha0, - dd_alpha0alpha1],
                  [ - dd_theta0alpha1, - dd_betaalpha1, - dd_alpha0alpha1, - dd_alpha1]])

theta = np.array([theta0, beta, alpha0, alpha1])

theta_new = theta + np.matmul(np.linalg.inv(dd_1) , d_1)
theta_new = np.array(theta_new)

theta0_new = theta_new[0][0]
beta_new = theta_new[0][1]
alpha0_new = theta_new[0][2]
alpha1_new = theta_new[0][3]

diff = max(abs(theta0 - theta0_new), abs(beta - beta_new),
           abs(alpha0 - alpha0_new), abs(alpha1 - alpha1_new))

iterest = pd.DataFrame([iters, theta0, theta0_new, beta, beta_new, alpha0,
                       alpha0_new, alpha1, alpha1_new, diff])

result = pd.concat([result,iterest.T], axis=0)

```

```

if diff < 0.00001 :
    break
else :
    iters = iters + 1
    theta0 = theta0_new
    beta = beta_new
    phi = phi_new ;

result.rename(columns= {0 : "iters", 1 : "theta0", 2 : "theta0_new", 3 : "beta",
                        4 : "beta_new", 5 : "alpha0", 6 : "alpha0_new",
                        7 : "alpha1", 8 : "alpha1_new", 9 : "diff"}, inplace = True)

print("\n\n", np.round(result,6))

est = np.array([theta_new, beta_new, alpha0_new, alpha1_new])
stderr = np.sqrt(np.diag(np.linalg.inv(dd_1)))

# fitting
trt = (kidney["type"]==1)

# Kaplan-Meier function
from lifelines import KaplanMeierFitter
kmf = KaplanMeierFitter()

```

```

import matplotlib.pyplot as plt
plt.rcParams["figure.figsize"] = (10, 8)
ax = plt.subplot(111)

kmf.fit(T[trt], event_observed=E[trt], label="percutaneously")
kmf.plot(ax=ax, ci_show = False, c = 'r', style = '-', fontsize=14)

kmf.fit(T[~trt], event_observed=E[~trt], label="surgically")
kmf.plot(ax=ax, ci_show = False, c = 'k', style = '--')

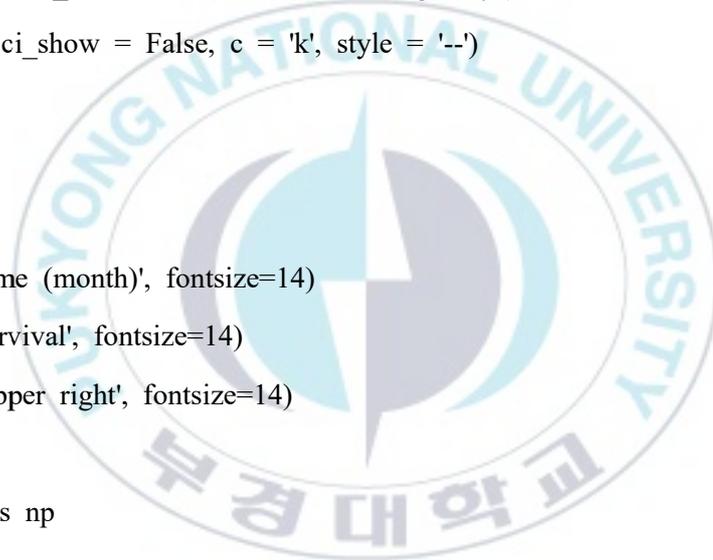
plt.ylim(0, 1);

ax.set_xlabel('Time (month)', fontsize=14)
ax.set_ylabel('Survival', fontsize=14)
ax.legend(loc='upper right', fontsize=14)

import numpy as np

# initial value
beta = 3.6699
theta0 = 0.0031
alpha0 = 1.9528
alpha1 = 2.2943

```



```
cumhaz0 = np.log(1 + theta0*(T**alpha0))
```

```
cumhaz1 = np.log(1 + theta0*(T**alpha1)*np.exp(beta*alpha1)) * np.exp(-beta)
```

```
st0 = np.exp(-cumhaz0)
```

```
st1 = np.exp(-cumhaz1)
```

```
ax.plot(T, st0, 'k--', linewidth = 2)
```

```
ax.plot(T, st1, 'r-', linewidth = 2)
```

