5-2005

Survival Model and Estimation for Lung Cancer Patients.

Xingchen Yuan
East Tennessee State University

Recommended Citation

Yuan, Xingchen, "Survival Model and Estimation for Lung Cancer Patients." (2005). *Electronic Theses and Dissertations.* Paper 1002. https://dc.etsu.edu/etd/1002

Survival Model and Estimation for Lung Cancer Patients

A thesis

presented to

the faculty of the Department of Mathematics

East Tennessee State University

In partial fulfillment

of the requirements for the degree

Master of Science in Mathematical Sciences

by

Xingchen Yuan

May 2005

Don Hong, Ph.D., Chair

Robert Gardner, Ph.D.

Jeff Knisley, Ph.D.

Tiejian Wu, Ph.D., MD

Keywords: Lung Cancer, Survival, hazard, Cox regression, Neural Network

ABSTRACT

Survival Model and Estimation for Lung Cancer Patients

by

Xingchen Yuan

Lung cancer is the most frequent fatal cancer in the United States. Following the notion in actuarial math analysis, we assume an exponential form for the baseline hazard function and combine Cox proportional hazard regression for the survival study of a group of lung cancer patients. The covariates in the hazard function are estimated by maximum likelihood estimation following the proportional hazards regression analysis. Although the proportional hazards model does not give an explicit baseline hazard function, the baseline hazard function can be estimated by fitting the data with a non-linear least square technique. The survival model is then examined by a neural network simulation. The neural network learns the survival pattern from available hospital data and gives survival prediction for random covariate combinations. The simulation results support the covariate estimation in the survival model.

DEDICATION

I dedicate this thesis to my wife, Xiaoling, and my son, Muyao

ACKNOWLEDGMENTS

I have been honored to be associated with the faculty of mathematics at East Tennessee State University from whom I have learned the principle of this art. Dr. Hong has been my mentor and advisor who taught me the principles of scientific research in this field and the ethics of this profession.

I would like to thank all the committee members for all of their precious advice and support during the preparation of this thesis.

I also would like to acknowledge William Wu at Biostatistics Shared Resource, Vanderbilt Ingram Cancer Center for providing the data set and many useful suggestions in this study. This research was supported in part by Lung Cancer SPORE (Special Program of Research Excellence) (P50 CA90949), Breast Cancer SPORE (1P50 CA98131-01), GI (5P50 CA95103-02), and Cancer Center Support Grant (CCSG) (P30 CA68485) by NSF IGMS #0408086 (awarded to Dr. Don Hong).

CONTENTS

Page

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

Cancer develops when the cells in a part of the body begin to grow out of control. It is the second most significant reason for US mortality. In 2001, cancer caused 553,768 deaths in the United States, which accounted for 22.9% of all deaths in that year[1] (Table 1.1). In the past fifty years, efforts have been made to reduce death rates for different diseases, but the death rate for cancer remains almost unchanged[2] (Figure 1.1).

Table 1.1 Cause of Death Distribution in USA

RANK	CAUSE OF DEATH	NO. OF DEATH	% OF ALL DEATH
1	Heart Diseases	700,142	29.0
2	*Cancer*	*553,768*	*22.9*
3	Cerebrovascular diseases	163,538	6.8
4	Chronic lower respiratory diseases	123,013	5.1
5	Accidents (Unintentional injuries)	101,537	4.2
6	Diabetes mellitus	71,372	3.0
7	Influenza and Pneumonia	62,034	2.6
8	Alzheimer’s disease	53,852	2.2
9	Nephritis	39,480	1.6

[1] Source: US Mortality Public Use Data Tape 2001, National Center for Health Statistics, Centers for Disease Control and Prevention, 2003.

[2] Sources: 1950 Mortality Data - CDC/NCHS, NVSS, Mortality Revised. 2001 Mortality Data–NVSR-Death Final Data 2001–Volume 52, No. 3.

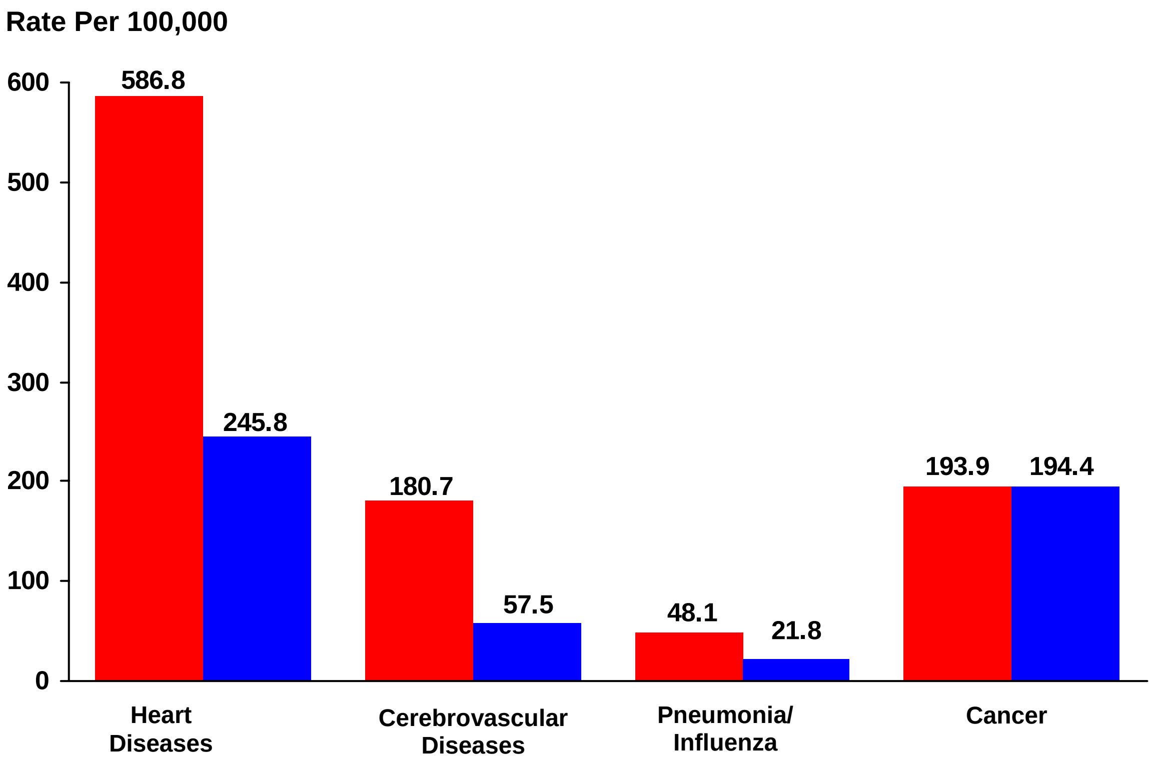


Figure 1.1 Changing of Death Rates in Last 50 Years

Left column: 1950; Right column: 2000

Among different types of cancers, lung cancer is the most frequent fatal cancer in the United States for both men and women. Each year, there are about 170,000 new cases of lung cancer in the U.S.A. and 150,000 deaths attributable to this disease. Men are affected somewhat more frequently (100,000 cases/year) than women (70,000 cases/year). Worldwide, there are 1 million new cases per year. Over the past 5 decades, the number of yearly cases has increased, and the worldwide incidence may double to 2 million per year in the coming decade. The average patient is 60 years old, and only 1% of patients are under 40 years old. Historically, about 90% of patients have died from the disease.

Recently, there has been a great deal of interest in modeling survival data of cancer patients (see [2], [8], [12] for example). Survival analysis is concerned with studying the time between entry to a study and a subsequent event, such as death. In practice, after a lung cancer patient is hospitalized, medical data regarding the patients' condition is recorded. This data set may include information such as the patient's survival time, the tumor's stage, the health grade, the disease free time etc. With these data, we wish to predict a patient's survival chance, or a group of patients' survival distribution over time.

The goal of this study was to develop a survival model for relating the hospital data profile to censored survival data such as time to cancer death or recurrence. Censored survival times occur if the event of interest (i.e., death) does not occur for a patient during the study period. Traditionally, there are two approaches to modeling the unknown survival distribution. One is to assume a classical parametric model such as normal, lognormal, gamma, Weibull, Pareto or beta, and then use a histogram, kernel or other nonparametric estimate of the unknown density function. In [5], a survival density curve is estimated using a logspline model for lung cancer patients. This method is straightforward, but cannot reflect the contribution of patients' hospital conditions to the survival distribution. Another approach is the proportional hazards model, which was first proposed by Cox in 1972 and is also well-known as the Cox regression model [7]. The model can incorporate the patients' hospital conditions as a vector of covariates in the hazard function and can estimate the unknown parameters for the covariates by partial likelihood without assuming a structure of baseline hazard. In this study, however, we proposed an exponential structure of the baseline hazard function following the notion in actuarial mathematics, and estimated the parameters by the available censored survival data so that the explicit survival function was determined. This estimation was achieved by a least squares fit for the cumulative hazard value computed by the statistical software, SPSS.

In a survey study, the design parameters for the survey are sometimes related to the hazard function but are not fit into the model. On some other occasions, the independence assumption of the covariates may be violated. Sometimes correlations exist within each level of nesting. This could cause biases and affect variances of parameter estimation [10,11]. Therefore, tests need to be done to evaluate the goodness of the estimated survival function. There are two popular ways to test the model in the survival analysis. One is to use 1/2 or 2/3 of the time scale in the survival data to determine the parameters and then use the whole data set to validate the model; another is to use the whole data set to set up the model, and then use a resample method to check the model. In this study, due to the lack of patient data, we proposed a neural network model to simulate the patients' survival pattern and used the neural network to generate a long list of "virtual data" to test the survival model.

The thesis is organized as follows: In chapter 2, we give a description for the survival model. In this chapter, we first introduce the concepts of hazard function and survival function and their relationship, followed by an outline of the method of proportional hazard model, after which we propose and justify the exponential form for baseline hazard function. In chapter 3, we discuss the parameter estimation by statistical methods including maximum likelihood estimation (MLE) and non-linear least square estimation (LSE). We also introduce the idea and concept of neural network and set up the proper neural network by Matlab programs for test purposes. In chapter 4, we present the computational result with actual patient data. Discussions and conclusions are given in chapter 5.

CHAPTER 2

DESCRIPTION OF MODELS

2.1 Survival Function and Hazard Function

Following the notion in Actuarial Mathematics [4], we denote by *T* a non-negative random variable representing the failure time of an individual in the population. If *T* is distributed with a probability density function (pdf) *f(t)*, then the cumulative distribution function (cdf) is

$$F(t) = \Pr\{T \le t\} = \int_0^t f(z)dz\,, \tag{2.1}$$

which gives the probability that the event has duration t. The survival function, *S*(*t*), is defined as the complement of the cdf of *T*. That is,

$$S(t) = \Pr\{T > t\} = 1 - F(t) = \int_t^\infty f(z)dz \tag{2.2}$$

The survival function gives the probability of being alive at duration t. Naturally, we have *S(0)=1* and $S(t) \to 0$ as $t \to \infty$.

An alternative characterization of the distribution of *T* is given by the hazard function. Sometimes it is also called the force of mortality, the mortality intensity function, or the failure rate. The hazard function is the probability that an individual will experience an event (for example, death) within a small time interval, given that the individual has survived up to the beginning of the interval. It can therefore be interpreted as the instantaneous risk of occurrence of dying at time t.

The hazard function h(t) can be expressed using the following equation:

$$h(t) = \lim_{dt \to 0} \frac{\Pr\{t < T \le t + dt | T > t\}}{dt} \qquad (2.3)$$

The numerator of this expression is the conditional probability that the event will occur in the interval *(t, t+dt)* given that it has not occurred before, and the denominator is the width of the interval. We obtain a rate of event occurrence per unit of time. Taking the limit as the width of the interval goes down to zero, we obtain an instantaneous rate of occurrence.

The conditional probability in the numerator may be written as the ratio of the joint probability that *T* is in the interval (*t, t* + *dt*) and *T* > *t*, to the probability of the condition *T* > *t*. The former may be written as *f(t)dt* for small *dt*, while the latter is *S(t)* by definition. Dividing by *dt* and taking the limit, we obtain

$$\begin{aligned} h(t) &= \frac{f(t)}{S(t)} \\ &= \frac{\frac{d}{dt}F(t)}{S(t)} = \frac{\frac{d}{dt}\left(1 - S(t)\right)}{S(t)} = \frac{-\frac{d}{dt}S(t)}{S(t)} \\ &= -\frac{S'(t)}{S(t)} \end{aligned} \qquad (2.4)$$

This equation suggests the relationship between the survival function and the hazard function. That is, the rate of occurrence of the event at duration t equals the density of events at t divided by the probability of surviving to that duration without experiencing the event.

Furthermore, equation (2.4) suggests that

$$h(t) = -\frac{d}{dt}\log S(t), \qquad (2.5)$$

and then,

$$\log S(t) = -\int_0^t h(z)dz + C \quad . \tag{2.6}$$

The boundary condition *S(0)=1* implies *C=0*, and thus

$$S(t) = \exp\left\{-\int_0^t h(z)dz\right\} \quad . \tag{2.7}$$

Combining (2.7) with (2.4), we get

$$f(t) = h(t)S(t) = h(t)\exp\left\{-\int_0^t h(z)dz\right\} . \tag{2.8}$$

2.2 Survival Times and Kaplan-Meier Method

In our study, the survival time is counted as the time period from lung cancer detection to death. A significant feature of survival times is that the event of interest is very rarely observed in all subjects. For example, although the patients may be followed up for several years, there will be some patients who are still alive at the end of the study. We do not, therefore, know what their survival time is after the cancer detection. Such survival times are termed *censored*. There are also many other reasons leading to censoring, such as

- Death from unrelated causes
- Loss of follow-up
- Termination of study

From a set of observed survival times (including censored times) in a sample of individuals, we can estimate the proportion of the population of such people who would survive a given length of time under the same circumstances. This method is called the product limit or *Kaplan-Meier method.*

Suppose there are n individuals and k distinct failure times $t_1 < t_2 < \ldots < t_k$. Let d_j be the number deaths at time t_j. Let n_j be number of individuals at risk at time t_j, that is the number of individuals alive and uncensored just prior to time t_j. If a censoring time is a t_j, they are assumed to be censored just after the deaths.

With these assumptions, the Kaplan-Meier estimator for the survival function is

$$\hat{S}(t) = \prod_{j:t_j<t} \frac{n_j - d_j}{n_j} \quad . \tag{2-9}$$

This gives the same answer no matter if there is censoring or not.

However, Kaplan-Meier has its limitation. It only allows comparisons with discrete predictors; besides, it does not allow for additional structure to be included in the analysis. To permit the patients' hospital conditions as a vector of covariates in the hazard function and survival function, we may consider the Cox regression, i.e., the proportional hazards model.

2.3 Cox Regression

A Cox model is a well-recognized statistical technique for exploring the relationship between the survival of a patient and a set of explanatory variables (see [1], [16] for example). We call these explanatory variables *covariates*.

Suppose that we have collected n patients with lung cancer. For the i^{th} patient, let $(t_i; \delta_i)$ be the observed phenotype, where t_i is the failure time (or when death happens) when $\delta_i = 1$, and is the censoring time (e.g., time of last known being cancer-free) when $\delta_i = 0$. Let $x_i = (x_{i1}, \cdots x_{ip})$ be the vector of p covariates for the i^{th} sample taken from the i^{th} patient. We assume the following general Cox model and the hazard function for the i^{th} patient is modeled as

$$h(t|x_i) = h_0(t)\exp(f(x_i)), \tag{2.10}$$

where $h_0(t)$ is called the baseline hazard function. Although $f(x_i)$ may assume many formats, a popular and simple model for $f(x)$ is

$$f(x_i) = x_i\beta = x_{i1}\beta_1 + \cdots + x_{ip}\beta_p \tag{2.11}$$

where β is a column vector of coefficients. In this equation, it is assumed that the effects of the different covariates on survival are constant over time and are addictive in a particular scale. A Cox model makes no assumptions about the form of $h_0(t)$, but assumes a parametric form for the effect of the covariates (predictors) on the hazard. In this sense, a Cox model is a semi-parametric model.

The parameter β can be estimated by the partial likelihood method. Let the observed follow up time of the i^{th} individual be t_i with corresponding covariates x_i, $i = 1, ..., n$. The conditional probability for i^{th} individual failing at t_i given that the individual is from the risk set $R(t_i)$ (i.e., $R(t_i) = \{j: t_j >= t_i\}$) is [10] :

$$\frac{h_0(t_i)\exp(x_i\beta)}{\sum_{l\in R(t_i)} h_0(t_i)\exp(x_l\beta)}. \tag{2.12}$$

Assume there are K failures; the partial likelihood function is then:

$$\prod_{i=1}^{K} \frac{\exp(x_i\beta)}{\sum_{l\in R(t_i)} \exp(x_l\beta)}. \tag{2.13}$$

Recalling the definition of δ_i at the beginning of this section, the partial likelihood function can be expressed as:

$$L(\beta)=\prod_{i=1}^{n}\left[\frac{\exp(x_i\beta)}{\sum_{j=1}^{n}y_j(t)\exp(x_j\beta)}\right]^{\delta_i}, \qquad (2.14)$$

where $y_j(t)=0$ when $t\le t_j$, otherwise $y_j(t)=1$. Equation (2.14) can be also written as

$$L(\beta)=\prod_{i\_uncensored}\left[\frac{\exp(x_i\beta)}{\sum_{j=1}^{n}y_j(t)\exp(x_j\beta)}\right]. \qquad (2.15)$$

For a sample of size n, the log partial likelihood for expression (2.15) is

$$l(\beta)=\log L(\beta)=\sum_{i\_uncensored}\left\{x_i\beta-\log\left[\sum_{j=1}^{n}y_j(t)\exp(x_j\beta)\right]\right\}. \qquad (2.16)$$

The maximum partial likelihood estimation of β can be obtained as a solution of the equation

$$\frac{\partial l(\beta)}{\partial\beta}=0,$$

and thus,

$$\sum_{i\_uncensored}^{n}x_i-\frac{\sum_{j=1}^{n}y_j(t)x_j\exp(x_j\beta)}{\sum_{j=1}^{n}y_j(t)\exp(x_j\beta)}=0. \qquad (2.17)$$

Cox and others have shown that this partial log-likelihood can be treated as an ordinary log-likelihood to derive valid (partial) MLE of β. Therefore we can estimate hazard ratios and confidence intervals using maximum likelihood techniques and the principle will be discussed in the next chapter. To avoid the baseline hazard, estimates are based on the partial as opposed to the full likelihood.

Usually, Cox proportional hazard regression model is a very useful tool to estimate the coefficients in a linear combination of covariates in survival analysis since both the SAS PHREG procedure and the SPSS Survival Package perform regression analysis of the survival data based on the proportional hazards model. However, because of the nature of proportional hazard regression, neither software packages give an explicit function expression for the baseline hazard function $h_0(t)$. In the next section, we will justify an explicit function of the baseline hazard function $h_0(t)$ and also estimate the parameters in $h_0(t)$ using non-linear least square technique based on the result obtained from the Cox regression for the survival function fitting the data set of lung cancer patients.

2.4 Baseline Hazard for Lung Cancer Patients

Like any cancer, the exact reason why people get lung cancer remains unknown. However, studies have shown that certain factors are strongly correlated with an increase in lung cancer. By rank, these factors are listed below [13]:

1. Tobacco smoking or exposure to smoke
2. Carcinogen exposures
3. Radiation exposure
4. Miscellaneous risks factors (such as old scars in the lungs)

The first three factors involve an interaction between an individual and the environment. Presumably, an individual is continuously exposed to and absorbs certain levels and amounts of smoke, radiation, or some kind of toxic material like a carcinogen which leads to lung cancer. Although a portion of the absorbed toxic material is discharged from the body, the cumulative effect of retained toxins contributes to the individual's death [6].

For a given τ in [*0,t*] and the infinitesimal time element [*τ, τ+dτ*], let the sum *$\delta d\tau+o(d\tau)$* be the probability that a unit of toxic material is absorbed during [*τ, τ+dτ*] and the sum *$\upsilon d\tau+o(d\tau)$* be the probability that a unit of toxic material in the body is discharged during [*τ, τ+dτ*]. Assuming that *δ* and *υ* are independent of time, then the probability that an individual will absorb a unit of toxic material during [*τ, τ+dτ*] and will retain it in his or her body up to time *t* is given by (see [6])

$$\delta d\tau \exp\left\{-(t-\tau)\upsilon\right\}. \tag{2.18}$$

Integrating (2.18) over all possible value of *τ* yields

$$\int_0^t \delta \exp\{-(t-\tau)\upsilon\}d\tau = \frac{\delta}{\upsilon}\left[1-\exp\{-\upsilon\cdot t\}\right], \tag{2.19}$$

The quantity in (2.19) is the expected amount of toxic material absorbed during the interval [*0, t*] and present in the body at time *t*. It also suggests a possible function format for the hazard for "exposure-caused cancer types." Suppose the baseline hazard for lung cancer patients is proportional to the quantity in (2.20), i.e.,

$$h_0(t) = \frac{a}{b}\left(1-\exp(-bt)\right). \tag{2.20}$$

Defining the cumulative baseline hazard function, *$H_0(t)$*, by integrating *$h_0(t)$* and applying boundary condition *$h_0(0)$*=0 yields

$$H_0(t) = \int_0^t h_0(x)dx = \frac{a}{b}\left[x-\frac{1}{b}\left(1-\exp(-bt)\right)\right]. \tag{2.21}$$

CHAPTER 3

STATISTICS METHODS AND NEURAL NETWORK

3.1 Maximum Likelihood Estimation

Maximum likelihood estimation begins with writing a mathematical expression known as the *likelihood function* of the sample data. Loosely speaking, the likelihood of a set of data is the probability of obtaining that particular set of data, given the chosen probability distribution model. This expression contains the unknown model parameters. The values of these parameters that maximize the sample likelihood are known as the *maximum likelihood estimates* (MLE).

Maximum likelihood estimation is a totally analytical maximization procedure. It applies to every form of censored or multi-censored data, and it is even possible to use the technique across several stress cells and to estimate acceleration model parameters at the same time as life distribution parameters. Moreover, MLE and likelihood functions generally have very desirable large sample properties:

- They become unbiased minimum variance estimators as the sample size increases.
- They have approximate normal distributions and approximate sample variances that can be calculated and used to generate confidence bounds.
- Likelihood functions can be used to test hypotheses about models and parameters.

Although MLE has many good attributes, it has an important drawback in that, with small numbers of failures (say less than 30 or sometimes less than 50), MLE might be heavily biased and the large sample optimality properties do not apply.

Let X be a continuous random variable with pdf

$$f(x;\beta_1,\beta_2\cdots\beta_p), \tag{3.1}$$

where $\beta_1,\beta_2\cdots\beta_p$ are p unknown constant parameters which need to be estimated.

Let $\beta^T=(\beta_1,\beta_2\cdots\beta_p)$. One conducts an experiment and obtains N independent observations, $x_1,x_2\cdots x_N$, which correspond in the case of life data analysis to failure times. The likelihood function is given by

$$\begin{aligned} L &= L(x_1,x_2,\cdots,x_N \left| \beta_1,\beta_2,\cdots,\beta_p \right.) \\ &= \prod_{i=1}^{N} f(x_i;\beta_1,\beta_2,\cdots,\beta_p), i=1,2,\cdots,N. \end{aligned} \tag{3.2}$$

The Logarithmic function is

$$l=\log L=\sum_{i=1}^{N}\log\bigl(f(x_i;\beta_1,\beta_2,\cdots,\beta_p)\bigr). \tag{3.3}$$

For the survival analysis, we assume (2.9) and (2.10). Then the pdf becomes

$$\begin{aligned} f(t_i\left|x_i\right.) &= h(t_i\left|x_i\right.)S(t_i\left|x_i\right.) \\ &= h_0(t_i)\exp\left\{x_i\beta-\int_0^{t_i}h_0(z)\cdot\exp(x_i\beta)dz\right\}. \end{aligned} \tag{3.4}$$

The log-likelihood function $l(\beta)$ has the expression

$$\begin{aligned} l &= \sum_{i=1}^{N}\log f(t_i\left|x_i\right.) \\ &= \sum\left\{\log h_0(t_i)+\left[x_i\beta-\int_0^{t_i}h_0(z)\exp(x_i\beta)dz\right]\right\} \\ &= N\log h_0(t_i)+\sum x_i\beta-\sum\int_0^{t_i}h_0(z)\exp(x_i\beta)dz \end{aligned} \tag{3.5}$$

When taking partial derivatives with respect to *β* to maximize *l*, the computation becomes very difficult. That is why in a Cox model, a proportional hazard model is used so that the term *h₀(z)* can be cancelled out for MLE calculation.

Recall (2.16), the maximum likelihood estimation for $\widehat{\beta}$ is $s(\widehat{\beta}) = 0$, where the score function

$$s(\beta) = \begin{pmatrix} \dfrac{\partial l(\beta)}{\partial \beta_1} \\ \dfrac{\partial l(\beta)}{\partial \beta_2} \\ \vdots \\ \dfrac{\partial l(\beta)}{\partial \beta_p} \end{pmatrix}. \qquad (3.6)$$

One of the nonlinear algorithms to compute this maximization is the Newton-Raphson iteration. The Newton-Raphson algorithm for computing $\widehat{\beta}$ starts with an initial guess $\widehat{\beta}^{(0)}$ and iteratively determine $\widehat{\beta}^{(m)}$ by using the formula

$$\widehat{\beta}^{(m)} = U^{-1}\left(\widehat{\beta}^{(m-1)}\right)s\left(\widehat{\beta}^{(m-1)}\right), \qquad (3.7)$$

where

$$\begin{aligned} U(\beta) &= -N \cdot Hessian(\beta) \\ &= N \cdot \begin{pmatrix} \dfrac{\partial^2 l(\beta)}{\partial^2 \beta_1} & \dfrac{\partial^2 l(\beta)}{\partial \beta_1 \partial \beta_2} & \cdots & \dfrac{\partial^2 l(\beta)}{\partial \beta_1 \partial \beta_p} \\ \dfrac{\partial^2 l(\beta)}{\partial \beta_2 \partial \beta_1} & \dfrac{\partial^2 l(\beta)}{\partial^2 \beta_2} & \cdots & \dfrac{\partial^2 l(\beta)}{\partial \beta_2 \partial \beta_p} \\ \vdots & \vdots & \ddots & \vdots \\ \dfrac{\partial^2 l(\beta)}{\partial \beta_p \partial \beta_1} & \dfrac{\partial^2 l(\beta)}{\partial \beta_p \partial \beta_2} & \cdots & \dfrac{\partial^2 l(\beta)}{\partial^2 \beta_p} \end{pmatrix}. \end{aligned} \qquad (3.8)$$

The Hessian matrix is positive definite, so it is strictly concave on *β*. However, the computation is obviously tough work. In practice, we use software to carry out this process for MLE.

3.2 Non-Linear Least Square Fit

Least square regression (LSE) is a very popular and useful tool used in statistics and many other fields. Suppose we want to find a relationship between a dependent (response) variable Y and an independent (predictor) variable *X*, under a statistical relation

$$Y = g(X|\theta) + \varepsilon\ , \tag{3.9}$$

where *ε* is the error, and *θ* is a vector of parameters to be estimated in function g. If g assumes a non-linear format in terms of X, we are facing a non-linear regression.

For $X = (x_1, x_2 \cdots x_m)^T$, $Y = (y_1, y_2 \cdots y_m)^T$, let

$$f_i(\theta) = y_i - \hat{y}_i = y_i - g(x_i|\theta). \tag{3.10}$$

Then, the non-linear least square regression is to find $\hat{\theta}$ which minimizes $F(\hat{\theta})$, where $F(\theta)$ is defined as

$$F(\theta) = \frac{1}{2}\sum_{i=1}^{m}(f_i(\theta))^2 = \frac{1}{2}\|f(\theta)\|^2 = \frac{1}{2}f(\theta)^T f(\theta)\,. \tag{3.11}$$

There are many algorithms for finding $\hat{\theta}$ including Gauss-Newton method, Levenberg-Marquardt method, Powell’s Dog Leg method, etc. (see [7]). We will use the

Gauss-Newton method. It is based on implementing first derivatives of the components of the vector function. In some special cases, it can give quadratic convergence the same as the Newton-method does for general optimization (see [8]).

The Gauss–Newton method is based on a linear approximation to the components of *f* in the neighborhood of θ. For small $\|h\|$ we see from the Taylor expansion that

$$f(\theta + h) = l(h) \equiv f(\theta) \cdot J(\theta)h\,. \tag{3.12}$$

J is the Jacobian matrix. Inserting this to the definition for *F* we get

$$\begin{aligned} F(\theta + h) = L(h) &\equiv \frac{1}{2} l(h)^T l(h) \\ &= \frac{1}{2} f^T f + h^T J^T f + \frac{1}{2} h^T J^T Jh \\ &= F(\theta) + h^T J^T f + \frac{1}{2} h^T J^T Jh \quad . \end{aligned} \tag{3.13}$$

The Gauss-Newton step $\hat{h}$ minimizes *L(h)*.

In real practice, the Gauss-Newton least squares fit for baseline hazard function can be achieved by using Matlab software package.

3.3 Neural Network Testing

In the Cox model, the main interest is usually about the parameter vector *β*. However, when one is interested in making predictions about the failure time for a given set of covariates, or when one assumes a parametric family for the baseline hazard function, just as what we have done, then it becomes important to test that h_0 is equal to some specified hazard rate function or to evaluate how stable h_0 is for different data sources [15]. In the field survival analysis, there are two popular ways to test a model. One is to use 1/2 or 2/3 of the time scale in the survival data to determine the parameters and then use the whole data set to examine the model. In our study, however, because of the short length of data (total 66 rows, among which around two-thirds are censored) and the high data demand from MLE (refer to section 3.1), this solution is not feasible. Another way is to use the whole data set to set up the model then use a resample method to check the model. As we have known, MLE relies heavily on the given data set, especially when the length of data is not too long. If we randomly resample the original data, the selected data for testing may be far from the "pattern" of the whole data set, e.g., have quite different mean and variance.

In the following, we propose an *artificial neural network* (ANN) testing model. First we let the ANN "learn" the patients' survival pattern from the given hospital data. Next, we use the ANN to generate a long list of "virtual data" and "simulate" the survival pattern, to test our covariate estimation and baseline hazard estimation. By this process, we also show the great potential as a research tool in survival analysis.

The concept of a neural network came up as early as the middle of this century. A Neural Network is an information processing paradigm that is inspired by the way biological nervous systems, such as the brain, process information. Or simply speaking, it is software that is "trained" by presenting it examples of input and the corresponding desired output.

Neural networks, with their remarkable ability to derive meaning from complicated or imprecise data, can be used to extract patterns and detect trends that are too complex to be noticed by either humans or other computer techniques. A trained neural network can be thought of as an "expert" in the category of information it has been given to analyze.

The typical structure of a feed forward neural network consists of a layer of *d* (the dimension of the futures) input units, a layer of output units, and a variable number of hidden layers of units, as shown in Figure 3.1. Generally, more layers result in higher accuracy, but also are more time-consuming on computation.

The construction of the ANN for this study and test results will be shown in the next chapter.

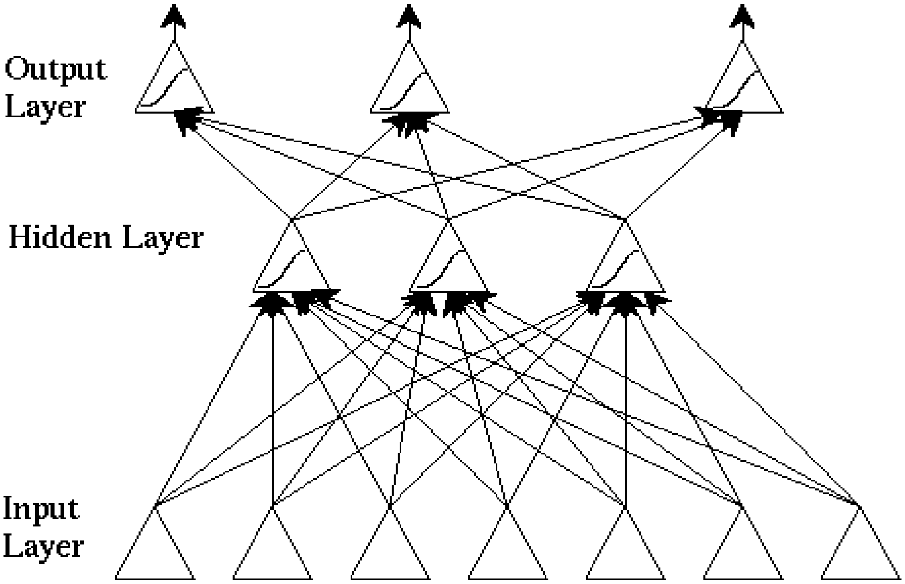


Figure 3.1 Typical Structure of Neural Network

CHAPTER 4

APPLICATION TO LUNG CANCER DATA

4.1 Data Structure

A data set records the survival times (S_INT, in months) of the lung cancer patients seen at Vanderbilt University School of Medicine Hospital. The data also record patients' hospital conditions including

- PT: patient term, ranges from T1 to T4
- PN: occurrence of lymph notes, a symptom of cancer invasion, ranges from N0 to N2
- STAGE: pathological diagnosis of cancer and it is ordinal, ranges from 1A to IV
- DF_INT: disease free time, in months
- GRADE: the fitness condition when patient in hospital, ranges from well to poor
- STATUS: indicating if the patient is alive (A) or dead (D). If the Status of a patient is "A" (alive), this row of data are censored.

In our study, we take PT, PN, STAGE, DF_INT, and GRADE as covariates in estimation. The original hospital data set records information for 66 patients and is listed in Appendix I. To perform the regression, we need to convert the categorical data in PT, PN, STAGE, STATUS, and GRADE columns to quantitative data. For example, for patient term, let *T1*=1, *T2*=2, *T3*=3 and *T4*=4.

4.2 Estimation for Covariates

The proportional hazard regression to estimate β was performed by SPSS. The results are shown in Appendix II.

The Cox regression gives the mean and standard deviation for each covariate in given data. The β is estimated at a certain significance level. For "patient term" and "grade," β is positive, which means a higher value for these two variables will result in higher hazard or risk of death. For "disease free time," β assumes a negative value. This means the longer the patient is disease free, the less likely that he or she will die shortly, which is reasonable. The β values for PN and STAGE are both near zero, which indicates that these two variables do not associate well with the hazard rate.

The Cox regression gives baseline cumulative hazard and overall cumulative hazard vs. survival time, at the mean value of covariates. To estimate the hazard function, we fix the covariates at their mean values, then use least squares regression to estimate the parameters a and b in (2.20) by fitting two columns of data in the survival table in Appendix II.

4.3 Estimation for Baseline Hazard Function

Starting from the results from Cox regression, let

X^T=Survival Time=[1 2 3 4 5 6 8 9 11 16 17 18 33],

H^T=Cum Baseline Hazard= [0.006 0.010 0.022 0.029 0.037 0.054 0.065 0.089 0.129 0.163 0.303 0.377 0.991].

Following the Gauss-Newton least square estimation discussed by section 3.2, we find estimations for *a* and *b*. The Matlab computation results are summarized below.

```
FITTEDMODEL =

   General model:
   FITTEDMODEL(x) = a/b*(x-1/b*(1-exp(-b*x)))
   Coefficients (with 95% confidence bounds):
   a =    0.002185  (0.001524, 0.002845)
   b =    0.01727  (-0.01574, 0.05029)

     GOODNESS =
          sse: 0.0129
          rsquare: 0.9854
          dfe: 11
          adjrsquare: 0.9840
          rmse: 0.0342
     OUTPUT =
          numobs: 13
          numparam: 2
          residuals: [13x1 double]
          Jacobian: [13x2 double]
          exitflag: 1
          iterations: 7
          funcCount: 22
          firstorderopt: 1.4601e-004
          algorithm: 'Gauss-Newton'
```

The estimated baseline hazard function is

$$h_0(t) = 0.1265\left(1 - \exp(-0.01727t)\right) \qquad (4\text{-}1)$$

Figure 4.1 shows the fit for cumulative baseline hazard. Figure 4.2 plots the baseline hazard as a function of time.

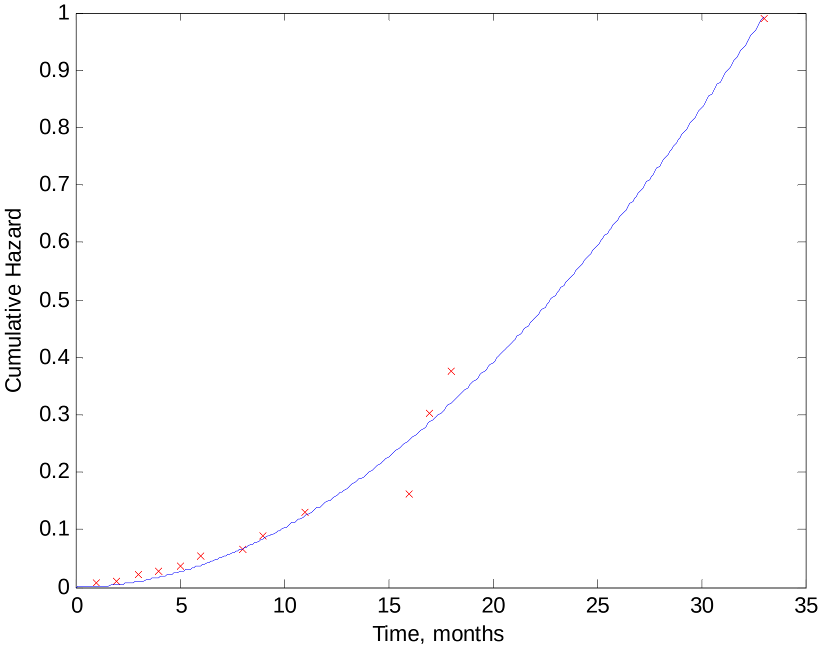


Figure 4.1 Fit for Cumulative Hazard

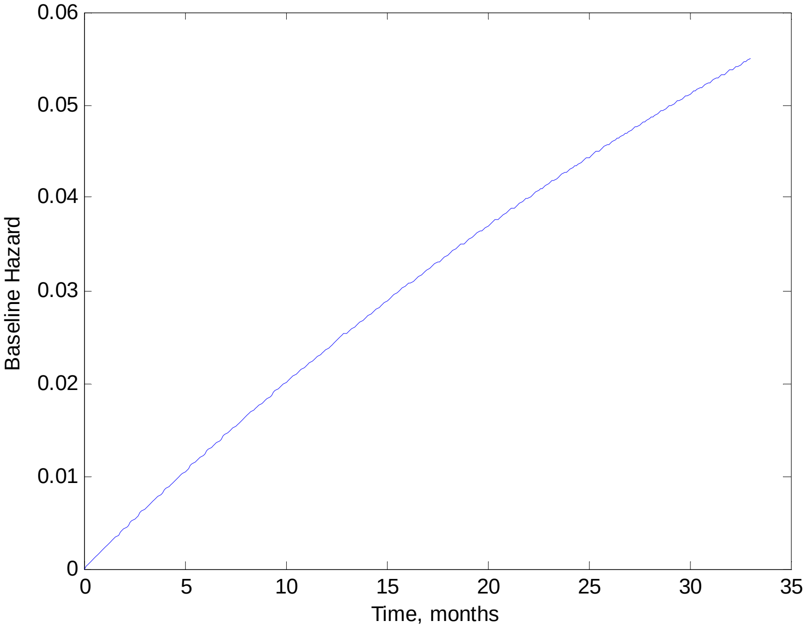


Figure 4.2 Baseline Hazard as a Function of Time

4.4 Survival Model Testing

With the help of MatLab function "**newff**", a feed-forward backpropagation network is constructed to simulate the survival model. This network has a total of three layers: an input layer of dimension 6, a hidden layer of dimension 6, and an output layer of dimension 1. The unit of output layer may assume value "0" or "1", representing "alive" and "dead" respectively. More hidden levels have been proven not to improve NN performance. Since the output values assume only two possible values, we use "**logsig**" as the nonlinear transfer function between layers.

When having "**traingda / learngdm**" as the training / learning function, the NN reaches best performance and the error rate for the training set is 9%. The error rate is defined as the rate of false "alive-dead" judgments for all 66 training cases. The network performance is shown in Figure 4.3.

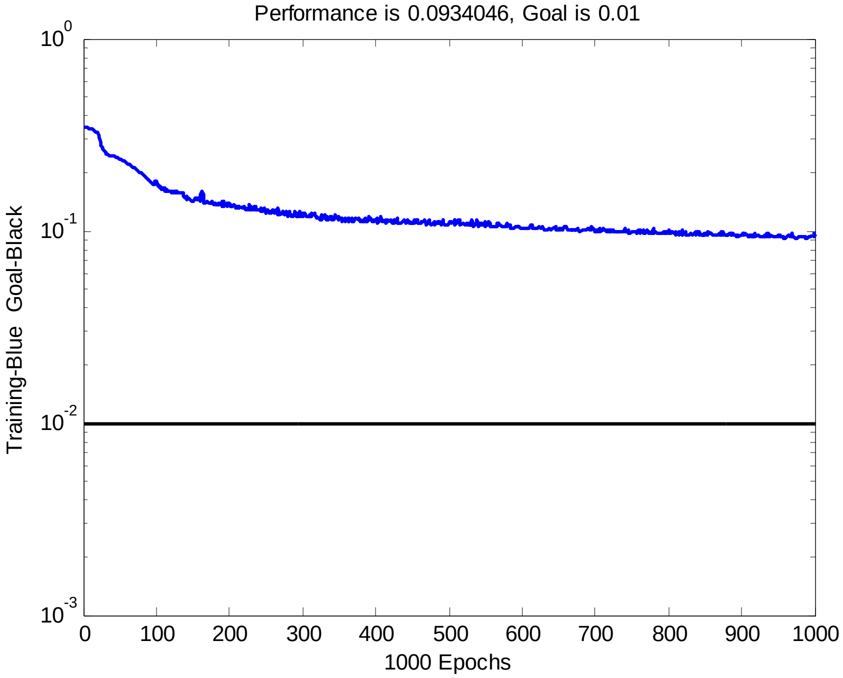


Figure 4.3 Network Performance over Epochs

After the ANN is set up, we generate a 1000 ×6 matrix to simulate 1000 patient records. Each column of the matrix corresponds to a covariate, and each row stores a patient's information on PT, PN, STAGE, S_INT, D_INT, and GRADE. Then, we use the trained ANN to judge the STATUS of the patient, as we "believe" the NN has learned the "right" survival pattern of lung cancer patients.

At first, we generate the data for each column randomly and uniformly distributed in the domain. For example, the domain for PN column is the closed interval [1, 4]. All numbers are rounded to integers. After a Cox regression analysis, the computation does not converge. This result shows that randomly generated data are not acceptable. The covariates for lung cancer patients must be distributed with a certain pattern.

Recall the Cox regression results for the original hospital data. The mean and standard deviation for each covariate are calculated. With these results, another 1000 ×6 matrix is generated. For each column, the generated data assume normal distribution with corresponding mean and standard deviation, as shown in Table 4.1. Still, all numbers are rounded to integers (disregarding that the rounding may shift the mean and deviations for each column).

Table 4.1 Statistics for Hospital Data and ANN Generated Data

	HOSPITAL DATA		ANN GENERATED DATA	
	Mean	SD	Mean	SD
PN	1.837	0.789	1.858171	0.768
PT	0.581	0.808	0.668378	0.671
STAGE	3.350	1.780	3.379877	1.634
DF_INT	10.050	9.203	7.533881	6.215
GRADE	2.750	1.157	2.727926	1.048
S_INT	14.125	9.878	14.03593	8.996

After a Cox regression and a least square fit for the cumulative baseline hazard as we did before, the baseline hazard for the ANN generated data is plotted as a function of time. It is compared to the baseline hazard function we found for the original hospital data, as shown in Figure 4.4.

Furthermore, define the score function

$$s(x;\beta) = x^T \cdot \beta \,. \tag{4-2}$$

Then the hazard function changes to be

$$h(t|x_i) = h_0(t)\exp(s) \ . \tag{4-3}$$

The score function determines the risk of death. The higher the score, the more likely a patient will die (or die sooner). Table 4.2 gives the scores for each patient for the NN generated data.

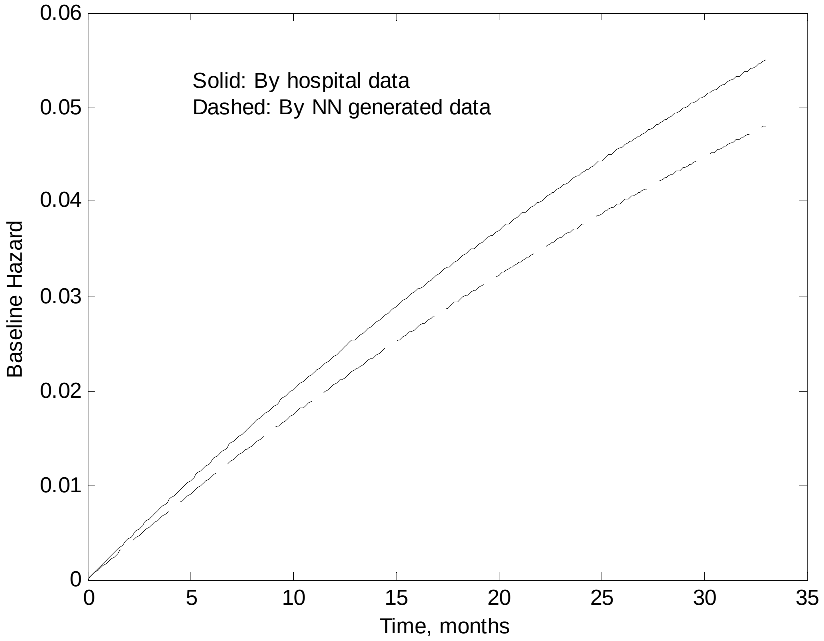


Figure 4.4 Estimated Baseline Functions

Table 4.2 Patients' Scores

PT	PN	STAGE	DF_INT	STAT	S_INT	STATUS	SCORE	(S_INT)*(STATUS)
2	0	3	1	0	1	-1	0.601	-1
1	1	5	1	2	1	1	1.865	1
2	0	3	1	3	1	-1	1.072	-1
3	0	3	1	2	1	-1	0.676	-1
2	1	2	1	2	1	-1	0.306	-1
2	0	5	1	3	1	1	1.952	1
1	0	3	1	2	1	1	1.154	1
2	0	2	1	1	1	-1	0.318	-1
2	1	4	2	2	2	1	0.945	2
1	1	3	0	3	2	1	1.383	2
2	2	3	2	3	2	-1	0.493	-2
1	0	3	2	4	2	1	1.227	2
1	1	6	2	1	2	1	1.907	2
2	1	3	2	3	2	-1	0.662	-2
2	1	2	2	1	2	-1	-0.092	-2
4	0	6	2	4	2	1	1.83	2
2	0	4	2	2	2	1	1.114	2
2	2	1	2	2	2	-1	-0.544	-2
2	1	3	2	2	2	-1	0.505	-2
1	2	4	2	0	2	-1	0.701	-2
1	0	4	2	0	2	1	1.039	2
2	1	2	3	4	3	-1	0.138	-3
2	1	6	3	1	3	1	1.427	3
2	2	2	3	2	3	-1	-0.345	-3
2	0	2	3	2	3	-1	-0.007	-3
…………Data Truncated ………								
3	0	5	10	2	31	-1	-0.613	-31
1	1	5	8	1	31	-1	0.021	-31
1	0	1	31	3	31	-1	-6.799	-31
2	2	4	10	2	31	-1	-1.152	-31
1	1	5	7	3	33	1	0.576	33
1	1	2	0	4	33	1	1.1	33
1	1	5	4	3	34	1	1.299	34
2	0	5	2	4	34	1	1.868	34
3	0	3	5	0	34	-1	-0.602	-34
1	1	4	15	4	34	-1	-1.635	-34
1	1	3	18	1	34	-1	-3.269	-34
3	1	6	11	3	34	-1	-0.426	-34
2	1	2	17	4	34	-1	-3.236	-34
1	1	3	0	3	34	1	1.383	34

A scatter plot for score vs. survival time is shown in Figure 4.5. Notice that time assumes a negative value if it is censored (patient is still alive.)

Figure 4.5 shows that when a patient scores a negative or very small value, he or she tends to survive; the lower the score, the longer he or she will live. A high positive score means higher percentage of death instead. This proves that proportional hazard regression is a good way to estimate β coefficients.

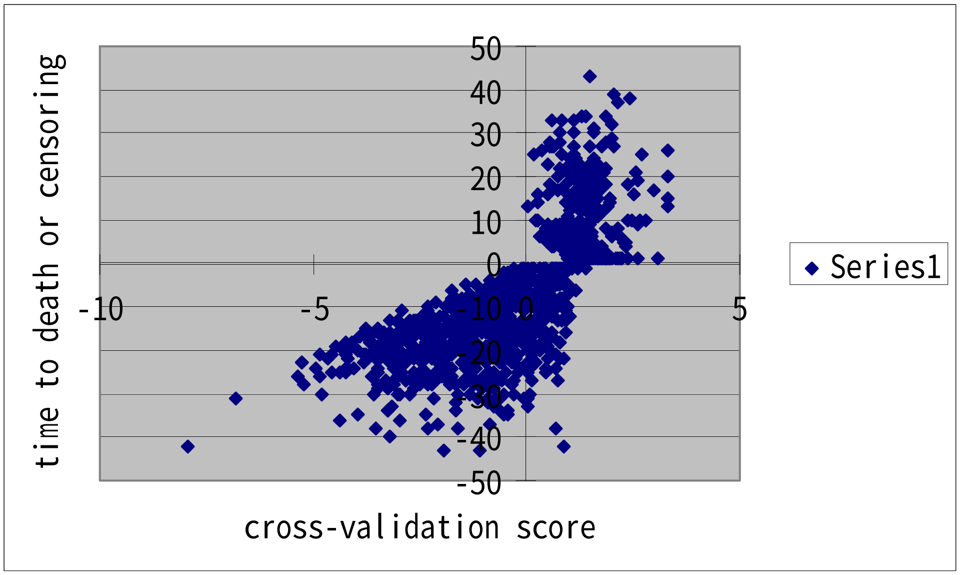


Figure 4.5 Scores vs. Time to Death or Censoring

CHAPTER 5

CONCLUSIONS AND FUTURE WORK

1. In this study we set up a survival model for lung cancer patients. This was achieved by three steps: using proportional hazard regression to estimate the coefficients for five covariates; using non-linear least square fit to estimate the exponential baseline hazard function; and using a neural network to examine the survival model. The analysis tools used in this research were SPSS, EXCEL and Matlab.
2. MLE is a powerful statistical tool but it has its own limitation. When the data length is short, MLE might be heavily biased. In this study, there were data for 66 patients, but two thirds were censored and only one third is used in MLE. The shortage of data results in a not very ideal significance level of the estimation.
3. Neural network simulation is a new idea for testing the model, especially when the original data set is short. Neural network application in survival analysis has promising prospects.
4. Although we assume a linear combination format in the score function, the five covariates are believed to be correlated with each other. A randomly generated covariate matrix may not result in a convergent Cox regression.
5. When the NN generated data assume the same mean and SD with the original data, they tend to have similar baseline hazard functions by LSE. This supports our assumption on the format of baseline function.
6. The score function provides a good indication for the risk of death. This backs up the Cox regression for β estimation.

7. Future work includes:

   - Regression for larger volume of hospital data for more stable β estimation.
   - Find out the correlation among the parameters. Assume a more accurate model for $f(x|\beta)$ in the hazard function and re-formulate the MLE in proportional hazards regression. This is tough work but truly worthwhile to do in the future.
   - Explore more NN applications in survival analysis.

BIBLIOGRAPHY

1. P. K. Andersen and R. D. Gill (1982), Cox's Regression Model for Counting Processes: A Large Sample Study, *The Annals of Statistics*, 1100-1120.

2. V. Bagdonavicius, M.A. Hafdi, and M. Nikulin (2004), Analysis of Survival Data with Cross-Effects of Survival Functions, *Biostatistics*, Vol. 5, 415-425.

3. D.A. Binder (1992), Fitting Cox's Proportional Hazards Models from Survey Data, *Biometrika*, 79 (1), 139-147.

4. N.L. Bowers, *Actuarial Mathematics*, Second Edition, Society of Actuaries, 1997.

5. Y. Chen, D. Hong, and Y. Shyr (2005), Logspline Density Estimation with an Application to the Study of Survival Data of Lung Cancer Patients, *manuscript*.

6. C. L. Chiang and P. M. Conforti (1989), A Survival Model and Estimation of Time to Tumor, *Mathematical Biosciences*, 94, 1-29.

7. D.R. Cox (1972), Regression Models and Life Tables, *Journal of the Royal Statistical Society,* Series B, 34, 187-220.

8. P.W. Dickman, A. Sloggett, M. Hills, and T. Hakulinen (2004), Regression Models for Relative Survival, *Statistics in Medicine*, 23, 51-64.

9. P.E. Frandsen, K. Jonasson, H.B. Nielsen, and O. Tingleff, *Unconstrained Optimization*, 3rd Edition, IMM, DTU, 2004.

10. H.-W. Liao (1998), A Simulation Study of Estimation in Stratified Proportional Hazards Model. *NESUG 1998 Proceedings*, 118-125

11. D.Y. Lin and L.J. Wei (1989), The Robust Inference for the Cox Proportional Hazards Model. *Journal of American Statistician Association*, 84, 1074-1078.

12. M.P. Little and E.G. Wright (2002), A Stochastic Carcinogenesis Model Incorporating Genomic Instability Fitted to Colon Cancer Data, *Mathematical Biosciences*, 183, 111-134.

13. Lung Cancer Transcripts, www.canceranswers.com, October 2004.

14. K. Madsen, H.B. Nielsen, and O. Tingleff, *Methods for Non-Linear Least Squares problems,* 2nd Edition, Informatics and Mathematical Modeling, Technical University of Denmark, April 2004.

15. E.A. Pena (1998), Smooth Goodness-of-Fit Tests for the Baseline Hazard in Cox's Proportional Hazards Mode, *Journal of American Statistical Association,* 93 (442), 673-692.

16. S.J. Walters, What Is a Cox Model, *Hayward Medical Communications*, volume 1, number 10, May 2001.

APPENDICES

APPENDIX A

Patients Data

#	PT	PN	STAGE	STAT	S_INT	DF_INT	GRADE
1	T1	N2	IIIA	D	11	5	mod
2	T4	N2	IIIB	D	11	9	poor
3	T1	N1	IV	D	17	0	poor
4	T2	N0	IB	A	24	24	well-mod
5	T2	N0	IV	D	9	0	mod-poor
6	T2	N2	IIIA	A	21	7	well-mod
7	T4	N0	IV	D	1	1	poor
8	T1	N0	IA	A	21	13	well-mod
9	T3	N0	IIB	D	2	0	mod-poor
10	T2	N0	IB	A	20	20	mod
11	T1	N0	IA	D	3	3	mod
12	T2	N0	IB	A	23	23	poor
13	T1	N0	IA	D	8	8	mod-poor
14	T2	N1	IIB	A	21	21	mod
15	T2	N0	IB	A	20	20	mod
16	T2	N0	IB	D	33	30	mod-poor
17	T2	N0	IB	A	18	18	mod-poor
18	T2	N2	IIIA	D	6	0	poor
19	T2	N2	IIIA	D	3	3	mod-poor
20	T1	N1	IIA	D	5	0	poor
21	T2	N2	IIIA	A	21	17	poor
22	T2	N0	IB	A	23	10	mod-poor
23	T2	N0	IB	A	26	26	well-mod
24	T2	N0	IB	A	26	26	mod
25	T1	N2	IIIA	D	18	0	poor
26	T2	N1	IIB	A	17	17	mod-poor
27	T2	N0	IIB	A	33	9	mod
28	T2	N0	IB	D	17	17	mod
29	T2	N0	IIB	A	42	42	mod-poor
30	T2	N0	IIB	D	16	5	poor
31	T1	N1	IIA	D	1	0	poor
32	T2	N0	IB	D	17	15	poor
33	T2	N2	IIIA	D	9	0	poor
34	T2	N2	IIIA	D	4	0	mod-poor
35	T2	N0	IB	A	2	1	poor

36	T2	N0	IB	A	5	1	well-mod
37	T2	N2	IIIA	A	6	6	mod
38	T1	N0	IA	A	1	1	well
39	T1	N0	IA	A	1	1	mod
40	T1	N0	IA	A	3	3	mod-poor
41	T1	N0	IA	A	1	1	mod-poor
42	T1	N0	IA	A	1	1	well-mod
43	T3	N0	IIB	A	1	1	well
44	T1	N0	IA	A	1	1	poor
45	T2	N0	IB	A	2	2	poor
46	T2	N0	IB	A	1	1	well-mod
47	T2	N0	IB	A	1	1	mod
48	T1	N0	IA	A	12	0	mod-poor
49	T1	N2	IIIA	A	6	4	mod-poor
50	T2	N0	IB	A	1	1	mod
51	T2	N0	IB	A	3	3	poor
52	T3	N0	IIB	A	10	4	poor
53	T3	N1	IIIA	D	6	6	poor
54	T2	N0	IB	A	1	0	mod
55	T4	N1	IIIB	A	2	0	mod-poor
56	T2	N0	IB	A	1	1	mod
57	T2	N0	IB	A	1	1	mod-poor
58	T2	N0	IB	A	5	4	poor
59	T1	N2	IIIA	A	1	1	poor
60	T1	N0	IA	A	1	1	mod
61	T1	N0	IA	A	7	7	poor
62	T2	N0	IB	A	2	2	mod
63	T2	N1	IIB	A	1	1	mod
64	T2	N2	IIIA	A	11	4	poor
65	T1	N0	IA	A	10	3	poor
66	T1	N0	IA	A	1	1	poor

APPENDIX B

Cox Regression Results

Case Processing Summary

		N	Percent
Cases available in analysis	Event	20	30.3%
	Censored	46	69.7%
	Total	66	100.0%
Cases dropped	Cases with missing values	0	.0%
	Cases with negative time	0	.0%
	Censored cases before the earliest event in a stratum	0	.0%
	Total	0	.0%
Total		66	100.0%

Note: Dependent Variable: S_TIME

Block 0: Beginning Block

Omnibus Tests of Model Coefficients

-2 Log Likelihood
133.322

Block 1: Method = Enter

Omnibus Tests of Model Coefficients

-2 Log Likelihood	Overall (score)			Change From Previous Step			Change From Previous Block		
	Chi-square	df	Sig.	Chi-square	df	Sig.	Chi-square	df	Sig.
109.761	19.581	5	.001	23.561	5	.000	23.561	5	.000

.

Variables in the Equation

	B	SE	Wald	df	Sig.	Exp(B)
PT	.550	.326	2.855	1	.091	1.734
PN	-.010	.315	.001	1	.974	.990
STAGE	-.057	.182	.099	1	.753	.944
D_FREE	-.141	.050	8.131	1	.004	.868
GRADE	.418	.280	2.219	1	.136	1.518

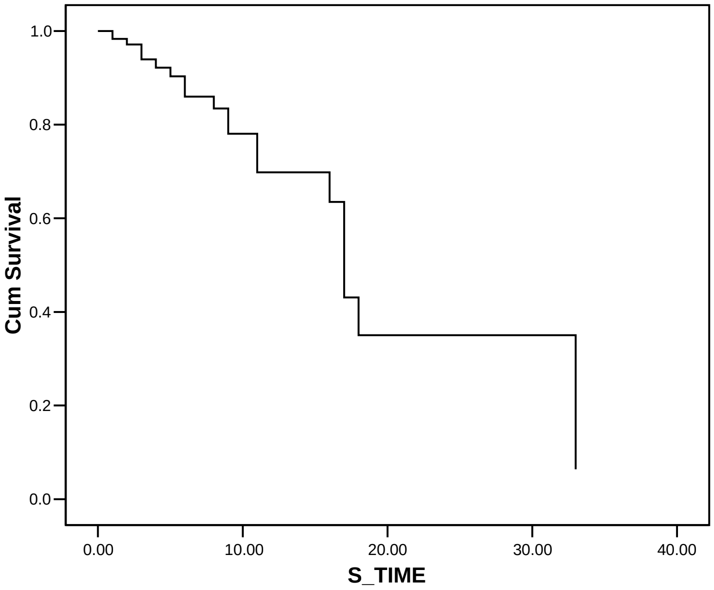
Covariate Means

	Mean
PT	1.833
PN	.515
STAGE	3.000
D_FREE	6.879
GRADE	2.788

Survival Table

		At mean of covariates		
Time	Baseline Cum Hazard	Survival	SE	Cum Hazard
1.00	.006	.983	.012	.017
2.00	.010	.971	.018	.029
3.00	.022	.939	.030	.062
4.00	.029	.922	.036	.082
5.00	.037	.903	.042	.102
6.00	.054	.860	.053	.151
8.00	.065	.835	.061	.181
9.00	.089	.780	.073	.248
11.00	.129	.698	.091	.359
16.00	.163	.635	.105	.455
17.00	.303	.431	.123	.842
18.00	.377	.350	.121	1.050
33.00	.991	.064	.115	2.755

Survival Function at mean of covariates



APPENDIX C

Selected Matlab Programs

```
%This program is to find out baseline hazard function by non-linear least
%square fit using original hospital data
%Also plot h0(x) and H0(x)
%Last revised on 10/02/04
%Copyrighted by Xingchen Yuan

clear;

T=[1 2 3 4 5 6 8 9 11 16 17 18 33]';
H=[0.006 0.010 0.022 0.029 0.037 0.054 0.065 0.089 0.129 0.163 0.303 0.377 0.991]';
plot(T,H,'rx');
hold on;

g=fittype('a/b*(x-1/b*(1-exp(-b*x)))');
F = FITOPTIONS('METHOD','NonlinearLeastSquares','StartPoint',[0.1,0.1]);
[FITTEDMODEL,GOODNESS,OUTPUT]=fit(T,H,g,F)

a=0.002185;
b=0.01727;
c=a/b;

for i=1:331
   x(i)=(i-1)*0.1;
   y(i)=c*(x(i)-1/b*(1-exp(-b*x(i))));
end

plot(x,y,'b');
xlabel('Time, months');
ylabel('Cumulative Hazard');

hold off;
h=0.1265*(1-exp(-0.01727*x));
plot(x,h);
xlabel('Time, months');
ylabel('Hazard');
```

```
%This program is to construct a Neural Network and Train the NN with hospital data
%Last revised on 10/05/04
%Copyrighted by Xingchen Yuan


clear;

data1=[1        2       5       5       2       11      1
4       2       6       9       4       11      1
1       1       7       0       4       17      1
2       0       2       24      1       24      0
2       0       7       0       3       9       1
3       2       5       7       1       21      0
4       0       7       1       4       1       1
1       0       1       13      1       21      0
3       0       4       0       3       2       1
2       0       2       20      2       20      0
1       0       1       3       2       3       1
2       0       2       23      4       23      0
1       0       1       8       3       8       1
2       1       4       21      2       21      0
2       0       2       20      2       20      0
2       0       2       30      3       33      1
2       0       2       18      3       18      0
2       2       5       0       4       6       1
2       2       5       3       3       3       1
1       1       3       0       4       5       1
2       2       5       17      4       21      0
2       0       2       10      3       23      0
2       0       2       26      1       26      0
2       0       2       26      2       26      0
1       2       5       0       4       18      1
2       1       4       17      3       17      0
2       0       4       9       2       33      0
2       0       2       17      2       17      1
2       0       4       42      3       42      0
2       0       4       5       4       16      1
1       1       3       0       4       1       1
2       0       2       15      4       17      1
2       2       5       0       4       9       1
2       2       5       0       3       4       1
2       0       2       1       4       2       0
2       0       2       1       1       5       0
2       2       5       6       2       6       0
1       0       1       1       0       1       0
1       0       1       1       2       1       0
1       0       1       3       3       3       0
1       0       1       1       3       1       0
1       0       1       1       1       1       0
2       0       4       1       0       1       0
1       0       1       1       4       1       0
2    0    2    2    4    2    0
2    0    2    1    1    1    0
2    0    2    1    1    1    0
1    0    1    0    3    12   0
1    2    5    4    3    6    0
2    0    2    1    2    1    0
2    0    2    3    4    3    0
3    0    4    4    4    10   0
3    1    5    6    4    6    1
2    0    2    0    2    1    0
4    1    6    0    3    2    0
2    0    2    1    2    1    0
2    0    2    1    3    1    0
2    0    2    4    4    5    0
1    2    5    1    4    1    0
1    0    1    1    2    1    0
1    0    1    7    4    7    0
2    0    2    2    2    2    0
2    1    2    1    2    1    0
2    2    5    4    4    11   0
1    0    1    3    4    10   0
1    0    1    1    4    1    0];

[a,b]=size(data1);
p=data1(1:a,1:b-1)';
t=data1(1:a,b)';

number=5;
pr=[min(p');max(p')]';
s=[number 3 1];                              %two layer network
funct={'logsig','logsig','logsig'};

net=newff(pr,s,funct,'traingda','learngdm','mse');
net.trainParam.epochs=1000;
net.trainparam.goal=0.01;
net=train(net,p,t);                          %train the NN

y=round(sim(net,p));
disp('Error for trainging samples: By Neural Network');
error1=sum(abs(y-t))/66                      % Neural Network Error for training data
```

```
%This program is to generated 1000 patient data to feed
%Data could be uniformly distributed or normal distributed
%Run this program ONLY AFTER running the NN construction program, which is given on last
page.
%Last revised on 10/05/04
%Copyrighted by Xingchen Yuan

n=1000

p1=round(NORMRND(1.833,0.789,1,n));
p2=round(NORMRND(0.581,0.808,1,n));
p3=round(NORMRND(3.35,1.78,1,n));
p4=round(NORMRND(10.05,9.023,1,n));
p5=round(NORMRND(2.75,1.157,1,n));
p6=round(NORMRND(14.125,9.878,1,n));

for i=1:n
   if p1(i)>4
      p1(i)=4;
   end
   if p1(i)<1
      p1(i)=1;
   end

   if p2(i)>2
      p2(i)=2;
   end
   if p2(i)<0
      p2(i)=0;
   end

   if p3(i)>7
      p3(i)=7;
   end
   if p3(i)<1
      p3(i)=1;
   end

   if p5(i)>4
      p5(i)=4;
   end
   if p5(i)<0
      p5(i)=0;
   end

   if p6(i)<1
      p6(i)=1;
   end

   if p4(i)>=p6(i)
      p4(i)=p6(i);
```

```
    end
    if p4(i)<0
        p4(i)=0;
    end
end

%p1=ceil(4*rand(1,n));   %1-4
%p2=floor(3*rand(1,n));  %0-2
%p3=ceil(7*rand(1,n));   %1-7
%p5=floor(5*rand(1,n));  %0-4
%p6=ceil(33*rand(1,n));  %1-42
%p4=round(rand(1)*p6);

pp=[p1; p2; p3; p4; p5; p6];

tt=round(sim(net,pp));
disp('total patients number is');
n
disp('the total death number is:');
death=sum(tt)

data2=[pp;tt]';
save data2.dat data2;
save data3.dat data2 -ASCII;
```

VITA

Xingchen Yuan

Personal Data: Date of Birth: September 10th, 1973

Place of Birth: Chongqing, China

Education: Tsinghua University, Beijing, China;

BE in Electrical Engineering, 1996

Chinese Academy of Sciences, Beijing, China;

MS in Electrical Engineering, 1998

Tennessee Technological University, Cookeville, TN, USA;

Ph.D. in Engineering, 2001

East Tennessee State University, Johnson City, Tennessee, USA;

MS in Mathematics,2005

Professional Experience Teaching Assistant/Associate, Mathematics Department, East Tennessee State University, 2003-2005