Chandan K. Reddy and Yan Li, "A Review of Clinical Predictive Models", in Healthcare Data Analytics, Chandan K. Reddy and Charu C. Aggarwal (eds.), CRC Press, 2014.

Contents

Chapter 1

A Review of Clinical Predictive Models

1.1 Introduction

Clinical prediction is one of the most important branches of healthcare data analytics. In this chapter, we provide a relatively comprehensive review of the supervised learning methods which have been employed in clinical prediction. Some of these methods are very basic and well studied statistical methods such as linear regression, logistic regression, and Bayesian models; some appeared with the rise of machine learning and data mining such as decision trees and artificial neural networks; in addition, survival models are motivated by clinical analysis. The remaining of this chapter is organized as follows.

There are multiple prediction aims in clinical analysis, and based on the outcomes of the prediction model these aims are fall into five categories: continuous outcomes, binary outcomes, categorical outcomes, ordinal outcomes, and survival outcomes. The first class of outcomes can be seen in medical costs prediction [1][2] and the estimation of some medical inspection [3]; linear regression and generalized additive model are always employed in solving this kind of problems. Binary outcomes are the most common outcomes in clinical prediction models; disease diagnostic [4], prediction the death of patient [5], and medical image segmentation [6] are all binary classification problems. A lot of statistical and machine learning methods such as logistic regression, binary classification trees, and Bayesian models have been designed to solve this binary classification problem. Categorical outcomes are generated by a multi-class classification, and there is no specific ordering among those classes; in healthcare domain categorical outcomes always appears in multiple disease diagnostic such as cancer [7] and tumor [8] classification. In clinical prediction models such as polytomous logistic regression [9] and some ensemble approaches [7][10] are used to estimate the categorical outcomes. Ordinal outcomes are also quite common in clinical prediction, in some case, people grade the severity of illness [11]. Finally, survival outcomes are particularly correlated to survival analysis which aims to study the time to event data and want to predict the time to event of interest.

In section 1.2, we review some statistic predictive models. Some machine learning methods are introduced in section 1.3, and the survival models are discussed in section 1.4. We also provide some model evaluation and validation methods in section 1.5, and finally, section 1.6 concludes the whole chapter.

1.2 Basic Statistical Predictive Models

In this section, we review some of the well known basic statistical models which are widely used in biomedical and clinical fields.

1.2.1 Linear Regression

In linear regression the dependent variable or outcome is assumed to be a linear combination of the attributes with the corresponding estimated regression perimeters [12]. In clinical analysis, linear regression is often employed in clinical cost prediction [1][2] and the estimation of some medical inspection [3]. Let's consider a sample of N subjects with p attributes, which can be represented as an $N \times p$ matrix X, and the observed output is a vector $Y^T = (y_1, y_2, ..., y_N)$. For a specific individual $X_i = (x_{i1}, x_{i2}, ..., x_{ip})$ is the covariate vector, and the output is a continuous real number Y_i, and the linear regression can be mathematically expressed as:

$$\hat{y}_i = \alpha + \sum_{j=1}^{p} x_{ij}\beta_j, \tag{1.1}$$

where $\beta^T = (\beta_1, \beta_2, ..., \beta_p)$ is the coefficient vector, α is the intercept, and $\hat{y}_i$ is the estimated output based on the linear regression model. It needs to be emphasized that all the input covariate values should be numeric numbers; otherwise, the addition and multiplication computation of the covariate values do not make sense. In supervised learning, parameter estimation can be viewed as the minimization of a loss function over a training dataset. *Least squares* is the most commonly used coefficient estimation method in linear regression; the chosen loss function is the *residual sum of squares* which is defined as the squared euclidean distance between the observed output vector Y and the estimated output $\hat{Y}$. It has the form

$$\begin{aligned} RSS(\beta) &= \sum_{i=1}^{N}(y_i - \hat{y}_i)^2 \\ &= \sum_{i=1}^{N}(y_i - \alpha + \sum_{j=1}^{p} x_{ij}\beta_j)^2. \end{aligned} \tag{1.2}$$

We can see that the $RSS(\beta)$ is a quadratic equation for β, and the minimization can be calculated by letting the first derivative of the $RSS(\beta)$ equal to 0. For convenience, the $RSS(\beta)$ can be rewritten in the matrix representation

$$RSS(\beta) = (Y - X\beta)^T(Y - X\beta), \tag{1.3}$$

notice that the X here is different from the definition above; here it is an $N \times (p + 1)$ matrix where an unit column vector is added in the left of the original input matrix X, and correspondingly, the coefficient vector is $\beta^T = (\alpha, \beta_1, \beta_2, ..., \beta_p)$. The partial derivative of the $RSS(\beta)$ is

$$\frac{\partial RSS}{\partial \beta} = -2X^TY + 2(X^TX)\beta, \tag{1.4}$$

Let Equation (1.4) equal to 0, and the estimated parameter is

$$\hat{\beta} = (X^TX)^{-1}X^TY. \tag{1.5}$$

For computational efficiency, usually the input covariant matrix X is normalized in preprocessing, so $X^TX = \mathbf{1}$ and the estimated coefficient vector can be simplified as $\hat{\beta} = X^TY$.

1.2.2 Generalized Additive Model

The generalized additive model (GAM) [13] is a linear combination of smooth functions. It can be viewed as a variant of linear regression which can handle nonlinear distribution.

In GAM, for individual X_i the continuous outcome y_i can be estimated by:

$$\hat{y}_i = \alpha + f_1(x_{i1}) + f_2(x_{i2}) + \cdots + f_p(x_{ip}), \tag{1.6}$$

where $f_i(\cdot),\ i = 1, 2, ..., p$ is a set of smooth functions, and p is the number of features.

Initially, the GAM was fitted by the backfitting algorithm which was introduced in 1985 by Leo Breiman and Jerome Friedman [14]. It is an iterative method which can handle a wide variety of smooth functions; however, the stop point of the iteration is difficult to choose, and it always suffers from overfitting. An alternative method of GAM estimation is using semi-parametric smoothing function and fit the model by penalized regression splines, more detail can be found in[15].

1.2.3 Logistic Regression

Logistic Regression is considered to be a linear method for classification. It is one of the most porpular binary classification methods which is widely adopted in clinical prediction [4][16][17]. Rather than directly predicting the output via a linear combination of features, it assumes that there is a linear relationship between the features and the log-odds of the probabilities. For simplicity's sake, let's consider a two-class scenario. For a certain individual $X_i = (x_{i0}, x_{i1}, x_{i2}, ..., x_{ip})$ in a N-sampled subject, the observed output y_i can be labeled either 0 or 1; the formulation of the logistic regression is

$$log\frac{Pr(y_i = 1|X_i)}{Pr(y_i = 0|X_i)} = \sum_{k=0}^{p} x_{ik}\beta_k = X_i\beta, \tag{1.7}$$

Here, $x_{i0} = 1$ and β_0 is the intercept (in the rest part of this section without specifying the X_i and β will be defined the same as here). Consider the fact that in a two-class classification $Pr(y_i = 1|X_i) + Pr(y_i = 0|X_i) = 1$; thus, from Equation (1.7) we have

$$Pr(y_i = 1|X_i) = \frac{exp(X_i\beta)}{1 + exp(X_i\beta)}. \tag{1.8}$$

The parameter estimation in logistic regression models is usually fitted by maximizing the likelihood function. The joint conditional probability of all N examples in the training dataset is

$$Pr(y = y_1|X_1) \cdot Pr(y = y_2|X_2)\cdot, ..., \cdot Pr(y = y_N|X_N) = \prod_{i=1}^{N} Pr(y = y_i|X_i), \tag{1.9}$$

where $y_i, i = 1, 2, ..., N$ is the actual observed labels in the training set; therefore, the log-likelihood for N observations is

$$\mathfrak{L}(\beta) = \sum_{i=1}^{N} \log[Pr(y = y_i|X_i)], \tag{1.10}$$

note that in the "$(0, 1)$ scenario", the logit transformation of conditional probability for an individual X_i is

$$\log[Pr(y = y_i|X_i)] = \begin{cases} X_i\beta - \log[1 + exp(X_i\beta)] & : \quad y_i = 1 \\ -\log[1 + exp(X_i\beta)] & : \quad y_i = 0 \end{cases}, \tag{1.11}$$

thus, equation(1.10) can be rewritten as:

$$\mathfrak{L}(\beta) = \sum_{i=1}^{N} \{X_i\beta \cdot y_i - \log[1 + exp(X_i\beta)]\}. \tag{1.12}$$

Usually the Newton-Raphson algorithm is introduced to maximize this log-likelihood, where the coefficient vector is iteratively updated based on

$$\beta^{(t+1)} = \beta^{(t)} - \left[\frac{\partial^2 \mathfrak{L}(\beta)}{\partial\beta\partial\beta^T}\right]^{-1} \frac{\partial \mathfrak{L}(\beta)}{\partial\beta}, \tag{1.13}$$

where

$$\frac{\partial \mathfrak{L}(\beta)}{\partial\beta} = \sum_{i=1}^{N} X_i(y_i - \frac{exp(X_i\beta)}{1+exp(X_i\beta)}) \tag{1.14}$$

$$\frac{\partial^2 \mathfrak{L}(\beta)}{\partial\beta\partial\beta^T} = -\sum_{i=1}^{N} X_i X_i^T \frac{exp(X_i\beta)}{[1+exp(X_i\beta)]^2}. \tag{1.15}$$

The iteration always starts at $\beta = 0$. It is proven that the algorithm can guarantee the convergence towards the global optimum, but overshooting can occur.

1.2.3.1 Multi-class Logistic Regression

In multi-class logistic regression[18], conditional on one specific individual X_i, the probability that its observed output $y_i = j$ is

$$Pr(y_i = j|X_i) = \frac{exp(X_i\beta_j)}{\sum_{k\neq j} exp(X_i\beta_k)}, \tag{1.16}$$

where $j, k \in L$ and L is the label set. With this definition, the log-likelihood for N observation can be written as:

$$\mathfrak{L}(\beta) = \sum_{i=1}^{N}[(X_i\beta_j) - \log(\sum_{k\neq j} exp(X_i\beta_k))]. \tag{1.17}$$

This objective function can be minimized by the Broyden–Fletcher–Goldfarb–Shanno (BFGS) algorithm [19]. The BFGS is a kind of hill-climbing optimization technique [20], which solves the nonlinear optimization by iteratively updating the approximation to the Hessian using information gleaned from the gradient vector at each step [18].

1.2.3.2 Polytomous Logistic Regression

Polytomous logistic regression [21][22] is an extension of the basic logistic regression, which is designed to handle multi-class problems. Polytomous logistic regression is used when there is no order among the categories; in clinical analysis it has been used to deal with some complex datasets [9]. It assumes that data are case specific; in other words, each feature has a specific coefficient value for each category; in addition, it also assumes that the output cannot be perfectly estimated by the covariate for any single class. It can be viewed as a simple combination of the standard two-class logistic regression. For a C-class problem, $C-1$ binary logistic regression will be fitted; for example, if we set the last category (Cth class) as the reference category, then the model will be:

$$\begin{aligned}
\log\frac{Pr(y=1|X_i)}{Pr(y=C|X_i)} &= X_i\beta_1 \\
\log\frac{Pr(y=2|X_i)}{Pr(y=C|X_i)} &= X_i\beta_2 \\
&\vdots \\
\log\frac{Pr(y=C-1|X_i)}{Pr(y=C|X_i)} &= X_i\beta_{C-1}.
\end{aligned} \tag{1.18}$$

Note that for individual X_i the sum of all the posterior probabilities of all C categories should be 1; thus, for each possible outcome we get:

$$\begin{aligned} Pr(y=k|X_i) &= \frac{exp(X_i\beta_k)}{1+\sum_{j=1}^{C-1} exp(X_i\beta_j)},\ k=1,2,...,C-1 \\ ,Pr(y=C|X_i) &= \frac{1}{1+\sum_{j=1}^{C-1} exp(X_i\beta_j)}. \end{aligned} \tag{1.19}$$

The model can be fitted by maximum a posteriori (MAP); more detail can be found in [23].

1.2.3.3 Ordered Logistic Regression

Ordered logistic regression (or ordered logit) is an artless extension of the logistic regression that aims to solve the ordered output prediction problem. Here we will briefly introduce the two most popular logit models: proportional odds logistic regression and generalized ordered logit.

Proportional odds logistic regression

Proportional odds logistic regression [24] was founded upon the basic assumption that for different categories all differences are introduced by different intercepts, while the regression coefficients among all levels are same. In [25], proportional odds logistic regression was employed in the meta-analyses to deal with an increasing diversity of diseases and conditions. Consider a C ordered output example; for an individual X_i the proportional odds logistic regression can be represented as:

$$logit[Pr(y \le j|X_i)] = \log \frac{Pr(y \le j|X_i)}{1-Pr(y \le j|X_i)} = \alpha_j - X_i\beta, \tag{1.20}$$

where $j = 1,2,...,C$, and $\alpha_1 < \alpha_2 < \cdots < \alpha_{C-1}$. One other point to note is that the coefficient vector β here is a $P \times 1$ vector, where P is the number of features and $X_i = (x_{i1}, x_{i2}, ..., x_{iP})$. Apparently, this is a highly efficient model, and only one set of regression parameters has to be learned during the training process; however, this assumption is too specific to have a wide range of applications.

Generalized ordered logit

The generalized ordered logit (gologit)[26] can be mathematically defined as:

$$Pr(y_i > j|X_i) = \frac{exp(X_i\beta_j)}{1+exp(X_i\beta_j)} = g(X_i\beta_j),\ j=1,2,...,C-1 \tag{1.21}$$

where C is the number of ordinal categories. From the equation (1.21), the posterior probabilities that Y will take on each of the values $1,...,C$, conditional on X_i are equal to

$$Pr(y_i=j|X_i) = \begin{cases} 1-g(X_i\beta_1) & : \quad j=1 \\ g(X_i\beta_{j-1}) - g(X_i\beta_j) & : \quad j=2,...,C-1 \\ g(X_i\beta_{C-1}) & : \quad j=C \end{cases}. \tag{1.22}$$

This model can be an efficient fit by using the Stata program "gologit2" [27].

1.2.4 Bayesian Models

The Bayes theorem is one of the most important principles in probability theory and mathematical statistics; it provides a link between the *posterior probability* and the *prior*

probability, so we can see the probability changes before and after accounting for a certain random event. The formulation of the Bayes theorem is

$$Pr(Y|X) = \frac{Pr(X|Y) \cdot Pr(Y)}{Pr(X)}, \tag{1.23}$$

where $Pr(Y|X)$ is the probability of event Y, conditional on event X. Based on this theory, there are two implementations: naïve Bayes and the Bayesian network, which are commonly used in clinical prediction [28][29].

1.2.4.1 Naïve Bayes Classifier

The intuition behind the Bayesian classifiers is comparing $Pr(Y = y|X_i)$ for different $y \in Y$ where Y is the label set and choosing the most possible y_{chosen} as the estimated label for individual $X_i = (x_{i1}, x_{i2}, ..., x_{ip})$. From equation (1.23) we can see that to calculate $Pr(Y = y|X_i)$ we need to know $Pr(X_i|Y = y)$, $Pr(Y = y)$, and $Pr(X_i)$. Among these three terms, $Pr(Y = y)$ can be easily estimated from the training dataset; $Pr(X_i)$ can be ignored because when we do a comparison among different y; the denominator in the Equation (1.23) is constant; thus, the main work in Bayesian classifiers is to choose the proper method to estimate $Pr(X_i|Y = y)$.

In naïve Bayes classifier the elements in the covariant vector $(x_{i1}, x_{i2}, ..., x_{ip})$ of X_i are assumed to be conditionally independent; therefore, the $Pr(X_i|Y = y)$ can be calculated as:

$$Pr(X_i|Y = y) = \prod_{k=1}^{p} Pr(x_{ik}|Y = y), \tag{1.24}$$

where each $Pr(x_{ik}|Y = y),\ k = 1, 2, ..., p$ can be separately estimated from the given training set. Thus, to classify a test record X_i based on the Bayes theorem and ignore the $Pr(X_i)$, the conditional probability for each possible output y in the label set Y can be represented as:

$$Pr(Y = y|X_i) \propto Pr(Y = y) \prod_{k=1}^{p} Pr(x_{ik}|Y = y). \tag{1.25}$$

Then we choose the class label y_{chosen}, which maximizes the $Pr(Y = y) \prod_{k=1}^{p} Pr(x_{ik}|Y = y)$ as the output.

1.2.4.2 Bayesian Network

Although the naïve Bayes classifier is a straightforward implementation of Bayesian Classifier, in most real-word scenarios there are some relationships among the attributes. A Bayesian network introduces a *directed acyclic graph* (DAG) to represent a set of random variables by nodes and their dependence relationships by edges. Each node is associated with a probability function that gives the probability of the current node conditional on its parent nodes' probability. If the node does not have any parents, then the probability function proves the prior probability of the current node.

Based on the given information, a Bayesian network can be generated. Specifically, in decision making or prediction problems, this Bayesian network can be viewed as a hierarchical structure. Only the independent attributes that have prior probability are in the top level. For example, in Figure (1.1) there are 5 attributes that contribute to the output; among them "Attribute 3" and "Attribute 5" do not have any predecessors, so we can get the prior probabilities $Pr(3)$ and $Pr(5)$; "Attribute 1" and "Attribute 4" are in the second level, and their conditional probability are $Pr(1|3)$ and $Pr(4|3)$ separately; "Attribute 2"

is in the third level and its conditional probability is $Pr(2|1,3,4)$. "Attribute 6" and "Attribute 7" can be viewed as some observation. Specific to healthcare, "Attributes $1-5$" can be viewed as 5 kinds of pathogenic factors, "Attribute 6,7" can be viewed as two clinical measurements, and "Output" is the disease that needs to be prevented. Based on this

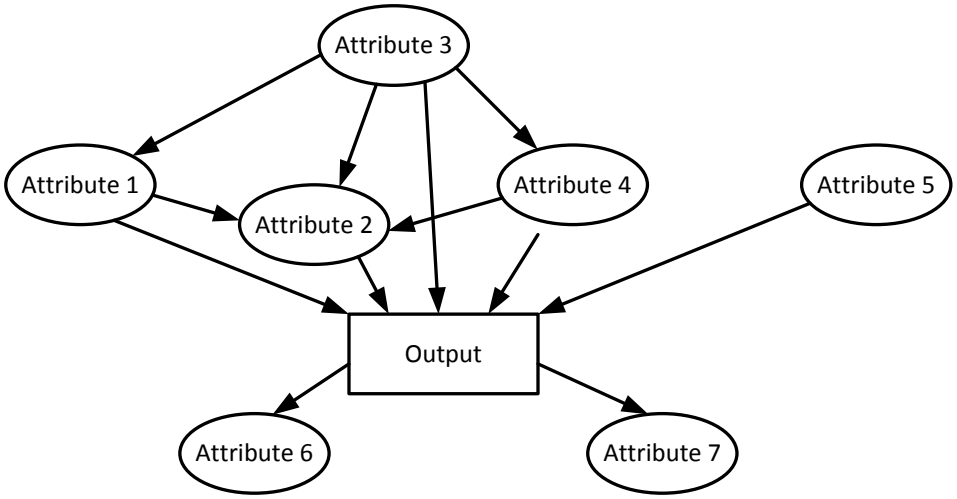

FIGURE 1.1: An example of Bayesian network for decision making

network the joint probability function is

$$\begin{aligned} Pr(\text{Output},1,2,3,4,5,6,7) &= Pr(6|\text{Output})\cdot Pr(7|\text{Output})\cdot Pr(\text{Output}|1,2,3,4,5) \\ &\quad \cdot Pr(2|1,3,4)\cdot Pr(1|3)\cdot Pr(4|3)\cdot Pr(3)\cdot Pr(5) \end{aligned} \tag{1.26}$$

and based on equation(1.26) the $Pr(\text{Output}|1,2,3,4,5)$ for each kind of output can be calculated conditional on a specific combination of 5 different attributes. Therefore, it can provide a suggestion of how to prevent the disease.

1.2.5 Markov Random Fields

In the Bayesian network the nodes are connected based on causality; however, in real-world causality is not the only relationship. For example, in clinical inspection although there is no causality between the quantity of blood leukocytes and the image of an X-ray, these two are correlated. It is awkward to represent the dataset by a directed acyclic graph in this scenario; thus, an undirected graphical model, which is also known as a *Markov random field* (MRF) or a *Markov network*, is needed. In healthcare domain markov random fields was often adopted in medical image analyses such as magnetic resonance images [30], and digital mammography [31].

Given an undirected graph $G=(V,E)$, where V is the set of vertexes and E is the set of edges; each vertex $v\in V$ represents a covariant vector X_v. In MRF the conditional independence relationship is defined via the topology of the undirected graphical. In total there are three categories of Markov properties: *global Markov property*, *local Markov property*, and *pairwise Markov property*. The global Markov property is defined as: $X_A \perp X_B|X_C$, where $A\subset V$, $B\subset V$, and $C\subset V$; that is, in the graph G subset A and B are independent conditionally on the separating subset C; in other words, every path from a node in A to a node in B passes through C. From the global Markov property we can easily deduce that

for a certain node ($X_v,\ v \in V$) all its neighbors ($X_{ne(v)},\ ne(v) \subset V$) will separate the node from the nodes in the rest part of graph G; this is called the local Markov property and can be represented as: $X_v \perp X_{rest}|X_{ne(v)}$. It is obvious that two non-adjacent nodes, X_v and X_u, are independent conditionally on all the nodes in the rest part of the graph, which is known as the pairwise Markov property, and can be mathematically represented as: $X_v \perp X_u|X_{rest}$.

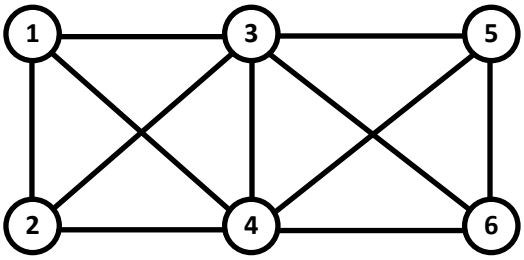

FIGURE 1.2: An example of an undirected graph

In order to describe the Markov properties more intuitively, let us illustrate these conditional independents based on Figure (1.2):

1. **Global Markov property**, $\{1,2\} \perp \{5,6\}|\{3,4\}$.
2. **Local Markov property**, $\{1\} \perp \{5,6\}|\{2,3,4\}$.
3. **Pairwise Markov property**, $\{1\} \perp \{6\}|\{2,3,4,5\}$.

1.3 Alternative Clinical Prediction Models

1.3.1 Decision Tree

A decision tree is the most widely used heuristic method in clinical prediction [32]. In a decision tree predictions are made by asking a series of well designed questions(splitting criteria) about a test record; based on the answers to these questions the test record hierarchically falls into a smaller subgroup where the contained individuals are similar to each other respecting to the predict outcome. Choosing the proper splitting criteria, obviously, is the priority for decision tree building. These criteria can help to find the locally optimum decisions which can minimize the within-node homogeneity or maximize the between-node heterogeneity in the current situation. In $C4.5$ [33] and $ID3$ [34] *information entropy* is used to determine the best splits, and multiple child nodes can be generated. In classification and regression tree (CART) [35], which can only produce binary splits, the best split is selected where the *gini* is minimized. The CHi-squared Automatic Interaction Detection (CHAID) [36] uses the statistical *Chi-square test* as its splitting criterion. Usually the tree is built by recursively choosing the best attribute to split the data to new subsets until meeting the termination criteria which are designed to prevent *overfitting*.

Compared with other methods, a decision tree is more straightforward and can represent the actual human thinking process. Different from parametric methods, such as linear regression and logistic regression, constructing a decision tree does not require knowledge of the underlying distribution. In addition, a decision tree is very convenient for handling

all kinds of data types for the input data. However, as finding an optimal decision tree is an NP-complete problem, usually a tree induction algorithm is a heuristic-based approach which makes the decision tree very unstable [37]. For the regression problem, the decision can only roughly separate the output into different categories rather than give continuous output. Furthermore, one of the most important drawbacks of a decision tree is that it can only consider the paramount attribute and simply omit the other attributes rather than take them as a whole.

1.3.2 Ensemble Methods

Ensemble methods play an important role in clinical prediction; an ensemble method is built by combining a set of base prediction models and provides a final prediction based on all these base models. It has been theoretically proved that if the base classifiers do a better job than random guessing and are independent from each other, then the ensemble method will provide a better prediction than any of those base classifiers. In terms of target manipulating, ensemble methods fall into four categories [38]: data based, feature based, label based, and learning procedure based. In the first category, the training data are resampled to build multiple datasets to achieve independence of the base models which are built based on those new training dataset separately; the well known*Bagging* [37] and *Boosting* [39] belong to this category. In the second class of approaches, such as *random forest* [40], independence is gained through using different features to train the model. The label-based ensemble only works in a multiclass scenario; in the SVM(support vector machine)-based multiclass classification one-against-all (OAA) [10], one-against-one (OAO) [41], and DB2 [42] are all label-based ensemble schemes, and the learning procedure based ensemble makes sense only to those unstable predictive models.

1.3.2.1 Bootstrap Aggregation

Bootstrap aggression, which is also know as bagging, was proposed by Breiman [37] in 1996. As mentioned above, it is a train data set-based ensemble; based on the bootstrap procedure, a number of B bootstrap samples can be repeatedly generated from the original training dataset. B base prediction models are separately learned from each bootstrap sample, and a test instance is estimated by taking a majority vote(in classification) or taking an average(in regression) among all these B base models. Bootstrap aggregation improves the performance by reducing the variance of the base models; thus, it may not be capable of improving the performance of stable base classifiers. The general procedure of bootstrap aggregation is shown in Algorithm1.

1.3.2.2 Random Forests

Random forest is an ensemble method designed specifically for tree structured predictive models. Different from bagging, in random forest each tree is built depending on a random vector sampled independently and following some constant distribution [40]. The random vector can be generated by multiracial manipulating, such as random selecting a subset of features, generating new features by a linear combination of the existing features, and random selecting one of the top ordered splits to partition the node. Similar to bootstrap aggregation the ensemble prediction is given by taking a majority vote(in classification) or taking an average(in regression) among all base decision trees. Actually bootstrap aggregation can be viewed as a particular case of random forests. Breiman has proved that in the the upper bound for the generalization error of the random forests is $\bar{\rho}(1-s^2)/s^2$, where s is the strength of the set of classifiers and $\bar{\rho}$ is the mean value of the correlation [40].

Algorithm 1 Bootstrap Aggregation

Training:

1: Let B be the number of bootstrap samples.
2: **for** $i = 1$ to B **do**
3: Generate a bootstrap sample D_i, and train a base model M_i based on D_i.
4: Store the base model M_i.
5: **end for**

Testing:

1: Let x be a test instance.
2: **for** $i = 1$ to B **do**
3: Estimate the output $M_i(x)$ based on M_i
4: **end for**
5: $M_{ensemble}(x) = \frac{\sum_{i=1}^{B} M_i(x)}{B}$(in regression scenario) or,
$M_{ensemble}(x) = \sum_{i=1}^{B} \delta(M_i(x) = y)$ (in classification scenario).
$\{\delta(\cdot) = 1$ if its argument is true and 0 otherwise$\}$.

1.3.3 Artificial Neural Networks

Inspired by biological neural systems in 1958 Frank Rosenblatt published the first paper [43] about artificial neural network(ANN), in which, simple artificial nodes, called "neurons", are combined via a weighted link to form a network which simulates a biological neural network. Each neural is a computing element which consists of sets of adaptive weights and generates the output based on a certain kind of *activation function*.

In [43], Frank Rosenblatt proposed a simple artificial neural network named *perceptron*, which only has input and output layers. For a specific input attribute vector X_i the perception model can be written as:

$$\hat{y}_i = sign(X_i W), \tag{1.27}$$

where $X_i = (x_{i0}, x_{i1}, ..., x_{ip})$ is the input attribute vector, W is the coefficient vector, and the sign function $sign(\cdot)$ is the activation function. We can see that this formulation is very similar to linear regression; however, here the model is fitted not by derivation but by an iteration approach:

$$w_j^{(t+1)} = w_j^{(t)} + \lambda(y_i - \hat{y}_i^{(t)})x_{ij}, \tag{1.28}$$

where λ is a parameter known as the *learning rate*.

General artificial neural networks are much more complex than the perceptron; they may consist of one or more intermediary layers which are known as *hidden layers* and have multiple output. In addition, diverse mapping functions, such as linear, logistic, and tanh function, can be chosen as the activation function. Therefore, a multilayer artificial neural network is capable of handling more complex nonlinear relationships between the input and output. An example of a multilayer artificial neural network is shown in Figure(1.3).

Similarly, as with other parametric supervised learning, the artificial neural network learning process is aims to minimize a cost function. In ANN learning the commonly used cost function is the *mean-squared error*, which is the average squared difference between the estimated output and the real one. Because of the complexity of finding the global minimum, the *gradient descent*, which finds a local minimum of a function, is involved in minimizing the cost function; As the hidden nodes do not influence the cost function directly, without the oupput information we can not identify its influence; thus, the common and well-known *backpropagation* technique is used for training neural networks.

In general, ANN is a self-adaptive algorithm which is able to address all kinds of complex

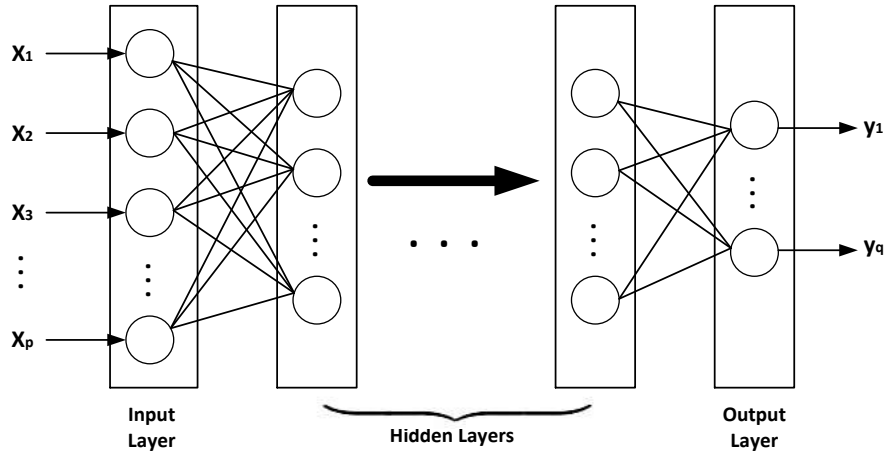

FIGURE 1.3: Example of a multilayer artificial neural network(ANN)

relationships between input and output. However, it is a time consuming and noise sensitive approach; furthermore, it is very difficult to pre-decide the network topology. Due to these drawbacks, there are many criticism of ANN [44][45], but over the years several methods have been developed to overcome these issues[46][47], and in recent years ANN has gotten some practical applications in bio-medication domain[48][49].

1.3.4 Cost-Sensitive Learning

Everything comes at a price; some clinical prediction models [50] also can be viewed as cost-sensitive models. In [51], Turney summarized 9 categories of cost in predictive models: misclassification cost [35], test cost [52], teacher cost [53], intervention cost [54], unwanted achievements cost [55], computation cost, costs of cases [56], human-computer interaction cost, and the cost of instability [57]. Without a doubt, the optimum model is the "cheapest" model, and in the predictive model the learning process aims to minimize the sum of the above 9 types of cost. Among all these different types of cost, computation cost and the cost of instability are two vital factors that need to be considered while designing all kinds of algorithms. In this section we only focus on two categories of cost: misclassification costs and test costs, which are particular to prediction science.

Misclassification cost is introduced by classification error. In the real world, the price of each error is different , and for a certain error its costs change under different circumstances. For instance, in disease diagnosis there are two possible errors: the false negative error (a patient is wrongly predicted to be healthy) and the false positive error (a healthy person is wrongly predicted to be a patient). Obviously, in this scenario, compared with the false positive error, the false negative error is an even greater mistake because this places the patient in a very dangerous situation. Some profound studies about the misclassification cost have been done in studies such as [58][59], and the outline of this work has been mathematically generalized as follows. let L be the labelset, $a \in L$ is the actual label of a certain individual, and $p \in L$ is the predicted label; for each combination of a and p there is an element c_{ap} in the cost matrix C to represent the misclassification cost. Consider a sample of N subjects for a certain individual $x_i,\ i = 1, 2, ..., N$ its actual label is $y_i = a$, and $Pr(p|x_i, a)$ is the estimated probability that x_i belongs to class p. Thus, for misclassification

cost the cost-sensitive learning aim is to minimize the following function

$$\min \sum_{i=1}^{N} \sum_{p \in L} Pr(p|x_i, a) c_{ap}. \tag{1.29}$$

In a two-class scenario, the cost matrix C is structured followers:

TABLE 1.1: Cost matrix for the two-class case

	Predict positive	Predict negative
Actual positive	c_{11}	c_{10}
Actual negative	c_{01}	c_{00}

The cost matrix can be used either during the learning approach, such as re-selecting the threshold [59] and changing the splitting criteria during the tree induction [60], or after the learning approach to evaluate the performance of the model [61] where we just need to multiply the corresponding elements from the cost matrix and confusion matrix [61] and then calculate the sum of these products.

Test cost or *the cost of obtaining the information* is incurred for obtaining attribute values. For example, in disease diagnosis, a patient already had the X-ray test but hadn't had the Nuclear Magnetic Resonance (NMR) test yet. Of course, a prediction can be made within the current information, but the NMR test will provide more information and may improve the performance of the predictive model. Thus, we have to make a trade-off between the costs and benefits of Nuclear Magnetic Resonance. This test-cost sensitive learning is kind of a feature selection to factor into the cost of each attribute; the survey paper [62] summarizes some tree induction algorithms toward minimizing the test cost during tree construction. Algorithms such as CS-ID3 [63], and CS-C4.5 [64] combine the cost of attributes with the information gain; other tree induction algorithms like Zhang et al. [65] and Zhang [66] can simultaneously minimize both test cost and missclassification cost. In addition to, tree structured models more alternative models have already been extended to deal with test-cost sensitive learning. A cost-sensitive probably approximately correct (PAC) learning framework was proposed in [67]; Chai invented a Naive Bayes based cost-sensitive [68]; in 2004 Zubek et al. introduced a cost-sensitive Markov Decision Process (MDP) [69](these three approaches also considered both misclassification and test costs). In general, today, the cost-sensitive learning framework tends to minimize various kinds of costs simultaneously rather than focus on only one category.

1.3.5 Other Models

1.3.5.1 Multiple Instance Learning

Unlike other prediction methods, in multiple instance learning [70] we do not know the exact label of each individual in the training dataset; instead, the training data are packaged into a set of labeled groups. A group is labeled positive if there is at least one positive record in it; whereas, a group is labeled negative if all the individuals in it are negative. Because of this probability, multiple instance learning is often applied in fields such as image classification, text mining, and the analysis of molecule activity. In clinical fields it is usually used to analyze radiology images; for example, in CT pulmonary angiography, multiple instance learning has been employed to detect pulmonary emboli [71].

1.3.5.2 Reinforcement Learning

In [72], Sahba et al. proposed a reinforcement learning framework for medical image segmentation. Generally, in reinforcement learning there are several basic concepts: states, actions, reward functions, policy functions, and value functions. An action is any decision an agent might need to learn how to make, and a state is any factor that the agent might take into consideration in making that decision; in addition, associated with some states and state-action pairs, the rewards function is the objective feedback from the environment. The policy function is often a stochastic function that maps the possible states to the possible actions, and the value function reflects the long-term reward. Reinforcement learning aims to maximize the long-term rewards; it is particularly well suited to problems which include a long-term versus short-term reward trade-off [73]. In medical computer-aided detection (CAD) systems reinforcement learning could be used to incorporate the knowledge gained from new patients into old models.

1.3.5.3 Sparse Methods

Sparse methods perform feature selection by inducing the coefficient vector β to be sparse, in other words, contain many zero elements. The primary motivation for using sparse methods is that in high dimensions, it is wise to proceed under the assumption that most of the attributes are not significant, and it can be used to identify the vital features in prediction [74]. Sparse methods can also be used to select a subset of features to prevent overfitting in the scenarios when $N \leq P$, where N is the number of training samples, and P is the dimension of feature space. In biomedical data analysis sparsity inducing norms are also widely used to penalize the loss function of a prediction [75].

Consider the L_p norm penalty; the smaller the p that is chosen, the sparser the solution, but when $0 \leq P < 1$, the penalty is not convex, and the solution is difficult and often impossible. Least absolute shrinkage and selection operator (Lasso) [76] is a L_1 norm penalty which can select at most $K = min(N, P)$ features while estimating the regression coefficient. In linear regression the Lasso can be defined as:

$$\hat{\beta}_{lasso} = \min_{\beta}\{RSS + \lambda \sum_{p=1}^{P} |\beta_p|\}, \tag{1.30}$$

where the RSS(residual sum of squares) is the most popular loss function in linear regression, and the Lasso penalized term is located after "+".

Elastic Net is a widely used extension of Lasso; it is a combination of the L_1 and squared L_2 norm penalty to obtain both sparsity and handle correlated feature spaces [77].

$$\hat{\beta}_{elastic\ net} = \min_{\beta}\{RSS + \lambda[\alpha \sum_{p=1}^{P} |\beta_p| + \frac{1}{2}(1-\alpha) \sum_{p=1}^{P} \beta_p^2]\} \tag{1.31}$$

where $0 \leq \alpha \leq 1$. Different from Lasso, Elastic Net can select more than N features if $N \leq P$.

Besides sparsity inducing norms, a greedy search is an alternative approach to sparse methods. Stepwise regression [12] [78] can automatically select predictive variables during the procedure; generally, it can be categorized into three frameworks: forwards selection [79], backwards elimination [12], and bidirectional elimination [80]. The forwards selection starts from the empty set of features and adds the current best feature at each selection, and the least angle regression [81] is one of the most popular forwards selections so far; however, backwards elimination starts from the full set of features and deletes the current worst feature at each step; bidirectional elimination is a combination of the above two strategies.

1.3.5.4 Kernel Methods

Combining statistical and geometric features for colonic polyp detection in CTC based on multiple kernel learning.

Kernel methods (KMs) are an effective alternative to explicit feature extraction. In [82], wang et al. proposed a colonic polyp detection framework where multiple kernels are used to extract and combine the features from different sources. Kernel functions map the attributes from the original feature space to an abstract space where the underlying distribution is much clearer; they can also be viewed as ways to measure the similarity between objects. The initial paper on kernel methods is [83], which extended the support vector machine (SVM) to address the nonlinear distribution. The kernel methods can always achieve a high performance model; however, choosing a proper kernel is a time consuming empirical process, which can only be done through many attempts. Here we briefly introduce the three most commonly used kernels: string kernel, polynomial kernel, and Gaussian kernel; more kernels can be found in [18].

In text mining, a string kernel is involved in representing the information contained in a document; it measures the similarity between two strings: the more similar two strings X and X' are, the higher the value of a string kernel $K(X, X')$ will be. In [84], the kernel function between two strings X and X' is defined as:

$$K(X, X') = \sum_{s \in A^*} < \phi_s(X) \cdot \phi_s(X') >, \tag{1.32}$$

and

$$\phi_s(X) = T_s(X) \cdot \lambda^{l(s)}. \tag{1.33}$$

In the formula, A^* is the set of all strings in $X \cup X'$, $T_s(X)$ is the number of occurrences of subsequences s in X, and $\lambda^{(s)}$ is their weight according to the length of substring s where $0 \leq \lambda \leq 1$.

A polynomial kernel is a non-stationary kernel. Polynomial kernels are well suited for problems where all the training data is normalized. The formulation of the polynomial kernel is:

$$K(X, X') = (\alpha X^T X' + c)^d, \tag{1.34}$$

where α is a constant coefficient, $c \geq 0$ is a constant trading off the influence of higher-order versus lower-order terms in the polynomial, and d is the polynomial degree.

A Gaussian kernel is an example of radial basis function (RBF) kernel [85]; the definition of a Gaussian kernel is

$$K(X, X') = exp\left(-\frac{||X - X'||^2}{2\sigma^2}\right) \tag{1.35}$$

where σ^2 is known as the *bandwith*, which plays a major role in the performance of the Gaussian kernel. Therefore, it should be carefully tuned to the problem at hand; if overestimated, the exponential will behave almost linearly and the higher-dimensional projection will start to lose its non-linear power. On the other hand, if underestimated, the function will lack regularization and the decision boundary will be highly sensitive to noise in the training data.

1.4 Survival Models

Survival analysis [86][87] studies the time to event data; the observation starts from a particular starting time and will continue until the occurrence of a certain event or observed

objects are missed. In the healthcare domain the starting point of the observation is normally a medical intervention such as a hospitalization admission, the beginning of taking a certain medication or a diagnosis of a given disease. The event might be death, discharge from the hospitalization or any interesting event that can happen during the observation. The missing trace of observation is a key characteristic of survival data; for example, during the hospitalization some patients may be changed to an other hospital. Survival analysis is useful whenever the researcher is interested not only in the frequency of a particular type of event, but also in the time process underlying such an occurrence. In the healthcare field the survival prediction models mainly aim to estimate the failure time distribution and point out the prognostic evaluation of different variables, jointly or singularly considered, such as biochemical, histological and clinical characteristics [88].

1.4.1 Basic Concepts in Survival Model

In this section, the basic concepts and characteristics of the survival model will be introduced along with some examples. In particular, all the examples come from our own Re-Hospitalization analysis which is funded by NSF. In this particular problem the survival analysis has been used to study the length of time between hospitalization and readmission of heart failure patients. Here, the event of interest is hospital readmission, and the beginning of the observation starts from the discharge from the hospitalization. By reading this section, the reader will clearly understand the difference between survival analysis and other predictive models.

1.4.1.1 Survival Data and Censoring

In survival data the event of interest may not always be observed during the study; this scenario happens because of time limits or missing traces caused by other uninteresting events. This feature is known as censoring [86].

Let us consider a small number of N heart failure patients in the Re-Hospitalization problem; suppose the observation terminates after 30 days of discharge. Thus, the time of the hospital readmission is known precisely only on those subjects who present the event before the ending point. For the remaining subjects, it is only known that the time to the event is greater than the observation time. Also during this observation time, we lose track of some patients because of death, moving out of the area or other reasons. Both of these scenarios are considered as censoring in this particular example. Figure(1.4) provides a more intuitive way to describe the idea of censoring. Formally, let T be the time to event of interest, and U be the censoring variable which is the time of the withdrawn, lost, or ended time of observation. For a certain subject if only the $Z = min(T, U)$ can be observed during the study, it is known as the *Right Censoring*; otherwise, if $Z = max(T, U)$, it is named the *Left Censoring*. Practically, in the healthcare domain the majority of the survival data is right censored [88].

In survival analysis, survival data are normally represented by a triple of variables (X, Z, δ), where X is the feature vector, and δ is an indicator. $\delta = 1$ if Z is the time to the event of interest and $\delta = 0$ if Z is the censored time; for convenience, Z is usually named the *observed time* [89]. An example of a small survival dataset, which is from our Re-Hospitalization project, is shown in Table(1.2). In this dataset, the patients' age and sex are considered to be two features, which is the X in the notation; the "status" is the indicator δ, and the "gap" is the observed time.

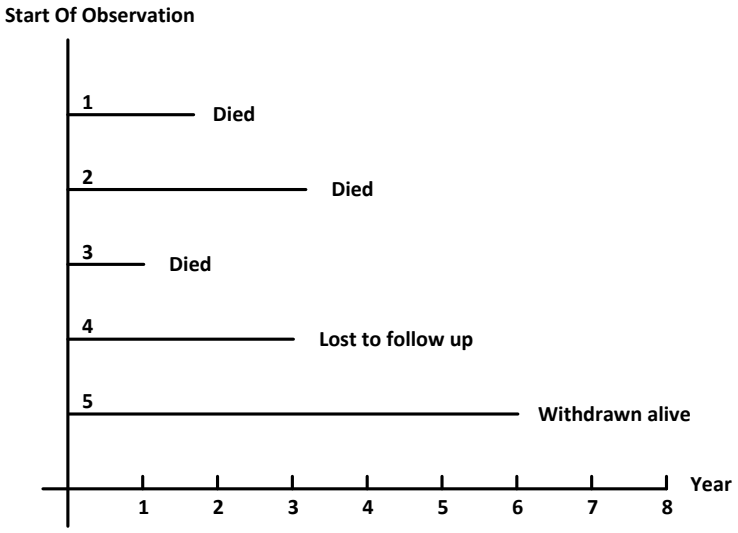

FIGURE 1.4: example of censoring

1.4.1.2 Survival and Hazard Function

The object of primary interest of survival analysis is the ***survival function***, which is the probability that the time to the event of interest is no earlier than some specified time t [89][86]. conventionally survival function is denoted as S, which is defined as:

$$S(t) = Pr(T \geq t). \tag{1.36}$$

It is certain that in the healthcare domain the survival function monotonically decreases with t, and the initial value is 1 when $t = 0$, which represents the fact that in the beginning of the observation 100% of the observed subjects survive; in other words, none of the events of interest are observed.

In contrast, the *cumulative death distribution function* $F(t)$ is defined as $F(t) = 1 - S(t)$, which represents the probability of time to the event of interest is earlier than t, and *death density function* $f(t)$ is defined as $f(t) = \frac{d}{dt}F(t)$ in continuous scenarios, and $f(t) = \frac{F(t+\Delta t)-F(t)}{\Delta t}$, where Δt is a short interval, in discrete scenarios. The relationship among these functions can be clearly described in Figure(1.5).

One other function commonly used in survival analysis is the ***hazard function*** $\lambda(t)$, which is also known as the *force of mortality*, the *conditional failure rate*, or the *instantaneous death rate* [90]. The hazard function is not the chance or probability of the event of interest, but instead it is the event rate at time t conditional on survival until time t or later . Mathematically, the hazard function is defined as:

$$\begin{aligned} \lambda(t) &= \lim_{\Delta t \to 0} \frac{Pr(t \leq T < t + \Delta t | T \geq t)}{\Delta t} \\ &= \lim_{\Delta t \to 0} \frac{F(t + \Delta t) - F(t)}{\Delta t \cdot S(t)} \\ &= \frac{f(t)}{S(t)}. \end{aligned} \tag{1.37}$$

Like $S(t)$, $\lambda(t)$ is a non-negative function. Whereas all survival functions, $S(t)$, decrease over

TABLE 1.2: Survival data on 40 Heart Failure patients.

	Features					Features			
Patient ID	Sex	Age	Gap	Status	Patient ID	Sex	Age	Gap	Status
1	F	91	29	1	21	M	77	82	1
2	M	70	57	1	22	M	69	615	1
3	F	91	6	1	23	F	79	251	0
4	M	58	1091	1	24	M	86	21	1
5	M	43	166	1	25	M	67	921	0
6	F	43	537	1	26	F	73	904	0
7	F	90	10	1	27	F	55	354	0
8	M	53	63	1	28	F	76	896	1
9	M	65	203	0	29	F	58	102	1
10	F	91	309	1	30	M	82	221	1
11	F	68	1155	1	31	F	54	1242	1
12	M	65	40	1	32	F	70	33	1
13	F	77	1046	1	33	F	38	272	0
14	F	40	12	1	34	M	57	136	1
15	F	42	48	1	35	F	55	424	1
16	F	68	86	1	36	F	59	110	1
17	F	90	126	1	37	M	74	173	1
18	M	58	1802	1	38	M	48	138	1
19	F	81	27	1	39	M	55	105	1
20	M	61	371	1	40	F	75	3	1

time, the hazard function can take on a variety of shapes. Consider the definition of $f(t)$, which can be also expressed as $f(t) = -\frac{d}{dt}S(t)$, so the hazard function can be represented as:

$$\lambda(t) = \frac{f(t)}{S(t)} = -\frac{d}{dt}S(t) \cdot \frac{1}{S(t)} = -\frac{d}{dt}[lnS(t)]. \tag{1.38}$$

Thus, the survival function can be rewritten as

$$S(t) = exp(-\Lambda(t)) \tag{1.39}$$

where $\Lambda(t) = \int_0^t \lambda(u)du$ is the *cumulative hazard function*(CHF) [89].

1.4.2 Non-Parametric Survival Analysis

Nonparametric or distribution-free methods are quite easy to understand and apply. They are less efficient than parametric methods when survival times follow a theoretical distribution and more efficient when no suitable theoretical distributions are known.

1.4.2.1 Kaplan-Meier Curve and Clinical Life Table

In this section we introduce nonparametric methods for estimating the survival probabilities for censored data. Among all functions, the survival function or its graphical presentation, the *survival curve*, is the most widely used. In 1958 Kaplan and Meier [91] developed the *product-limit estimator* or the *Kaplan-Meier Curve* to estimate the survival function based on the actual length of observed time. However, if the data have already been grouped into intervals, or the sample size is very large, or the interest is in a large population, it may be more convenient to perform a *Clinical Life Table* analysis [92].

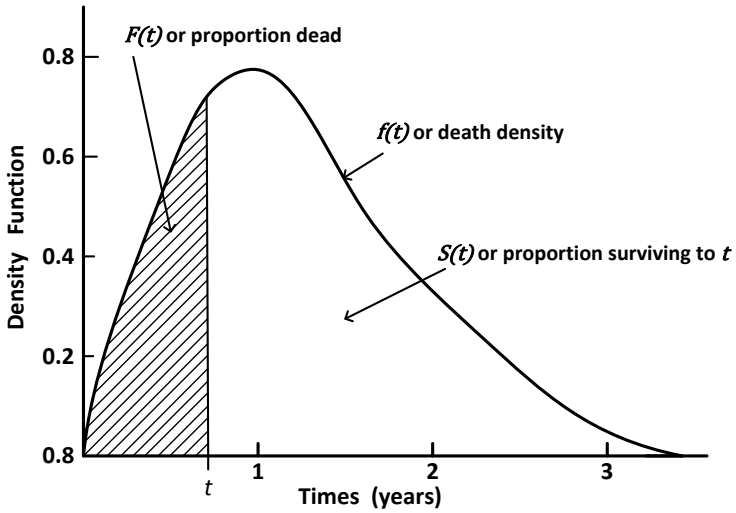

FIGURE 1.5: Relationship among $f(t)$, $F(t)$, and $S(t)$

Kaplan-Meier Curve

Let $T_1 < T_2 < ... < T_K$, $K \leq N$, is a set of distinct ordered death(failure) times observed in N individuals; in a certain time $T_j (j = 1, 2, ..., K)$, the number $d_j \geq 1$ of deaths are observed, and the number r_j of subjects are "at risk", whose either death or censored time is greater than or equal to T_j. The obvious conditional probability of surviving beyond time T_j can be defined as:

$$p(T_j) = \frac{r_j - d_j}{r_j} \tag{1.40}$$

and based on this conditional probability the survival function at t is estimated by the product,

$$\hat{S}(t) = \prod_{j:T_j<t} p(T_j) = \prod_{j:T_j<t} (1 - \frac{d_j}{r_j}) \tag{1.41}$$

and its validation is defined as:

$$Var(\hat{S}(t)) = \hat{S}(t)^2 \sum_{j:T_j<t} \frac{d_j}{r_j(r_j - d_j)} \tag{1.42}$$

It is worth noting that because of the censoring, r_j is not simply equal to the difference between r_{j-1} and d_{j-1}; the correct way to calculate r_j is $r_j = r_{j-1} - d_{j-1} - c_{j-1}$, where c_{j-1} is the number of censoring between T_{j-1} and T_j. Here, we illustrate the computation of Kaplan-Meier Curves with the example survival dataset which is shown in Table(1.2). The calculated result is shown in Table(1.3), and the corresponding K-M survival curve is shown in Figure(1.6).

Clinical Life Table

As mentioned above the Clinical Life Table [92] is the application of the product-limit methods to the interval grouped survival data. The total number of N subjects are partitioned into J intervals based on the observed time. The jth interval, normally denoted I_j,

TABLE 1.3: Kaplan-Meier estimator of 40 heart failure patients in Table1.2

						K-M Estimator								K-M Estimator	
j	T_j	δ_j	d_j	c_j	r_j	$\hat{S}(t)$	std.err	j	T_j	δ_j	d_j	c_j	r_j	$\hat{S}(t)$	std.err
1	3	1	1	0	39	0.975	0.025	21	166	1	1	0	19	0.475	0.079
2	6	1	1	0	38	0.95	0.034	22	173	1	1	0	18	0.45	0.079
3	10	1	1	0	37	0.925	0.042	23	203	0	0	1	17	.	.
4	12	1	1	0	36	0.9	0.047	24	221	1	1	0	16	0.424	0.078
5	21	1	1	0	35	0.875	0.052	25	251	0	0	1	15	.	.
6	27	1	1	0	34	0.85	0.056	26	272	0	0	1	14	.	.
7	29	1	1	0	33	0.825	0.06	27	309	1	1	0	13	0.393	0.078
8	33	1	1	0	32	0.8	0.063	28	354	0	0	1	12	.	.
9	40	1	1	0	31	0.775	0.066	29	371	1	1	0	11	0.361	0.078
10	48	1	1	0	30	0.75	0.068	30	424	1	1	0	10	0.328	0.078
11	57	1	1	0	29	0.725	0.071	31	537	1	1	0	9	0.295	0.077
12	63	1	1	0	28	0.7	0.072	32	615	1	1	0	8	0.262	0.075
13	82	1	1	0	27	0.675	0.074	33	896	1	1	0	7	0.229	0.072
14	86	1	1	0	26	0.65	0.075	34	904	0	0	1	6	.	.
15	102	1	1	0	25	0.625	0.077	35	921	0	0	1	5	.	.
16	105	1	1	0	24	0.6	0.077	36	1046	1	1	0	4	0.184	0.071
17	110	1	1	0	23	0.575	0.078	37	1091	1	1	0	3	0.138	0.066
18	126	1	1	0	22	0.55	0.079	38	1155	1	1	0	2	0.092	0.058
19	136	1	1	0	21	0.525	0.079	39	1242	1	1	0	1	0.046	0.044
20	138	1	1	0	20	0.5	0.079	40	1802	1	1	0	0	0	0

is defined as $I_j = [t_j, t_{j+1}), j = 0, 1, \cdots, J-1$, and the length of I_j is $h_j = t_{j+1} - t_j$. For I_j, let

r'_j =number of survivors at the beginning of jth interval;

c_j =number of censoring during the jth interval;

d_j =number of deaths in the jth interval;

$r_j = r'_j - c_j/2$ is assumed to be the number of survivors on average halfway through the interval.

Similarly, as the Kaplan-Meier estimator, the conditional probability of surviving during jth interval is estimated as

$$\hat{p_j} = 1 - \frac{d_j}{r_j} \tag{1.43}$$

and the corresponding survival function is estimated by the product

$$\hat{S}(I_j) = \prod_{i:i<j} (1 - \frac{d_i}{r_i}) \tag{1.44}$$

and the standard variation of this $\hat{S}(I_j)$ can be calculated in a similar way as it in Kaplan-Meier Curve. Table(1.4) illustrated the computation of Clinical Life Table withing 40 heart failure patients which are shown in the previous Table(1.2). In this example we chose the interval length as 0.5 years(183 days), and all 40 patients are partitioned into 10 intervals.

1.4.2.2 Mantel-Haenzel Test

In clinical research, one is concerned not only with estimating the survival probability but, more often, with the comparison of the life experience of two or more groups of subjects

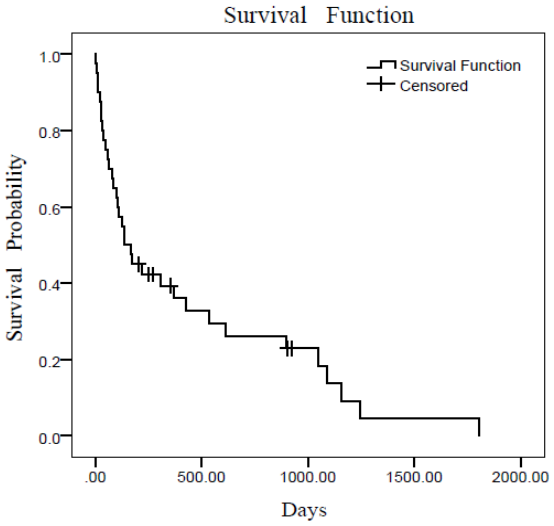

FIGURE 1.6: Kaplan-Meier survival curve of 40 heart failure patients in Table1.2

TABLE 1.4: Clinical Life Table of 40 heart failure patients.

							Estimated	
j	jth interval(days)	r'_j	c_j	r_j	d_j	$\hat{p_j}$	$\hat{S}(I_j)$	std.err
0	0 to $<$ 183	40	0	40	22	0.45	1	0
1	183 to $<$ 366	18	4	16	2	0.88	0.45	0.08
2	366 to $<$ 549	12	0	12	3	0.75	0.39	0.08
3	549 to $<$ 732	9	0	9	1	0.89	0.3	0.08
4	732 to $<$ 915	8	1	7.5	1	0.87	0.26	0.07
5	915 to $<$ 1098	6	1	5.5	2	0.64	0.23	0.07
6	1098 to $<$ 1281	3	0	3	2	0.33	0.14	0.07
7	1281 to $<$ 1464	1	0	1	0	1	0.05	0.05
8	1464 to $<$ 1647	1	0	1	0	1	0.05	0.05
9	1647 to $<$ 1830	1	0	1	1	0	0.05	0.05

differing for a given characteristic or randomly allocated to different treatments. The non-parametric approach is usually adopted also to compare survival curves. Among the various non-parametric tests one can find in the statistical literature, the Mantel-Haenzel(M-H) test [93] is one of the most frequently used statistical tools in medical reports on survival data.

Let $T_1, T_2, ...T_J$ represent the J ordered, distinct death times, and in the jth death time, the number of r_j patients are survived, and number of d_j deaths are happened. Suppose that based on some features, these patients can be divided into two groups, and at this T_j the data can be represented in a 2×2 contingency table:

Mantel and Haenzel suggested considering the distribution of the observed cell frequencies conditional on the observed marginal totals under the null hypothesis of no survival difference between these two groups. Under the null hypothesis, the d_{1j} follows hypergeometric distribution, so the expectation of d_{1j} is

$$E(d_{1j}) = r_{1j} \cdot \frac{d_j}{r_j}, \tag{1.45}$$

TABLE 1.5: Mantel-Haenzel Test in 2

Group	Number of Deaths	Number of Survival	Total
0	d_{0j}	$r_{0j} - d_{0j}$	r_{0j}
1	d_{1j}	$r_{1j} - d_{1j}$	r_{1j}
Total	d_j	$r_j - d_j$	r_j

and the variance of d_{1j} is

$$Var(d_{1j}) = \left[r_{1j} \cdot \frac{d_j}{r_j}(1 - \frac{d_j}{r_j}) \right] \frac{r_j - r_{1j}}{r_j - 1} = \frac{r_{1j} r_{0j} d_j (r_j - d_j)}{r_j^2 (r_j - 1)}. \tag{1.46}$$

The ratio is approximately distributed as chi-square with one degree of freedom [94], so for all J ordered distinct death times, the ratio is

$$X^2 = \frac{[\sum_{j=1}^{J} (d_{1j} - E(d_{1j}))]^2}{\sum_{j=1}^{J} Var(d_{1j})}. \tag{1.47}$$

Beside this Mantel-Haenzel test, there are also some non-parametric methods that have been used to compare the survival difference. In 1965 Gehan et al. [95] proposed a generalized Wilcoxon test which is an extension of the Wilcoxon test of censored data; Peto et al. [96] suggested another version of the generalized Wilcoxon test in 1972. These non-parametric methods are less efficient than parametric methods when the baseline distributions of survival times are known and more efficient when no suitable theoretical distributions are known.

1.4.3 Cox Proportional Hazard Model

The Cox proportional hazard model [97] is the most commonly used model in survival analysis. Unlike parametric methods, this model does not require knowledge of the underlying distribution, but the attributes are assumed based on an exponential influence on the output. The baseline hazard function in this model can be an arbitrary nonnegative function, but the baseline hazard functions of different individuals are assumed to be the same. The estimation and hypothesis testing of parameters in the model can be calculated by minimizing the negative partial likelihood function rather than the ordinary likelihood function.

1.4.3.1 The Basic Cox Model

Let N be the number of subjects in the survival analysis, and as mentioned in the section(1.4.1) each of the individual can be represented by a triple of variables (X, Z, δ). Consider an individual specific hazard function $\lambda(t, X_i)$ in the Cox model the proportional hazard assumption is

$$\lambda(t, X_i) = \lambda_0(t) exp(X_i \beta), \tag{1.48}$$

for $i = 1, 2, ..., N$, where the $\lambda_0(t)$ is the *baseline hazard function*, which can be an arbitrary non-negative function of time, $X_i = (x_{i1}, x_{i2}, ..., x_{ip})$ is the corresponding covariate vector for individual i, and $\beta^T = (\beta_1, \beta_2, ..., \beta_p)$ is the coefficient vector. The Cox model is a semi-parametric model since it does not specify the form of $\lambda_0(t)$; in fact, the hazard ratio does

not depend on the baseline hazard function; for two individuals the hazard ratio is

$$\frac{\lambda(t, X_1)}{\lambda(t, X_2)} = \frac{\lambda_0(t)exp(X_1\beta)}{\lambda_0(t)exp(X_2\beta)} = exp[(X_1 - X_2)\beta]. \tag{1.49}$$

Since the hazard ratio is a constant, and all the subjects share the same baseline hazard function, the Cox model is a proportional hazard model. Based on this Cox assumption the survival function is given by

$$S(t) = exp(-\Lambda_0(t)exp(X\beta)) = S_0(t)^{exp(X\beta)} \tag{1.50}$$

where $\Lambda_0(t)$ is the *cumulative baseline hazard function*, and $S_0(t) = exp(-\Lambda_0(t)$ is the baseline survival function.

1.4.3.2 Estimation of the Regression Parameter

Since in the Cox proportional hazard model the baseline hazard function $\lambda_0(t)$ is not specified, it is impossible to fit the model based on the usual likelihood function. To estimate the coefficient Cox [97] proposed a partial likelihood which represents the data only depending on the β. Consider the definition of the hazard function the probability that an individual with covariate X fails at time t conditional on survival until time t or later can be expressed by $\lambda(t, X)dt, dt \to 0$. Again, let N be the number of subjects who have a total number of $J \leq N$ events of interest occurring during the observation, and $T_1 < T_2 < ... < T_J$ is the distinct ordered time to the event of interest. Without considering the ties, let X_j be the corresponding covariate vector for the individual who fails at time T_j, and $R(T_j)$ be the set of subjects at time T_j. Thus, conditional on the fact that one individual is observed to fail at T_j, the probability that its corresponding covariate is X_j is

$$\frac{\lambda(T_j, X_j)dt}{\sum_{i \in R(T_j)} \lambda(T_j, X_i)dt}, \tag{1.51}$$

and the partial likelihood is the product of this probability; referring to the Cox assumption and the existence of the censoring, the formula definition of the partial likelihood is given by

$$L(\beta) = \prod_{j=1}^{N} \left[\frac{exp(X_j\beta)}{\sum_{i \in R_j} exp(X_i\beta)} \right]^{\delta_j}. \tag{1.52}$$

It should be noted that here $j = 1, 2, ..., N$; if $\delta_j = 1$, the jth term in the product is the conditional probability; otherwise, when $\delta_j = 0$, the corresponding term is 1 and has no effect on the result. The estimated coefficient vector $\hat{\beta}$ can be calculated by maximizing this partial likelihood; to achieve more time efficiency, it is usually equivalently estimated by minimizing the negative *log-partial likelihood*

$$LL(\beta) = \sum_{j=1}^{N} \delta_j \{X_j\beta - log[\sum_{i \in R_j} exp(X_i\beta)]\}. \tag{1.53}$$

1.4.3.3 Penalized Cox models

Currently, with the development of medical procedures and detection methods, electronic health records (EHR) tend to have more features than before. In some case, the number of features (P) is almost equivalent to or even larger than the number of subjects (N); it is unnecessary or even wrong to fit the prediction model with all the features because

of the overfitting [98]. The primary motivation of using sparsity inducing norms is that in high dimensions, it is wise to proceed under the assumption that most of the attributes are not significant, and it can be used to identify the vital features in prediction [74]. In biomedical data analysis the sparsity inducing norms are also widely used to penalize the loss function of a prediction [75]. Consider the L_p norm penalty; the smaller the p that is chosen, the sparser the solution, but when $0 \leq P < 1$, the penalty is not convex, and the solution is difficult and often impossible. Commonly, the penalized methods have also been used to do feature selection in the scenarios when $N > P$. In the following paragraph we will introduce three commonly used penalty functions and their applications in the Cox proportional hazard model.

Lasso [76] is a L_1 norm penalty which can select at most $K = min(N, P)$ features while estimating the regression coefficient. In [99], the Lasso penalty was used along with the log-partial likelihood to obtain the Cox-Lasso algorithm.

$$\hat{\beta}_{lasso} = \min_{\beta}\{-\frac{2}{N}[\sum_{j=1}^{N}\delta_j X_j\beta - \delta_j log(\sum_{i\in R_j} e^{X_i\beta})] + \lambda\sum_{p=1}^{P}|\beta_p|\} \tag{1.54}$$

Elastic Net is a combination of the L_1 and squared L_2 norm penalties to obtain both sparsity and handle correlated feature spaces [77]. As for Cox-Elastic Net [100], Noah Simon et al. implement the Elastic Net to penalize the log-partial likelihood function

$$\begin{aligned}\hat{\beta}_{elastic\ net} &= \min_{\beta}\{-\frac{2}{N}[\sum_{j=1}^{N}\delta_j X_j\beta - \delta_j log(\sum_{i\in R_j} e^{X_i\beta})] \\ &+ \lambda[\alpha\sum_{p=1}^{P}|\beta_p| + \frac{1}{2}(1-\alpha)\sum_{p=1}^{P}\beta_p^2]\}\end{aligned} \tag{1.55}$$

where $0 \leq \alpha \leq 1$. Different from Cox-Lasso, Cox-Elastic Net can select more than N features if $N \leq P$.

Ridge regression was proposed by Hoerl and Kennard [101] and introduced to Cox regression by verweij et al. [102]. It is a L_2 norm regularization tends to select all the correlated variables, and shrink their values towards each other. The regression parameters of Cox-Ridge can be estimated by

$$\hat{\beta}_{ridge} = \min_{\beta}\{-\frac{2}{N}[\sum_{j=1}^{N}\delta_j X_j\beta - \delta_j log(\sum_{i\in R_j} e^{X_i\beta})] + \frac{\lambda}{2}\sum_{p=1}^{P}\beta_p^2\} \tag{1.56}$$

Among all three equations(1.54,1.55,1.56), $\lambda \geq 0$ is used to adjust the influence introduced by the penalty. The performance of these penalized estimator depends strongly on λ, and the optimal λ_{opt} can be chosen via cross-validation.

1.4.4 Survival Trees

The intuition behind the decision tree is to recursively partition the subjects based on a specific splitting rule, and the individuals who belong to the same node are similar to each other according to the outcome of interest. The earliest attempt at using tree structure analysis for survival data can be traced back to [103].

1.4.4.1 Survival tree building methods

The selection of different in splitting criteria is the main cause of the difference between a survival tree and a common decision tree.The splitting criteria for survival trees

can be grouped into two categories: minimizing within-node homogeneity or maximizing between-node heterogeneity. The first class of approaches minimizes a loss-based within-node homogeneity criterion. Gordon and Olshen [104] measured the homogeneity by L_P, L_P Wasserstein metric, and Hellinger distances between estimated distribution functions; Davis and Anderson [105] employed an exponential log-likelihood loss function in recursive partitioning; based on the sum of residuals from the Cox model, Loh [106] proposed a partitioning criterion in 1991; Leblanc and Crowley [107] measured the node deviance based on the first step of a full likelihood estimation procedure; Cho and Hong [108] proposed an L_1 loss function to measure the within-node homogeneity. In the second category of splitting criteria, Ciampi et al. [109] employed log-rank test statistics for between-node heterogeneity measures. Then, in 1986 Ciampi et al. [110] proposed a likelihood ratio statistic(LRS) to measure the dissimilarity between two nodes. Based on the Tarone-Ware class of two-sample statistics, Segal [111] introduced a procedure to measure the between-node dissimilarity.

1.4.4.2 Ensemble methods with survival trees

To overcome the instability of a single tree, bagging and random forests, proposed by Breiman [37][40], are commonly used to do the ensemble estimation. Hothorn et al. [112] proposed a general bagging method which was implemented in the R package "ipred". In 2008 Ishwaran et al. introduced a general random forest method, called random survival forest (RSF) [113] and implemented it in the R package "randomSurvivalForest".

Bagging Survival Trees

Bagging is one of the oldest and most commonly used ensemble methods; it improves performance by reducing the variance of the base models. In survival analysis, rather than taking a majority vote, the aggregated survival function is generated by taking the average of the predictions made by each survival tree [112]. The main step of this method is:

1. Draw B booststrap samples from the original dataset.
2. Grow a survival tree for each bootstrap sample, and make sure that in each terminal node the number death is no less than d.
3. Compute the bootstrap aggregated survival function by average leaf nodes predictions.

For each leaf node the survival function is estimated by Kaplan–Meier estimator[91], and all individuals within the same node are assumed to follow the same survival function.

Random Survival Forest

Random forest is an ensemble method designed specifically for the tree structured prediction model. It is based on a framework similar to Bagging; the only difference between random forest and bagging is that in a certain node, rather than using all residual attributes, random forest only uses a random subset of the residual attributes to select attributes based on the splitting criterion. Breiman proved that randomization can be involved to reduce the correlation among trees and improve the prediction performance. In random survival forest, Nelson–Aalen estimator [114][115] are used to predict the cumulative hazard function(CHF); the formula definition of Nelson–Aalen estimator is

$$\hat{\Lambda}(t) = \sum_{t_j \leq t} \frac{d_j}{r_j}, \tag{1.57}$$

where d_j is the number of deaths at time t_j, and r_j is the number of individuals at risk at t_j. Based on this CHF, the ensemble CHF of OOB(Out Of Bag) data can be calculated by taking the average of the corresponding CHF [113].

1.5 Evaluation and Validation

1.5.1 Evaluation Metrics

When we design a new prediction model or apply an existing model to a certain dataset, we want to know whether the model is suitable for this dataset; thus, some evaluation metrics are needed to quantify the performance of the model. In this section we will introduce some well known methods which are commonly used to evaluate the performance of the clinical prediction models.

1.5.1.1 Brier Score

Named after the inventor Glenn W. Brier, the Brier score [116] is designed to evaluate prediction models which have either binary or categorical outcomes. Note that the Brier score can only evaluate the models which have probabilistic outcomes; that is, the outcome must remain in [0,1], and the sum of all the possible outcomes for a certain individual should be 1. Let's consider a sample of N subjects for each individual $X_i,\ i = 1, 2, ...N$ the predict outcome is $\hat{y}_i$, and the actual outcome is y_i; therefore, the most common definition of the Brier score can be given by

$$\text{Brier score} = \frac{1}{N}\sum_{i=1}^{N}(\hat{y}_i - y_i)^2, \tag{1.58}$$

which only suits binary outcomes where the y_i can only be 1 or 0. In more general terms, the original Brier score, defined by Brier [116], has the form:

$$\text{Brier score} = \frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{C}(\hat{y}_{ic} - y_{ic})^2, \tag{1.59}$$

for C-class output problem, where $\sum_{c=1}^{C}\hat{y}_{ic} = 1$ and $\sum_{c=1}^{C} y_{ic} = 1$. From the above two definitions of the Brier score, we know that it measures the mean squared difference between predictions and outcomes; therefore, the lower the Brier score, the better the prediction.

1.5.1.2 R^2

The R^2 or *coefficient of determination* [117] is used to measure the performance of regression prediction models, which can be formulized as:

$$R^2 = 1 - \frac{RSS(\hat{Y})}{Var(Y)}, \tag{1.60}$$

where $RSS(\hat{Y})$ is the residual sum of squares, and $Var(Y)$ is the variance of actual outcomes. For an N sampled dataset, these two terms can be mathematically defined as:

$$RSS(\hat{Y}) = \sum_{i=1}^{N}(y_i - \hat{y}_i)^2, \text{ and } Var(Y) = \sum_{i=1}^{N}(y_i - \bar{y})^2, \tag{1.61}$$

where $\bar{y}$ is the mean valve of the outcomes; in addition, for each individual X_i, y_i is the actual outcome, and $\hat{y}_i$ is the estimated outcome. Obviously, a good prediction model provides a small $RSS(\hat{Y})$; then, the closer the R^2 is to 1, the better the prediction. At the same time, we should also noted that the R^2 could be negative if the predictive model cannot represent the distribution of the dataset and even worse than the mean value [118].

1.5.1.3 Accuracy

In general, the accuracy of measurement is defined as the closeness of agreement between a quantity value obtained by measurement and the true value of the measurand [119][120]. Here we only consider its definition in the binary classification where it can be used to measure the performance of the predicted model. Consider a confusion matrix [121] for a 2-

TABLE 1.6: Confusion matrix for a 2-class problem

	Predict positive	Predict negative
Actual positive	TP	FN
Actual negative	FP	TN

class problem which is shown in Table 1.6, where the components can be separately defined as:

1. True positive (TP) is the number of positive individuals correctly predicted as positive.
2. False positive (FP) is the number of negative individuals incorrectly predicted as positive.
3. False negative (FN) is the number of positive individuals incorrectly predicted as negative.
4. True negative (TN) is the number of negative individuals correctly predicted as negative.

Based on this confusion matrix, the accuracy can be formulized as:

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + FN + TN}, \tag{1.62}$$

which is the proportion of correct predictions.

1.5.1.4 Evaluation Metrics for Class Imbalance

Class imbalance[122] is quite common in clinical prediction; for example, the World Health Organization(WHO) [123] indicated that in 2008 the Northern American incidence rate of lung cancer was 36 per 100,000 for females and 49 per 100,000 for males. Although compared with the total population very few people have been diagnosed with lung cancer, it is the most common cause of death from cancer worldwide (in 2008, about 1.38 million people died from lung cancer [123]). In this case, the accuracy measure is no longer suitable. For a lung cancer diagnosis, the model predicts no one getting lung cancer has an accuracy of almost 100%; however, it is clear that this is a bad prediction. Here we introduce some common evaluation metrics which are suitable for the class imbalance problem in 2-class scenario [61], and all the terms used in the definition of these metrics are predefined in section 1.5.1.3.

Sensitivity

Sensitivity, which is also known as the *true positive rate* (TPR) or *Recall*, measures the ratio of actual positives which are correctly identified; for example, it can be used to measure the performance of a lung cancer prediction model. The formal definition of the sensitivity is

$$\text{Sensitivity} = \frac{TP}{TP + FN}. \tag{1.63}$$

Specificity

Specificity, which is also known as the *true negative rate* (TNR), measures the ratio of actual negatives which are correctly identified [124]; this measurement can be employed in those problems where the negative individuals are more interested, and it can be defined as:

$$\text{Specificity} = \frac{TN}{TN + FP}. \tag{1.64}$$

False positive rate

The *false positive rate* (FPR) measures the ratio of actual negatives which are incorrectly identified, which is formalized as:

$$FPR = \frac{FP}{TN + FP}. \tag{1.65}$$

Precision

Precision, which is also known as the *positive predictive value* (PPV), measures the ratio of true positives to predicted positives [125]; this measurement is suitable for those problems where the positive individuals are considered more important than the negatives, and it can be mathematically represented as:

$$\text{Precision} = \frac{TP}{TP + FP}. \tag{1.66}$$

F-measure

F-measure [126] is the harmonic mean of recall and precision:

$$\text{F-measure} = \frac{2 \times Precision \times Recall}{Recall + Precision}. \tag{1.67}$$

Thus, a high value of F-measure indicates that both precision and recall are reasonably high [61]. F-measure varies in the range [0, 1] where the best value is reached at 1 and the worst score at 0.

1.5.1.5 ROC Curve

The receiver operating characteristic (ROC) curve is a graphical technique which can be used to measure and visualize the performance of a prediction model over a whole range of possible cutoffs [127]. In biomedical fields the ROC curve has been employed in the evaluation of disease diagnosis [128]. In an ROC curve, see Figure(1.7), the x axis is the false positive rate (FPR), and the y axis is the true positive rate (TPR). The cutoff varies from the highest possible value, where all subjects are predicted as negative ($TPR = 0,\ FPR = 0$), to the lowest possible value, where all subjects are predicted as positive ($TPR = 1,\ FPR = 1$), and in each possible cutoff, the TPR and FPR are calculated based on the corresponding confusion matrix.

In an ideal model, $TPR = 1$ and $FPR = 0$; that is, the area under the ROC curve (AUC)[129] is equal to 1. In [129][130], the meaning of AUC has been well discussed, and it has been proved that AUC is equal to the probability that a binary classifier will give an arbitrary positive record a higher score than an arbitrary negative record, conditional on the assumption that the positive individual should receive a higher score than the negative one. A random classifier's AUC is 0.5, and when AUC is higher than 0.5, the higher the AUC, the better the prediction model. When AUC is less than 0.5, it does not mean the prediction model is bad; however, it means the assumption is wrong, so to solve this problem, we just need to exchange the definition of positive individual and negative individual.

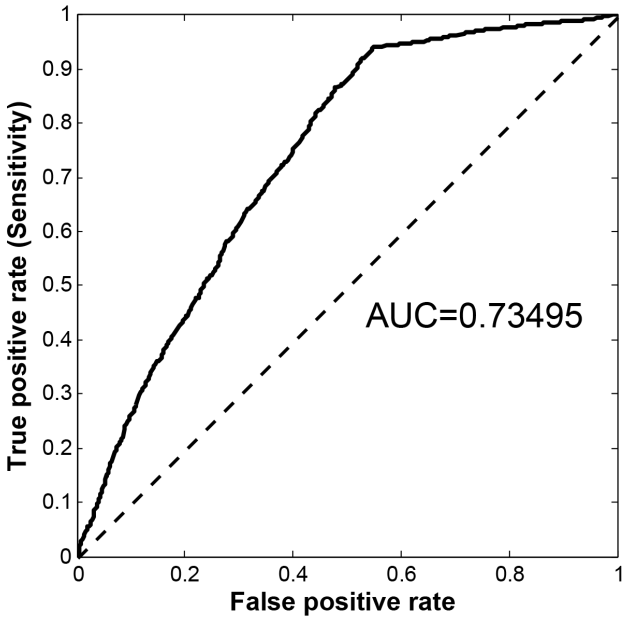

FIGURE 1.7: example of ROC curve

1.5.1.6 C-index

C-index, or the *concordance probability*, is used to measure the performance of a regression prediction model [131]. Originally, it was designed to evaluate the performance of the survival estimation [132][133]. Consider a pair of bivariate observations $(y_1, \hat{y}_1)$ and $(y_2, \hat{y}_2)$, where y_i is the actual observation, and $\hat{y}_i$ is the predicted value. The concordance probability is defined as:

$$c = Pr(\hat{y}_1 > \hat{y}_2 | y_1 \leq y_2). \tag{1.68}$$

Thus, we can see that if y_i is binary, then the c-index is the AUC. As this definition is not straightforward , in practice there are multiple ways to calculate c-index: in 1982 Harrell et al. proposed the first defination of c-index [132], Heagerty and Zheng defined the c_τ in [134], and in [135] a c-index which is specific for the Cox model was designed. Among these three methods, Harrell et al.'s c-index [132] is suitable in all cases; in contrast, in [134] and [135] c-index are designed specifically for the proportional hazard model, where $X_i\beta$ (see section "cox") is used to instead the estimated outcome $\hat{y}_i$.

1.5.2 Validation

In section 1.5.1, we reviewed several quantitative metrics for performance of the clinical prediction models, and the model can be evaluated based on its performance on the testing dataset. This section reviews some commonly used validation techniques which can provide an unbiased evaluation of a predictive model, and in general, these techniques fall into two categories: internal validation and external validation.

1.5.2.1 Internal Validation Methods

The internal validation works by random separating the training data and the testing data from the dataset where the labels of the contained individuals are already known.

Here we just briefly introduce two of the most commonly used internal validation methods: cross-validation and bootstrap validation.

Cross-Validation

The use of cross-validation can be trace back to [136]. In k-fold cross-validation, first, the labeled dataset will be randomly partitioned into k equal-sized subsets based on the uniform distribution. Then one subset is chosen as the testing dataset, while the remaining $k-1$ subsets are used to train the model. Repeat this process k times and each time use different subset as the testing dataset; therefore, each individual is used for training exactly once, and each time the training dataset is different from the testing dataset. Finally, the model will be evaluated based either on the averaged performance of k times or the combined prediction of all subjects. Using cross-validation a model can achieve a relatively high performance by fully using all the datasets, and the variance of the estimated performance metric tends to be very low because of multiple rounds. Through many contrast analyses Kohavi declared that the ten-fold cross-validation is the best choice in most situations [137].

Bootstrap Validation

In cross-validation there are no duplicate individuals in the training dataset, while in bootstrap the training records are sampled with replacement, and the number of bootstrap samples is the same as in the original samples [138]. In cross-validation, sampling is based on the uniform distribution; thus, it suppose that the data distribution of training data and testing data are same, and the variance of the estimated performance metric is introduced by insufficient sampling. However, in bootstrap validation the data distribution of training data and resting data are not same but approximate; the training samples abide the empirical distribution of the original data. It has been proved that if the number of the original samples is sufficiently large, the training dataset will contain around 63.2% of the original records, and the remaining 36.8% is called OOB (out of bag) data. In bootstrap validation B bootstrap samples are repeatedly generated based on the above strategy; for each bootstrap sample a prediction model is learned and evaluated both in the original data and in the corresponding bootstrap sample, and this approach guarantees the stability of the performance estimate of the bootstrap validation.

1.5.2.2 External Validation Methods

In clinical analysis, external validation methods are involved in to validate whether the learned model can be generalized to other scenarios and other patients [139]. For example, a clinical prediction model is learned from the previous patients, and its performance is validated by the most recently treated patients; this validation method is known as the ***temporal validation***. ***Geographic validation*** is another commonly used external validation technique; in this method the training data and testing data are separated not based on the random sampling but the geographic. Once the prediction model has been learned from a local hospital, people are always willing to see whether it can be viewed as a generalized model; thus, the geographic validation is needed. In general, conditional on the same performance, the larger the difference between the training and testing dataset, the more general model we can get.

1.6 Conclusion

This chapter primarily focused on reviewing some basic supervised learning methods which have been involved in clinical prediction. Commonly, linear regression is used to estimate a continuous outcome; logistic regression is a linear binary classification method; decision trees are more suitable for categorical inputs and outcomes; survival models are specifically designed for survival analysis. In addition, we also provide some state-of-the art extensions for some of these basic models; the sparse methods can deal with high-dimensional problems; within kernels nonlinear distribution can be translated into linear distribution; and ensemble approaches can somewhat improve the performance of the base models. As space is limited, many of these methods are not fully detailed, and some other interesting branches such as longitudinal prediction models and latent linear models are not included in this chapter. In general, this chapter is a brief review of clinical predictive models; by reading this chapter, we hope readers can have a general understanding of different models, and we also provide references for readers if they have particular interest in a certain model. We look forward to receiving your criticism upon this chapter for its oversight errors due to limited ideological level.

Bibliography

[1] Edwin Rietveld, Hendrik C.C. de Jonge, Johan J. Polder, Yvonne Vergouwe, Henk J. Veeze, Henriëtte A. Moll, and Ewout W. Steyerberg. Anticipated costs of hospitalization for respiratory syncytial virus infection in young children at risk. *The Pediatric infectious disease journal*, 23(6):523–529, 2004.

[2] Michael A. Cucciare and William O'Donohue. Predicting future healthcare costs: how well does risk-adjustment work? *Journal of health organization and management*, 20(2):150–162, 2006.

[3] P. Krijnen, B.C. Van Jaarsveld, MGM Hunink, and JDF Habbema. The effect of treatment on health-related quality of life in patients with hypertension and renal artery stenosis. *Journal of human hypertension*, 19(6):467–470, 2005.

[4] Daryl Pregibon. Logistic regression diagnostics. *The Annals of Statistics*, pages 705–724, 1981.

[5] Taya V. Glotzer, Anne S. Hellkamp, John Zimmerman, Michael O. Sweeney, Raymond Yee, Roger Marinchak, James Cook, Alexander Paraschos, John Love, Glauco Radoslovich, et al. Atrial high rate episodes detected by pacemaker diagnostics predict death and stroke report of the atrial diagnostics ancillary study of the mode selection trial (most). *Circulation*, 107(12):1614–1619, 2003.

[6] Shijun Wang and Ronald M. Summers. Machine learning and radiology. *Medical image analysis*, 16(5):933–951, 2012.

[7] Li M. Fu and Casey S. Fu-Liu. Multi-class cancer subtype classification based on gene expression signatures with reliability analysis. *FEBS letters*, 561(1):186–190, 2004.

[8] Yongxi Tan, Leming Shi, Weida Tong, G.T. Gene Hwang, and Charles Wang. Multi-class tumor classification by discriminant partial least squares using microarray gene expression data and assessment of classification models. *Computational Biology and Chemistry*, 28(3):235–243, 2004.

[9] Anila Wijesinha, Colin B. Begg, H. Harris Funkenstein, and Barbara J. McNeil. Methodology for the differential diagnosis of a complex data set. a case study using data from routine ct scan examinations. *Medical decision making: an international journal of the Society for Medical Decision Making*, 3(2):133–154, 1982.

[10] Vladimir Vapnik. *The nature of statistical learning theory*. springer, 2000.

[11] Frank E. Harrell, Peter A. Margolis, Sandy Gove, Karen E. Mason, E. Kim Mulholland, Deborah Lehmann, Lulu Muhe, Salvacion Gatchalian, and Heinz F. Eichenwald. Development of a clinical prediction model for an ordinal outcome: the world health organization multicentre study of clinical signs and etiological agents of pneumonia, sepsis and meningitis in young infants. *Statistics in medicine*, 17(8):909–944, 1998.

[12] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *Linear Methods for Regression*. Springer, 2009.

[13] Trevor Hastie and Robert Tibshirani. Generalized additive models. *Statistical science*, pages 297–310, 1986.

[14] Leo Breiman and Jerome H. Friedman. Estimating optimal transformations for multiple regression and correlation. *Journal of the American Statistical Association*, 80(391):580–598, 1985.

[15] Simon Wood. *Generalized additive models: an introduction with R*. CRC Press, 2006.

[16] A. Cecile J.W. Janssens, Yazhong Deng, Gerard J.J.M. Borsboom, Marinus J.C. Eijkemans, J. Dik F. Habbema, and Ewout W. Steyerberg. A new logistic regression approach for the evaluation of diagnostic test results. *Medical decision making*, 25(2):168–177, 2005.

[17] D.M. Wingerchuk, V.A. Lennon, S.J. Pittock, C.F. Lucchinetti, and B.G. Weinshenker. Revised diagnostic criteria for neuromyelitis optica. *Neurology*, 66(10):1485–1489, 2006.

[18] Kevin P. Murphy. *Machine learning: a probabilistic perspective*. The MIT Press, 2012.

[19] Dimitri P. Bertsekas. Nonlinear programming. 1999.

[20] Stephen M. Goldfeld, Richard E. Quandt, and Hale F. Trotter. Maximization by quadratic hill-climbing. *Econometrica: Journal of the Econometric Society*, pages 541–551, 1966.

[21] J. Engel. Polytomous logistic regression. *Statistica neerlandica*, 42(4):233–252, 1988.

[22] Balaji Krishnapuram, Lawrence Carin, Mario A. T. Figueiredo, and Alexander J. Hartemink. Sparse multinomial logistic regression: Fast algorithms and generalization bounds. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 27(6):957–968, 2005.

[23] Morris H. DeGroot. *Optimal statistical decisions*, volume 82. Wiley-Interscience, 2005.

[24] Rollin Brant. Assessing proportionality in the proportional odds model for ordinal logistic regression. *Biometrics*, pages 1171–1178, 1990.

[25] Anne Whitehead, Rumana Z. Omar, Julian Higgins, Elly Savaluny, Rebecca M. Turner, and Simon G. Thompson. Meta-analysis of ordinal outcomes using individual patient data. *Statistics in medicine*, 20(15):2243–2260, 2001.

[26] Richard Williams. Generalized ordered logit/partial proportional odds models for ordinal dependent variables. *Stata Journal*, 6(1):58–82, 2006.

[27] Richard Williams. Gologit2: Stata module to estimate generalized logistic regression models for ordinal dependent variables. *Statistical Software Components*, 2013.

[28] Igor Kononenko. Inductive and bayesian learning in medical diagnosis. *Applied Artificial Intelligence an International Journal*, 7(4):317–337, 1993.

[29] Margaret Sullivan Pepe. The statistical evaluation of medical tests for classification and prediction. 2003.

[30] Karsten Held, E. Rota Kops, Bernd J. Krause, William M. Wells III, Ron Kikinis, and H.W. Muller-Gartner. Markov random field segmentation of brain mr images. *Medical Imaging, IEEE Transactions on*, 16(6):878–886, 1997.

[31] H.D. Li, M. Kallergi, L.P. Clarke, V.K. Jain, and R.A. Clark. Markov random field for tumor detection in digital mammography. *Medical Imaging, IEEE Transactions on*, 14(3):565–576, 1995.

[32] Mary Jo Aspinall. Use of a decision tree to improve accuracy of diagnosis. *Nursing Research*, 28(3):182–185, 1979.

[33] John Ross Quinlan. *C4. 5: programs for machine learning*, volume 1. Morgan kaufmann, 1993.

[34] J. Ross Quinlan et al. *Discovering rules by induction from large collections of examples*. Expert systems in the micro electronic age. Edinburgh University Press, 1979.

[35] L. Breiman J. H. Friedman R. A. Olshen and Charles J. Stone. Classification and regression trees. *Wadsworth International Group*, 1984.

[36] Gordon V. Kass. An exploratory technique for investigating large quantities of categorical data. *Applied statistics*, pages 119–127, 1980.

[37] Leo Breiman. Bagging predictors. *Machine learning*, 24(2):123–140, 1996.

[38] Thomas G. Dietterich. Ensemble methods in machine learning. In *Multiple classifier systems*, pages 1–15. Springer, 2000.

[39] Yoav Freund and Robert E. Schapire. A desicion-theoretic generalization of on-line learning and an application to boosting. In *Computational learning theory*, pages 23–37. Springer, 1995.

[40] Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[41] Ulrich H. G. Kreßel. Pairwise classification and support vector machines. In *Advances in kernel methods*, pages 255–268. MIT Press, 1999.

[42] Volkan Vural and Jennifer G. Dy. A hierarchical method for multi-class support vector machines. In *Proceedings of the twenty-first international conference on Machine learning*, page 105. ACM, 2004.

[43] Frank Rosenblatt. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, 65(6):386, 1958.

[44] Alexander Keewatin Dewdney and John Richardson. *Yes, we have no neutrons: An eye-opening tour through the twists and turns of bad science*. Wiley New York, NY, USA, 1997.

[45] Jack V. Tu. Advantages and disadvantages of using artificial neural networks versus logistic regression for predicting medical outcomes. *Journal of clinical epidemiology*, 49(11):1225–1231, 1996.

[46] Alex Graves and Juergen Schmidhuber. Offline handwriting recognition with multidimensional recurrent neural networks. In *Advances in Neural Information Processing Systems*, pages 545–552, 2008.

[47] Dan Cireşan, Ueli Meier, Jonathan Masci, and Jürgen Schmidhuber. Multi-column deep neural network for traffic sign classification. *Neural Networks*, 32:333–338, 2012.

[48] Leonardo Bottaci, Philip J. Drew, John E. Hartley, Matthew B. Hadfield, Ridzuan Farouk, Peter W. R. Lee, Iain Macintyre, Graeme S. Duthie, and John R. T. Monson. Artificial neural networks applied to outcome prediction for colorectal cancer patients in separate institutions. *The Lancet*, 350(9076):469–472, 1997.

[49] N. Ganesan, K. Venkatesh, M. A. Rama, and A Malathi Palani. Application of neural networks in diagnosing cancer disease using demographic data. *International Journal of Computer Applications. http://www. ijcaonline. org/journal/number26/pxc387783. p df*, 2010.

[50] Laurent G. Glance, Turner Osler, and Tamotsu Shinozaki. Intensive care unit prognostic scoring systems to predict death: a cost-effectiveness analysis. *Critical care medicine*, 26(11):1842–1849, 1998.

[51] Peter Turney. Types of cost in inductive concept learning. 2000.

[52] Peter Turney. Cost-sensitive classification: Empirical evaluation of a hybrid genetic decision tree induction algorithm. *Journal of Artificial Intelligence Research (JAIR)*, 2, 1995.

[53] David A. Cohn, Zoubin Ghahramani, and Michael I. Jordan. Active learning with statistical models. *arXiv preprint cs/9603104*, 1996.

[54] Floor Verdenius. A method for inductive cost optimization. In *Machine Learning—EWSL-91*, pages 179–191. Springer, 1991.

[55] Maarten van Someren, Cristina Torres, and Floor Verdenius. A systematic description of greedy optimisation algorithms for cost sensitive generalisation. In *Advances in Intelligent Data Analysis Reasoning about Data*, pages 247–257. Springer, 1997.

[56] Tom Fawcett and Foster Provost. Activity monitoring: Noticing interesting changes in behavior. In *Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 53–62. ACM, 1999.

[57] Peter Turney. Technical note: Bias and the quantification of stability. *Journal of Machine Learning*, 20, 1995.

[58] Pedro Domingos. Metacost: a general method for making classifiers cost-sensitive. In *Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 155–164. ACM, 1999.

[59] Charles Elkan. The foundations of cost-sensitive learning. In *International joint conference on artificial intelligence*, volume 17, pages 973–978. Citeseer, 2001.

[60] Chris Drummond and Robert C. Holte. Exploiting the cost (in) sensitivity of decision tree splitting criteria. In *ICML*, pages 239–246, 2000.

[61] Pang-Ning Tan et al. *Introduction to data mining.* Pearson Education India, 2007.

[62] Susan Lomax and Sunil Vadera. A survey of cost-sensitive decision tree induction algorithms. *ACM Computing Surveys (CSUR)*, 45(2):16, 2013.

[63] Ming Tan and Jeffrey C. Schlimmer. Cost-sensitive concept learning of sensor use in approach and recognition. In *Proceedings of the sixth international workshop on Machine learning*, pages 392–395. Morgan Kaufmann Publishers Inc., 1989.

[64] Alberto Freitas, Altamiro Costa-Pereira, and Pavel Brazdil. Cost-sensitive decision trees applied to medical data. In *Data Warehousing and Knowledge Discovery*, pages 303–312. Springer, 2007.

[65] Shichao Zhang, Xiaofeng Zhu, Jilian Zhang, and Chengqi Zhang. Cost-time sensitive decision tree with missing values. In *Knowledge Science, Engineering and Management*, pages 447–459. Springer, 2007.

[66] Shichao Zhang. Cost-sensitive classification with respect to waiting cost. *Knowledge-Based Systems*, 23(5):369–378, 2010.

[67] Russell Greiner, Adam J. Grove, and Dan Roth. Learning cost-sensitive active classifiers. *Artificial Intelligence*, 139(2):137–174, 2002.

[68] Xiaoyong Chai, Lin Deng, Qiang Yang, and Charles X. Ling. Test-cost sensitive naive bayes classification. In *Data Mining, 2004. ICDM'04. Fourth IEEE International Conference on*, pages 51–58. IEEE, 2004.

[69] Valentina Bayer Zubek, Thomas Glen Dietterich, et al. Pruning improves heuristic search for cost-sensitive learning. Technical report, Corvallis, OR: Oregon State University, Dept. of Computer Science, 2004.

[70] Thomas G. Dietterich, Richard H. Lathrop, and Tomás Lozano-Pérez. Solving the multiple instance problem with axis-parallel rectangles. *Artificial Intelligence*, 89(1):31–71, 1997.

[71] Jianming Liang and Jinbo Bi. Computer aided detection of pulmonary embolism with tobogganing and mutiple instance classification in ct pulmonary angiography. In *Information Processing in Medical Imaging*, pages 630–641. Springer, 2007.

[72] Farhang Sahba, Hamid R. Tizhoosh, and Magdy M. A. Salama. A reinforcement learning framework for medical image segmentation. In *Neural Networks, 2006. IJCNN'06. International Joint Conference on*, pages 511–517. IEEE, 2006.

[73] Richard S. Sutton and Andrew G. Barto. *Reinforcement learning: An introduction*, volume 1. Cambridge Univ Press, 1998.

[74] Trevor. Hastie, Robert. Tibshirani, and J. Jerome H. Friedman. *The elements of statistical learning*, volume 1. Springer New York, 2001.

[75] Jieping Ye and Jun Liu. Sparse methods for biomedical data. *ACM SIGKDD Explorations Newsletter*, 14(1):4–15, 2012.

[76] Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288, 1996.

[77] Hui Zou and Trevor Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(2):301–320, 2005.

[78] Ronald R. Hocking. A biometrics invited paper. the analysis and selection of variables in linear regression. *Biometrics*, 32(1):1–49, 1976.

[79] Robert B. Bendel and Abdelmonem A. Afifi. Comparison of stopping rules in forward "stepwise" regression. *Journal of the American Statistical Association*, 72(357):46–53, 1977.

[80] Tong Zhang. Adaptive forward-backward greedy algorithm for sparse learning with linear models. In *Advances in Neural Information Processing Systems*, pages 1921–1928, 2008.

[81] Bradley Efron, Trevor Hastie, Iain Johnstone, and Robert Tibshirani. Least angle regression. *The Annals of statistics*, 32(2):407–499, 2004.

[82] Shijun Wang, Jianhua Yao, Nicholas Petrick, and Ronald M. Summers. Combining statistical and geometric features for colonic polyp detection in ctc based on multiple kernel learning. *International journal of computational intelligence and applications*, 9(01):1–15, 2010.

[83] Bernhard E. Boser, Isabelle M. Guyon, and Vladimir N. Vapnik. A training algorithm for optimal margin classifiers. In *Proceedings of the fifth annual workshop on Computational learning theory*, pages 144–152. ACM, 1992.

[84] Huma Lodhi, Craig Saunders, John Shawe-Taylor, Nello Cristianini, and Chris Watkins. Text classification using string kernels. *The Journal of Machine Learning Research*, 2:419–444, 2002.

[85] Bernhard Scholkopf, Kah-Kay Sung, Christopher J. C. Burges, Federico Girosi, Partha Niyogi, Tomaso Poggio, and Vladimir Vapnik. Comparing support vector machines with gaussian kernels to radial basis function classifiers. *Signal Processing, IEEE Transactions on*, 45(11):2758–2765, 1997.

[86] John P. Klein and Mei-Jie Zhang. *Survival analysis, software*. Wiley Online Library, 2005.

[87] Rupert G. Miller Jr. *Survival analysis*, volume 66. John Wiley & Sons, 2011.

[88] Ettore Marubini and Maria Grazia Valsecchi. Analysing survival data from clinical trials and observational studies; e. marubini & mg valsecchi published by john wiley & sons 414 pages isbn 0–971-93987-0. *British Journal of Clinical Pharmacology*, 41(1):76–76, 1996.

[89] Elisa T Lee and John Wang. *Statistical methods for survival data analysis*, volume 476. Wiley. com, 2003.

[90] Olive Jean Dunn and Virginia A. Clark. *Basic statistics: a primer for the biomedical sciences*. Wiley. com, 2009.

[91] Edward L. Kaplan and Paul Meier. Nonparametric estimation from incomplete observations. *Journal of the American statistical association*, 53(282):457–481, 1958.

[92] Sidney J. Cutler and Fred Ederer. Maximum utilization of the life table method in analyzing survival. *Journal of chronic diseases*, 8(6):699–712, 1958.

[93] N. Mantel and W. Haenszel. Statistical aspects of the analysis of data from retrospective studies of disease. *Journal of the National Cancer Institute*, 22(4):719, 1959.

[94] Nathan Mantel. Chi-square tests with one degree of freedom; extensions of the mantel-haenszel procedure. *Journal of the American Statistical Association*, 58(303):690–700, 1963.

[95] Edmund A. Gehan. A generalized wilcoxon test for comparing arbitrarily singly-censored samples. *Biometrika*, 52(1-2):203–223, 1965.

[96] Richard Peto and Julian Peto. Asymptotically efficient rank invariant test procedures. *Journal of the Royal Statistical Society. Series A (General)*, pages 185–207, 1972.

[97] David R. Cox. Regression models and life-tables. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 187–220, 1972.

[98] Hans van Houwelingen and Hein Putter. *Dynamic Prediction in Clinical Survival Analysis*. CRC Press, Inc., 2011.

[99] Robert Tibshirani et al. The lasso method for variable selection in the cox model. *Statistics in medicine*, 16(4):385–395, 1997.

[100] Noah Simon, Jerome Friedman, Trevor Hastie, and Rob Tibshirani. Regularization paths for cox's proportional hazards model via coordinate descent. *Journal of Statistical Software*, 39(5):1–13, 2011.

[101] Arthur E. Hoerl and Robert W. Kennard. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67, 1970.

[102] Pierre J. M. Verweij and Hans C. Van Houwelingen. Penalized likelihood in cox regression. *Statistics in Medicine*, 13(23-24):2427–2436, 1994.

[103] A. Ciampi, R. S. Bush, M. Gospodarowicz, and J. E. Till. An approach to classifying prognostic factors related to survival experience for non-hodgkin's lymphoma patients: Based on a series of 982 patients: 1967–1975. *Cancer*, 47(3):621–627, 1981.

[104] L. Gordon and R. A. Olshen. Tree-structured survival analysis. *Cancer treatment reports*, 69(10):1065, 1985.

[105] Roger B. Davis and James R. Anderson. Exponential survival trees. *Statistics in Medicine*, 8(8):947–961, 1989.

[106] Wei-Yin Loh. Survival modeling through recursive stratification. *Computational statistics & data analysis*, 12(3):295–313, 1991.

[107] Michael LeBlanc and John Crowley. Relative risk trees for censored survival data. *Biometrics*, pages 411–425, 1992.

[108] Hyung Jun Cho and Seung-Mo Hong. Median regression tree for analysis of censored survival data. *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on*, 38(3):715–726, 2008.

[109] Antonio Ciampi, Johanne Thiffault, Jean-Pierre Nakache, and Bernard Asselain. Stratification by stepwise regression, correspondence analysis and recursive partition: a comparison of three methods of analysis for survival data with covariates. *Computational statistics & data analysis*, 4(3):185–204, 1986.

[110] A. Ciampi, C. H. Chang, S. Hogg, and S. McKinney. Recursive partition: A versatile method for exploratory-data analysis in biostatistics. In *Biostatistics*, pages 23–50. Springer, 1986.

[111] Mark Robert Segal. Regression trees for censored data. *Biometrics*, pages 35–47, 1988.

[112] Torsten Hothorn, Berthold Lausen, Axel Benner, and Martin Radespiel-Tröger. Bagging survival trees. *Statistics in medicine*, 23(1):77–91, 2004.

[113] Hemant Ishwaran, Udaya B. Kogalur, Eugene H. Blackstone, and Michael S. Lauer. Random survival forests. *The Annals of Applied Statistics*, pages 841–860, 2008.

[114] Wayne Nelson. Theory and applications of hazard plotting for censored failure data. *Technometrics*, 14(4):945–966, 1972.

[115] Odd Aalen. Nonparametric inference for a family of counting processes. *The Annals of Statistics*, 6(4):701–726, 1978.

[116] Glenn W. Brier. Verification of forecasts expressed in terms of probability. *Monthly weather review*, 78(1):1–3, 1950.

[117] R. G. D. Steel and J. H. Torrie. I960. principles and procedures of statistics with special reference to the biological sciences.

[118] A. Colin Cameron and Frank A. G. Windmeijer. An R-squared measure of goodness of fit for some common nonlinear regression models. *Journal of Econometrics*, 77(2):329–342, 1997.

[119] ISO 3534–1:2006. *Statistics – vocabulary and symbols – Part 1: General statistical terms and terms used in probability.* Geneva, Switzerland: ISO, 2006.

[120] Antonio Menditto, Marina Patriarca, and Bertil Magnusson. Understanding the meaning of accuracy, trueness and precision. *Accreditation and quality assurance*, 12(1):45–47, 2007.

[121] Stephen V. Stehman. Selecting and interpreting measures of thematic classification accuracy. *Remote sensing of Environment*, 62(1):77–89, 1997.

[122] Nathalie Japkowicz and Shaju Stephen. The class imbalance problem: A systematic study. *Intelligent data analysis*, 6(5):429–449, 2002.

[123] J. Ferlay, H. R. Shin, F. Bray, D. Forman, C. Mathers, and D. M. Parkin. Globocan 2008 v1. 2, cancer incidence and mortality worldwide: Iarc cancerbase no. 10 [internet]. international agency for research on cancer, lyon, france, 2011.

[124] Douglas G. Altman and J. Martin Bland. Diagnostic tests. 1: Sensitivity and specificity. *BMJ: British Medical Journal*, 308(6943):1552, 1994.

[125] Robert H. Fletcher, Suzanne W. Fletcher, Grant S. Fletcher, et al. *Clinical epidemiology: the essentials.* Lippincott Williams & Wilkins, 2012.

[126] Gerard Salton and Michael J. McGill. Introduction to modern information retrieval. 1986.

[127] Tom Fawcett. An introduction to roc analysis. *Pattern recognition letters*, 27(8):861–874, 2006.

[128] Mark H. Zweig and Gregory Campbell. Receiver-operating characteristic (roc) plots: a fundamental evaluation tool in clinical medicine. *Clinical chemistry*, 39(4):561–577, 1993.

[129] J. A. Hanely and B. J. McNeil. The meaning and use of the area under a receiver operating characteristic (roc) curve. *Radiology*, 143(1):29–36, 1982.

[130] Donald Bamber. The area above the ordinal dominance graph and the area below the receiver operating characteristic graph. *Journal of mathematical psychology*, 12(4):387–415, 1975.

[131] Frank E. Harrell, Kerry L. Lee, Robert M. Califf, David B. Pryor, and Robert A. Rosati. Regression modelling strategies for improved prognostic prediction. *Statistics in medicine*, 3(2):143–152, 1984.

[132] Frank E. Harrell Jr., Robert M. Califf, David B. Pryor, Kerry L. Lee, and Robert A. Rosati. Evaluating the yield of medical tests. *JAMA: the journal of the American Medical Association*, 247(18):2543–2546, 1982.

[133] Michael J. Pencina and Ralph B. D'Agostino. Overall c as a measure of discrimination in survival analysis: model specific population value and confidence interval estimation. *Statistics in medicine*, 23(13):2109–2123, 2004.

[134] Patrick J. Heagerty and Yingye Zheng. Survival model predictive accuracy and roc curves. *Biometrics*, 61(1):92–105, 2005.

[135] Mithat Gönen and Glenn Heller. Concordance probability and discriminatory power in proportional hazards regression. *Biometrika*, 92(4):965–970, 2005.

[136] Mervyn Stone. Cross-validatory choice and assessment of statistical predictions. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 111–147, 1974.

[137] Ron Kohavi et al. A study of cross-validation and bootstrap for accuracy estimation and model selection. In *IJCAI*, volume 14, pages 1137–1145, 1995.

[138] Bradley Efron and Robert Tibshirani. *An introduction to the bootstrap*, volume 57. CRC press, 1993.

[139] Inke R. König, J.D. Malley, C. Weimar, H.C. Diener, and A. Ziegler. Practical experiences on the necessity of external validation. *Statistics in medicine*, 26(30):5499–5511, 2007.