Chapter 10

A Review of Clinical Prediction Models

Chandan K. Reddy
Department of Computer Science
Wayne State University
Detroit, MI
reddy@cs.wayne.edu

Yan Li
Department of Computer Science
Wayne State University
Detroit, MI
rock_liyan@wayne.edu

10.1 Introduction

Clinical prediction is one of the most important branches of healthcare data analytics. In this chapter, we will provide a relatively comprehensive review of the supervised learning methods that have been employed successfully for clinical prediction tasks. Some of these methods such as linear regression, logistic regression, and Bayesian models are basic and widely investigated in the statistics literature. More sophisticated methods in machine learning and data mining literature such as decision trees and artificial neural networks have also been successfully used in clinical applications. In addition, survival models in statistics that try to predict the time of occurrence of a particular event of interest have also been widely used in clinical data analysis.

Generally, supervised learning methods can be broadly classified into two categories: classification and regression. Both of these two classes of techniques focus on discovering the underlying relationship between covariate variables, which are also known as attributes and features, and a dependent variable (outcome). The main difference between these two approaches is that a classification model generates class labels while a regression model predicts real-valued outcomes. The choice of the model to be used for a particular application significantly depends on the outcomes to be predicted. These outcomes can fall into one of the five different categories: continuous outcomes, binary outcomes, categorical outcomes, ordinal outcomes, and survival outcomes.

The continuous outcomes can be seen in applications such as medical costs prediction [1, 2] and the estimation of some medical inspection [3]; linear regression and generalized additive models have been successfully employed for solving these kinds of problems. Binary outcomes are the most common outcomes in clinical prediction models; disease diagnostic [4], prediction of the patient's death or risk [5], and medical image segmentation [6] are some of the commonly studied binary classification problems in clinical medicine. Several 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 typically generated by multiclass classification problems, and usually there is no specific ordering among those classes. In the healthcare domain, categorical outcomes always appears in multiple disease diagnostics 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 and in several cases it is to predict the grade/severity of illness [11]. Finally, survival outcomes are particularly used for studying survival analysis that aims at analyzing the time to event data and the goal here is to predict the time to event of interest.

In this chapter we will provide more details about all these models and their applications in clinical medicine. In addition, we will also discuss different ways to evaluate such models in practice. The remainder of this chapter is organized as follows: In Section 10.2, we review some statistical prediction models. Some machine learning methods are introduced in Section 10.3, and the survival models are discussed in Section 10.4. We also provide some model evaluation and validation methods in Section 10.5, and finally, Section 10.6 concludes this chapter.

10.2 Basic Statistical Prediction Models

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

10.2.1 Linear Regression

In linear regression the dependent variable or outcome is assumed to be a linear combination of the attributes with corresponding estimated regression parameters [12]. In clinical data analysis, linear regression is often employed in clinical cost prediction [1, 2] and the estimation of some medical inspection [3]. Let us consider a sample of N subjects with p attributes, which can be represented as a $N \times p$ matrix X, and the observed output is a vector $Y^T = (y_1, y_2, ..., y_N)$. For a particular individual i, let $X_i = (x_{i1}, x_{i2}, ..., x_{ip})$ denote the covariate vector, and the output is a continuous real number denoted by Y_i. The linear regression model can be mathematically expressed as:

$$\hat{y}_i = \alpha + \sum_{j=1}^{p} x_{ij}\beta_j, \tag{10.1}$$

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 should be noted that all the input covariate values should be numeric; otherwise, the addition and multiplication computation of the covariate values is not feasible. 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

$$\begin{aligned} RSS(\beta) &= \sum_{i=1}^{N} (y_i - \hat{y}_i)^2 \qquad (10.2) \\ &= \sum_{i=1}^{N} (y_i - \alpha + \sum_{j=1}^{p} x_{ij}\beta_j)^2. \end{aligned}$$

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

$$RSS(\beta) = (Y - X\beta)^T (Y - X\beta). \tag{10.3}$$

It should be noticed that the X here is different from the definition above; here it is an $N \times (p+1)$ matrix where a unit column vector is added to 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

$$\frac{\partial RSS}{\partial \beta} = -2X^TY + 2(X^TX)\beta, \tag{10.4}$$

By letting Equation 10.4 equal 0 we will get the estimated parameter to be

$$\hat{\beta} = (X^TX)^{-1}X^TY. \tag{10.5}$$

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

10.2.2 Generalized Additive Model

To model the continuous outcomes in regression, the popular choice is to use the generalized additive model (GAM) [13], which is a linear combination of smooth functions. It can be viewed as a variant of linear regression that can handle nonlinear distribution. In GAM, for individual X_i, the continuous outcome y_i can be estimated by:

$$\hat{y}_i = \alpha + f_1(x_{i1}) + f_2(x_{i2}) + \cdots + f_p(x_{ip}), \tag{10.6}$$

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 learned using the backfitting algorithm that was introduced in 1985 by Leo Breiman and Jerome Friedman [14]. It is an iterative method that can handle a wide variety of smooth functions; however, the termination criterion of the iterations is difficult to choose, and it almost always suffers from overfitting. An alternative method of GAM estimation is using the semi-parametric smoothing function and fit the model by penalized regression splines. More details about these models can be found in [15].

10.2.3 Logistic Regression

Logistic Regression is one of the most popular binary classification methods which is widely adopted for clinical prediction tasks [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, let us consider a two-class scenario with N-samples. For a certain individual $X_i = (x_{i0}, x_{i1}, x_{i2}, ..., x_{ip})$, the observed output y_i can be labeled as either 0 or 1; the formulation of the logistic regression is

$$log \frac{Pr(y_i = 1|X_i)}{Pr(y_i = 0|X_i)} = \sum_{k=0}^{p} x_{ik}\beta_k = X_i\beta. \tag{10.7}$$

Here, $x_{i0} = 1$ and β_0 is the intercept. 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 (10.7), we have

$$Pr(y_i = 1|X_i) = \frac{exp(X_i\beta)}{1 + exp(X_i\beta)}. \tag{10.8}$$

The parameter estimation in logistic regression models is usually done by maximizing the likelihood function. The joint conditional probability of all N samples in the training data is

$$Pr(y = y_1|X_1) \cdot Pr(y = y_2|X_2) \cdot\ \ldots\ \cdot Pr(y = y_N|X_N) = \prod_{i=1}^{N} Pr(y = y_i|X_i), \tag{10.9}$$

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

$$\mathfrak{L}(\beta) = \sum_{i=1}^{N} \log[Pr(y = y_i|X_i)], \tag{10.10}$$

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

$$\log[Pr(y = y_i|X_i)] = \left\{ \begin{array}{rcl} X_i\beta - \log[1 + exp(X_i\beta)] & : & y_i = 1 \\ -\log[1 + exp(X_i\beta)] & : & y_i = 0 \end{array} \right., \tag{10.11}$$

thus, Equation (10.10) can be rewritten as:

$$\mathfrak{L}(\beta) = \sum_{i=1}^{N} \{X_i\beta \cdot y_i - \log[1 + exp(X_i\beta)]\}. \tag{10.12}$$

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

$$\beta^{(t+1)} = \beta^{(t)} - \left[\frac{\partial^2 \mathfrak{L}(\beta)}{\partial\beta\partial\beta^T}\right]^{-1} \frac{\partial \mathfrak{L}(\beta)}{\partial\beta}, \tag{10.13}$$

where

$$\frac{\partial \mathfrak{L}(\beta)}{\partial\beta} = \sum_{i=1}^{N} X_i(y_i - \frac{exp(X_i\beta)}{1 + exp(X_i\beta)}) \tag{10.14}$$

$$\frac{\partial^2 \mathfrak{L}(\beta)}{\partial\beta\partial\beta^T} = -\sum_{i=1}^{N} X_i X_i^T \frac{exp(X_i\beta)}{[1 + exp(X_i\beta)]^2}. \tag{10.15}$$

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.

10.2.3.1 Multiclass Logistic Regression

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

$$Pr(y_i = j|X_i) = \frac{exp(X_i\beta_j)}{\sum_{k \neq j} exp(X_i\beta_k)}, \tag{10.16}$$

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

$$\mathfrak{L}(\beta) = \sum_{i=1}^{N} [(X_i\beta_j) - \log(\sum_{k \neq j} exp(X_i\beta_k))]. \tag{10.17}$$

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].

10.2.3.2 Polytomous Logistic Regression

Polytomous logistic regression [21, 22] is an extension of the basic logistic regression, which is designed to handle multiclass problems. Polytomous logistic regression is used when there is no predefined order among the categories; in clinical analysis it has been used to deal with some complex datasets such as CT scans [9]. It learns different set of coefficients for different classes, in other words, each feature has a different 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 (C^{th} class) as the reference category, then the model will be:

$$\begin{aligned} \log\frac{Pr(y=1|X_i)}{Pr(y=C|X_i)} &= X_i\beta_1 \\ \log\frac{Pr(y=2|X_i)}{Pr(y=C|X_i)} &= X_i\beta_2 \\ &\vdots \\ \log\frac{Pr(y=C-1|X_i)}{Pr(y=C|X_i)} &= X_i\beta_{C-1}. \end{aligned} \tag{10.18}$$

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:

$$\begin{aligned} Pr(y=k|X_i) &= \frac{exp(X_i\beta_k)}{1+\sum_{j=1}^{C-1} exp(X_i\beta_j)},\ k=1,2,...,C-1 \\ Pr(y=C|X_i) &= \frac{1}{1+\sum_{j=1}^{C-1} exp(X_i\beta_j)}. \end{aligned} \tag{10.19}$$

The model can then be learned by maximum a posteriori (MAP). More details about the learning procedure can be found in [23].

10.2.3.3 Ordered Logistic Regression

Ordered logistic regression (or ordered logit) is an extension of the logistic regression that aims to solve an ordered output prediction. 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 proposed based on the basic assumption that all the differences between different categories are introduced by different intercepts, while the regression coefficients among all levels are the 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:

$$logit[Pr(y \leq j|X_i)] = \log\frac{Pr(y \leq j|X_i)}{1-Pr(y \leq j|X_i)} = \alpha_j - X_i\beta, \tag{10.20}$$

where $j=1,2,...,C$, and $\alpha_1 < \alpha_2 < \cdots < \alpha_{C-1}$. The other thing 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 restricted and thus is not applicable to a wide range of problems.

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

$$Pr(y_i > j|X_i) = \frac{exp(X_i\beta_j)}{1+exp(X_i\beta_j)} = g(X_i\beta_j),\ j=1,2,...,C-1, \tag{10.21}$$

where C is the number of ordinal categories. From the Equation (10.21), the posterior probabilities that Y will take on each of the values $1,...,C$, conditional on X_i, are equal to

$$Pr(y_i = j|X_i) = \begin{cases} 1 - g(X_i\beta_1) & : \quad j = 1 \\ g(X_i\beta_{j-1}) - g(X_i\beta_j) & : \quad j = 2,...,C-1 \\ g(X_i\beta_{C-1}) & : \quad j = C \end{cases} . \tag{10.22}$$

A popular Stata program "gologit2" [27] can be used to efficiently fit this model.

10.2.4 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

$$Pr(Y|X) = \frac{Pr(X|Y) \cdot Pr(Y)}{Pr(X)}, \tag{10.23}$$

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

10.2.4.1 Naïve Bayes Classifier

The main intuition of 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 label (y_{chosen}) as the estimated label for individual $X_i = (x_{i1}, x_{i2}, ..., x_{ip})$. From Equation (10.23), we can see that, in order 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 while comparing different y's; the denominator in the Equation (10.23) remains a 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 covariate 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:

$$Pr(X_i|Y = y) = \prod_{k=1}^{p} Pr(x_{ik}|Y = y), \tag{10.24}$$

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:

$$Pr(Y = y|X_i) \propto Pr(Y = y) \prod_{k=1}^{p} Pr(x_{ik}|Y = y). \tag{10.25}$$

Finally, the class label y_{chosen}, which maximizes the $Pr(Y = y)\prod_{k=1}^{p} Pr(x_{ik}|Y = y)$ is chosen to be the output.

10.2.4.2 Bayesian Network

Although the naïve Bayes classifier is a straightforward implementation of Bayesian classifier, in most real-word scenarios there are certain relationships that exist among the attributes. A Bayesian network introduces a *directed acyclic graph* (DAG), which represent a set of random variables by nodes and their dependency 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 will be the prior probability of the current node.

More specifically, in decision making or prediction problems, this Bayesian network can be viewed in terms of a hierarchical structure. Only the independent attributes that have prior probability are in the top level. For example, in Figure 10.1, there are 5 attributes that contribute to the output; among them "Smoking (Attribute 3)" and "Family history of heart disease (Attribute 5)" do not have any predecessors, so we can compute the prior probabilities $Pr(\text{Smoking})$ and $Pr(\text{Family history of heart disease})$ directly; "Aortic rupture (Attribute 1)" and "Hypertension (Attribute 4)" are in the second level, and their conditional probabilities are $Pr(\text{Aortic rupture}|\text{Smoking})$ and $Pr(\text{Hypertension}|\text{Smoking})$ respectively; "Stroke (Attribute 2)" is in the third level and its conditional probability is $Pr(\text{Stroke}|\text{Aortic rupture, Smoking, Hypertension})$. "Heart rate (Attribute 6)" and "Blood pressure (Attribute 7)" are two medical observations, and "Heart attacks" is the disease that needs to be predicted.

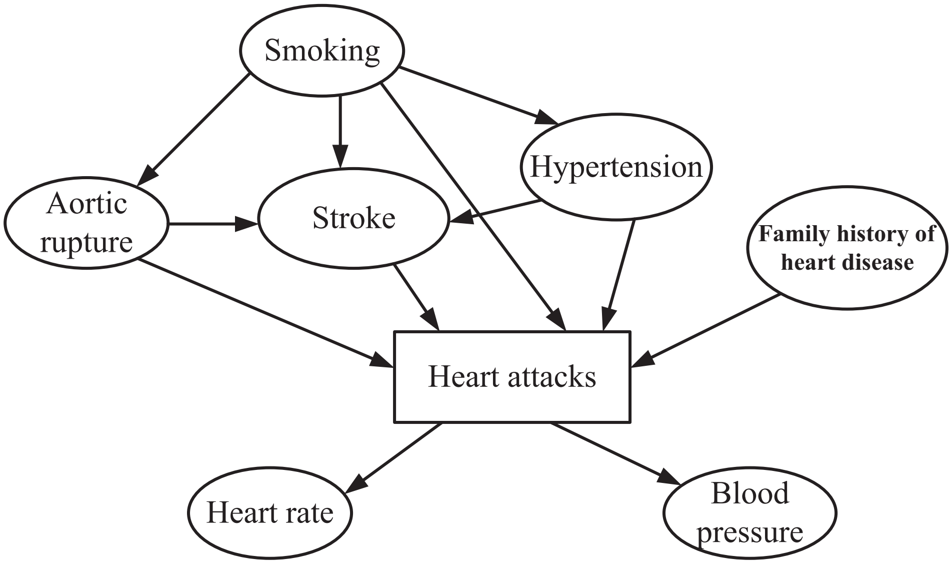


FIGURE 10.1: An example of a Bayesian network for decision making.

Based on this network, the joint probability function is computed as follows:

$$\begin{aligned} Pr(\text{Heart attacks}, 1,2,3,4,5,6,7) &= Pr(6|\text{Heart attacks}) \cdot Pr(7|\text{Heart attacks}) \\ &\quad \cdot Pr(\text{Heart attacks}|1,2,3,4,5) \cdot Pr(2|1,3,4) \\ &\quad \cdot Pr(1|3) \cdot Pr(4|3) \cdot Pr(3) \cdot Pr(5) \end{aligned} \tag{10.26}$$

Based on Equation(10.26), the $Pr(\text{Heart attacks}|1,2,3,4,5)$ for each kind of output can be calculated conditional on a specific combination of 5 different attributes.

10.2.5 Markov Random Fields

In the Bayesian network, the nodes are connected based on causality; however, in real-world applications, 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 the healthcare domain, Markov random fields were 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 vertices and E is the set of edges; each vertex $v \in V$ represents a covariate vector X_v. In MRF, the conditional independence relationship is defined via the topology of the undirected graphical model. 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 conditionally independent of 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 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 nonadjacent nodes, X_v and X_u, are conditionally independent of all the nodes in the rest of the graph, which is known as the pairwise Markov property, and can be mathematically represented as: $X_v \perp X_u | X_{rest}$.

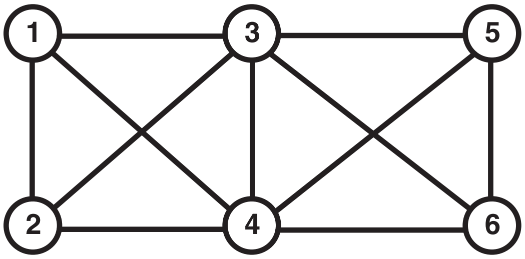


FIGURE 10.2: An example of an undirected graph.

In order to describe the Markov properties more intuitively, let us illustrate these conditional independence relations based on Figure 10.2.

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**, $\{1\} \perp \{6\} | \{2,3,4,5\}$.

10.3 Alternative Clinical Prediction Models

In addition to the basic prediction models explained in the previous section, more recent developments in the machine learning and data mining literature allowed the biomedical researchers to apply other prediction models in clinical applications. These models include decision trees, artificial neural networks. While there are many other traditional prediction models that have been used in certain specific biomedical application, a complete discussion about the prediction models and their applications is out of scope of this chapter. We focus on the most widely used prediction models in this section. In addition, an important concept of cost-sensitive learning in the context of prediction which was motivated through some of the important biomedical problems will also be discussed in this section. In addition to these models and algorithm, more advanced clinical prediction methods such as multiple instance learning, reinforcement learning, sparse models, and kernel methods will also be discussed in this section.

10.3.1 Decision Trees

A decision tree is the most widely used clinical prediction model that has been successfully used in practice [32]. In a decision tree model, the 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 with respect to the predicted outcome. Choosing the proper splitting criteria, obviously, is a critical component for decision tree building. These criteria can help to find the locally optimum decisions that 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 that makes the decision tree very unstable [37]. Decision trees have been heavily used in the medical decision making in a wide range of applications [38, 39].

10.3.2 Artificial Neural Networks

Inspired by biological neural systems, in 1958, Frank Rosenblatt published the first paper [40] about the artificial neural network (ANN), in which simple artificial nodes, called "neurons," are combined via a weighted link to form a network that simulates a biological neural network. A neuron is a computing element that consists of sets of adaptive weights and generates the output based on a certain kind of *activation function*. A simple artificial neural network named *perceptron* only has input and output layers. For a specific input attribute vector X_i the perception model can be written as: $\hat{y}_i = sign(X_i W)$ 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 using an iterative algorithm that updates the weights using the following update rule: $w_j^{(t+1)} = w_j^{(t)} + \lambda(y_i - \hat{y}_i^{(t)})x_{ij}$ 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 the 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 10.3.

In ANN learning the commonly used cost function to minimize is the *mean-squared error*, which is the average squared difference between the estimated output and the real one. Because of

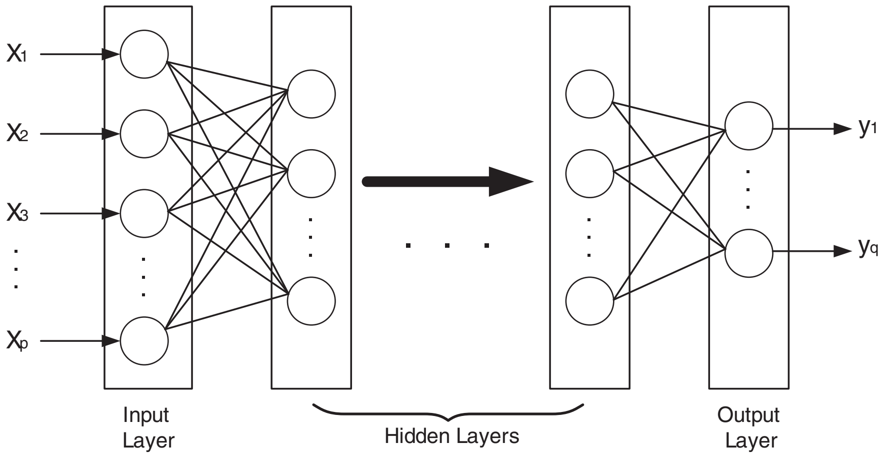


FIGURE 10.3: Example of a multilayer artificial neural network (ANN).

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. Due to their ability to model the complex mapping function and the rich literature in the machine learning community, the artificial neural networks have been widely used in various biomedical applications [41, 42]. Some of the prominent applications include decision support systems [43], medical intervention [44], and medical decision making [45]. A more detailed comparison of the performance of ANN with logistic regression is provided in this paper [16].

10.3.3 Cost-Sensitive Learning

A somewhat related concept that is extremely popular in the healthcare domain is the topic of *cost-sensitive learning*. Certain clinical prediction models [46] also can be viewed as cost-sensitive models. In the predictive model the learning process aims to minimize the sum of the total costs. Among different types of costs [47], computation cost and the cost of instability are two vital factors that need to be considered while designing various algorithms. In this section we only focus on two categories of cost that are widely used in the biomedical domain, namely, misclassification costs and test costs.

Misclassification cost is introduced by classification error. In the real world, the cost associated with 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 the literature [48, 49], and the basic idea of these works can be 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. Let us consider a set of N subjects. For each individual $x_i,\ i = 1,2,...,N$, the actual label is $y_i = a$, and $Pr(p|x_i,a)$ is the estimated probability that x_i belongs to the class p. Thus, for misclassification cost the cost-sensitive learning aims to minimize the following function $\min \sum_{i=1}^{N} \sum_{p \in L} Pr(p|x_i, a) c_{ap}$. In a two-class scenario, the cost matrix C is structured as in Table 10.1 below:

TABLE 10.1: 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 process, such as re-selecting the threshold [49] and changing the splitting criteria during the tree induction [50], or after the learning phase during the performance evaluation of the model step [51] where we will just multiply the corresponding elements from the cost matrix and confusion matrix [51] and then calculate the sum of these products. This concept of learning in a cost-sensitive manner is very much related to the problem of imbalanced learning, which is heavily investigated in the biomedical literature [52]. Here the class with minority samples will be assigned a large misclassification cost.

Test cost or *the cost of obtaining the information* is incurred while obtaining the attribute values. For example, in disease diagnosis, a patient already had the X-ray test but did not have 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 [53].

10.3.4 Advanced Prediction Models

More recent advances in the machine learning literature allowed the clinical researchers to apply complex prediction models to achieve better accuracy in nontrivial situations. Some examples of these methods include multiple instance learning, reinforcement learning, sparse methods, and kernel methods. We will now briefly discuss these approaches in this section.

10.3.4.1 Multiple Instance Learning

Unlike other prediction methods, in multiple instance learning [54], the exact label of each individual is actually unknown. Instead, the training data are packed into a set of labeled groups. A group is labeled positive if there is at least one positive instance in it; whereas, a group is labeled negative only when all the individuals in that group are negative. Multiple instance learning is often applied in diverse fields such as image classification, text mining, and the analysis of molecular activity. In clinical fields it is usually used to analyze radiology images especially when there are several hundreds of image slices for each patient. These slices are highly correlated and a patient is termed as having cancer even if a single image slice has a suspicious mass. In [55], researchers have successfully deployed the multiple instance learning algorithm based on convex hulls into practical computer-aided diagnostic tools to detect pulmonary embolism and colorectal cancer. In another study [56], for CT pulmonary angiography, multiple instance learning has been employed to detect pulmonary emboli.

10.3.4.2 Reinforcement Learning

Reinforcement learning aims to maximize the long-term rewards; it is particularly well suited to problems that include a long-term versus short-term reward trade-off [57]. In reinforcement learning, an action corresponds to 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. Zhao et al. [58] used reinforcement learning to discover individualized treatment regimens. An optimal policy is learned from a single training set of longitudinal patient trajectories. Sahba et al. [59] proposed a reinforcement learning framework for medical image segmentation. In medical computer-aided detection (CAD) systems reinforcement learning could be used to incorporate the knowledge gained from new patients into old models.

10.3.4.3 Sparse Methods

Sparse methods perform feature selection by inducing the model coefficient vector to be sparse, in other words, contain many zero terms. 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 the most important features [60]. 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. An excellent survey on sparsity inducing norms and their utility in biomedical data analysis and prediction problems is available in [61]. With the availability of several high-dimensional genomic and clinical datasets in recent times, sparse methods have gained a lot of popularity in biomedical applications. Methods such as LASSO and Elastic Net are popular choices for penalty functions.

10.3.4.4 Kernel Methods

Kernel methods map the attributes from the original feature space to an abstract space where it is often much easier to distinguish multiple classes [62]. Kernel methods typically achieve a better performance by projecting the data into a higher-dimensional kernel space where a linear classifier can accurately separate the data into multiple categories. Choosing the right kernel is a challenging problem and in practice, researchers resort to some of the standard ones available in the literature and tune their parameters based on experimental results [18]. A kernel measures the similarity between two data objects: the more similar two objects X and X' are, the higher the value of a kernel $K(X,X')$ will be. Several kernel functions have been proposed in the literature. Polynomial kernels are well suited for problems where all the training data is normalized. The formulation of the polynomial kernel is:

$$K(X,X') = (\alpha X^T X' + c)^d, \tag{10.27}$$

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 [63]; the definition of a Gaussian kernel is

$$K(X,X') = exp\left(-\frac{||X - X'||^2}{2\sigma^2}\right) \tag{10.28}$$

where σ^2 is known as the *bandwidth*, which plays a major role in the performance of the Gaussian kernel.

Kernel methods are an effective alternative to perform data integration in the presence of heterogeneous data sources. In such problems, one does not have to perform explicit feature extraction before combining data sources. The learning method can automatically learn the appropriate feature spaces in each of the data sources and effectively integrate them to provide a robust prediction model with better accuracy compared to the models built on individual data sources. The authors in [64] provided a comprehensive set of experimental results in several biomedical applications to demonstrate the power of multiple kernel learning. Such multiple kernel learning methods fall into the category of intermediate integration where the prediction models are simultaneously learned from heterogeneous data sources by choosing the optimal feature space. Wang et al. [65] proposed a colonic polyp detection framework where multiple kernels are used to extract and combine the features from different sources (such as statistical and geometric features).

10.4 Survival Models

Survival analysis [66, 67] aims at modeling the time to event data; the observation starts from a particular starting time and will continue until the occurrence of a certain event or the observed objects become missing (not observed) from the study. In the healthcare domain, the starting point of the observation is usually a particular medical intervention such as a hospitalization admission, the beginning of taking a certain medication or a diagnosis of a given disease. The event of interest might be death, discharge from the hospitalization, or any other interesting incident that can happen during the observation period. The missing trace of the observation is also an important characteristic of survival data. For example, during a given hospitalization some patients may be moved to another hospital and in such cases, that patient will become unobserved from the study with respect to the first hospital. Survival analysis is useful whenever we are interested not only in the frequency of occurrence of a particular type of event, but also in estimating the time for such an event occurrence. In healthcare applications, the survival prediction models mainly aim at estimating the failure time distribution and estimating the prognostic evaluation of different variables (jointly or individually considered) such as biochemical, histological, and clinical characteristics [68].

10.4.1 Basic Concepts

In this section, the basic concepts and characteristics of survival models will be introduced along with some examples. The examples come from real patient data about heart failure readmission problems collected at a major hospital in the southeastern Michigan region. In this problem, survival analysis is used to estimate the time between the discharge of a patient from hospitalization and the readmission of that patient for heart failure diagnosis. Here, the event of interest is hospital readmission, and the beginning of the observation starts from the discharge date of the previous hospitalization. From this section, we hope that the difference between survival analysis and the standard predictive models will become clear to the readers.

10.4.1.1 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 [66].

Let us consider a small number of N heart failure patients in the rehospitalization problem; suppose the observation terminates after 30 days of discharge. Thus, the time of the hospital readmission is known precisely only for those subjects for whom the event has occurred before the ending point (30 days in this case). 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 being hospitalized due to other conditions. All of these scenarios are considered as censoring in this particular example. Figure 10.4 describes the concept of censoring in a more intuitive manner. 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 *Right*

Censoring; otherwise, if $Z = max(T,U)$, it is termed as *Left Censoring*. Practically, in the healthcare domain the majority of the survival data is right censored [68].

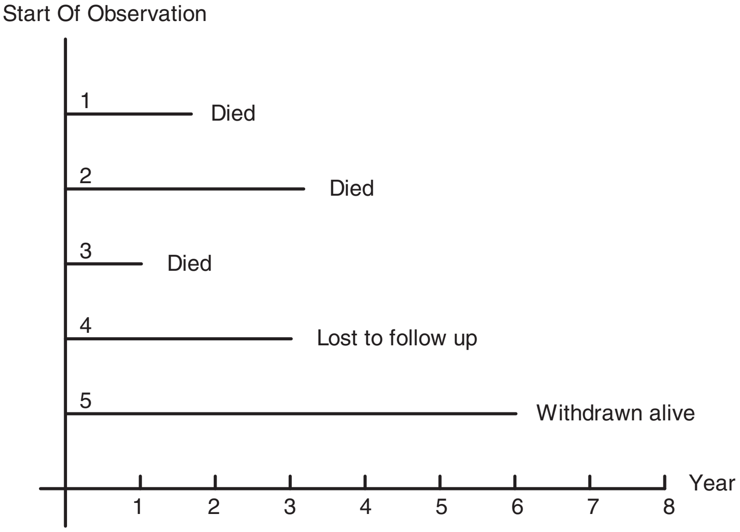


FIGURE 10.4: An illustration that demonstrates the concept of censoring.

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* [69]. An example of a small survival dataset, which is from our heart failure readmission problem, is shown in Table 10.2. In this dataset, for simplicity, we show only the patients' age and sex as our feature set (which is the X in the notation); the "status" is the indicator δ, and the "Gap" is the observed time.

10.4.1.2 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 [69, 66]. Conventionally, survival function is denoted as S, which is defined as:

$$S(t) = Pr(T \geq t). \tag{10.29}$$

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 less than t, and *death density function* $f(t)$ is defined as $f(t) = \frac{d}{dt}F(t)$ for continuous scenarios, and $f(t) = \frac{F(t+\Delta t)-F(t)}{\Delta t}$, where Δt is a short time interval, for discrete scenarios. The relationship among these functions is clearly described in Figure 10.5.

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* [70]. 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

TABLE 10.2: 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

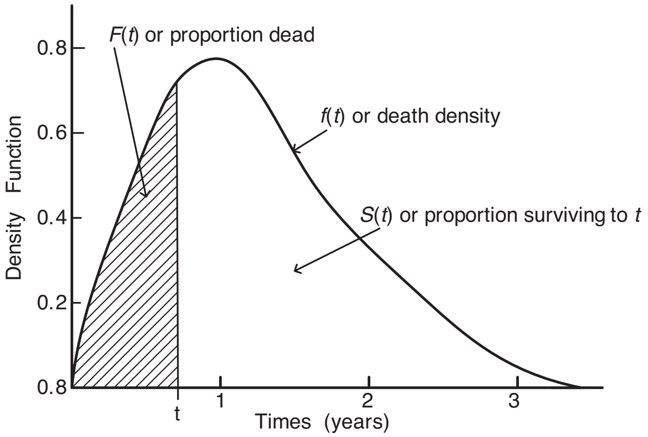


FIGURE 10.5: Relationship among $f(t)$, $F(t)$, and $S(t)$.

is defined as:

$$\begin{aligned}\lambda(t) &= \lim_{\Delta t \to 0} \frac{Pr(t \leq T < t + \Delta t \mid T \geq t)}{\Delta t} \\ &= \lim_{\Delta t \to 0} \frac{F(t+\Delta t) - F(t)}{\Delta t \cdot S(t)} \\ &= \frac{f(t)}{S(t)}. \end{aligned} \qquad (10.30)$$

Similar to $S(t)$, $\lambda(t)$ is also a nonnegative function. Whereas all survival functions, $S(t)$, decrease over time, the hazard function can take on a variety of shapes. Consider the definition of $f(t)$, which can also be expressed as $f(t) = -\frac{d}{dt}S(t)$, so the hazard function can be represented as:

$$\lambda(t) = \frac{f(t)}{S(t)} = -\frac{d}{dt}S(t) \cdot \frac{1}{S(t)} = -\frac{d}{dt}[lnS(t)]. \tag{10.31}$$

Thus, the survival function can be rewritten as

$$S(t) = exp(-\Lambda(t)) \tag{10.32}$$

where $\Lambda(t) = \int_0^t \lambda(u)du$ is the *cumulative hazard function* (CHF) [69].

10.4.2 Nonparametric 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.

10.4.2.1 Kaplan–Meier Curve and Clinical Life Table

In this section, we will 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 one. In 1958, Kaplan and Meier [71] 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 [72]. We will describe both of these methods in this section.

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, whose either death or censored time is greater than or equal to T_j, are considered to be "at risk." The obvious conditional probability of surviving beyond time T_j can be defined as:

$$p(T_j) = \frac{r_j - d_j}{r_j} \tag{10.33}$$

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

$$\hat{S}(t) = \prod_{j:T_j<t} p(T_j) = \prod_{j:T_j<t} (1 - \frac{d_j}{r_j}) \tag{10.34}$$

and its variance is defined as:

$$Var(\hat{S}(t)) = \hat{S}(t)^2 \sum_{j:T_j<t} \frac{d_j}{r_j(r_j - d_j)}. \tag{10.35}$$

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 censored cases 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 10.2. The calculated result is shown in Table 10.3, and the corresponding K–M survival curve is shown in Figure 10.6.

TABLE 10.3: Kaplan–Meier Estimator of 40 Heart Failure Patients in Table 10.2

						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

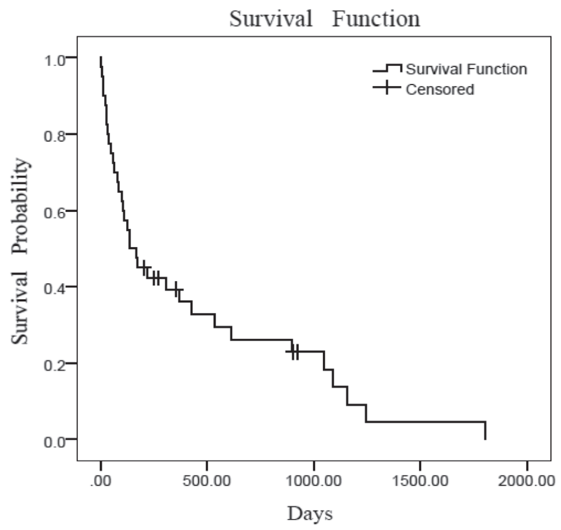


FIGURE 10.6: Kaplan–Meier survival curve of 40 heart failure patients in Table 10.2.

Clinical Life Table As mentioned above, the Clinical Life Table [72] 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, 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 censored cases 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 in the case of the Kaplan–Meier estimator, the conditional probability of surviving during jth interval is estimated as

$$\hat{p}_j = 1 - \frac{d_j}{r_j} \tag{10.36}$$

and the corresponding survival function is estimated by the product

$$\hat{S}(I_j) = \prod_{i:i<j} (1 - \frac{d_i}{r_i}) \tag{10.37}$$

and the standard variation of this $\hat{S}(I_j)$ can be calculated in a similar way as it is in the Kaplan–Meier Curve. Table 10.4 illustrates the computation of the Clinical Life Table within 40 heart failure patients, which are shown in Table 10.2. In this example, we chose the interval length as 0.5 years (183 days), and all 40 patients are partitioned into 10 intervals.

TABLE 10.4: 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

10.4.2.2 Mantel–Haenszel Test

In clinical research, one is concerned not only with estimating the survival probability but also, more often, with the comparison of the life experience of two or more groups of subjects differing for a given characteristic or randomly allocated to different treatments. The nonparametric approach is usually adopted also to compare survival curves. Among the various nonparametric tests that are available in the statistical literature, the Mantel–Haenszel (M–H) test [73] is one of the most frequently used statistical tools in medical reports for analyzing survival data (Table 10.5).

Let $T_1, T_2, ...T_J$ represent the J ordered, distinct death times, and in the jth death time, r_j number of patients survived, and d_j number of deaths occurred. Suppose that, based on certain characteristics, 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 Haenszel 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

TABLE 10.5: Mantel–Haenszel Test in 2 Groups

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

the expectation of d_{1j} is

$$E(d_{1j}) = r_{1j}\cdot\frac{d_j}{r_j}, \tag{10.38}$$

and the variance of d_{1j} is

$$Var(d_{1j}) = \left[r_{1j}\cdot\frac{d_j}{r_j}(1-\frac{d_j}{r_j})\right]\frac{r_j-r_{1j}}{r_j-1} = \frac{r_{1j}r_{0j}d_j(r_j-d_j)}{r_j^2(r_j-1)}. \tag{10.39}$$

The ratio is approximately distributed as a chi-square with one degree of freedom [74], and hence, for all J ordered distinct death times, the ratio is

$$X^2 = \frac{[\sum_{j=1}^{J}(d_{1j}-E(d_{1j}))]^2}{\sum_{j=1}^{J}Var(d_{1j})}. \tag{10.40}$$

Beside this Mantel–Haenszel test, there are also some nonparametric methods that have been used to compare the survival difference. In 1965, Gehan [75] proposed a generalized Wilcoxon test that is an extension of the Wilcoxon test of censored data. Later, Peto and Peto [76] suggested another version of the generalized Wilcoxon test. These nonparametric 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.

10.4.3 Cox Proportional Hazards Model

The Cox proportional hazards model [77] 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.

10.4.3.1 The Basic Cox Model

Let N be the number of subjects in the survival analysis, and as mentioned in Section 10.4.1, each of the individuals can be represented by a triple of variables (X,Z,δ). Considering an individual specific hazard function $\lambda(t,X_i)$ in the Cox model, the proportional hazards assumption is

$$\lambda(t,X_i) = \lambda_0(t)exp(X_i\beta), \tag{10.41}$$

for $i = 1,2,...,N$, where the $\lambda_0(t)$ is the *baseline hazard function*, which can be an arbitrary nonnegative 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 semiparametric 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

$$\frac{\lambda(t,X_1)}{\lambda(t,X_2)} = \frac{\lambda_0(t)exp(X_1\beta)}{\lambda_0(t)exp(X_2\beta)} = exp[(X_1 - X_2)\beta]. \tag{10.42}$$

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

$$S(t) = exp(-\Lambda_0(t)exp(X\beta)) = S_0(t)^{exp(X\beta)} \tag{10.43}$$

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

10.4.3.2 Estimation of the Regression Parameters

Because the baseline hazard function $\lambda_0(t)$ in the Cox proportional hazards model is not specified, it is impossible to fit the model using the standard likelihood function. To estimate the coefficients, Cox [77] proposed a partial likelihood that represents the data only depending on the β values. 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 $\lambda(t,X)dt, dt \rightarrow 0$. Again, let N be the number of subjects who have a total number of $J \leq N$ events of interest occurring during the observation period, 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 the event occurs at T_j, the probability of the individual's corresponding covariate is X_j can be formulated as

$$\frac{\lambda(T_j,X_j)dt}{\sum_{i\in R(T_j)}\lambda(T_j,X_i)dt}, \tag{10.44}$$

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

$$L(\beta) = \prod_{j=1}^{N}\left[\frac{exp(X_j\beta)}{\sum_{i\in R_j} exp(X_i\beta)}\right]^{\delta_j}. \tag{10.45}$$

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*

$$LL(\beta) = \sum_{j=1}^{N} \delta_j\{X_j\beta - log[\sum_{i\in R_j} exp(X_i\beta)]\}. \tag{10.46}$$

10.4.3.3 Penalized Cox Models

Currently, with the development of medical procedures and detection methods, medical records tend to have more features than ever before. In some cases, the number of features (P) is almost equivalent to or even larger than the number of subjects (N); building the prediction model with all the features might provide inaccurate results because of the overfitting issues [78]. The primary motivation of using sparsity-inducing norms is that in high dimensions, it becomes appropriate 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 [60]. In biomedical data analysis the sparsity-inducing norms are also widely used to penalize the loss function of a prediction [61]. Consider the L_p norm penalty; the smaller the value of 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 to obtain. Commonly, the penalized methods have also been used to do feature selection in the scenarios when $N > P$. We will now introduce three commonly used penalty functions and their applications in the Cox proportional hazards model.

Lasso [79] is a L_1 norm penalty that can select at most $K = min(N,P)$ features while estimating the regression coefficient. In [80], the Lasso penalty was used along with the log-partial likelihood to obtain the Cox-Lasso algorithm.

$$\hat{\beta}_{lasso} = \min_{\beta}\{-\frac{2}{N}[\sum_{j=1}^{N} \delta_j X_j \beta - \delta_j log(\sum_{i \in R_j} e^{X_i \beta})] + \lambda \sum_{p=1}^{P} |\beta_p|\} \tag{10.47}$$

Elastic Net, which is a combination of the L_1 and squared L_2 norm penalties, has the potential to obtain both sparsity and handle correlated feature spaces [81]. The Cox-Elastic Net method was proposed by Noah Simon et al. [82] wherein the Elastic Net penalty term was introduced into the log-partial likelihood function

$$\begin{aligned} \hat{\beta}_{elastic\ net} &= \min_{\beta}\{-\frac{2}{N}[\sum_{j=1}^{N} \delta_j X_j \beta - \delta_j log(\sum_{i \in R_j} e^{X_i \beta})] \\ &+ \lambda[\alpha \sum_{p=1}^{P} |\beta_p| + \frac{1}{2}(1-\alpha) \sum_{p=1}^{P} \beta_p^2]\} \end{aligned} \tag{10.48}$$

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 originally proposed by Hoerl and Kennard [83] and introduced to the Cox regression by Verweij and Van Houwelingen [84]. It is a L_2 norm regularization that tends to select all the correlated variables and shrink their values towards each other. The regression parameters of Cox-Ridge can be estimated by

$$\hat{\beta}_{ridge} = \min_{\beta}\{-\frac{2}{N}[\sum_{j=1}^{N} \delta_j X_j \beta - \delta_j log(\sum_{i \in R_j} e^{X_i \beta})] + \frac{\lambda}{2} \sum_{p=1}^{P} \beta_p^2\}. \tag{10.49}$$

Among three equations (10.47), (10.48), and (10.49), $\lambda \geq 0$ is used to adjust the influence introduced by the penalty term. The performance of these penalized estimator significantly depends on λ, and the optimal λ_{opt} can be chosen via cross-validation. Also, few other penalties based on kernel and graph-based similarities have been recently proposed to tackle the inherent correlations within the variables in the context of the Cox proportional hazards model [85].

10.4.4 Survival Trees

Survival trees are one form of classification and regression trees that are tailored to handle censored data. The basic intuition behind the tree models is to recursively partition the data based on a particular splitting criterion, and the objects that belong to the same node are similar to each other based on the event of interest. The earliest attempt at using tree structure analysis for survival data was made in [86].

10.4.4.1 Survival Tree Building Methods

The primary difference between a survival tree and the standard decision tree is the choice of splitting criterion. The splitting criteria used 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 function based within-node homogeneity criterion. Gordon and Olshen [87] measured the homogeneity by using L_P, the L_P Wasserstein metric, and Hellinger distances between estimated distribution functions. Davis and Anderson [88] employed an exponential log-likelihood loss function in recursive partitioning based on the sum of residuals from the Cox model. LeBlanc and Crowley [89] measured the node deviance based on the first step of a full likelihood estimation procedure; Cho and Hong [90] proposed an L_1 loss function to measure the within-node homogeneity.

In the second category of splitting criteria, Ciampi et al. [91] employed log-rank test statistics for between-node heterogeneity measures. Later, Ciampi et al. [92] proposed a likelihood ratio statistic (LRS) to measure the dissimilarity between two nodes. Based on the Tarone-Ware class of two-sample statistics, Segal [93] introduced a procedure to measure the between-node dissimilarity.

10.4.4.2 Ensemble Methods with Survival Trees

To overcome the instability of a single tree, bagging [37] and random forests [94], proposed by Breiman, are commonly used to perform the ensemble-based model building. Hothorn et al. [95] proposed a general bagging method that was implemented in the R package “ipred.” In 2008, Ishwaran et al. introduced a general random forest method, called random survival forest (RSF) [96] and implemented it in the R package “randomSurvivalForest”.

***Bagging Survival Trees*:** Bagging is one of the oldest and most commonly used ensemble methods that typically reduces the variance of the base models being used. 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 [95]. The main steps of this method are as follows:

1. Draw B booststrap samples from the original dataset.
2. Grow a survival tree for each bootstrap sample, and ensure that in each terminal node the number of events occurred is no less than d.
3. Compute the bootstrap aggregated survival function by averaging the leaf nodes’ predictions.

For each leaf node the survival function is estimated by the Kaplan–Meier estimator [71], and all individuals within the same node are assumed to follow the same survival function.

***Random Survival Forests*:** Random forest is an ensemble method designed specifically for the tree structured prediction models [94]. It is based on a framework similar to bagging; the main difference between random forest and bagging is that at a certain node, rather than using all the 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 reduce the correlation among trees and thus improve the prediction performance.

In random survival forest, the Nelson–Aalen estimator [97, 98] is used to predict the cumulative hazard function (CHF). The definition of the Nelson–Aalen estimator is

$$\hat{\Lambda}(t) = \sum_{t_j \leq t} \frac{d_j}{r_j} \tag{10.50}$$

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 [96].

10.5 Evaluation and Validation

In this section, we will describe some of the widely studied evaluation metrics that are used in clinical medicine. We will also discuss different validation mechanisms used to obtain robust estimations of these evaluation metrics.

10.5.1 Evaluation Metrics

When we design and construct a new prediction model or apply an existing model to a particular clinical dataset, it is critical to understand whether the model is suitable for this data; thus, some evaluation metrics are needed to quantify the performance of the model. In this section, we will introduce some of the well-known metrics that are commonly used to evaluate the performance of the clinical prediction models.

10.5.1.1 Brier Score

Named after the inventor Glenn W. Brier, the Brier score [99] is designed to evaluate the performance of prediction models where the outcome to be predicted is either binary or categorical in nature. Note that the Brier score can only evaluate the prediction models that have probabilistic outcomes; that is, the outcome must remain in the range [0,1], and the sum of all the possible outcomes for a certain individual should be 1. Let us consider a sample of N individuals and for each X_i, $i = 1,2,...,N$, the predicted 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

$$\text{Brier score} = \frac{1}{N}\sum_{i=1}^{N}(\hat{y}_i - y_i)^2, \tag{10.51}$$

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 [99], has the form:

$$\text{Brier score} = \frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{C}(\hat{y}_{ic} - y_{ic})^2, \tag{10.52}$$

for C-class output problem (categorical outcome), 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, it is evident that it measures the mean-squared difference between predictions made and the actual outcomes; therefore, the lower the Brier score, the better the prediction.

10.5.1.2 R^2

The R^2 or *coefficient of determination* [100] is used to measure the performance of regression models, which can be formalized as:

$$R^2 = 1 - \frac{RSS(\hat{Y})}{Var(Y)}, \tag{10.53}$$

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

$$RSS(\hat{Y}) = \sum_{i=1}^{N}(y_i - \hat{y}_i)^2, \text{ and } Var(Y) = \sum_{i=1}^{N}(y_i - \bar{y})^2, \tag{10.54}$$

where $\bar{y}$ is the mean value of the actual 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 $RSS(\hat{Y})$; in other words, the closer the R^2 is to one, the better the prediction will be. At the same time, we should also note that the R^2 could be negative if the prediction model cannot well represent the distribution of the dataset and even worse than the mean value of the actual outcomes [101].

10.5.1.3 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 [102, 103]. Here we only consider its definition in the binary classification case where it can be used to measure the performance of the predicted model.

TABLE 10.6: Confusion Matrix for a 2-Class Problem

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

Consider a confusion matrix [104] for a 2-class problem that is shown in Table 10.6, 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.

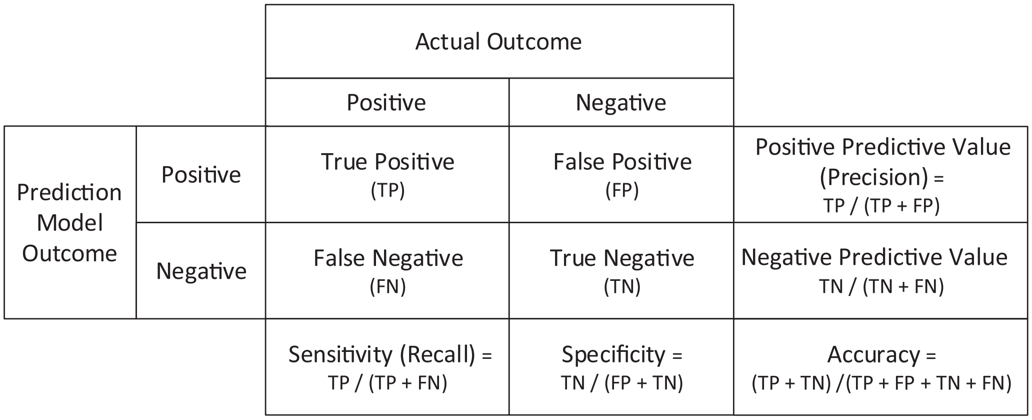
Based on this confusion matrix, the accuracy can be formalized as:

$$\text{Accuracy} = \frac{TP+TN}{TP+FP+FN+TN}, \tag{10.55}$$

which is the proportion of correct predictions over the entire set of samples.

10.5.1.4 Other Evaluation Metrics Based on Confusion Matrix

Even though accuracy is a good estimate of the model performance, it has some major drawbacks when applied in medical problems. For instance, one might be more interested in the performance of the model in prediction of the positive cases compared to the negative ones. Also, when the class distribution is imbalanced (i.e., one class completely dominates the other one), accuracy will not provide a good estimate of the model performance. Such class imbalance problems [105] are quite common in clinical applications. Let us consider a real-world example that demonstrates this class imbalance problem in a biomedical context. The World Health Organization (WHO) [106] 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. In this case, the accuracy measure is no longer suitable. For such lung cancer diagnosis, the model that predicts no one getting lung cancer has an accuracy very close to 100%; however, it is clear that this is not a good prediction because we are more interested in a model that can accurately predict the lung cancer cases (which is a minority class in this application domain). We will now introduce some of the commonly studied evaluation metrics that are suitable for such problems especially in the 2-class scenario [51]. All the terms used in the definition of these metrics are already defined in the previous section. Figure 10.7 shows the popular evaluation metrics and the manner in which they are derived from the components of the confusion matrix.



		Actual Outcome		
		Positive	Negative	
Prediction Model Outcome	Positive	True Positive (TP)	False Positive (FP)	Positive Predictive Value (Precision) = TP / (TP + FP)
	Negative	False Negative (FN)	True Negative (TN)	Negative Predictive Value TN / (TN + FN)
		Sensitivity (Recall) = TP / (TP + FN)	Specificity = TN / (FP + TN)	Accuracy = (TP + TN) /(TP + FP + TN + FN)

FIGURE 10.7: Various evaluation metrics derived from the confusion matrix.

Sensitivity Sensitivity, which is also known as the *true positive rate* (TPR) or *Recall*, measures the ratio of actual positives that are correctly identified. The formal definition of the sensitivity is

$$\text{Sensitivity} = \frac{TP}{TP+FN}. \tag{10.56}$$

Specificity Specificity, which is also known as the *true negative rate* (TNR), measures the ratio of actual negatives that are correctly identified [107]; this measurement can be employed in those problems where the negative individuals are more interesting, and it can be defined as:

$$\text{Specificity} = \frac{TN}{TN+FP}. \tag{10.57}$$

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

$$FPR = \frac{FP}{TN+FP}. \tag{10.58}$$

Precision Precision, which is also known as the *positive predictive value* (PPV), measures the ratio of true positives to predicted positives [108]; 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:

$$\text{Precision} = \frac{TP}{TP+FP}. \tag{10.59}$$

F-measure F-measure [109] is the harmonic mean of recall and precision:

$$\text{F-measure} = \frac{2 \times Precision \times Recall}{Recall + Precision}. \tag{10.60}$$

Thus, a high value of F-measure indicates that both precision and recall are reasonably high [51]. F-measure varies in the range [0, 1] where the best value is reached at 1 and the worst score will be 0.

10.5.1.5 ROC Curve

The receiver operating characteristic (ROC) curve is a graphical technique that can be used to measure and visualize the performance of a prediction model over the entire range of possible cutoffs [110]. In the biomedical domain, the ROC curve has been employed in the evaluation of disease diagnosis [111]. In an ROC curve (see Figure 10.8), 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.

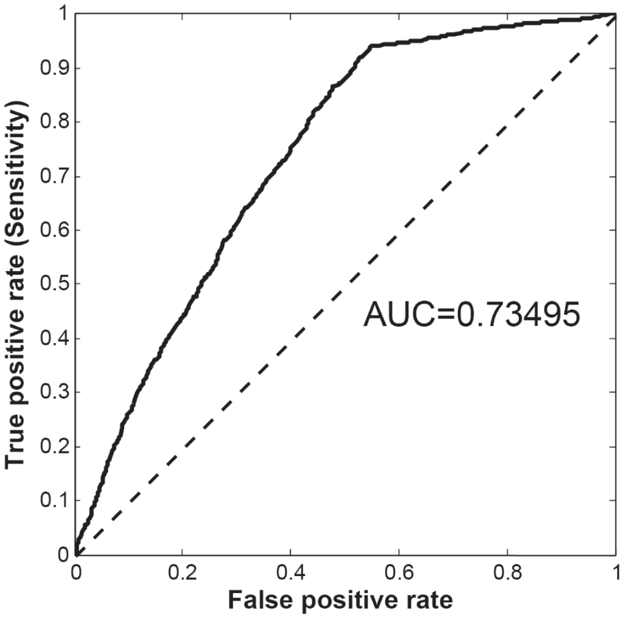


FIGURE 10.8: An example of a ROC curve.

For an ideal model, $TPR = 1$ and $FPR = 0$; that is, the area under the ROC curve (AUC) [112] will be equal to 1. In [112, 113], the meaning of AUC is thoroughly discussed in more detail, 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 value, 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 made by the model is incorrect and hence, to solve this problem, we just need to exchange the definition of the positive individual and negative individual.

10.5.1.6 *C*-index

C-index, or the *concordance probability*, is used to measure the performance of a regression model [114]. Originally, it was designed to evaluate the performance of the survival estimation [115, 116]. 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:

$$c = Pr(\hat{y}_1 > \hat{y}_2 | y_1 \leq y_2). \tag{10.61}$$

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 the C-index. In 1982, Harrell et al. proposed the first definition of the C-index [115]. Later, Heagerty and Zheng defined the c_τ in [117] which is calculated based on AUC values at all possible observation times. In [118], a c-index that is specific for the Cox model was designed. Among these three methods, Harrell et al.'s C-index [115] is suitable for all cases; in contrast, in [117] and [118] the C-index is designed specifically for the proportional hazards model, where $X_i\beta$ (see Section 10.4.3) is used instead of the estimated outcome $\hat{y}_i$.

10.5.2 Validation

In Section 14.6, we reviewed several quantitative metrics used for estimating the performance of clinical prediction models, and the model can be evaluated based on its performance on an unseen testing data. This section reviews some of the commonly used validation techniques that can provide an unbiased estimate for the evaluation of a predictive model. In general, these techniques fall into two categories: internal validation and external validation.

10.5.2.1 Internal Validation Methods

The internal validation works by randomly separating the training data and the testing data from the dataset where the labels of the individuals are already known. Here we briefly introduce two of the most commonly used internal validation methods: cross-validation and bootstrap validation.

Cross-Validation In k-fold cross-validation, first, the labeled dataset will be randomly partitioned into k equal-sized subsets based on uniform distribution. Then one subset is chosen as the testing dataset, while the remaining $k-1$ subsets are used to train the model [119]. This process is repeated k times and each time a different subset is used 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 on either the averaged performance of k subsets or the combined prediction of all samples. Using the cross-validation scheme, 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 the multiple rounds. Through empirical analyses, Kohavi et al. indicated that the tenfold cross validation is the best choice in many practical situations [120].

Bootstrap Validation In cross-validation there are no duplicate samples 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 [121]. In cross-validation, sampling is based on the uniform distribution; thus, it assumes that the data distribution of training data and testing data are the 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 testing data are not the same but are approximately similar; the training samples follow 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 samples, 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; a prediction model is learned for each bootstrap sample, and the model is evaluated using both the original data and the corresponding OOB data. The final prediction error will be a combination of the training error and the testing error. This approach guarantees the stability of the performance estimate of the bootstrap validation.

10.5.2.2 External Validation Methods

In clinical data analysis, external validation methods are also used to validate whether the learned model can be generalized to other scenarios and other patients [122]. 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, wherein the training data and testing data are separated not based on the random sampling but on the geographical location from where the data was collected. Once the prediction model has been learned from a local hospital, it will be interesting to see whether it can be viewed as a generalized model and if it will be applicable at other facilities and locations; thus, the geographic validation is needed. In general, if the model performance is similar, the larger the difference between the training and testing dataset, the more general the model is.

10.6 Conclusion

In this chapter, we reviewed some of the basic and advanced supervised learning methods that have been used for clinical prediction. Some of the widely used basic statistical methods include: (i) linear regression that is used to estimate a continuous outcome; (ii) logistic regression that is a linear binary classification method; (iii) decision trees that are more suitable for categorical inputs and outcomes; and (iv) survival models that are specifically designed for survival analysis. In addition, we also provided a few state-of-the art extensions for some of these basic models. These extensions include: (i) methods for handling sparse data and high-dimensional problems; (ii) kernel tricks to effectively handle nonlinear data distributions; (iii) ensemble approaches to improve the performance of the base models; and (iv) cost-sensitive learning methods to handle imbalanced data. By going through this chapter, we hope that the readers can get a general understanding of different models and pointers to articles that provide more details about effectively using prediction models in clinical medicine. In addition, we also discussed some of the popular evaluation metrics and validation schemes used for estimating the accuracy and utility of the prediction models when applied to healthcare applications.

Acknowledgments

This work was supported in part by National Science Foundation grant IIS-1231742 and the National Institutes of Health grant R21CA175974.

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, M.G.M. Hunink, and J.D.F. 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*, 9: 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*, 1(3): 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] Stephan Dreiseitl and Lucila Ohno-Machado. Logistic regression and artificial neural network classification models: A methodology review. *Journal of Biomedical Informatics*, 35(5):352–359, 2002.

[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*. Athena Scientific, 1999.

[20] Stephen M. Goldfeld, Richard E. Quandt, and Hale F. Trotter. Maximization by quadratic hill-climbing. *Econometrica: Journal of the Econometric Society*, 34(3): 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. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 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*, 46(4): 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*. Oxford University Press, 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. *IEEE Transactions on Medical Imaging*, 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. *IEEE Transactions on Medical Imaging*, 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*, 29(2): 119–127, 1980.

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

[38] S.S. Gambhir, C.K. Hoh, M.E. Phelps, I Madar, and J Maddahi. Decision tree sensitivity analysis for cost-effectiveness of FDG-PET in the staging and management of non-small-cell lung carcinoma. *Journal of Nuclear Medicine: Official Publication, Society of Nuclear Medicine*, 37(9):1428–1436, 1996.

[39] William J. Long, John L. Griffith, Harry P. Selker, and Ralph B. D'Agostino. A comparison of logistic regression to decision-tree induction in a medical domain. *Computers and Biomedical Research*, 26(1):74–97, 1993.

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

[41] 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.

[42] 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. pdf, 2010.

[43] Paulo J. Lisboa and Azzam F.G. Taktak. The use of artificial neural networks in decision support in cancer: A systematic review. *Neural Networks*, 19(4):408–415, 2006.

[44] Paulo J.G. Lisboa. A review of evidence of health benefit from artificial neural networks in medical intervention. *Neural Networks*, 15(1):11–39, 2002.

[45] Maciej A. Mazurowski, Piotr A. Habas, Jacek M. Zurada, Joseph Y. Lo, Jay A. Baker, and Georgia D. Tourassi. Training neural network classifiers for medical decision making: The effects of imbalanced datasets on classification performance. *Neural Networks*, 21(2):427–436, 2008.

[46] 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.

[47] Peter Turney. Types of cost in inductive concept learning. WCSL at ICML-2000. Stanford University, California.

[48] 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.

[49] Charles Elkan. The foundations of cost-sensitive learning. In *International Joint Conference on Artificial Intelligence*, volume 17, pages 973–978. Citeseer, 2001.

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

[51] Pang-Ning Tan et al. *Introduction to Data Mining*. Pearson Education, 2005.

[52] Yusuf Artan, Masoom A. Haider, Deanna L. Langer, Theodorus H. van der Kwast, Andrew J. Evans, Yongyi Yang, Miles N. Wernick, John Trachtenberg, and Imam Samil Yetik. Prostate cancer localization with multispectral MRI using cost-sensitive support vector machines and conditional random fields. *IEEE Transactions on Image Processing*, 19(9):2444–2455, 2010.

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

[54] 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.

[55] Glenn Fung, Murat Dundar, Balaji Krishnapuram, and R. Bharat Rao. Multiple instance learning for computer aided diagnosis. *Advances in Neural Information Processing Systems*, 19:425, 2007.

[56] 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.

[57] Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*, volume 1. Cambridge University Press, 1998.

[58] Yufan Zhao, Michael R Kosorok, and Donglin Zeng. Reinforcement learning design for cancer clinical trials. *Statistics in Medicine*, 28(26):3294–3315, 2009.

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

[60] Trevor Hastie, Robert Tibshirani, and J. Jerome H. Friedman. *The Elements of Statistical Learning*, volume 1. Springer New York, 2001.

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

[62] 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.

[63] 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. *IEEE Transactions on Signal Processing*, 45(11):2758–2765, 1997.

[64] Shi Yu, Tillmann Falck, Anneleen Daemen, Leon-Charles Tranchevent, Johan A.K. Suykens, Bart De Moor, and Yves Moreau. L2-norm multiple kernel learning and its application to biomedical data fusion. *BMC Bioinformatics*, 11(1):309, 2010.

[65] 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.

[66] John P. Klein and Mei-Jie Zhang. *Survival Analysis Software*. Wiley Online Library, 2005.

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

[68] Ettore Marubini and Maria Grazia Valsecchi. *Analysing Survival Data from Clinical Trials and Observational Studies*. John Wiley & Sons, 414 pages ISBN 0–971-93987-0.

[69] Elisa T. Lee and John Wang. *Statistical Methods for Survival Data Analysis*, volume 476. Wiley.com, 2003.

[70] Olive Jean Dunn and Virginia A. Clark. *Basic Statistics: A Primer for the Biomedical Sciences*. Wiley.com, 2009.

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

[72] 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.

[73] 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.

[74] 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.

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

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

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

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

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

[80] Robert Tibshirani et al. The LASSO method for variable selection in the Cox model. *Statistics in Medicine*, 16(4):385–395, 1997.

[81] 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.

[82] 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.

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

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

[85] Bhanukiran Vinzamuri and Chandan K Reddy. Cox regression with correlation based regularization for electronic health records. In *IEEE 13th International Conference on Data Mining (ICDM)*, pages 757–766. IEEE, 2013.

[86] 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.

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

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

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

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

[91] 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.

[92] A. Ciampi, C. H. Chang, S. Hogg, and S. McKinney. Recursive partition: A versatile method for exploratory-data analysis in biostatistics. In *Biostatistics*, 38: 23–50. Springer, 1986.

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

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

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

[96] Hemant Ishwaran, Udaya B. Kogalur, Eugene H. Blackstone, and Michael S. Lauer. Random survival forests. *The Annals of Applied Statistics*, 2(3): 841–860, 2008.

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

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

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

[100] R. G. D. Steel and J. H. Torrie. *Principles and Procedures of Statistics with Special Reference to the Biological Sciences*. McGraw-Hill, 484 pages, I960.

[101] 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.

[102] ISO 35341:2006. *Statistics Vocabulary and Symbols Part 1: General Statistical Terms and Terms Used in Probability*. Geneva, Switzerland: ISO, 2006.

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

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

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

[106] 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.

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

[108] Robert H. Fletcher, Suzanne W. Fletcher, Grant S. Fletcher et al. *Clinical Epidemiology: The Essentials*. Lippincott Williams & Wilkins, 2012.

[109] Gerard Salton and Michael J. McGill. *Introduction to Modern Information Retrieval*. McGraw-Hill, 1986.

[110] Tom Fawcett. An introduction to ROC analysis. *Pattern Recognition Letters*, 27(8):861–874, 2006.

[111] 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.

[112] 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.

[113] 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.

[114] 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.

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

[116] 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.

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

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

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

[120] 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.

[121] Bradley Efron and Robert Tibshirani. *An Introduction to the Bootstrap*, volume 57. CRC Press, 1993.

[122] 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.