ABSTRACT

ZHU, RUI. Bayesian Semi-supervised Learning with Application to ROC Surface Estimation. (Under the direction of Subhashis Ghoshal).

Semi-supervised learning is a classification method which makes use of both labeled data and unlabeled data for training. Since labeling data can be expensive and time consuming, and requires human labor and expertise, semi-supervised learning is becoming more and more popular in recent years.

Semi-supervised learning can be classified to two big classes – generative methods and discriminative methods. We focus on generative methods in this thesis. We first consider one specific application of semi-supervised learning – the Receiver Operating Characteristic (ROC) surface estimation under verification bias. The ROC surface, as a generalization of the ROC curve, has been widely used to assess the accuracy of a diagnostic test for three categories. It plots the True Class Fractions (TCFs) in three axes respectively, and thus illustrates the trade-off among the three TCFs as the cut-off points vary. A problem in the data that complicates the ROC surface estimation is verification bias, referring to the situation where not all subjects have their true classes verified. This is a common problem in disease diagnosis since the gold standard test to get labels, i.e., the true disease status, can be invasive and expensive. The same situation happens in the evaluation of semi-supervised learning, where the unlabeled data are incorporated.

Estimating the ROC surface under verification bias can be considered as an application of semi-supervised learning since both of them want to study the underlying distributions when true labels are partially known. Two methods are proposed for solving this problem. We first propose a semi-parametric Bayesian method based on continuous data under a semi-parametric trinormality assumption in Chapter 2. That is, we assume that the data will follow three normal distributions under the same transformation. To address missing

labels, we impose a general missing at random assumption for verification process. The posterior distribution is then computed using a rank-based likelihood and the consistency of the posterior under a mild condition is established. The method can also be extended to situations without verification bias.

In Chapter 3, we adopt a Bayesian nonparametric method by directly modeling the underlying distributions of the three categories by Dirichlet Process mixture priors. We propose a robust computing algorithm by only imposing a missing at random assumption for missingness but no assumption on the distributions. The method can also accommodate covariates information in estimating the ROC surface, which can lead to a more comprehensive understanding of the diagnostic accuracy. It can be adapted and hugely simplified in the case where there is no verification bias, and very fast computation is possible through the Bayesian bootstrap process. Both methods are compared with other commonly used methods by extensive simulations. We find that the proposed methods generally outperforms other approaches. We also applied the methods to two real datasets, the key findings are as follows: (i) HE4 has a slightly better diagnosis ability compared to CA125 in discriminating healthy, early stage and late stage patients of epithelial ovarian cancer. (ii) Serum albumin has a prognostic ability in distinguishing different stages of hepatocellular carcinoma.

In Chapter 4, we propose a general generative semi-supervised learning algorithm. Like Chapter 2, we assume that the observations will follow two multivariate normal distributions depending on their true labels after the same unknown transformation. To estimate the transformation, we use B-splines on each component of the transformation. The function estimation is reduced to parameter estimation problem which we then can impose Gaussian priors. The posterior distributions can be calculated based on our assumptions and we will use Gibbs sampling framework to update the parameters. The proposed method is then compared with several other state-of-art methods in extensive simulation studies

and real data application. We are able to illustrate that it has better prediction accuracy for a wide variety of cases.

Bayesian Semi-supervised Learning with Application to ROC Surface Estimation

by
Rui Zhu

A dissertation submitted to the Graduate Faculty of
North Carolina State University
in partial fulfillment of the
requirements for the Degree of
Doctor of Philosophy

Statistics

Raleigh, North Carolina
2020

APPROVED BY:

Wenbin Lu	Brian Reich
Ryan Martin	Subhashis Ghoshal Chair of Advisory Committee

ACKNOWLEDGEMENTS

I would like to express my most sincere gratitude towards my advisor, Dr. Subhashis Ghoshal. I have been working with him for my whole PhD career. He is always so supportive by providing me with great ideas and right directions. He managed to meet with me every week and helped me out with every single problem I encountered during my research. Without him, there is no way I can get two manuscripts published before graduation and successfully finished three projects ahead of time. I have to say I am so fortunate to have him as my PhD advisor.

I would also like to thank Dr. Wenbin Lu, Dr. Brian Reich, and Dr. Ryan Martin, for being my committee members. I am so honored to have them as my committee members and I really appreciate their time and their valuable suggestion on my prelim tests and my thesis.

I am so proud to have spent four years and a half in this fantastic department. I owe gratitude to all faculties, stuffs and fellow students, for making me a better person.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER

1

INTRODUCTION

Semi-supervised learning is a classification method which makes use of both labeled data and unlabeled data for training. There has been a growing interest in semi-supervised learning in recent years. Traditional classification methods are supervised in nature. They use only labeled data for training. To train a traditional classifier, we have to obtain the true class information for each unit in the dataset, along with the measurements associated with it. However, labels are often difficult to obtain. It can be expensive and time consuming, and it requires human labor and expertise. On the other hand, unlabeled data in most cases are substantial and easy to obtain. Unsupervised learning methods like clustering provide a way to make use of unlabeled data. However, it may not be appropriate or useful in a classification problem. Supervised learning seems to be the best solution for these cases.

When a few units in a dataset have labels, then semi-supervised learning technique can use these large number of unlabeled units in classification instead of discarding them; thus greatly improves the performance of the classifier. In summary, semi-supervised learning lies in between supervised learning and unsupervised learning and the goal is to make most efficient use with a dataset containing both labeled and unlabeled observations.

Semi-supervised learning can be classified to two big classes – generative methods and discriminative methods. Generative methods are mainly based on Expectation-Maximization (EM) algorithm (Dempster et al. 1977) and they have to make some assumptions on the underlying distributions of different classes. Discriminative methods, on the other hand, only learn the boundary between classes. In the recent literature, discriminative methods are explored much more compared to generative methods. Self-Training (Yarowsky 1995) is the most widely-used semi-supervised learning method. The basic idea is that the chosen classifier teaches itself the labels and then learn iteratively. Co-Training (Blum and Mitchell 1998) is conducted by splitting the features to two subsets. The idea is that each subsets can learn and teach the other some labels they are confident of. Another commonly used algorithm is Semi-Supervised Support Vector Machines(S^3VM). The goal of S^3VM is to find a linear boundary which has the maximum margin for both labeled and unlabeled data. This is an NP-hard problem and there are many algorithms being proposed to solve this problem. The method S^3VM is based on the assumption that the boundary will not cut in dense regions. Other methods like Gaussian Process (Lawrence and Jordan 2005) also make use of this assumption. There are also a huge number of methods based on graphical analysis, like Balcan et al. (2009); Zemel and Carreira-Perpiñán (2005); Zhang and Lee (2007); Hein and Maier (2007) etc. A thorough introduction of semi-supervised learning methods can be found in Zhu (2005).

In this thesis, we will focus on generative methods to solve semi-supervised learning problems. This semi-supervised learning classification algorithm will be proposed in Chap-

ter 4. Before considering a general semi-supervised learning algorithm, we will first consider one specific application of semi-supervised learning-the Receiver Operating Characteristic (ROC) surface estimation under verification bias, in Chapter 2 and 3.

So what is ROC surface? An ROC surface is a natural generalization of an ROC curve which is intended to evaluate the classifier for three-class classification problem (Scurfield 1996). It plots the True Class Fractions (TCFs) in three axes respectively, and thus illustrates the trade-off among the three TCFs as the cut-off points vary. Analogous to the AUC for ROC curve, the volume under the surface (VUS) is proposed as a measure of the accuracy for three-dimensional classifiers, which can be calculated by integrating out the ROC surface.

One common problem in the data that complicates the ROC surface estimation is called verification bias. In reality, the true class each individual belongs to may not be completely known to us. In biomedical settings, for example, we want to estimate the accuracy of a biomarker in distinguishing healthy patients, early phase patients and late phase patients for certain disease. We can use ROC surface for this purpose. Here the class may be the true disease status of the patient, which is usually verified through a gold standard test, the perfect existing diagnostic test. However, such a test may be expensive and invasive. It is more ethical to apply a gold standard test only to those high risk subjects according to the screening test. This leads to the problem of missing data. Because patients who are marked as low-risk are more likely to have their true disease status missing, simply ignoring this missingness and estimating the ROC surface using only subjects with verified disease status may lead to biased results, and will cause a loss of efficacy (quantitative measure of the effectiveness of the test) as well. The same problem can happen when evaluating a classification algorithm. Since labeled data are hard to obtain, we can face the situation when we do not have enough labeled data to evaluate an algorithm. We definitely would like to use unlabeled data as well if possible to evaluate the algorithm in these cases.

Estimating the ROC surface under verification bias can be considered as an application

of semi-supervised learning since both of them want to study the underlying distributions when true labels are partially known. We choose to start with this particular application first since the data for estimating ROC surface is univariate, which is easier to handle even though three classes are in present instead of two. We consider two different approaches to solve the problem of estimating the ROC surface. In Chapter 2, we propose a semi-parametric Bayesian method based on continuous data under a semi-parametric trinormality assumption. That is, we assume that under some unknown transformation, the data from three categories follow three different normal distributions. The posterior distribution is computed using a rank-based likelihood and the consistency of the posterior under a mild condition is established. In Chapter 3, we consider a Bayesian nonparametric method by directly modeling the underlying distributions of the three categories by Dirichlet Process mixture priors. With no assumption on the distributions, this method is even more robust. Moreover, this method can accommodate covariates information in estimating the ROC surface as well, which can lead to a more comprehensive understanding of the diagnostic accuracy. Both methods can be adapted to the case where there is no verification bias. For the second method, it can be adapted and hugely simplified in the case where there is no verification bias, and very fast computation is possible through the Bayesian bootstrap process. Both methods are compared with other commonly used methods by extensive simulations. We find that the proposed methods generally outperforms other approaches. Applying the methods to two real datasets, the key findings are as follows: (i) HE4 has a slightly better diagnosis ability compared to CA125 in discriminating healthy, early stage and late stage patients of epithelial ovarian cancer. (ii) Serum albumin has a prognostic ability in distinguishing different stages of hepatocellular carcinoma.

In Chapter 4, we propose a general generative semi-supervised learning algorithm. Unlike most generative methods, which are mainly based on Expectation-Maximization (EM) algorithm, we use Bayesian methods under more general assumptions on the distributions.

Like in Chapter 2, we assume that the observations will follow two multivariate normal distributions depending on their true labels after the same unknown transformation. Instead of getting rid of the unknown transformation using a rank likelihood, here we have to explicitly put a prior on the transformation and average out with respect to its posterior distribution. To do that, we use B-splines on each component of the transformation. The function estimation is then reduced to the problem of estimation which we can impose Gaussian priors with an ordering constraint. The posterior distributions can be calculated using the Gibbs sampling framework to update the parameters. The proposed method is then compared with several other state-of-art methods in extensive simulation studies and real data application. We find that our proposed method has better prediction accuracy for a wide variety of cases.

CHAPTER

2

BAYESIAN SEMIPARAMETRIC ESTIMATION OF ROC SURFACE UNDER VERIFICATION BIAS

2.1 Introduction

The Receiver Operating Characteristic (ROC) curve analysis has been widely used as an effective tool in measuring the accuracy of diagnostic tests in the two-class classification problem. It shows a tradeoff between sensitivity and specificity by varying the cut-off point

through all possible values of the diagnostic marker. Recently some surfaces related to the ROC curve have been proposed for practical use (Martos and de Carvalho 2018; de Carvalho et al. 2013). One of the most commonly used surface among them is called ROC surface. The ROC surface is a generalization of the ROC curve for classification problems of three outcomes. Let X, Y and Z be continuous measurements from three different classes, $X \sim F_0$ are measurements from Class 0, $Y \sim F_1$ are measurements from Class 1, and $Z \sim F_2$ are measurements from Class 2. Suppose that the ordering of interest for these three classes is $X < Y < Z$. A decision rule that classifies subjects can be defined by using two ordered threshold points $c_1 < c_2$, i.e. choose Class 0 when a measurement is less than c_1, choose Class 1 when it is between c_1 and c_2, and choose Class 2 otherwise. This will result in three True Class Fractions (TCFs):

$$\begin{aligned}
\mathrm{TCF}_0 &= \mathrm{P}(X \le c_1) = F_0(c_1), \\
\mathrm{TCF}_1 &= \mathrm{P}(c_1 \le Y \le c_2) = F_1(c_2) - F_1(c_1), \\
\mathrm{TCF}_2 &= \mathrm{P}(Z > c_2) = 1 - F_2(c_2).
\end{aligned}$$

Varying c_1 and c_2 will give us a set of TCFs. To construct the ROC surface, we plot (TCF_0, TCF_1, TCF_2) in a three-dimensional coordinate system. The functional form of the ROC surface can be obtained by expressing TCF_1 as a function of (TCF_0, TCF_2) given by (Nakas and Yiannoutsos 2004)

$$\mathrm{ROC}_s(\mathrm{TCF}_0, \mathrm{TCF}_2) = F_1(F_2^{-1}(1 - \mathrm{TCF}_2) - F_1(F_0^{-1}(\mathrm{TCF}_0))). \tag{2.1}$$

The Volume under the ROC Surface (VUS) was proposed as an important index for the

assessment of the diagnostic accuracy. The VUS is given by

$$\mathrm{VUS} = \int_0^1 \int_0^{1-F_2(F_0^{-1}(p_0))} \mathrm{ROC}_s(p_0, p_2) d\, p_2 d\, p_0;$$

see Nakas and Yiannoutsos (2004). This can be shown to be equal to $\mathrm{P}(X < Y < Z)$ (Mossman 1999).

The ROC surface has been used in diagnostic testing in medical sciences when the disease has two phases, for example, the early phase and late phase of a progressive disease. The symptoms in the early phase may be mild and ignorable, while the late phase tends to have more severe symptoms. Clearly, there is an inherent ordering between healthy, early phase diseased and late phase diseased. One example is Alzheimer's disease, which can be graded low, intermediate and high according to the progress of the disease (Chi and Zhou 2008). Different medical treatments should be applied to different phases. The treatment for the late phase of the disease can be expensive and invasive to patients, even requiring surgeries, while the treatment for the early phase can be conservative. This necessitates the identification of the two phases of the disease and thus leads to the consideration of three classes. We assume, without loss of generality, that a higher test value indicates a higher level of disease.

Many methods have been proposed for estimating the ROC surface. The empirical estimator of ROC surface was obtained by simply replacing the cumulative distribution functions in (2.3) by their empirical estimators (Li and Zhou 2009), and VUS can be calculated by integrating out this ROC estimator. Another empirical estimator was given by the unbiased nonparametric Mann-Whitney U statistic of the probability $\mathrm{P}(X < Y < Z)$ (Dreiseitl et al. 2000), which was later extended to the case with ties (Nakas and Yiannoutsos 2004). Kang and Tian (2013) proposed using Gaussian kernel approaches for estimation of distribution functions. A non-parametric Bayesian method of the ROC surface estimation

based on Finite Pólya Tree prior distributions was proposed by Inácio et al. (2011).

A parametric method of estimating the ROC surface is based on the parametric trinormality assumption: $X \sim \mathrm{N}(\mu_0, \sigma_0^2)$, $Y \sim \mathrm{N}(\mu_1, \sigma_1^2)$, $Z \sim \mathrm{N}(\mu_2, \sigma_2^2)$, where $\mathrm{N}(\mu, \sigma^2)$ denotes the normal distribution with mean μ and variance σ^2. Xiong et al. (2006) obtained a closed form expression of the ROC surface under this assumption. If the data are not normally distributed, Kang and Tian (2013) proposed using the Box-Cox transform. Li and Zhou (2009) introduced two semi-parametric estimators of the ROC surface by extending the methods of Hsieh and Turnbull (1996) and Nze Ossima et al. (2015) for estimating the ROC curve.

In this chapter, we propose a new semiparametric method for estimating the ROC surface. This is a generalization of a Bayesian method for estimating the ROC curve using a rank-based likelihood (BRL) introduced by Gu and Ghosal (2009). We assume that, under some unknown strictly monotone increasing transformation, the measurements follow three different normal distributions. Since ranks are invariant under a strictly monotone increasing transformation, exploiting the rank-likelihood eliminates the need for estimating the unknown transformation, and enable us to construct a Bayesian estimator of ROC surface.

We shall also consider the possibility of verification bias in the data which widely occurs in practice but is rarely considered in the existing literature. That is to say, the true labels of the subjects are partially missing. As we introduced in Chapter 1, this is a common problem in the biomedical setting, where the label of a subject refers to the true disease status. Since label is verified only through a very accurate existing diagnostic test called a gold standard test, which is usually expensive and can even be invasive, the common practice is to apply it only on high-risk subjects identified through a screening test. Because patients who are marked as low-risk are more likely to have their true disease status missing, simply ignoring this missingness and estimating the ROC surface using only subjects with verified disease

status may lead to biased results, and will cause a loss of efficacy as well.

So to deal with the missing labels, the commonly used missing at random (MAR) assumption (Little and Rubin 2014) will be adopted. The assumption means that the probability of a subject being verified does not depend on the disease status given the observed measurements. This is reasonable in biomedical contexts since the decision to obtain the gold standard test is generally made by looking at the diagnostic test results and other external factors, while the effect of the true disease status is already incorporated through its influence in diagnostic tests.

The existing literature is very limited for this problem. Chi and Zhou (2008) obtained the maximum likelihood estimator for the ROC surface and the VUS for ordinal diagnostic tests. To Duc et al. (2016) considered this problem for continuous diagnostic tests. They proposed several bias-corrected estimators of TCFs and thus constructed several bias-corrected ROC surfaces. These methods are extensions of Full Imputation (FI), Mean Score Imputation (MSI), Inverse Probability Weighted (IPW), Semi-Parametric Efficient (SPE) estimators for the ROC curves in Alonzo and Pepe (2005). They chose a suitable parametric model to compute the probability of each individual belonging to each class based on verified subjects, and then applied this model to unverified subjects. They also used a suitable parametric model to compute the probability of verification. These probabilities are used to adjust for the influence caused by missing labels.

Through some modifications, our method for estimating the ROC surface can be extended to the setting under verification bias. This can also be regarded as a generalization of the ROC curve estimation under verification bias proposed by Gu et al. (2014). Covariates are not considered in our formulation. If additional covariates information is available, it may be used to model disease prevalence rates $(\lambda_0, \lambda_1, \lambda_2)$ of the three levels, by multicategory logistic regression, for instance. This will make the classification probabilities (2.10) dependent on the covariates, and should lead to more efficient labeling. However, we do

not pursue this proposed method in this Chapter.

The chapter is organized as follows. The methodology is described in Section 2.2 and 2.3. We first consider the setting without verification bias and then consider the case under verification bias. A result on consistency of the posterior distribution obtained from the rank likelihood is presented in Section 2.4. Extensive simulation studies are conducted in Section 2.6. The proposed Bayesian rank likelihood method will be applied to a real data set in the next chapter after introducing another method of estimating ROC surface under verification bias. The proof of the posterior consistency result is given in the appendix. Because the methods in Chapter 3 are proposed for the same purpose, we will apply the two methods along with other methods on a real data in Chapter 3.

2.2 Method for fully verified data

We first consider the setting without verification bias. The diagnostic measurements for all N subjects in the study is denoted by $\boldsymbol{S}$=$(S_1,\ldots,S_N)$ and their true disease status are denoted by $\boldsymbol{D}=(D_1,\ldots,D_N)$, where $D_i=0$ means healthy, $D_i=1$ means level-1 diseased group and $D_i=2$ means level-2 diseased group. As we observe the true disease status, the ith subject can be labeled simply as $L_i=D_i$, $i=1,\ldots,N$. Let $\boldsymbol{L}=(L_1,\ldots,L_N)$. So we have $n_0=\sum\mathbb{1}\{D_i=0\}$ observations from healthy group, $n_1=\sum\mathbb{1}\{D_i=1\}$ observations from level-1 disease group and $n_2=\sum\mathbb{1}\{D_i=2\}$ observations from level-2 diseased group.

Let F_0, F_1 and F_2 be the underlying continuous cumulative distribution functions of the diagnostic measurements for healthy group, level-1 diseased group and level-2 diseased group respectively, so that

$$S_i|\{D_i=k\}\sim F_k \quad ,k=0,1,2, \quad i=1,\ldots N.$$

Based on the trinomality assumption, under some strictly monotone increasing transformation H, the transformed observations $Q_i = H(S_i)$, $i = 1, \ldots, N$, satisfy

$$Q_i | \{D_i = k\} \overset{\text{i.i.d.}}{\sim} \mathrm{N}(\mu_k, \sigma_k^2), \quad k = 0, 1, 2, \tag{2.2}$$

for some $\mu_0 < \mu_1 < \mu_2$ and $\sigma_0, \sigma_1, \sigma_2 > 0$. To ensure the identifiability of the model, the distribution of the middle group (without loss of generality) has been set to be the standard normal (i.e. $\mu_1 = 0$, $\sigma_1 = 1$). Nevertheless, to unify certain formulas for different groups, we shall occasionally use the notations (μ_k, σ_k), $k = 0, 1, 2$, with the interpretation that $\mu_1 = 0$ and $\sigma_1 = 1$. Note that since H is unknown, $\boldsymbol{Q} = H(\boldsymbol{S})$ is unobservable.

Since the ROC function remains unchanged under monotonic transformations, it is immediate that the ROC function in the semiparametric trinormal model coincides with that of the parametric trinormal model, admitting the explicit functional form (Xiong et al. 2006)

$$R(x, y) = \left[\Phi\Big(\frac{\Phi^{-1}(1-y) + d}{c} \Big) - \Phi\Big(\frac{\Phi^{-1}(x) + b}{a} \Big) \right]_+, \tag{2.3}$$

where $\Phi(\cdot)$ denote the cumulative distribution function of $\mathrm{N}(0, 1)$, the plus sign stands for the positive part and

$$a = 1/\sigma_0, \quad b = \mu_0/\sigma_0, \quad c = 1/\sigma_2, \quad d = \mu_2/\sigma_2. \tag{2.4}$$

The volume under the ROC surface (VUS) is given by

$$\mathrm{VUS} = \int_{-\infty}^{\infty} \Phi(as - b)\Phi(-cs + d)\phi(s)ds, \tag{2.5}$$

where $\phi(\cdot)$ denotes the standard normal density function (Xiong et al. 2006). Thus inference on the ROC surface and the VUS can be made through that on (a, b, c, d), or equivalently

on $(\mu_0, \sigma_0, \mu_2, \sigma_2)$. This is especially convenient in the Bayesian setting since Markov chain Monte Carlo (MCMC) algorithms can easily sample from the posterior distribution of $(\mu_0, \sigma_0, \mu_2, \sigma_2)$ through a data augmentation technique, to be below.

It is hard to get an assessment of $(\mu_0, \sigma_0, \mu_2, \sigma_2)$ beforehand as there are no direct observations from the corresponding normal populations. Following Gu et al (2014), we consider the standard improper prior density $\pi(\mu_0, \sigma_0, \mu_2, \sigma_2) \propto \sigma_0^{-1}\sigma_2^{-1}$ for the locational-scale parameters but restrict to the region $\mu_0 < 0 < \mu_2$. As the effect of the unknown transform H has been eliminated, we do not need a prior distribution on H.

Since H is a strictly monotone increasing transformation, $\boldsymbol{R}(\boldsymbol{Q})$, the ranks of $\boldsymbol{Q}$ preserve those of $\boldsymbol{S}$. The rank-likelihood is therefore given by the probability that $\boldsymbol{Q}$ maintains these observed ranking, as a function of the parameters $(\mu_0, \sigma_0, \mu_2, \sigma_2)$. The numerical evaluation of the likelihood is expensive because it involves computing the probability of a cone defined by ordering in a high dimensional normal distribution. On the other hand, the simple restriction that the ith smallest entry $\tilde{Q}_i$ of $\boldsymbol{Q}$ lies between $\tilde{Q}_{i-1}$ and $\tilde{Q}_{i+1}$ allows sampling the latent variables $\boldsymbol{Q}$ in a Gibbs sampling framework, making the Bayesian computation much more feasible than obtaining the maximum likelihood estimates. With the knowledge of the rank vector $\boldsymbol{R}$, clearly $\boldsymbol{Q}$ can be recovered from the ordered values $\tilde{\boldsymbol{Q}} = (\tilde{Q}_1, \ldots, \tilde{Q}_N)$.

Let $\widetilde{\boldsymbol{S}}$ and $\widetilde{\boldsymbol{Q}}$ stand for the order statistics of $\boldsymbol{S}$ and $\boldsymbol{Q}$ respectively. Because H is a strictly monotone increasing function, we have $\widetilde{\boldsymbol{Q}} = H(\widetilde{\boldsymbol{S}})$. In addition, let $\widetilde{\boldsymbol{L}}$ and $\widetilde{\boldsymbol{D}}$ stand for the label and the true disease status corresponding to $\widetilde{\boldsymbol{Q}}$ respectively.

Posterior sampling can be done by Gibbs sampling. To start with, we need to specify the initial value for $\widetilde{\boldsymbol{Q}}$, which must satisfy three conditions: (i) it has to be ordered; (ii) it must have labels according to $\widetilde{\boldsymbol{D}}$; (iii) $Q_i|\{D_i = 1\} \sim \mathrm{N}(0,1), i = 1, \ldots, N$. We consider using the transformed value of observed $\widetilde{\boldsymbol{S}}$ as our initial value for $\widetilde{\boldsymbol{Q}}$. Transformation is needed because of the last condition. We can use a linear transformation if the observed data appears

to be approximately normal, which can be checked using a Q-Q plot. Otherwise, we may try some standard transformations first and check if the transformed data is approximately normal. If this fails as well, we can use the Box-Cox transformation so that the transformed value for the level-1 diseased group will be roughly N(0, 1). We can then generate the initial value for $(\mu_0, \sigma_0, \mu_2, \sigma_2)$ according to the initial value of $\widetilde{\boldsymbol{Q}}$. Specifically,

$$\begin{aligned} \mu_{0,0} &= \sum_{i=1}^{N} \widetilde{Q_i} \mathbb{1}\{\widetilde{D_i} = 0\}/n_0, \quad \sigma_{0,0}^2 = \sum_{i=1}^{N} (\widetilde{Q_i} \mathbb{1}\{\widetilde{D_i} = 0\} - \mu_{0,0})^2/(n_0 - 1), \\ \mu_{2,0} &= \sum_{i=1}^{N} \widetilde{Q_i} \mathbb{1}\{\widetilde{D_i} = 2\}/n_2, \quad \sigma_{2,0}^2 = \sum_{i=1}^{N} (\widetilde{Q_i} \mathbb{1}\{\widetilde{D_i} = 2\} - \mu_{2,0})^2/(n_2 - 1). \end{aligned} \tag{2.6}$$

where $n_0 = \sum_{i=1}^{N} \mathbb{1}\{\widetilde{D_i} = 0\}$, $n_2 = \sum_{i=1}^{N} \mathbb{1}\{\widetilde{D_i} = 2\}$. With those initial values, we shall calculate the ROC surface based on posterior distributions of Gibbs sampling in Algorithm 1.

Algorithm 1 Bayesian Rank Likelihood (BRL) without verification bias.

input : $\widetilde{\boldsymbol{D}}$, initial values $\widetilde{\boldsymbol{Q}}$, $(\mu_{0,0}, \sigma_{0,0}, \mu_{2,0}, \sigma_{2,0})$, (n_0, n_1, n_2), niter
output: $(\boldsymbol{\mu_0}, \boldsymbol{\sigma_0}, \boldsymbol{\mu_2}, \boldsymbol{\sigma_2})$

```
for m ← 1 to niter do
    for i ← 1 to N do
        Q̃_i | {D̃_i = k}, rest ~ TN(μ_{k,m-1}, σ²_{k,m-1}, (Q̃_{i-1}, Q̃_{i+1}))
    end
    Ē_{n_0} = Σ_{i=1}^{N} Q̃_i 1{D̃_i = 0}/n_0
    s_0² = Σ_{i=1}^{N} (Q̃_i 1{D̃_i = 0} − Ē_{n_0})²/(n_0 − 1)
    Ḡ_{n_2} = Σ_{i=1}^{N} Q̃_i 1{D̃_i = 2}/n_2
    s_2² = Σ_{i=2}^{N} (Q̃_i 1{D̃_i = 2} − Ḡ_{n_2})²/(n_2 − 1)
    σ²_{0,m} | rest ~ IG((n_0 − 1)/2, (n_0 − 1)s_0²/2)
    μ_{0,m} | rest ~ TN(Ē_{n_0}, σ²_{0,m}/n_0, (−∞, 0))
    σ²_{2,m} | rest ~ IG((n_2 − 1)/2, (n_2 − 1)s_2²/2)
    μ_{2,m} | rest ~ TN(Ḡ_{n_2}, σ²_{2,m}/n_2, (0, ∞))
end
```

Notice that in the algorithm, the symbol niter is the total number of iterations in the MCMC algorithm, TN stands for truncated normal distribution, IG stands for inverse gamma distribution, and that $\widetilde{Q}_{0,m} = -\infty$, $\widetilde{Q}_{N+1,m} = \infty$, $\mu_{1,m} = 0$, $\sigma_{1,m} = 1$, $m = 0, \ldots,$ niter.

In each iteration, we get $(\mu_{0,m}, \sigma_{0,m}, \mu_{2,m}, \sigma_{2,m})$, $m = 1, \ldots,$ niter. We can calculate (a_m, b_m, c_m, d_m) for each iteration according to (2.4) and VUS_m according to (2.5). Monitoring the convergence through trace plot and discarding all samples for a suitable burn-in period, we can obtain the estimated parametric ROC surface estimates according to the sample mean of the parameters (a, b, c, d). The VUS can be also estimated by averaging out the computed value VUS_m in each MCMC iteration.

2.3 Method in presence of verification bias

Under verification bias, only a fraction of patients go through the gold standard test and have their true disease status D_i observed, $i = 1, \ldots, N$, so $L = (L_1, \ldots, L_N)$ is different from the previous case. If D_i is obtained, i.e., the true disease status is known, the label $L_i = D_i$; otherwise we set $L_i = 3$, indicating the absence of applying a gold standard test. Thus L_i indicates missing status as well as true disease status if the latter is actually observed.

Under verification bias, the number of patients in each disease group is unknown. Assume that, the disease prevalence rates for level-1 and level-2 in the population are λ_1 and λ_2 respectively. So the number of patients in each group follows $(n_0, n_1, n_2) \sim \text{Mult}(N, (\lambda_0, \lambda_1, \lambda_2))$, where Mult stands for the Multinomial distribution, $\lambda_0 = 1 - \lambda_1 - \lambda_2$, $0 < \lambda_1 < 1$, $0 < \lambda_2 < 1$, and $0 < \lambda_1 + \lambda_2 < 1$.

Because of the existence of verification bias, we need to build a model for observing labels, i.e., going through the gold standard test. In a clinical practice, the gold standard test will typically be prescribed according to the screening test results. To be specific, a subject with a higher risk of disease, suggested by a higher score in the diagnostic test, is more

likely to take the gold standard test. Because of this, missing completely at random may not be appropriate. Simply ignoring unverified subjects will lead to a biased estimation. Here we follow Gu et al. (2014) and model this as missing at random. In general, this model of verifying disease status can be represented as

$$\mathrm{P}(L_i \neq 3 | Q_i, D_i) = g(Q_i), \tag{2.7}$$

for a given monotone increasing function g.

There are reasonable missing mechanisms which follow (2.7). In a scheme proposed by Alonzo and Pepe (2005), verification is mandatory for the top p_1 fraction according to the diagnostic test, while it is done with probability p_2 for the remaining subjects, where $0 < p_1, p_2 < 1$ are chosen beforehand. Since the ordering will not change under the monotone transformation H, Qs and Ss share the same ordering, this can be considered as a special case of (2.7) with

$$g(Q) = \begin{cases} 1, & \text{if } Q > Q_{(p_1 N)}, \\ p_2, & \text{if } Q \leq Q_{(p_1 N)}. \end{cases} \tag{2.8}$$

This verification mechanism will be called threshold model later in the simulation part. If a probit regression model is followed for verification,

$$g(Q) = \Phi(\alpha + \beta Q), \tag{2.9}$$

where α, β are known parameters with $\beta > 0$. Notice that the case without verification bias can be regarded as a special case of (2.7) by simply setting $g(Q_i) = 1$.

We adopt a Bayesian approach to estimate $(\lambda_1, \lambda_2, \mu_0, \sigma_0, \mu_2, \sigma_2)$. We again use the improper prior $\pi(\mu_0, \sigma_0, \mu_2, \sigma_2) \propto \sigma_1^{-1}\sigma_2^{-1}$ for $(\mu_1, \sigma_1, \mu_2, \sigma_2)$. For disease prevalence rates, the prior is set to be $(\lambda_0, \lambda_1, \lambda_2) \sim \mathrm{Dir}(\alpha_0, \alpha_1, \alpha_2)$, where $\mathrm{Dir}(\cdot)$ stands for the Dirichlet distribu-

tion, α_0, α_1 and α_2 are chosen according to the mean and standard error from our prior knowledge.

If the ith subject is unverified, its disease status D_i given the latent variable Q_i follows the probability distribution given by Lemma 2.5.1, namely,

$$\mathrm{P}(D_i = k | Q_i = t, L_i = 3) = \lambda_k \phi_{(\mu_k, \sigma_k)}(t) / \Delta(t) = p_k(t), \tag{2.10}$$

say, where $\Delta(t) = \sum_{k=0}^{2} \lambda_k \phi_{(\mu_k, \sigma_k)}(t)$ and $\phi_{(\mu,\sigma)}(\cdot)$ denote the density function of $\mathrm{N}(\mu, \sigma^2)$. A remarkable feature in our modeling using the MAR assumption is that the expression in (2.10) does not depend on the verification function g, allowing us to compute the posterior distribution without actually knowing g. Thus, this method is guarded against misspecification in the verification model, and is more robust compared with other methods which focus on the verification probability.

To obtain samples from the posterior distribution, additionally we need to consider the collection of unobserved status as $\boldsymbol{D}_{\mathrm{un}} = \{D_i : L_i = 3, i \le N\}$ as latent variables and apply Gibbs sampling with these variables augmented with $\boldsymbol{\theta} = (\lambda_1, \lambda_2, \mu_0, \sigma_0, \mu_2, \sigma_2, \widetilde{\boldsymbol{Q}})$. Then given $\boldsymbol{D}_{\mathrm{un}}$, sampling from the posterior distribution of $\boldsymbol{\theta}$ is exactly as in the case without verification bias; when given $\boldsymbol{\theta}$, the posterior sampling of $\boldsymbol{D}_{\mathrm{un}}$ follows from (2.10). For the initial values, first we need to specify the initial values for unobserved D_i's, i.e., $D_i \in \boldsymbol{D}_{\mathrm{un}}$. We sample $\widetilde{D_i} \sim \mathrm{Mult}(1, (\lambda_0, \lambda_1, \lambda_2))$ if $\widetilde{L}_i = 3$, $i = 1, \ldots, N$, where $(\lambda_0, \lambda_1, \lambda_2)$ are sampled from $\mathrm{Dir}(\alpha_0, \alpha_1, \alpha_2)$. The initial values can then be generated exactly as previously introduced. With those initial values, we can obtain the ROC surface as in Algorithm 2.

As in the previous algorithm, we have $\widetilde{Q}_{0,m} = -\infty$, $\widetilde{Q}_{N+1,m} = \infty$, $\mu_{1,m} = 0$, $\sigma_{1,m} = 1$, $m = 0, \ldots,$ niter. We can calculate (a_m, b_m, c_m, d_m) according to $(\mu_{0,m}, \sigma_{0,m}, \mu_{2,m}, \sigma_{2,m})$, $m = 1, \ldots,$ niter for each iteration. Monitoring the convergence through trace plot and discarding all samples for a suitable burn-in period, we obtain the estimated parametric ROC surface

estimates according to the sample mean of the parameters (a, b, c, d). The VUS is also estimated by averaging out the computed value VUS_m in each MCMC iteration.

2.4 Posterior consistency

As the case without verification bias can be regarded as a special case of under verification bias model by simply setting $g(Q_i) = 1$ in (2.7), we only need to show posterior consistency under verification bias.

Let ν denote the Lebesgue measure and π denote the prior density for $(\mu_0, \sigma_0, \mu_2, \sigma_2)$ with respect to ν. Let $(\mu_0^*, \sigma_0^*, \mu_2^*, \sigma_2^*)$ be the true value of $(\mu_0, \sigma_0, \mu_2, \sigma_2)$, that is, the parameter value responsible for the data generating process. The following says that for most values of $(\mu_0^*, \sigma_0^*, \mu_2^*, \sigma_2^*)$, the posterior concentrates around the true value in large samples, providing a theoretical justification of the proposed method from the frequentist point of view.

Theorem 2.4.1 *Assume that (2.2) and (2.7) hold for a monotone increasing function* $g : \mathbb{R} \to (0, 1)$ *and* $\pi(\mu_0, \sigma_0, \mu_2, \sigma_2) > 0$ *a.e.* $[\nu]$ *on* $\mathbb{R}^- \times \mathbb{R}^+ \times \mathbb{R}^+ \times \mathbb{R}^+$. *Then for* $(\mu_0^*, \sigma_0^*, \mu_2^*, \sigma_2^*)$ *a.e.* $[\nu]$, *and any neighborhood* $\mathscr{U}_0$ *of* $(\mu_0^*, \sigma_0^*, \mu_2^*, \sigma_2^*)$ *, we have that*

$$\lim_{N \to \infty} \Pi((\mu_0, \sigma_0, \mu_2, \sigma_2) \in \mathscr{U}_0 | R(\boldsymbol{S}), \boldsymbol{L}) = 1, a.s. \tag{2.11}$$

with respect to $[P^{\infty}_{\mu_0^*, \sigma_0^*, \mu_2^*, \sigma_2^*, g, H}]$, *the true joint distribution of all ranks and labels.*

Since (a, b, c, d) is a continuous reparameterization of $(\mu_0, \sigma_0, \mu_2, \sigma_2)$, it is immediate that the induced posterior of (a, b, c, d) is also consistent for almost all of its true values. Clearly, as the ROC surface and the VUS function depend continuously on (a, b, c, d), these objects are also consistently estimated by the proposed Bayesian method.

We prove Theorem 2.4.1 using a general posterior consistency theorem by Doob. The complete proof is given in Section 2.5 following the formulation of Doob's theorem pre-

sented in Ghosal and van der Vaart (2017). We show that the vector of parameters can be expressed as a function of the observations and then apply Doob's theorem to show posterior consistency.

While Theorem 2.4.1 does not give posterior consistency at all possible true values of the parameter, the assertion is still very attractive as the possible exceptional set where convergence may fail has Lebesgue measure zero, meaning it is a "small set" that "can be ignored". This is a lot more useful than the conclusion generally obtained from Doob's theorem which characterizes the exceptional set as only having measure zero with respect to the prior, and hence is "small" only in a "subjective sense". In contrast, the "objectivity" of the exceptional set in our setting is obtained because we can effectively treat the parameter space as Euclidean (where there is a prior density which can be chosen to be positive throughout the parameter space). This is because the procedure does not depend on the transformation function H and the verification function g, and hence these can be treated as fixed.

2.5 Lemmas and proofs

The following lemma gives the probability distribution of the disease status when unobserved, conditional on the latent measurement Q introduced in the trinormal model.

Lemma 2.5.1 *Let $S_1, ..., S_N$ be the independent diagnostic variables with underlying disease status $D_1, ..., D_N$, where $D_i \in \{0,1,2\}$ for $i = 1, \ldots, N$, $\mathrm{P}(D_i = k) = \lambda_k$, $k = 0,1,2$ where $\lambda_0 + \lambda_1 + \lambda_2 = 1$. Let $S|\{D = k\} \sim f_k$, $k = 0,1,2$, and $L = D$ if D is observed and $L = 3$ otherwise. Assume that verification bias follows MAR assumption as in Eq. 2.7. Then*

$$\mathrm{P}(D_i = k|S_i = t, L_i = 3) = \mathrm{P}(D_i = k|S_i = t, L_i \neq 3) = \frac{\lambda_k f_k(t)}{f(t)}, \quad k = 0,1,2, \quad i = 1, \ldots, N. \tag{2.12}$$

where $f(t) = \lambda_0 f_0(t) + \lambda_1 f_1(t) + \lambda_2 f_2(t)$.

Proof By Bayes's theorem,

$$\mathrm{P}(D_i = k | S_i = t, L_i = 3) == \frac{\lambda_k f_k(t)(1-g(t))}{\sum_{l=0}^{2} \lambda_l f_l(t)(1-g(t))} = \frac{\lambda_k f_k(t)}{\sum_{l=0}^{2} \lambda_l f_l(t)},$$

and $\mathrm{P}(L_i = 3)$ is given by

$$\sum_{k=0}^{2} \int \mathrm{P}(L_i = 3, D_i = k | S_i = t) f_k(t) dt = \sum_{k=0}^{2} \lambda_k \int f_k(t)(1-g(t)) dt.$$

In Chapter 2, $f_k = \phi(\mu_k, \sigma_k^2)$. The following lemma will be needed in the proof of the consistency theorem in 2.4.1.

Lemma 2.5.2 *In the setup of Lemma 2.5.1, the conditional density of* Q *given that* $L \neq 3$ *is given by* $f_Q(t|L \neq 3) = \sum_{k=0}^{2} \lambda_k^* \phi_{(\mu_k, \sigma_k, g)}^*$, *where for* $k = 0, 1, 2$,

$$\lambda_k^* = \frac{\lambda_k \int g(s) \phi_{(\mu_k, \sigma_k)}(s) ds}{\sum_{l=0}^{2} \lambda_l \int g(s) \phi_{(\mu_l, \sigma_l)}(s) ds}, \quad \phi_{(\mu, \sigma, g)}(t) = \frac{g(t) \phi_{(\mu, \sigma)}(t)}{\int g(s) \phi_{(\mu, \sigma)}(s) ds}. \tag{2.13}$$

Proof By an application of Bayes's theorem, we evaluate $f_Q(t|L \neq 3)$ as

$$\begin{aligned}
&\frac{\mathrm{P}(L \neq 3 | Q = t) \sum_{k=0}^{2} \mathrm{P}(D = k) \phi_{(\mu_k, \sigma_k)}(t)}{\int \mathrm{P}(L \neq 3 | Q = s) \sum_{k=0}^{2} \mathrm{P}(D = k) \phi_{(\mu_k, \sigma_k)}(s) ds} \\
&\quad = \frac{g(t) \sum_{k=0}^{2} \lambda_k \phi_{(\mu_1, \sigma_1)}(t)}{\int g(s) \sum_{k=0}^{2} \lambda_k \phi_{(\mu_1, \sigma_1)}(s) ds} \\
&\quad = \sum_{k=0}^{2} \frac{g(t) \phi_{(\mu_k, \sigma_k)}(t)}{\int g(s) \phi_{(\mu_k, \sigma_k)}(s) ds} \times \frac{\lambda_k \int g(s) \phi_{(\mu_k, \sigma_k)}(s) ds}{\sum_{l=0}^{2} \lambda_l \int g(s) \phi_{(\mu_l, \sigma_l)}(s) ds} \\
&\quad = \sum_{k=0}^{2} \lambda_k^* \phi_{(\mu_k, \sigma_k, g)}^*(t).
\end{aligned}$$

Proof of Theorem 2.4.1 Although g and H are unknown, the BRL method does not depend

on them. Hence we can treat g and H as known to prove consistency of the posterior distribution of $(\mu_0, \sigma_0, \mu_2, \sigma_2)$.

We verify the conditions of Doob's Theorem as presented by Ghosal and van der Vaart (2017), Theorem 6.9 and Proposition 6.10. Let the set of all permutations of $\{1, \ldots, N\}$ be denoted by Ω_N. We need to show the existence of a function $h^* : \Omega_1 \times \Omega_2 \times \cdots \times \{0,1,2,3\}^\infty \to \mathbb{R}^- \times \mathbb{R}^+ \times \mathbb{R}^+ \times \mathbb{R}^+$, such that $(\mu_0, \sigma_0, \mu_2, \sigma_2) = h^*(\boldsymbol{R}_N, N \geq 1, (L_1, L_2, \ldots))$ a.s.

Let $1 \leq i_1 < \cdots < i_{N^*} \leq N$ be all indices with $L_{i_j} = 0, 1$ or 2, $j = 1, ..., N^*$. Disregarding the information on the true disease status, by Lemma 2.5.2, the corresponding Q values are generated from the mixture distribution $Q_{i_j} \overset{\text{i.i.d.}}{\sim} \sum_{k=0}^{2} \lambda_k^* \Phi_{(\mu_k, \sigma_k, g)}$ where $\Phi^*_{(\mu,\sigma,g)}$ is the cumulative distribution function of $\phi^*_{(\mu,\sigma,g)}$. Thus we have

$$U_j^* = \sum_{k=0}^{2} \lambda_k^* \Phi_{(\mu_k, \sigma_k, g)}(Q_{i_j}) \overset{\text{i.i.d.}}{\sim} \text{Uniform}(0,1).$$

Let $(R'_{N^*1}, \ldots, R'_{N^*N^*})$ and $(L'_{N^*1}, \ldots, L'_{N^*N^*})$ be the rank vector and labels of $(U_1^*, \ldots, U_{N^*}^*)$ respectively. According to Theorem a on page 157 of Hájek and Šidák (1967), we have

$$\begin{aligned}
\mathrm{E}\left(U_j^* - \frac{R'_{N^*_{i_j}}}{N^*+1}\right)^2 &= \frac{1}{N^*} \sum_{k=1}^{N^*} \mathrm{E}\Big[\Big(U_j^* - \frac{k}{N^*+1}\Big)^2 | R'_{N^*_{i_j}} = k\Big] \\
&= \frac{1}{N^*} \sum_{k=1}^{N^*} \frac{k(N^*-k+1)}{(N^*+1)^2(N^*+2)} < \frac{1}{N^*},
\end{aligned}$$

so $\mathrm{E}\left(U_j^* - \frac{R'_{N^*_{i_j}}}{N^*+1}\right)^2 \to 0$ as $N \to \infty$ because N^*/N converges to a positive limit, there exists a subsequence $\{N_k^*\}$ of $\{N^*\}$ such that for $j \geq 1$, $U_j^* = \lim_{k\to\infty} (N_k^*+1)^{-1} R'_{N^*_{k_{i_j}}}$ a.s. Thus we can write $U_j^* = h_j(\boldsymbol{R}_N, N \geq 1, L_1, L_2, \ldots)$ for some function $h_j : \Omega_1 \times \Omega_2 \times \cdots \times \{0,1,2,3\}^\infty \to [0,1]$.

Now given $\{Q_{i_j} : L_{i_j} = 1\} \overset{\text{i.i.d.}}{\sim} \Phi^*_{(0,1,g)}$, so that $\{U_j^* : L_{i_j} = 1\} \overset{\text{i.i.d.}}{\sim} V_{(\mu_0,\sigma_0,\mu_2,\sigma_2,g)}$, where $V_{(\mu_0,\sigma_0,\mu_2,\sigma_2,g)}$ is the distribution of $\sum_{k=0}^{2} \lambda_k^* \Phi^*_{(\mu_k,\sigma_k,g)}(\xi)$, with $\xi \sim \Phi^*_{(0,1,g)}$. Since $(U_j^* : L_{i_j} = 1)$ are independent and identically distributed samples from it, $V_{(\mu_0,\sigma_0,\mu_2,\sigma_2,g)}$ is consistently es-

timable. Thus we only need to show that the family of $\{V_{(\mu_0,\sigma_0,\mu_2,\sigma_2,g)} : \mu_0 < 0 < \mu_2, \sigma_0 > 0, \sigma_2 > 0\}$ is identifiable, i.e., if $V_{(\mu_0,\sigma_0,\mu_2,\sigma_2,g)} = V_{(\mu_0',\sigma_0',\mu_2',\sigma_2',g)}$, then $(\mu_0,\sigma_0,\mu_2,\sigma_2) = (\mu_0',\sigma_0',\mu_2',\sigma_2')$.

To prove this, first we show that $\Phi^*_{(\mu,\sigma,g)}$ and $\Phi^*_{(\mu',\sigma',g)}$ are linearly independent whenever $(\mu,\sigma) \neq (\mu',\sigma')$. If not, there exists some $c_1, c_2 \in \mathbb{R}$ such that $c_1\Phi^*_{(\mu,\sigma,g)}(t) + c_2\Phi^*_{(\mu',\sigma',g)}(t) = 0$ for all t. By differentiation, we obtain $c_1\phi^*_{(\mu,\sigma,g)}(t) + c_2\phi^*_{(\mu',\sigma',g)}(t) = 0$ for all t. Now as $g(t) \neq 0$ for all t, this leads to $c_1\phi_{(\mu,\sigma)}(t) + c_2\phi_{(\mu',\sigma')}(t)$ for all t in view of the definition of $\phi_{(\mu,\sigma,g)}(t)$. This contradicts the obvious linear independence of $\phi_{(\mu,\sigma)}$ and $\phi_{(\mu',\sigma')}$, two distinct normal densities. The argument easily extends to finitely many distributions. Thus the assertion of identifiability follows because of the restrictions $\mu_0 < 0$ and $\mu_2 > 0$, which allow to separate the roles of the two mixture components.

Therefore we conclude that for some function h of $(U_1, U_2, \ldots, L_1, L_2, \ldots)$, and consequently for a function h^* of all ranks and labels, almsot surely we have

$$\begin{aligned}(\mu_0,\sigma_0,\mu_2,\sigma_2,g) &= h(U_1, U_2, \ldots) \\ &= h(h_1(\boldsymbol{R}_N, N \geq 1, L_1, L_2, \ldots), h_2(\boldsymbol{R}_N, N \geq 1, L_1, L_2, \ldots), \ldots) \\ &= h^*(\boldsymbol{R}_N, N \geq 1, L_1, L_2, \ldots).\end{aligned}$$

This verifies the condition of Doob's theorem and hence posterior consistency holds at almost every $(\mu_0^*, \sigma_0^*, \mu_2^*, \sigma_2^*)$ with respect to the joint prior distribution of these parameters. Since the latter has positive density over the whole parameter space $\mathbb{R}_- \times \mathbb{R}_+ \times \mathbb{R}_+ \times \mathbb{R}_+$, the exceptional set has Lebesgue measure zero as well.

2.6 Simulation

2.6.1 Without verification bias

We first consider applying our method when there is no verification bias in the data. In this simulation, let $n = (n_0, n_1, n_2)$, so we generate n_0, n_1 and n_2 samples from $\mathrm{N}(-1.8, 1.5^2)$, $\mathrm{N}(0,1)$ and $\mathrm{N}(2, 2^2)$ respectively, and then apply the inverse-transform H^{-1} on them to obtain the observable data. Note that by (2.3), the true value of (a, b, c, d) is $(0.667, -1.2, 0.5, 1)$ and the corresponding value of the VUS is 0.671. We consider both balanced cases where $n = (50, 50, 50), (100, 100, 100)$ and an unbalanced case where $n = (100, 40, 20)$. We set $n_0 > n_1 > n_2$ for the unbalanced case since in biomedical settings, there are much more healthy patients than diseased patients. We consider two different true transformations H here — (i) logarithmic: $H(x) = \log x$; (ii) logit transformation: $H(x) = \log(x/(1-x))$.

We compare our method (BRL) with other existing methods. The first method, proposed by Kang and Tian (2013) estimates the unknown transformation H parametrically from the Box-Cox transformation family using the maximum likelihood method (Xiong et al. 2006), and will be referred to as BC in the tables. Note that one of the true transformations we consider, the logarithmic transformation, belongs to the Box-Cox transformation family. We also compare BRL with two other semiparametric methods (denoted here by Semi1 and Semi2) proposed by Li and Zhou (2009), which fit an ROC surface of the stated form under the trinormality assumption to the empirical ROC surface estimate. Finally, we consider comparing BRL to a non-parametric Bayesian surface estimate based on Finite Pólya Tree prior distributions, proposed by Inácio et al. (2011), denoted as Pólya. The simulation results for the bias and the MSE of the VUS estimates, and the average L_1-distances and L_∞-distances between the estimated surfaces and the true ROC surface are shown in Table 2.1. All the estimates are based on 100 simulated data sets. For each data set, the BRL and

Table 2.1: Bias ($\times 10$) and MSE ($\times 10^2$) for VUS estimates, L_1-distances ($\times 10$) and L_∞-distances between the estimated ROC surface and the true ROC surface without verification bias, VUS $= 0.671$.

	True transformation=$\log x$					True transformation=$\log(x/(1-x))$				
	BRL	BC	Semi1	Semi2	Pólya	BRL	BC	Semi1	Semi2	Pólya
	$n=(50,50,50)$									
bias	0.07	−0.16	0.02	−0.21	−0.96	−0.02	−0.43	−0.02	−0.22	−1.31
MSE	0.21	0.27	0.21	0.36	1.36	0.20	0.53	0.22	0.32	1.93
L_1	0.33	0.34	0.46	0.49	1.19	0.32	0.54	0.47	0.48	2.14
L_∞	0.20	0.21	0.30	0.28	0.43	0.21	0.28	0.31	0.28	0.48
	$n=(100,40,20)$									
bias	−0.01	−0.48	−0.33	−0.38	−1.35	0.01	−0.73	−0.26	−0.30	−1.36
MSE	0.26	0.75	0.48	0.44	2.17	0.32	1.11	0.39	0.36	2.10
L_1	0.39	0.63	0.62	0.60	1.38	0.39	0.70	0.59	0.57	2.13
L_∞	0.26	0.37	0.41	0.37	0.44	0.25	0.37	0.38	0.35	0.46
	$n=(100,100,100)$									
bias	−0.01	0.01	−0.07	−0.11	−0.46	−0.05	−0.24	−0.05	−0.15	−1.30
MSE	0.11	0.11	0.12	0.13	0.52	0.17	0.21	0.12	0.16	1.82
L_1	0.24	0.23	0.39	0.40	0.87	0.26	0.45	0.37	0.38	2.19
L_∞	0.15	0.15	0.27	0.25	0.37	0.16	0.24	0.26	0.25	0.49

Pólya methods are calculated with 90000 Gibbs samples (100000 MCMC iterations after 10000 samples used for burn-in). The biases shown in Table 2.1 are multiplied by 10, MSEs are multiplied by 10^2, L_1-distances are multiplied by 10, L_∞-distances are shown in the original scale, for convenience of display. The same applies to the remaining tables.

From the tables above, we can conclude that BRL succeeds in almost every case considered. Relatively low bias and MSE suggest that the method has an accurate estimate of VUS, while low L_1-distance and L_∞-distance suggest that the estimated surface is very close to the true ROC surface. The Box-Cox method also has a good performance when the true transformation is logarithmic, which is expected since this transformation belongs to the Box-Cox family, but not as good for the logit transformation. BRL, on the other hand, is quite robust with respect to the true transformation, since the BRL method eliminates the effect of the transformation through ranks. Semi1 has a relatively good performance in estimating the VUS but not so accurate in estimating the ROC surface. All methods have

more accurate estimates of the ROC surface and the VUS when the number of observations doubled for the balanced case. For the unbalanced case, even though the total number of patients actually increased slightly (from 150 to 160), the performances of most methods are still worse compared with the balanced case when $n = (50, 50, 50)$.

2.6.2 Under the trinormality assumption with verification bias

In this simulation, n is set to be $(100, 100, 100)$ for the balanced case and to $(200, 150, 100)$ for the unbalanced case. We consider two different trinormal distributions settings of $(\mu_0, \sigma_0, \mu_2, \sigma_2)$: (a) $(-1.8, 1.5, 2, 2)$ and (b) $(-3, 2, 2, 1)$, which respectively correspond to VUS values 0.671 and 0.833, and the underlying transformation is chosen to be the logit transformation.

Within each simulation setting, consider two verification mechanisms: (i) threshold model with $p_1 = 0.8, p_2 = 0.4$ in 2.8, and (ii) probit regression model with $\beta = 1, \alpha = 0.106$ in 2.9 for balanced cases; $\beta = 1, \alpha = 0.5$ for the unbalanced cases of (a); $\beta = 1, \alpha = 1$ for the unbalanced cases of (b). Both schemes will generate about half of the missing labels. The BRL estimates are obtained by 90000 Gibbs samples (100000 MCMC iterations after 10000 iterations used for burn-in). In total 100 datasets are simulated for the study.

There are only a few methods available to deal with data with verification bias for estimating the ROC surface. We compare the proposed BRL method with Full Imputation (FI), Mean Score Imputation (MSI), Inverse Probability Weighted (IPW), Semi-Parametric Efficient (SPE) methods proposed by To Duc et al. (2016). We use a multivariate logistic model to estimate the true disease rate and a logistic model to estimate verification rate in order to apply those methods. We compare those methods in terms of the estimated accuracy of the VUS, and also the average L_1-distance and L_∞-distance between the estimated and the true ROC surface in Table 2.2.

Table 2.2: Bias ($\times 10$) and MSE ($\times 10^2$) for VUS estimates, L_1-distances ($\times 10$) and L_∞-distances between the estimated ROC surface and the true ROC surface under verification bias.

	Threshold					Probit				
	BRL	FI	MSI	IPW	SPE	BRL	FI	MSI	IPW	SPE
	$(a,b,c,d)=(0.667,-1.2,0.5,1)$, VUS $=0.671$									
	$n=(100,100,100)$									
bias	0.02	−0.39	−0.03	0.55	0.26	−0.37	−0.93	−0.83	−1.02	−1.07
MSE	0.22	0.52	0.29	0.54	0.32	0.50	1.59	1.45	2.79	2.39
L_1	0.51	1.08	0.73	0.77	0.69	0.66	1.67	1.47	1.37	1.52
L_∞	0.20	0.36	0.29	0.39	0.31	0.22	0.54	0.47	0.47	0.49
	$n=(200,150,100)$									
bias	−0.15	−0.31	−0.08	0.38	0.09	−0.59	−0.73	−0.62	−0.54	−0.69
MSE	0.15	0.44	0.30	0.48	0.43	0.55	0.75	0.60	0.80	0.87
L_1	0.48	1.09	0.73	0.68	0.73	0.78	1.47	1.22	0.87	1.03
L_∞	0.18	0.42	0.32	0.36	0.34	0.24	0.52	0.42	0.34	0.35
	$(a,b,c,d)=(0.5,-1.5,1,2)$, VUS $=0.833$									
	$n=(100,100,100)$									
bias	−0.07	−0.38	−0.20	0.33	0.15	−0.38	−2.27	−2.22	−2.21	−2.17
MSE	0.12	0.40	0.23	0.20	0.14	0.37	6.55	6.36	8.70	6.35
L_1	0.34	0.70	0.50	0.49	0.46	0.51	2.20	2.14	1.96	2.18
L_∞	0.20	0.27	0.24	0.34	0.29	0.23	0.48	0.49	0.54	0.53
	$n=(200,150,100)$									
bias	−0.42	−0.56	−0.32	0.26	0.12	−0.64	−1.69	−1.59	−0.87	−1.39
MSE	0.26	0.49	0.23	0.13	0.09	0.52	3.16	2.80	1.08	2.21
L_1	0.50	0.76	0.49	0.39	0.41	0.69	1.64	1.53	0.85	1.37
L_∞	0.19	0.24	0.20	0.28	0.23	0.24	0.36	0.35	0.34	0.32

Table 2.3: Bias ($\times 10$) and MSE ($\times 10^2$) for VUS estimates, L_1-distances ($\times 10$) and L_∞-distances between the estimated ROC surface and the true ROC surface when departing from the trinormality assumption, n=(200, 200, 200), VUS = 0.358.

	Threshold					Probit				
	BRL	FI	MSI	IPW	SPE	BRL	FI	MSI	IPW	SPE
bias	0.10	0.47	0.25	0.13	−0.12	−0.14	0.46	0.23	−0.01	0.01
MSE	0.04	0.31	0.15	0.16	0.16	0.07	0.28	0.13	0.10	0.10
L_1	0.74	0.61	0.42	0.48	0.45	0.61	0.57	0.38	0.39	0.39
L_∞	0.37	0.16	0.13	0.18	0.19	0.33	0.15	0.11	0.16	0.15

We can see that BRL almost always has the lowest bias and MSE for the VUS estimates, and the lowest L_1 and L_∞-distances. There are a few exceptions though. For example, for the unbalanced case of $(a, b, c, d) = (0.5, -1.5, 1, 2)$ where verification bias is generated using threshold model, IPW has lower bias and MSE for the VUS estimates, and lower L_1-distance estimates as well. MSI estimates of the VUS are quite comparable to BRL in terms of bias and MSE for $(a, b, c, d) = (0.667, -1.2, 0.5, 1)$ setting, but the estimated ROC surfaces have larger distances to the true ROC surface. Overall, BRL has a satisfying performance for all cases considered above.

2.6.3 Departure from the trinormality assumption with verification bias

It is important to study the performance of our method when the data does not satisfy the trinormality assumption. Here we generate $(X_1, \ldots, X_{n_0})$ independently from Beta(3, 5), $(Y_1, \ldots, Y_{n_1})$ independently from Beta(2, 2), and $(Z_1, \ldots, Z_{n_2})$ independently from Beta(5, 3), where $n = (200, 200, 200)$. The corresponding VUS = 0.358 and 100 simulated data sets are used in the study. For the verification bias models, we use the threshold mechanism with $p_1 = 0.8$ and $p_2 = 0.4$, and the probit regression mechanism with $\alpha = 0.01$ and $\beta = 0.07$. Both of them generate approximately half of the missing labels. The BRL estimates are obtained by 90000 Gibbs samples (100000 MCMC iterations after 10000 iterations used for burn-in).

Table 2.4: Bias ($\times 10$) and MSE ($\times 10^2$) for VUS estimates, L_1-distances ($\times 10$) and L_∞-distances between the estimated ROC surface and the true ROC surface when departing from the MAR assumption, n=(200, 200, 200), VUS = 0.671.

	Threshold					Probit				
	BRL	FI	MSI	IPW	SPE	BRL	FI	MSI	IPW	SPE
bias	−1.41	2.09	2.40	2.01	1.90	−1.13	−1.34	−1.07	−1.07	−1.05
MSE	2.13	4.53	5.95	4.34	3.96	1.41	1.94	1.28	1.34	1.28
L_1	0.86	1.44	1.00	1.30	1.57	0.69	1.54	1.58	1.03	1.02
L_∞	0.36	0.40	0.29	0.59	0.67	0.29	0.36	0.29	0.39	0.35

We see from Table 2.3 that although the VUS estimates given by BRL are relatively accurate in terms of bias and MSE, the distances between the estimated surfaces and the true ROC surface are larger comparing to the surfaces estimated by other methods. This makes sense since other methods do not require the trinormality assumption. Still, BRL gives a robust estimate of VUS under departure from trinormality.

2.6.4 Departure from the MAR assumption with verification bias

When the MAR assumption fails, the verification probability is no longer independent of disease status conditional on observed values. Thus the expression we have in (2.7) is no longer valid. If

$$\mathrm{P}(L_i \neq 3 | Q_i = t, D_i = k) = g_k(t),\ k = 0, 1, 2, \tag{2.14}$$

then as in the proof of Lemma 2, it follows that

$$\mathrm{P}(D_i = k | Q_i = t, L_i = 3) = \frac{\lambda_k g_k(t) \phi_{(\mu_k, \sigma_k)}(t)}{\Delta^*(t)},\ k = 0, 1, 2, \tag{2.15}$$

where $\Delta^*(t) = \lambda_0 g_0(t) \phi_{(\mu_1, \sigma_1)}(t) + \lambda_1 g_1(t) \phi(t) + \lambda_2 g_2(t) \phi_{(\mu_2, \sigma_2)}(t)$.

As expected, both the bias and the MSE for the VUS estimates and the distances between the surfaces are larger compared with their counterparts under the MAR mechanism. Under

the threshold verification scheme, BRL has the lowest bias and MSE for the VUS estimates, and the lowest L_1-distance as well. Under the probit verification scheme, MSI, IPW and SPE have lower biases and MSEs for the VUS estimates, but BRL has the best performance in the ROC surface estimates with the lowest L_1 and L_∞-distances. Thus BRL is relatively robust against departure from the MAR assumption.

Algorithm 2 Bayesian Rank Likelihood (BRL) with verification bias.

input : $\widetilde{\boldsymbol{L}}$, initial values $\widetilde{\boldsymbol{Q}}$, $(\mu_{0,0}, \sigma_{0,0}, \mu_{2,0}, \sigma_{2,0})$, $(\alpha_0, \alpha_1, \alpha_2)$, niter

output: $(\boldsymbol{\mu_0}, \boldsymbol{\sigma_0}, \boldsymbol{\mu_2}, \boldsymbol{\sigma_2})$

for $m \leftarrow 1$ **to** niter **do**

  for $i \leftarrow 1$ **to** N **do**

    if $\widetilde{L}_i == 3$ **then**

      for $k \leftarrow 0$ **to** 2 **do**

        $p_k = \frac{\lambda_0 \phi_{\mu_{k,m-1}, \sigma_{k,m-1}}(\widetilde{Q}_i)}{\sum_{l=0}^{2} \lambda_l \phi_{\mu_{l,m-1}, \sigma_{l,m-1}}(\widetilde{Q}_i)}$

      end

      $\widetilde{D}_i \sim \text{Mult}(1, (p_0, p_1, p_2))$

    end

    $\widetilde{Q}_i | \{\widetilde{D}_i = k\}, \text{rest} \sim \text{TN}(\mu_{k,m-1}, \sigma^2_{k,m-1}, (\widetilde{Q}_{i-1}, \widetilde{Q}_{i+1}))$

  end

  $n'_0 = \sum_{i=1}^{N} \mathbb{1}\{\widetilde{D}_i = 0\}$

  $n'_1 = \sum_{i=1}^{N} \mathbb{1}\{\widetilde{D}_i = 1\}$

  $n'_2 = \sum_{i=1}^{N} \mathbb{1}\{\widetilde{D}_i = 2\}$

  $\bar{E}_{n'_0} = \sum_{i=1}^{N} \widetilde{Q}_i \mathbb{1}\{\widetilde{D}_i = 0\} / n'_0$

  $s_0^2 = \sum_{i=1}^{N} (\widetilde{Q}_i \mathbb{1}\{\widetilde{D}_i = 0\} - \bar{E}_{n'_0})^2 / (n'_0 - 1)$

  $\bar{G}_{n'_2} = \sum_{i=1}^{N} \widetilde{Q}_i \mathbb{1}\{\widetilde{D}_i = 2\} / n'_2$

  $s_2^2 = \sum_{i=2}^{N} (\widetilde{Q}_i \mathbb{1}\{\widetilde{D}_i = 2\} - \bar{G}_{n'_2})^2 / (n'_2 - 1)$

  $\sigma^2_{0,m} | \text{rest} \sim \text{IG}((n'_0 - 1)/2, (n'_0 - 1) s_0^2 / 2)$

  $\mu_{0,m} | \text{rest} \sim \text{TN}(\bar{E}_{n'_0}, \sigma^2_{0,m} / n_0, (-\infty, 0))$

  $\sigma^2_{2,m} | \text{rest} \sim \text{IG}((n'_2 - 1)/2, (n'_2 - 1) s_2^2 / 2)$

  $\mu_{2,m} | \text{rest} \sim \text{TN}(\bar{G}_{n_2}, \sigma^2_{2,m} / n_2, (0, \infty))$

  $(\lambda_0, \lambda_1, \lambda_2) | \text{rest} \sim \text{Dir}(\alpha_0 + n'_0, \alpha_1 + n'_1, \alpha_2 + n'_2)$

end

CHAPTER

3

BAYESIAN NONPARAMETRIC ROC SURFACE ESTIMATION UNDER VERIFICATION BIAS

3.1 Introduction

As introduced in the last Chapter, ROC surface is a generalization of the ROC curve, and is used for assessing the accuracy of classifiers on three classes. It plots the True Class Fractions (TCFs) in three axes respectively, and thus illustrates the trade-off among the

three TCFs as the cut-off points vary.

In this chapter, we address the ROC surface and the VUS estimation for continuous measurements under verification bias using a nonparametric Bayesian way by using Dirichlet Process mixture. This nonparamtric method is even more robust comparing to our previous method since we do not impose any assumption on the data distributions. The only assumption we have here is the MAR assumption we proposed previously to address the missingness.

Moreover, our method can easily accommodate covariates in estimating the ROC surface. Covariates information like gender, age, height, weight, ethnicity is usually available under biomedical settings since the patients need to fill out the information when doing a prognostic test. It is absolutely a good idea to include them for assessing the accuracy of a diagnostic test since the distributions for three classes can be different conditional on the covariates. So far, none of the literature have considered incorporating the covariates in estimating the ROC surface in the presence of verification bias. To Duc et al. (2016) used covariates for building parametric models, but not directly on building ROC surface. Our method, on the other hand, can directly accommodate covariates by expressing ROC surface as a function of the covariates, and then get the covariate-adjusted ROC surface by integrating out the covariates. This can lead to a more comprehensive understanding of the diagnostic accuracy.

The rest of the chapter is organized as follows: we start with the method under verification bias, and then propose a simplified computationally easier version of the method without verification bias. We then run extensive simulation studies in Section 3.5, and apply the methods proposed in 2 and 3 as well as other methods to two real datasets in Section 3.6. We first apply the methods to compare the two biomarkers-CA125 and HE4 on their performances in discriminating healthy subjects, the early stage patients and the late stage patients of the epithelial ovarian cancer. We then apply this method to assess the ability

of serum albumin in distinguishing different stages of hepatocellular carcinoma when incorporating gender as a covariate.

3.2 Method under verification bias

3.2.1 Notation

Same as Chapter 2, denote the diagnostic measurements from the whole population by $\boldsymbol{S} = (S_1, \ldots, S_N)$, where N is the total number of subjects involved in the study. Among N observations, we have n_0 observations from healthy group, n_1 observations from level-1 disease group and n_2 observations from level-2 disease group, $N = n_0 + n_1 + n_2$. Note that n_0, n_1 and n_2 are unknown under verification bias. The true disease status of those N individuals are denoted by $\boldsymbol{D} = (D_1, \ldots, D_N)$. Only a small proportion of the true disease statuses are verified through the gold standard test. Let $\boldsymbol{L} = (L_1, \ldots, L_N)$ carry information on missingness for all subjects, as well as the true disease status if observed. Define a label variable L_i as follows:

$$L_i = \begin{cases} d, \text{if label is observed and } D_i = d, \quad d = 0, 1, 2, \\ 3, \text{if label is not observed.} \end{cases} \tag{3.1}$$

3.2.2 Propositions and Theorems

To develop our algorithm, we have used some propositions and theorems in Ghosal and van der Vaart (2017). Here we listed those theorems using the same numbering in the book.

Definition 4.1 A random measure P on $(X, \mathscr{X})$ is said to possess a Dirichlet process distribution DP(α) with base measure α, if for every finite measurable partition $A_1, \ldots, A_k$ of X,

$(P(A_1),\ldots,P(A_k)) \sim \mathrm{Dir}(k;\alpha(A_1),\ldots,\alpha(A_k))$, where $\mathrm{Dir}(k;\alpha(A_1),\ldots,\alpha(A_k))$ denotes a Dirichlet distribution with k categories and concentration parameters $\alpha(A_1),\ldots,\alpha(A_k)$.

Theorem 4.6 The posterior distribution given an i.i.d sample $X_1,\ldots,X_n$ from the DP(α)-process is DP($\alpha+\sum_{i=1}^n \delta_{X_i}$)-process.

Proposition G.10 If $X \sim \mathrm{Dir}(k;\alpha)$, $Y \sim \mathrm{Dir}(k;\beta)$, and $V \sim \mathrm{Be}(|\alpha|,|\beta|)$ are independent random vectors, then $VX+(1-V)Y \sim \mathrm{Dir}(k;\alpha+\beta)$. In particular, if $X \sim \mathrm{Dir}(k;\alpha)$ and $V \sim \mathrm{Be}(|\alpha|,|\beta|)$, then $VX+(1-V)e_i \sim \mathrm{Dir}(k;\alpha+\beta e_i)$, where e_i is the ith unit vector in $\mathbb{R}^k$, and $i \in \{1,\ldots,k\}$.

Theorem 4.19 (ii) Let P_N be a Dirichlet-multinomial process of order $N \to \infty$ with parameters $(M/N,\ldots,M/N)$ and G, then $\int \phi\, dP_N \rightsquigarrow \int \phi\, dP$, where $P \sim \mathrm{DP}(MG)$, for any $\phi \in \mathbb{L}_1(G)$

3.2.3 Model

As Chapter 2, We assume the disease prevalence rates for level-1 and level-2 disease in the population are λ_1 and λ_2 respectively, where $0<\lambda_1<1$ and $0<\lambda_2<1$, $\lambda_0 = 1-\lambda_1-\lambda_2$, so $(n_0,n_1,n_2) \sim \mathrm{Mult}(N,(\lambda_0,\lambda_1,\lambda_2))$. Conditional on the true disease labels, we have

$$S_i|\mu_i,\sigma_i^2 \sim \mathrm{N}(\mu_i,\sigma_i^2), \quad (\mu_i,\sigma_i^2)|\{D_i=d\} \sim G_d, \quad d=0,1,2, \tag{3.2}$$

where $\mathrm{N}(\mu_i,\sigma_i^2)$ stands for the normal distribution with mean μ_i and variance σ_i^2.

Based on (3.2), the distribution functions of the observations for the healthy group, level-1 disease group and level-2 disease group (denoted by F_0,F_1,F_2), are mixtures of normal distributions with mixing distributions G_0, G_1 and G_2 respectively, and the density functions

f_0, f_1, f_2 are mixtures of normal densities, i.e.,

$$F_d = \int \Phi(\frac{x-\mu}{\sigma}) d\, G_d(\mu,\sigma), \quad f_d(x) = \int \phi(\frac{x-\mu}{\sigma}) d\, G_d(\mu,\sigma), \quad d = 0,1,2, \tag{3.3}$$

where $\Phi(\cdot)$ and $\phi(\cdot)$ stand for the distribution function and density function respectively of a standard normal random variable.

We can express the points in the ROC surface as $(F_0(c_1), F_1(c_2) - F_1(c_1), 1 - F_2(c_2))$, where $c_1 \in \mathbb{R}$ and $c_2 \in \mathbb{R}$ are the two cut points for classifying the three categories. By varying the values of two cut points, we can obtain the ROC surface.

Again, we assume that the true disease status is missing at random, following Chapter 2, i.e.,

$$\mathrm{P}(L_i \neq 3 | S_i, D_i) = g(S_i), \quad i = 1, \ldots, N, \tag{3.4}$$

where g is an increasing function.

3.2.4 Prior distribution

To design a Bayesian algorithm, first of all we need to specify the prior distributions for distributions G_0, G_1 and G_2. We consider Dirichlet process priors (See Definition 4.1 in 3.2.2) on these three distributions. Let $\mathrm{DP}(MF)$ stand for the Dirichlet process with prior precision $M > 0$ and centering measure F, we have:

$$G_d \sim \mathrm{DP}(M_d G_d^*), \quad G_d^* : \mathrm{NIG}(m_d, a_d, s_d, \beta_d) \quad d = 0,1,2, \tag{3.5}$$

where $\mathrm{NIG}(m_d, a_d, s_d, \beta_d)$ stands for the normal-inverse gamma distribution, i.e., $\sigma^{-2} \sim \mathrm{Ga}(s_d, \beta_d), \mu|\sigma \sim \mathrm{N}(m_d, \sigma^2/a_d), M_d, s_d, \beta_d, m_d, a_d, d \in \{0,1,2\}$ are treated as constants which may be determined from the prior knowledge if available. The prior precision parameter M_d controls the variability of G_d around G_d^*, with larger values of M_d leading to realizations

of G_d that are closer to G_d^*.

3.2.5 Posterior distribution

We apply Gibbs sampling technique to sample from the posterior distributions . Let θ_i denote $(\mu_i, \sigma_i^2)^T$. Because θ_i is closely related to D_i, we choose to update them as a block. We use subscript $-i$ to denote the set of all index $j \neq i$. For example, $\theta_{-i} = (\theta_j : j \neq i)$, $D_{-i} = (D_j : j \neq i)$. The posterior distributions can be described as follows:

- The posterior distribution of D_i given $\boldsymbol{S}, \theta_{-i}, D_{-i}, L_i = 3$ is given according to the Bayes rule:

$$\Big(\mathbb{1}\{D_i = 0\}, \mathbb{1}\{D_i = 1\}, \mathbb{1}\{D_i = 2\}\Big) | (\boldsymbol{S}, \theta_{-i}, D_{-i}, L_i = 3, G_0, G_1, G_2) \sim \text{Mult}\left(1, \left(\frac{\lambda_0 f_0(S_i)}{f(S_i)}, \frac{\lambda_1 f_1(S_i)}{f(S_i)}, \frac{\lambda_2 f_2(S_i)}{f(S_i)}\right)\right), \tag{3.6}$$

  where $\lambda_0, \lambda_1, \lambda_2$ are the disease prevalence rates for healthy, level-1 and level-2 disease in the population, f_0, f_1, f_2 are defined in (3.3) above and $f(t) = \lambda_0 f_0(t) + \lambda_1 f_1(t) + \lambda_2 f_2(t)$. However, notice that G_0, G_1 and G_2 are unknown. We can only generate a set of samples of G_0, G_1 and G_2 from their posterior distributions given θ_{-i}, D_{-i} and then plug in (3.3) to calculate f_0, f_1 and f_2. Take G_0 as an example. The posterior distribution of G_0 given θ_{-i}, D_{-i} is $\text{DP}(M_0 G_0^* + \sum_{j \neq i} \delta_{\theta_j} \mathbb{1}\{D_j = 0\})$ according to Theorem 4.6 in 3.2.2. The sampling process for this posterior distribution is as follows:

  1. According to Proposition G.10 in 3.2.2, generate $V \sim \text{Be}(M_0, n_0^*)$ first, where $n_0^* = \sum_{j \neq i} \mathbb{1}\{D_j = 0\}$.
  2. Generate P as a sample from $\text{DP}(M_0 G_0^*)$ using Theorem 4.19 (ii) in 3.2.2. An appropriate sample is obtained by calculating $\sum_{j=1}^{N^*} q_j \delta_{\theta_j}$, where $N^* \gg M_0$ is a constant, here $\gg$ means far greater than, $(q_1, ..., q_{N^*}) \sim \text{Dir}(N^*; M_0/N^*, ..., M_0/N^*)$,

and $\theta_j \overset{\text{i.i.d.}}{\sim} G_0^*$ for $j = 1, \ldots, N^*$.

3. Generate Q as a sample from $\mathrm{DP}(\sum_{j \neq i} \delta_{\theta_j})$, i.e., calculate $\sum_{k=1}^{n_0^*} w_k \delta_{\theta_{j_k}}$, where $(w_1, \ldots, w_{n_0^*}) \sim \mathrm{Dir}(n_0^*; 1, 1, ..., 1)$, j_k satisfies that $D_{j_k} = 0$, $k = 1, \ldots, n_0^*$.

4. Calculate $VP + (1-V)Q$ and register as a posterior sample.

Plugging this sample in (3.3), we get f_0. Similarly for f_1 and f_2.

- The posterior distribution of θ_i conditional on $\boldsymbol{D}, \boldsymbol{L}, \boldsymbol{S}, \theta_{-i}$ is given by

$$\theta_i | (\theta_{-i}, \boldsymbol{S}, D_i = d) \sim \sum_{j \neq i} q_{i,j} \delta_{\theta_j} \mathbb{1}\{D_j = d\} + q_{i,0} G_{b,i}, \tag{3.7}$$

where $d \in \{0, 1, 2\}$, $(q_{i,j} : j \in \{1, \ldots, N\} \backslash \{i\})$ is the probability vector satisfying

$$q_{i,j} \propto \begin{cases} \dfrac{M_d \sqrt{a_d} \Gamma(s_d + 1/2) \beta_d^{s_d}}{\sqrt{1+a_d} \Gamma(s_d) \{\beta_d + a_d (S_i - m_d)^2 / (2(1+a_d))\}^{s_d + 1/2}}, & j = 0, \\ \sigma_j^{-1} \exp\{-(S_i - \mu_j)^2 / (2\sigma_j^2)\}, & D_j = d, j \neq i, j \geq 1, \\ 0, & \text{otherwise}, \end{cases} \tag{3.8}$$

and $G_{b,i}$ is the baseline measure given by

$$\begin{aligned} \mu | \sigma, S_i &\sim \mathrm{N}((S_i + a_d m_d)/(1 + a_d), \sigma^2/(1 + a_d)), \\ \sigma^{-2} | S_i &\sim \mathrm{Ga}(s_d + 1/2, \beta_d + a_d (S_i - m_d)^2 / (2(1 + a_d))). \end{aligned} \tag{3.9}$$

- To compute the ROC surface, we need to obtain the distribution functions F_0, F_1 and F_2. To do this, we need to sample from the posterior distribution of G_0, G_1 and G_2 conditional on $\boldsymbol{\theta}, \boldsymbol{D}$ and then plug in (3.3) to calculate F_0, F_1 and F_2. The posterior distribution of G_0 given $\boldsymbol{\theta}, \boldsymbol{D}$ is $\mathrm{DP}(M_0 G_0^* + \sum_{i=1}^N \delta_{\theta_i} \mathbb{1}\{D_i = 0\})$ according to Theorem 4.6 in 3.2.2. Similar to the sampling procedure mentioned above, with probability V,

we generate a sample from $\mathrm{DP}(M_0 G_0^*)$, and with probability $1-V$, generate a sample from $\mathrm{DP}(\sum_{i=1}^N \delta_{\theta_i} \mathbb{1}\{D_i=0\})$, where $V \sim \mathrm{Be}(M_0, n_0')$, $n_0' = \sum_{i=1}^N \mathbb{1}\{D_i=0\}$. Through the same process we can obtain a posterior sample from G_1 and G_2.

Remark Here we assume that the disease prevalence rates $(\lambda_0, \lambda_1, \lambda_2)$ are known. The disease prevalence rates in the population can be found in historical data. If the data we are looking at is representative of the population, using the population level disease prevalence rates is a good choice. We can also treat $(\lambda_0, \lambda_1, \lambda_2)$ as unknown and assign some priors on them. The likelihood is given by

$$\begin{aligned}\mathrm{P}(\boldsymbol{S}, \boldsymbol{D}|\lambda_0, \lambda_1, \lambda_2, \theta) = &\prod_{i:D_i=0} f_0(S_i) \prod_{i:D_i=1} f_1(S_i) \Pi_{i:D_i=2} f_2(S_i) \prod_{i:D_i=3} \{\lambda_0 f_0(S_i) + \lambda_1 f_1(S_i) + \lambda_2 f_2(S_i)\} \\ &\times \prod_{d=0}^{2} \{\lambda_d \int g(s) f_d(s) ds\}^{\#\{i:D_i=d\}} \left\{ \int [1-g(s)](\lambda_0 f_0(s) + \lambda_1 f_1(s) \right. \\ &\left. + \lambda_2 f_2(s)) ds \right\}^{\#\{i:D_i=3\}}\end{aligned}$$

We can update the posterior distributions accordingly.

A computational algorithm can be developed following the posterior distributions given above. The algorithm is given in Algorithm 3.[1] Notice that when $L_i \neq 3$ then $D_i = L_i$, $i = 1, \ldots, N$, we need to assign an initial value for $\{D_i : L_i = 3\}$. We will use a Bayes classifier to determine the initial value for D_i when $L_i = 3$, i.e. $D_i = \mathrm{argmax}_{d=0,1,2} \mathrm{P}(S_i|D_i = d)$, where $\mathrm{P}(S_i|D_i = d)$ is a rough estimate given by a kernel estimator based on the verified subjects, $d = 0, 1, 2$. We need to pre-specify the parameter $\boldsymbol{M} = (M_0, M_1, M_2), \boldsymbol{m} = (m_0, m_1, m_2), \boldsymbol{a} = (a_0, a_1, a_2), \boldsymbol{s} = (s_0, s_1, s_2), \boldsymbol{\beta} = (\beta_0, \beta_1, \beta_2)$ and also the number of iterations niter; here (M_0, M_1, M_2) are the precision parameters and can be chosen according to how confident we are about the prior distributions. We will choose a small value (like 1) if we

[1]The code for this method along with the real data analysis can be found in https://github.com/RrZzZz/Bayesian-nonparametric-estimation-of-ROC-surface-under-verification-bias

do not want to use a very informative one. The hyperparameters $\boldsymbol{m}, \boldsymbol{a}, \boldsymbol{s}, \boldsymbol{\beta}$ are chosen according to our prior knowledge about the distributions for different classes. If we the prior information is not available, then we will choose them to be non-informative. In the simulation part in Section 3.5, m_d is chosen according to the mean value of S_i whose $L_i = d$, where $d = 0, 1, 2$. For others, since the prior information is not available, the parameters $\boldsymbol{M}$ and $\boldsymbol{a}$ are chosen to be $(1, 1, 1)$, $\boldsymbol{s}$ and $\boldsymbol{\beta}$ are chosen to be $(0.1, 0.1, 0.1)$, N^* is chosen to be 1000. To reduce the computation memory needed, the output will be stored by the function values on the grid points, i.e., the outputs are $\boldsymbol{F_0}, \boldsymbol{F_1}, \boldsymbol{F_2}$, where $\boldsymbol{F_d} = (F_{d,1}, \ldots, F_{d,\text{niter}})$ for $d = 0, 1, 2$. The distribution functions are stored on a grid of points $(C_1, C_2, \ldots, C_K)$, i.e., record the values of $(F_{0,m}(C_1), \ldots, F_{0,m}(C_K), F_{1,m}(C_1), \ldots, F_{1,m}(C_K), F_{2,m}(C_1), \ldots, F_{2,m}(C_K))$ for each iteration, $m = 1, 2, \ldots, \text{niter}$.

We monitor the trace plot to make sure the algorithm converges and discard all samples before a suitable burn-in period. The three distribution functions F_0, F_1, F_2 are finally estimated according to the sample means of the values in each grid point $(C_1, C_2, \ldots, C_K)$. That is, $\bar{F}_0(C_1), \ldots, \bar{F}_0(C_K)$ for F_0, $\bar{F}_1(C_1), \ldots, \bar{F}_1(C_K)$ for F_1 and $\bar{F}_2(C_1), \ldots, \bar{F}_2(C_K)$ for F_2. The ROC surface is estimated based on a $K \times K$ grid points given by $(\bar{F}_0(c_1), \bar{F}_1(c_2) - \bar{F}_1(c_1), 1 - \bar{F}_2(c_2))$, where c_1 and c_2 can take values in $(C_1, C_2, \ldots, C_K)$. The VUS can then be calculated based on this ROC surface.

3.3 Method under verification bias with covariates

Generally in biomedical settings, apart from the test results, we also have some covariates information. Typical covariates can be patients' gender, age, height, weight, ethnicity etc. The accuracy of the diagnostic test can vary for different covariates values. So we also want to incorporate this information when estimating the ROC surface. The methodology is very similar to the one without covariates, just a few adjustments to the model are needed.

Algorithm 3 Dirichlet Process (DP) method for estimating ROC surface under verification bias.

Input: $\boldsymbol{S}$, $\boldsymbol{L}$, initial value $\boldsymbol{D}$, $(\lambda_0, \lambda_1, \lambda_2)$, niter, $\boldsymbol{M}$, $\boldsymbol{a}$, $\boldsymbol{s}$, $\boldsymbol{\beta}$, N^*

Output: $(\boldsymbol{F_0}, \boldsymbol{F_1}, \boldsymbol{F_2})$

$\boldsymbol{\mu} = \boldsymbol{S}$

$\boldsymbol{\sigma}^2 \sim \mathrm{IG}(0.1, 0.1)$ # Generate initial values for $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$

for $m \leftarrow 1$ to niter **do**

  for $i \leftarrow 1$ to N **do**

    # Step 1: Impute labels for D_i if the true disease status are unknown

    if $L_i = 3$ **then**

    for $d \leftarrow 0$ to 2 **do**

      $n_d^* = \sum_{j \neq i} \mathbb{1}\{D_j = d\}$

      $V \sim \mathrm{Be}(M_d, n_d^*)$

      $(q_1, ..., q_{N^*}) \sim \mathrm{Dir}(N^*; \frac{M_d}{N^*}, ..., \frac{M_d}{N^*})$

      for $k \leftarrow 0$ to N^* **do**

        $\theta_{d,k} \overset{\text{i.i.d.}}{\sim} G_d^*$

      end for

      $f_{1d}(S_i) = \sum_{k=1}^{N^*} q_k \phi(\frac{S_i - \mu_{d,k}}{\sigma_{d,k}})$

      $(w_1, \ldots, w_{n_d^*}) \sim \mathrm{Dir}(n_d^*; 1, 1, ..., 1)$

      $f_{2d}(S_i) = \sum_{k=1}^{n_d^*} w_k \phi(\frac{S_i - \mu_{j_k}}{\sigma_{j_k}})$ where j_k satisfies that $D_{j_k} = d$ and $j_k \neq i$

      $f_d(S_i) = V f_{1d}(S_i) + (1 - V) f_{2d}(S_i)$

    end for

    $f(S_i) = \lambda_0 f_0(S_i) + \lambda_1 f_1(S_i) + \lambda_2 f_2(S_i)$

    $(\mathbb{1}\{D_i = 0\}, \mathbb{1}\{D_i = 1\}, \mathbb{1}\{D_i = 2\}) \sim \mathrm{Mult}\left(1, \left(\frac{\lambda_0 f_0(S_i)}{f(S_i)}, \frac{\lambda_1 f_1(S_i)}{f(S_i)}, \frac{\lambda_2 f_2(S_i)}{f(S_i)}\right)\right)$

    end if

    # Step 2: Update θ_i according to Dirichlet Process

    for $j \leftarrow 0$ to N **do**

      if $j = 0$ **then**

      $q_{i,j} = \frac{M_{D_i} \sqrt{a_{D_i}} \Gamma(s_{D_i} + 1/2) \beta_{D_i}^{s_{D_i}}}{\sqrt{1 + a_{D_i}} \Gamma(s_{D_i}) \{\beta_{D_i} + a_{D_i}(S_i - m_{D_i})^2 / (2(1 + a_{D_i}))\}^{s_{D_i} + 1/2}}$

      else if $D_j = D_i$ **and** $j \neq i$

      $q_{i,j} = \sigma_j^{-1} \exp\{-(S_i - \mu_j)^2 / (2\sigma_j^2)\}$

      else

      $q_{i,j} = 0$

      end if

    end for $=0$

```
    normalize q_i = (q_{i,0}, ..., q_{i,N})
    σ_{0,i}^{-2} ~ Ga(s_{D_i} + 1/2, β_{D_i} + a_d(S_i − m_{D_i})^2/(2(1 + a_{D_i})))
    μ_{0,i} ~ N((S_i + a_{D_i} m_{D_i})/(1 + a_{D_i}), σ_{0,i}^2/(1 + a_{D_i}))
    θ_{0,i} = (μ_{0,i}, σ_{0,i}^2)^T
    r_i = Mult(1, q_i)
    θ_i = (θ_{0,i}, θ_1, ..., θ_N) r_i^T
  end for
  # Step 3: Sample distribution functions for different class
  for d ← 0 to 2 do
    n*_d = Σ_{j=1}^{N} 𝟙{D_j = d}
    V ~ Be(M_d, n*_d)
    (q_1, ..., q_{N*}) ~ Dir(N*; M_d/N*, ..., M_d/N*)
    for k ← 0 to N* do
      θ_{d,k} ~(i.i.d.) G*_d
    end for
    F_{1d} = Σ_{k=1}^{N*} q_k Φ((x − μ_{d,k})/σ_{d,k})
    (w_1, ..., w_{n*_d}) ~ Dir(n*_d; 1, 1, ..., 1)
    F_{2d} = Σ_{k=1}^{n*_d} w_k Φ((x − μ_{j_k})/σ_{j_k}) where j_k satisfies that D_{j_k} = d
    F_d = V F_{1d} + (1 − V) F_{2d}
  end for
end for
```

Denote the covariates for patient i by $\boldsymbol{w_i} = (1, w_{i1}, \ldots, w_{ik})^T$, where k is the number of covariates we have. Assume that the observed diagnostic measurement S_i has a linear relationship with the covariates $\boldsymbol{w_i}$, i.e.,

$$S_i|\gamma_i, w_i, \sigma_i^2 \sim \mathrm{N}(\gamma_i^T w_i, \sigma_i^2), \quad (\gamma_i, \sigma_i^2)|\{D_i = d\} \sim G_d, \quad d = 0, 1, 2. \tag{3.10}$$

Notice that the parameters γ_is and σ_i^2s are specific to each individual i. The distribution functions and density functions of the observations for each group will depend on the covariates $\boldsymbol{w}$. Let F_{0w} denote the conditional distribution function for observations in the healthy group conditional on the covariates, F_{1w} denote the conditional distribution in level-1 disease group and F_{2w} denote the conditional distribution in level-2 disease group. The conditional density functions are f_{0w}, f_{1w} and f_{2w} respectively. Based on (3.10), we have:

$$F_{dw}(x) = \int \Phi(\frac{x - \gamma^T w}{\sigma}) dG_d(\gamma, \sigma), \quad f_{dw}(x) = \int \phi(\frac{x - \gamma^T w}{\sigma}) dG_d(\gamma, \sigma), \quad d = 0, 1, 2, \tag{3.11}$$

which gives a semiparametric model.

Following Li et al. (2012), the covariate-specific ROC surface is defined as

$$\mathrm{ROC}_w(p_1, p_3) = \begin{cases} F_{1w}(F_{2w}^{-1}(1 - p_3)) - F_{1w}(F_{0w}^{-1}(p_1)), & \text{if } F_{0w}^{-1}(p_1) \leq F_{2w}^{-1}(1 - p_3) \\ 0, & \text{otherwise,} \end{cases} \tag{3.12}$$

and the covariate-adjusted ROC surface is defined as

$$\mathrm{ROC}(p_1, p_3) = \int \mathrm{ROC}_w(p_1, p_3) dJ_w(w) \tag{3.13}$$

where $J_w(w)$ stands for the distribution function for w.

The prior distributions are adjusted to be

$$G_d \sim \mathrm{DP}(M_d G_d^*), \quad G_d^* : \mathrm{NIG}(\gamma_{0d}, \boldsymbol{\Lambda}_d, s_d, \beta_d) \quad d = 0, 1, 2, \tag{3.14}$$

where $\mathrm{NIG}(m_d, a_d, s_d, \beta_d)$ still stands for the normal inverse gamma distribution, i.e., $\sigma^{-2} \sim \mathrm{Ga}(s_d, \beta_d), \gamma|\sigma \sim \mathrm{N}(\gamma_{0d}, \sigma^2 \boldsymbol{\Lambda}_d^{-1})$, here γ_{0d} is a $(k+1) \times 1$ vector and $\boldsymbol{\Lambda}_d$ is a $(k+1) \times (k+1)$ matrix for $d = 0, 1, 2$.

Let $\theta_i = (\gamma_i, \sigma_i^2)$, the posterior distribution of θ_i conditional on $\boldsymbol{D}, \boldsymbol{L}, \boldsymbol{S}, \theta_{-i}$ is still given by

$$\theta_i | (\theta_{-i}, \boldsymbol{S}, D_i = d) \sim \sum_{j \neq i} q_{i,j} \delta_{\theta_j} \mathbb{1}\{D_j = d\} + q_{i,0} G_{b,i}, \tag{3.15}$$

where $d \in \{0, 1, 2\}$, the probability vector $(q_{i,j} : j \in \{1, \ldots, N\} \backslash \{i\})$ is now satisfying

$$q_{i,j} \propto \begin{cases} \dfrac{M_d \beta_d^{s_d} \Gamma(s_d + 1/2) \sqrt{|\boldsymbol{\Lambda}_d|}}{(2\pi)^{1/2} \Gamma(s_d) \sqrt{|\boldsymbol{\Lambda}_d^*|}} \Big[\beta_d + \dfrac{1}{2} \Big(S_i^2 + \gamma_{0d}^T \boldsymbol{\Lambda}_d \gamma_{0d} - \gamma_{0d}^{*T} \boldsymbol{\Lambda}_d^* \gamma_{0d}^* \Big) \Big]^{-(s_d + 1/2)}, & j = 0, \\ \sigma_j^{-1} \exp\{-(S_i - \gamma_j^T w_i)^2 / (2\sigma_j^2)\}, & D_j = d, \\ & j \neq i, j \geq 1, \\ 0, & \text{otherwise}, \end{cases} \tag{3.16}$$

and $G_{b,i}$ is given by

$$\begin{aligned} &\gamma | \sigma, S_i \sim \mathrm{N}(\gamma_{0d}^*, \sigma^2 \boldsymbol{\Lambda}_d^{*-1}), \\ &\sigma^{-2} | S_i \sim \mathrm{Ga}(s_d + 1/2, \beta_d + \frac{1}{2}(S_i^2 + \gamma_{0d}^T \boldsymbol{\Lambda}_d \gamma_{0d} - \gamma_{0d}^{*T} \boldsymbol{\Lambda}_d^* \gamma_{0d}^*). \end{aligned} \tag{3.17}$$

where $\gamma_{0d}^* = (\boldsymbol{\Lambda}_d + w_i w_i^T)^{-1} (\boldsymbol{\Lambda}_d \gamma_{0d} + S_i w_i)$, $\boldsymbol{\Lambda}_d^* = \boldsymbol{\Lambda}_d + w_i w_i^T$.

3.4 Method without verification bias

The above Bayesian nonparametric method can be substantially simplified for the case without verification bias by skipping the step to update D_i since, under the gold standard, all D_is are known to us. As no classification step will be needed, the kernel smoothing step will also be unnecessary, and hence the Dirichlet process can be directly applied on the distributions of X, Y and Z. However, then the sampling process will be very close to the Bayesian bootstrap process, which is much easier to sample from with only a fixed number of independent exponential random variables (see Section 4.7 of Ghosal and van der Vaart (2017)). After writing this chapter, we learned that this method was also recently proposed by de Carvalho et al. (2018), which is an extension of the Bayesian bootstrap method of estimation of ROC curve proposed by Gu and Ghosal (2009) for two categories.

3.4.1 Notation

The notation is the same as above. Note that under the gold standard, $\boldsymbol{L} = \boldsymbol{D}$, i.e., the label reflects the true disease status. Because of that, we can denote the observations from the healthy group by $\boldsymbol{X}$=$\boldsymbol{X}_{n_0}$=$(1, ..., X_{n_0})$=$(S_i : D_i = 0, i = 1, ..., N)$, the observations from level-1 disease by $\boldsymbol{Y}$=$\boldsymbol{Y}_{n_1}$=$(Y_1, ..., Y_{n_1})$=$(S_i : D_i = 1, i = 1, ..., N)$ and the observations from level-2 disease by $\boldsymbol{Z}$=$\boldsymbol{Z}_{n_2}$= $(Z_1, ..., Z_{n_2})$=$(S_i : D_i = 2, i = 1, ..., N)$. In the absence of verification bias, $\boldsymbol{X}$, $\boldsymbol{Y}$ and $\boldsymbol{Z}$ are all known to us.

3.4.2 Model

The model part is simplified. The only reason we are modeling the underlying distributions for three different disease groups as Dirichlet Gaussian mixture models is that we need to sample the density functions f_0, f_1 and f_2 when updating the unobserved underlying disease

status $\{D_i : L_i = 3\}$. Since in this case $\boldsymbol{D} = \boldsymbol{L}$, we no longer need to update $\{D_i : L_i = 3\}$, and thus can only focus on the distributions themselves without specifying any specific model. Thus we have

$$X_i \overset{\text{i.i.d.}}{\sim} F_0, \quad i = 1, 2, \ldots, n_0; \quad Y_j \overset{\text{i.i.d.}}{\sim} F_1, \quad j = 1, 2, \ldots, n_1; \quad Z_k \overset{\text{i.i.d.}}{\sim} F_2, \quad k = 1, 2, \ldots, n_2.$$

The general functional form of ROC surface is given by (2.3). Let $G(t) = F_0(F_1^{-1}(t))$, $H(t) = F_2(F_1^{-1}(t))$, then we have

$$\mathrm{ROC}_s(p_1, p_3) = \begin{cases} H^{-1}(1-p_3) - G^{-1}(p_1), & \text{if } G^{-1}(p_1) \le H^{-1}(1-p_3) \\ 0, & \text{otherwise.} \end{cases} \tag{3.18}$$

As for the VUS, it is equal to the probability that three randomly selected measurements, one in each class, are in the correct order (Dreiseitl et al., 2000), i.e., $\mathrm{VUS} = \mathrm{P}(X < Y < Z)$. With G and H as defined above, we have

$$\begin{aligned}
\mathrm{VUS} &= \mathrm{P}(X < Y < Z) \\
&= \mathrm{P}(F_0^{-1}(U_0) < F_1^{-1}(U_1) < F_2^{-1}(U_2)) \\
&= \int_0^1 \mathrm{P}(F_0^{-1}(U_0) < F_1^{-1}(u_1) < F_2^{-1}(U_2) | U_1 = u_1) d\, u_1 \\
&= \int_0^1 \mathrm{P}(F_0^{-1}(U_0) < F_1^{-1}(u_1) | U_1 = u_1) \mathrm{P}(F_2^{-1}(U_2) > F_1^{-1}(u_1) | U_1 = u_1) d\, u_1 \\
&= \int_0^1 \mathrm{P}(U_0 < F_0(F_1^{-1}(u_1)) | U_1 = u_1) \mathrm{P}(U_2 > F_2(F_1^{-1}(u_1)) | U_1 = u_1) d\, u_1 \\
&= \int_0^1 \mathrm{P}(U_0 < G(u_1) | U_1 = u_1) \mathrm{P}(U_2 > H(u_1) | U_1 = u_1) d\, u_1 \\
&= \int_0^1 G(u_1)[1 - H(u_1)] d\, u_1,
\end{aligned} \tag{3.19}$$

where $U_0, U_1, U_2 \overset{\text{i.i.d.}}{\sim} \mathrm{U}(0,1)$; here $\mathrm{U}(0,1)$ stands for the uniform distribution on $(0,1)$. Thus we only need to estimate the functions G and H to obtain the estimates of the ROC surface and the VUS.

3.4.3 Prior distributions and posterior computation

To conduct a Bayesian analysis, a natural choice for priors on F_0, F_1 and F_2 is the Dirichlet process. Let $F_0 \sim \mathrm{DP}(M_0\xi_0)$, $F_1 \sim \mathrm{DP}(M_1\xi_1)$, $F_2 \sim \mathrm{DP}(M_2\xi_2)$. Then conditional on the data $(X_1,\ldots,X_{n_0})$, $(Y_1,\ldots,Y_{n_1})$ and $(Z_1,\ldots,Z_{n_2})$, the posterior distribution F_1 is

$$F_1|\text{data} \sim \mathrm{DP}(M_1\xi_1 + n_1\mathbb{F}_1).$$

Given F_1, the posterior distributions of G and H are given by

$$G|(F_1,\text{data}) \sim \mathrm{DP}(M_0\xi_0 \circ F_1^{-1} + n_0\mathbb{F}_0 \circ F_1^{-1}),$$
$$H|(F_1,\text{data}) \sim \mathrm{DP}(M_2\xi_2 \circ F_1^{-1} + n_2\mathbb{F}_2 \circ F_1^{-1}),$$

where $\mathbb{F}_0, \mathbb{F}_1$ and $\mathbb{F}_2$ stand for the empirical distributions based on $\boldsymbol{X}, \boldsymbol{Y}$ and $\boldsymbol{Z}$ respectively. However, this realization involves generating an infinite collection of random variables, which is computationally more intensive. Hence we consider the non-informative limit of the Dirichlet process by letting $M_0 \to 0$, $M_1 \to 0$ and $M_2 \to 0$. In this case, we do not even need to specify the centering measure ξ_1, ξ_2 and ξ_3. The posteriors in this case would be

$$F_1|\text{data} \sim \mathrm{DP}(n_1\mathbb{F}_1),\ G|(F_1,\text{data}) \sim \mathrm{DP}(n_0\mathbb{F}_0 \circ F_1^{-1}),\ H|(F_1,\text{data}) \sim \mathrm{DP}(n_2\mathbb{F}_2 \circ F_1^{-1}),$$

known as the Bayesian bootstrap distribution.

Based on the posterior distributions above, we can obtain the algorithm for estimating

the ROC surface without verification bias in Algorithm 4. We can obtain the estimates of $\hat{G}$ and $\hat{H}$ by averaging the MCMC samples of $\boldsymbol{G}$ and $\boldsymbol{H}$. Plugging in the estimates to (3.18) and (3.19), we get the estimate of the ROC surface as well as the VUS.

Algorithm 4 Bayesian Bootstrap (BB) method for estimating ROC surface without verification bias.

Input: $\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z}, (n_0, n_1, n_2)$, niter
Output: $\boldsymbol{H}, \boldsymbol{G}$

for $m \leftarrow 1$ to niter **do**
  $(p_1, \ldots, p_{n_1}) \overset{\text{i.i.d.}}{\sim} \text{Dir}(n_1; 1, \ldots, 1)$
  $F_{1,m}(u) = \sum_{j=1}^{n_1} p_j \mathbb{1}(Y_j \leq u)$
  $\boldsymbol{U} = F_{1,m}(\boldsymbol{X})$
  $\boldsymbol{T} = F_{1,m}(\boldsymbol{Z})$
  $(q_1, \ldots, q_{n_0}) \overset{\text{i.i.d.}}{\sim} \text{Dirichlet}(n_0; 1, \ldots, 1)$
  $G_m(t) = \sum_{i=1}^{n_0} q_i \mathbb{1}(U_i \leq t)$
  $(r_1, \ldots, r_{n_2}) \overset{\text{i.i.d.}}{\sim} \text{Dirichlet}(n_2; 1, \ldots, 1)$
  $H_m(v) = \sum_{k=1}^{n_2} r_k \mathbb{1}(T_k \leq v)$
end for

3.5 Simulation

3.5.1 With verification bias

MAR assumption satisfied

In this simulation, we set $n_0 = n_1 = n_2 = n$, i.e., we generate n data from each category and get $N = 3n$ data in total. Here we consider 6 different underlying true models:

1. Norm 1: $X \sim \text{N}(-4, 9)$, $Y \sim \text{N}(3, 16)$, $Z \sim \text{N}(5, 25)$, $\text{VUS} = 0.548$.

2. Norm 2: $X \sim \text{N}(0, 4)$, $Y \sim \text{N}(5, 9)$, $Z \sim \text{N}(8, 9)$, $\text{VUS} = 0.679$.

3. Mixnorm 1: $X \sim 0.4\mathrm{N}(2,25)+0.6\mathrm{N}(-5,9)$, $Y \sim 0.5\mathrm{N}(5,16)+0.5\mathrm{N}(-3,9)$, $Z \sim 0.6\mathrm{N}(10,9)+0.4\mathrm{N}(0,25)$, $\mathrm{VUS}=0.462$.

4. Mixnorm 2: $X \sim \mathrm{N}(-3,9)$, $Y \sim 0.3\mathrm{N}(11,9)+0.3\mathrm{N}(-5,9)+0.4\mathrm{N}(3,4)$, $Z \sim 0.6\mathrm{N}(15,4)+0.4\mathrm{N}(3,9)$, $\mathrm{VUS}=0.555$.

5. Logis 1: $X \sim \mathrm{Logis}(-4,9)$, $Y \sim \mathrm{Logis}(3,4)$, $Z \sim \mathrm{Logis}(4,36)$, $\mathrm{VUS}=0.452$, where $\mathrm{Logis}(\mu, s^2)$ stands for the logistic distribution with location μ and scale s.

6. Logis 2: $X \sim \mathrm{Logis}(0,4)$, $Y \sim \mathrm{Logis}(5,9)$, $Z \sim \mathrm{Logis}(10,9)$, $\mathrm{VUS}=0.556$.

Within each model setting, we consider two different verification mechanisms which satisfy the MAR assumption proposed by Gu et al. (2014) as follows:

$$\mathrm{P}(L_i \neq 3|S) = \begin{cases} 1, & \text{if } S > S_{(0.6N)}, \\ 0.6, & \text{otherwise.} \end{cases} \tag{3.20}$$

$$\mathrm{P}(L_i \neq 3|S) = \Phi(0.1 + 0.1Q_i). \tag{3.21}$$

The first verification mechanism will give a missing rate at 36% while the second missing mechanism will give different missing rates to different model settings but will always lie in the range of 30%–45%.

The literature on ROC surface estimation is very limited for the time being. We compare our proposed Bayesian non-parametric method (denoted by DP) to Bayesian Rank Likelihood (BRL) proposed in Chapter 2, and Full Imputation (FI), Mean Score Imputation (MSI), Inverse Probability Weighted (IPW) and Semi-Parametric Efficient (SPE) methods proposed by To Duc et al. (2016). We consider a multivariate logistic model to estimate the true disease status and a logistic model to estimate the verification probability when applying FI, MSI, IPW and SPE methods. Those methods are implemented using R package

bcROCsurface (To Duc 2017). In addition, to show that using unlabeled data can actually help with estimating ROC surface, we consider using a nonparametric empirical method on only labeled data with Mann-Whitney U-statistic. We can see in Section 3.5.2 that this method gives the best accuracy for ROC surface estimate without verification bias. We will denote this method by MW.

The simulation results are given in Tables 3.1 and 3.2. Consider $n = 100$ and $n = 50$ and simulate 100 datasets for each setting. Both DP and BRL estimates are obtained based on 90000 Gibbs samples (100000 MCMC iterations after 10000 iterations used for burn-in). Some results are not shown either because the bias and MSE are too large or because the corresponding algorithm fails to converge.

Based on the simulation results, we find out that DP method demonstrates superiority in certain cases and is at least comparable with others in other cases. More specifically, comparing DP to BRL, DP is very competitive with BRL in Norm 1 and Norm 2 cases, where BRL is expected to have the best performance since the trinormality assumption is perfectly satisfied. DP has much better performances in Mixnorm 2 and Logis 1 compared to BRL, maybe because the model settings are too far away from the trinormality assumption. Indeed, BRL even failed to MCMC converge in logis 1 case when $n = 50$. Compared to other methods for ROC surface estimate with verification bias, we can see that DP outperforms FI, MSI, IPW and SPE in terms of both MSE and bias in most cases, especially in Mixnorm 1, Mixnorm 2 and Logis 1. Another observation is that FI, MSI, IPW and SPE give different results to different true verification models, while DP and BRL are more robust to the true verification models. Compared to ROC surface estimate using only labeled data, we can see that DP and BRL are at least comparable with MW. In some cases, for example, Mixnorm 2 and Logis1, MW is much worse than almost all other methods. So this confirms that there is necessity to use unlabeled data when it is not missing completely at random.

We can thus reach a conclusion that DP is preferred in general. While the performance

of BRL depends on whether the trinormality assumption is satisfied or not, FI, MSI, IPW and SPE depend on the specification of the verification model or the true disease model, MW with only labeled data will cause large bias in some cases, DP has a satisfactory performance in all cases considered and should be considered as a safe choice. Moreover, the ROC surfaces fitted using FI, MSI, IPW or SPE are not smooth, while DP and BRL give smooth estimates of the surface, which is a desirable property.

Departure from the MAR assumption

The only assumption we made for the DP method is the MAR assumption for missing true disease status. Thus it is important to study the performance of the DP method when this assumption is not satisfied. Here we consider the underlying distribution the same as Mixnorm 2 case, i.e., $X \sim \mathrm{N}(-3,9)$, $Y \sim 0.3\mathrm{N}(11,9)+0.3\mathrm{N}(-5,9)+0.4\mathrm{N}(3,4)$, $Z \sim 0.6\mathrm{N}(15,4)+0.4\mathrm{N}(3,9)$, $\mathrm{VUS}=0.555$. We generate $n=100$ from each category. Since the verification no longer satisfies the MAR assumption, (3.4) no longer holds. Instead, we have

$$\mathrm{P}(L_i \neq 3|S_i, D_i = d) = g_d(S_i), \quad d = 0,1,2. \tag{3.22}$$

The posterior distribution for disease status in this case is actually:

$$\mathrm{P}(D_i = k|S_i = t, L_i = 3) = \frac{\lambda_k g_k(t) f_k(t)}{\lambda_0 g_0(t) f_0(t) + \lambda_1 g_1(t) f_1(t) + \lambda_2 g_2(t) f_2(t)}, \quad k = 0,1,2;$$

see Section 2.6.4.

We consider two cases when the MAR assumption fails:

1. Let $g(S;\alpha,\beta) = \Phi(\alpha+\beta S)$.

   The verification model for healthy group is $g_0(S) = g(S;0.15,0.2)$; the verification model for level-1 disease is $g_1(S) = g(S;0.1,0.1)$; the verification model for level-2

Table 3.1: Bias ($\times 10$) and MSE ($\times 10^2$) for the estimate of the VUS, with verification bias generated using the threshold model, $n = 100$ and 50. (DP: Dirichlet Process, BRL: Bayesian Rank Likelihood, FI: Full Imputation, MSI: Mean Score Imputation, IPW: Inverse Probability Weighted, SPE: Semi-Parametric Efficient, MW: Mann-Whitney U-statistic)

		DP	BRL	FI	MSI	IPW	SPE	MW
		Norm 1, VUS = 0.548						
$n = 100$	bias	−0.03	0.07	0.08	0.10	0.09	−0.16	0.43
		(0.05)	(0.04)	(0.05)	(0.05)	(0.06)	(0.06)	(0.04)
	MSE	0.21	0.16	0.29	0.27	0.32	0.37	0.37
		(0.03)	(0.02)	(0.06)	(0.05)	(0.06)	(0.06)	(0.04)
$n = 50$	bias	−0.21	0.25	0.18	0.19	0.19	−0.06	0.36
		(0.05)	(0.05)	(0.07)	(0.07)	(0.08)	(0.08)	(0.06)
	MSE	0.29	0.34	0.57	0.53	0.65	0.67	0.44
		(0.04)	(0.06)	(0.08)	(0.07)	(0.09)	(0.12)	(0.05)
		Norm 2, VUS = 0.679						
$n = 100$	bias	−0.17	0.09	0.13	0.08	0.01	−0.13	0.36
		(0.04)	(0.04)	(0.04)	(0.04)	(0.05)	(0.05)	(0.04)
	MSE	0.18	0.16	0.16	0.16	0.25	0.30	0.16
		(0.02)	(0.02)	(0.02)	(0.02)	(0.03)	(0.05)	(0.02)
$n = 50$	bias	−0.12	0.03	0.11	0.07	0.07	−0.07	0.09
		(0.05)	(0.05)	(0.07)	(0.07)	(0.07)	(0.08)	(0.06)
	MSE	0.27	0.29	0.45	0.43	0.52	0.60	0.32
		(0.04)	(0.04)	(0.06)	(0.06)	(0.08)	(0.10)	(0.05)
		Mixnorm 1, VUS = 0.462						
$n = 100$	bias	−0.06	−0.20	−0.34	−0.17	0.07	−0.15	−0.07
		(0.04)	(0.04)	(0.04)	(0.04)	(0.07)	(0.07)	(0.04)
	MSE	0.18	0.22	0.31	0.23	0.46	0.52	0.20
		(0.02)	(0.03)	(0.03)	(0.03)	(0.06)	(0.08)	(0.03)
$n = 50$	bias	−0.19	0.04	−0.23	−0.06	0.17	−0.77	0.05
		(0.06)	(0.06)	(0.07)	(0.07)	(0.10)	(0.12)	(0.06)
	MSE	0.39	0.38	0.48	0.43	1.04	1.43	0.37
		(0.05)	(0.06)	(0.06)	(0.06)	(0.17)	(0.47)	(0.05)
		Mixnorm 2, VUS = 0.555						
$n = 100$	bias	0.31	0.40	0.70	0.63	−0.10	−0.41	0.79
		(0.04)	(0.04)	(0.04)	(0.05)	(0.07)	(0.08)	(0.04)
	MSE	0.25	0.34	0.65	0.61	0.49	0.80	0.81
		(0.03)	(0.04)	(0.06)	(0.06)	(0.06)	(0.13)	(0.08)
$n = 50$	bias	0.17	0.61	0.67	0.57	−0.25	−0.62	0.88
		(0.05)	(0.06)	(0.06)	(0.06)	(0.10)	(0.12)	(0.06)
	MSE	0.29	0.76	0.75	0.74	0.98	1.80	1.13
		(0.05)	(0.09)	(0.09)	(0.09)	(0.11)	(0.31)	(0.11)
		Logis 1, VUS = 0.452						
$n = 100$	bias	−0.05	0.18	−0.21	0.19	−0.77	-	0.78
		(0.05)	(0.09)	(0.06)	(0.05)	(0.11)	-	(0.05)
	MSE	0.26	0.83	0.39	0.30	1.69	-	0.83
		(0.04)	(0.03)	(0.06)	(0.04)	(0.33)	-	(0.08)
$n = 50$	bias	−0.21	-	−0.20	0.16	−0.58	-	0.80
		(0.06)	-	(0.08)	(0.07)	(0.13)	-	(0.07)
	MSE	0.39	-	0.71	0.58	2.14	-	1.15
		(0.05)	-	(0.08)	(0.08)	(0.35)	-	(0.14)
		Logis 2, VUS = 0.556						
$n = 100$	bias	−0.20	−0.09	0.07	0.10	−0.29	-	0.24
		(0.04)	(0.05)	(0.04)	(0.04)	(0.11)	-	(0.04)
	MSE	0.18	0.23	0.17	0.18	1.18	-	0.25
		(0.02)	(0.04)	(0.02)	(0.03)	(0.03)	-	(0.03)
$n = 50$	bias	−0.35	0.06	−0.10	−0.08	−0.42	-	0.29
		(0.06)	(0.07)	(0.07)	(0.07)	(0.12)	-	(0.06)
	MSE	0.44	0.45	0.46	0.45	1.62	-	0.49
		(0.06)	(0.07)	(0.08)	(0.07)	(0.28)	-	(0.07)

Table 3.2: Bias ($\times 10$) and MSE ($\times 10^2$) for the estimate of the VUS, with verification bias generated using the probit model, $n = 100$ and 50. (DP: Dirichlet Process, BRL: Bayesian Rank Likelihood, FI: Full Imputation, MSI: Mean Score Imputation, IPW: Inverse Probability Weighted, SPE: Semi-Parametric Efficient, MW: Mann-Whitney U-statistic)

		DP	BRL	FI	MSI	IPW	SPE	MW
		Norm 1, VUS = 0.548						
$n = 100$	bias	−0.13	0.10	0.17	0.18	0.09	< 0.01	0.26
		(0.04)	(0.04)	(0.05)	(0.05)	(0.06)	(0.06)	(0.04)
	MSE	0.20	0.18	0.30	0.29	0.30	0.30	0.24
		(0.02)	(0.03)	(0.04)	(0.04)	(0.04)	(0.04)	(0.03)
$n = 50$	bias	−0.20	0.26	0.14	0.15	−0.05	−0.05	0.30
		(0.07)	(0.06)	(0.07)	(0.07)	(0.07)	(0.07)	(0.06)
	MSE	0.48	0.42	0.51	0.47	0.53	0.55	0.50
		(0.06)	(0.07)	(0.07)	(0.07)	(0.08)	(0.08)	(0.08)
		Norm 2, VUS = 0.679						
$n = 100$	bias	−0.17	0.05	0.10	0.05	< 0.01	−0.01	0.08
		(0.04)	(0.04)	(0.04)	(0.04)	(0.04)	(0.04)	(0.04)
	MSE	0.16	0.15	0.14	0.14	0.16	0.16	0.21
		(0.02)	(0.02)	(0.02)	(0.02)	(0.02)	(0.02)	(0.03)
$n = 50$	bias	−0.36	0.21	0.28	0.22	0.15	0.15	0.13
		(0.05)	(0.06)	(0.05)	(0.05)	(0.06)	(0.06)	(0.05)
	MSE	0.36	0.41	0.35	0.33	0.34	0.33	0.30
		(0.05)	(0.05)	(0.05)	(0.05)	(0.05)	(0.05)	(0.04)
		Mixnorm 1, VUS = 0.462						
$n = 100$	bias	−0.05	0.11	−0.33	−0.21	−0.18	−0.19	−0.07
		(0.05)	(0.05)	(0.05)	(0.05)	(0.06)	(0.05)	(0.04)
	MSE	0.21	0.23	0.32	0.27	0.35	0.34	0.40
		(0.03)	(0.03)	(0.04)	(0.04)	(0.06)	(0.06)	(0.05)
$n = 50$	bias	−0.17	0.27	−0.25	−0.14	−0.15	−0.17	−0.12
		(0.07)	(0.06)	(0.06)	(0.07)	(0.09)	(0.09)	(0.06)
	MSE	0.48	0.48	0.46	0.46	0.77	0.79	0.37
		(0.06)	(0.08)	(0.06)	(0.07)	(0.12)	(0.12)	(0.05)
		Mixnorm 2, VUS = 0.555						
$n = 100$	bias	0.21	0.46	0.77	0.70	0.08	0.07	0.82
		(0.04)	(0.04)	(0.03)	(0.04)	(0.04)	(0.05)	(0.04)
	MSE	0.22	0.36	0.72	0.62	0.19	0.24	0.86
		(0.03)	(0.04)	(0.06)	(0.05)	(0.03)	(0.05)	(0.08)
$n = 50$	bias	0.26	0.55	0.83	0.74	0.08	0.05	0.89
		(0.05)	(0.06)	(0.06)	(0.07)	(0.08)	(0.09)	(0.07)
	MSE	0.30	0.66	1.04	0.97	0.67	0.74	1.26
		(0.04)	(0.08)	(0.11)	(0.11)	(0.09)	(0.11)	(0.14)
		Logis 1, VUS = 0.452						
$n = 100$	bias	−0.08	0.38	0.39	0.54	−0.01	−0.06	0.87
		(0.05)	(0.04)	(0.05)	(0.05)	(0.07)	(0.08)	(0.05)
	MSE	0.20	0.28	0.44	0.55	0.44	0.57	1.05
		(0.04)	(0.04)	(0.06)	(0.07)	(0.06)	(0.09)	(0.09)
$n = 50$	bias	−0.10	-	0.45	0.60	0.07	−0.05	0.91
		(0.06)	-	(0.08)	(0.08)	(0.09)	(0.13)	(0.08)
	MSE	0.31	-	0.85	0.95	0.87	1.73	1.40
		(0.05)	-	(0.12)	(0.13)	(0.11)	(0.90)	(0.18)
		Logis 2, VUS = 0.556						
$n = 100$	bias	−0.14	0.15	0.14	0.16	−0.05	−0.06	0.20
		(0.04)	(0.05)	(0.04)	(0.04)	(0.05)	(0.05)	(0.04)
	MSE	0.20	0.23	0.16	0.20	0.23	0.26	0.20
		(0.04)	(0.03)	(0.02)	(0.03)	(0.03)	(0.04)	(0.02)
$n = 50$	bias	−0.27	0.17	0.18	0.21	−0.03	−0.05	0.12
		(0.06)	(0.07)	(0.06)	(0.06)	(0.07)	(0.07)	(0.06)
	MSE	0.40	0.54	0.36	0.38	0.49	0.53	0.41
		(0.04)	(0.07)	(0.06)	(0.06)	(0.07)	(0.08)	(0.04)

Table 3.3: Bias ($\times 10$) and MSE ($\times 10^2$) for the estimate of the VUS, with verification bias departing from NMAR model, $n = 100$. (DP: Dirichlet Process, BRL: Bayesian Rank Likelihood, FI: Full Imputation, MSI: Mean Score Imputation, IPW: Inverse Probability Weighted, SPE: Semi-Parametric Efficient, MW: Mann-Whitney U-statistic)

	DP	BRL	FI	MSI	IPW	SPE	MW
	NMAR model 1						
bias	0.01	0.53	0.60	0.51	−0.60	−0.69	0.59
	(0.05)	(0.04)	(0.04)	(0.05)	(0.07)	(0.10)	(0.05)
MSE	0.24	0.44	0.54	0.47	0.88	1.42	0.55
	(0.03)	(0.05)	(0.06)	(0.06)	(0.11)	(0.24)	(0.06)
	NMAR model 2						
bias	0.39	0.36	0.80	0.75	−0.20	−0.65	1.26
	(0.04)	(0.04)	(0.04)	(0.04)	(0.06)	(0.08)	(0.04)
MSE	0.30	0.31	0.80	0.75	0.45	1.09	1.78
	(0.03)	(0.04)	(0.07)	(0.08)	(0.07)	(0.18)	(0.11)

disease is $g_2(S) = g(S; 0, 0.15)$.

2. Let $g(S; p_1, p_2) = \begin{cases} 1, & \text{if } S > S_{(p_1 N)}, \\ p_2, & \text{if } S \leq S_{(p_1 N)}. \end{cases}$

   The verification model for healthy group is $g_0(S) = g(S; 0.7, 0.3)$; the verification model for level-1 disease is $g_1(S) = g(S; 0.6, 0.4)$; the verification model for level-2 disease is $g_2(S) = g(S; 0.7, 0.4)$.

The results are shown in Table 3.3. The DP method has much better accuracy in terms of both bias and MSE. Even if the MAR assumption is not satisfied, the results from DP are much better than MW using only labeled data. This shows that DP is quite robust when the MAR assumption is not satisfied.

3.5.2 Without verification bias

We set $n_0 = n_1 = n_2 = n$ and n is taken to be 50 and 100. Still we consider the 6 different underlying true model as we done in the simulations under verification bias.

Table 3.4: Bias ($\times 10$) and MSE ($\times 10^2$) for the estimate of the VUS without verification bias, $n = 100$ and 50. (BB: Bayesian Bootstrap, BRL: Bayesian Rank Likelihood, EP: Empirical intergration, MW: Mann-Whitney U statistic, K1, K2: two kernel methods)

		BB	BRL	EP	MW	K1	K2	BB	BRL	EP	MW	K1	K2
		Norm 1, VUS = 0.548						Norm 2, VUS = 0.679					
$n = 100$	bias	0.04	0.05	−0.28	−0.04	−0.26	−0.23	0.02	0.05	−0.33	−0.03	−0.35	−0.33
		(0.04)	(0.03)	(0.04)	(0.04)	(0.04)	(0.04)	(0.03)	(0.03)	(0.04)	(0.04)	(0.04)	(0.04)
	MSE	0.15	0.10	0.21	0.15	0.19	0.17	0.09	0.10	0.23	0.14	0.25	0.23
		(0.02)	(0.02)	(0.03)	(0.02)	(0.02)	(0.02)	(0.01)	(0.01)	(0.03)	(0.02)	(0.03)	(0.03)
$n = 50$	bias	0.10	0.12	−0.24	−0.03	−0.32	−0.27	< 0.01	0.10	−0.27	−0.01	−0.39	−0.37
		(0.05)	(0.05)	(0.04)	(0.04)	(0.04)	(0.04)	(0.04)	(0.04)	(0.05)	(0.05)	(0.05)	(0.05)
	MSE	0.26	0.22	0.24	0.20	0.28	0.24	0.18	0.20	0.29	0.24	0.36	0.34
		(0.03)	(0.03)	(0.04)	(0.03)	(0.04)	(0.04)	(0.02)	(0.03)	(0.04)	(0.03)	(0.05)	(0.04)
		Mixnorm 1, VUS = 0.462						Mixnorm 2, VUS = 0.555					
$n = 100$	bias	−0.11	−0.16	−0.27	−0.07	−0.32	−0.26	−0.02	0.30	−0.22	0.02	−0.26	−0.20
		(0.04)	(0.03)	(0.04)	(0.04)	(0.03)	(0.03)	(0.03)	(0.03)	(0.04)	(0.04)	(0.04)	(0.03)
	MSE	0.15	0.15	0.19	0.14	0.22	0.18	0.10	0.20	0.17	0.13	0.19	0.16
		(0.02)	(0.03)	(0.02)	(0.02)	(0.02)	(0.02)	(0.02)	(0.03)	(0.02)	(0.02)	(0.02)	(0.02)
$n = 50$	bias	−0.09	−0.22	−0.28	−0.11	−0.41	−0.33	< 0.01	0.32	−0.26	−0.04	−0.38	−0.31
		(0.05)	(0.05)	(0.05)	(0.05)	(0.05)	(0.04)	(0.05)	(0.05)	(0.05)	(0.05)	(0.05)	(0.05)
	MSE	0.22	0.25	0.32	0.27	0.38	0.31	0.25	0.35	0.32	0.28	0.37	0.30
		(0.03)	(0.03)	(0.04)	(0.04)	(0.04)	(0.04)	(0.04)	(0.05)	(0.04)	(0.04)	(0.05)	(0.04)
		Logis 1, VUS = 0.452						Logis 2, VUS = 0.556					
$n = 100$	bias	< 0.01	0.15	−0.23	−0.04	−0.17	−0.15	0.03	0.15	−0.26	−0.01	−0.32	−0.12
		(0.03)	(0.03)	(0.04)	(0.04)	(0.03)	(0.03)	(0.03)	(0.05)	(0.03)	(0.03)	(0.03)	(0.03)
	MSE	0.12	0.11	0.18	0.14	0.14	0.13	0.11	0.23	0.17	0.12	0.21	0.12
		(0.01)	(0.01)	(0.02)	(0.02)	(0.02)	(0.02)	(0.01)	(0.03)	(0.03)	(0.02)	(0.03)	(0.01)
$n = 50$	bias	0.07	0.43	−0.21	−0.04	−0.19	−0.16	−0.12	−0.09	−0.24	−0.03	−0.42	−0.35
		(0.06)	(0.04)	(0.06)	(0.06)	(0.05)	(0.05)	(0.05)	(0.05)	(0.05)	(0.05)	(0.05)	(0.05)
	MSE	0.31	0.34	0.39	0.37	0.31	0.29	0.30	0.22	0.30	0.27	0.41	0.34
		(0.04)	(0.04)	(0.05)	(0.05)	(0.04)	(0.04)	(0.04)	(0.03)	(0.04)	(0.04)	(0.05)	(0.04)

The results for the BB method will be based on 1000 MCMC resamples. We compare the BB method to two empirical methods and two kernel methods. The first empirical method is proposed by Li and Zhou (2009) by integrating the empirical estimate of the ROC surface, which will be denoted as EP in the table given below. The second empirical method is the unbiased nonparametric Mann-Whitney U-statistic of the probability $\mathrm{P}(X < Y < Z)$ (Dreiseitl et al. 2000), and is extended to the circumstances with ties by Nakas and Yiannoutsos (2004). This method will be denoted as MW in the table. Notice that the empirical estimate of ROC surface is not smooth. The two kernel estimator are proposed by Kang and Tian (2013), which smooth out the surface using Gaussian kernel. The results are given in Table 3.4.

The methods are all very comparable. BB has relatively low bias and MSE in most cases although the difference is insignificant. BRL, as expected, has the best performance for the trinormal setting. MW has the best performance among other estimators and should also be considered as a good choice when estimating the VUS. The problem with MW is that it cannot give a smooth estimate for the ROC surface, which is often desirable.

Note that the speed of the BB method is a lot faster compared with the DP method. In fact, if we run a dataset with 50 observations in each categories using the method under verification bias (implemented in R), it takes approximately 30 minutes. The simplified method without verification bias (implemented in MatLab), however, finish the computation in about 1 second.

3.6 Real data analysis

3.6.1 Epithelial ovarian cancer

Epithelial ovarian cancer (EOC) is one of the most lethal cancers in adult women. CA125 and Human Epididymis protein 4 (HE4) are the most powerful and widely used biomarkers in diagnosing EOC. They are actually the only two markers approved by the FDA for detecting the disease to date (Montagnana et al. 2011). There are a lot of studies which compare the performance of these two biomarkers in detecting EOC. Zheng and Gao (2012); Hamed et al. (2013) claim that HE4 is a more powerful diagnostic tool while Cramer et al. (2011) suggest that CA125 is the best biomarker. However, most studies did not consider the performance of those biomarkers in discriminating the early stage and the late stage of the disease. Since different treatments should be given to different stages of EOC, this discrimination is of interest. Here we will look at their ROC surface to distinguish healthy subjects (may include benign diseased patients), early-stage cancer patients, and late-stage cancer patients.

The dataset we considered is publicly available[2]. This is the experiment data taken from the SPORE/Early Detection Network/Prostate, Lung, Colon, and Ovarian Cancer Ovarian Validation Study. The data contains 703 control cases (156 controls with benign disease, 471 general population controls and 76 controls unsure with benign disease or not), 72 early-stage cases, 78 late-stage cases and 87 cases unverified. We will use this data to compare CA125 and HE4 by applying our two proposed methods along with other methods which can treat verification bias.

To use the method proposed in Chapter 2, recall the trinormality assumption is that, under some strictly monotone increasing transformation H, the transformed observations $Q_i = H(S_i)$, $i = 1, \ldots, N$, satisfy

$$Q_i | \{D_i = k\} \overset{\text{i.i.d.}}{\sim} \mathrm{N}(\mu_k, \sigma_k^2), \quad k = 0, 1, 2, \tag{3.23}$$

for some $\mu_0 < \mu_1 < \mu_2$ and $\sigma_0, \sigma_1, \sigma_2 > 0$. To ensure the identifiability of the model, the distribution of the middle group (without loss of generality) has been set to be the standard normal (i.e. $\mu_1 = 0$, $\sigma_1 = 1$).

To do a quick visual check on the trinormality assumption, we can compose the transformation function $H(x) = \Phi^{-1} \circ F_1(x)$ where $F_1(x)$ can be estimated using a empirical distribution function of $\{S_i : L_i = 1\}$. We can then apply this transformation on S_i for $i = 1, \ldots, N$. Denote the transformed observations by $\hat{Q}_i$. The transformed observations $\{\hat{Q}_i : L_i = 1\}$ should follow a standard normal distribution so if we plot the quantiles of $\{\hat{Q}_i : L_i = 1\}$ versus the quantiles of a standard normal distribution, it should follow the $y = x$ line. If the trinormality assumption holds, the transformed observations $\{\hat{Q}_i : L_i = 0\}$ should follow a normal distribution as well. So if we plot the quantiles of $\{\hat{Q}_i : L_i = 0\}$ versus the quantiles of a standard normal distribution, it should also follow a straight line. Similarly

[2]https://edrn.nci.nih.gov/protocols/119-spore-edrn-pre-plco-ovarian-phase-ii-validation

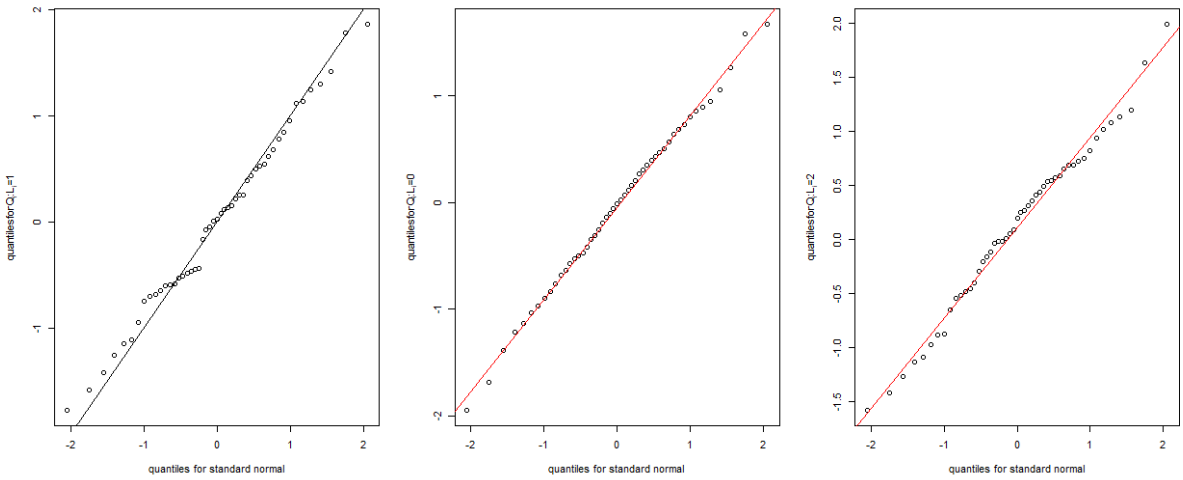

Figure 3.1: Trinormality check for CA125

for $\{\hat{Q}_i : L_i = 2\}$. The result is shown in Figure 3.1 and 3.2.

Judging from the plots, the trinormality assumption seems to be approximately satisfied for both CA125 and HE4. So we can apply the BRL method to estimate the ROC surface.

Before estimating ROC surface, we did a quick check on the data by plotting out the boxplot of the labeled observations of CA125 and HE4 within each class. Judging from the plots 3.3 and 3.4, CA125 and HE4 scores increase with the severity of the disease.

Table 3.5 summarizes the estimates of the VUS for CA125 and HE4. The DP and BRL methods are both based on 80000 iterations (100000 iterations with the first 20000 iterations as burn-in). The VUS for CA125 is estimated to be approximately 0.36, while the estimate given by FI is slightly larger than others. The VUS for HE4 is estimated to be around 0.44, with a slightly smaller estimate given by BRL. The estimates given by different methods are quite similar overall.

It seems that the VUS is slightly larger for HE4 but it is hard to conclude which biomarker is better, since the confidence intervals intersect with one another. To better compare the fitted ROC surface using all those methods, we also plot out the ROC surface.

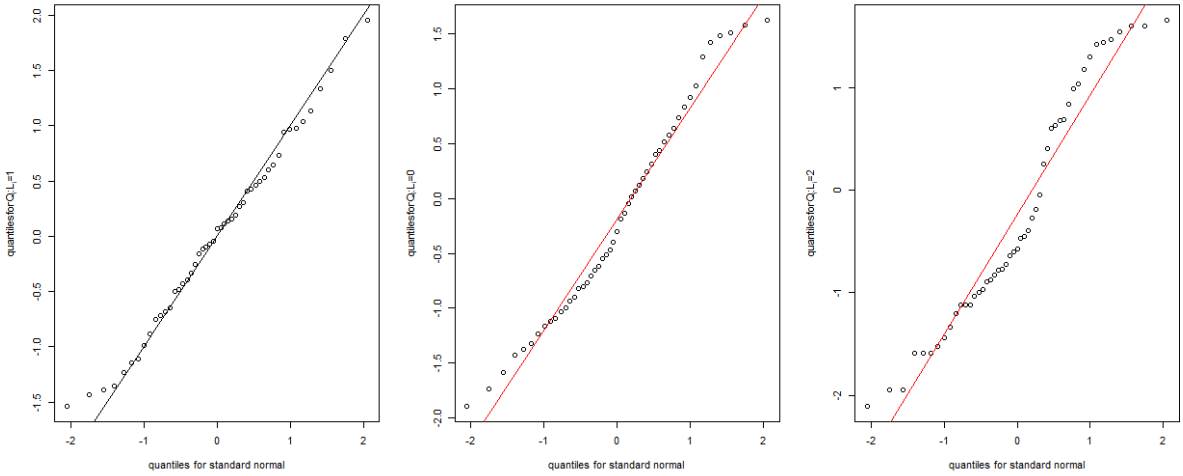

Figure 3.2: Trinormality check for HE4

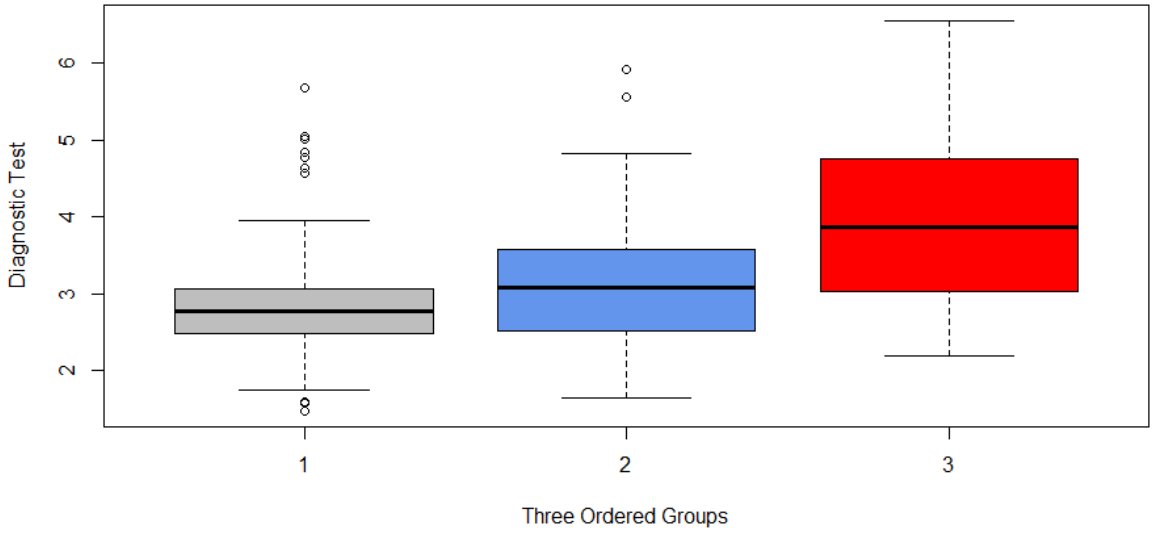

Figure 3.3: Boxplot for CA125

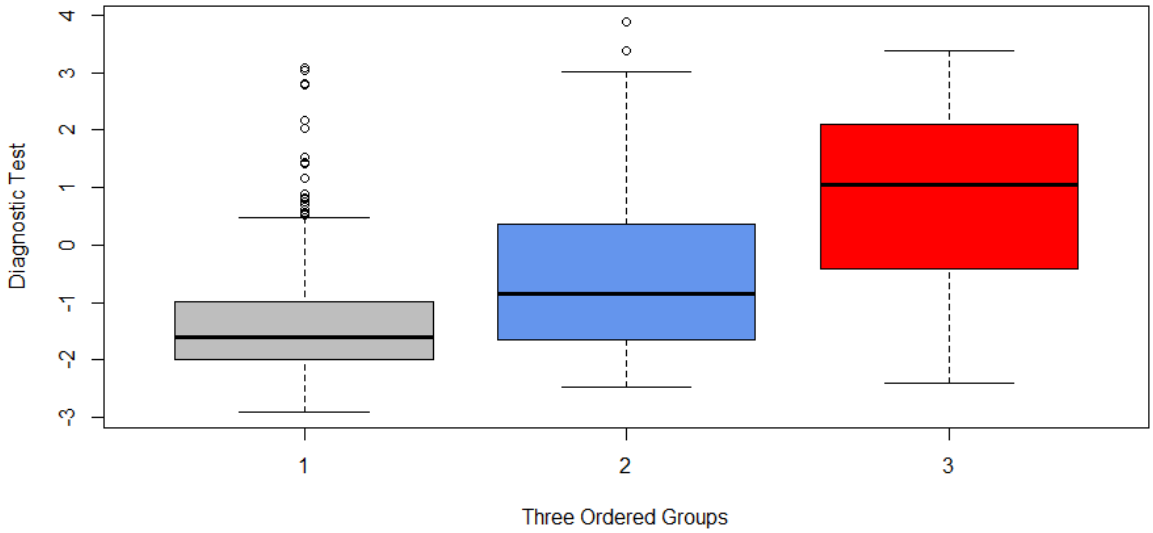

Figure 3.4: Boxplot for HE4

Table 3.5: VUS estimates for CA125 and HE4

	CA125			HE4		
	estimated VUS	sd	95% C.I.	estimated VUS	sd	95% C.I.
DP	0.355	0.033	[0.291, 0.420]	0.450	0.037	[0.378, 0.522]
BRL	0.369	0.029	[0.311, 0.426]	0.414	0.030	[0.354, 0.473]
FI	0.427	0.030	[0.369, 0.484]	0.441	0.029	[0.385, 0.497]
MSI	0.365	0.034	[0.298, 0.432]	0.438	0.033	[0.373, 0.502]
IPW	0.356	0.035	[0.286, 0.425]	0.453	0.036	[0.382, 0.524]
SPE	0.354	0.035	[0.285, 0.422]	0.449	0.034	[0.382, 0.517]

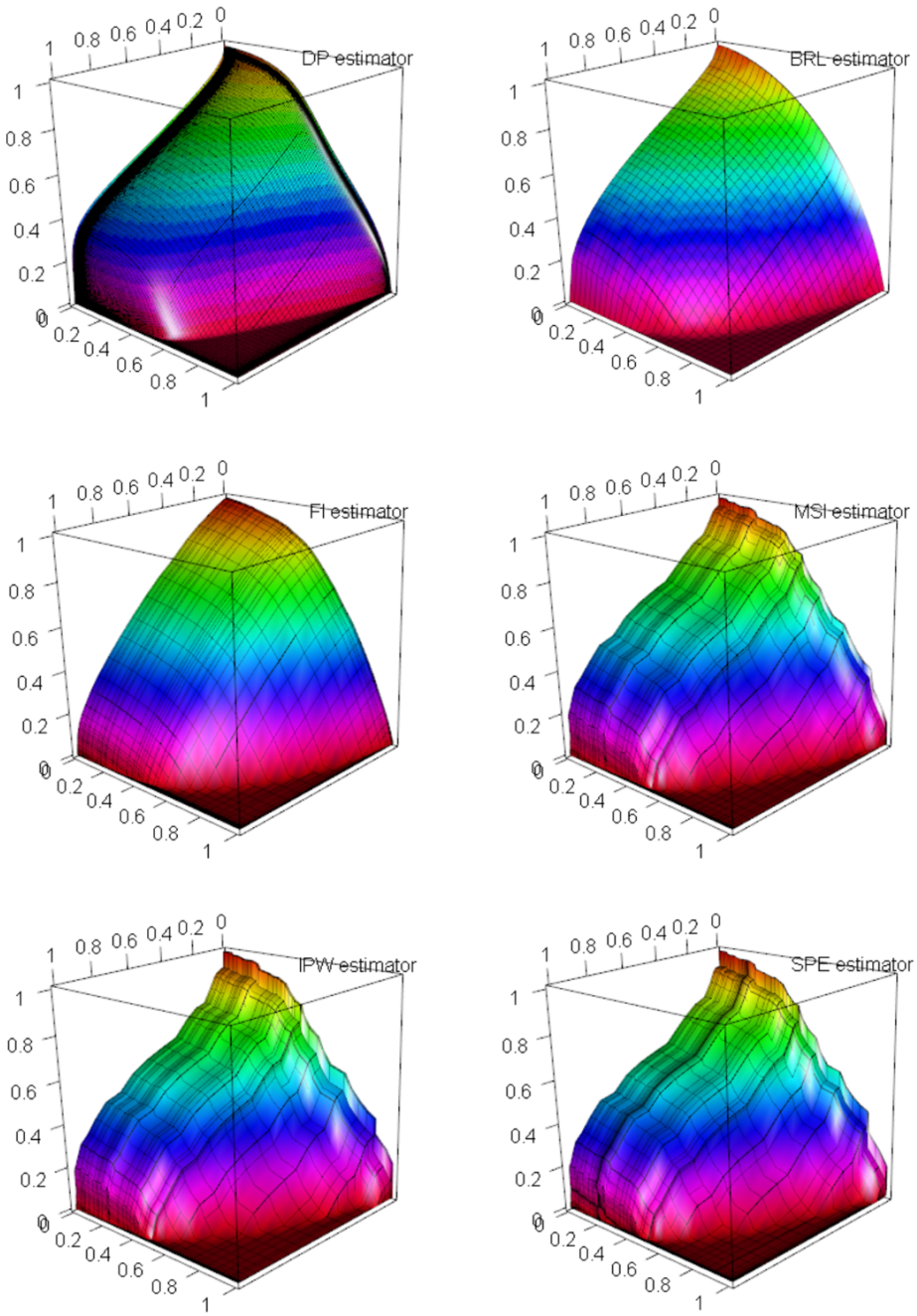

Figure 3.5: Estimated ROC surfaces for CA125

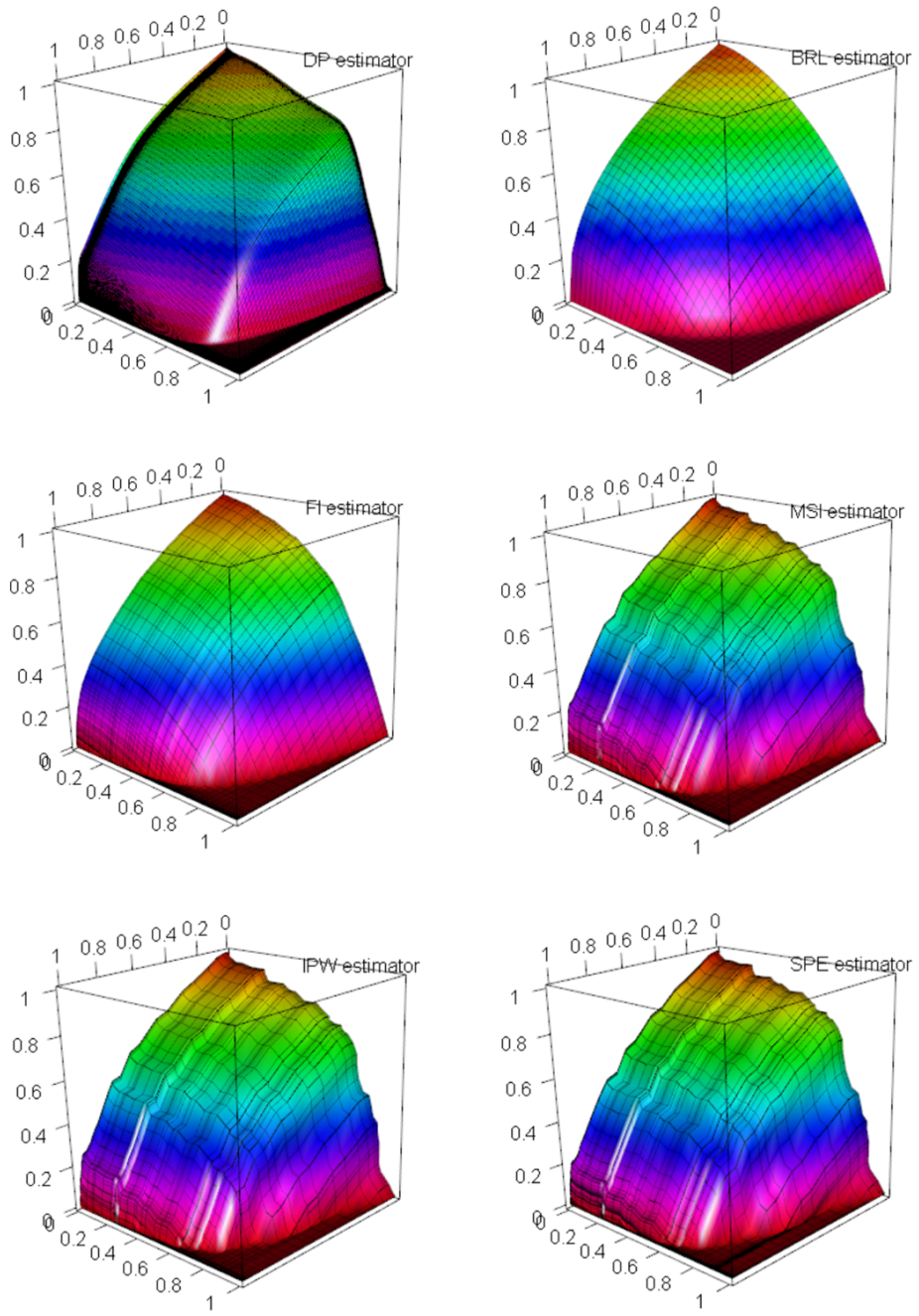

Figure 3.6: Estimated ROC surfaces for HE4

We can see from Figures 3.5 and 3.6 that the DP and BRL methods generate smooth surfaces while others do not. Judging from the plots, the ROC surfaces estimated by MSI, IPW and SPE are very much alike and the ROC surfaces generated by DP are like a smoother version of those estimated surfaces. The surfaces estimated by BRL, on the other hand, seem to have smoothed out too many details, and for some reasons, look more like the surfaces generated by FI. The figures give us an rough impression that HE4 has better diagnosis ability than CA125.

3.6.2 Hepatocellular carcinoma

Hepatocellular carcinoma (HCC) is the one of the most common type of liver cancer. It is the fifth most common tumor worldwide and the second most common cause of cancer-related death(Heimbach et al. 2018). HCC usually occurs in people with chronic liver diseases, for example cirrhosis caused by hepatitis B or hepatitis C.

Albumin is an important serum protein in human body. Studies have shown that serum albumin levels have prognostic significance in HCC, with lower serum albumin levels having significantly larger maximum tumor diameters(Carr and Guerra 2017; Bağırsakçı et al. 2017; Tanriverdi 2014). In fact, serum albumin level is already being used as a criterion for the HCC staging systems.

Here, we want to apply our methods to assess the ability of serum albumin level in distinguishing different stages of HCC. The dataset we are using is publicly available on Kaggle[3]. The dataset contains 159 records in total with 2 patients in the very early stage, 29 patients in the early stage, 39 patients in the intermediate stage, 53 patients in the advanced stage, 37 patients in the terminal stage and 5 patients unverified. We combine the very early stage and early stage, intermediate stage and advanced stage to get three outcomes in

[3]https://www.kaggle.com/mrsantos/hcc-dataset/version/5

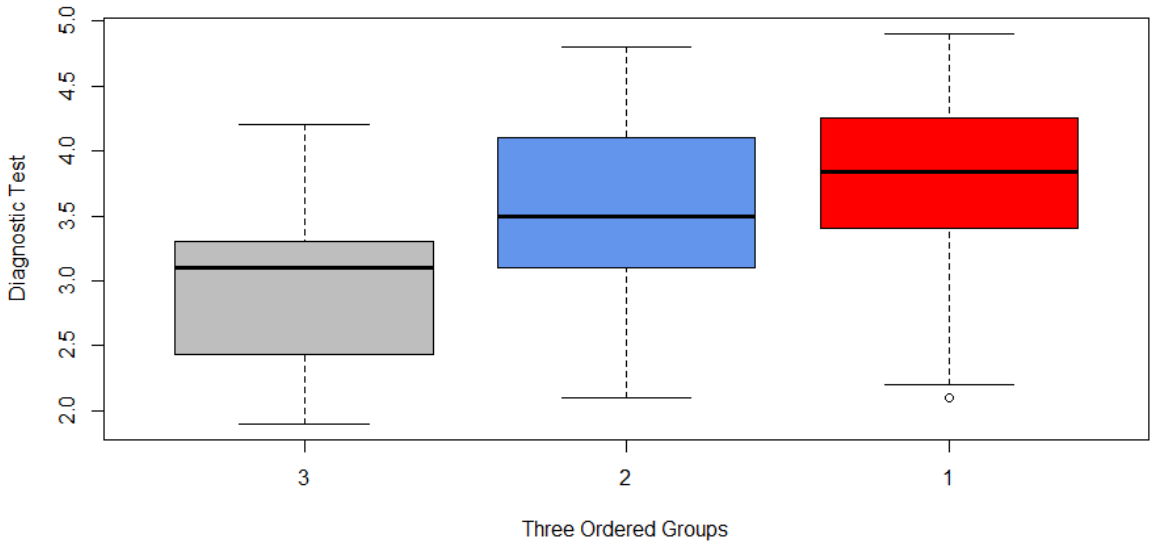

Figure 3.7: Boxplot for serum albumin

total. The dataset contains the serum albumin levels for every patient as well as a covariate, gender. A quick check of the Figure 3.7 suggests that the serum albumin is lower for later stages of hepatocellular carcinoma.

There are 29 females and 130 males in this dataset. (The male-to-female incidence ratio varies between 2:1 to 4:1 in the population(Liu et al. 2017).) We want to utilize the covariate information when we are assessing this biomarker. We then use the extension of our proposed method to incorporate covariates when estimating the ROC surface. The other methods for estimating the ROC surface under verification bias cannot incorporate covariates, so, we only use the DP method for illustration. The results is shown in Table 3.6. We calculate the VUS for the covariate-specific ROC surface as well as for the overall ROC surface.

The ROC surface plots are shown in Figure 3.8. We can see that the ROC surfaces for female and male are slightly different. The overall ROC surface looks more like the ROC

Table 3.6: The VUS estimates for serum albumin

	estimated VUS	sd	95% C.I.
Female	0.374	0.123	$[0.133, 0.615]$
Male	0.376	0.040	$[0.298, 0.455]$
Overall	0.376	0.045	$[0.291, 0.461]$

surface for male which make sense since the ratio of male patients is much larger than the ratio of female patients.

Our method confirms that serum albumin has a prognostic ability to identify the stage of HCC. Its VUS for distinguishing different stages of HCC is around 0.406. However, serum albumin alone does not appear to be sufficient for staging the HCC.

3.7 Discussion

Our methods can be easily extended to estimate the ROC hypersurface which is defined when there are more than three categories to be classified. This is originally defined by Scurfield (1996)

$$\{(F_1(c_1), F_2(c_2)-F_2(c_1), F_3(c_3)-F_3(c_2), \ldots, F_{d-1}(c_{d-1})-F_{d-1}(c_{d-2}), 1-F_d(c_{d-1})) : \\ c_1 < c_2 < \cdots < c_{d-1}\}$$

for a d-dimensional ROC hypersurface, where F_i denote the true distribution for measurements from the ith category, $i = 1, 2, \ldots, d$. Similar to the AUC and the VUS, he proposed using hypervolume under the ROC manifold (HUM) as a measure of the performance of a classifier, and HUM can be calculated as $\mathrm{HUM} = \mathrm{P}(X^{(1)} < X^{(2)} < \cdots < X^{(d)})$, where $X^{(i)}$ denote the measurement from the ith category. With the knowledge of the true distributions $F_1, F_2, \ldots, F_d$, it can also be calculated as $\mathrm{HUM} = \mathrm{P}(F_1^{-1}(U_1) < F_2^{-1}(U_2) < \cdots < F_d^{-1}(U_d))$, where

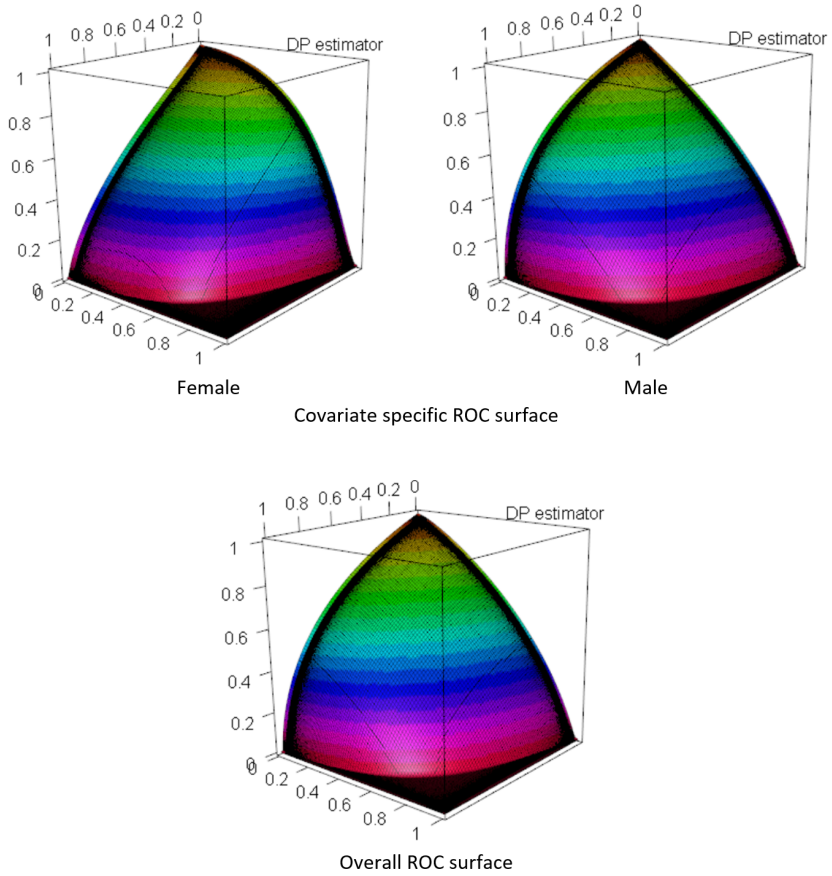

Figure 3.8: Estimated ROC surfaces for serum albumin

$U_1, \ldots, U_2 \overset{\text{i.i.d.}}{\sim} \mathrm{U}(0,1)$. Through some simple modifications, our methods introduced above can be extended to estimate $F_1, F_2, \ldots, F_d$ with the measurements from d categories. We can then give an estimate of the ROC hypersurface as well as the HUM.

CHAPTER

4

BAYESIAN SEMI-SUPERVISED LEARNING

4.1 Introduction

Semi-supervised learning is a classification method which makes use of both labeled data and unlabeled data for training. As introduced in Chapter 1, semi-supervised learning can be classified into two big class – generative methods and discriminative methods. Generative methods based on EM algorithm have two widely known disadvantages: First, some assumptions on the underlying distributions have to be made. If the assumptions are wrong, then the mislabeled data will hurt the accuracy (Cozman et al. 2003). Second, EM algorithm has a tendency to stick to a local maximum instead of the global maximum. This

may also cause problems when using unlabeled data.

In this chapter, we propose a new generative method for semi-supervised learning which can solve the two problems mentioned. First of all, we make a flexible semi-parametric modeling assumption. Similar to the method introduced in 2, we assume that the two underlying distributions map to multivariate normal distributions after an unknown same transformation applied on each class. This is a lot more general than a specific parametric assumption. This is called the Nonparanormal model in a graphical model (Liu et al. 2012; Mulgrave and Ghosal 2018). To estimate the transformation, we shall use B-spline of the transformation, so this will be reduced to parameter estimation. Secondly, because we obtain the posterior distributions instead of point estimates given by the EM algorithm, we use the Gibbs Sampling framework which prevents the problem of trapping at a local maximum.

The rest of the chapter is organized as follows. We describe the model and the algorithm given by Gibbs Sampling in Section 4.2. We give a method of selecting hyperparameters in Section 4.4. Then we present results of a simulation study to compare our method with other semi-parametric methods in Section 4.5. Finally we apply our method to real data sets in Section 4.6.

4.2 Model

Below we shall use the following notation list: $\mathrm{N}_p(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ denote a p dimensional normal distribution with mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$; $\Phi(\cdot; \boldsymbol{\mu}, \boldsymbol{\Sigma})$ and $\phi(\cdot; \boldsymbol{\mu}, \boldsymbol{\Sigma})$ denote the cumulative distribution and the density function of normal distribution with mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$ respectively; $\mathrm{TN}_p(\boldsymbol{\mu}, \boldsymbol{\Sigma}, \mathscr{T})$ stands for a normal distribution $\mathrm{N}_p(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ restricted on a set $\mathscr{T}$; $\mathrm{W}(n, V)$ stands for the Wishart distribution with n degree of freedom and scale matrix V; $\mathrm{Beta}(\alpha, \beta)$ stands for Beta distribution with shape parameters α and β.

Suppose we observe $\boldsymbol{X}^{(1)}, \ldots, \boldsymbol{X}^{(n)}$ independently, which take value in $\mathbb{R}^p$ for some $p \geq 1$, each observation as $\boldsymbol{X}^{(i)} = (X_1^{(i)}, \ldots, X_p^{(i)})$. We denote the class label for $\boldsymbol{X}^{(i)}$ by $L^{(i)} \in \{0, 1\}$. The observation belongs to the Class 0 if $L^{(i)} = 0$ and it belongs to the Class 1 otherwise, $i = 1, \ldots, n$. Notice that in the semi-supervised learning settings, not all $L^{(i)}$ are available to us. If i-th label is missing, then we set $L^{(i)} = 2$. When we talk about a generic observation, we omit the index i from $\boldsymbol{X}^{(i)}$ and just writes $\boldsymbol{X}$. We assume that under some unknown increasing transformation f, the transformed observations follow one of the two normal distributions according to their labels,

$$\begin{aligned} \boldsymbol{f}(\boldsymbol{X}) | \{L = 0\} &\sim \mathrm{N}_p(\boldsymbol{\mu_0}, \boldsymbol{\Sigma_0}), \\ \boldsymbol{f}(\boldsymbol{X}) | \{L = 1\} &\sim \mathrm{N}_p(\boldsymbol{\mu_1}, \boldsymbol{\Sigma_1}). \end{aligned} \tag{4.1}$$

where $\boldsymbol{f}$ is a p dimensional vector of functions, $\boldsymbol{f}(X_1, \ldots, X_p) = (f_1(X_1), \ldots, f_p(X_p))$. Notice that any continuous random variable can be transformed to a normal variable by a strictly increasing transformation and hence is not an assumption if considered individually. The model assumption here is that the two distributions for two classes are mapped to normal distributions under the same transformation. This is called nonparanormality assumption in the literature.

The method we use to estimate the transformation $\boldsymbol{f}$ is very similar to the method used by Mulgrave and Ghosal (2018). However the purpose of estimating this transformation is completely different. Here we are estimating the transformation $\boldsymbol{f}$ for semi-supervised learning, while Mulgrave and Ghosal (2018) used this approach to learn the graphical structure. Like Mulgrave and Ghosal (2018), we shall estimate each component of the transformation $\boldsymbol{f}$. We denote the d-th dimension of the transformation by f_d. We put prior distributions on the unknown transformation functions through a random series based on

B-splines, i.e.,

$$f_d(X_d) = \sum_{j=1}^{J} \theta_{dj} B_j(X_d), \quad d = 1, \ldots, p, \tag{4.2}$$

where X_d is the d-th dimension of an observation, $B_j(\cdot)$ are the B-spline basis functions, θ_{dj} is a coefficient in the expansion of the function, $j = 1, \ldots, J$, and J is the number of B-spline basis functions with equispaced knots used in the expansion. The coefficients are ordered to induce monotonicity, and the smoothness is controlled by the order of the B-splines and the number of basis functions used in the expansion. The posterior means of the coefficients give a monotone smooth Bayes estimate of the transformations. In this chapter, we choose cubic splines, which correspond to B-splines of order 4.

Notice that for B-spline functions, $\boldsymbol{X}$ can only take values in the range of $[0, 1]$. Because of that, we have to do some transformation to the data if it is not in this range. For example, we can calculate the mean μ_d and variance σ_d^2 for dimension d of the training data, and then apply the cumulative distribution function $\Phi(\cdot; \mu_d, \sigma_d^2)$ on the d-th dimension of the data, $d = 1, \ldots, p$.

4.2.1 Prior distributions

Prior on the B-spline coefficients

The prior we use here directly follows Mulgrave and Ghosal (2018). Here we include the prior specification for contiguity.

Let us first put the monotonicity aside, and put normal prior on the coefficients of the B-splines, i.e., $\boldsymbol{\theta}_d = (\theta_{d1}, \ldots, \theta_{dJ}) \sim \mathrm{N}_J(\boldsymbol{\zeta}, \sigma^2 \boldsymbol{I})$, where σ^2 is some positive constant, $\boldsymbol{\zeta}$ is a J dimensional vector of constants, and $\boldsymbol{I}$ is the $J \times J$ identity matrix. We choose normal prior for conjugacy. Because the means and the covariances of the normal distributions are unknown, this gives the flexibility in the location and the scale of the transformation

and causes identifiability issues. To address this and to retain the conjugacy normal prior, impose the following constraints on the locations and the scales of the transformation function f_d:

$$\begin{cases} 0 = f_d(1/2) = \sum_{j=1}^{J} \theta_{dj} B_j(1/2), \\ 1 = f_d(3/4) - f_d(1/4) = \sum_{j=1}^{J} \theta_{dj}[B_j(3/4) - B_j(1/4)]. \end{cases} \tag{4.3}$$

The constriants can be written in matrix form $\boldsymbol{A}\boldsymbol{\theta}_d = \boldsymbol{c}$, where

$$\boldsymbol{A} = \begin{pmatrix} B_1(1/2) & B_2(1/2) & \cdots & B_J(1/2) \\ B_1(3/4) - B_1(1/4) & B_2(3/4) - B_2(1/4) & \cdots & B_J(3/4) - B_J(1/4) \end{pmatrix}$$

and $\boldsymbol{c} = (0,1)^T$.

By the property of the normal distribution, we have

$$\boldsymbol{\theta}_d | \{\boldsymbol{A}\boldsymbol{\theta}_d = \boldsymbol{c}\} \sim \mathrm{N}_J(\boldsymbol{\xi}, \boldsymbol{\Gamma}), \tag{4.4}$$

where

$$\boldsymbol{\xi} = \boldsymbol{\zeta} + \boldsymbol{A}^T(\boldsymbol{A}\boldsymbol{A}^T)^{-1}(\boldsymbol{c} - \boldsymbol{A}\boldsymbol{\zeta}),$$
$$\boldsymbol{\Gamma} = \sigma^2[\boldsymbol{I} - \boldsymbol{A}^T(\boldsymbol{A}\boldsymbol{A}^T)^{-1}\boldsymbol{A}].$$

Because of the two linear constraints, the covariance matrix $\boldsymbol{\Gamma}$ is actually singular. So we remove two coefficients by representing them as linear combinations of the others. Here we choose J_1 where $B_{J_1}(1/2)$ is the largest (middle one if J is odd, either of middle two if J is even) and J_2 where $B_{J_2}(3/4) - B_{J_2}(1/4)$ is the largest (upper 75th quantile one) and that $J_2 \neq J_1$. In principal, any J_1 and J_2 with non-zero coefficients can work. Here we choose them in this way to guarantee numerical stability in later calculation. Then by $\boldsymbol{A}\boldsymbol{\theta}_d = \boldsymbol{c}$ we

can get

$$\begin{pmatrix} \theta_{dJ_1} \\ \theta_{dJ_2} \end{pmatrix} = \boldsymbol{W}_d \bar{\boldsymbol{\theta}}_d + \boldsymbol{q}_d, \tag{4.5}$$

where $\bar{\boldsymbol{\theta}}_d$ is the reduced vector after removing θ_{dJ_1} and θ_{dJ_2} from $\boldsymbol{\theta}_d$, and $\boldsymbol{W}_d$ and $\boldsymbol{q}_d$ can be calculated correspondingly.

The reduced vector $\bar{\boldsymbol{\theta}}_d$ follows the prior distribution:

$$\bar{\boldsymbol{\theta}}_d | \{\boldsymbol{A}\boldsymbol{\theta}_d = \boldsymbol{c}\} \sim \mathrm{N}_{J-2}(\bar{\boldsymbol{\xi}}, \bar{\boldsymbol{\Gamma}}) \tag{4.6}$$

where $\bar{\boldsymbol{\xi}}$ and $\bar{\boldsymbol{\Gamma}}$ are obtained by removing the J_1 and J_2 dimension correspondingly.

Finally consider the monotonicity constraint $\theta_{d1} < \theta_{d2} < \cdots < \theta_{dJ}$. Written in matrix form, that is, $\boldsymbol{F}\boldsymbol{\theta}_d > 0$, where

$$\boldsymbol{F} = \begin{pmatrix} -1 & 1 & 0 & \cdots & 0 & 0 \\ 0 & -1 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & -1 & 1 \end{pmatrix}.$$

This can be further reduced to $\bar{\boldsymbol{F}}\bar{\boldsymbol{\theta}}_d + \bar{\boldsymbol{g}} > 0$ according to (4.5).

The final prior is given by

$$\bar{\boldsymbol{\theta}}_d | \{\boldsymbol{A}\boldsymbol{\theta}_d = c\} \sim \mathrm{TN}_{J-2}(\bar{\boldsymbol{\xi}}, \bar{\boldsymbol{\Gamma}}, \mathscr{T}) \tag{4.7}$$

where $\mathscr{T} = \{\bar{\boldsymbol{\theta}}_d : \bar{\boldsymbol{F}}\bar{\boldsymbol{\theta}}_d + \bar{\boldsymbol{g}} > 0\}$. Notice that this prior reserves the conjugacy.

The parameter $\boldsymbol{\zeta}$ is chosen to be $\zeta_j = \nu + \tau\Phi^{-1}\left(\frac{j-0.375}{J-0.75+1}\right)$, $j = 1, \ldots, J$, where ν is a constant, τ is a positive constant, and Φ^{-1} is the inverse of the cumulative distribution of the standard normal distribution. The motivation behind this prior specification is that

this is an approximation for the expected values of the order statistics of a $\mathrm{N}(\nu, \tau^2)$ random variable.

Prior on the means and covariances

Because the means and covariances corresponds to transformed measurements, it's hard to obtain an prior information. We put an noninformative prior, i.e., $\pi(\boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0) \propto 1$, $\pi(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1) \propto 1$.

4.3 Posterior computation

Let $\boldsymbol{Y}$ be the transformed observations, i.e., $\boldsymbol{Y}_d^{(i)} = f_d(\boldsymbol{X}_d^{(i)}) = \boldsymbol{B}_d^{(i)}\boldsymbol{\theta}_d$, where $\boldsymbol{B}_d^{(i)} = (B_1(X_d^{(i)}), \ldots, B_J(X_d^{(i)}))$, $i = 1, \ldots, n$. Because of (4.5), we can calculate $\boldsymbol{Y}^{(i)}$ based on $\bar{\boldsymbol{\theta}}_d$, i.e.,

$$\boldsymbol{Y}_d^{(i)} = (\bar{\boldsymbol{B}}_d^{(i)} + \boldsymbol{B}_d^{(i)*}\boldsymbol{W}_d)\bar{\boldsymbol{\theta}}_d + \boldsymbol{B}_d^{(i)*}\boldsymbol{q}_d, \tag{4.8}$$

where $\bar{\boldsymbol{B}}_d^{(i)}$ is obtained by removing J_1 and J_2 column of $\boldsymbol{B}_d^{(i)}$, and $\boldsymbol{B}_d^{(i)*}$ is $(B_{J_1}(X_d^{(i)}), B_{J_2}(X_d^{(i)}))$.

According to assumption above, the full posterior distribution is

$$\begin{aligned}
\pi^*(\bar{\boldsymbol{\theta}}_1, \ldots, \bar{\boldsymbol{\theta}}_d, \boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0, \boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1) \propto & \prod_{L^{(i)}=0} (\det \boldsymbol{\Sigma}_0)^{-1/2} \exp(-\frac{1}{2}(\boldsymbol{Y}^{(i)} - \boldsymbol{\mu}_0)^T \boldsymbol{\Sigma}_0^{-1}(\boldsymbol{Y}^{(i)} - \boldsymbol{\mu}_0)) \\
& \times \prod_{L^{(i)}=1} (\det \boldsymbol{\Sigma}_1)^{-1/2} \exp(-\frac{1}{2}(\boldsymbol{Y}^{(i)} - \boldsymbol{\mu}_1)^T \boldsymbol{\Sigma}_1^{-1}(\boldsymbol{Y}^{(i)} - \boldsymbol{\mu}_1)) \\
& \times \pi(\boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0) \times \pi(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1) \times \prod_{d=1}^{p} \pi_d(\bar{\boldsymbol{\theta}}_d),
\end{aligned} \tag{4.9}$$

where $\pi_d(\bar{\theta}_d)$ is the prior distribution of $\bar{\theta}_d$ given by (4.7).

4.3.1 Gibbs sampling algorithm

Before doing the Gibbs sampling, we need to assign some initial values. We assign the initial values for B-spline coefficients $\bar{\boldsymbol{\theta}}_1, \ldots, \bar{\boldsymbol{\theta}}_p$ by assuming the true transformation is $\Phi^{-1}(\cdot)$. This initial values will also be useful when sampling from the posterior truncated normal distributions. After finding the initial values $\bar{\boldsymbol{\theta}}_d$, we can calculate the initial value for $\boldsymbol{Y}$ according to (4.8). The initial values for $\boldsymbol{\mu}_0$ and $\boldsymbol{\mu}_1$ are

$$\begin{aligned}
\boldsymbol{\mu}_0 &= \sum_{i=1}^{n} \mathbb{1}\{L_{\text{org}}^{(i)} = 0\}\boldsymbol{Y}^{(i)} / \sum_{i=1}^{n} \mathbb{1}\{L_{\text{org}}^{(i)} = 0\}, \\
\boldsymbol{\mu}_1 &= \sum_{i=1}^{n} \mathbb{1}\{L_{\text{org}}^{(i)} = 1\}\boldsymbol{Y}^{(i)} / \sum_{i=1}^{n} \mathbb{1}\{L_{\text{org}}^{(i)} = 1\},
\end{aligned}$$

where $\boldsymbol{L}_{\text{org}}$ stands for the original label. The original value for the label $\boldsymbol{L}$ is given as follows: for $i = 1, \ldots, n$,

$$L^{(i)} = \begin{cases} L_{\text{org}}^{(i)}, & \text{if } L_{\text{org}}^{(i)} \neq 2, \\ 0, & \text{if } L_{\text{org}}^{(i)} = 2 \text{ and } \|\boldsymbol{Y}^{(i)} - \boldsymbol{\mu}_0\| < \|\boldsymbol{Y}^{(i)} - \boldsymbol{\mu}_1\|, \\ 1, & \text{if } L_{\text{org}}^{(i)} = 2 \text{ and } \|\boldsymbol{Y}^{(i)} - \boldsymbol{\mu}_0\| > \|\boldsymbol{Y}^{(i)} - \boldsymbol{\mu}_1\|, \end{cases} \tag{4.10}$$

where $\|\cdot\|$ stands for the Euclidean distance. Then we can calculate the initial values for $\boldsymbol{\Sigma}_0$ which is the covariance matrix of $\boldsymbol{Y}$ whose initial $L = 0$, and similarly the initial values for $\boldsymbol{\Sigma}_1$ is the covariance matrix of the $\boldsymbol{Y}$ whose initial $L = 1$.

1. First sample the B-spline coefficients for $d = 1, \ldots, p$.

The joint posterior for the B-spline coefficients is a truncated normal with density

$$\pi^*(\bar{\boldsymbol{\theta}}_1,\ldots,\bar{\boldsymbol{\theta}}_p|\ \boldsymbol{Y},\boldsymbol{\mu}_0,\boldsymbol{\Sigma}_0,\boldsymbol{\mu}_1,\boldsymbol{\Sigma}_1) \tag{4.11}$$

$$\propto \prod_{L^{(i)}=0}(\det\boldsymbol{\Sigma}_0)^{-1/2}\exp(-\frac{1}{2}(\boldsymbol{Y}^{(i)}-\boldsymbol{\mu}_0)^T\boldsymbol{\Sigma}_0^{-1}(\boldsymbol{Y}^{(i)}-\boldsymbol{\mu}_0))$$

$$\times\prod_{L^{(i)}=1}(\det\boldsymbol{\Sigma}_1)^{-1/2}\exp(-\frac{1}{2}(\boldsymbol{Y}^{(i)}-\boldsymbol{\mu}_1)^T\boldsymbol{\Sigma}_1^{-1}(\boldsymbol{Y}^{(i)}-\boldsymbol{\mu}_1))$$

$$\times\prod_{d=1}^{p}\pi_d(\bar{\boldsymbol{\theta}}_d). \tag{4.12}$$

Let $\bar{\boldsymbol{\theta}}_{-d}$ denote the vector $(\bar{\boldsymbol{\theta}}_1,\ldots,\bar{\boldsymbol{\theta}}_{d-1},\bar{\boldsymbol{\theta}}_{d+1},\ldots,\bar{\boldsymbol{\theta}}_p)$. We sample $\bar{\boldsymbol{\theta}}_d$ according to

$$\begin{aligned}
&\pi^*(\bar{\boldsymbol{\theta}}_d|\boldsymbol{Y},\bar{\boldsymbol{\theta}}_{-d},\boldsymbol{\mu}_0,\boldsymbol{\Sigma}_0,\boldsymbol{\mu}_1,\boldsymbol{\Sigma}_1)\\
&\quad\propto \exp\{-\frac{1}{2}\bar{\boldsymbol{\theta}}_d^T[(\boldsymbol{\Sigma}_0)^{-1}_{dd}\sum_{L^{(i)}=0}(\bar{\boldsymbol{B}}_d^{(i)}+\boldsymbol{B}_d^{(i)*}\boldsymbol{W}_d)^T(\bar{\boldsymbol{B}}_d^{(i)}+\boldsymbol{B}_d^{(i)*}\boldsymbol{W}_d)\\
&\qquad+(\boldsymbol{\Sigma}_1)^{-1}_{dd}\sum_{L^{(i)}=1}(\bar{\boldsymbol{B}}_d^{(i)}+\boldsymbol{B}_d^{(i)*}\boldsymbol{W}_d)^T(\bar{\boldsymbol{B}}_d^{(i)}+\boldsymbol{B}_d^{(i)*}\boldsymbol{W}_d)+\bar{\boldsymbol{\Gamma}}^{-1}]\bar{\boldsymbol{\theta}}_d\\
&\qquad+\bar{\boldsymbol{\theta}}_d^T\{\bar{\boldsymbol{\xi}}\bar{\boldsymbol{\Gamma}}^{-1}-\sum_{L^{(i)}=0}(\bar{\boldsymbol{B}}_d^{(i)}+\boldsymbol{B}_d^{(i)*}\boldsymbol{W}_d)[(\boldsymbol{\Sigma}_0)^{-1}_{dd}(\boldsymbol{B}_d^{(i)*}\boldsymbol{q}-\boldsymbol{\mu}_{0d})\\
&\qquad+(\boldsymbol{\Sigma}_0)^{-1}_{d,-d}(\boldsymbol{Y}^{(i)}_{-d}-\boldsymbol{\mu}_{0,-d})]-\sum_{L^{(i)}=1}(\bar{\boldsymbol{B}}_d^{(i)}+\boldsymbol{B}_d^{(i)*}\boldsymbol{W}_d)[(\boldsymbol{\Sigma}_1)^{-1}_{dd}(\boldsymbol{B}_d^{(i)*}\boldsymbol{q}-\boldsymbol{\mu}_{1d})\\
&\qquad+(\boldsymbol{\Sigma}_1)^{-1}_{d,-d}(\boldsymbol{Y}^{(i)}_{-d}-\boldsymbol{\mu}_{1,-d})]\}\}\times\mathbb{1}\{\bar{\boldsymbol{F}}\bar{\boldsymbol{\theta}}_d+\bar{\boldsymbol{g}}>0\}.
\end{aligned} \tag{4.13}$$

The truncated normal distributions are sampled using the method proposed by Li and Ghosh (2015). After obtaining the posterior samples of $\bar{\boldsymbol{\theta}}_1,\ldots,\bar{\boldsymbol{\theta}}_p$, we can update $\boldsymbol{Y}$ according to (4.8).

2. Update $\boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0, \boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1$ according to their posterior probability. That is,

$$
\begin{aligned}
\boldsymbol{\Sigma}_0^{-1} &\sim \mathrm{W}(n_0, \sum_{L^{(i)}=0} (\boldsymbol{Y}^{(i)} - \bar{\boldsymbol{Y}}_{0\cdot})(\boldsymbol{Y}^{(i)} - \bar{\boldsymbol{Y}}_{0\cdot})^T), \\
\boldsymbol{\mu}_0 &\sim \mathrm{N}(\bar{\boldsymbol{Y}}_{0\cdot}, \boldsymbol{\Sigma}_0/n_0), \\
\boldsymbol{\Sigma}_1^{-1} &\sim \mathrm{W}(n_1, \sum_{L^{(i)}=1} (\boldsymbol{Y}^{(i)} - \bar{\boldsymbol{Y}}_{1\cdot})(\boldsymbol{Y}^{(i)} - \bar{\boldsymbol{Y}}_{1\cdot})^T), \\
\boldsymbol{\mu}_1 &\sim \mathrm{N}(\bar{\boldsymbol{Y}}_{1\cdot}, \boldsymbol{\Sigma}_1/n_1),
\end{aligned}
\tag{4.14}
$$

where $n_0 = \sum_{i=1}^n \mathbb{1}\{L^{(i)} = 0\}$, $\bar{\boldsymbol{Y}}_{0\cdot} = \sum_{i=1}^n \mathbb{1}\{L^{(i)} = 0\}\boldsymbol{Y}^{(i)}/n_0$, $n_1 = \sum_{i=1}^n \mathbb{1}\{L^{(i)} = 1\}$, $\bar{\boldsymbol{Y}}_{1\cdot} = \sum_{i=1}^n \mathbb{1}\{L^{(i)} = 1\}\boldsymbol{Y}^{(i)}/n_1$.

3. Update the missing labels according to the current distributions; for $i = 1, \ldots, n$, if label is missing, update

$$
L^{(i)} \sim \text{Bernoulli}\left(\frac{\lambda_1 \phi(Y^{(i)}; \boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)}{\lambda_0 \phi(Y^{(i)}; \boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0) + \lambda_1 \phi(Y^{(i)}; \boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)}\right),
$$

since $\dfrac{P(L^{(i)} = 0 | Y^{(i)})}{P(L^{(i)} = 1 | Y^{(i)})} = \dfrac{\lambda_0 \phi(Y^{(i)}; \boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0)}{\lambda_1 \phi(Y^{(i)}; \boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)}$, where λ_0 and λ_1 stand for the proportion of class 0 and class 1. There are two ways to figure out λ_0 and λ_1, one is to specify them in advance if we know the proportion of each category in the population. The other way is to treat them as unknown parameters. We can specify the prior distribution $\lambda_0 \sim \text{Beta}(l_0, l_1)$ and update them with each MCMC iteration $\lambda_0 \sim \text{Beta}(l_0 + n_0, l_1 + n_1)$.

We can then obtain the posterior mean of $\bar{\boldsymbol{\theta}}_1, \ldots, \bar{\boldsymbol{\theta}}_p, \bar{\boldsymbol{\mu}}_0, \bar{\boldsymbol{\Sigma}}_0, \bar{\boldsymbol{\mu}}_1, \bar{\boldsymbol{\Sigma}}_1$ (and $\bar{\lambda}_0, \bar{\lambda}_1$ if treated as unknown parameters). For the new data $X^{(\text{new})}$ coming in, we shall first apply the transformation as the one applied to the original training data. Then calculate $Y^{(\text{new})}$ according to (4.8). If $\bar{\lambda}_0 \phi(Y^{(\text{new})}, \bar{\boldsymbol{\mu}}_0, \bar{\boldsymbol{\Sigma}}_0) > \bar{\lambda}_1 \phi(Y^{(\text{new})}, \bar{\boldsymbol{\mu}}_1, \bar{\boldsymbol{\Sigma}}_1)$, then assign it to Class 0; otherwise, assign it to Class 1. We will call our method Nonparanormal method in the simulations.

4.4 Model selection

Notice that we have to choose the number of basis functions J we want to use to estimate the transformation function. If the labeled data is sufficiently extensive, we can consider using cross validation to choose J. However, in semi-supervised learning settings, the labeled data are usually very limited. The method we propose is inspired by the low density assumption which is widely used in the semi-supervised learning literature. The idea is that we choose the classifier which best satisfies the low density assumption, i.e., the one with less points on the boundary. More specifically, we define the points that is close to the boundary by

$$\{X^{(i)} : m^{-1} < \frac{\lambda_0 \phi(Y^{(i)}; \bar{\boldsymbol{\mu}}_0, \bar{\boldsymbol{\Sigma}}_0)}{\lambda_1 \phi(Y^{(i)};, \bar{\boldsymbol{\mu}}_1, \bar{\boldsymbol{\Sigma}}_1)} < m, i = 1, \ldots, n\}. \tag{4.15}$$

where $\bar{\boldsymbol{\mu}}_0, \bar{\boldsymbol{\Sigma}}_0, \bar{\boldsymbol{\mu}}_1, \bar{\boldsymbol{\Sigma}}_1$ are the posterior means of $\boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0, \boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1$ respectively, m is an integer to be chosen. According to our simulation results, $m = 3$ is a good choice. Other values like 2 and 4 can also be taken into consideration. Notice that here $\boldsymbol{Y}$ is calculated based on the posterior mean $\bar{\boldsymbol{\theta}}_1, \ldots, \bar{\boldsymbol{\theta}}_p$. We choose the J which has the smallest number of points in the set defined by (4.15).

In the simulation studies, to save computation time, we run the procedure with $J = 8, \ldots, 15$, each 500 iterations to get a decision rule. Then we choose the best J according to the low density assumption introduced above and then run 10000 iterations to get the final classifier.

4.5 Simulations

Since our proposed method relies on the nonparanormality assumption, in the simulation studies, we consider two cases: the case when the assumption is satisfied and the case when the assumption is violated.

4.5.1 Nonparanormality assumption satisfied

We consider $p = 5, 10, 15$ in our simulations. For each dimension p, instead of specifying some values for means and covariances of the underlying Gaussian distributions $\mathrm{N}(\boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0)$ and $\mathrm{N}(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$, we decide to randomly generate some values as follows:

- Generate μ_{0d} and μ_{1d} independently and identically from a uniform distribution ranging in $[0, 4]$ for $d = 1, \ldots, p$.
- Generate $\boldsymbol{\Sigma}_0$ is by $\boldsymbol{A}^T\boldsymbol{A}$, where every element in $\boldsymbol{A}$ is generated from a uniform distribution ranging in $[-1, 1]$. $\boldsymbol{\Sigma}_1$ is generated the same way independently.

Here we considered two true transformations. The first one is the logistic transformation $\{1 + \exp(-\frac{X_d - (\mu_{0d} + \mu_{1d})/2}{(\sqrt{\Sigma_{0d,d}} + \sqrt{\Sigma_{1d,d}})/2})\}^{-1}$ for each dimension $d = 1, \ldots, p$. The second one is the probit transformation $\Phi(X_d, (\mu_{0d} + \mu_{1d})/2, (\sqrt{\Sigma_{0d,d}} + \sqrt{\Sigma_{1d,d}})/2)$ for each dimension $d = 1, \ldots, p$.

For each of the settings above, generate n^* number of observations for each class and randomly select n_l^* of them to be labeled. Here n^* takes values in $(50, 100)$ and n_l^* takes values in $(3, 5, 10)$. For testing set, we will generate 5000 data for each class. We compare our Nonparanormal (NN) method to other widely used semi-supervised learning methods in R package 'RSSL' (Krijthe 2016). We have tried all methods included in this package. However, EMLinearDiscriminantClassifier (Expectation Maximization applied to the linear discriminant classifier assuming Gaussian classes with a shared covariance matrix) (Dempster et al. 1977), IC Linear Discriminant Classifier (Krijthe and Loog 2014), Kernel Least Squares Classifier, LaplacianKernelLeastSquaresClassifier (Belkin et al. 2006), MC Linear Discriminant Classifier (Loog 2011), Quadratic Discriminant Classifier do not work for our generated datasets. Entropy Regularized LogisticRegression (Grandvalet and Bengio 2005), Linear SVM, Logistic Loss Classifier, Logistic Regression, MC Nearest Mean Classifier (Loog 2010) do not have good performances. Here we only include 5 methods which turned out

Table 4.1: Classification error rate ($\times 10^2$) for the test data when the data is generated with a logistic transformation. (Here * means there are 1–6 cases failed to output a result. The error is calculated based on the remianing outputs.)

p	(n^*, n_l^*)	NN	ICLS	LSVM	WSVM	SVML	EM	SVM	RF
5	(50,3)	3.39	15.82	*29.29	19.7	29.20	41.00	26.00	31.55
	(50,5)	5.35	12.48	29.00	17.65	26.06	37.24	21.11	26.43
	(50,10)	3.23	7.70	30.37	12.00	19.75	31.02	13.73	19.94
	(100,3)	3.90	18.16	*29.46	19.86	28.88	41.94	28.40	29.76
	(100,5)	2.81	13.61	*30.01	14.03	24.05	40.08	21.09	24.70
	(100,10)	2.80	8.24	31.94	10.7	19.44	34.09	14.72	19.84
10	(50,3)	0.20	18.13	*13.98	6.26	11.52	7.44	16.54	16.89
	(50,5)	0.20	11.43	10.13	5.62	9.17	7.53	10.57	14.44
	(50,10)	0.20	4.66	7.05	4.84	6.46	7.05	6.15	11.23
	(100,3)	0.12	18.38	*12.35	4.53	10.71	7.76	17.23	16.82
	(100,5)	0.12	12.28	10.38	4.42	8.83	7.72	11.26	15.39
	(100,10)	0.12	5.20	7.51	3.98	6.52	7.51	6.63	11.15
15	(50,3)	0.10	27.29	13.99	3.48	13.98	4.33	17.10	18.29
	(50,5)	0.08	16.36	8.09	3.17	8.97	4.12	7.64	13.93
	(50,10)	0.10	6.91	5.11	2.43	5.87	3.93	3.81	8.89
	(100,3)	0.04	26.38	14.67	2.08	14.05	3.97	17.94	18.41
	(100,5)	0.04	18.78	8.36	1.95	9.07	3.97	10.44	13.58
	(100,10)	0.04	7.55	6.17	1.90	6.55	3.88	4.49	10.09

to be the the best ones for our data. These methods are IC Least Squares Classifier(ICLS) (Krijthe and Loog 2015), Laplacian SVM(LSVM) (Belkin et al. 2006), Well SVM(WSVM) (Li et al. 2013), svmlin(SVML) (Sindhwani and Keerthi 2006), EM Nearest Mean Classifier(EM) (Dempster et al. 1977). To prove that using unlabeled data can actually improve the accuracy of the classifier, we also include Support Vector Machine (SVM) and Random Forest (RF), two supervised classification methods, trained only on labeled data for comparison. We simulate 30 datasets for each setting and the results shown in Table 4.1 and 4.2 are the average of classification error rates for 30 cases.

From the above results, we can see that clearly our Nonparanormal method outperforms other methods when our assumption is satisfied. It is worth pointing out that because the

Table 4.2: Classification error rate ($\times 10^2$) for the test data when the data is generated with a probit transformation. (Here * means there are 1–6 cases failed to output a result. The error is calculated based on the remianing outputs.)

p	(n^*, n_l^*)	NN	ICLS	LSVM	WSVM	SVML	EM	SVM	RF
5	(50,3)	3.88	18.47	*30.31	22.44	28.97	42.77	26.22	31.11
	(50,5)	3.75	14.96	29.18	18.49	25.11	39.14	22.12	26.16
	(50,10)	3.66	9.77	31.52	13.23	18.51	31.84	14.57	20.12
	(100,3)	6.21	19.37	*29.32	20.72	28.47	43.99	28.55	29.24
	(100,5)	3.10	14.92	30.18	15.29	23.26	42.53	21.29	24.49
	(100,10)	3.08	10.39	27.96	12.26	18.37	36.81	17.61	19.79
10	(50,3)	0.35	20.08	*13.66	6.89	12.11	7.95	16.57	16.72
	(50,5)	0.36	12.77	10.27	6.19	9.52	7.97	10.40	14.45
	(50,10)	0.33	6.74	7.60	5.39	7.00	7.54	6.54	11.34
	(100,3)	0.23	19.87	*12.86	5.10	11.42	8.10	17.44	16.80
	(100,5)	0.22	14.95	10.56	4.99	9.32	8.08	11.33	15.69
	(100,10)	0.21	7.16	7.93	4.47	6.91	7.83	7.07	11.17
15	(50,3)	0.10	27.21	*14.04	3.97	14.29	4.66	17.50	18.18
	(50,5)	0.11	16.91	8.44	3.55	9.44	4.44	8.04	13.89
	(50,10)	0.11	7.57	5.54	2.80	6.39	4.26	4.25	8.86
	(100,3)	0.05	26.50	14.79	2.61	14.37	4.33	18.09	18.62
	(100,5)	0.05	20.21	8.92	2.28	9.56	4.34	10.58	13.56
	(100,10)	0.06	8.95	6.57	2.26	7.15	4.26	4.99	10.11

parameters generated for different dimensions are different, the difficulty for classification is different as well. Obviously here the difficulty for classification decreases as the dimensions increases. The accuracy increases as the number of labels increases, which is expected. The accuracy also increases as the number of samples increases in general, (except when $p = 5$ and $(n^*, n_l^*) = (50, 3)$ comparing to $(n^*, n_l^*) = (100, 3)$ maybe because the problem is too difficult and the labeling proportion is playing a more important part). Comparing different methods, the EM algorithm is highly dependent on the assumption that the underlying distributions are Gaussian. This assumption is not applicable in these cases so it does not have a good performance as expected. Indeed it is the worst when $p = 5$. It is actually a bit surprising to find out that for the case when $p = 10$ and $p = 15$, EM is actually doing reasonably, maybe because the distributions can be approximate by Gaussian to some extent. Among those SVM methods, WSVM has the best performance. LSVM sometimes even failed to generate a result. The reason why SVM does not work well in our cases is maybe because the assumption of SVM (low density on the decision boundary) is too weak so it cannot work well for hard classification problem when the two groups are close to each other. Looking at supervised classifiers trained with only labeled data, for some cases they perform even better than some semi-supervised learning classifiers. For example, SVM is better than LSVM, SVML and EM when $p = 5$. Indeed, this shows unlabeled data can hurt accuracy when used inappropriately. Only when used in a good way, unlabeled data can greatly improve the accuracy, like the NN methods we proposed in these cases.

4.5.2 Nonparanormality assumption fails

To simulate the case when the nonparanormality assumption is violated, we still generate Gaussian distributions $\mathrm{N}(\boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0)$ and $\mathrm{N}(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$ as the underlying distributions, but we will apply different transformation on different classes. For simplicity, we will only con-

Table 4.3: Classification error rate ($\times 10^2$) for the test data when the data violate the Nonparanormal assumption. (Here * means there are 7–9 cases failed to output a result. The error is calculated based on the remianing outputs.)

p	(n^*, n_l^*)	NN	ICLS	LSVM	WSVM	SVML	EM	SVM	RF
5	(50,3)	5.32	16.38	31.58	17.41	28.45	42.95	22.54	31.30
	(50,5)	2.17	14.19	29.74	16.03	25.75	38.14	17.18	23.99
	(50,10)	2.00	8.89	31.73	10.40	18.65	31.26	10.41	16.31
	(100,3)	3.46	18.97	30.49	17.95	28.18	44.01	23.24	29.12
	(100,5)	1.66	13.82	33.13	11.96	23.39	40.46	17.42	23.11
	(100,10)	1.67	9.51	31.97	9.23	18.81	35.83	11.07	16.04

sider the case when $p = 5$ and we will use the same values generated above for means and covariances. We will apply the logisic transformation $\{1+\exp(-\frac{X_d-(\mu_{0d}+\mu_{1d})/2}{(\sqrt{\Sigma_{0d,d}}+\sqrt{\Sigma_{1d,d}})/2})\}^{-1}$ for each dimension $d = 1,\ldots,p$ on observations from Class 0, and the probit transformation $\Phi(X_d,(\mu_{0d}+\mu_{1d})/2,(\sqrt{\Sigma_{0d,d}}+\sqrt{\Sigma_{1d,d}})/2)$ for each dimension $d = 1,\ldots,p$, on observations from Class 1. We generate n^* number of observations for each class and randomly select n_l^* of them to be labeled. Once again, n^* takes values in $(50,100)$ and n_l^* takes values in $(3,5,10)$. For testing set, we will generate 5000 data for each class. 30 datasets are generated for each setting and we compare our proposed method with other methods mentioned above. The results are shown in Table 4.3.

From the result above, we can see that our proposed NN method has substantially lower error rates than others even when nonparanormality assumption fails. This shows our method are robust to this assumption.

4.6 Real data

Here we consider two datasets from UCI Machine Learning Repository[1]. For each dataset, we randomly select 70% of the data for training purpose and 30% of the data as testing

[1]https://archive.ics.uci.edu/ml/index.php

set. For the training set, we again randomly select 10% of the label and regard the rest as unlabeled data for our semi-supervised learning method. For each dataset, we repeat the process for 10 times and take the mean of the false positive rate, false negative rate, overall error rate and Matthews correlation coefficient (MCC) for all semi-supervised methods considered. We calculate MCC because it takes into account true and false positives and negatives and is generally regarded as a balanced measure which can be used even if the classes are of very different sizes. The two data sets we considered are:

- Breast Cancer Wisconsin (Diagnostic) Data Set [2]

  The purpose of this data is to use the features of the cell nucleus to predict whether the disease is malignant or benign. The features are computed from a digitized image of a fine needle aspirate (FNA) of a breast mass. Those features include radius, texture, perimeter, area, smoothness, compactness, concavity, concave points, symmetry and fractal dimension. For each feature, the mean, standard error, and "worst" or largest (mean of the three largest values) are calculated for each image. All features are continuous variables. The data have 357 benign cases and 212 malignant cases.

- Ionosphere Data Set [3]

  The purpose of this data is to classify the radar returns of free electrons in the ionosphere as "good" or "bad". "Good" radar returns are the ones showing evidence of some type of structure in the ionosphere while "bad" returns are those whose signals pass through the ionosphere. There are 17 pulse numbers for the system and instances are described by 2 attributes per pulse number. There are 34 features in this data set and all of them are continuous. The data contains 126 bad cases and 225 good cases.

[2] https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+(Diagnostic)
[3] https://archive.ics.uci.edu/ml/datasets/ionosphere

Table 4.4: Classification results on Breast Cancer Wisconsin (Diagnostic) Data Set

	NN	ICLS	LSVM	WSVM	SVML	EM
False Positive Rate	0.083	0.026	0.009	0.052	0.057	0.003
False Negative Rate	0.077	0.208	0.164	0.168	0.130	0.377
Overall Error Rate	0.080	0.096	0.068	0.095	0.085	0.146
MCC	0.835	0.798	0.858	0.799	0.822	0.705

Table 4.5: Classification results on Ionosphere Data Set

	NN	ICLS	LSVM	WSVM	SVML	EM
False Positive Rate	0.083	0.460	0.601	0.329	0.468	0.263
False Negative Rate	0.182	0.060	0.044	0.134	0.048	0.339
Overall Error Rate	0.147	0.182	0.215	0.198	0.184	0.281
MCC	0.713	0.505	0.445	0.489	0.513	0.350

The results are given in Table 4.4 and Table 4.5:

For Breast Cancer Wisconsin (Diagnostic) Data Set, NN and LSVM have comparable results. LSVM has the lowest overall error rate and highest MCC. Our proposed method NN has slightly higher error rate and lower MCC. But Unlike LSVM, whose false negative rate is much higher than false positive rate, the false positive rate and false negative rate for NN are almost the same. For Ionosphere Data Set, NN has much better result in terms of both overall error rate and MCC. In conclusion, NN is a safe choice for semi-supervised learning in real world.

REFERENCES

Alonzo, T. A. and Pepe, M. S. (2005). Assessing accuracy of a continuous screening test in the presence of verification bias. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 54(1):173–190.

Bağırsakçı, E., Şahin, E., Atabey, N., Erdal, E., Guerra, V., and Carr, B. I. (2017). Role of albumin in growth inhibition in hepatocellular carcinoma. *Oncology*, 93(2):136–142.

Balcan, M.-F., Blum, A., Choi, P. P., Lafferty, J. D., Pantano, B., Rwebangira, M. R., and Zhu, X. (2009). Person identification in webcam images: An application of semi-supervised learning.

Belkin, M., Niyogi, P., and Sindhwani, V. (2006). Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. *Journal of Machine Learning Research*, 7(Nov):2399–2434.

Blum, A. and Mitchell, T. (1998). Combining labeled and unlabeled data with co-training. In *Proceedings of the Eleventh Annual Conference on Computational Learning Theory*, pages 92–100. Citeseer.

Carr, B. I. and Guerra, V. (2017). Serum albumin levels in relation to tumor parameters in hepatocellular carcinoma patients. *The International Journal of Biological Markers*, 32(4):391–396.

Chi, Y.-Y. and Zhou, X.-H. (2008). Receiver operating characteristic surfaces in the presence of verification bias. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 57(1):1–23.

Cozman, F. G., Cohen, I., and Cirelo, M. C. (2003). Semi-supervised learning of mixture models. In *Proceedings of the 20th International Conference on Machine Learning (ICML-03)*, pages 99–106.

Cramer, D. W., Bast, R. C., Berg, C. D., Diamandis, E. P., Godwin, A. K., Hartge, P., Lokshin, A. E., Lu, K. H., McIntosh, M. W., Mor, G., et al. (2011). Ovarian cancer biomarker performance in prostate, lung, colorectal, and ovarian cancer screening trial specimens. *Cancer Prevention Research*, 4(3):365–374.

de Carvalho, V. I., de Carvalho, M., and Branscum, A. (2018). Bayesian bootstrap inference for the roc surface. *Stat*, 7.

de Carvalho, V. I., Jara, A., Hanson, T. E., and de Carvalho, M. (2013). Bayesian nonparametric ROC regression modeling. *Bayesian Analysis*, 8(3):623–646.

Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)*, 39(1):1–22.

Dreiseitl, S., Ohno-Machado, L., and Binder, M. (2000). Comparing three-class diagnostic tests by three-way ROC analysis. *Medical Decision Making*, 20(3):323–331.

Ghosal, S. and van der Vaart, A. (2017). *Fundamentals of Nonparametric Bayesian Inference*, volume 44. Cambridge University Press.

Grandvalet, Y. and Bengio, Y. (2005). Semi-supervised learning by entropy minimization. In *Advances in neural information processing systems*, pages 529–536.

Gu, J. and Ghosal, S. (2009). Bayesian ROC curve estimation under binormality using a rank likelihood. *Journal of Statistical Planning and Inference*, 139(6):2076–2083.

Gu, J., Ghosal, S., and Kleiner, D. E. (2014). Bayesian ROC curve estimation under verification bias. *Statistics in {M*edicine, 33(29):5081–5096.

Hamed, E. O., Ahmed, H., Sedeek, O. B., Mohammed, A. M., Abd-Alla, A. A., and Ghaffar, H. M. A. (2013). Significance of HE4 estimation in comparison with CA125 in diagnosis of ovarian cancer and assessment of treatment response. *Diagnostic Pathology*, 8(1):11.

Heimbach, J. K., Kulik, L. M., Finn, R. S., Sirlin, C. B., Abecassis, M. M., Roberts, L. R., Zhu, A. X., Murad, M. H., and Marrero, J. A. (2018). Aasld guidelines for the treatment of hepatocellular carcinoma. *Hepatology*, 67(1):358–380.

Hein, M. and Maier, M. (2007). Manifold denoising. In *Advances in Neural Information Processing Systems*, pages 561–568.

Hsieh, F. and Turnbull, B. W. (1996). Nonparametric and semiparametric estimation of the receiver operating characteristic curve. *The Annals of Statistics*, pages 25–40.

Inácio, V., Turkman, A. A., Nakas, C. T., and Alonzo, T. A. (2011). Nonparametric bayesian estimation of the three-way receiver operating characteristic surface. *Biometrical Journal*, 53(6):1011–1024.

Kang, L. and Tian, L. (2013). Estimation of the volume under the ROC surface with three ordinal diagnostic categories. *Computational Statistics & Data Analysis*, 62:39–51.

Krijthe, J. H. (2016). Rssl: Semi-supervised learning in R. In *International Workshop on Reproducible Research in Pattern Recognition*, pages 104–115. Springer.

Krijthe, J. H. and Loog, M. (2014). Implicitly constrained semi-supervised linear discriminant analysis. In *2014 22nd International Conference on Pattern Recognition*, pages 3762–3767. IEEE.

Krijthe, J. H. and Loog, M. (2015). Implicitly constrained semi-supervised least squares classification. In *International Symposium on Sntelligent Data Analysis*, pages 158–169. Springer.

Lawrence, N. D. and Jordan, M. I. (2005). Semi-supervised learning via gaussian processes. In *Advances in Neural Information Processing Systems*, pages 753–760.

Li, J., Zhou, X., and Fine, J. P. (2012). A regression approach to roc surface, with applications to alzheimer's disease. *Science China Mathematics*, 55(8):1583–1595.

Li, J. and Zhou, X.-H. (2009). Nonparametric and semiparametric estimation of the three way receiver operating characteristic surface. *Journal of Statistical Planning and Inference*, 139(12):4133–4142.

Li, Y. and Ghosh, S. K. (2015). Efficient sampling methods for truncated multivariate normal and student-t distributions subject to linear inequality constraints. *Journal of Statistical Theory and Practice*, 9(4):712–732.

Li, Y.-F., Tsang, I. W., Kwok, J. T., and Zhou, Z.-H. (2013). Convex and scalable weakly labeled SVMs. *The Journal of Machine Learning Research*, 14(1):2151–2188.

Little, R. J. and Rubin, D. B. (2014). *Statistical Analysis with Missing Data*. John Wiley & Sons.

Liu, H., Han, F., Yuan, M., Lafferty, J., Wasserman, L., et al. (2012). High-dimensional semiparametric gaussian copula graphical models. *The Annals of Statistics*, 40(4):2293–2326.

Liu, P., Xie, S.-H., Hu, S., Cheng, X., Gao, T., Zhang, C., and Song, Z. (2017). Age-specific sex difference in the incidence of hepatocellular carcinoma in the united states. *Oncotarget*, 8(40):68131.

Loog, M. (2010). Constrained parameter estimation for semi-supervised learning: the case of the nearest mean classifier. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 291–304. Springer.

Loog, M. (2011). Semi-supervised linear discriminant analysis using moment constraints. In *IAPR International Workshop on Partially Supervised Learning*, pages 32–41. Springer.

Martos, G. and de Carvalho, M. (2018). Discrimination surfaces with application to region-specific brain asymmetry analysis. *Statistics in Medicine*, 37(11):1859–1873.

Montagnana, M., Danese, E., Giudici, S., Franchi, M., Guidi, G., Plebani, M., and Lippi, G. (2011). HE4 in ovarian cancer: from discovery to clinical application. *Advances in Clinical Chemistry*, 55:1–20.

Mossman, D. (1999). Three-way ROCs. *Medical Decision Making*, 19(1):78–89.

Mulgrave, J. J. and Ghosal, S. (2018). Bayesian inference in nonparanormal graphical models. *Bayesian Analysis (to appear)*.

Nakas, C. T. and Yiannoutsos, C. T. (2004). Ordered multiple-class ROC analysis with continuous measurements. *Statistics in Medicine*, 23(22):3437–3449.

Nze Ossima, A. D., Daurès, J.-P., Bessaoud, F., and Trétarre, B. (2015). The generalized lehmann ROC curves: Lehmann family of ROC surfaces. *Journal of Statistical Computation and Simulation*, 85(3):596–607.

Scurfield, B. K. (1996). Multiple-event forced-choice tasks in the theory of signal detectability. *Journal of Mathematical Psychology*, 40(3):253–269.

Sindhwani, V. and Keerthi, S. S. (2006). Large scale semi-supervised linear SVMs. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 477–484. ACM.

Tanriverdi, O. (2014). A discussion of serum albumin level in advanced-stage hepatocellular carcinoma: a medical oncologist's perspective. *Medical Oncology*, 31(11):282.

To Duc, K. (2017). bcrocsurface: An R package for correcting verification bias in estimation of the ROC surface and its volume for continuous diagnostic tests. *BMC Bioinformatics*, 18:503.

To Duc, K., Chiogna, M., and Adimari, G. (2016). Bias–corrected methods for estimating the receiver operating characteristic surface of continuous diagnostic tests. *Electronic Journal of Statistics*, 10(2):3063–3113.

Xiong, C., van Belle, G., Miller, J. P., and Morris, J. C. (2006). Measuring and estimating diagnostic accuracy when there are three ordinal diagnostic groups. *Statistics in Medicine*, 25(7):1251–1273.

Yarowsky, D. (1995). Unsupervised word sense disambiguation rivaling supervised methods. In *33rd Annual Meeting of the Association for Computational Linguistics*, pages 189–196.

Zemel, R. S. and Carreira-Perpiñán, M. Á. (2005). Proximity graphs for clustering and manifold learning. In *Advances in Neural Information Processing Systems*, pages 225–232.

Zhang, X. and Lee, W. S. (2007). Hyperparameter learning for graph based semi-supervised learning algorithms. In *Advances in Neural Information Processing Systems*, pages 1585–1592.

Zheng, H. and Gao, Y. (2012). Serum HE4 as a useful biomarker in discriminating ovarian cancer from benign pelvic disease. *International Journal of Gynecological Cancer*, 22(6):1000–1005.

Zhu, X. (2005). Semi-supervised learning literature survey. Technical Report 1530, Computer Sciences, University of Wisconsin-Madison.