Estimating the Effect of Classification Errors on Domain Statistics

Yanzhe Li

Thesis advisor: Prof. Dr. Peter Grünwald

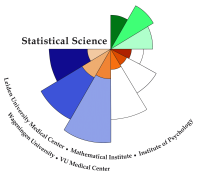
External advisors: Dr. Sander Scholtus (CBS)

Dr. Arnout Van Delden (CBS)

MASTER THESIS

Defended on December 10th, 2020

Specialization: Statistical Science



STATISTICAL SCIENCE

FOR THE LIFE AND BEHAVIOURAL SCIENCES

Acknowledgement

How time flies. I have finally come to this stage, about to paint a full end to my master program. It has been a difficult year for me and also for the world with COVID-19. I really want to express my sincere gratitude to the following people. It is their support and encouragement that helped me finish the thesis during such a hard time.

First of all, I would like to thank my supervisors from CBS, Dr. Sander Scholtus and Dr. Arnout Van Delden for sharing their expertise, experiences, ideas and their patience.

I also want to thank Prof. Dr. Peter Grünwald, for providing ideas and advice on my thesis, and keeping following my progress.

Last but most important, many thanks to my family, for all the unconditional support in this year.

Abstract

The reliability of statistics is essential for official statistics. With administrative data more often used instead of survey data, non-sampling errors become important factors in the accuracy of statistics. For domain statistics, such as yearly turnover of enterprises, classification errors occur. This study aims to measure the effect of classification errors on domain statistics, more specifically, bias and variance due to classification errors.

In this study, a new method was developed that applies a a Gaussian mixture model, estimated by the EM algorithm, in short referred to as *the EM method*. Further another method was introduced that combined the EM method with bootstrapping, referred to as the *combined method*. Among them, the EM method only estimates bias, and the combined method is able to estimate both bias and variance. Together with a previously used bootstrap method, the three methods were tested in a simulation study and in a case study. The bias and variance estimates from the three methods were compared with their corresponding true values in different settings. The results showed that the bias estimates from the EM and the combined method were closer to the true values compared to the bootstrap method; The combined method had closer outputs on variance estimation than the bootstrap method. The EM and the combined method were equally accurate in estimating the true bias.

These results suggest that the EM and the combined method estimated the bias and variance more accurately than the bootstrap method. In practice, the combined method is recommended since both the bias and the variance can be estimated. In a situation with a very large data set, where the variance is usually small and the bias is of most concern, the EM method may be preferred.

Contents

List of Figures

List of Tables

1 Introduction

It is widely recognized that reliability of statistics is of top importance in any field of research, especially in official statistics, where most published statistics are used for policy making. It is a trend in official statistics that administrative data is now replacing survey data in the production of statistics. As a result, the accuracy of statistics is more determined by non-sampling errors, such as measurement errors in comparison to sampling errors (Zhang, 2011). Among them, classification errors play a big role.

Classification errors often appear wherever there is a categorical variable in the administrative data. Although administrative data has the advantage of efficiency, where data already exists in administrative registers and covers a large part of the target population, statistical agencies have little control over the collection of these data. The concepts and definitions used by the register owners sometimes differ from those used by official statistics which can easily cause misclassification (Magnusson, Palm, Branden, & Mörner, 2017).

It is common in official statistics that classification codes are used for identifying units into domains of interests. A wide range of standard statistical classifications are used. Eurostat and the Member States, for example, use coding system CPA as the classification of products by activity, and NACE for classifying economic activities. Classifications such as educational levels and types of occupation are also used quite often. A standard set of classification codes is used to make sure statistics are comparable and reliable. Nowadays, statistical agencies have also started to apply machine learning algorithms to classify units into subgroups (Meertens, Diks, Van den Herik, & Takes, 2020).

Because the correct use of these codes is not guaranteed in administrative data, the effect of classification errors cannot be neglected when assessing the quality of statistics. Even when the accuracy of the classifier increases, the classification errors will still cause bias to subgroup proportion if its misclassification probabilities do not conform to certain conditions (Scholtus & Van Delden, 2020). Measures such as increasing the accuracy of classification or strict controls in registers, can be taken to reduce the effect of classification errors.

The effect of misclassification has been studied in different research fields, such as machine learning and epidemiology. Furthermore, the effect of misclassification has been studied for, and applied to different types of problems. A study in the field of software engineering reported that the misclassified bug reports have introduced bias to their bug prediction model (Herzig, Just, & Zeller, 2013). Another study assessed the uncertainty of predictive performance of a classification model and showed that the uncertainty of the model increased when the probability of making classification errors became larger (Morais, Lima, & Martin, 2019). A clinical research estimated the treatment effect in observational studies and found the misclassification of a confounder would cause bias on the estimation of the average treatment effect (Nab, Groenwold, Van Smeden, & Keogh, 2020).

For National Statistical Institutes (NSIs) and other statistical agencies, to ensure the quality of official statistics is of central importance (Nordbotten, 2010). The main purpose of our study is to investigate the effect of

classification errors on the accuracy of statistics, such as subgroup means (Selén, 1986), counts (Scholtus & Van Delden, 2020) and growth rates (Scholtus & Van Delden, 2020; Scholtus, Van Delden, & Burger, 2019).

In the studies considering the impact of classification errors on estimators, analytical expressions and a bootstrap method have been proposed. Greenland (1988) formulated variance estimates for estimated log odds ratios that are often used to measure effect size in epidemiology. Formulas for bias and variance brought by misclassification for other domain estimators have also been derived, such as counts and growth rates (Scholtus & Van Delden, 2020; Scholtus et al., 2019).

Zhang (2011) used the bootstrap method to measure the variance of total sum caused by misclassification of households. The bootstrap method has been applied by Van Delden, Scholtus, and Burger (2016) to quantify the bias and variance of level estimates brought by classification errors. Scholtus et al. (2019) also applied the bootstrap method to quantify the bias and variance of growth rates.

The advantages of analytical expressions are quite obvious. Once the analytical solutions for corresponding estimators are obtained, the calculation for bias and variance is straightforward. With the expression, the bias and variance estimation is relatively more transparent where people can easily understand why misclassification brings bias and uncertainty and how classification errors are constructed.

However, the estimations from the analytical expressions are inaccurate. In practice, the estimators in the bias and variance expressions are replaced by their estimated values. When classification errors bring bias and variance to the estimated values, the bias and variance estimated from the analytical expressions will also contain bias and variance. Besides, the analytical expressions are different for every estimator. When the assumptions of the classification model or other conditions are changed, expressions will no longer be applicable.

Compared to the analytical method, the bootstrap method has more flexibility since it can be adopted to almost any estimator. But like the use of analytical expressions, the bootstrap estimates may be biased in practice (Burger, Van Delden, & Scholtus, 2015).

In order to evaluate the effect of classification errors, the probabilities of making classification errors need to be obtained. Among these studies related to misclassification, there are situations where classification error probabilities are known beforehand, and also the situation in which those errors need to be estimated. The most common assumption is that the misclassification probabilities estimated from a suitable set of data, which is called "validation data" or "audit sample", are the same as in the target data set. An audit sample can either be sampled from the target data set (Gravel & Platt, 2018), or from a different set of data where the classification error probabilities are assumed to be representative (Edwards, Bakoyannis, Yiannoutsos, Mburu, & Cole, 2019; Edwards, Cole, & Fox, 2020).

Our study focuses on estimating the accuracy of statistics, more specifically on measuring the bias and variance that misclassification has brought to the statistics. We aim to develop methods that can estimate the bias and

variance more accurately and overcome the weakness of the analytical method and the bootstrap method. By doing so, a new method is developed by applying a Gaussian mixture model, estimated by EM algorithm, in short referred to as EM method. Further a method combines EM method with bootstrapping, referred to as the combined method. Together with the bootstrap method, the three methods will been tested in a simulation study and in a case study. In the simulation study, Gaussian mixture distributions with various number of components are simulated. In the case study, the criterion of BIC will be used to choose the optimal number of components in a real data set. The bias and variance estimates from the three methods will be compared with their corresponding true values.

Apart from more accurate estimation, the new methods estimate the classification error probabilities as a part of their procedures. An audit sample is no longer needed in the new methods for the misclassification probabilities estimation, which is necessary in the previous methods. Still, we will make use of an audit sample when applying the new methods, but it plays a different and unnecessary role, to accelerate and stabilize processes of model parameter estimations.

The remainder of this thesis is organised as follows. Section 2 describes the three methods that we compare, including the details of the EM algorithm in the methods. The three methods are tested with a simulation study, given in section 3 and with a case study in section 4. Section 5 provides practical guidelines of applying the new method in real applications. Section 6 concludes results of our study and proposes future directions. Appendices give more details of our study: Appendix A provides technical details of using the EM algorithm with an audit sample; Appendix B includes more experiments in order to figure out what factors influence the performance of our methods in the case study; Appendix C discusses more about the use of BIC in our study.

2 Methodology

In this section, the methodology of the study is provided. To simplify our setting, we consider the condition with only two classes. It includes the general setting of the study, how the data is modeled, how bias and variance are defined, what statistics of interest include, and an audit sample used in our study. The three methods used in the study are also described. One of them is the bootstrap method used in the previous studies. Another two, the EM method and the combined method, are the new methods we develop in this study. The EM algorithm that is applied in the new methods is introduced in detail.

2.1 General Settings

2.1.1 Notations

Consider a data set with N rows and two columns, where each column represents a particular variable and each row corresponds to a unit $i = 1, ..., N$ in a population. There are in total two classes in the population, which are called class 1 and class 0. The proportion of class 1 in the population is denoted as α_1 and that of class 0 is α_0, with $\alpha_0 = 1 - \alpha_1$. We assume there is no missing data in the data set, so that all values of all variables are observed for all units in the population.

One of the variables is a continuous variable y. The continuous variable has some statistics we are interested to estimate. For example, it could be yearly turnover of entrepreneurs, emission of nitrogen oxides of industries, or harvest of vegetables of farms. We assume the continuous variable y is error-free.

Another variable is an observed classification variable $\hat{z}$ identifying to which group these units belong in practice. In official statistics, the observed classification variable $\hat{z}$ usually comes from a register or a machine-learning algorithm and contains classification errors. We assume the probabilities of making classification errors are random in the same way for all units, which also means that they are independent of the continuous variable y.

In order to better explain the misclassification probabilities, we introduce a true classification variable z which does not have any classification errors. In practice, the true classification variable z is often unknown, and only the observed classification variable $\hat{z}$ really exists.

In our study, the distribution of the continuous variable y is assumed to be a mixture of normal distributions for each class (see more details in Section 2.1.2). We also introduce a component variable m to identify which component these units belong to. The variable m is unknown beforehand. The true classification variable z and the component variable m together decide the distribution of y. The next section will explain it in detail.

2.1.2 General Model

To structure the probabilities of making classification errors, we use a 2×2 transition matrix $\mathbf{P}$ (Formula 1),

$$\mathbf{P} = \begin{pmatrix} p_{11} & 1 - p_{11} \\ 1 - p_{00} & p_{00} \end{pmatrix}. \tag{1}$$

The value p_{11} indicates the probability of identifying units as class 1 when their true class is 1, which can be expressed as $p(\hat{z} = 1|z = 1) = p_{11}$; p_{00} indicates the probability of identifying units as class 0 when their true class is 0, which can be expressed as $p(\hat{z} = 0|z = 0) = p_{00}$. Other probabilities are expressed as $p(\hat{z} = 0|z = 1) = 1 - p_{11}$ and $p(\hat{z} = 1|z = 0) = 1 - p_{00}$.

Besides, we assume that the distribution of the continuous variable y for each class conforms to a Gaussian mixture model with a certain number of components n_i ($i \in \{0, 1\}$), which means that all the components for each class conform to the normal distributions.

Here we introduce a component indicator j to identify components in class 1 with $j \in \{1, ...n_1\}$ and a component indicator k to identify components in class 0 with $k \in \{1, ...n_0\}$.

For component j of class 1 ($j \in \{1, ...n_1\}$), π_{1j} is its prior probability in class 1 (also called *mixture weight*) and $\pi_{11} + ... + \pi_{1j} + ... + \pi_{1n_1} = 1$; μ_{1j} is its mean and σ_{1j} is its standard deviation.

For component k of class 0 ($k \in \{1, ...n_0\}$), π_{0k} is its prior probability in class 0 with $\pi_{01}+...+\pi_{0k}+...+\pi_{0n_0} = 1$; μ_{0k} is its mean and σ_{0k} is its standard deviation.

The probability of y_i depends on the class it belongs to (the values of z_i) and also by which component in this class it belongs to (the value of m_i). From Bayes's theorem, the probability of y_i conditional on its true class z_i is:

$$P\left(y_i|z_i\right) = \begin{cases} \sum_{j=1}^{n_1} P\left(m_i = j \,|\, z_i = 1\right) P\left(y_i \,|\, z_i = 1, m_i = j\right) , \\ \sum_{k=1}^{n_0} P\left(m_i = k \,|\, z_i = 0\right) P\left(y_i \,|\, z_i = 0, m_i = k\right) , \end{cases}$$

where $P\left(m_i = j \,|\, z_i = 1\right)$ is the mixture weight for component j in class 1 and equals to π_{1j} ; $P\left(m_i = k \,|\, z_i = 0\right)$ is the mixture weight for component k in class 0 and equals to π_{0k} .

Therefore, the density function for y_i can be expressed as,

$$f\left(y_i|z_i\right) = \begin{cases} \sum_{j=1}^{n_1} \pi_{1j} \cdot \mathcal{N}\left(y_i \,|\, \mu_{1j}, \sigma_{1j}^2\right) & \text{when } z_i = 1 , \\ \sum_{k=1}^{n_0} \pi_{0k} \cdot \mathcal{N}\left(y_i \,|\, \mu_{0k}, \sigma_{0k}^2\right) & \text{when } z_i = 0 , \end{cases} \tag{2}$$

where $\mathcal{N}(y \mid \mu, \sigma)$ is the probability density of y under a normal distribution with mean μ and variance σ^2. The identification of parameters in the density function can be guaranteed by restricting the order of means in the Gaussian mixture model. Here we simply assume $\mu_{11} < ... < \mu_{1j} < ... < \mu_{1n_1}$ and $\mu_{01} < ... < \mu_{0k} < ... < \mu_{0n_0}$.

We introduce $\boldsymbol{\theta}$ to represent the full set of parameters in our model, which include the class proportion α_1, p_{11} and p_{00} in the transition matrix $\mathbf{P}$ (Formula 1) and the parameters of components for each class. Therefore, θ is denoted as $\boldsymbol{\theta} = (\alpha_1, p_{11}, p_{00}, \pi_{1j}, \pi_{0k}, \mu_{1j}, \mu_{0k}, \sigma_{1j}, \sigma_{0k})$ with $j = 1, ..., n_1$ and $k = 1, ..., n_0$. In practice, Gaussian mixture models are often estimated by an EM algorithm. Section 2.3 describes how $\boldsymbol{\theta}$ are estimated from the EM algorithm.

2.1.3 Bias and Variance

With only the error-prone observed classification variable $\hat{z}$ observable in practice, the accuracy of estimated statistics for each class decreases, compared to their true values which can only be obtained from the true classification variable z. Two aspects of the accuracy are discussed: bias and variance.

In this study, we only consider the bias and variance due to classification errors. The bias is defined as the difference between the expected values of the estimated output and the correct value of domain statistics. The variance is a measure of the expected amount that the estimated domain statistics will change if different classification variables with same error distributions are used. The followings are the mathematical definitions of bias and variance in our study:

$$\begin{aligned} \textbf{Bias} &= \mathrm{E}(\hat{\zeta}) - \zeta, \\ \textbf{Variance} &= \mathrm{Var}(\hat{\zeta}) = \mathrm{E}\left(\left(\hat{\zeta} - \mathrm{E}(\hat{\zeta}) \right)^2 \right), \end{aligned} \tag{3}$$

where $\hat{\zeta}$ are the estimated domain statistics from $\hat{z}$, and ζ is the true value from z (details below). We assume that no other errors occur in the study.

2.1.4 Statistics of Interest

We are interested to estimate the accuracy of certain output for official statistics, namely the total sum for class 1 (T_1) and the proportion for class 1 (α_1). Since our study is under the setting of a binary classifier, the total sum and the proportion for class 0 can be calculated once these statistics for class 1 have been achieved. Additionally, we also verify how well we are able to estimate descriptive statistics for each class: the mean μ_1 for class 1 and μ_0 for class 0; and the standard deviation σ_1 for class 1 and σ_0 for class 0.

All the domain statistics we included in our study are denoted as $\zeta = (T_1, \alpha_1, \mu_1, \mu_0, \sigma_1, \sigma_0)$. Table 1 has listed the formulas for the maximum likelihood statistics ζ given variable y and z. By replacing z_i with $\hat{z}_i$ in Table

1, the observed domain statistics $\hat{\zeta} = (\hat{T}_1, \hat{\alpha}_1, \hat{\mu}_1, \hat{\mu}_0, \hat{\sigma}_1, \hat{\sigma}_0)$ can be calculated. Other forms of ζ (such as ζ^* in Section 2.2) can also be calculated in the same way.

The theoretical ζ (Table 1) requires the true y and z values and so cannot be computed in practice. In practice, we replace the unknown true z by the observed $\hat{z}$ in these expressions, and this yields the estimated $\hat{\zeta}$. Once z is replaced by $\hat{z}$, bias and variance occur (see details in Section 2.1.3).

Table 1: List of Formulas for the Maximum Likelihood Statistics ζ

Statistics of Interest	Notation	Formula given y and z
Total sum for class 1	T_1	$\sum z_i y_i$
Proportion of class 1	α_1	$\frac{\sum z_i}{N}$
Mean for class 1	μ_1	$\frac{\sum z_i y_i}{\sum z_i}$
Mean for class 0	μ_0	$\frac{\sum (1-z_i) y_i}{N - \sum z_i}$
Standard deviation for class 1	σ_1	$\sqrt{\frac{1}{\sum z_i} \sum z_i (y_i - \mu_1)^2}$
Standard deviation for class 0	σ_0	$\sqrt{\frac{1}{N - \sum z_i} \sum (1 - z_i)(y_i - \mu_0)^2}$

Note: 1. i stands for a unit in the population.
2. By replacing z_i with $\hat{z}_i$ in the formulas, $\hat{\zeta} = (\hat{T}_1, \hat{\alpha}_1, \hat{\mu}_1, \hat{\mu}_0, \hat{\sigma}_1, \hat{\sigma}_0)$ can be calculated. Other forms of ζ (such as ζ^* in Section 2.2) can also be calculated in the same way.

Maximum likelihood estimators are quite often applied in the context of mixture models. For the estimated standard deviation, it should be noticed that the maximum likelihood statistics are not unbiased in finite data sets since we divide by N rather than $N - 1$, but the bias disappears when N goes to infinity. Given that N in our setting is quite large, maximum likelihood estimation is used.

2.1.5 Audit sample

We select a small group of units as an audit sample in our study, where the true classes z of these units are known through manual checking. The audit sample is randomly sampled from the population to make sure it is representative.

In our study, the audit sample is used in the EM algorithm in two ways: one is to set the starting values; another is to add more information in the E step and the M step. Appendix A provides detailed information about applying the audit sample to the EM algorithm.

2.2 Methods

In our study, three methods were compared: the bootstrap method, the EM method and the combined method. The first method originates from Van Delden et al. (2016) and the other two methods are new methods, where the EM algorithm is applied to estimate the bias and variance (details in Section 2.3).

The three methods all start with the error-free continuous variable y and the observed classification variable $\hat{z}$ with classification errors.

The first method (see Algorithm 1) is the bootstrap method (Van Delden et al., 2016). It generates multiple sets of z^* by introducing classification errors again to $\hat{z}$ with the same transition matrix $\mathbf{P}$ (Formula 1 in Section 2.1.1), where $p(z^* = 1|\hat{z} = 1) = p_{11}$ and $p(z^* = 1|\hat{z} = 0) = p_{00}$. For each set of z^*, corresponding statistics ζ^* are calculated by formulas in Table 1. The bias is then estimated by $\mathbf{Bias_{boot}}$ and variance is estimated by $\mathbf{Var_{boot}}$, which refer to the empirical expectation (i.e., the mean) and the empirical variance of the S simulated values. In this way, the algorithm is used to find a numerical approximation to the true expectation and variance.

The classification error probabilities p_{11} and p_{00} in matrix $\mathbf{P}$ are assumed known in the bootstrap method. In our study, we used the estimated p_{11} and p_{00} from the EM method or the combined method and applied the estimated values in the bootstrap method.

Algorithm 1 The Bootstrap Method

Input: Variable y, $\hat{z}$, matrix $\mathbf{P}$ and S.

1: **for** $s = 1 \dots S$ **do**
2: Generate z^* by $\mathbf{P}$ (Formula 1), conditional on $\hat{z}$ for every unit i in the data set
3: Calculate the corresponding ζ^* for each z^*
4: **end for**
5: Calculate $\mathbf{Bias_{boot}} = \mathrm{E}(\zeta^*|\hat{z}) - \hat{\zeta}$, $\mathbf{Var_{boot}} = \mathrm{Var}(\zeta^*|\hat{z})$ based on S bootstraps

Output: $\mathbf{Bias_{boot}}$ and $\mathbf{Var_{boot}}$

The second method (see Algorithm 2) makes use of the EM algorithm Therefore we call it the EM method. The EM algorithm is used to estimate the parameters of the mixture Gaussian model (in Section 2.1.2) through achieving its maximum likelihood. The parameters of the model are returned from the EM algorithm, including p_{11} and p_{00} in the matrix $\mathbf{P}$. With parameters of the model obtained, maximum likelihood statistics $\tilde{\zeta}$ can be achieved (formulas in Section 2.3.5). The maximum-likelihood estimates $\tilde{\zeta} = (\tilde{T}_1, \tilde{\alpha}_1, \tilde{\mu}_1, \tilde{\mu}_0, \tilde{\sigma}_1, \tilde{\sigma}_0)$ can be seen close to the true value of our target statistics ζ (Table 1). Therefore, bias can be directly estimated from $\mathbf{Bias_{em}} = \hat{\zeta} - \tilde{\zeta}$, where $\tilde{\zeta}$ is the statistics estimated from EM algorithm and $\hat{\zeta}$ is the statistics from the observed classes. However, the variance that classification errors have brought to the statistics can not be estimated in this method. A further step should be taken which becomes our third method.

Algorithm 2 The EM Method

Input: Variable y, $\hat{z}$

1: Estimate parameters of the model $\boldsymbol{\theta}$, including p_{11} and p_{00}, through iterating over an E step and an M step until convergence in the EM algorithm, conditional on $\hat{z}$ and y
2: Calculate the $\tilde{\zeta}$ from the estimated parameters
3: Calculate $\mathbf{Bias_{em}} = \hat{\zeta} - \tilde{\zeta}$

Output: $\mathbf{Bias_{em}}$ and matrix $\mathbf{P}$

The third method (see Algorithm 3) combines EM method and bootstrapping in order to compute the variance that classification errors have brought to estimators. By applying the EM algorithm, the parameters of the model are obtained, including p_{11} and p_{00} in the matrix $\mathbf{P}$. With the estimated parameters of the model, statistics $\tilde{\zeta}$ is achieved (see details in the second method) and $p(z|y, \hat{z})$ can also be calculated (Formulas 8 in Section 2.3.2). For unit i, given y_i and $\hat{z}_i$, multiple $\tilde{z}_i$ are generated with success probability of $p(z_i = 1|y = y_i, \hat{z} = \hat{z}_i)$. Through it, multiple sets of the predicted error-free classification variable $\tilde{z}$ are returned, which play the same role as the true classification variable z. So far, we have come back to the status with no classification error.

Then multiple sets of $\tilde{z}^*$ are generated by introducing classification errors to the variable $\tilde{z}$ with the transition matrix $\mathbf{P}$ (Formula 1 in Section 2.1.1) where $p(\tilde{z}^* = 1|\tilde{z} = 1) = p_{11}$ and $p(\tilde{z}^* = 0|\tilde{z} = 0) = p_{00}$. Through it, bias and variance have been brought by classification errors to statistics $\tilde{\zeta}^*$. And the statistics $\tilde{\zeta}^*$ can be calculated from formulas in Table 1 by replacing z_i with $\tilde{z}_i^*$ in the formulas. Therefore, bias is estimated by $\mathbf{Bias_{comb}} = \mathrm{E}_{\tilde{z}}((\mathrm{E}(\tilde{\zeta}^*|\hat{z}, \tilde{z}) - \tilde{\zeta})|\hat{z})$ and variance is estimated by $\mathbf{Var_{comb}} = \mathrm{E}_{\tilde{z}}(\mathrm{Var}(\zeta^*|\hat{z}, \tilde{z})|\hat{z})$, where $\tilde{\zeta}$ is the statistics with classes $\tilde{z}$ and $\tilde{\zeta}^*$ is the statistics with classes $\tilde{z}^*$. Here, again, these are empirical expectations and variances of the simulated values, used as a numerical approximation to the true expectation and variance. The outer expectation is evaluated using the outer for loop (over s_1), the inner expectation/variance is evaluated using the inner for loop (over s_2).

Algorithm 3 The Combined Method

Input: Variable y, $\hat{z}$ and S_1, S_2.

1: Estimate parameters of the model $\boldsymbol{\theta}$, including p_{11} and p_{00}, from the EM algorithm, conditional on $\hat{z}$ and y
2: Calculate $\tilde{\zeta}$ and $p(z|y, \hat{z})$
3: **for** $s_1 = 1 \dots S_1$ **do**
4: Generate $\tilde{z}$ by $p(z|y, \hat{z})$ for every unit i in the data set
5: **for** $s_2 = 1 \dots S_2$ **do**
6: Generate $\tilde{z^*}$ by $\mathbf{P}$ (Matrix 1), conditional on $\tilde{z}$ for every unit i in the data set
7: Calculate the corresponding $\tilde{\zeta^*}$ for each $\tilde{z^*}$
8: **end for**
9: **end for**
10: Calculate $\mathbf{Bias_{comb}} = \mathrm{E}_{\tilde{z}}((\mathrm{E}(\tilde{\zeta}^*|\hat{z}, \tilde{z}) - \tilde{\zeta})|\hat{z})$, $\mathbf{Var_{comb}} = \mathrm{E}_{\tilde{z}}(\mathrm{Var}(\zeta^*|\hat{z}, \tilde{z})|\hat{z})$ over S_1 and S_2 bootstraps

Output: $\mathbf{Bias_{comb}}$, $\mathbf{Var_{comb}}$ and matrix $\mathbf{P}$

In our study, the three methods will be applied both in the simulation study (Section 3) and in the case study (Section 4). The estimated bias and variance from the three methods will be compared to each other. What's

more, the true bias and variance will be estimated by formulas 3 in Section 2.1.3. The performance of bias and variance estimation of the three methods will be assessed by comparing their estimated bias and variance with the true bias and variance.

2.3 EM Algorithm

The EM algorithm, with its full name as Expectation-Maximization algorithm, is used to estimate parameters by achieving maximum likelihood (Dempster, Laird, & Rubin, 1977; Little & Rubin, 2002). As illustrated by its name, it contains two steps: E step & M step. The E step describes the expected value of the complete-data log-likelihood function, conditional on the observed data. The M step then finds the parameters that maximize the log likelihood function.

In our study, the EM algorithm is applied in the EM method and the combined method to estimate the accuracy of domain statistics (details in Section 2.2). In the EM method, we apply the EM algorithm to calculate the maximum likelihood statistics $\tilde{\zeta}$ and estimate $\mathbf{Bias_{em}}$. In the combined method, the parameters estimated from the EM algorithm help us to come back to the status with no classification error.

In order to explain how the EM algorithm works in our methods, we start with a derivation of the log-likelihood function of the Gaussian mixture model in the study.

2.3.1 Log-likelihood Function for Complete Data in the Mixture Model

In Section 2.1.2, we have already introduced the Gaussian mixture model in our study. We assume that the continuous variable y for each class conforms to a mixture of normal distributions. Besides, we also assume that the classification errors only depend on the true classification variable z and not on the variable y. Therefore, the density function of the complete data $(z, m, \hat{z}, y)$ for unit i is:

$$\begin{aligned} f_{\boldsymbol{\theta}}(z = z_i, m = m_i, \hat{z} = \hat{z}_i, y = y_i) &= f_{\boldsymbol{\theta}}(z = z_i, m = m_i) f_{\boldsymbol{\theta}}(\hat{z} = \hat{z}_i | z = z_i) f_{\boldsymbol{\theta}}(y = y_i | z = z_i, m = m_i) \\ &= P(z = z_i, m = m_i) P(\hat{z} = \hat{z}_i | z = z_i) f_{\boldsymbol{\theta}}(y = y_i | z = z_i, m = m_i). \end{aligned}$$

Here, $\boldsymbol{\theta} = (\alpha_1, p_{11}, p_{00}, \pi_{1j}, \pi_{0k}, \mu_{1j}, \mu_{0k}, \sigma_{1j}, \sigma_{0k})$ stands for the full set of parameters in our model.

When $z_i = 1, m_i = j$ with $j = 1, 2, ..., n_1$,

$$f_{\boldsymbol{\theta}}\left(z = 1, m = j, \hat{z} = \hat{z}_i, y = y_i\right) = \alpha_1 p_{11}^{\hat{z}_i} \left(1 - p_{11}\right)^{1 - \hat{z}_i} \frac{\pi_{1j}}{\sigma_{1j}\sqrt{2\pi}} \exp\left\{ -\frac{\left(y_i - \mu_{1j}\right)^2}{2\sigma_{1j}^2} \right\} \triangleq \omega_{1ji}\,; \tag{4}$$

When $z_i = 0, m_i = k$ with $k = 1, 2, ..., n_0$,

$$f_{\boldsymbol{\theta}}\left(z = 0, m = k, \hat{z} = \hat{z}_i, y = y_i\right) = \left(1 - \alpha_1\right) \left(1 - p_{00}\right)^{\hat{z}_i} p_{00}^{1 - \hat{z}_i} \frac{\pi_{0k}}{\sigma_{0k}\sqrt{2\pi}} \exp\left\{ -\frac{\left(y_i - \mu_{0k}\right)^2}{2\sigma_{0k}^2} \right\} \triangleq \omega_{0ki}\,. \tag{5}$$

In order to indicate which component of the mixed distribution unit i belongs to, indication functions for m_i

are set as:

$$\mathbf{1}_{(m_i=j)} = \begin{cases} 1 & m_i = j \\ 0 & m_i \neq j \end{cases}; \qquad \mathbf{1}_{(m_i=k)} = \begin{cases} 1 & m_i = k, \\ 0 & m_i \neq k \end{cases}. \tag{6}$$

Note that $z_i \left(\sum_{j=1}^{n_1} \mathbf{1}_{(m_i=j)} \right) = z_i$ and $(1-z_i) \left(\sum_{k=1}^{n_0} \mathbf{1}_{(m_i=k)} \right) = 1 - z_i$ for all units i. Therefore, the likelihood function for $\boldsymbol{\theta} = (\alpha_1, p_{11}, p_{00}, \pi_{1j}, \pi_{0k}, \mu_{1j}, \mu_{0k}, \sigma_{1j}, \sigma_{0k})$ conditional on the complete data $(z, m, \hat{z}, y)$ is:

$$L(\boldsymbol{\theta}|z, m, \hat{z}, y) = \prod_{i=1}^{N} \left[\prod_{j=1}^{n_1} f_{\boldsymbol{\theta}}^{z_i \mathbf{1}_{(m_i=j)}} (z=1, m=j, \hat{z}=\hat{z}_i, y=y_i) \prod_{k=1}^{n_0} f_{\boldsymbol{\theta}}^{(1-z_i)\mathbf{1}_{(m_i=k)}} (z=0, m=k, \hat{z}=\hat{z}_i, y=y_i) \right],$$

where $\boldsymbol{\theta}$ stands for the full set of parameters which need to be estimated.

Then, the log-likelihood function for $\boldsymbol{\theta}$ conditional on the complete data $(z, m, \hat{z}, y)$ is derived by:

$$\begin{aligned} LL(\boldsymbol{\theta}|z, m, \hat{z}, y) =& \sum_{i=1}^{N} \left[\sum_{j=1}^{n_1} z_i \mathbf{1}_{(m_i=j)} \log f_{\boldsymbol{\theta}} (z=1, m=j, \hat{z}=\hat{z}_i, y=y_i) \right. \\ & \left. + \sum_{k=1}^{n_0} (1-z_i) \mathbf{1}_{(m_i=k)} \log f_{\boldsymbol{\theta}} (z=0, m=k, \hat{z}=\hat{z}_i, y=y_i) \right\} \\ =& \sum_{i=1}^{N} \left\{ \sum_{j=1}^{n_1} z_i \mathbf{1}_{(m_i=j)} \log \left[\alpha_1 p_{11}^{\hat{z}_i} (1-p_{11})^{1-\hat{z}_i} \frac{\pi_{1j}}{\sigma_{1j}\sqrt{2\pi}} \exp\{-\frac{(y_i-\mu_{1j})^2}{2\sigma_{1j}^2}\} \right] \right. \\ & \left. + \sum_{k=1}^{n_0} (1-z_i) \mathbf{1}_{(m_i=k)} \log \left[(1-\alpha_1)(1-p_{00})^{\hat{z}_i} p_{00}^{1-\hat{z}_i} \frac{\pi_{0k}}{\sigma_{0k}\sqrt{2\pi}} \exp\{-\frac{(y_i-\mu_{0k})^2}{2\sigma_{0k}^2}\} \right] \right\}. \end{aligned}$$

2.3.2 E Step and M Step

We need to find parameters $\boldsymbol{\theta} = (\alpha_1, p_{11}, p_{00}, \pi_{1j}, \pi_{0k}, \mu_{1j}, \mu_{0k}, \sigma_{1j}, \sigma_{0k})$ that maximize the above log-likelihood function. Therefore, partial derivative of the log-likelihood function is taken for each parameter (with $j = 1, ..., n_1$ and $k = 1, ..., n_0$):

$$\begin{aligned} \frac{\partial LL}{\partial \alpha_1} &= \frac{\sum_{i=1}^{N} z_i}{\alpha_1} - \frac{\sum_{i=1}^{N} (1-z_i)}{1-\alpha_1}, \\ \frac{\partial LL}{\partial p_{11}} &= \frac{\sum_{i=1}^{N} z_i \hat{z}_i}{p_{11}} - \frac{\sum_{i=1}^{N} z_i (1-\hat{z}_i)}{1-p_{11}}, \\ \frac{\partial LL}{\partial p_{00}} &= \frac{\sum_{i=1}^{N} (1-z_i)(1-\hat{z}_i)}{p_{00}} - \frac{\sum_{i=1}^{N} (1-z_i)\hat{z}_i}{1-p_{00}}, \\ \frac{\partial LL}{\partial \pi_{1j}} &= \frac{\sum_{i=1}^{N} z_i \mathbf{1}_{(m_i=j)}}{\pi_{1j}} - \frac{\sum_{i=1}^{N} z_i \mathbf{1}_{(m_i=n_1)}}{\pi_{1n_1}}, \\ \frac{\partial LL}{\partial \pi_{0k}} &= \frac{\sum_{i=1}^{N} (1-z_i) \mathbf{1}_{(m_i=k)}}{\pi_{0k}} - \frac{\sum_{i=1}^{N} (1-z_i) \mathbf{1}_{(m_i=n_0)}}{\pi_{0n_0}}, \\ \frac{\partial LL}{\partial \mu_{1j}} &= \frac{\sum_{i=1}^{N} z_i \mathbf{1}_{(m_i=j)} (y_i - \mu_{1j})}{\sigma_{1j}^2}, \end{aligned}$$

$$\frac{\partial LL}{\partial \mu_{0k}} = \frac{\sum_{i=1}^{N}(1-z_i)\,\mathbf{1}_{(m_i=k)}\,(y_i-\mu_{0k})}{\sigma_{0k}^2}\,,$$
$$\frac{\partial LL}{\partial \sigma_{1j}} = -\frac{\sum_{i=1}^{N} z_i \mathbf{1}_{(m_i=j)}}{\sigma_{1j}} + \frac{\sum_{i=1}^{N} z_i \mathbf{1}_{(m_i=j)}\,(y_i-\mu_{1j})^2}{\sigma_{1j}^3}\,,$$
$$\frac{\partial LL}{\partial \sigma_{0k}} = -\frac{\sum_{i=1}^{N}(1-z_i)\,\mathbf{1}_{(m_i=k)}}{\sigma_{0k}} + \frac{\sum_{i=1}^{N}(1-z_i)\,\mathbf{1}_{(m_i=k)}\,(y_i-\mu_{0k})^2}{\sigma_{0k}^3}\,.$$

The parameters that maximize the log-likelihood are returned when all these partial derivatives are equal to zero (with $j = 1, ..., n_1$ and $k = 1, ..., n_0$):

$$\begin{aligned}
\alpha_1 &= \frac{\sum_{i=1}^{N} z_i}{N}\,,\\
p_{11} &= \frac{\sum_{i=1}^{N} z_i \hat{z}_i}{\sum_{i=1}^{N} z_i}\,,\\
p_{00} &= \frac{\sum_{i=1}^{N}(1-z_i)(1-\hat{z}_i)}{\sum_{i=1}^{N}(1-z_i)}\,,\\
\pi_{1j} &= \frac{\sum_{i=1}^{N} z_i \mathbf{1}_{(m_i=j)}}{\sum_{i=1}^{N} z_i}\,,\\
\pi_{0k} &= \frac{\sum_{i=1}^{N}(1-z_i)\,\mathbf{1}_{(m_i=k)}}{\sum_{i=1}^{N}(1-z_i)}\,,\\
\mu_{1j} &= \frac{\sum_{i=1}^{N} z_i \mathbf{1}_{(m_i=j)} y_i}{\sum_{i=1}^{N} z_i \mathbf{1}_{(m_i=j)}}\,,\\
\mu_{0k} &= \frac{\sum_{i=1}^{N}(1-z_i)\,\mathbf{1}_{(m_i=k)} y_i}{\sum_{i=1}^{N}(1-z_i)\,\mathbf{1}_{(m_i=k)}}\,,\\
\sigma_{1j} &= \sqrt{\frac{\sum_{i=1}^{N} z_i \mathbf{1}_{(m_i=j)}\,(y_i-\mu_{1j})^2}{\sum_{i=1}^{N} z_i \mathbf{1}_{(m_i=j)}}}\,,\\
\sigma_{0k} &= \sqrt{\frac{\sum_{i=1}^{N}(1-z_i)\,\mathbf{1}_{(m_i=k)}\,(y_i-\mu_{0k})^2}{\sum_{i=1}^{N}(1-z_i)\,\mathbf{1}_{(m_i=k)}}}\,.
\end{aligned}$$

Since in practice, z and m are unknown, the corresponding expectations are taken in the EM algorithm. Before reaching that, the conditional probability $f_{\boldsymbol{\theta}}(z = z_i, m = m_i | \hat{z} = \hat{z}_i, y = y_i)$ is calculated,

$$f_{\boldsymbol{\theta}}(z = z_i, m = m_i | \hat{z} = \hat{z}_i, y = y_i) = \frac{f_{\boldsymbol{\theta}}(z = z_i, m = m_i, \hat{z} = \hat{z}_i, y = y_i)}{\sum_{(z_i,m_i)} f_{\boldsymbol{\theta}}(z = z_i, m = m_i, \hat{z} = \hat{z}_i, y = y_i)}\,. \tag{7}$$

Therefore, the expectation of $z_i \mathbf{1}_{(m_i=j)}$ and $(1-z_i)\,\mathbf{1}_{(m_i=k)}$ are calculated as:

$$\begin{aligned}
E\left(z_i \mathbf{1}_{(m_i=j)} | \hat{z} = \hat{z}_i, y = y_i\right) &= f_{\boldsymbol{\theta}}(z = 1, m = j | \hat{z} = \hat{z}_i, y = y_i) = \frac{\omega_{1ji}}{\sum_{j=1}^{n_1}\omega_{1ji} + \sum_{k=1}^{n_0}\omega_{0ki}} \triangleq A_{1ji}\,,\\
E\left((1-z_i)\,\mathbf{1}_{(m_i=k)} | \hat{z} = \hat{z}_i, y = y_i\right) &= f_{\boldsymbol{\theta}}(z = 0, m = k | \hat{z} = \hat{z}_i, y = y_i) = \frac{\omega_{0ki}}{\sum_{j=1}^{n_1}\omega_{1ji} + \sum_{k=1}^{n_0}\omega_{0ki}} \triangleq A_{0ki}\,,
\end{aligned}$$

and from this it follows that the expectations of z_i and $(1-z_i)$ are calculated as:

$$\begin{aligned}
E\left(z_i|\hat{z}=\hat{z}_i, y=y_i\right) &= \sum_{j=1}^{n_1} f_{\boldsymbol{\theta}}\left(z=1, m=j|\hat{z}=\hat{z}_i, y=y_i\right) \\
&= \frac{\sum_{j=1}^{n_1}\omega_{1ji}}{\sum_{j=1}^{n_1}\omega_{1ji}+\sum_{k=1}^{n_0}\omega_{0ki}} \\
&= \sum_{j=1}^{n_1} A_{1ji}\,, \\
E\left(1-z_i|\hat{z}=\hat{z}_i, y=y_i\right) &= \sum_{k=1}^{n_0} f_{\boldsymbol{\theta}}\left(z=0, m=k|\hat{z}=\hat{z}_i, y=y_i\right) \\
&= \frac{\sum_{k=1}^{n_0}\omega_{0ki}}{\sum_{j=1}^{n_1}\omega_{1ji}+\sum_{k=1}^{n_0}\omega_{0ki}} \\
&= \sum_{k=1}^{n_0} A_{0ki}\,.
\end{aligned} \tag{8}$$

By replacing the true (unknown) classification variable z and the (unknown) component variable m with the above expectations, parameters in EM algorithm become (with $j=1,...,n_1$ and $k=1,...,n_0$):

$$\begin{aligned}
\alpha_1^{(t+1)} &= \frac{\sum_{i=1}^{N}\left(\sum_{j=1}^{n_1} A_{1ji}^{(t)}\right)}{N}\,, \\
p_{11}^{(t+1)} &= \frac{\sum_{i=1}^{N}\left(\sum_{j=1}^{n_1} A_{1ji}^{(t)}\right)\hat{z}_i}{\sum_{i=1}^{N}\left(\sum_{j=1}^{n_1} A_{1ji}^{(t)}\right)}\,, \\
p_{00}^{(t+1)} &= \frac{\sum_{i=1}^{N}\left(\sum_{k=1}^{n_0} A_{0ki}^{(t)}\right)(1-\hat{z}_i)}{\sum_{i=1}^{N}\left(\sum_{k=1}^{n_0} A_{0ki}^{(t)}\right)}\,, \\
\pi_{1j}^{(t+1)} &= \frac{\sum_{i=1}^{N} A_{1ji}^{(t)}}{\sum_{i=1}^{N}\left(\sum_{j=1}^{n_1} A_{1ji}^{(t)}\right)}\,, \\
\pi_{0k}^{(t+1)} &= \frac{\sum_{i=1}^{N} A_{0ki}^{(t)}}{\sum_{i=1}^{N}\left(\sum_{k=1}^{n_0} A_{0ki}^{(t)}\right)}\,, \\
\mu_{1j}^{(t+1)} &= \frac{\sum_{i=1}^{N} A_{1ji}^{(t)} y_i}{\sum_{i=1}^{N} A_{1ji}^{(t)}}\,, \\
\mu_{0k}^{(t+1)} &= \frac{\sum_{i=1}^{N} A_{0ki}^{(t)} y_i}{\sum_{i=1}^{N} A_{0ki}^{(t)}}\,, \\
\sigma_{1j}^{(t+1)} &= \sqrt{\frac{\sum_{i=1}^{N} A_{1ji}^{(t)}\left(y_i-\mu_{1j}^{(t)}\right)^2}{\sum_{i=1}^{N} A_{1ji}^{(t)}}}\,, \\
\sigma_{0k}^{(t+1)} &= \sqrt{\frac{\sum_{i=1}^{N} A_{0ki}^{(t)}\left(y_i-\mu_{0k}^{(t)}\right)^2}{\sum_{i=1}^{N} A_{0ki}^{(t)}}}\,.
\end{aligned}$$

2.3.3 Log-likelihood Function for Observed Data

The log-likelihood function of $\boldsymbol{\theta}$ for the given observed data ($\hat{z}$ and y) is given:

$$\begin{aligned} LL\left(\boldsymbol{\theta}|\hat{z},y\right) &= \sum_{i}^{N} \log f_{\boldsymbol{\theta}}\left(\hat{z}=\hat{z}_i, y=y_i\right) \\ &= \sum_{i}^{N} \log\left(\sum_{z_i}\sum_{m_i} f_{\boldsymbol{\theta}}\left(\hat{z}=\hat{z}_i, y=y_i, z=z_i, m=m_i\right)\right) \\ &= \sum_{i}^{N} \log\left(\sum_{j=1}^{n_1}\omega_{1ji} + \sum_{k=1}^{n_0}\omega_{0ki}\right), \end{aligned} \tag{9}$$

where the true (unknown) values z and true (unknown) components m are left out.

The log-likelihood function for the observed data is used when choosing the optimal number of components with a criterion such as BIC, finding outliers in the model and avoiding the local maximum when using random starting values.

2.3.4 With an Audit Sample

An audit sample (details in Section 2.1.5) is selected to help with the initialization of the EM algorithm.

The use of the audit sample can also accelerate the convergence of the EM algorithm. Appendix A provides technical details of how to apply the audit sample to the EM algorithm, including combining the information of the audit sample into the E step and the M step, and choosing starting values.

It does not mean that the audit sample is necessary in our methods. Other initialization methods can also be used in the EM algorithm. The updated parameter estimates and log-likelihood function in the above sections work when there is no audit sample in the study.

2.3.5 Maximum Likelihood Statistics

When parameters of the model are obtained from the EM algorithm, the statistics of interest can be calculated . In our setting, statistics α_1 are directly estimated from the EM algorithm; μ_1 is calculated by $\sum_{j=1}^{n_1}\pi_{1j}\mu_{1j}$; μ_0 is calculated by $\sum_{k=1}^{n_0}\pi_{0k}\mu_{0k}$; $\sigma_1 = \sqrt{\sum_{j=1}^{n_1}\pi_{1j}\times\left(\sigma_{1j}^2+\left(\mu_{1j}-\mu_1\right)^2\right)}$; $\sigma_0 = \sqrt{\sum_{k=1}^{n_0}\pi_{0k}\times\left(\sigma_{0k}^2+\left(\mu_{0k}-\mu_0\right)^2\right)}$; and T_1 is calculated by $T_1 = N\times\alpha_1\times\mu_1$.

3 Simulation Study

To assess the performance of the bootstrap method, the EM method and the combined method (details in Section 2.2), a simulation study was conducted to test the methods under different conditions (described in Section 2.1.2). The bias and variance estimated from the three methods were compared with their corresponding true values.

3.1 Procedures

For the simulation study we used a population of $N = 2000$ units. For each unit we know the true classification variable z and the value of the continuous variable y. Different from the situation in practice, one of the variables is the true classification variable z, and another variable is the continuous variable y which is generated conditional on z (Formula 2 in Section 2.1.2). The settings of parameters of our model will be introduced later in Table 2.

Given the true classes z, 1000 sets of observed classes $\hat{z}$ were generated by using the transition matrix (Formula 1), where $p(\hat{z} = 1|z = 1) = p_{11}$ and $p(\hat{z} = 0|z = 0) = p_{00}$. Multiple values for $\hat{\zeta}$ were obtained by replacing z with $\hat{z}$ in the formulas in Table 1. Through that, the true bias and variance were estimated accordingly by $\mathbf{Bias_{true}} = E(\hat{\zeta}) - \zeta$ and $\mathbf{Var_{true}} = \mathrm{Var}(\hat{\zeta})$ over 1000 simulated values. Here, ζ denote the true value of any statistics of interest in Table 1 and $\hat{\zeta}$ denote the observed statistics of interest. We simulated $\hat{z}$ 1000 times to ensure an accurate estimate of the true bias and variance.

Another $S_0 = 100$ sets of observed classes $\hat{z}$ were generated in the same way. Given each set of $\hat{z}$ and y, 5% units from the population were random sampled as the audit sample, of which the true classes z were known. The three methods described in Section 2.2 were applied. The number of iterations in the bootstrap method and the combined method were set as $S = S_1 = S_2 = 100$. For each set of $\hat{z}$, bias and variance estimates were obtained for the three methods, such as $\mathbf{Bias_{boot}}$ and $\mathbf{Var_{boot}}$ from the bootstrap method. In the end, the performance of bias and variance estimation of the three methods were represented by the average over those S_0 bias and variance estimates.

3.2 Set-ups

Li (2020) has presented the performance of these methods in the setting of a normal distribution conditional on z with different values of N, α_1, p_{11} and p_{00}. In the current study, we only varied parameters in the mixture Gaussian distribution and considered $N = 2000$, $\alpha_1 = 0.5$, $p_{11} = 0.8$, $p_{00} = 0.8$ fixed in the simulation.

There were three set-ups in the simulation study. In these set-ups, the number of components and values of the parameters in the model of each class are varied. Table 2 shows the different setting of the components in

each class of the three set-ups. The number of mixture components ranges from 1 to 3 in the set-ups, which corresponds to the situation of data we have in the case study (Section 4). By manipulating the parameters of mixture components, we can evaluate the methods in various conditions.

Table 2: Parameters of Components in Different Settings in the Simulation Study

Set-up	Class	Components in Each Class			
		Number	Proportion	Mean	Standard Deviation
Set-up 1	Class 1	2	$(0.1, 0.9), (0.3, 0.7), (0.5, 0.5), (0.7, 0.3), (0.9, 0.1)$	$(2, 4), (2, 6), (2, 8)$	$(1, 2)$
	Class 0	1	-	15	3
Set-up 2	Class 1	2	$(0.1, 0.9), (0.3, 0.7), (0.5, 0.5), (0.7, 0.3), (0.9, 0.1)$	$(2, 4), (2, 6), (2, 8)$	$(1, 2)$
	Class 0	2	$(0.5, 0.5)$	$(15, 18)$	$(1, 3)$
Set-up 3	Class 1	3	$(0.1, 0.2, 0.7), (0.2, 0.3, 0.5), (0.3, 0.3, 0.4)$	$(2, 3, 4), (2, 4, 6), (2, 5, 8)$	$(1, 1, 2)$
	Class 0	2	$(0.5, 0.5)$	$(15, 18)$	$(1, 3)$

3.3 Results

3.3.1 Bias Estimation

The performance of bias estimation from the three methods was compared and evaluated by the true bias under the three set-ups in Figure 2 - 4, where the results were similar for the three set-ups. In the plots, the y-axis indicates the bias from different methods and the x-axis indicates the mean of components in class 1 (μ_{11}, μ_{12}); columns represent the components proportion in class 1 (π_{11}, π_{12}) and rows represent the domain statistics of interest ζ. For the points on the plot, the red dot stands for the true values; the symbol $\triangle$ stands for the bootstrapped estimates; the symbol $\bigtriangledown$ stands for the estimates from the EM method; the symbol $\times$ stands for the estimates from the combined method.

The estimated biases from the EM method (symbol $\bigtriangledown$) and the combined method (symbol $\times$) were closer to its true value (red dot) compared to the one estimated from the bootstrap method (symbol $\triangle$) for all estimators in general. Exceptions occurred under some cases for α_1, where the estimates for all three methods were quite similar.

For T_1, the bias estimated by the EM method and the combined method were accurate, while the bias estimated by the bootstrap method tended to underestimate the bias caused by classification errors. When $(\pi_{11}, \pi_{12}) = (0.1, 0.9)$, $\mu_{11} = 2$ and $\mu_0 = 15$, with μ_{12} increasing from 4 to 8, the bias caused by classification

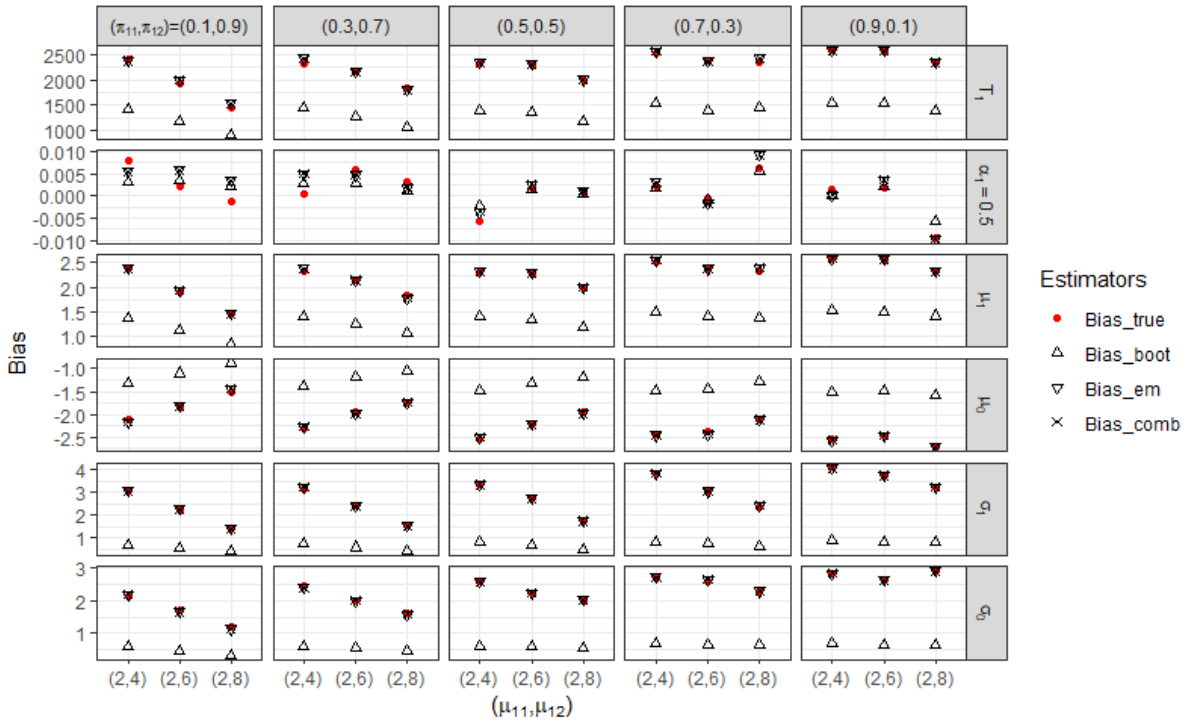


Figure 2: Bias estimation for set-up 1.

error decreased. This makes sense since classification errors tend to average the distributions of the two classes. When the distributions of the two classes gets closer, the difference of domain total in each class is smaller and the extent of bias that classification errors can cause will be lower. In an extreme case where the accuracy of classification error p_{11} and p_{00} were 0, the estimated value of T_1 would be equal to the true value of $total_0$ and the difference of domain *total* would be equal to its bias. Therefore, with closer distance of two classes, the absolute bias for T_1 and $total_0$ will be lower. The same is true also for the domain means μ_1 and μ_0.

The estimated bias for α_1 from the bootstrap method was as close to the true bias as the other two methods. This is reasonable when considering the variance of the true biases and these bias estimations and the scale of the y-axis in the second row compared to other statistics. Actually, with $\alpha_1 = 0.5$, $p_{00} = 0.8$ and $p_{11} = 0.8$, the theoretical true bias of the observed α_1 is zero (Scholtus & Van Delden, 2020). The deviations from zero of the red dot for alpha in Figure 2 - 4 are all due to simulation noise.

For μ_1 and μ_0, the bias estimates from the EM method and the combined method were close to the true biases and the bootstrap method tended to underestimate the bias. The bias of μ_1 was always positive while that for μ_0 was always negative. With $\alpha_1 = 0.5$, the absolute bias for μ_1 and μ_0 should be equal which was also validated from Figure 2.

For σ_1 and σ_0, their true biases overlapped with the estimates of bias from the EM method and the combined method and showed distance with the estimates from the bootstrapped method. When the distributions of the two classes are far apart, units that are misclassified will increase the spread of the original distribution where

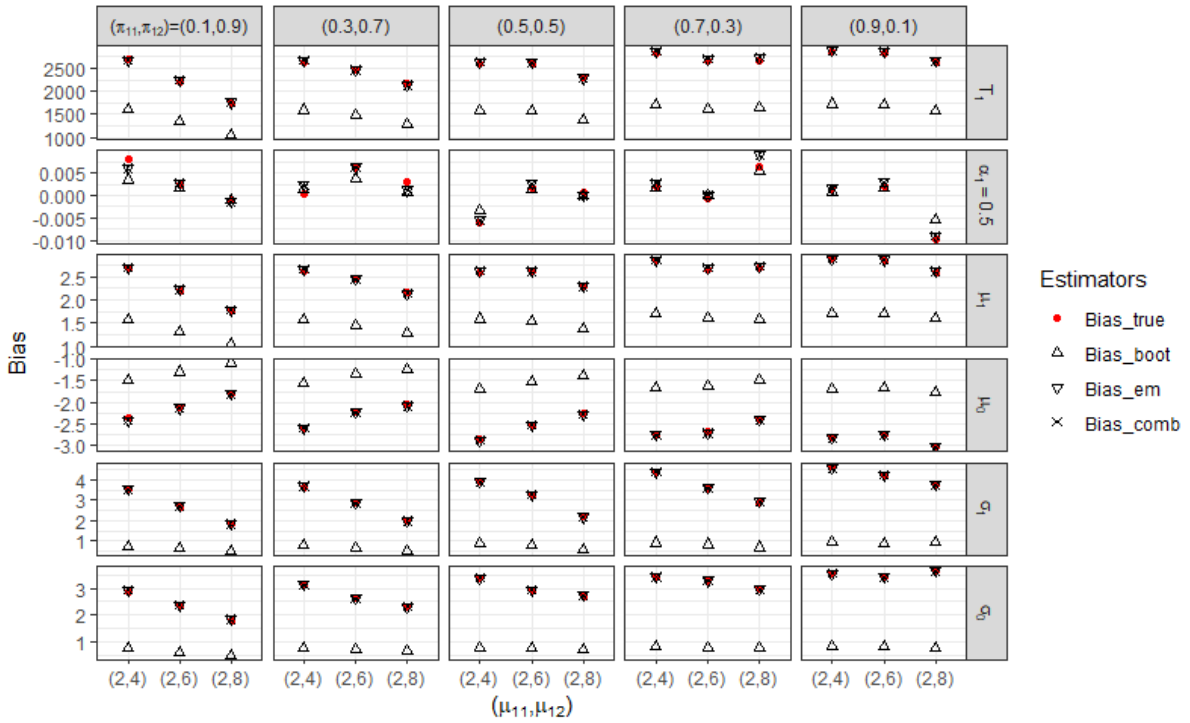


Figure 3: Bias estimation for set-up 2.

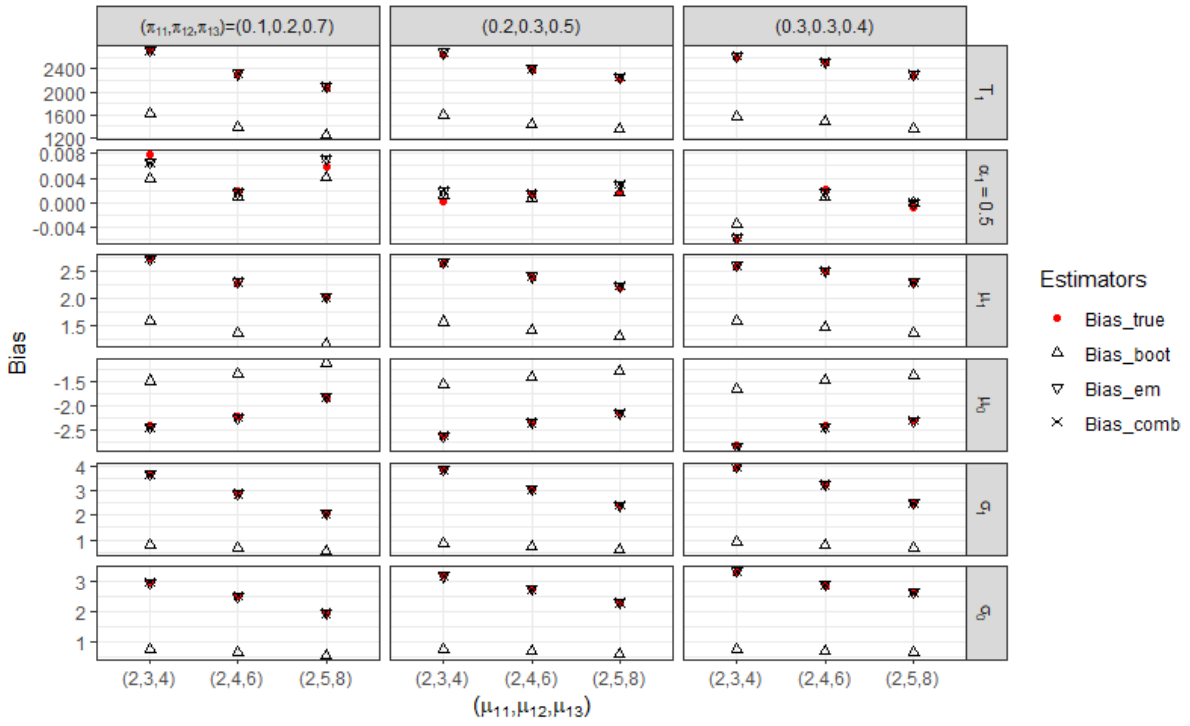


Figure 4: Bias estimation for set-up 3.

bias is generated. With two distributions getting closer, the distance of misclassified units from the original units will be lower and the absolute bias should become smaller. This is validated from Figure 2. When $(\pi_{11}, \pi_{12}) = (0.1, 0.9)$, $\mu_{11} = 2$ and $\mu_0 = 15$, with μ_{12} increasing from 4 to 8, their bias caused by classification

error also decreased.

From the plots, we notice that the bias estimate of the EM and the combined method are nearly the same for all of the statistics, and they are close to the true bias, which is reasonable. There are two elements in the definition of bias (Formula 3): the true values of statistics ζ ($\tilde{\zeta}$ are used to approximate ζ in the two methods) and the observed statistics $\hat{\zeta}$. The $\tilde{\zeta}$ from the two methods are the same, are all calculated through parameters estimated from EM algorithm (Section 2.3.5). By estimating the model parameters and misclassification probabilities, the combined method 'restores' the classes without and with classification error(s), which refer to $\tilde{z}$ and $\tilde{z}^*$ separately.

Furthermore, 95% confidence intervals of the bias estimates from the three methods and the estimated true bias in the set-up 1 were calculated. From Figure 5, the 95% confidence intervals for bias estimates from the EM method and the combined method are still closer to the true bias, compared to the bootstrap method. It suggests that in a practical situation where there is only one single set of $\hat{z}$, the EM and the combined method lead to better bias estimates than the bootstrap method.

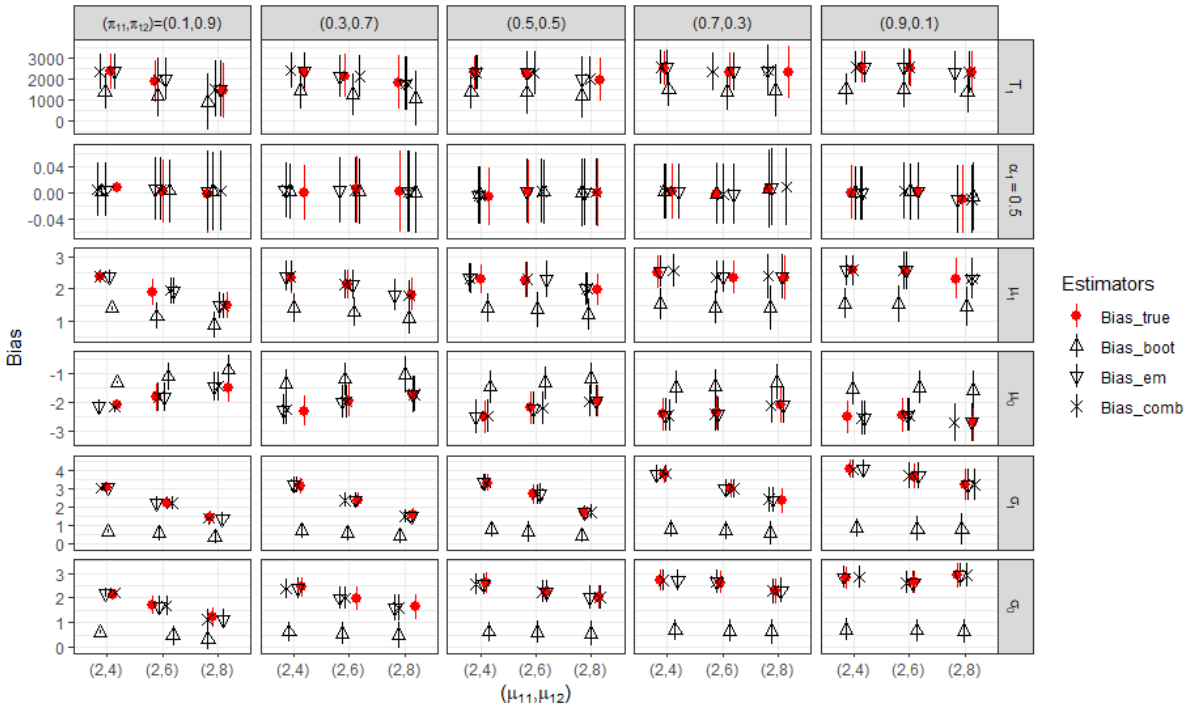


Figure 5: Bias estimates with 95% confidence intervals for set-up 1. The range of points stands for 95% confidence interval of the corresponding estimators.

3.3.2 Variance Estimation

In this section, the variance estimation performance was compared between the bootstrap method and the combined method and was evaluated by comparing them to the true values in Figure 6 to 8. To better show the results, the standard error, which is the square root of the variance for domain statistics, was shown to

illustrate the variance estimation in the plots.

The results for all three set-ups were similar. In the plots, the x-axis indicates the mean of components in class 1 (μ_{11}, μ_{12}) and the y-axis indicates the bias from different methods; columns represent the components proportion in class 1 (π_{11}, π_{12}) and rows represent the domain statistics of interest ζ. For the points on the plot, the red dot shows the true value; the symbol $\triangle$ shows the bootstrapped estimate; the symbol $\times$ shows the estimation from the combined method.

The variance estimation performance of the combined method in variance estimation was better than that of the bootstrap method for all estimators except α_1 under all conditions. For estimator α_1, the true value was closer to the estimation from the combined method only under some cases. For all other estimators, the estimated variance from the combined method was overlapped with its corresponding true variance. The bootstrapped estimates were also equal to the true values for estimator T_1 but tended to underestimate the variance for μ_1, μ_0, σ_1 and σ_0.

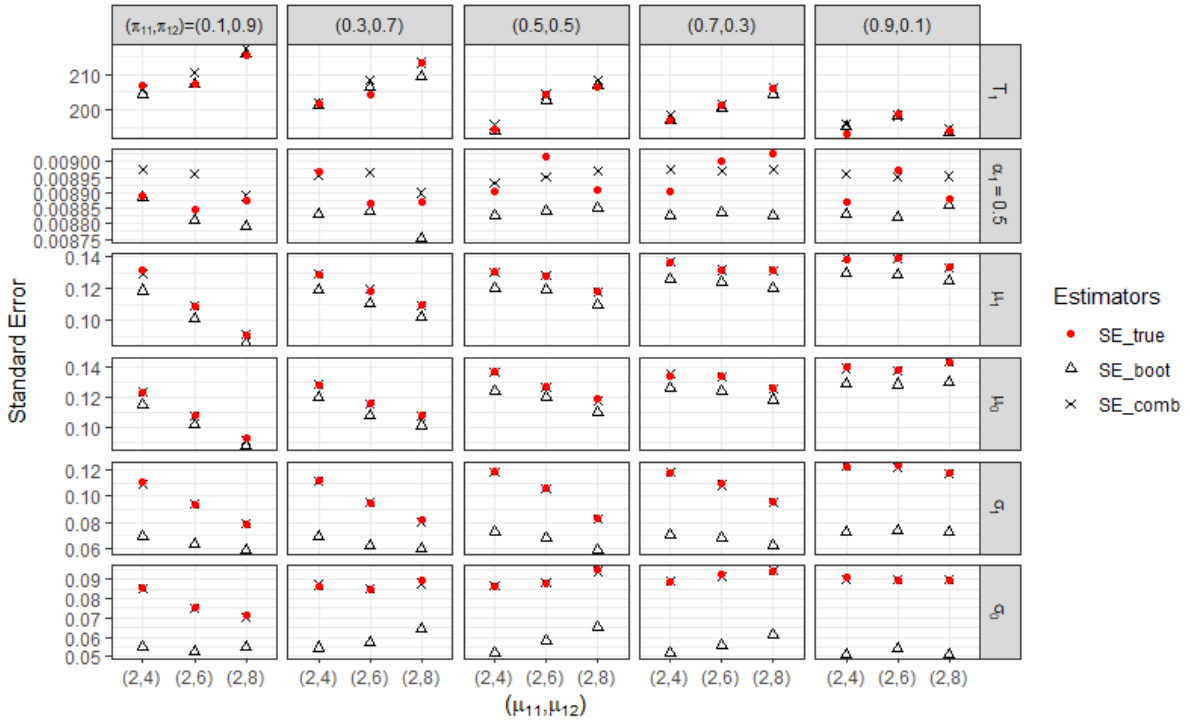


Figure 6: Variance estimation for set-up 1.

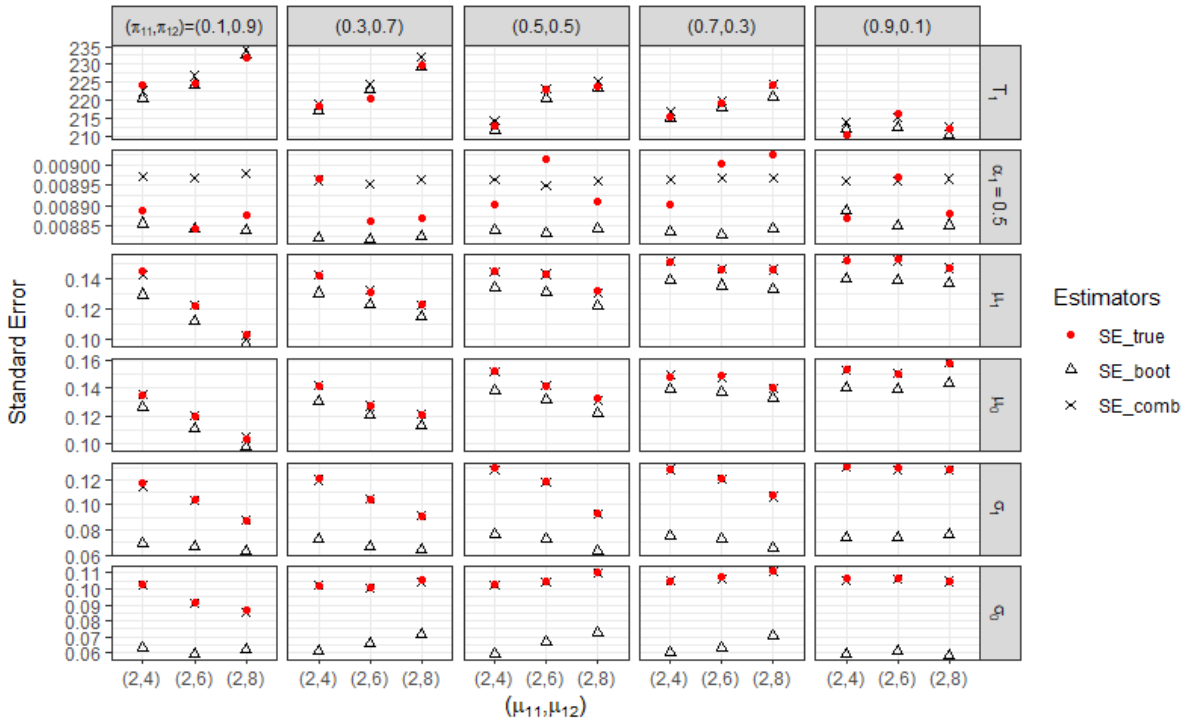


Figure 7: Variance estimation for set-up 2.

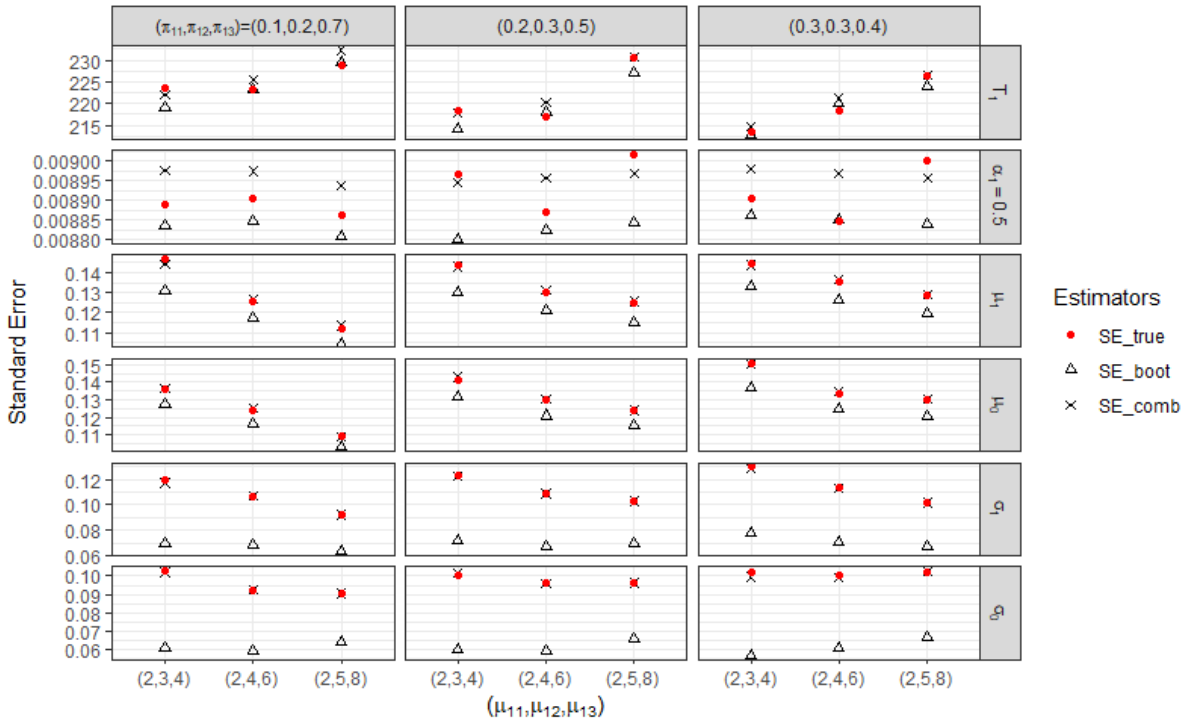


Figure 8: Variance estimation for set-up 3.

4 Case Study

In order to assess the performance of our methods in real applications, a case study was conducted. In the case study, the bias and variance of domain statistics, including subgroup total, proportion, mean and standard deviation, were estimated for an economic sector. The methods used in the case study were exactly the same as those we used in the simulation study of the previous chapter, which have been introduced in Section 2.2.

4.1 Data

For the case study, we started with a data set (Oosterveen, 2020) that contains log yearly turnover for a population of enterprises. The enterprises in this data set is divided into 25 economic activities, which are presented by NACE codes (Eurostat, 2008). Next, we made three selections to obtain the final data for the case study.

Firstly, the units of the population were limited to the smaller enterprises (with three or less legal units), because they contain most errors in economic activity. Economic activity codes of larger enterprises are checked manually, which usually does not lead to any classification error.

Secondly, the economic activities were checked and we only preserved the units with true economic activities. The economic activities are the observed ones, which contain classification errors. With only observed classification variable $\hat{z}$, we are not able to estimate true bias and variance, and the estimates of our methods will not be assessed. We therefore made a selection of units for which the NACE code is likely to be correct.

This selection of units with true classes was made by Oosterveen (2020, described the details of units selection). The website texts of the units (which can be seen as auxiliary variable) were used to train three machine learning models. The classes of the units were then predicted by the three machine learning algorithms. Only units whose codes were consistent with the results from three algorithms were selected and these codes were assumed as true classes.

Thirdly, certain classes were selected. We only considered our methods in the setting of a binary classifier in our study. Therefore, two classes were chosen for each test (Table 3). By matching two classes together for each test, we wanted to test our methods in different settings, where the distributions of the two classes have different 'distance' and the shapes of the distributions vary. We also made sure that the numbers of units in the two groups are not too small and not too unbalanced, which also means that α_1 in each test is not close to 0 or 1.

The distributions of log yearly turnover by 25 NACE codes were plotted in Figure 9. Seven groups (NACE codes) were selected to form the final 4 data sets (Case 1, 2, 3a and 3b). Table 3 has shown the allocation of the 7 groups and Table 4 has described the basic statistics of each group.

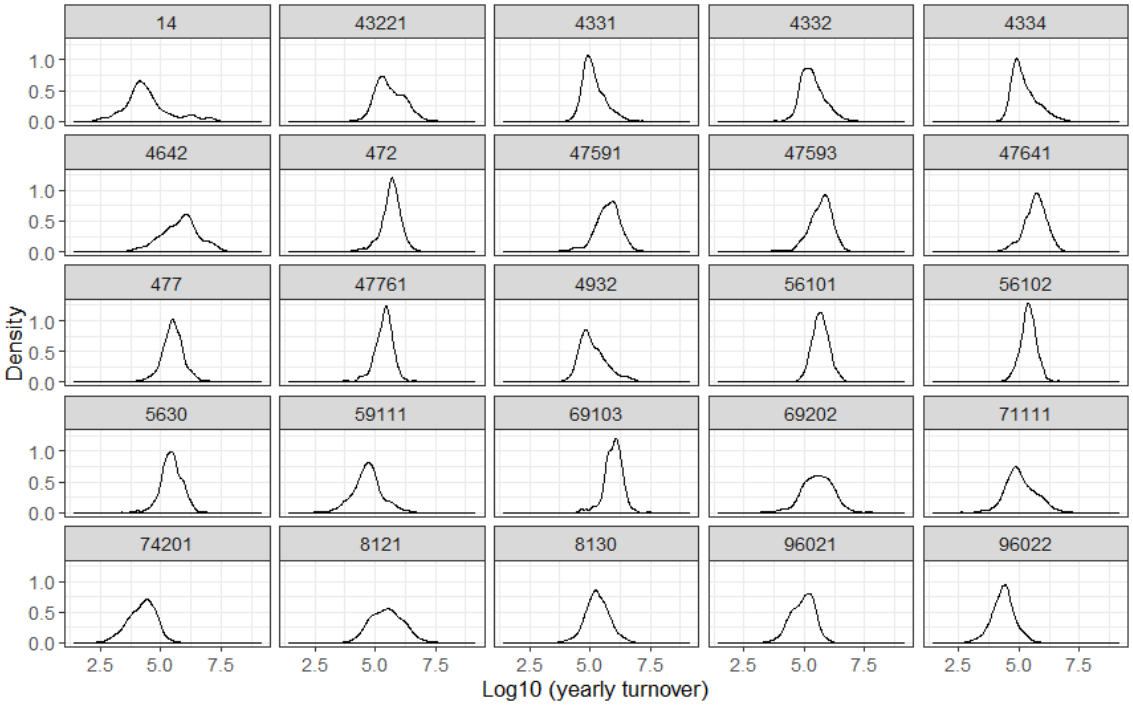


Figure 9: Density distribution of log turnover for all groups. The labels refer to NACE codes (Eurostat, 2008).

The four cases stand for different settings in the real application. In the first case, the distributions of the two classes are well separated with a similar shape. In the second case, the distributions of the two classes are also well separated but with different shapes. In the case 3a and 3b, the distributions of the two classes are not well separated with different shapes.

After the manipulation of the original data, the data used for each case was analysed by our methods. There are only two classes in each data set, and the classification variable in each data set was assumed to be true.

4.2 Procedures and Set-ups

The case study started with data sets with two columns, where each column stands for a variable. We regarded the log yearly turnover as the continuous variable y and the allocated classes as the true classification variable $z \in \{1, 0\}$ (Table 3). The number of rows differed among the cases. Table 4 describes the true domain statistics ζ of each class.

The procedures of the case study were similar to that of the simulation study, except that the number of components in the Gaussian mixture model needs to be obtained in the case study. Given z and y, we fitted the Gaussian mixture model to the data of each class and BIC (Bayesian information criterion) was chosen as a criterion for selecting the optimal number (McLachlan & Peel, 2004; Scrucca, Fop, Murphy, & Raftery, 2016) (Appendix C has more details about using the criterion of BIC). The selected optimal number of components is also listed in Table 4 for each class.

Table 3: The Selected Groups and Their Allocations in the Case Study

NACE Code	Description of Economic Activity	Case No.	Class
56101	Restaurants	Case 1	Class 1
96022	Beauty treatment, pedicures and manicures, make-up and image consulting	Case 1	Class 0
43221	Plumbing and fitting; installation of sanitary fittings	Case 2	Class 1
74201	Photography	Case 2	Class 0
4932	Taxi operation	Case 3a, 3b	Class 1
5630	Bars	Case 3a	Class 0
8121	General cleaning of buildings	Case 3b	Class 0

Given the true classification variable z, $S_0 = 100$ sets of observed classification variable $\hat{z}$ were generated by using the transition matrix (Formula 1 in Section 2.1.2), where $p(\hat{z} = 1 \,|\, z = 1) = p_{11}$ and $p(\hat{z} = 0 \,|\, z = 0) = p_{00}$.

Since p_{11} and p_{00} represent the probabilities of being classified correctly, they are always larger than 0.5. If any of them is lower than 0.5, it is useful to reverse the classification rules to make it higher than 0.5. We simulated three levels of classification error probabilities in the case study: high (0.6), medium (0.75), low (0.9). The values of p_{11} and p_{00} were chosen from 0.6, 0.75, 0.9.

Multiple $\hat{\zeta}$ were obtained by replacing z with $\hat{z}$ in the formulas in Table 1. Through that, the true bias and variance were estimated accordingly by $\mathbf{Bias_{true}} = E(\hat{\zeta}) - \zeta$ and $\mathbf{Var_{true}} = \mathrm{Var}(\hat{\zeta})$ which refer to empirical expectation and the empirical variance over S_0 simulated values.

Given each set of $\hat{z}$ and y, 5% units from the population were randomly selected as the audit sample, of which the true classes z were known. The three methods described in Section 2.2 were applied. By applying the three methods, the bias and variance estimates given each set of $\hat{z}$ were obtained, such as $\mathbf{Bias_{boot}}$ and $\mathbf{Var_{boot}}$ from the bootstrap method. The number of iterations in the bootstrap method and the combined method were set as $S = S_1 = S_2 = 100$. In the end, the performance of bias and variance estimation of the three methods were represented by the average over those S_0 bias and variance estimates.

Table 4: Domain Statistics for Each Class in the Case Study

Case No.	Class 1					Class 0				
	Number of Components	Size	Total	Mean	Standard Deviation	Number of Components	Size	Total	Mean	Standard Deviation
Case 1	1	3076	17415	5.66	0.358	2	6993	30377	4.34	0.501
Case 2	3	1535	8606	5.61	0.587	2	3932	16551	4.21	0.597
Case 3a	2	642	3294	5.13	0.597	2	947	5162	5.45	0.436
Case 3b						1	1067	5847	5.48	0.687

4.3 Results

4.3.1 Bias Estimation

The performance of bias estimation in the case study is shown from Figure 10 to 13. In the plots, the y-axis indicates the bias estimated from different methods and the x-axis indicates values of p_{11}; columns represent values of p_{00} and rows represent domain statistics of interest ζ. For the points on the plot, the red dot shows the true value; the symbol $\triangle$ shows the bootstrapped estimates; the symbol $\triangledown$ shows the estimation from the EM method; the symbol $\times$ shows the estimation from the combined method. The estimated biases from the bootstrap, EM and combined method were compared with the true bias.

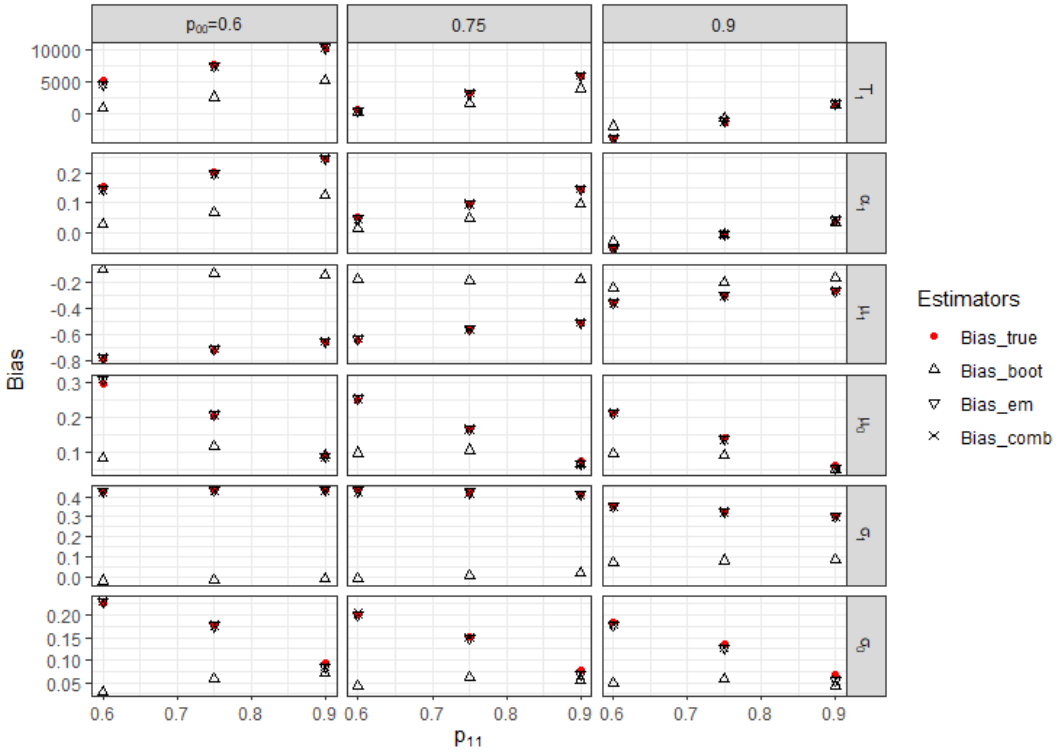


Figure 10: Bias estimation in case 1.

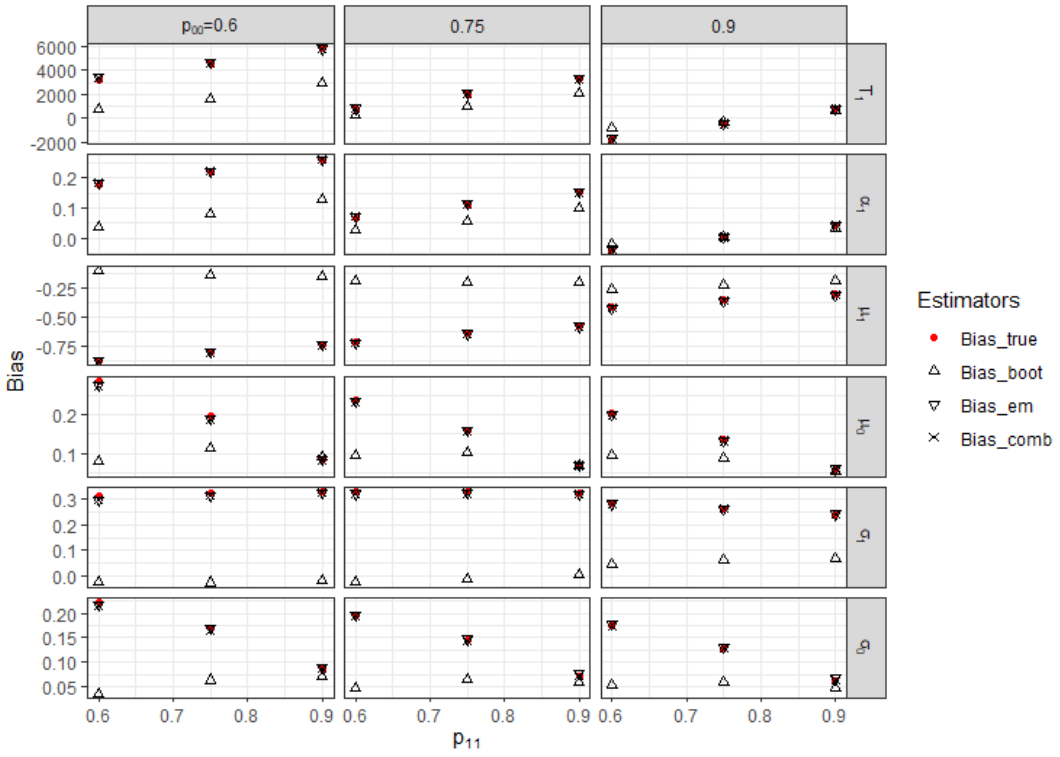


Figure 11: Bias estimation in case 2.

For case 1 and 2, the estimated bias from EM and combined method were overlapping with the corresponding true bias for all estimators; the bootstrap method always tended to underestimate the true bias. For case 3a and 3b, the results for T_1 and α_1 conformed to what had been found in case 1 and 2, while the results for subgroup mean μ_1, μ_0 and subgroup standard deviation σ_1, σ_0 had slightly different results.

Parameter p_{11} is the probability of a unit being classified as class 1 when its true class is also class 1; parameter p_{00} is the probability of a unit being classified as class 0 when its true class is class 0. With p_{00} fixed and p_{11} increased, the number of units being classified as class 1 was increased. Therefore, the bias of α_1 increased which conformed to what has been shown in the figures (Figure 10-13).

When p_{11} was increased and p_{00} was fixed, among those units being classified as class 1, the relative proportion of units whose true class was 1 became larger; among those units being classified as class 0, the relative proportion of units whose true class was 0 became larger. This can explain the tendency that the absolute bias for μ_1, μ_0, σ_1 and σ_0 decreased with larger p_{11} within each block of the plots.

In case 3a and 3b, the bias estimation from EM and combined method for T_1 and α_1 was overlapping with the true bias, while that for domain statistics μ_1, μ_0, σ_1 and σ_0 was not better than the estimation from the bootstrap method. The cases 3a and 3b compared to case 1 and 2 and various set-ups in the simulation study, have two differences: the distributions of the two classes are closer and the population size is smaller. Therefore, we simulated a condition with closer distributions of the two classes and also with a smaller population to test

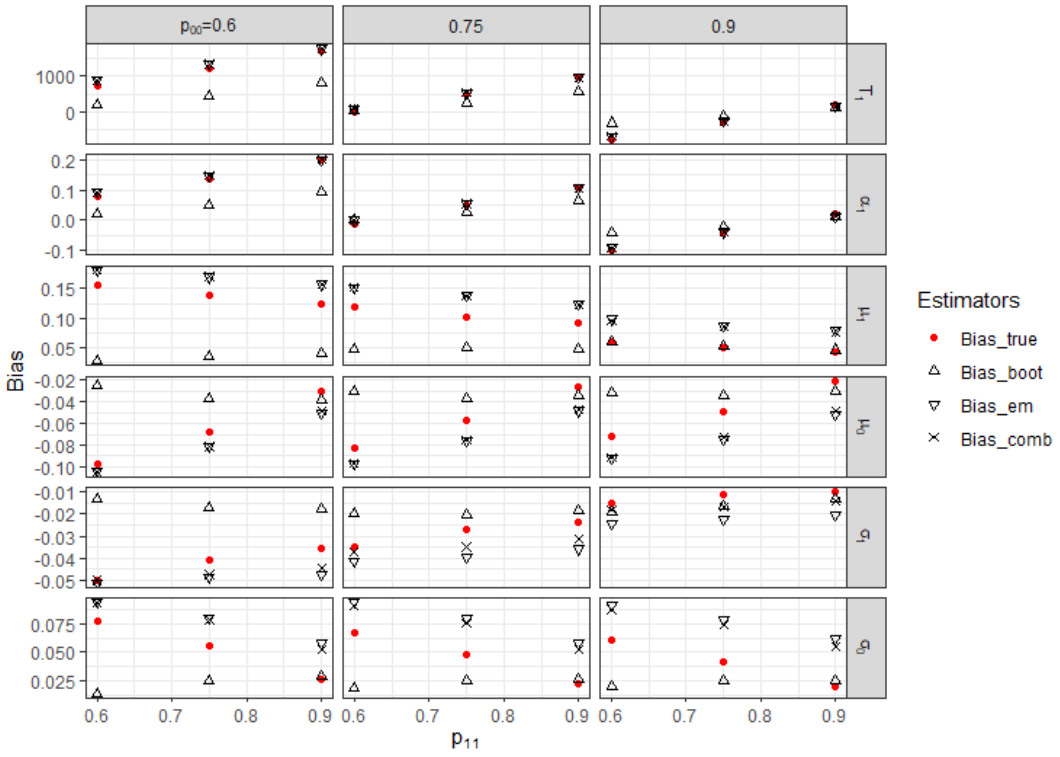


Figure 12: Bias estimation in case 3a.

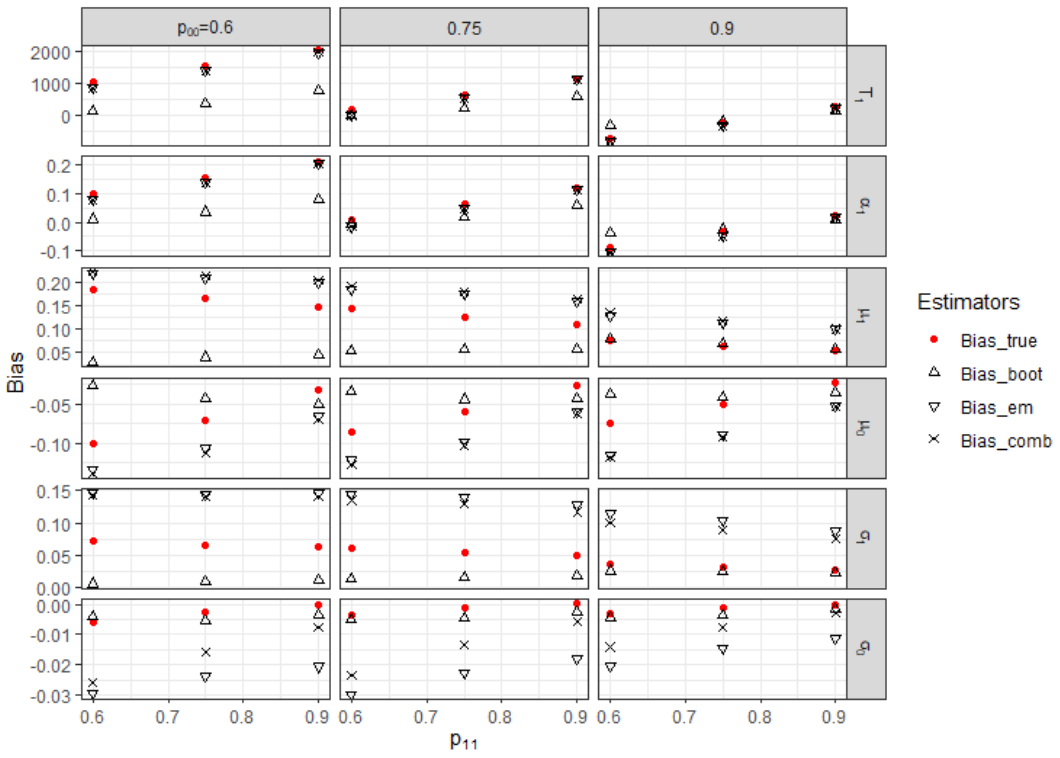


Figure 13: Bias estimation in case 3b.

whether the methods could work properly in such extreme conditions (see details in Appendix B). From the results of the experiments, neither of these two factors would influence the performance of our methods. Using further experiments, it was found that it was the outliers in these cases that influenced the estimation of our method (Appendix B).

4.3.2 Variance Estimation

The performance of variance estimation from bootstrap and combined method for all cases is shown in Figure 14 - 17. To better show the results, the standard error, which is the square root of the variance for domain statistics, was shown to illustrate the variance estimation in the plots.

The results from all cases suggest that the variance estimation from the bootstrap method was already close to that from the combined method. Comparing the estimation results from the bootstrap method, the estimate from the combined method was usually still closer to the true value.

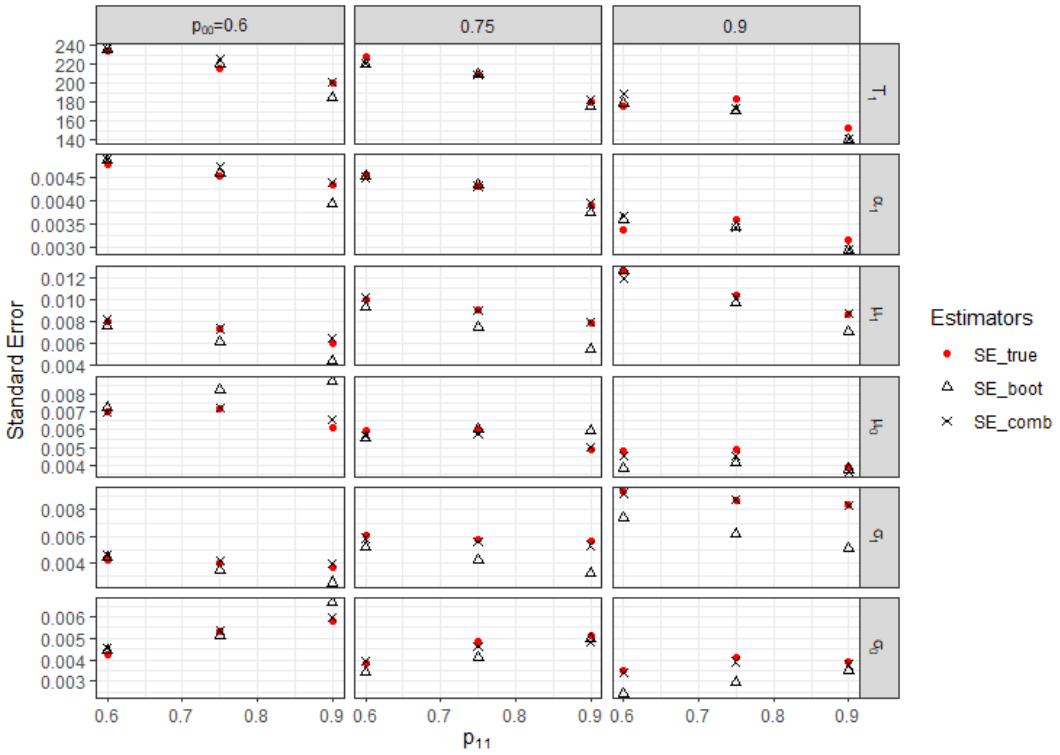


Figure 14: Variance estimation in case 1.

Comparing the variance estimation results from the simulation study, the gap between the bootstrap method and the combined method was closer in the case study. One of the reasons might be that the distributions of the two classes are much closer.

In order to test this assumption, we simulated a setting with closer distributions where the mean of the components in class 1 was set as $(\mu_{11}, \mu_{12}) = (2, 4)$ and the mean in class 0 was set as $\mu_0 = 5$ (see experiment

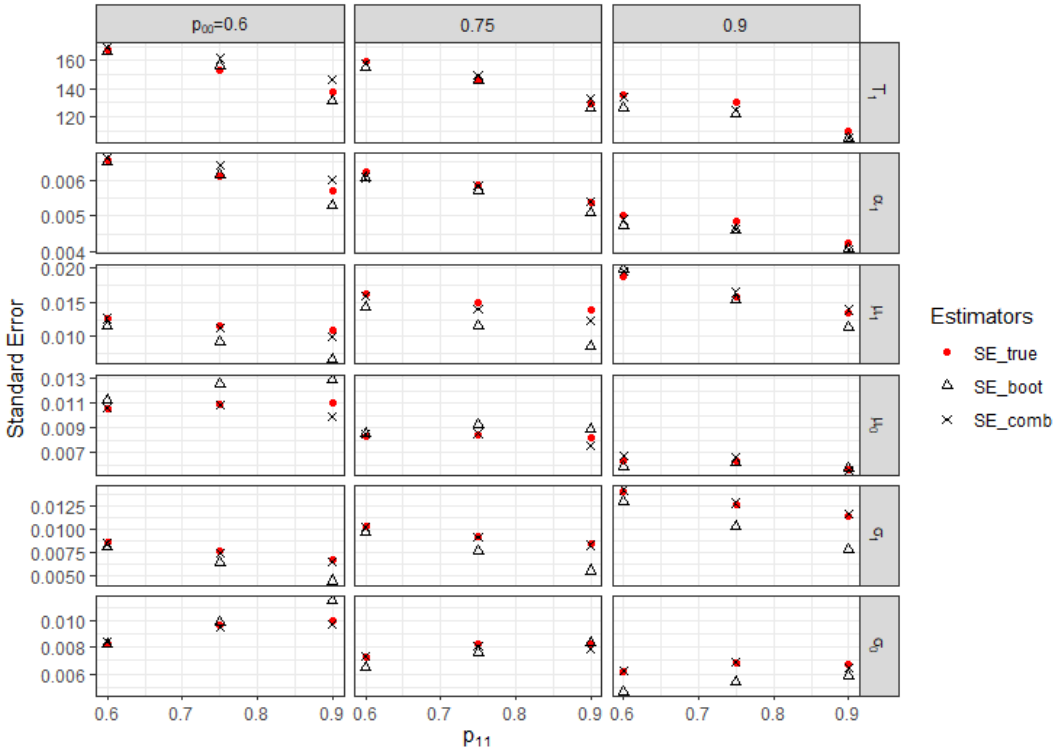


Figure 15: Variance estimation in case 2.

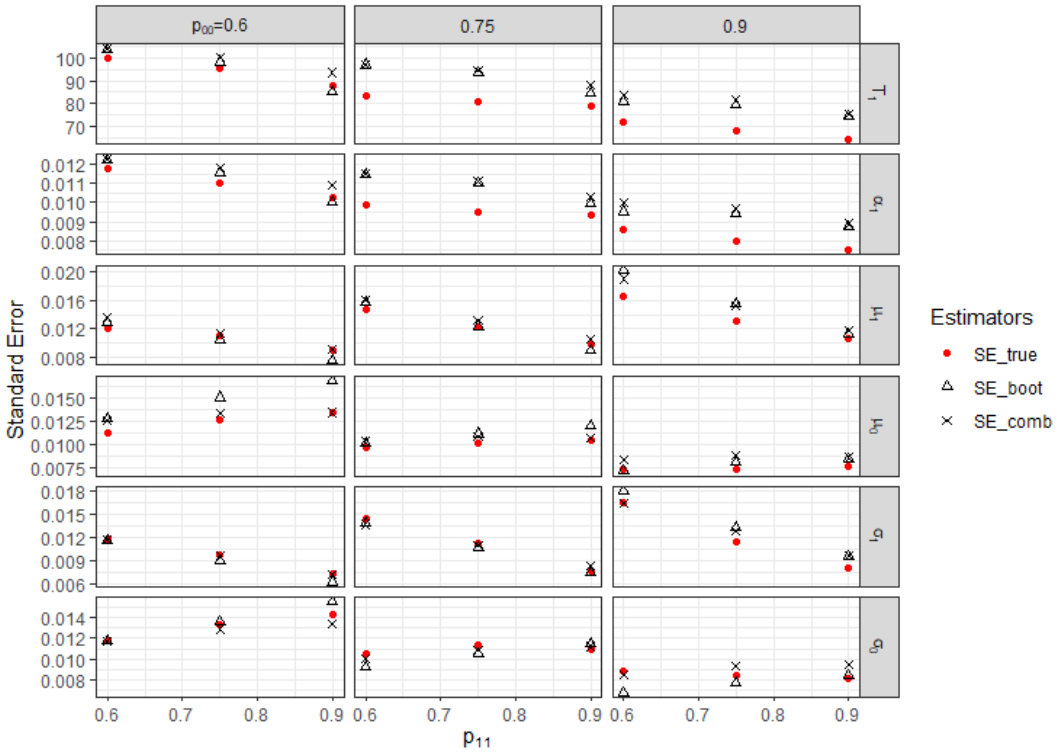


Figure 16: Variance estimation in case 3a.

1 in Appendix B). Under this condition, the variance estimates from the bootstrap and the combined methods for domain statistics T_1, α_1, μ_0 and σ_0 were close to what was shown in the case study, and the variance estimation from the combined method was far from the true variance. When the simulated number of $\hat{z}$ increased from 100 to 1000, the accuracy of estimated standard error improved and the variance estimates from the combined method was overlapping with the true value.

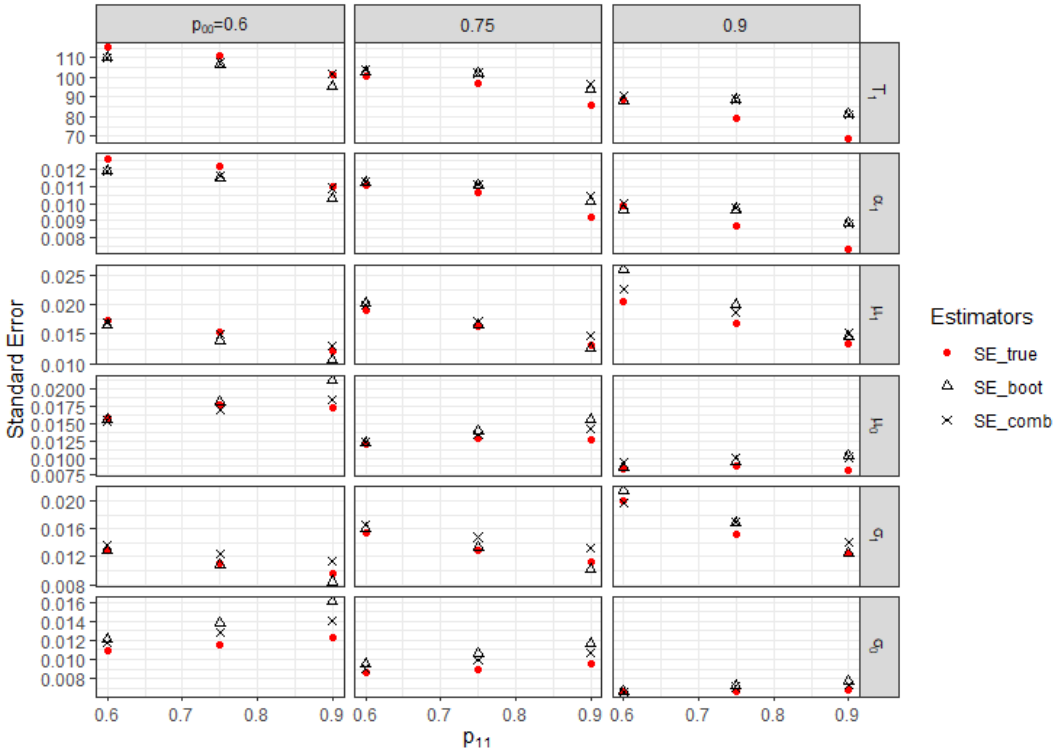


Figure 17: Variance estimation in case 3b.

5 Practical Guidelines

In this part, we provide simple guidelines on how to use the EM method and the combined method in real applications, where data sets only contain the observed classification variable $\hat{z}$ and the continuous variable y. We consider the situation with only two classes.

5.1 Procedures

To begin with, the number of components for each class in the Gaussian mixture model needs to be estimated before applying the methods. The criterion of BIC can be applied with the log-likelihood function as Formula 9 in the context without any audit sample, or as Formula 11 when there is an audit sample.

With $\hat{z}$ and y, the EM method and the combined method can be applied according to Algorithm 2 and Algorithm 3 in Section 2.2. Through it, bias and variance caused by classification errors can be estimated.

5.2 Outliers

The problem of outliers will influence the performance of our methods as shown in case 3a and case 3b (Section 4.3 and Appendix B).

With only observed classes in practice, two ways to find the unusual observations were proposed by McLachlan and Peel (2004, p. 74-75). One way is through a modified likelihood ratio test, where the likelihood ratio statistic λ for each unit is calculated and tested. For unit i, the corresponding λ_i is to test the null hypothesis H_0 that unit i is from the mixture normal model versus the alternative hypothesis H_1 that it is not. To apply the modified likelihood ratio test, the mixture model first has to be estimated on the data set without this observation. So if we want to check a data set of N units for outliers, the model has to be estimated N times. This is clearly a lot of work. A simpler method based on the Mahalanobis distance was also proposed, where for a single continuous variable like in our case, this distance is equivalent to a z score. This method requires much less work, but it does not contain a formal statistical test. Still, it may be useful in practice as a quick way to find outliers.

In order to deal with outliers, unusual data that do not conform to the mixture of Gaussian models need to be checked manually. These outliers might be misclassified. We can then assign them to the correct classes. Some of them are not necessarily misclassified and just do not conform to the assumption of the mixture normal distribution, which influence our estimation of bias and variance. As long as we are sure that there isn't any classification error among these units after this manual check, we can still apply the methods on the rest of the data to get the bias and variance, and include the outliers only in the final statistics (e.g. total turnover).

6 Discussion

In this thesis, we have proposed two new methods, the EM method and the 'combined' method (a combination of the EM method and bootstrapping), for estimating the accuracy of statistics when there are classification errors in the data sets. The performance of the EM method and the combined method were compared with the bootstrap method, which is commonly used nowadays for official statistics. We have tested the performance of the three methods in simulated data sets and also in real applications. Two aspects of the accuracy have been investigated: bias and variance.

The results of our study have shown that, in general, the EM method and the combined method perform better than the bootstrap method. The estimated bias from the EM method and the combined method were closer to the true bias than the bootstrap method. The variance estimation results from the combined method were also more accurate than the bootstrap method.

We have also shown the evidence by comparing the 95% confidence intervals of bias estimates in the simulation study, that even in a practical situation where there is only one single set of $\hat{z}$, the EM and the combined method lead to better bias estimates than the bootstrap method.

The EM method directly uses the model parameters obtained from the EM algorithm to calculate maximum-likelihood statistics, through which the bias is estimated. Apart from its accurate bias estimation, the EM method can estimate the bias easier and saves lots of computation time compared to the combined method. However, it is not able to estimate the variance of statistics caused by classification errors.

The combined method applies the estimated model parameters from the EM algorithm and combines it with the bootstrap method to generate multiple sets of true classes conditional on the continuous variable y. The accuracy of its bias estimation is almost equal to that of the EM method, even though it requires more computation. Compared to the original bootstrap method, its performance on bias and variance estimation is much better.

The EM method and the combined method also have the advantage of flexibility. The use of a Gaussian mixture model ensures that these two methods can accommodate various distributions for the continuous variable (McLachlan & Peel, 2004). Besides, when all the model parameters have been estimated from the EM algorithm, various maximum-likelihood estimators can be evaluated by our methods. Moreover, the probabilities of classification errors can also be estimated from the EM algorithm (details in Section 2.3) . In previous studies these could only be obtained from audit samples (Van Delden et al., 2016). If an audit sample is available, it can be incorporated into the EM algorithm (details in Appendix A) and this benefits the stability of the methods.

For real applications, the combined method is recommended since both the bias and the variance can be estimated. In a situation with a very large data set, where the variance is usually small and the bias is of most

concern, the EM method which is easier and simpler, may be preferable.

The use of the EM algorithm is not new in the studies concerning classification errors. Sinclair and Hooker (2017) and Kosinski and Flanders (1999) have applied the EM algorithm to improve the performance of their classification model. Greenland (2008) has used it to improve the accuracy of estimators. It's natural to apply the EM algorithm once we build statistical models depending on the unobserved true classes. With the purpose of assessing the effect of classification errors, we have broadened the usage of the EM algorithm.

In official statistics, using model-based approaches is not common. The main reason is that NSIs (National Statistical Institutes) tend to avoid model-based estimators, particularly if assumptions of the model are not verifiable (Brakel & Bethlehem, 2008). In our study, we assume the data for each class conform to a Gaussian mixture model. For the purpose of quality estimation, we only use the Gaussian mixture model to estimate the accuracy of estimators, such as total sum and proportion, which does not contradict the traditions of the NSIs. Once the bias and variance are obtained, we either accept the current accuracy and publish the statistics, or apply further data editing to improve the accuracy of the estimator based on the estimated misclassification probabilities of each class.

Application of our methods has a number of limitations. For the combined method, the computation load is relatively high, especially for a large data set. Furthermore, the use of the EM algorithm in our methods has the risk of finding a local maximum of the likelihood function which will influence the accuracy of estimates of bias and variance.

We used only one set of starting values in our study since the starting values of the EM algorithm were obtained from the audit sample which was sufficient to obtain accurate bias and variance estimates. Concerning the potential problem of local maxima, usually, a better strategy is to apply multiple random starting values which is what we did in experiment 6 in Appendix B. When the empirical distribution gets more complicated, it is likely that the parameters of the model achieve a local maximum in the EM algorithm. Then multiple starting values would be helpful to avoid this problem.

Outliers are another potential risk to influence the performance of our methods, as the results for case 3a in the case study show. In case 3a, the performance of the EM method and the combined method was not as good as expected in the beginning. After removing atypical observations in the experiment 8 in Appendix B, the performance improved greatly. The EM algorithm in our study is sensitive to violations of the underlying assumption that the data for each class are a mixture of normals, and may be markedly influenced by one or a few atypical observations.

The number of iterations for each method will influence the accuracy of their estimates. In our study, we used a fixed number of iterations for the experiments. Based on the results, we judged that we had performed enough iterations. In practice though, the number of iterations should be adjusted according to properties of the data sets, such as size, distribution of each class, distance between the two classes, etc. Normally, more

iterations are needed for a smaller population size. Section 5 has provided more guidelines when applying our methods in real applications.

In a future study, the assumptions for the probabilities of classification errors could be relaxed. In our study, the probabilities of making classification errors, p_{11} and p_{00}, were assumed independent of the continuous variable. However, in real situations, this assumption does not always hold. For instance, in the case of turnover, the error probabilities for very large as well as for small enterprises are usually low. For the top enterprises, their influence on the final statistics is significant therefore their recordings are often checked manually, decreasing the probability of wrong classifications. The small enterprises tend to have more uniform economic activities which also reduce their misclassification probabilities. For medium-size enterprises, their probabilities of being misclassified will be larger than for other companies, since fewer manual checks are done for these enterprises and their businesses are at the same time diverse. Auxiliary variables can be added as predictors of the probabilities, where the method of regression models can be applied (Oosterveen, 2020; Shaw et al., 2020).

Furthermore, instead of only two classes as in our study, future research can be expanded to more classes. With more classes, the transition matrix describing the probabilities of misclassification will no long be 2×2, depending on how many classes are involved (Burger et al., 2015). When there are n classes in the population, the probabilities of misclassification among these classes will be described by an $n \times n$ matrix. The number of parameters estimated from the EM algorithm will also correspondingly increase.

Another possible extension is to use multiple numerical target variables, such as height and weight of patients in medical records. Then there will be more than one target variable in the general model. A multivariate Gaussian mixture model can be a suitable model for this case (McLachlan & Peel, 2004).

References

Brakel, J., & Bethlehem, J. (2008, 01). *Model-based estimation for official statistics* (Tech. Rep.). Retrieved from https://dare.uva.nl/search?identifier=e2ee6956-daf7-460d-8ff6-1d52619a5617

Burger, J., Van Delden, A., & Scholtus, S. (2015, 9). Sensitivity of mixed-source statistics to classification errors. *Journal of Official Statistics*, *31*(3), 489 – 506.

Dempster, A. P., Laird, N. M., & 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.

Drton, M., Plummer, M., et al. (2017). A bayesian information criterion for singular models. *Journal of the Royal Statistical Society*, *79*(2), 323 – 380.

Edwards, J. K., Bakoyannis, G., Yiannoutsos, C. T., Mburu, M. W., & Cole, S. R. (2019). Nonparametric estimation of the cumulative incidence function under outcome misclassification using external validation data. *Statistics in Medicine*, *38*(29), 5512 – 5527.

Edwards, J. K., Cole, S. R., & Fox, M. P. (2020, 01). Flexibly Accounting for Exposure Misclassification With External Validation Data. *American Journal of Epidemiology*, *189*(8), 850 – 860. Retrieved from https://doi.org/10.1093/aje/kwaa011 doi: 10.1093/aje/kwaa011

Eurostat. (2008). *Nace rev.2 statistical classification of economic activities in the european community.* Retrieved from https://ec.europa.eu/eurostat/ramon/nomenclatures/index.cfm?TargetUrl=LST_NOM_DTL&StrNom=NACE_REV2

Gravel, C. A., & Platt, R. W. (2018). Weighted estimation for confounded binary outcomes subject to misclassification. *Statistics in medicine*, *37*(3), 425 – 436.

Greenland, S. (1988). Variance estimation for epidemiologic effect estimates under misclassification. *Statistics in medicine*, *7*(7), 745 – 757.

Greenland, S. (2008). Maximum-likelihood and closed-form estimators of epidemiologic measures under misclassification. *Journal of Statistical Planning and Inference*, *138*(2), 528 – 538.

Herzig, K., Just, S., & Zeller, A. (2013). It's not a bug, it's a feature: how misclassification impacts bug prediction. In *2013 35th international conference on software engineering (icse)* (pp. 392 – 401).

Kosinski, A. S., & Flanders, W. D. (1999). Evaluating the exposure and disease relationship with adjustment for different types of exposure misclassification: a regression approach. *Statistics in Medicine*, *18*(20), 2795 – 2808.

Li, Y. (2020). *Bias correction for classification errors.* (Internship Report)

Little, R. J., & Rubin, D. B. (2002). *Statistical analysis with missing data* (2nd ed., Vol. 793). John Wiley & Sons.

Magnusson, P., Palm, A., Branden, E., & Mörner, S. (2017). Misclassification of hypertrophic cardiomyopathy: validation of diagnostic codes. *Clinical Epidemiology*, *9*, 403.

McLachlan, G. J., & Peel, D. (2004). *Finite mixture models.* John Wiley & Sons.

Meertens, Q., Diks, C., Van den Herik, H., & Takes, F. (2020). A data-driven supply-side approach for estimating cross-border internet purchases within the european union. *Journal of the Royal Statistical*

Society: Series A (Statistics in Society), *183*(1), 61 – 90.

Morais, C. L., Lima, K. M., & Martin, F. L. (2019). Uncertainty estimation and misclassification probability for classification models based on discriminant analysis and support vector machines. *Analytica chimica acta*, *1063*, 40 – 46.

Nab, L., Groenwold, R. H., Van Smeden, M., & Keogh, R. H. (2020). Quantitative bias analysis for a misclassified confounder: A comparison between marginal structural models and conditional models for point treatments. *Epidemiology*, *31*(6), 796 – 805.

Nordbotten, S. (2010). The use of administrative data in official statistics – past, present and future: with special reference to the nordic countries. *Official Statistics: Methodology and Applications in Honour of Daniel Thorburn (Eds. Carlson, Nyquist Villani. Stockholm)*, 205 – 223. Retrieved from `https://officialstatistics.wordpress.com/`

Oosterveen, V. (2020). *Notice the noise: detecting misclassifications in register data* (Unpublished master's thesis). Utrecht University, the Netherlands.

Scholtus, S., & Van Delden, A. (2020, 02). *On the accuracy of estimators based on a binary classifier* (Tech. Rep.). Retrieved from `https://www.cbs.nl/en-gb/background/2020/06/the-accuracy-of-estimators-based-on-a-binary-classifier`

Scholtus, S., Van Delden, A., & Burger, J. (2019, 10). *Evaluating the accuracy of growth rates in the presence of classification errors* (Tech. Rep.). Retrieved from `https://www.cbs.nl/en-gb/background/2019/44/the-accuracy-of-growth-rates-with-classification-errors`

Scrucca, L., Fop, M., Murphy, T. B., & Raftery, A. E. (2016). mclust 5: clustering, classification and density estimation using Gaussian finite mixture models. *The R Journal*, *8*(1), 289 – 317. Retrieved from `https://doi.org/10.32614/RJ-2016-021`

Selén, J. (1986). Adjusting for errors in classification and measurement in the analysis of partly and purely categorical data. *Journal of the American Statistical Association*, *81*(393), 75 – 81.

Shaw, P. A., Gustafson, P., Carroll, R. J., Deffner, V., Dodd, K. W., Keogh, R. H., ... others (2020). Stratos guidance document on measurement error and misclassification of variables in observational epidemiology: Part 2—more complex methods of adjustment and advanced topics. *Statistics in Medicine*, *39*(16), 2232 – 2263. doi: 10.1002/sim.8531

Sinclair, D. G., & Hooker, G. (2017). An expectation maximization algorithm for high-dimensional model selection for the ising model with misclassified states. *arXiv preprint arXiv:1704.05995*. Retrieved from `https://arxiv.org/abs/1704.05995`

Van Delden, A., Scholtus, S., & Burger, J. (2016). Accuracy of mixed-source statistics as affected by classification errors. *Journal of Official Statistics*, *32*(3), 619 – 642. Retrieved from `https://content.sciendo.com/view/journals/jos/32/3/article-p619.xml` doi: https://doi.org/10.1515/jos-2016-0032

Zhang, L. C. (2011). *A unit-error theory for register-based household statistics* (Tech. Rep.). Retrieved from `https://www.ssb.no/a/publikasjoner/pdf/DP/dp598.pdf`

A EM algorithm with audit sample

A.1 E step and M step

An audit sample of size n is introduced with its z known. If an audit sample is available, the true classes observed in the audit sample can be used to improve the efficiency and accuracy of the EM algorithm. The audit sample can also be used to choose good starting values for the EM algorithm. Let Ω represent the index of the whole population, Ω_w be the part with the audit sample and Ω_{wo} be the part without the audit sample.

For units within the audit sample, their z are known but m unknown, so their m should be replaced by the corresponding expectation during the E step. For units without the audit sample, both z and m need to be replaced by their expectation as described in Section 2.3.2.

When $i \in \Omega_w$, the expectation of m is:

$$\begin{aligned}
E\left(\mathbf{1}_{(m_i=j)}|z=1,\hat{z}=\hat{z}_i,y=y_i\right) =& f_\theta\left(m=j|z=1,\hat{z}=\hat{z}_i,y=y_i\right)\\
=& \frac{f_\theta\left(m=j,z=1,\hat{z}=\hat{z}_i,y=y_i\right)}{\sum_{j=1}^{n_1} f_\theta\left(m=j,z=1,\hat{z}=\hat{z}_i,y=y_i\right)}\\
=& \frac{\omega_{1ji}}{\sum_{j=1}^{n_1}\left(\omega_{1ji}\right)} \triangleq Q_{1ji}\,,\\
E\left(\mathbf{1}_{(m_i=k)}|z=0,\hat{z}=\hat{z}_i,y=y_i\right) =& f_\theta\left(m=k|z=0,\hat{z}=\hat{z}_i,y=y_i\right)\\
=& \frac{f_\theta\left(m=k,z=0,\hat{z}=\hat{z}_i,y=y_i\right)}{\sum_{k=1}^{n_0} f_\theta\left(m=k,z=0,\hat{z}=\hat{z}_i,y=y_i\right)}\\
=& \frac{\omega_{0ki}}{\sum_{k=1}^{n_0}\left(\omega_{0ki}\right)} \triangleq Q_{0ki}\,.
\end{aligned} \tag{10}$$

Therefore, parameters in EM algorithm should be updated during the M step as follows,

$$\begin{aligned}
\alpha_1 =& \frac{\sum\limits_{i\in\Omega_w} z_i + \sum\limits_{i\in\Omega_{wo}}\left(\sum_{j=1}^{n_1} A_{1ji}\right)}{N}\,,\\
p_{11} =& \frac{\sum\limits_{i\in\Omega_w} z_i\hat{z}_i + \sum\limits_{i\in\Omega_{wo}}\left(\sum_{j=1}^{n_1} A_{1ji}\right)\hat{z}_i}{\sum\limits_{i\in\Omega_w} z_i + \sum\limits_{i\in\Omega_{wo}}\left(\sum_{j=1}^{n_1} A_{1ji}\right)}\,,\\
p_{00} =& \frac{\sum\limits_{i\in\Omega_w}\left(1-z_i\right)\left(1-\hat{z}_i\right) + \sum\limits_{i\in\Omega_{wo}}\left(\sum_{k=1}^{n_0} A_{0ki}\right)\left(1-\hat{z}_i\right)}{\sum\limits_{i\in\Omega_w}\left(1-z_i\right) + \sum\limits_{i\in\Omega_{wo}}\left(\sum_{k=1}^{n_0} A_{0ki}\right)}\,,\\
\pi_{1j} =& \frac{\sum\limits_{i\in\Omega_w} z_i Q_{1ji} + \sum\limits_{i\in\Omega_{wo}} A_{1ji}}{\sum\limits_{i\in\Omega_w} z_i + \sum\limits_{i\in\Omega_{wo}}\left(\sum_{j=1}^{n_1} A_{1ji}\right)}\,,\\
\pi_{0k} =& \frac{\sum\limits_{i\in\Omega_w}\left(1-z_i\right)Q_{0ki} + \sum\limits_{i\in\Omega_{wo}} A_{0ki}}{\sum\limits_{i\in\Omega_w}\left(1-z_i\right) + \sum\limits_{i\in\Omega_{wo}}\left(\sum_{k=1}^{n_0} A_{0ki}\right)}\,,
\end{aligned}$$

$$
\begin{aligned}
\mu_{1j} =& \frac{\sum\limits_{i\in\Omega_w} z_i Q_{1ji} y_i + \sum\limits_{i\in\Omega_{wo}} A_{1ji} y_i}{\sum\limits_{i\in\Omega_w} z_i Q_{1ji} + \sum\limits_{i\in\Omega_{wo}} A_{1ji}}\,, \\
\mu_{0k} =& \frac{\sum\limits_{i\in\Omega_w} (1-z_i)\, Q_{0ki} y_i + \sum\limits_{i\in\Omega_{wo}} A_{0ki} y_i}{\sum\limits_{i\in\Omega_w} (1-z_i)\, Q_{0ki} + \sum\limits_{i\in\Omega_{wo}} A_{0ki}}\,, \\
\sigma_{1j} =& \sqrt{\frac{\sum\limits_{i\in\Omega_w} z_i Q_{1ji} \left(y_i-\mu_{1j}\right)^2 + \sum\limits_{i\in\Omega_{wo}} A_{1ji} \left(y_i-\mu_{1j}\right)^2}{\sum\limits_{i\in\Omega_w} z_i Q_{1ji} + \sum\limits_{i\in\Omega_{wo}} A_{1ji}}}\,, \\
\sigma_{0k} =& \sqrt{\frac{\sum\limits_{i\in\Omega_w} (1-z_i)\, Q_{0ki} \left(y_i-\mu_{0k}\right)^2 + \sum\limits_{i\in\Omega_{wo}} A_{0ki} \left(y_i-\mu_{0k}\right)^2}{\sum\limits_{i\in\Omega_w} (1-z_i)\, Q_{0ki} + \sum\limits_{i\in\Omega_{wo}} A_{0ki}}}\,.
\end{aligned}
$$

A.2 Log-likelihood function for observed data (assume $n_1 \leq n_0$)

$$
\begin{aligned}
l_{obs}\theta &= \sum_{i\in\Omega_w} \log f_\theta\left(\hat{z}=\hat{z}_i, y=y_i, z=z_i\right) + \sum_{i\in\Omega_{wo}} \log f_\theta\left(\hat{z}=\hat{z}_i, y=y_i\right) \\
&= \sum_{i\in\Omega_w} \log\left(\sum_{m_i} f_\theta\left(\hat{z}=\hat{z}_i, y=y_i, z=z_i, m=m_i\right)\right) \\
&\quad + \sum_{i\in\Omega_{wo}} \log\left(\sum_{(z_i,m_i)} f_\theta\left(\hat{z}=\hat{z}_i, y=y_i, z=z_i, m=m_i\right)\right) \\
&= \sum_{i\in\Omega_w} \log\left(\sum_{j=1}^{n_1} \left(\omega_{1ji}\right)^{z_i} \left(\omega_{0ji}\right)^{1-z_i} + \sum_{k=n_1+1}^{n_0} \left(\omega_{0ki}\right)^{1-z_i}\right) \\
&\quad + \sum_{i\in\Omega_{wo}} \log\left(\sum_{j=1}^{n_1} \omega_{1ji} + \sum_{k=1}^{n_0} \omega_{0ki}\right)\,.
\end{aligned} \tag{11}
$$

A.3 Starting Values

According to the definition in Formula 10, Q_{1ji} and Q_{0ki} refer to the identification to which component of its (known) class unit i belongs. For example, Q_{1ji} is the expectation of unit i belonging to component j of class 1, given that we know unit i belongs to class 1.

We applied k-means clustering to partition the audit sample into a certain number of clusters (the number is n_1 for class 1 and n_0 for class 0). Through it, the starting values $Q_{1ji}^{(0)}$ and $Q_{0ki}^{(0)}$ are computed. In the audit sample, if unit i in class 1 is divided into component j, then $Q_{1ji}^{(0)}$ equals to 1, otherwise $Q_{1ji}^{(0)}$ is 0. The same holds for class 0. In the audit sample, if unit i in class 0 is divided into component k, then $Q_{0ki}^{(0)}$ equals to 1, otherwise $Q_{0ki}^{(0)} = 0$.

Once $Q_{0ji}^{(0)}$ and $Q_{0ki}^{(0)}$ are obtained from k-means clustering, all the starting values can be set as:

$$
\begin{aligned}
\alpha_1^{(0)} =& \frac{\sum\limits_{i\in\Omega_w} z_i}{n}\,, \\
p_{11}^{(0)} =& \frac{\sum\limits_{i\in\Omega_w} z_i\hat{z}_i}{\sum\limits_{i\in\Omega_w} z_i}\,, \\
p_{00}^{(0)} =& \frac{\sum\limits_{i\in\Omega_w} (1-z_i)\,(1-\hat{z}_i)}{\sum\limits_{i\in\Omega_w} (1-z_i)}\,, \\
\pi_{1j}^{(0)} =& \frac{\sum\limits_{i\in\Omega_w} z_i Q_{1ji}^{(0)}}{\sum\limits_{i\in\Omega_w} z_i}\,, \\
\pi_{0k}^{(0)} =& \frac{\sum\limits_{i\in\Omega_w} (1-z_i)\, Q_{0ki}^{(0)}}{\sum\limits_{i\in\Omega_w} (1-z_i)}\,, \\
\mu_{1j}^{(0)} =& \frac{\sum\limits_{i\in\Omega_w} z_i Q_{1ji}^{(0)} y_i}{\sum\limits_{i\in\Omega_w} z_i Q_{1ji}^{(0)}}\,, \\
\mu_{0k}^{(0)} =& \frac{\sum\limits_{i\in\Omega_w} (1-z_i)\, Q_{0ki}^{(0)} y_i}{\sum\limits_{i\in\Omega_w} (1-z_i)\, Q_{0ki}^{(0)}}\,, \\
\sigma_{1j}^{(0)} =& \sqrt{\frac{\sum\limits_{i\in\Omega_w} z_i Q_{1ji}^{(0)} \left(y_i-\mu_{1j}^{(0)}\right)^2}{\sum\limits_{i\in\Omega_w} z_i Q_{1ji}^{(0)}}}\,, \\
\sigma_{0k}^{(0)} =& \sqrt{\frac{\sum\limits_{i\in\Omega_w} (1-z_i)\, Q_{0ki}^{(0)} \left(y_i-\mu_{0k}^{(0)}\right)^2}{\sum\limits_{i\in\Omega_w} (1-z_i)\, Q_{0ki}^{(0)}}}\,.
\end{aligned}
$$

B Experiments

In order to figure out what factors have influenced the performance of the EM method and the combined method in case 3a and 3b in Section 4, some experiments have been conducted. We only explored the reasons behind case 3a since case 3b has similar results as case 3a.

Before the experiments are introduced, we should first to analyse the conditions of case 3a. First, compared to the situations in the simulation study, the distributions of the two classes are closer in the case study, especially for the case 3a and 3b where the distance between the two classes is the closest for all the cases. Besides, the population size of case 3a is the smallest among all the other cases in the case study.

We began with experiments where data sets were simulated and then to experiments where the data set of case 3a was used.

B.1 Simulating data sets

To create a similar situation as in case 3a, we simulated data sets where the distributions of the two classes were closer and the population size was smaller (Table 5). In the first experiment, the setting was adjusted according to the simulation study. The settings of components for the second and the third experiment were the same as the model parameters estimated from the EM algorithm in case 3a. Compared to the second experiment, the size of the third experiment was only half of it and was also smaller than the population size of case 3a.

The proportion of class 1 was fixed to $\alpha_1 = 0.5$. The probabilities of misclassification for each class, p_{11} and p_{00}, were chosen from 0.6, 0.75 and 0.9.

Table 5: Experiments by simulating similar data set

No.	Size	Components in Class 1				Components in Class 0			
		Number	Proportion	Mean	Standard deviation	Number	Proportion	Mean	Standard deviation
1*	2000	2	(0.5,0.5)	(2,4)	(1,2)	1	1	5	3
2	2000	2	(0.46,0.54)	(4.76,5.33)	(0.22,0.74)	2	(0.36,0.64)	(5.77,5.32)	(0.35,0.27)
3	1000	2	(0.46,0.54)	(4.76,5.33)	(0.22,0.74)	2	(0.36,0.64)	(5.77,5.32)	(0.35,0.27)

* Experiment 1a has the same condition as experiment 1. Only the number of simulations in Experiment 1a increases from 100 to 1000.

The results of these experiments by simulating data sets showed that the performances of the EM and the combined method were not influenced by more difficult conditions in general, where the distributions of the two classes were closer or the population size was smaller (Figure 18, 22 and 24). As for the bias estimation,

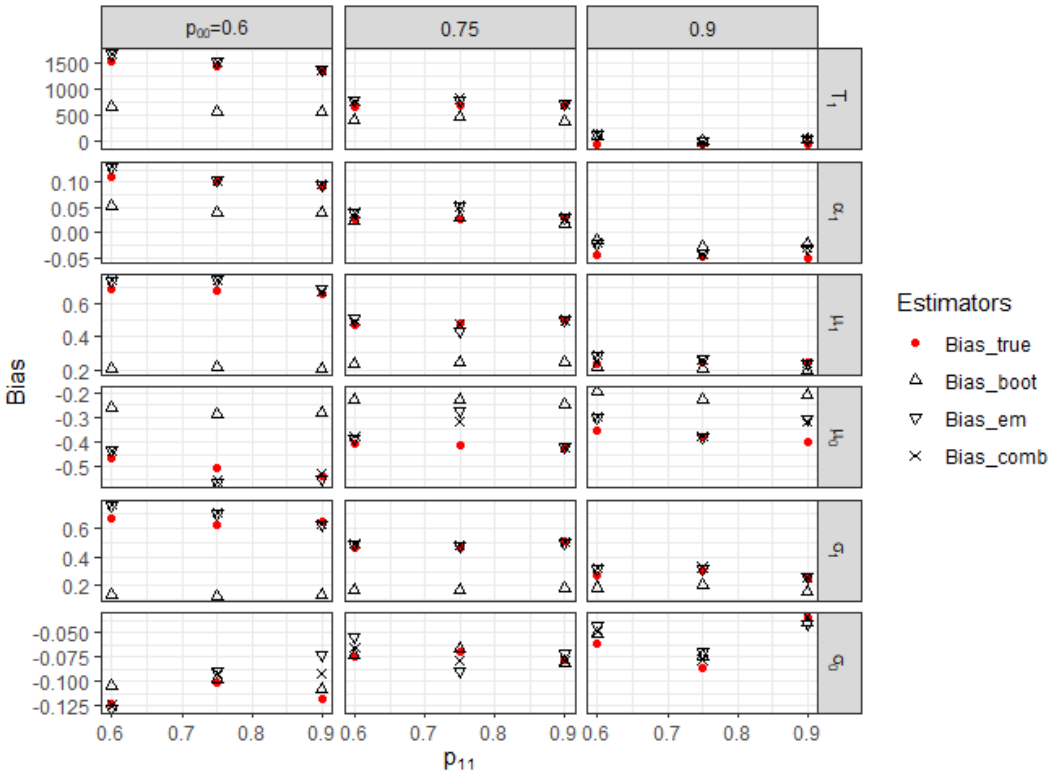


Figure 18: Bias estimation in experiment 1.

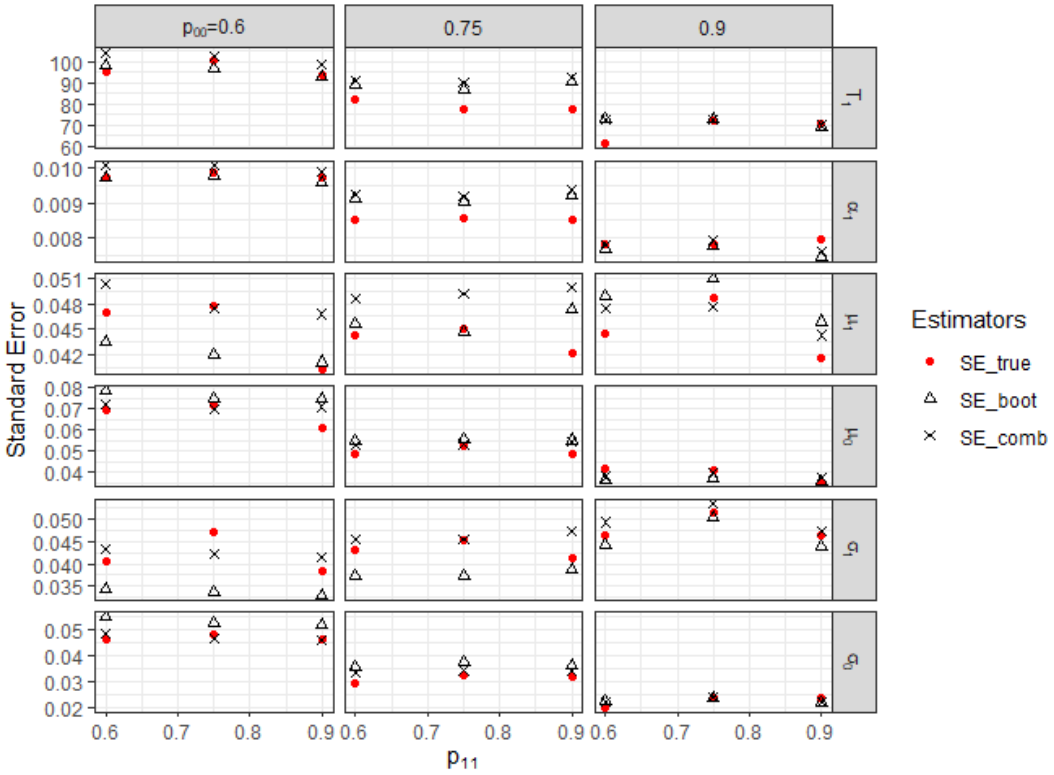


Figure 19: Variance estimation in experiment 1.

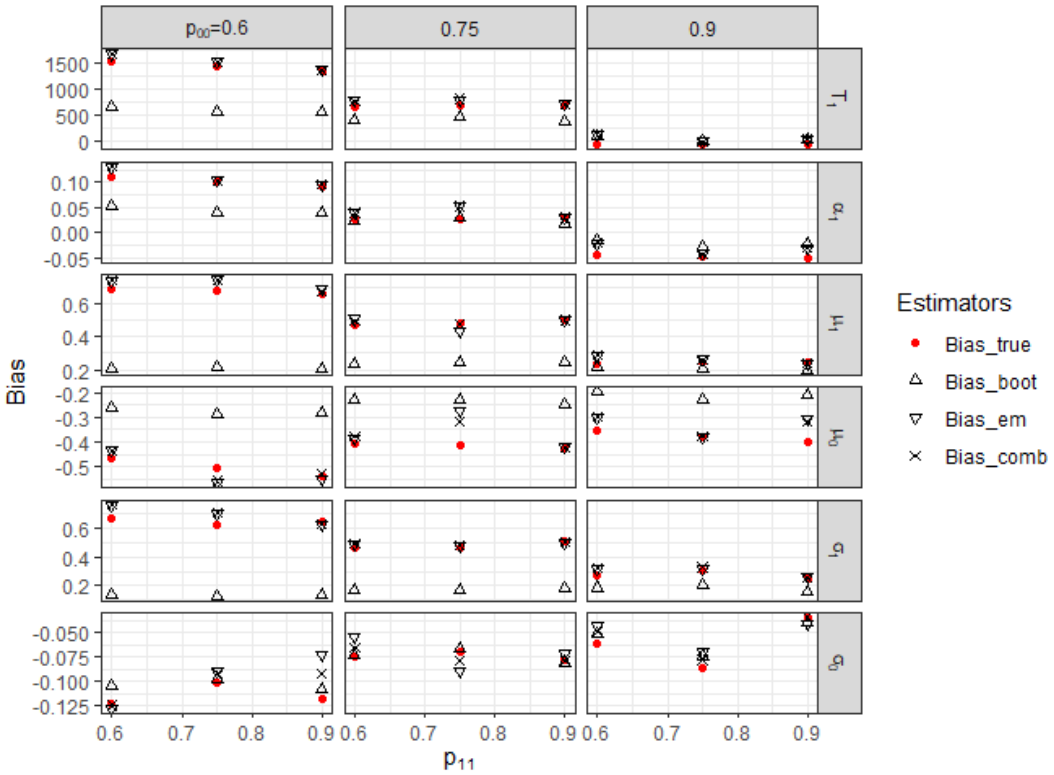


Figure 20: Bias estimation in experiment 1a.

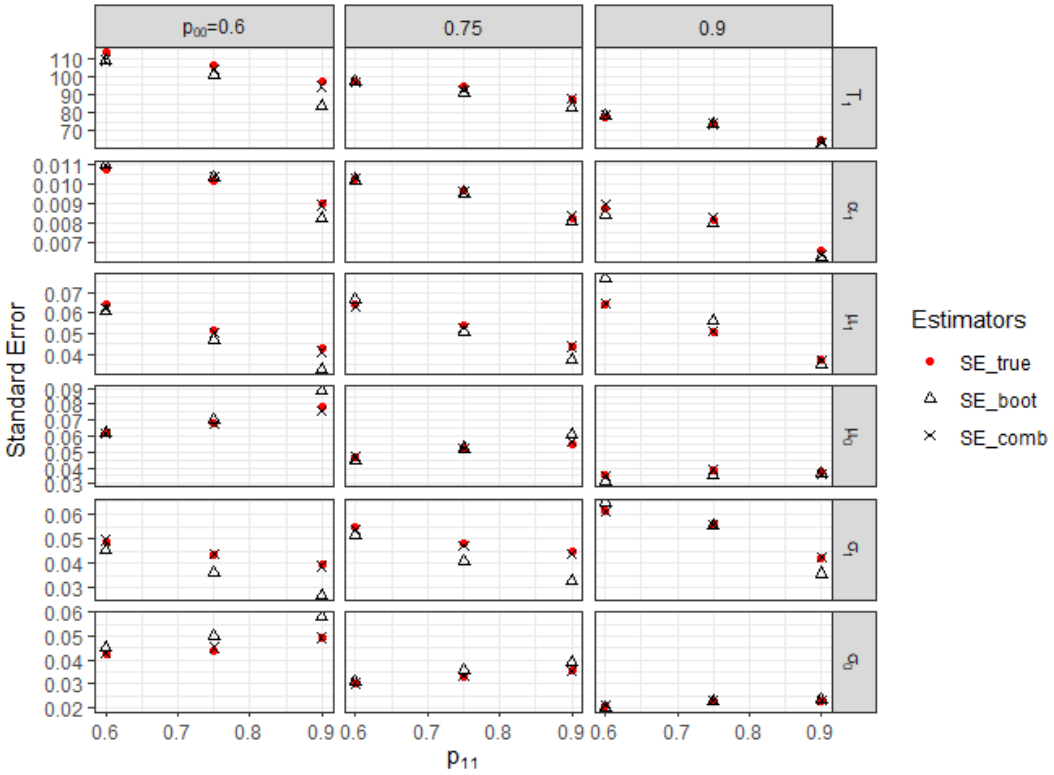


Figure 21: Variance estimation in experiment 1a.

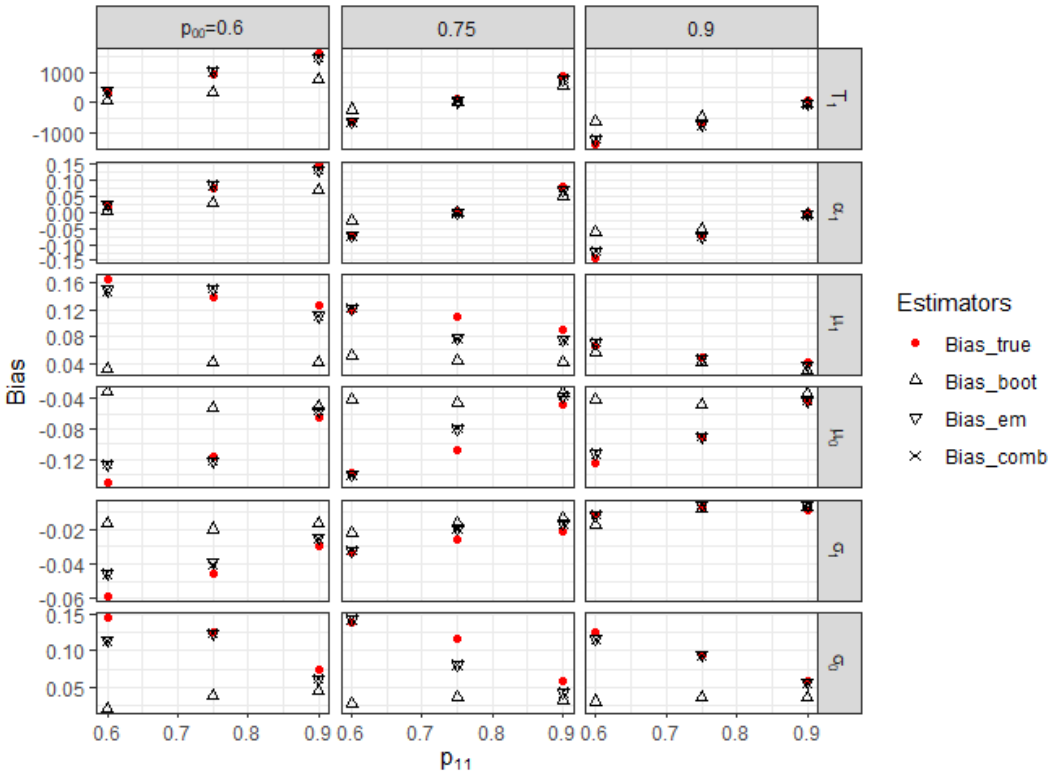


Figure 22: Bias estimation in experiment 2.

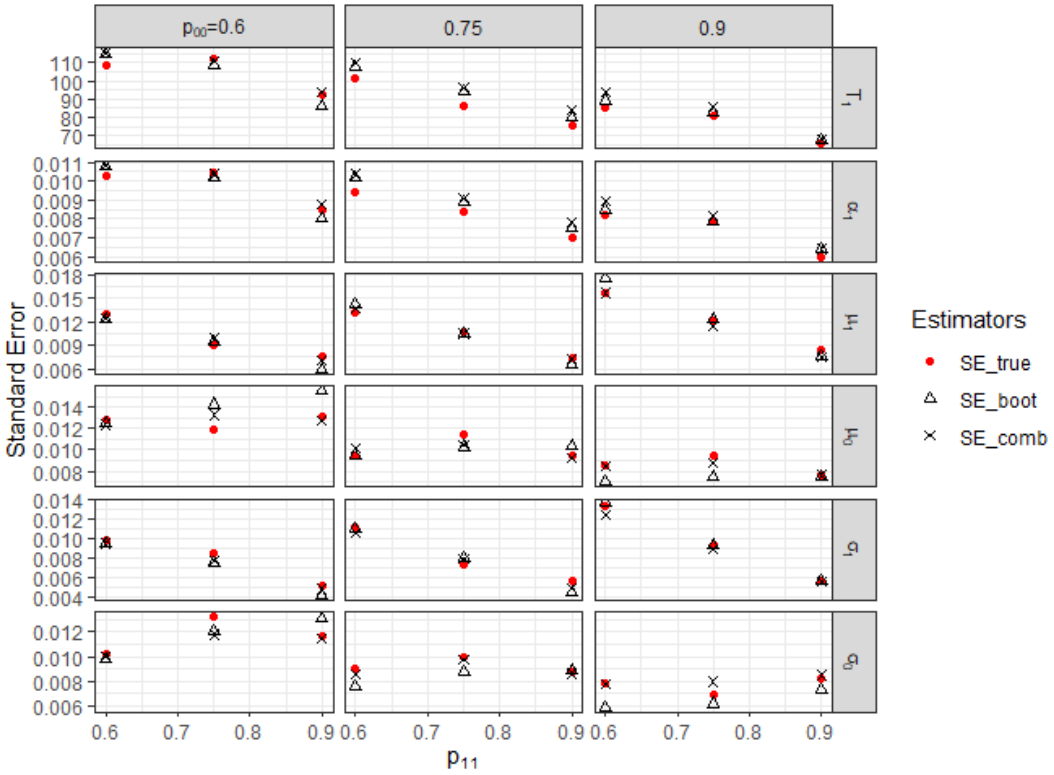


Figure 23: Variance estimation in experiment 2.

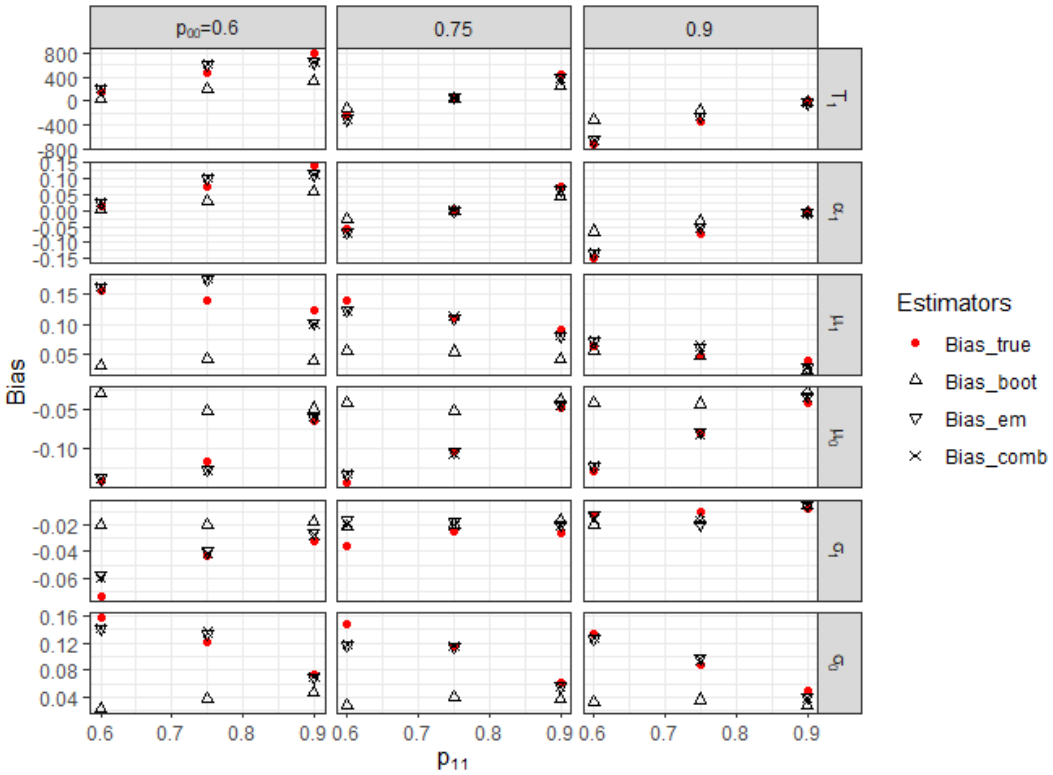


Figure 24: Bias estimation in experiment 3.

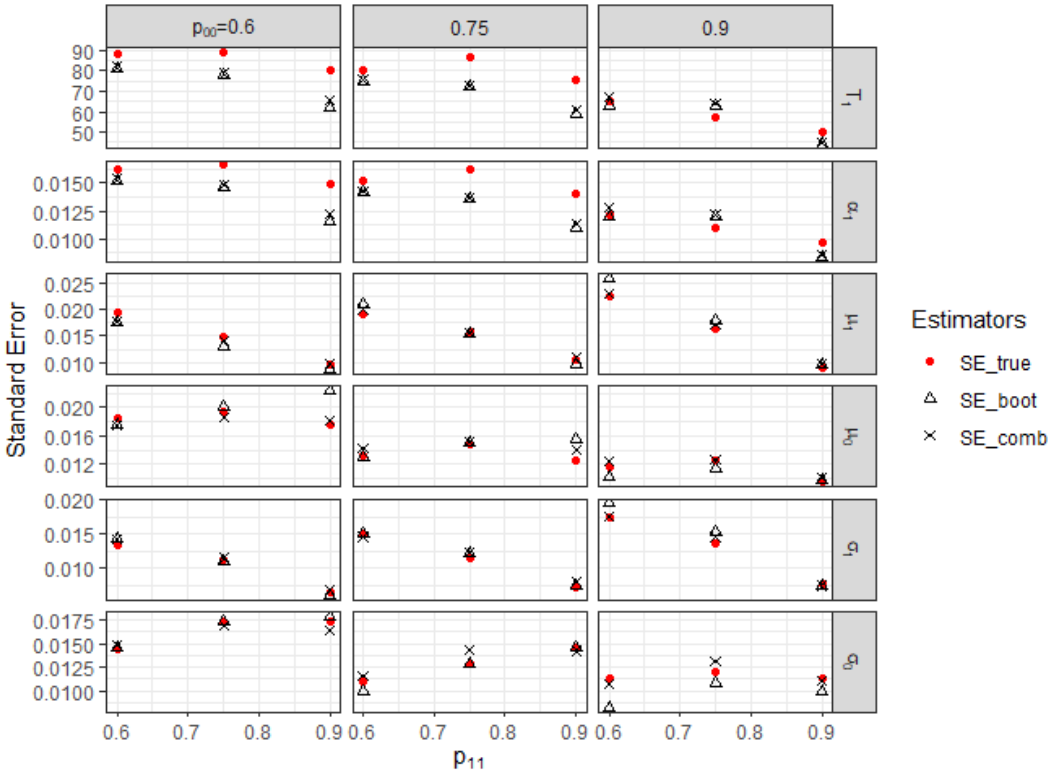


Figure 25: Variance estimation in experiment 3.

the bias estimates from the EM and the combined method were closer to the true bias compared to those from the bootstrap method.

The variance estimates (Figure 19, 23 and 25) from the combined method were also more accurate than those estimated from the bootstrap method. We have also noticed that the variance estimation results in experiment 1 to 3 were not that good. Although the estimates from the bootstrap method and the combined method were close, they still had some distance from the estimated true bias. It can be that the estimated true biases were not that accurate from only 100 simulations. Therefore, we conducted experiment 1a, where the settings were exactly the same as in experiment 1 and only the number of simulations increased from 100 to 1000. The results (Figure 20 and 21) showed that after improving the accuracy of the estimated true bias, the performance of variance estimation of the combined method was better.

B.2 Using the same data set

Then we realized that the empirical distribution of case 3a looked quite different from what was simulated from the three experiments. The empirical distribution of case 3a probably does not conform to the Gaussian mixture model, which makes the estimation in case 3a from the methods difficult. Therefore, we conducted five more experiments by using the original data set of case 3a (Table 6).

Table 6: Experiments by using the original data set of case 3a

No.	Strategies
4	Simulate more times to get a more accurate estimation of true bias
5	Apply the whole set of variable z to set starting values
6	Using 5 sets of random starting values and increase the size of audit sample from 5% to 50%
7	Double the population size by sampling with replacement from the empirical distribution of the case 3a
8	Remove data points which are below $Q_1 - 1.5 \times IQR$ or above $Q_3 + 1.5 \times IQR$

In the experiment 4, we wanted to make sure that the true bias estimated from simulations was accurate enough. We increased the number of simulations of $\hat{z}$ from 100 to 1000. The performance of our methods did not have any obvious improvement (Figure 26 and 27).

In order to decrease the difficulty of the estimation from the EM algorithm, the 5th and the 6th experiment were conducted. In the experiment 5, the true classes of all the units in the data set were used to set the starting values (Formulas in Appendix A.3). And in the experiment 6, random starting values and a larger audit sample were applied. Therefore, the strategies in experiment 5 and 6 used more information compared to the formal studies, which should help the EM algorithm to achieve the global maximum. However, the results from these two experiments did not show any improvement (Figure 28 and 30).

Another possibility would be the population size of case 3a, which was the smallest among all the cases. With a data set of smaller size, it is more likely that the performances of our methods would be influenced by noise, since the EM algorithm would get into the problem of a local minimum. In experiment 7, the kernel density of the continuous variable y for each class was estimated and units were sampled with replacement from the kernel density to get a data set of double size. From the results, the performance of our method did not show any improvement as well (Figure 32).

So apparently, finding the global maximum is not the problem for case 3a.

As should be done before fitting models to data, outlier detection was required, which generated our last but most important experiment 8. In this experiment, we removed the data points whose y values were below $Q_1 - 1.5 \times IQR$ or above $Q_3 + 1.5 \times IQR$, where Q_1 is the first quartile of the data in each (true) class and Q_3 is the third quartile of the data in each class and IQR is the difference between Q_3 and Q_1. Through it, 17 outliers were removed in class 1 $z = 1$ and 20 outliers were removed in class 0 $z = 0$. Then the remaining data for each class was fitted by the Gaussian mixture model and the optimal number of components was selected again by the criterion of BIC, which returned 2 components in class 1 and 1 component in class 0. After that, three methods were applied. It turned out that the performance of the EM and the combined method were highly improved (Figure 34). These two methods outperformed the bootstrap method on the bias estimation and the variance estimates from the combined method were also more accurate than from the bootstrap method.

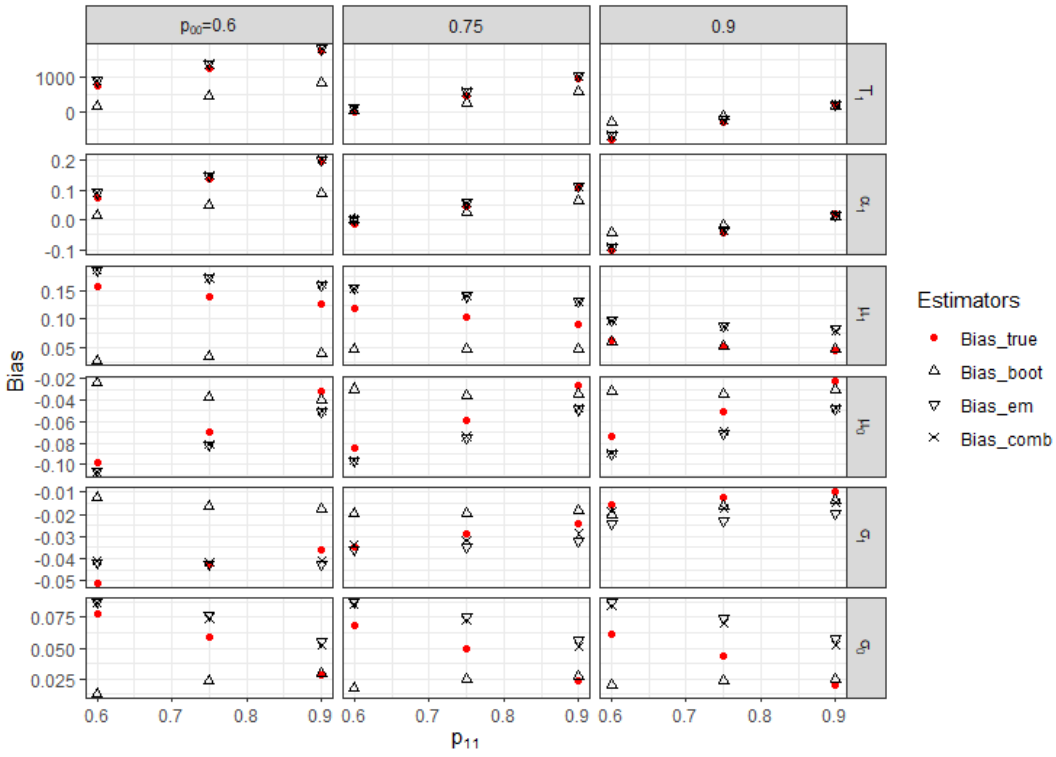


Figure 26: Bias estimation in experiment 4.

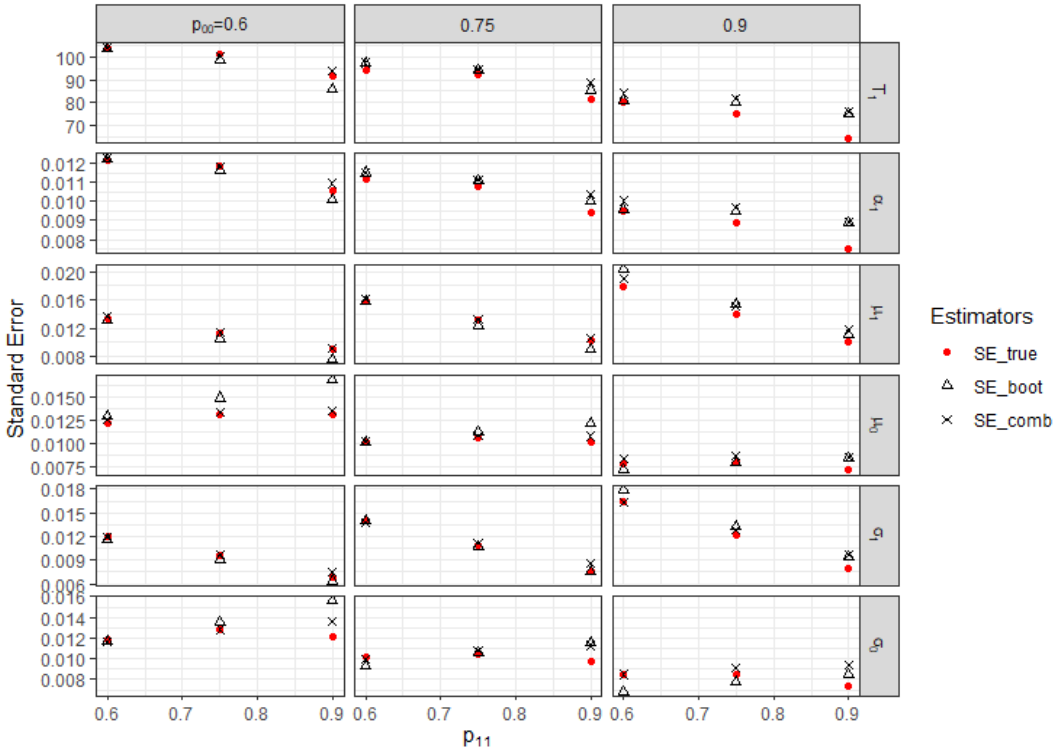


Figure 27: Variance estimation in experiment 4.

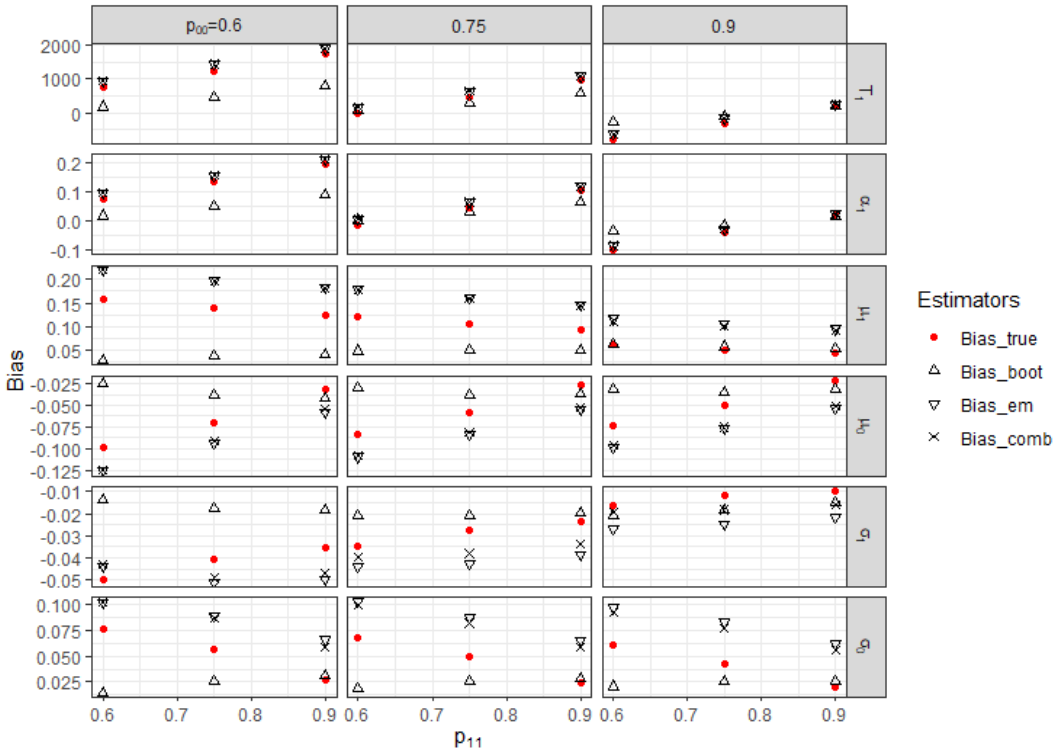


Figure 28: Bias estimation in experiment 5.

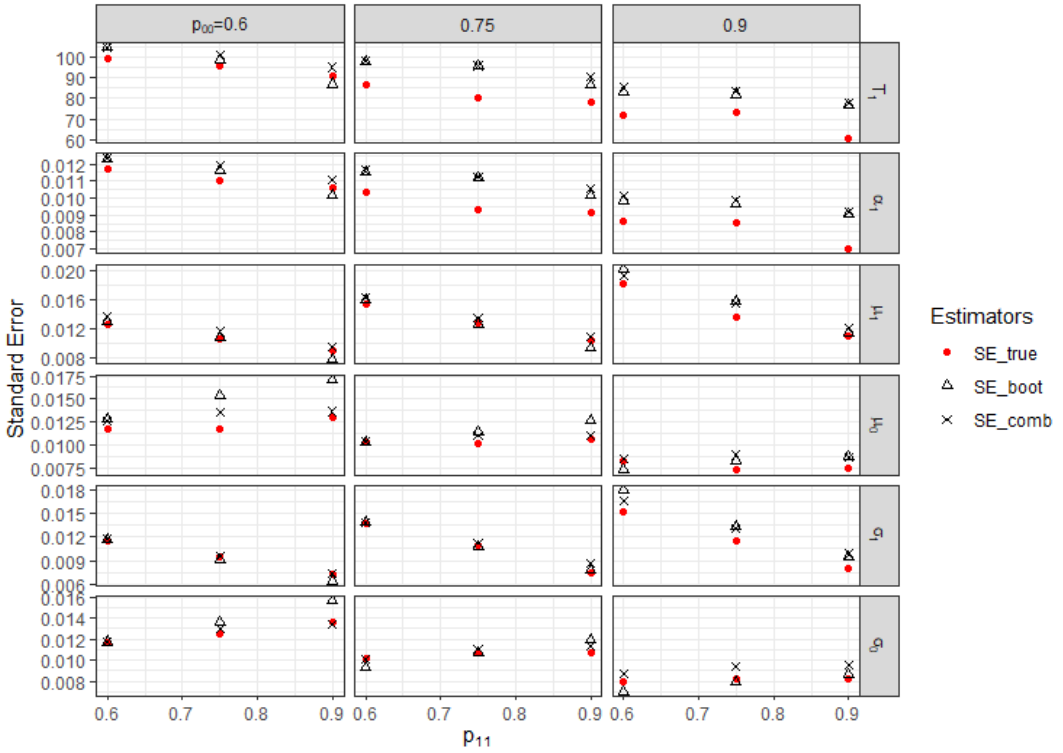


Figure 29: Variance estimation in experiment 5.

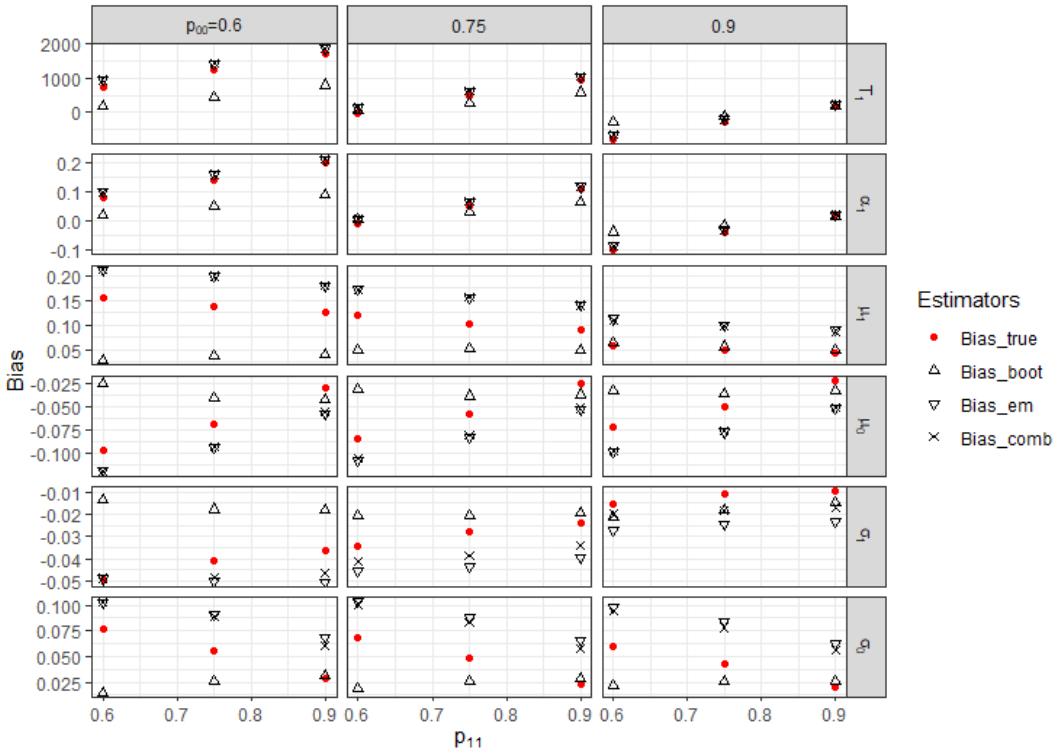


Figure 30: Bias estimation in experiment 6.

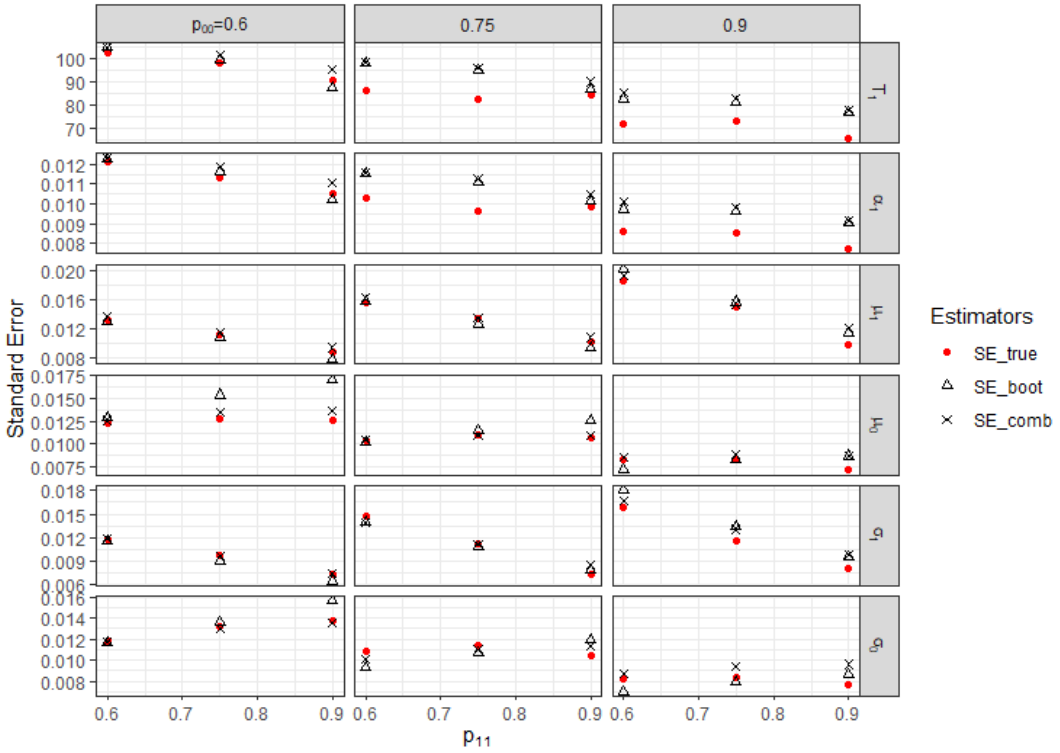


Figure 31: Variance estimation in experiment 6.

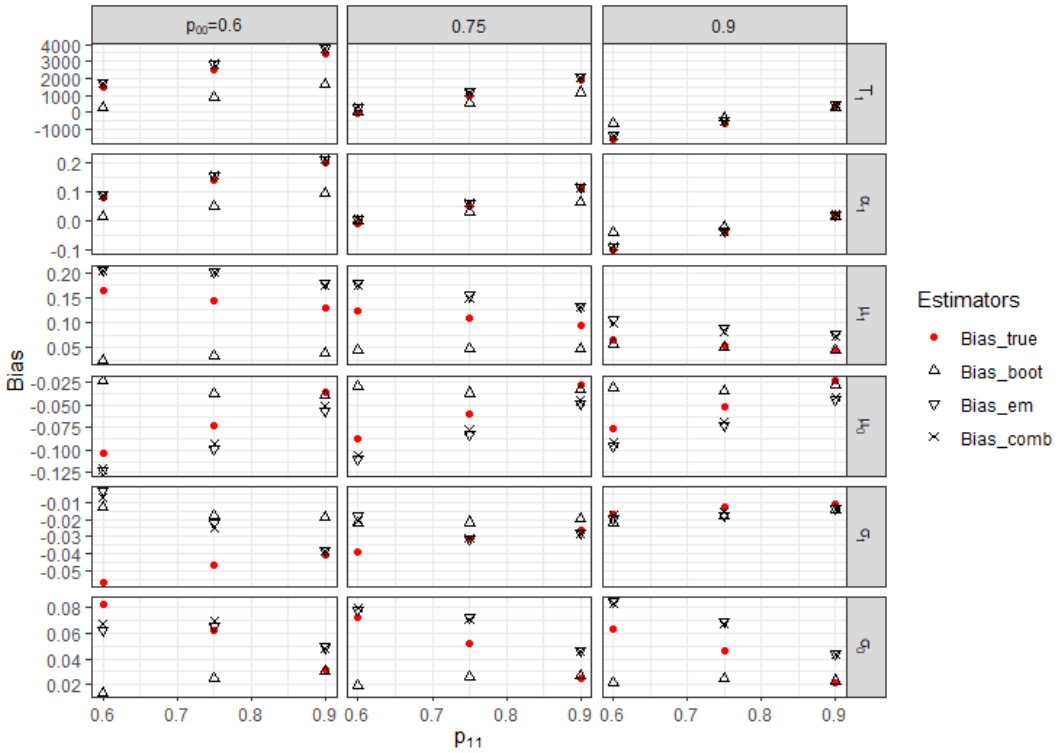


Figure 32: Bias estimation in experiment 7.

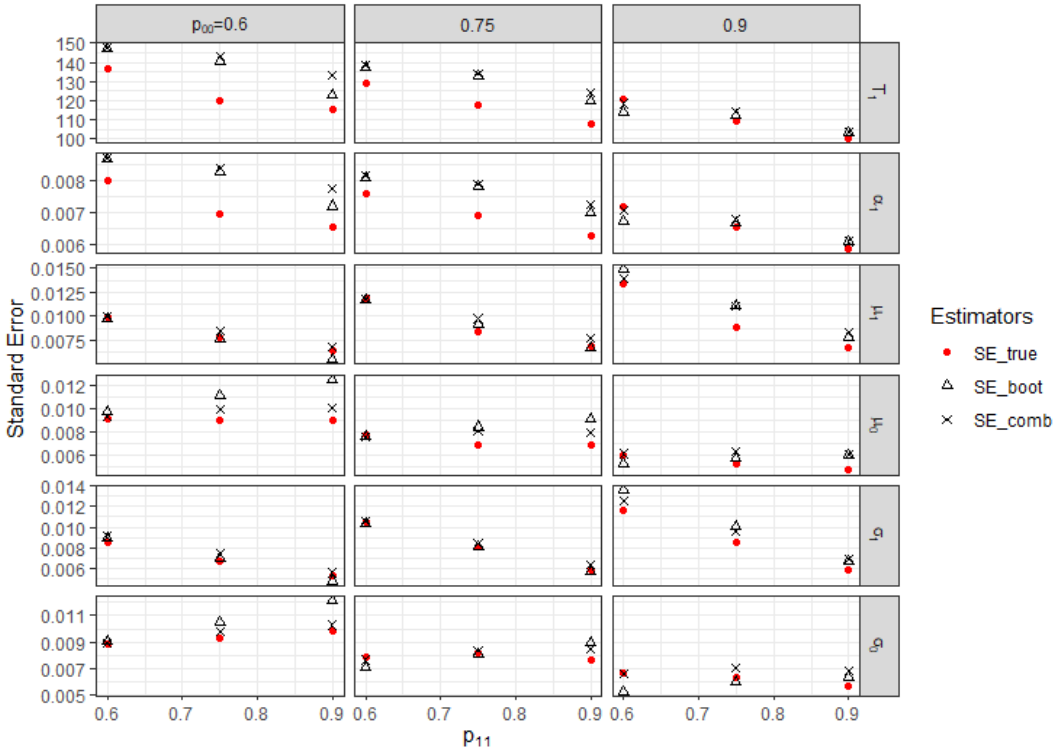


Figure 33: Variance estimation in experiment 7.

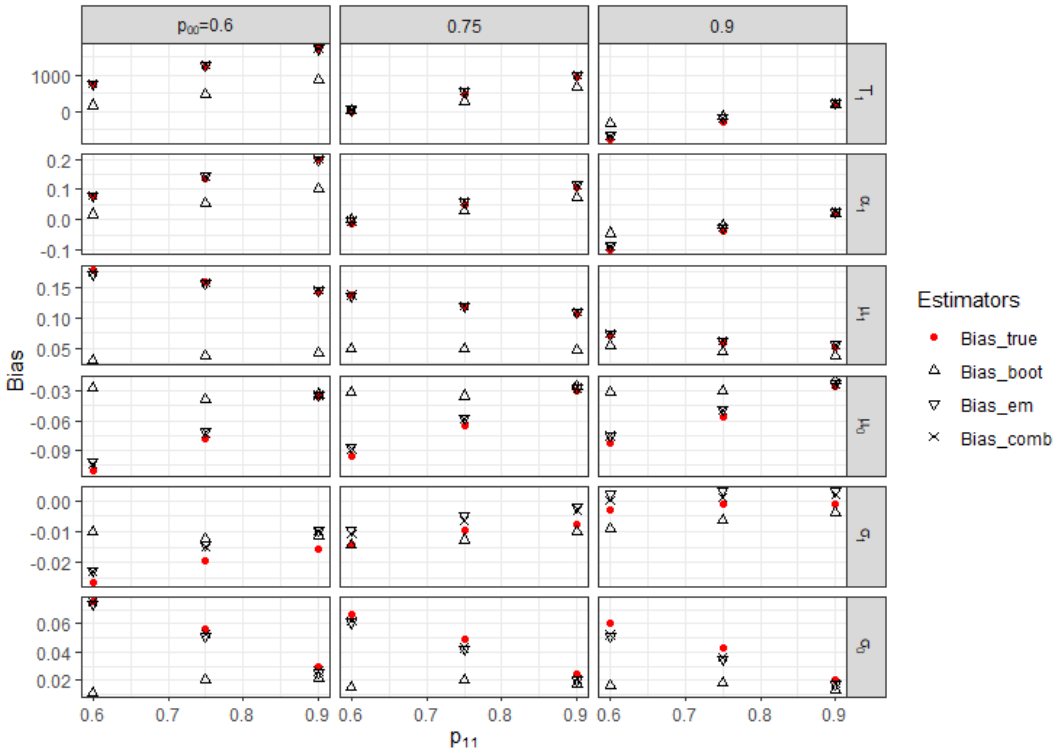


Figure 34: Bias estimation in experiment 8.

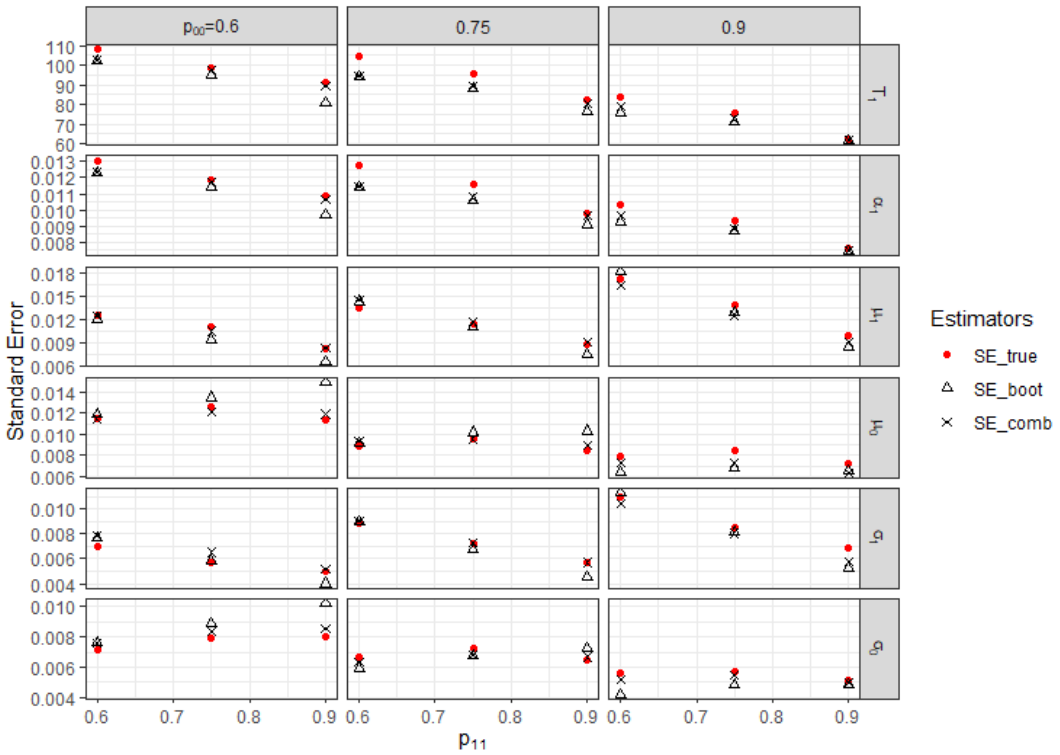


Figure 35: Variance estimation in experiment 8.

C More discussion about using BIC

In the case study, we applied the criterion of BIC to select the optimal number of components for data in each class. However, using BIC in the mixture of normals can be problematic because this model has 'singularities' (parameter vectors with noninvertible Fisher information) (Drton, Plummer, et al., 2017). When the two components in the Gaussian mixture model become very close, with similar mean and variance, using BIC will result in a different number than the actual Bayesian marginal likelihood. Thus we applied a modification of the BIC criterion that has better justification for mixture models, sBIC (Drton et al., 2017), to see whether the selection of optimal number of components in the case study would be same. Table 7 showed the optimal number of components selected by the criterion BIC and sBIC, where the results were close. In the group with code 4932, the optimal number of component selected by sBIC was 4, compared to 2 in BIC, which may be because of the problem of outliers (Appendix B).

Table 7: The Optimal Number of Components Selected by BIC and sBIC

NACE Code	Case No.	Class	Optimal Number	
			BIC	sBIC
56101	Case 1	Class 1	1	1
96022	Case 1	Class 0	2	2
43221	Case 2	Class 1	3	2
74201	Case 2	Class 0	2	3
4932	Case 3a, 3b	Class 1	2	4
5630	Case 3a	Class 0	2	2
8121	Case 3b	Class 0	1	1

In this study, the purpose of applying the Gaussian mixture model is just to describe the distribution of the data. It does not make much difference if the selected number of components is a bit too large.

On the one hand, choosing the optimal number of components is consistent with the principle of having a parsimonious model. With more parameters to estimate, the uncertainty of the model will increases. It is also preferred, if only a part of the data or an audit sample is applied to select the number of components.

On the other hand, using more components to fit the data helps reduce the bias of the model. In the case study, we used the data from the population to select the number of components and the model will not be used to predict other data, so the problem of overfitting does not have a strong influence in our case. As long as the number of components in the Gaussian mixture model is not too big (for example, 10% of the number of units), it will not a big issue.

So all in all, we suspect it does not make a lot of difference in the end whether BIC or sBIC is used, since, while different, for both methods the number of components chosen is always very small compared to the size of the data set. What criterion is better depends on the purpose of studies. Since we only found out that there was a difference at the very end of this project, re-running the experiments with the different number of components will be left for future work.