國立臺灣大學理學院統計與數據科學研究所

碩士論文

Institute of Statistics and Data Science

College of Science

National Taiwan University

Master's Thesis

處理多重實例學習中標籤歧義的貝氏方法

A Bayesian Approach for Addressing Label Ambiguity in Multiple Instance Learning

陳思妤

Szu-Yu Chen

指導教授: 楊鈞澔 博士

Advisor: Chun-Hao Yang, Ph.D.

中華民國 113 年 7 月

July, 2024

doi:10.6342/NTU202401899

致謝

在完成這篇碩士論文的過程中，我得到了許多人的幫助和支持。首先，我要感謝我的指導教授楊鈞澔教授這兩年的細心指導，並且在我研究過程中遇到瓶頸時，總是耐心地給予引領，協助我順利完成碩士論文的研究。另外，我特別感謝我最棒的家人們，一直以來，你們都相信著我、支持我的所有選擇，使我能夠專心致志地追求自己的目標，以及完成每件自己想做的事情。也要謝謝男朋友，在我遇到挫折的時候，你總會耐心的陪伴、給予我鼓勵，讓我有更多的信心面對各種難關。最後，我要感謝求學生涯中遇到的所有師長、同學們以及身旁的好朋友們，你們的陪伴和幫助使我的求學生涯更加豐富及充實。

doi:10.6342/NTU202401899

doi:10.6342/NTU202401899

摘要

多實例學習 (MIL) 是弱監督學習問題，已被應用於許多領域。多實例（MI）數據包括了袋子和實例的概念，其中每個袋子中都包含了一些實例。另外，袋子的資訊是已知的，而實例的資訊是缺失的。Carbonneau et al. (2018) 提到由於多實例數據中存在缺失值，因此標籤歧義 (label ambiguity) 是在 MIL 中的常見問題。在本論文中，我們了解了某些可能造成標籤歧義的來源，並提出了一個新的袋子模型來解決此問題。我們提出的模型具有幾個優勢：除了放寬現有 MIL 方法中常用的嚴格假設，也能提供更多實例與袋子關聯性的資訊，並且可以與於許多不同的實例分類方法一起合併使用，例如羅吉斯回歸。

本文討論的 MIL 模型是使用貝氏的吉布斯採樣進行模型推論。我們在吉布斯採樣過程中使用變量擴展的方法，具體來說是引入了玻利亞伽瑪的潛變量。我們對提出的袋子模型與現有方法（例如Haußmann et al. (2017) 中提出的方法）進行了比較分析，證明了我們方法的有效性。最後，通過模擬以及實際資料的實驗，驗證了我們方法的性能。

關鍵字：多重實例學習、標籤歧義、吉布斯取樣、玻利亞伽瑪擴充

doi:10.6342/NTU202401899

doi:10.6342/NTU202401899

Abstract

Multiple Instance Learning (MIL) is a weakly supervised learning problem, and it has been used in various fields. Multiple Instance (MI) data includes the concepts of bag and instance, where each bag contains several instances. Also, the bag information is observed, while the instance information is missing. Carbonneau et al. (2018) mentions that label ambiguity is a common issue in MIL due to the missing values in the MI data. In this thesis, we investigate the sources of label ambiguity and propose a novel bag model to address this issue. Our proposed model offers several advantages: (i) It relaxes the strict MIL assumption commonly employed in existing MIL methods. (ii) It provides greater insight into the relationship between instances and their corresponding bags. (iii) It can be integrated with various classifiers at the instance level, such as logistic regression.

The MIL models discussed here are inferred by a Gibbs sampling scheme, which is a Bayesian approach. We employ a variable augmentation technique on the Gibbs sampling process, specifically the Pólya-Gamma augmentation. Comparative analysis between our

doi:10.6342/NTU202401899

proposed bag model and existing methods, such as the one presented in Haußmann et al. (2017), demonstrate the effectiveness of our approach. Finally, we validate the performance of our model through various simulations and real data.

Keywords: multiple instance learning, label ambiguity, Gibbs sampling, Pólya-Gamma augmentation

doi:10.6342/NTU202401899

Contents

doi:10.6342/NTU202401899

doi:10.6342/NTU202401899

List of Figures

doi:10.6342/NTU202401899

doi:10.6342/NTU202401899

List of Tables

doi:10.6342/NTU202401899

doi:10.6342/NTU202401899

Chapter 1 Introduction

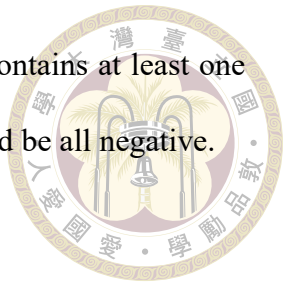
1.1 Multiple Instance Learning

We initially illustrate the problem of multiple instance learning (MIL, Dietterich et al. (1997)) by the following simple example. We imagine that each company member possesses a keychain, which contains many keys for different rooms. Some members' keychains contain the correct key to open a specific room, while others do not. Unfortunately, we do not know which key is the correct one. Our goal is to identify the correct key and predict whether a new key or keychain can get members to enter the specific room.

MIL can be seen as a weakly supervised learning task, where the training data includes missing values. Also, MIL assumes the training data is composed of many bags, with each bag containing multiple instances. Compared to traditional supervised learning, where each instance is directly associated with a response, MIL operates response variables differently. Response variables in MIL are only assigned at the bag level rather than the instance level. This means multiple instances within the same bag share a common bag label (response variable), however, the instance labels are considered missing values. These missing values may cause some challenges in the prediction, such as label ambiguity, which will be discussed in Section 1.4. Additionally, MIL can be employed on binary or multi-task classification problems, but we only emphasize the binary case in

doi:10.6342/NTU202401899

this thesis. Lastly, the MIL assumption defines that a positive bag contains at least one positive instance, on the other hand, instances in a negative bag should be all negative.



1.2 Applications of MIL

In the initial application of MIL, Dietterich et al. (1997) studies the task of drug activity prediction. Each drug molecule (bag) has many distinct shapes, called conformations (instances), according to the different angles of the molecule's rotatable bonds. It is important to note that only a few conformations can bind well to the target protein molecule. Dietterich et al. (1997) also demonstrates that considering MIL results in better prediction performance than ignoring it on their datasets.

MIL is widely used to tackle different tasks across various fields afterward. Some notable applications include the image and text classification problems, as studied by Maron and Ratan (1998), Andrews et al. (2002), Zhang et al. (2007), Zhou et al. (2008), and others. Specifically, Maron and Ratan (1998) focuses on classifying natural scene images, e.g. images of waterfalls. In this scenario, if an image (bag) is classified as the waterfall, it indicates that at least one of its sub-images (instances) contains characteristics of the waterfall. Andrews et al. (2002) provides an example of the text classification task. Its MI data are constructed by separating the whole documents (bags) into multiple smaller passages (instances). In the field of medical diagnosis, MIL is used to predict potential breast or Barrett's cancer patients based on hematoxylin and eosin (H&E) stained tissue microarray images (bags) (Kandemir et al., 2014). Small patches of the image represent instances, which are missing values. Moreover, Popescu and Mahnot (2012) emphasizes the detection of illnesses commonly occurring in elders, such as frailty and dementia. Each

doi:10.6342/NTU202401899

individual represents a bag, with 24 measurements (instances) captured by sensors over 24 hours. Finally, another significant application of MIL is object detection (Ali and Saenko, 2014; Ko et al., 2012; Li and Vasconcelos, 2015). Researchers in this domain are interested in identifying whether specific items, like horses, pedestrians, or even landmines, are within images. The entire picture is treated as a bag, and its sub-regions are considered as instances.

According to the numerous applications of MIL just mentioned, it is obvious that MIL offers an alternative interpretation of the traditional data structures. In MIL, an object (bag) can be interpreted through a collection of feature vectors (instances). This differs from conventional methods, such as supervised learning, which typically assumes that an object is described by just a single feature vector.

1.3 Existing MIL Approaches

Amores (2013) categorizes the existing MIL approaches into three different families, called instance space (IS), bag space (BS), and embedded space (ES). Each family employs a different perspective to build classifiers on multiple instance (MI) data. First, classifiers in IS are constructed on the instance level, and bag labels are determined by the prediction results of the instances. This approach is seen in works, such as Dietterich et al. (1997), Raykar et al. (2008), Chen et al. (2017), Haußmann et al. (2017), and Wang and Pinar (2021). Specifically, the studies in Raykar et al. (2008), Chen et al. (2017), and Haußmann et al. (2017) use logistic regression to establish classifiers for instance labels, while Wang and Pinar (2021) employs probit regression.

On the other hand, BS and ES consider the main classification process on the bag

doi:10.6342/NTU202401899

level, instead of the instance level. BS concentrates on the entire bag when modeling. The distance or similarity among bags usually serves as a guide to determining bag labels. Therefore, there are many distance-based techniques utilized in BS, including k-Nearest Neighbors (kNN) (Wang and Zucker, 2000), Support Vector Machine (SVM) (Andrews et al., 2002; Gärtner et al., 2002), and diverse density (Maron and Lozano-Pérez, 1997). Lastly, ES maps a whole bag, containing multiple instances, into only one feature vector (Chen et al., 2007; Ilse et al., 2018). In other words, MI data is transformed into the data with a traditional structure, which defines each feature vector linked with a response variable. As a result, many conventional supervised learning techniques can be applied to classify bag labels.

1.4 Label Ambiguity in MIL

In a MIL problem, the performance of MIL models depends on how we address the missing information, particularly the absence of instance labels, and how closely our additional assumptions align with the observed datasets. According to Carbonneau et al. (2018), an MIL model can be analyzed from four different perspectives: prediction level, bag composition, data distribution, and label ambiguity.

In this thesis, we specifically focus on label ambiguity, which is closely linked to the missing values, i.e. instance labels. Unobserved instance labels limit our understanding of the true connections between instances and their corresponding bags. Consequently, the missing information in MI data increases the difficulty of addressing the bag labels accurately, thereby leading to label ambiguity. Therefore, understanding the reasons behind label ambiguity in MI datasets is crucial for developing effective MIL models.

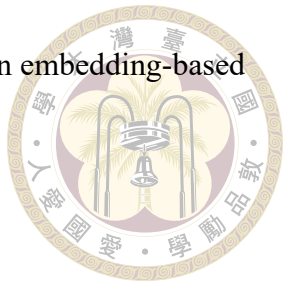
doi:10.6342/NTU202401899

Two possible sources of label ambiguity are proposed. One is label noise, meaning errors exist in the assigned labels. Label noise can be caused by two factors: mislabeling during the process of data collection or inherent complexities in the original data structure. The former is a typical mistake that happens when manual labeling. The latter often occurs in practice, since it is difficult to ensure no positive instance in a negative bag. For example, when classifying a picture of a house that contains a small flower in the sub-image (positive instance), it may not be appropriate to label this entire house picture as a positive bag for the flower, even though this picture includes some positive elements (Li and Vasconcelos, 2015). To deal with label noise, Carbonneau et al. (2018) suggests that relaxing the strict MIL assumption is a proper approach when modeling. We believe that a bag is more likely to be positive if it has a higher proportion of positive instances, and unlikely to be positive if it contains a lower proportion of positive instances. In summary, we can consider a threshold based on the percentage of positive instances to classify bags as positive or negative, such as in Li and Vasconcelos (2015). In other words, we suppose the amount of positive instances is required to classify a bag as positive.

Another source of label ambiguity is different label spaces, which means that the label spaces between instances and bags are different. As an example in Carbonneau et al. (2018), a positive bag represents a picture including a zebra, and a positive instance implies that the sub-region of the picture contains the specific characteristics of the zebra, like black and white stripes. They consider an example where a negative bag includes some positive instances. Specifically, a white tiger image (negative bag) can also extract similar black and white stripes of zebras in some patches (positive instances). In this situation, it is hard to clarify the exact meaning of those positive patches in the negative image, thus we say the labels of instances and bags exist in different spaces. Moreover, they mention

doi:10.6342/NTU202401899

that several methods, including MILES (Chen et al., 2007) which is an embedding-based method, are developed to tackle this kind of issue.



1.5 Motivation

Label ambiguity is inherent to MIL tasks as we discussed previously, and relaxing the restricted MIL assumption can be an approach to deal with this issue. If we still construct MIL algorithms using the MIL assumption, it may be too restrictive for models. Moreover, MI data often does not adhere to the MIL assumption precisely in practical situations. Consequently, we propose a new bag likelihood to relax the strict MIL assumption while modeling, thereby increasing the robustness of MIL models.

The proposed bag likelihood can be applied to any kind of instance classifier in IS, such as logistic or probit regression. Specifically, we can utilize logistic regression to construct the classifier for instance label, and then apply our bag likelihood to predict bag labels. Notably, the distribution of each bag label is conditional on its instance labels. Although Haußmann et al. (2017) has developed an IS approach with relaxed MIL assumption, it becomes difficult to explain the relationship between instances and their bags on MI data. On the other hand, the new proposed bag likelihood can illustrate the connections between instances and bags.

In this thesis, we employ Bayesian approaches to model and estimate parameters. We provide the prior distribution for the parameters in logistic regression and use the Gibbs sampling method to estimate those parameters. This is the first time that Gibbs sampling has been applied to the MIL model with logistic regression.

doi:10.6342/NTU202401899

Chapter 2 Background

This chapter introduces two common MIL models that satisfy the MIL assumption, and these two models are constructed based on the IS and the Bayesian scheme. The model in Section 2.1 assumes the linearity in the dataset, while the model in Section 2.2 imposes no restrictions on the dataset's distribution. Finally, we explain an augmentation technique known as Pólya-Gamma augmentation (Polson et al., 2013) in Section 2.3, which is frequently used in the Gibbs sampling process for logistic regression models.

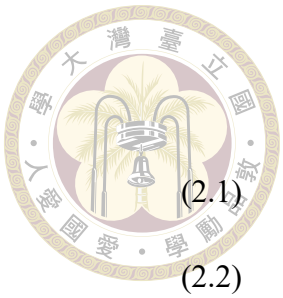
2.1 Multiple Instance Logistic Regression (MILR)

In the MIL setting, the training data $\{\{\boldsymbol{x}_{ij}\}_{j=1}^{m_i}, y_i\}_{i=1}^{N}$ consists of N bags, where m_i indicates the number of instances in the ith bag. There are $M = \sum_{i=1}^{N} m_i$ instances in total. All instances can be displayed by the matrix $\boldsymbol{X} = [\boldsymbol{x}_{11}, \ldots, \boldsymbol{x}_{1m_1}, \ldots, \boldsymbol{x}_{N1}, \ldots, \boldsymbol{x}_{Nm_N}]^T$, and we define $\boldsymbol{Y} = [y_1, \ldots, y_N]^T$ as all bag labels, where $\boldsymbol{x}_{ij} \in \mathbb{R}^{d\times 1}$, $y_i \in \{0, 1\}$, $i = 1, \ldots, N$, and $j = 1, \ldots, m_i$. For the ith bag, $\boldsymbol{z}_i = [z_{i1}, \ldots, z_{im_i}]^T$ indicates latent instance labels corresponding to instances $[\boldsymbol{x}_{i1}, \ldots, \boldsymbol{x}_{im_i}]^T$, where $z_{ij} \in \{0, 1\}$. Note that d is the number of features.

We let $\boldsymbol{\Sigma} \in \mathbb{R}^{d\times d}$ to be a diagonal matrix. Also, we consider $\boldsymbol{\beta} \in \mathbb{R}^{d\times 1}$ and $\boldsymbol{v} \in \mathbb{R}^{d\times 1}$. Multiple Instance Logistic Regression (MILR, Chen et al. (2017)) model is shown

doi:10.6342/NTU202401899

as following,



$$\boldsymbol{\beta} \sim \mathcal{N}(\boldsymbol{v}, \boldsymbol{\Sigma}), \tag{2.1}$$

$$z_{ij}|\boldsymbol{\beta}, \boldsymbol{x}_{ij} \sim \text{Ber}(p_{ij}),\ p_{ij} = \frac{1}{1+\exp(-\boldsymbol{x}_{ij}^T\boldsymbol{\beta})} \tag{2.2}$$

$$y_i|z_{i1}, \ldots, z_{im_i} = \begin{cases} 0, & \text{if } z_{ij} = 0,\ \forall j \\ 1, & \text{o.w.}, \end{cases} \tag{2.3}$$

where $i = 1, \ldots, N$ and $j = 1, \ldots, m_i$. This model adopts a Bayesian approach, so a prior distribution of parameter $\boldsymbol{\beta}$ is given in (2.1). Moreover, it is better to apply this model to the dataset satisfying the linear assumption, which is operated in logistic regression. Lastly, (2.3) shows that MILR fulfills the MIL assumption which is a strict assumption.

2.2 Gaussian Process MILR (GPMILR)

We denote some notations and briefly review Gaussian Process (GP) logistic regression before introducing Gaussian Process MILR (GPMILR). Gaussian Processes (GPs) consist of instance scalars $\boldsymbol{f} = [f_{11}, \ldots, f_{1m_1}, \ldots, f_{N1}, \ldots, f_{Nm_N}]^T$ for all instances, where $f_{ij} \in \mathbb{R}$. We define $K_{\boldsymbol{X}'\boldsymbol{X}''} \in \mathbb{R}^{M' \times M''}$ to be the gram matrix of the two datasets, $\boldsymbol{X}'$ and $\boldsymbol{X}''$. Here, M' and M'' are the number of instances in $\boldsymbol{X}'$ and $\boldsymbol{X}''$, respectively. The (i, j)th element of the gram matrix is denoted by $k(x_i', x_j'')$. We also consider the RBF kernel function $k(x_i', x_j'') = \exp\left(-\frac{1}{2\theta^2}(x_i' - x_j'')^T(x_i' - x_j'')\right)$ in this thesis, where θ is the length-scale parameter. A samller θ makes the differences between x_i' and x_j'' more significant, so it can create a more complex model. Conversely, a larger θ smooths the variations in $k(x_i', x_j'')$ to prevent overfitting. In addition, diag$(\cdot)$ denotes a function that retains the diagonal elements of the matrix while setting all off-diagonal elements to zero.

doi:10.6342/NTU202401899

The model of GP logistic regression (Rasmussen and Williams, 2005) can be written as

$$\boldsymbol{f}(\boldsymbol{X}) \sim \mathcal{N}(\mathbf{0}_M, K_{\boldsymbol{XX}}), \tag{2.4}$$

$$z_{ij}|f_{ij} \sim \text{Ber}(p_{ij}),\ p_{ij} = \frac{1}{1+\exp(-f_{ij})}. \tag{2.5}$$

GPMILR is defined by equations (2.4), (2.5), and (2.3), which include GP logistic regression. Moreover, other MIL models, such as the one proposed by Haußmann et al. (2017), also utilize GPs in their classifiers. This is because GPs eliminate the requirement for linear assumption in the dataset. However, GPs typically encounter computational challenges, especially when dealing with large datasets. For the prediction process on testing data $\boldsymbol{X}^*$, we derive the posterior predictive distribution as $\boldsymbol{f}^*|\boldsymbol{f}, \boldsymbol{X}, \boldsymbol{X}^* \sim \mathcal{N}(K_{\boldsymbol{XX}^*}^T K_{\boldsymbol{XX}}^{-1}\boldsymbol{f},\ K_{\boldsymbol{XX}} - K_{\boldsymbol{XX}^*}^T K_{\boldsymbol{XX}}^{-1} K_{\boldsymbol{XX}^*})$. The dimension of the matrix $K_{\boldsymbol{XX}} \in \mathcal{R}^{M\times M}$ becomes large when the size of training data grows. Therefore, computing the inverse of $K_{\boldsymbol{XX}}$ becomes inefficient due to its $\mathcal{O}(M^3)$ computational cost. As a result, Kandemir et al. (2016) introduces an approach with inducing points, called Fully Independent Training Conditional (FITC) approximation, to reduce the computational cost of $K_{\boldsymbol{XX}}^{-1}$ to $\mathcal{O}(q^2M)$, where q is the number of inducing points.

FITC approximation considers q inducing points $\boldsymbol{S} = [\boldsymbol{s}_1, \ldots, \boldsymbol{s}_q]^T$ and their corresponding inducing scalars $\boldsymbol{u} = [u_1, \ldots, u_q]^T$, where $\boldsymbol{s}_l \in \mathbb{R}^{d\times 1}$, $u_l \in \mathbb{R}$, and $l = 1, \ldots, q$. We can rewrite (2.4) in GPMILR to be

$$\boldsymbol{u}(\boldsymbol{S}) \sim \mathcal{N}(\mathbf{0}_q, K_{\boldsymbol{SS}}), \tag{2.6}$$

$$\boldsymbol{f}|\boldsymbol{u}, \boldsymbol{X}, \boldsymbol{S} \sim \mathcal{N}(K_{\boldsymbol{XS}}K_{\boldsymbol{SS}}^{-1}\boldsymbol{u}, \mathcal{K}), \tag{2.7}$$

where $\mathcal{K} = \text{diag}(K_{\boldsymbol{XX}} - K_{\boldsymbol{XS}}K_{\boldsymbol{SS}}^{-1}K_{\boldsymbol{SX}})$ is a diagonal matrix. Thus, it is apparent that

doi:10.6342/NTU202401899

elements in $\boldsymbol{f}$ are supposed to be independent. Notably, inducing points can be a subset of the training data, and we choose inducing points through the K-means method here.

2.3 Pólya-Gamma Augmentation

We employ a Gibbs sampling method to infer the MIL models in our study. This method estimates parameters through the true posterior distribution. We will utilize a technique called Pólya-Gamma augmentation (Polson et al., 2013) during the process of Gibbs sampling estimation, therefore we give a brief introduction to Pólya-Gamma augmentation in this section.

Polson et al. (2013) demonstrates that the specific type of log-odds can be represented by the form of Pólya-Gamma distribution. Their relationship is shown as the following equation,

$$\frac{(e^{\psi})^a}{(1+e^{\psi})^b} = 2^{-b} e^{\kappa\psi} \int_0^{\infty} e^{-\omega\psi^2/2} p(\omega) d\omega, \tag{2.8}$$

where $b > 0$, $\kappa = a - \frac{b}{2}$. Moreover, $\omega \sim \text{PG}(b, 0)$ is known as Pólya-Gamma latent variable, and $\omega \overset{\text{D}}{=} \frac{1}{2\pi^2} \sum_{i=1}^{\infty} \frac{h_i}{(i-1/2)^2}$, where $h_i \sim \text{Gamma}(b, 1)$. It is notable that $\overset{\text{D}}{=}$ indicates equality in distribution. The left-hand side of the equation (2.8) represents a specific type of log-odds, while the right-hand side of it is illustrated by the form of a Pólya-Gamma distribution. The posterior distribution of logistic regression exists a particular kind of log-odds, which is similar to the left-hand side of (2.8) and hard to sample directly. Consequently, adopting (2.8) allows us to acquire a transformed distribution with the Pólya-Gamma distribution, which is simpler to sample from.

doi:10.6342/NTU202401899

Chapter 3 Methodology

3.1 Logistic Aggregation Model (LAM)

Many MIL models, such as MILR and GPMILR, are inefficient to be applied to MI data with label ambiguity since they are developed based on the MIL assumption. To address this issue, we propose a novel bag likelihood called Logistic Aggregation Model (LAM). LAM can relax the MIL assumption and is formulated as

$$y_i|z_{i1},\ldots,z_{im_i} \sim \text{Ber}(g_i),\ g_i = \frac{1}{1+\exp(-t(\frac{k_i}{m_i}-r))}, \tag{3.1}$$

where $k_i = \#\{z_{ij}=1, \forall j\}, i = 1,\ldots,N$.

It is notable that r and t are both hyperparameters. The rate r can be seen as a threshold to classify the labels of bags, and the positive constant t indicates the degree of the model adhering to the relaxed MIL assumption. Compared to the MIL assumption, which claims that $y_i = 0$ if $z_{ij} = 0, \forall j$ and $y_i = 1$, otherwise; LAM relaxes this assumption, so it should be able to tolerate some noise in the data. Furthermore, LAM approximates the MIL assumption under the settings that $0 < r < \frac{1}{\max\{m_i\}_{i=1}^N}$ and t towards infinity. LAM can also be easily applied to various MIL models, which include the specific instance and bag models. We then show two examples in Section 3.2 and Section 3.3.

doi:10.6342/NTU202401899

3.2 MILR-LAM

We consider MILR with proposed LAM (MILR-LAM), and this model is demonstrated by (2.1), (2.2) and (3.1). Moreover, we write down the posterior distribution of parameters $\boldsymbol{\beta}$ and $\boldsymbol{Z}$ for Gibbs sampling in the below equation.

$$P(\boldsymbol{\beta}, \boldsymbol{Z}|\boldsymbol{Y}, \boldsymbol{X}) \propto \prod_{i=1}^{N} \left[P(y_i|z_{ij}, \forall j) \prod_{j=1}^{m_i} P(z_{ij}|\boldsymbol{\beta}, \boldsymbol{x}_{ij}) \right] \cdot \pi(\boldsymbol{\beta}). \tag{3.2}$$

The updating process of the Gibbs sampling approach is shown as follows.

Updating $\hat{\boldsymbol{\beta}}$ and $\hat{\boldsymbol{\omega}}$:

The conditional posterior distribution of $\boldsymbol{\beta}$ is

$$\begin{aligned} P(\boldsymbol{\beta}|\boldsymbol{Z}, \boldsymbol{Y}, \boldsymbol{X}) &\propto \left[\prod_{i=1}^{N} \prod_{j=1}^{m_i} P(z_{ij}|\boldsymbol{\beta}, \boldsymbol{x}_{ij}) \right] \cdot \pi(\boldsymbol{\beta}) \\ &\propto \left[\prod_{i=1}^{N} \prod_{j=1}^{m_i} p_{ij}^{z_{ij}} (1 - p_{ij})^{(1-z_{ij})} \right] \cdot \pi(\boldsymbol{\beta}) \\ &\propto \left[\prod_{i=1}^{N} \prod_{j=1}^{m_i} \frac{(e^{\boldsymbol{x}_{ij}^T \boldsymbol{\beta}})^{z_{ij}}}{(1 + e^{\boldsymbol{x}_{ij}^T \boldsymbol{\beta}})} \right] \cdot \pi(\boldsymbol{\beta}). \end{aligned} \tag{3.3}$$

Next, we apply the Pólya-Gamma augmentation technique on the equation (3.3). We denote $\boldsymbol{\omega} = (\omega_{11}, \ldots, \omega_{1m_1}, \ldots, \omega_{N1}, \ldots, \omega_{Nm_N})$ as Pólya-Gamma latent variables, where $\omega_{ij} \in \mathbb{R}$. We also define $a = z_{ij}$, $b = 1$, and $\psi = \boldsymbol{x}_{ij}^T \boldsymbol{\beta}$ according to (2.8). The augmented conditional posterior of $\boldsymbol{\beta}$ and $\boldsymbol{\omega}$ can be written as,

$$\begin{aligned} P(\boldsymbol{\beta}, \boldsymbol{\omega}|\boldsymbol{Z}, \boldsymbol{Y}, \boldsymbol{X}) &\propto \left[\prod_{i=1}^{N} \prod_{j=1}^{m_i} \exp\left((z_{ij} - \frac{1}{2}) \boldsymbol{x}_{ij}^T \boldsymbol{\beta} \right) \exp\left(-\frac{\omega_{ij}}{2} (\boldsymbol{x}_{ij}^T \boldsymbol{\beta})^2 \right) p(\omega_{ij}) \right] \cdot \pi(\boldsymbol{\beta}) \\ &\propto \exp\left\{ -\frac{1}{2} \left[\boldsymbol{\beta}^T (\boldsymbol{X}^T \boldsymbol{\Omega} \boldsymbol{X} + \boldsymbol{\Sigma}^{-1}) \boldsymbol{\beta} - 2 \boldsymbol{\beta}^T (\boldsymbol{X}^T \boldsymbol{C} + \boldsymbol{\Sigma}^{-1} \boldsymbol{v}) \right] \right\} \cdot p(\boldsymbol{\omega}), \end{aligned} \tag{3.4}$$

doi:10.6342/NTU202401899

where $\omega_{ij} \sim \text{PG}(1,0)$ and $\boldsymbol{C} = [(z_{11} - \frac{1}{2}), \ldots, (z_{Nm_N} - \frac{1}{2})]^T \in \mathbb{R}^{M\times 1}$. Also, $\boldsymbol{\Omega} \in \mathbb{R}^{M\times M}$ is a diagonal matrix with elements $(\omega_{11}, \ldots, \omega_{1m_1}, \ldots, \omega_{N1}, \ldots, \omega_{Nm_N})$ on the diagonal. Thus, the updating formula of $\hat{\boldsymbol{\beta}}$ and $\hat{\boldsymbol{\omega}}$ are

$$\boldsymbol{\beta}|\boldsymbol{\omega}, \boldsymbol{Z}, \boldsymbol{Y}, \boldsymbol{X} \sim \mathcal{N}(A, B) \tag{3.5}$$

$$\omega_{ij}|\boldsymbol{\beta}, \boldsymbol{Z}, \boldsymbol{Y}, \boldsymbol{X} \sim \text{PG}(1, \boldsymbol{x}_{ij}^T\boldsymbol{\beta}). \tag{3.6}$$

Here, $A = B \cdot (\boldsymbol{X}^T\boldsymbol{C} + \boldsymbol{\Sigma}^{-1}\boldsymbol{v})$ and $B = (\boldsymbol{X}^T\boldsymbol{\Omega}\boldsymbol{X} + \boldsymbol{\Sigma}^{-1})^{-1}$.

Updating $\hat{\boldsymbol{Z}}$:

Since the bag labels are observed, we separate the conditional posterior distribution $P(z_{ij}|\boldsymbol{\beta}, z_{is}, \forall s \neq j, \boldsymbol{Y}, \boldsymbol{X}) = P(z_{ij}|\boldsymbol{\beta}, z_{is}, \forall s \neq j, y_i, \boldsymbol{x}_{ij}) \propto P(y_i|z_{ij}, \forall j) \cdot \prod_{j=1}^{m_i} P(z_{ij}|\boldsymbol{\beta}, \boldsymbol{x}_{ij})$ into two different cases. For the first case, we consider the positive bag label ($y_i = 1$) and observe that the updating formula of z_{ij} is

$$z_{ij}|\boldsymbol{\beta}, z_{is}, \forall s \neq j, y_i = 1, \boldsymbol{x}_{ij} \sim \text{Ber}\left(\frac{P_{11}(\boldsymbol{x}_{ij}^T\boldsymbol{\beta})}{P_{11}(\boldsymbol{x}_{ij}^T\boldsymbol{\beta}) + P_{10}(\boldsymbol{x}_{ij}^T\boldsymbol{\beta})}\right), \tag{3.7}$$

where

$$P_{11}(x) = \frac{1}{1 + \exp(-t(\frac{k_i'+1}{m_i} - r))} \cdot \frac{1}{1 + \exp(-x)},$$
$$P_{10}(x) = \frac{1}{1 + \exp(-t(\frac{k_i'}{m_i} - r))} \cdot \left[1 - \frac{1}{1 + \exp(-x)}\right],$$

and $k_i' = \#\{z_{is} = 1, \forall s \neq j\}$. The second case occurs when $y_i = 0$, thus the updating formula of z_{ij} becomes

$$z_{ij}|\boldsymbol{\beta}, z_{is}, \forall s \neq j, y_i = 0, \boldsymbol{x}_{ij} \sim \text{Ber}\left(\frac{P_{01}(\boldsymbol{x}_{ij}^T\boldsymbol{\beta})}{P_{01}(\boldsymbol{x}_{ij}^T\boldsymbol{\beta}) + P_{00}(\boldsymbol{x}_{ij}^T\boldsymbol{\beta})}\right), \tag{3.8}$$

doi:10.6342/NTU202401899

where

$$P_{01}(x) = \left[1 - \frac{1}{1+\exp(-t(\frac{k_i'+1}{m_i} - r))}\right] \cdot \frac{1}{1+\exp(-x)},$$
$$P_{00}(x) = \left[1 - \frac{1}{1+\exp(-t(\frac{k_i'}{m_i} - r))}\right] \cdot \left[1 - \frac{1}{1+\exp(-x)}\right].$$

3.3 GPMILR-LAM

LAM can be also applied to GPMILR, and we name this model as GPMILR-LAM. That is, GPMILR-LAM can be demonstrated by (2.6), (2.7), (2.5), and (3.1), with the consideration of inducing points. We still infer the model using the Gibbs sampling approach, and the posterior distribution of $\boldsymbol{u}$, $\boldsymbol{f}$, $\boldsymbol{Z}$ can be written as

$$P(\boldsymbol{u}, \boldsymbol{f}, \boldsymbol{Z}|\boldsymbol{Y}, \boldsymbol{X}, \boldsymbol{S}) \propto \prod_{i=1}^{N} \left[P(y_i|z_{ij}, \forall j) \prod_{j=1}^{m_i} P(z_{ij}|f_{ij})\right] \cdot P(\boldsymbol{f}|\boldsymbol{u}, \boldsymbol{X}, \boldsymbol{S}) \cdot P(\boldsymbol{u}(\boldsymbol{S})). \tag{3.9}$$

Similarly, we derive the conditional posterior distributions of $\boldsymbol{u}$, $\boldsymbol{f}$, and $\boldsymbol{Z}$, respectively. We also show their updating formulas, including the Pólya-Gamma latent variables $\boldsymbol{\omega}$.

Updating $\hat{\boldsymbol{u}}$, $\hat{\boldsymbol{f}}$, and $\hat{\boldsymbol{\omega}}$:

$$\boldsymbol{u}|\boldsymbol{f}, \boldsymbol{z}, \boldsymbol{Y}, \boldsymbol{X}, \boldsymbol{S} \sim \mathcal{N}(A', B'), \tag{3.10}$$

$$\boldsymbol{f}|\boldsymbol{\omega}, \boldsymbol{u}, \boldsymbol{Z}, \boldsymbol{Y}, \boldsymbol{X}, \boldsymbol{S} \sim \mathcal{N}(A'', B''), \tag{3.11}$$

$$\omega_{ij}|f_{ij}, \boldsymbol{u}, \boldsymbol{Z}, \boldsymbol{Y}, \boldsymbol{X}, \boldsymbol{S} \sim \text{PG}(1, f_{ij}), \tag{3.12}$$

where $A' = B' \cdot (K_{\boldsymbol{XS}}K_{\boldsymbol{SS}}^{-1})^T\mathcal{K}^{-1}\boldsymbol{f}$, $B' = ((K_{\boldsymbol{XS}}K_{\boldsymbol{SS}}^{-1})^T\mathcal{K}^{-1}(K_{\boldsymbol{XS}}K_{\boldsymbol{SS}}^{-1}) + K_{\boldsymbol{SS}}^{-1})^{-1}$, $A'' = B'' \cdot (\boldsymbol{C} + \mathcal{K}^{-1}K_{\boldsymbol{XS}}K_{\boldsymbol{SS}}^{-1}\boldsymbol{u})$, and $B'' = (\boldsymbol{\Omega} + \mathcal{K}^{-1})^{-1}$. We realize that B'' is a

doi:10.6342/NTU202401899

diagonal matrix, so the conditional posterior distributions of each f_{ij} are independent.

Updating $\hat{\boldsymbol{Z}}$:

We still consider two cases according to the different values of observed y_i, which is the same updating process as in MILR-LAM. That is,

$$z_{ij}|f_{ij}, \boldsymbol{u}, z_{is}, \forall s \neq j, y_i = 1, \boldsymbol{X}, \boldsymbol{S} \sim \text{Ber}\left(\frac{P_{11}(f_{ij})}{P_{11}(f_{ij}) + P_{10}(f_{ij})}\right), \tag{3.13}$$

$$z_{ij}|f_{ij}, \boldsymbol{u}, z_{is}, \forall s \neq j, y_i = 0, \boldsymbol{X}, \boldsymbol{S} \sim \text{Ber}\left(\frac{P_{01}(f_{ij})}{P_{01}(f_{ij}) + P_{00}(f_{ij})}\right). \tag{3.14}$$

When a new data $\boldsymbol{X}^*$ comes, we derive the estimates to be $\hat{\boldsymbol{f}}^* = K_{\boldsymbol{X}^*\boldsymbol{S}} K_{\boldsymbol{S}\boldsymbol{S}}^{-1} \hat{\boldsymbol{u}}$. For the ith bag, the predicted probability of instance labels are $\hat{p_{ij}^*} = 1/(1 + \exp(-\hat{f_{ij}^*}))$, $\forall j$, and the predicted probability of bag label is $\hat{g_i^*} = 1/(1 + \exp(-t(\frac{\hat{k_i^*}}{m_i} - r)))$. In addition, since $k_i^* = \#\{z_{ij}^* = 1, \forall j\} = \sum_j \mathbb{I}(z_{ij}^* = 1)$, we consider the estimate of k_i^* to be its expectation, i.e. $\hat{k_i^*} = \sum_j \hat{p_{ij}^*}$.

3.4 Comparison between LAM and the Bag Likelihood of VGPMIL

Haußmann et al. (2017) proposes Variational Gaussian Process Multiple Instance Learning (VGPMIL) model, which is similar to GPMILR-LAM, but a distinction lies in the bag model. Therefore, we discuss the differences between the bag likelihoods of these two models in this section. Both of them aim to relax the strict MIL assumption. However, the bag likelihood of VGPMIL adopts an alternative approach shown in (3.15).

$$p(y_i|\{z_{ij}\}_{j=1}^{m_i}) = \left(\frac{H}{H+1}\right)^{G_i} \left(\frac{1}{H+1}\right)^{(1-G_i)} = \frac{H^{G_i}}{H+1}, \tag{3.15}$$

doi:10.6342/NTU202401899

where $G_i := y_i \max\{z_{ij}\}_{j=1}^{m_i} + (1-y_i)(1-\max\{z_{ij}\}_{j=1}^{m_i}), \forall i$, and H is a positive constant with the ability to deal with different level of noise in the MI data. When $\{z_{ij}\}_{j=1}^{m_i}$ and y_i are observed, G_i serves as an indicator to determine if the MI data satisfies the MIL assumption. Specifically, $G_i = 1$ implies that the MI data fulfills the MIL assumption due to satisfying $y_i = \max\{z_{ij}\}_{j=1}^{m_i}$; otherwise, $G_i = 0$. Moreover, the bag likelihood of VGPMIL approximates the MIL assumption as H approaches infinity.

It is evident that VGPMIL also exists a bag likelihood with relaxed MIL assumption. Nevertheless, there are some disadvantages in this model. Determining the value of H can solely rely on our personal interpretation of the MI data because they do not provide a specific process to choose it. Also, it is challenging to explain which type of label ambiguity in the MI data through the value of H since this hyperparameter lacks a sensible interpretation. In contrast, our proposed LAM introduces t and r, which have precise meanings, so they can bring additional information to explain the noise presented in the MI data.

To be more specific, we use the value of $0 < r < 1$ to determine the required proportion of positive instances within a bag to classify it as positive. A larger r represents more positive instances in a bag needed to label it as positive. On the other hand, when r is small, a bag only requires a few positive instances to label positive. The hyperparameter t determines how strictly the model adheres to the classification threshold set by r. Theoretically, the model strictly adheres to the classification principle when t is close to infinity, as the probability g_i of the Bernoulli distribution converges to 0 or 1. However, when t is close to zero, i.e., g_i towards 0.5, it suggests the presence of significant noise in the data. In summary, LAM provides more flexibility to ensure the robustness of the model than the bag model in VGPMIL does. By tuning r and t, we gain information about

doi:10.6342/NTU202401899

the sources of label ambiguity, which allows us to construct a more robust model that can effectively address noise in the data.

In addition to the LAM's ability to effectively interpret MI data, model inference using Gibbs sampling offers several benefits. Notably, VGPMIL uses variational inference, whereas GPMILR-LAM is inferred using Gibbs sampling. The Gibbs sampling approach considers the true posterior distribution, unlike variational inference, which relies on the approximation of the posterior distribution. This allows us to capture a more accurate model structure during statistical inference, and it helps lead to improved accuracy and stability in the estimation process.

doi:10.6342/NTU202401899

doi:10.6342/NTU202401899

Chapter 4 Simulation

In this chapter, we provide three distinct simulation scenarios to evaluate our proposed LAM by leveraging MILR-LAM and GPMILR-LAM on various MI datasets. We also compare MILR-LAM and GPMILR-LAM with the other four benchmark methods. MILR-LAM and GPMILR-LAM are mentioned in Section 3.2 and Section 3.3, respectively. Both of them utilize Gibbs sampling inference with the technique of Pólya-Gamma augmentation. Moreover, we use Area Under the Curve (AUC) to examine the performance results of various methods. An AUC close to 1 indicates that the model has a nearly perfect ability to distinguish between positive and negative groups. On the other hand, when AUC=0.5, it means that the model performs the random guessing of positive and negative groups. If the AUC value is less than 0.5, it indicates that the model has poor classification ability. We then explain the four benchmark methods as follows,

- **MILR:** Chen et al. (2017) proposes this model with inference performed by the Expectation-maximization (EM) algorithm and provides the R package `milr` to use. We do not consider least absolute shrinkage and selection operator (LASSO) penalty term in our simulations, which is the default setting in `milr`. It is notable that MILR model satisfies the MIL assumption.

- **MILR (Gibbs):** The model of MILR (Gibbs) is the same as MILR, but its statistical

doi:10.6342/NTU202401899

inference uses Gibbs sampling with the technique of Pólya-Gamma augmentation.

- **GPMILR:** We conduct Gibbs sampling inference on the GPMILR model mentioned in Section 2.2. This model fulfills the MIL assumption.
- **VGPMIL:** Haußmann et al. (2017) proposes this model with the variational inference method, and provides a Python package `vgpmil` to operate. We consider 2000 iterations for the variational inference. This is the only benchmark method not satisfying the MIL assumption.

Several methods employ Gibbs sampling, so we have to set the number of iterations for their inferences. The number of iterations varies based on the characteristics of the two types of models, linear or non-linear. Specifically, for MIL models with a linear assumption, such as MILR (Gibbs) and MILR-LAM, we iterate 1000 times with no thinning and burn-in the first 500 samples. For MIL models without a linear assumption, we account for the complex structure and instabilities of models by increasing the number of iterations. Therefore, we iterate 3000 times with no thinning and burn-in the first 2000 samples. The non-linear models include GPMILR and GPMILR-LAM, which consider GPs. For VGPMIL, we set the length-scale required in GPs to be $\sqrt{d}$, following the same setting as in Haußmann et al. (2017). For GPMILR and GPMILR-LAM, we set the length-scale to be $\frac{1}{\sqrt{2\sqrt{d}}}$.

We operate two types of simulated data: linear and non-linear. The following are the steps to generate simulated MI data.

1. Generate $M \times d$ samples from a standard normal distribution and put them into a $\mathbb{R}^{M \times d}$ matrix to create instances $\boldsymbol{X}$. It is notable that M is the total number of instances and we consider the case of $d = 2$ in our simulations.

doi:10.6342/NTU202401899

2. Provide each instance with an instance label. Explicitly, for ith bag, each instance label z_{ij} is determined by a Bernoulli distribution with probability p_{ij}, and $j = 1, \ldots, m_i$. That is,

$$z_{ij}|\boldsymbol{x}_{ij} \sim \text{Ber}(p_{ij}),$$

where p_{ij} depends on whether you want linear or non-linear MI data.

- **Linear MI data:**

  Given $\boldsymbol{\beta}$, and we calculate p_{ij} using the formula:

$$p_{ij} = \frac{1}{1 + \exp(-\boldsymbol{x}_{ij}^T \boldsymbol{\beta})}.$$

- **Non-linear MI data:**

  Define a non-linear function $f(\boldsymbol{x}_{ij})$, and then calculate p_{ij} by the following formula:

$$p_{ij} = \frac{1}{1 + \exp(-f(\boldsymbol{x}_{ij}))}$$

3. We choose a threshold value (rate) between 0 and 1, and compute the proportion of positive instances for each bag. If this proportion exceeds the threshold value, we label the bag as positive ($y_i = 1$); otherwise, we assign it with a negative bag label ($y_i = 0$).

4. Return the simulated MI data: $\{\{\boldsymbol{x}_{ij}\}_{j=1}^{m_i}, y_i\}_{i=1}^{N}$.

All the simulated MI datasets in this chapter contain label ambiguity, meaning they are designed not to fulfill the MIL assumption. In the first simulation, we analyze the effects of label ambiguity on various methods by adding different kinds of noise to the datasets. Secondly, we examine how the classification ability of different methods changes

doi:10.6342/NTU202401899

as the bag size increases, where the bag size represents the number of instances in the bag. Finally, we aim to investigate the impact of varying the true threshold value used to define bag labels. Specifically, we examine four simulated MI datasets, where bag labels in each dataset are generated based on a distinct threshold value (rate).

4.1 Impact of Label Noise on Model Performance

The first simulation evaluates the predicted results of several methods on MI datasets, which are designed not to fulfill the MIL assumption and contain some label noise. We break this simulation into two parts: equal bag size and unequal bag size. To be more specific, equal bag size means that the number of instances in different bags is equal throughout a simulated dataset. However, unequal bag size represents that different bags have different number of instances.

For the training and testing data of considering equal bag size, we set the number of bags to be 200 and 10 instances in each bag (i.e., bag sizes are all equal to 10). There are 2000 instances in total. Besides, we study two kinds of label noise: mislabeled bags and using different rates to label bags. The former fixes the threshold at 0.35, meaning a positive bag should include at least four positive instances. This mislabeling process also changes 50 negative bag labels to positive ones and only operates on the bags with the proportion of positive instances between 0.1 and 0.35. The latter assumes that two different rates, 0.15 and 0.35, decide the bag labels. Each rate determines 100 bag labels, respectively.

Furthermore, we set $\boldsymbol{\beta} = (\beta_0, \beta_1, \beta_2) = (-2, 3, 0.5)$ to generate linear MI data, where β_0 is the coefficient of the intercept, and define $f(\boldsymbol{x}_{ij}) = -3\cos(\boldsymbol{x}_{ij1}) + \frac{1}{2}\exp(\boldsymbol{x}_{ij2})$ for

doi:10.6342/NTU202401899

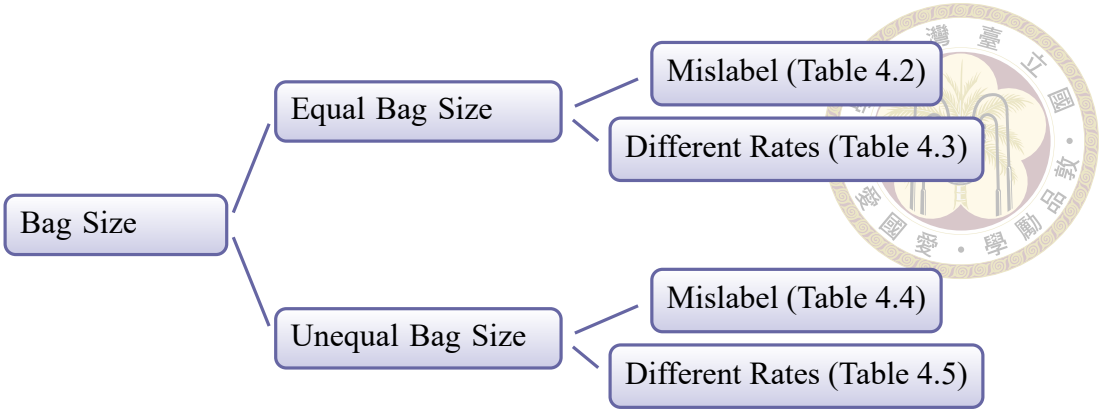


Figure 4.1: The summary of all cases in Section 4.1

non-linear MI data, where $\boldsymbol{x}_{ij} \in \mathbb{R}^2$. We set the number of inducing points for the GP models to 10, approximating 0.005 of the total number of instances. Simulations are replicated 50 times on each MIL model, demonstrating their performances through AUC's mean and standard deviation on both instance and bag labels. While utilizing the LAM, tuning the hyperparameters t and r is important. We consider all combinations of candidates for $t = \{10, 100\}$ and $r = \{0.15, 0.25, 0.35, 0.45\}$. We then rank these combinations from 1 to 8 twice based on the testing AUCs of instances and bags. Lastly, we sum each combination's ranking number of instance and bag, and choose the t and r with the highest ranking for LAM. We also provide the sensitivity analysis on hyperparameters.

While considering unequal bag size, the settings are similar to those in equal bag size. However, we do not fix the bag size for all 200 bags. Based on Table 4.1, we consider 100 bags with a size of 5 and another 100 bags with a size of 10, so there are 1500 instances in total. Additionally, we only consider 7 inducing points in unequal bag size due to fewer instances. The summary of all cases in this section is demonstrated in Figure 4.1.

	Bags	Bag Size	Total Instances	Inducing Points
Equal Bag Size	200	10	2000	10
Unequal Bag Size	100/100	5/10	1500	7

Table 4.1: Comparison between equal and unequal bag sizes.

doi:10.6342/NTU202401899

4.1.1 Results of Equal Bag Size

Table 4.2 (Mislabel) and Table 4.3 (Different rates) demonstrate the prediction results for equal bag size, and these two kinds of label noise acquire similar results. Each label noise operates simulation on both linear and non-linear MI data. For the linear MI data, although AUCs of MILR-LAM does not significantly differ among all the other models, except for GPMILR, as their 95% confidence intervals (CIs) of AUC overlap. A 95% CI is computed by the formula: average AUC $\pm$ ($1.96\times$ standard deviation (Std)). MILR-LAM still achieves the highest average AUC on instance and bag label predictions. This result is reasonable since MILR-LAM is a linear model that addresses the label noise. For the non-linear MI data, the models using GPs exhibit better prediction results regarding average AUC, especially GPMILR-LAM, which shows the highest one. In conclusion, our proposed MILR-LAM and GPMILR-LAM achieve the best average AUC across different data structures and are comparable to most benchmark methods.

Moreover, we gain insights into the sensitivity analysis of the hyperparameters of LAM based on Figure 4.2 and Figure 4.3. Each plot exhibits eight boxplots of predicted AUC according to the eight combinations of t and r. We take Figure 4.2 (c) and (d) as examples, which show predicted AUC results by GPMILR-LAM on the mislabeled non-linear MI data. Also, (c) is the result of the instance label, and (d) is for the bag label. We can see similar patterns in the boxplots for $t = 10$ (blue) and $t = 100$ (purple). However, when emphasizing a fixed number of t, there are noticeable differences in average AUCs based on different r. This indicates the selection of r might be more sensitive than that of t. Additionally, AUCs of GPMILR-LAM are more easily affected by the selection of these hyperparameters than AUCs of MILR-LAM due to the complexity of the GP model.

doi:10.6342/NTU202401899

	Train		Test	
	instances	bags	instances	bags
(Linear)				
MILR	0.91(0.020)	0.69(0.032)	0.91(0.020)	0.80(0.036)
MILR (Gibbs)	0.91(0.022)	0.69(0.033)	0.91(0.021)	0.80(0.038)
MILR-LAM(10/0.25)	0.92(0.012)	0.71(0.029)	**0.92**(0.013)	**0.85**(0.030)
MILR-LAM(10/0.35)	0.92(0.009)	0.71(0.029)	**0.92**(0.011)	**0.85**(0.030)
VGPMIL	0.88(0.030)	0.73(0.027)	0.88(0.029)	0.82(0.034)
GPMILR	0.76(0.042)	0.72(0.041)	0.76(0.041)	0.66(0.045)
GPMILR-LAM(10/0.35)	0.86(0.035)	0.75(0.023)	0.86(0.036)	0.80(0.045)
(Non-linear)				
MILR	0.60(0.040)	0.62(0.038)	0.60(0.045)	0.63(0.041)
MILR (Gibbs)	0.60(0.045)	0.62(0.039)	0.59(0.048)	0.62(0.049)
MILR-LAM(100/0.25)	0.63(0.028)	0.59(0.036)	0.63(0.032)	0.60(0.049)
VGPMIL	0.76(0.038)	0.68(0.037)	0.75(0.035)	0.70(0.056)
GPMILR	0.79(0.020)	0.76(0.031)	**0.78**(0.018)	0.71(0.040)
GPMILR-LAM(10/0.15)	0.78(0.029)	0.72(0.032)	**0.78**(0.023)	**0.72**(0.046)

Table 4.2: Fitted and predicted average AUCs (Std) results of equal bag size (Mislabel).

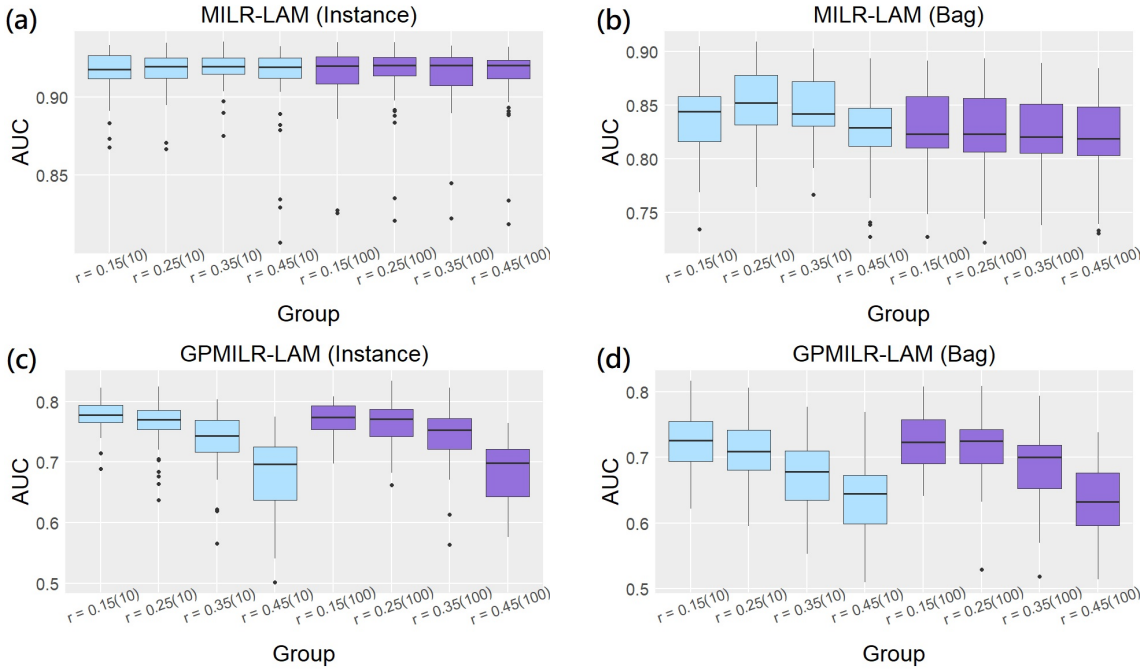


Figure 4.2: Equal bag size (Mislabel): predicted AUC results of instance and bag labels. Each plot has eight boxplots with eight different combinations of r and t. (a) and (b): operating MILR-LAM on the linear data. (c) and (d): operating GPMILR-LAM on the non-linear data.

doi:10.6342/NTU202401899

	Train		Test	
	instances	bags	instances	bags
(Linear)				
MILR	0.92(0.017)	0.73(0.033)	0.91(0.017)	0.73(0.045)
MILR (Gibbs)	0.91(0.018)	0.73(0.033)	0.91(0.018)	0.73(0.043)
MILR-LAM(10/0.25)	0.92(0.008)	0.77(0.034)	**0.92**(0.009)	**0.77**(0.038)
VGPMIL	0.90(0.016)	0.78(0.032)	0.89(0.016)	0.75(0.041)
GPMILR	0.79(0.029)	0.75(0.041)	0.79(0.028)	0.64(0.043)
GPMILR-LAM(10/0.25)	0.88(0.017)	0.78(0.032)	0.87(0.017)	0.74(0.042)
GPMILR-LAM(10/0.35)	0.88(0.020)	0.78(0.031)	0.87(0.022)	0.73(0.042)
(Non-linear)				
MILR	0.59(0.042)	0.65(0.039)	0.59(0.043)	0.61(0.040)
MILR (Gibbs)	0.59(0.053)	0.64(0.047)	0.58(0.055)	0.61(0.051)
MILR-LAM(100/0.35)	0.64(0.027)	0.60(0.044)	0.63(0.026)	0.58(0.044)
VGPMIL	0.78(0.033)	0.73(0.046)	0.78(0.031)	0.68(0.038)
GPMILR	0.79(0.018)	0.79(0.039)	**0.79**(0.019)	0.68(0.031)
GPMILR-LAM(10/0.15)	0.79(0.026)	0.76(0.037)	**0.79**(0.025)	**0.69**(0.036)

Table 4.3: Fitted and predicted average AUCs (Std) of equal bag size (Different rates).

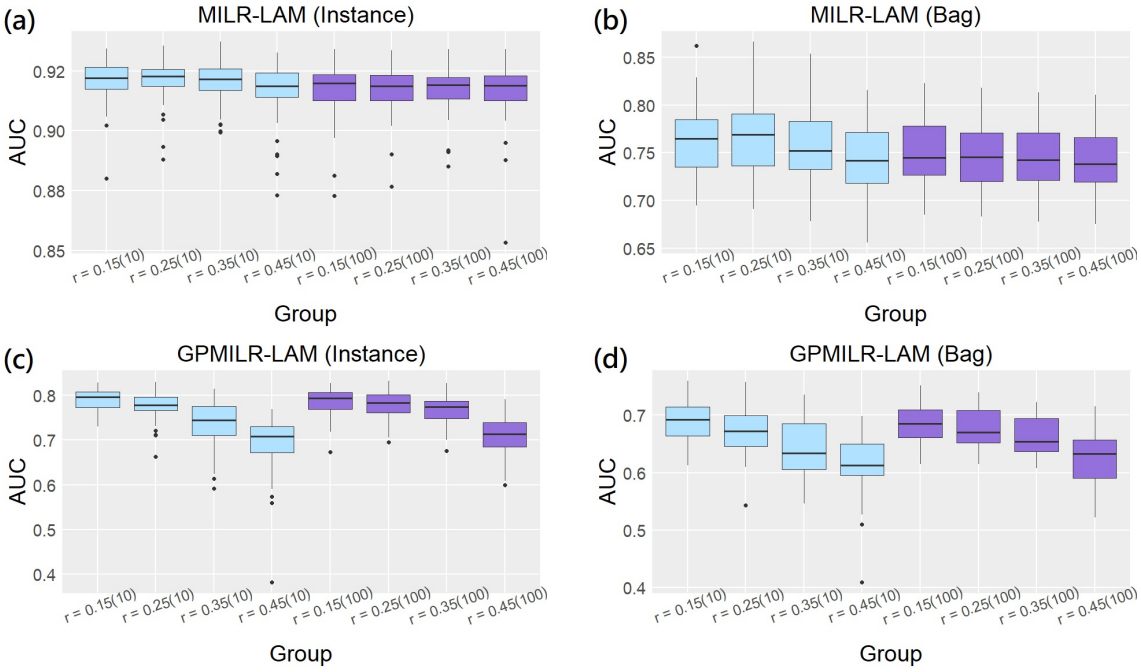


Figure 4.3: Equal bag size (Different rates): predicted AUC results of instance and bag labels. Each plot has eight boxplots with eight different combinations of r and t. (a) and (b): operating MILR-LAM on the linear data. (c) and (d): operating GPMILR-LAM on the non-linear data.

doi:10.6342/NTU202401899

4.1.2 Results of Unequal Bag Size

In the second part, we conduct simulations using a more complex MI data structure, which includes two different bag sizes. Table 4.4 (Mislabel) and Table 4.5 (Different rates) show the performance results. The outcomes of using the unequal bag size resemble those of using an equal one, but some distinctions exist. Explicitly, although benchmark models maintain the capabilities to classify instance labels correctly, they struggle with predicting bag labels, resulting in lower average AUCs. This situation is reasonable to observe in the models satisfying the MIL assumption, such as MILR and GPMILR. However, VGPMIL, which does not fully meet the MIL assumption, also returns poor bag prediction results. This is because the formulation of its bag likelihood has limitations when considering the MI data with different bag sizes. It is apparent that the MI data with various bag sizes does not easily influence the predicted performances of our proposed LAM.

	Train		Test	
	instances	bags	instances	bags
(Linear)				
MILR	0.92(0.010)	0.66(0.045)	**0.92**(0.011)	0.69(0.038)
MILR (Gibbs)	0.92(0.010)	0.66(0.045)	**0.92**(0.011)	0.69(0.040)
MILR-LAM(10/0.15)	0.92(0.008)	0.75(0.031)	**0.92**(0.008)	0.84(0.029)
MILR-LAM(10/0.25)	0.92(0.010)	0.75(0.031)	**0.92**(0.011)	**0.85**(0.030)
VGPMIL	0.88(0.024)	0.60(0.043)	0.88(0.023)	0.58(0.045)
GPMILR	0.80(0.024)	0.69(0.042)	0.78(0.025)	0.58(0.047)
GPMILR-LAM(100/0.35)	0.87(0.020)	0.94(0.014)	0.87(0.020)	0.78(0.036)
(Non-linear)				
MILR	0.60(0.045)	0.58(0.048)	0.60(0.042)	0.54(0.045)
MILR (Gibbs)	0.61(0.044)	0.57(0.048)	0.61(0.041)	0.53(0.046)
MILR-LAM(100/0.15)	0.63(0.037)	0.60(0.038)	0.63(0.036)	0.61(0.047)
VGPMIL	0.68(0.076)	0.56(0.051)	0.68(0.074)	0.52(0.056)
GPMILR	0.80(0.021)	0.71(0.041)	**0.79**(0.024)	0.61(0.048)
GPMILR-LAM(100/0.15)	0.80(0.019)	0.91(0.020)	0.78(0.025)	**0.73**(0.040)

Table 4.4: Fitted and predicted average AUCs (Std) of unequal bag size (Mislabel).

doi:10.6342/NTU202401899

	Train		Test	
	instances	bags	instances	bags
(Linear)				
MILR	0.92(0.010)	0.70(0.048)	**0.92**(0.009)	0.68(0.037)
MILR (Gibbs)	0.92(0.009)	0.70(0.048)	**0.92**(0.009)	0.68(0.038)
MILR-LAM(10/0.25)	0.92(0.009)	0.79(0.034)	**0.92**(0.008)	**0.78**(0.030)
VGPMIL	0.87(0.019)	0.61(0.040)	0.89(0.015)	0.59(0.037)
GPMILR	0.81(0.027)	0.71(0.040)	0.80(0.023)	0.60(0.038)
GPMILR-LAM(10/0.35)	0.87(0.022)	0.82(0.027)	0.87(0.016)	0.74(0.035)
(Non-linear)				
MILR	0.60(0.037)	0.59(0.047)	0.60(0.041)	0.56(0.039)
MILR (Gibbs)	0.60(0.041)	0.58(0.048)	0.60(0.045)	0.56(0.038)
MILR-LAM(100/0.35)	0.64(0.022)	0.61(0.032)	0.63(0.026)	0.59(0.045)
VGPMIL	0.73(0.056)	0.56(0.044)	0.72(0.056)	0.55(0.043)
GPMILR	0.80(0.016)	0.72(0.038)	**0.79**(0.018)	0.62(0.038)
GPMILR-LAM(100/0.15)	0.80(0.024)	0.90(0.019)	0.78(0.028)	**0.74**(0.038)

Table 4.5: Fitted and predicted average AUCs (Std) of unequal bag size (Different rates).

4.2 Impact of Different Bag Sizes on Model Performance

This simulation investigates four different simulated datasets, each having an equal bag size. We set the number of bags to be 50 and apply $\{5, 10, 20, 40\}$ to four datasets as their bag size, respectively. To keep it simple, we only consider linear MI data and ignore the non-linear case in this simulation. For generating simulated MI data, we utilize the true coefficients $\boldsymbol{\beta} = (-2, 3, 0.5)$ for the linear setting, and the threshold for defining the bag labels is set to be 0.35. When selecting the optimal t and r, we fix r to be 0.35 and only tune the hyperparameter $t = \{10, 100\}$ for LAM to reduce the computational workload. Each simulation case is replicated 10 times on every model, and we record their average AUCs. Lastly, 10 inducing points are used for the methods with GPs.

doi:10.6342/NTU202401899

4.2.1 Results

Based on Figure 4.4, we observe that average AUCs of methods involving the MIL assumption, such as MILR, MILR (Gibbs), and GPMILR, drop significantly as the number of bag size increases. However, the performance results of MILR-LAM and GPMILR-LAM decrease relatively more slowly as the bag size grows. VGPMIL also gains favorable outcomes when considering the fixed bag size for each MI dataset. Overall, our proposed LAM demonstrates confidence in maintaining the performance levels, regardless of the size of the bags.

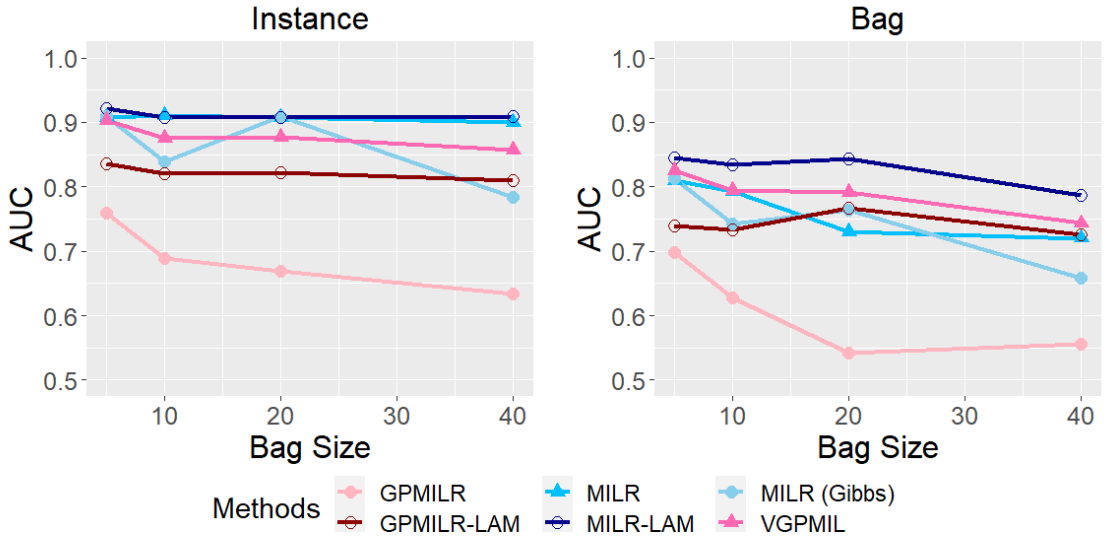


Figure 4.4: Predicted average AUCs of instance and bag labels based on various bag sizes.

4.3 Impact of Varying Threshold Values on Model Performance

In the last simulation, we evaluate the models' effectiveness by the following settings. Four studies are conducted by four different rates $\{0.1, 0.3, 0.6, 0.9\}$. The simulated dataset in each study contains 100 bags, with a fixed bag size of 10, and uses one of

doi:10.6342/NTU202401899

$\{0.1, 0.3, 0.6, 0.9\}$ as the threshold to determine the label of bags. Furthermore, we only consider linear MI data here, and the true coefficients $\boldsymbol{\beta}$ are set differently for each of the four studies to ensure the generated MI data have balanced bag labels. For the selection of hyperparameters of LAM, we only tune $t = \{10, 100\}$ and fix r to be the same as the threshold value of the MI data to save computational time. Additionally, we set the number of replications and inducing points to be the same as in simulation 2.

4.3.1 Results

We consider four different rates in the range from 0.1 to 0.9. As the rate value gets larger, the generated MI data violates the MIL assumption more severely. Based on Figure 4.5, the average AUCs for MILR, MILR (Gibbs), and GPMILR decrease when the rate increases. Nevertheless, MILR-LAM shows outstanding performance compared to other methods. Especially for the study utilizing a rate of 0.9, the difference in average AUCs becomes more apparent.

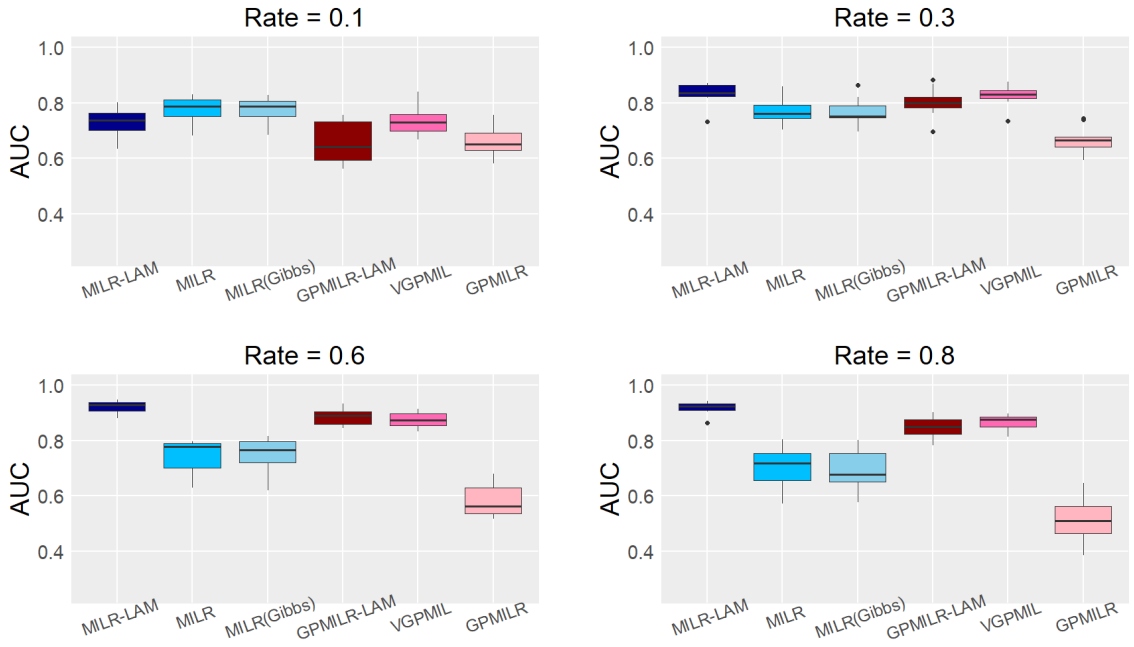


Figure 4.5: Predicted average AUCs of bag label based on four different rates.

doi:10.6342/NTU202401899

Chapter 5 Real Data Experiment

This chapter evaluates our models by using two datasets: Musk (Dietterich et al., 1997) and Mutagenesis (Srinivasan et al., 1994). Both datasets are used for drug activity prediction, a well-known task in MIL. The Musk data fully satisfies the MIL assumption and is often used as the benchmark data in MIL. Table 5.1 exhibits that each dataset is separated into two parts, and we will discuss their details in subsequent sections.

We operate the six algorithms mentioned in Chapter 4 on all real datasets. All settings remain the same as previously, except for choosing the number of inducing points, which varies based on dataset size. The last column of Table 5.1 shows the number of inducing points used in different datasets. We also normalize the training datasets before using them and apply similar scaling to the testing data. We conduct 5-fold cross-validation for the experiments. Furthermore, we use the cross-validation results to select the optimal t and r for LAM according to the combinations of $t = \{10, 100\}$ and $r = \{0.01, 0.1, 0.2, 0.5\}$.

	Total Instances	Bag Sizes	Bags (Pos/Neg)	Features	Inducing Points
Musk 1	476	2∼40	92 (47/45)	166	10
Musk 2	6598	1∼1044	102 (39/63)	166	30
Mutagenesis 1	10486	28∼88	188 (125/63)	7	50
Mutagenesis 2	2132	26∼86	42 (13/29)	7	10

Table 5.1: Descriptions of real datasets.

doi:10.6342/NTU202401899

5.1 Musk

Both Musk 1 and Musk 2 datasets aim to identify whether a molecule can produce a musky smell. The two Musk datasets contain molecules (bags) and their conformations (instances), as mentioned in Section 1.2. Also, each conformation is described by 166 features. According to Table 5.1, the main differences between Musk 1 and Musk 2 are the number of bag sizes and the total number of instances. The distributions of bag sizes of two datasets are shown in Figure 5.1. Although the range of bag sizes in Musk 2 extends from 1 to 1044, most bags are less than 100, with only a few exceeding 500. The bag sizes of Musk 1 are mostly less than 10. Besides, it is evident that the total number of instances in Musk 2 is much more than in Musk 1, so we consider more inducing points in the GP models.

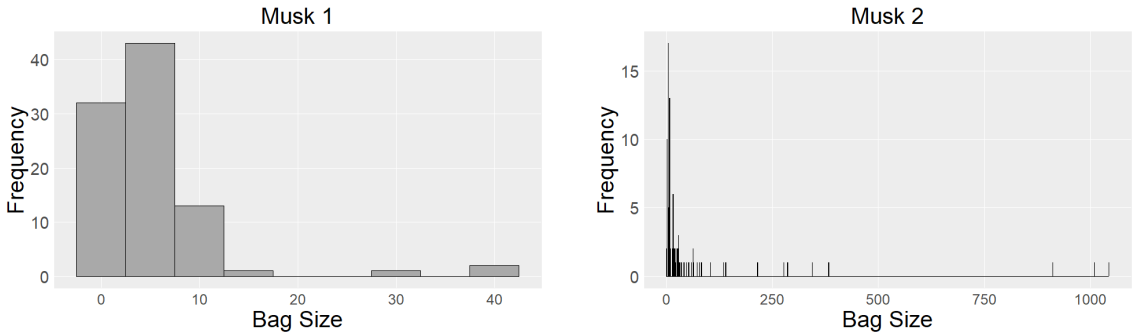


Figure 5.1: Histogram of the distribution of bag sizes in Musk 1 and Musk 2.

5.1.1 Results

We analyze the predicted results in Table 5.2 by considering linear and non-linear models. The linear models, including MILR-LAM and MILR, can perfectly capture the structures of two Musk datasets. In contrast, the non-linear models might be too complex to predict accurately. Therefore, we mainly focus on the linear models: MILR, MILR

doi:10.6342/NTU202401899

(Gibbs), and MILR-LAM. Since the two Musk data satisfy the MIL assumption, it is apparent that MILR, which fulfills the assumption, can perform well. MILR (Gibbs) has similar results to MILR because they use the same model and only differ in their inference methods. Our proposed MILR-LAM performs the best, particularly on Musk 1, which contains few samples. This result might indicate that MILR-LAM's performance is not significantly affected by the sample size in the Musk data.

Additionally, we explain some predicted processes in GPMILR-LAM. The original setting of length-scale is $\frac{1}{\sqrt{2\sqrt{166}}} \approx 0.2$. This value is too small, so it makes the model too complex to capture the correct data structure, which results in testing AUCs always equal to 0.5. Therefore, we conduct a larger length-scale value of $\sqrt{2}$ on both Musk 1 and Musk 2 datasets. Moreover, we examine the cross-validation process to decide the optimal value of t by testing smaller values, as larger values tend to overfit the data. The predicted AUC results of new settings are shown in Table 5.2.

	Musk 1			**Musk 2**	
	Train	Test		Train	Test
MILR	1.00	0.77	MILR	0.97	**0.82**
MILR (Gibbs)	1.00	0.76	MILR (Gibbs)	0.99	**0.82**
MILR-LAM(10/0.1)	0.95	**0.89**	MILR-LAM(100/0.1)	0.97	**0.82**
VGPMIL	0.78	0.63	VGPMIL	0.69	0.65
GPMILR	0.75	0.62	GPMILR	0.62	0.58
GPMILR-LAM(5/0.1)	1.00	0.66	GPMILR-LAM(3/0.1)	0.98	0.59

Table 5.2: Fitted and predicted AUC results of Musk 1 and Musk 2.

5.2 Mutagenesis

This data wants to identify the mutagenicity of molecules. There are a total of 230 molecules in the original Mutagenesis dataset. They are separated into Mutagenesis 1 and

doi:10.6342/NTU202401899

Mutagenesis 2, which include 188 and 42 molecules, respectively. As mentioned in Srinivasan et al. (1994), Mutagenesis 1 is more suitable for fitting statistical regression models than Mutagenesis 2. Using regression models to predict Mutagenesis 2 may present a significant challenge. Each bag represents a unique molecule in this dataset. If a molecule is mutagenic, then it has a positive bag label. Otherwise, it is assigned a negative one. An instance represents a molecule fragment instead of the whole molecule with different conformations. Each fragment is described by 7 features, which contain information about atoms, bonds, their types, and so on.

The major differences between Mutagenesis 1 and Mutagenesis 2 are the total number of instances and bags. Table 5.1 shows that Mutagenesis 1 is a larger dataset than Mutagenesis 2. Thus, we use more inducing points for GP models on Mutagenesis 1. Moreover, Figure 5.2 indicates the bag sizes of two datasets are roughly evenly distributed between 20 and 90. This situation suggests that bag sizes may not be the factor to affect the performance results between them.

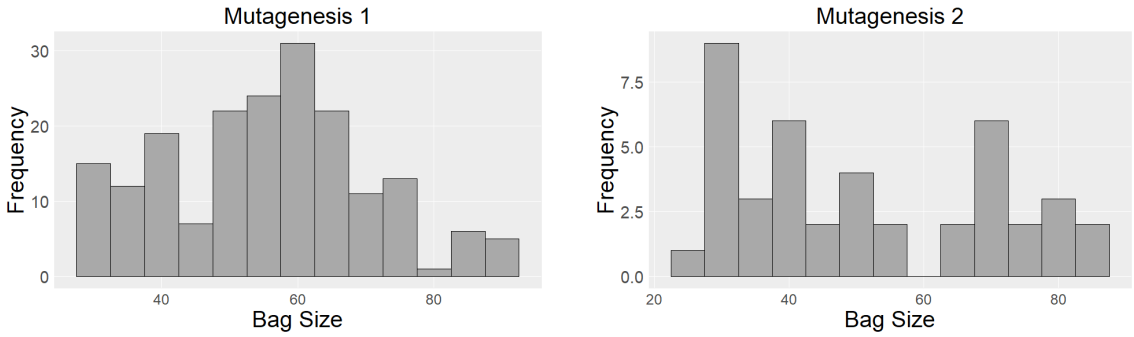


Figure 5.2: Histogram of the distribution of bag sizes in Mutagenesis 1 and Mutagenesis 2.

doi:10.6342/NTU202401899

5.2.1 Results

Table 5.3 demonstrate the performance results of Mutagenesis 1 and Mutagenesis 2. All the methods perform well on Mutagenesis 1 since their AUCs are all over 0.7, and MILR shows the best-predicted results. On the other hand, Mutagenesis 2 only contains 42 bags, which makes it more challenging for the model to predict accurately due to the small sample size. Moreover, its performance on regression models is poor according to Srinivasan et al. (1994). As a result, most methods achieve AUCs below 0.7, except for GPMILR-LAM and MILR-LAM. This result indicates that our proposed GPMILR-LAM and MILR-LAM can still perform well on the data with the structure not easily captured by traditional MI regression models. The situation where the predicted AUC is higher than the fitted one may be caused by the small number of bags. With a 5-fold cross-validation process, there are only approximately 8-9 bags to determine the predicted AUC.

	Mutagenesis 1			**Mutagenesis 2**	
	Train	Test		Train	Test
MILR	0.89	**0.90**	MILR	0.67	0.68
MILR (Gibbs)	0.88	0.87	MILR (Gibbs)	0.78	0.68
MILR-LAM(100/0.01)	0.80	0.77	MILR-LAM(100/0.1)	0.76	0.84
VGPMIL	0.72	0.71	VGPMIL	0.64	0.44
GPMILR	0.87	0.74	GPMILR	0.80	0.64
GPMILR-LAM(100/0.5)	0.89	0.75	GPMILR-LAM(10/0.5)	0.76	**0.88**

Table 5.3: Fitted and predicted AUC results of Mutagenesis 1 and Mutagenesis 2.

doi:10.6342/NTU202401899

doi:10.6342/NTU202401899

Chapter 6 Conclusion

In summary, we propose the Logistic Aggregation Model (LAM), which relaxes the strict MIL assumption presented in many existing MIL models. We use the Gibbs sampling approach with Pólya-Gamma augmentation to infer models. Furthermore, our LAM's performance is evaluated using both simulated and real datasets. Nevertheless, there are some limitations in this thesis, which we discuss below:

1. We only apply LAM to regression models, specifically MILR and GPMILR. These two MIL models in IS possess the specific instance and bag models. Further research can investigate applying the LAM to different kinds of MIL models in IS, such as mi-SVM (Andrews et al., 2002), as mentioned previously, to enhance the potential applicability of LAM.

2. Although we provide an explanation of hyperparameters t and r, we do not establish an exact procedure for choosing them. We only randomly select some candidates and determine the best ones based on the highest predicted AUC result. It will be beneficial to develop systematic rules for selecting the optimal t and r in the future, such as using Empirical Bayes methods. Specifically, we may adapt the estimated instance labels from MILR and then calculate the estimated average rate of positive instances in a bag. This estimated average rate can serve as a baseline

doi:10.6342/NTU202401899

for determining the threshold value in LAM.

3. This thesis only uses drug activity datasets to examine our LAM. Future works should explore the application of LAM to MI data in other fields, such as medical diagnosis or text classification. These investigations can provide more informative insights into the capability and utility of LAM.

doi:10.6342/NTU202401899

References

Ali, K. and Saenko, K. (2014). Confidence-rated multiple instance boosting for object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

Amores, J. (2013). Multiple instance classification: Review, taxonomy and comparative study. Artificial Intelligence, 201:81–105.

Andrews, S., Tsochantaridis, I., and Hofmann, T. (2002). Support vector machines for multiple-instance learning. In Advances in Neural Information Processing Systems, volume 15.

Carbonneau, M.-A., Cheplygina, V., Granger, E., and Gagnon, G. (2018). Multiple instance learning: A survey of problem characteristics and applications. Pattern Recognition, 77:329–353.

Chen, P.-Y., Chen, C.-C., Yang, C.-H., Chang, S.-M., and Lee, K.-J. (2017). milr: Multiple-instance logistic regression with lasso penalty. R Journal, 9:446–457.

Chen, Y., Bi, J., and Wang, J. (2007). Miles: Multiple-instance learning via embedded instance selection. IEEE transactions on pattern analysis and machine intelligence, 28:1931–47.

doi:10.6342/NTU202401899

Dietterich, T. G., Lathrop, R. H., and Lozano-Pérez, T. (1997). Solving the multiple instance problem with axis-parallel rectangles. Artificial Intelligence, 89(1):31–71.

Gärtner, T., Flach, P. A., Kowalczyk, A., and Smola, A. J. (2002). Multi-instance kernels. In Proceedings of the Nineteenth International Conference on Machine Learning, ICML '02, page 179–186. Morgan Kaufmann Publishers Inc.

Haußmann, M., Hamprecht, F. A., and Kandemir, M. (2017). Variational Bayesian multiple instance learning with Gaussian processes. In 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 810–819.

Ilse, M., Tomczak, J., and Welling, M. (2018). Attention-based deep multiple instance learning. In Proceedings of the 35th International Conference on Machine Learning, volume 80 of Proceedings of Machine Learning Research, pages 2127–2136.

Kandemir, M., Haußmann, M., Diego, F., Rajamani, K., Laak, J., and Hamprecht, F. (2016). Variational weakly supervised Gaussian processes. In British Machine Vision Conference.

Kandemir, M., Zhang, C., and Hamprecht, F. A. (2014). Empowering multiple instance histopathology cancer diagnosis by cell graphs. In MICCAI. Proceedings, volume 8674, pages 228–235. Springer. 1.

Ko, K. H., Jang, G., Park, K., and Kim, K. (2012). GPR-based landmine detection and identification using multiple features. International Journal of Antennas and Propagation, 2012.

Li, W. and Vasconcelos, N. (2015). Multiple instance learning for soft bags via top instances. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

doi:10.6342/NTU202401899

Maron, O. and Lozano-Pérez, T. (1997). A framework for multiple-instance learning. In Advances in Neural Information Processing Systems, volume 10.

Maron, O. and Ratan, A. L. (1998). Multiple-instance learning for natural scene classification. In International Conference on Machine Learning.

Polson, N. G., Scott, J. G., and Windle, J. (2013). Bayesian inference for logistic models using Pólya–Gamma latent variables. Journal of the American Statistical Association, 108(504):1339–1349.

Popescu, M. and Mahnot, A. (2012). Early illness recognition using in-home monitoring sensors and multiple instance learning. Methods of information in medicine, 51:359–67.

Rasmussen, C. E. and Williams, C. K. I. (2005). Gaussian Processes for Machine Learning. The MIT Press.

Raykar, V. C., Krishnapuram, B., Bi, J., Dundar, M., and Rao, R. B. (2008). Bayesian multiple instance learning: automatic feature selection and inductive transfer. In Proceedings of the 25th International Conference on Machine Learning, ICML '08, page 808–815.

Srinivasan, A., Muggleton, S., King, R. D., and Sternberg, M. J. E. (1994). Mutagenesis: ILP experiments in a non-determinate biological domain. In Proceedings of the Fourth Inductive Logic Programming Workshop.

Wang, F. and Pinar, A. (2021). The multiple instance learning Gaussian process probit model. In Proceedings of The 24th International Conference on Artificial Intelligence and Statistics, volume 130 of Proceedings of Machine Learning Research, pages 3034–3042.

doi:10.6342/NTU202401899

Wang, J. and Zucker, J.-D. (2000). Solving the multiple-instance problem: A lazy learning approach. In International Conference on Machine Learning.

Zhang, J., Marszałek, M., Lazebnik, S., and Schmid, C. (2007). Local features and kernels for classification of texture and object categories: A comprehensive study. International Journal of Computer Vision, 73(2):213–238.

Zhou, Z.-H., Sun, Y.-Y., and Li, Y.-F. (2008). Multi-instance learning by treating instances as non-i.i.d. samples. In International Conference on Machine Learning.

doi:10.6342/NTU202401899