OCCUPANCY ESTIMATION AND ACTIVITY RECOGNITION IN SMART BUILDINGS USING MIXTURE-BASED PREDICTIVE DISTRIBUTIONS

Jiaxun Guo

A thesis
in
The Department
of
Concordia Institute for Information Systems Engineering

Presented in Partial Fulfillment of the Requirements
For the Degree of Master of Applied Science
(Information Systems Security)at
Concordia University
Montréal, Québec, Canada

December 2021

© Jiaxun Guo, 2022

CONCORDIA UNIVERSITY

School of Graduate Studies

This is to certify that the thesis prepared

By: **Jiaxun Guo**

Entitled: **Occupancy Estimation and Activity Recognition in Smart Buildings using Mixture-Based Predictive Distributions**

and submitted in partial fulfillment of the requirements for the degree of

Master of Applied Science
(Information Systems Security)

complies with the regulations of this University and meets the accepted standards with respect to originality and quality.

Signed by the final examining commitee:

Dr. Walter Lucia ______________________ Chair
Dr. Walter Lucia ______________________ Examiner
Dr. Yong Zeng ______________________ Examiner
Dr. Nizar Bouguila ______________________ Supervisor
Dr. Manar Amayri ______________________ Co-supervisor
Dr. Wentao Fan ______________________ Co-supervisor

Approved ______________________
Dr. Mohammad Mannan, Graduate Program Director

__________ 20 ______ ______________________
Dr. Mourad Debbabi, Dean
Faculty of Engineering and Computer Science

Abstract

Occupancy Estimation and Activity Recognition in Smart Buildings using Mixture-Based Predictive Distributions

Jiaxun Guo

Labeled data is a necessary part of modern computer science, such as machine learning and deep learning. In that context, large amount of labeled training data is required. However, collecting of labeled data as a crucial step is time consuming, error prone and often requires people involvement. On the other hand, imbalanced data is also a challenge for classification approaches. Most approaches simply predict the majority class in all cases.

In this work, we proposed several frameworks about mixture models based predictive distribution. In the case of small training data, predictive distribution is data-driven, which can take advantage of the existing training data at its maximum and don't need many labeled data. The flexibility and adaptability of Dirichlet family distribution as mixture models further improve classification ability of frameworks.

Generalized inverted Dirichlet (GID), inverted Dirichlet (ID) and generalized Dirichlet (GD) are used in this work with predictive distribution to do classification. GID-based predictive distribution has an obvious increase for activity recognition compared with the approach of global variational inference using small training data. ID-based predictive distribution with over-sampling is applied in occupancy estimation. More synthetic data are sampling for small classes. The total accuracy is improved in the end. An occupancy estimation framework is presented based on interactive learning and predictive distribution of GD. This framework can find the most informative unlabeled data and interact with users to get the true label. New labeled data are added in data store to further improve the performance of classification.

Acknowledgments

First of all, I would like to express my sincere gratitude to my supervisor, Dr. Nizar Bouguila. In the past two years, he provided comprehensive support to help me grow into a qualified master student. When I faced the dual challenges of new language environment and learning environment at the beginning, he gave me enough wise guidance and time to adapt. When I doubt my ability, he always gives me encouragement. Besides, my admiration for him is not only academic, but also personality. Actually, I just wanted to get a better job by getting a master's degree after my undergraduate program. Because of him, I find the passion of research and have decided to continue studying as a PhD student.

I am also thankful to my co-supervisors, Dr. Manar Amayri and Dr. Wentao Fan. During the epidemic, they gave me a lot of advice and support remotely.

I want to express appreciation to my lab mates; Fatma Najar, Nuha Zamzami, Samr Ali, Fahdah, Kamal, Oumayma, Ornela and Omar. They not only gave me a lot of support and motivation, but also gave me an environment like home.

I would like to offer my special thanks to one of my best friends, MSc. Yunke Chen, for emotionally supporting during my hard time.

Last but not least, I would like to thank my parents. There is no words that can show my appreciation and love to them. They always support me, believe me and are my strong backing.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

Smart buildings and energy management systems have been the topic of extensive research recently, from different fields [4–8], which has been triggered by the growing adaptable sensor technologies. The main goal has been providing practical solutions to reduce energy consumption considering that buildings consume around 40% of total energy in the world [9]. To achieve this goal, building occupancy information and occupants activities recognition are key components [2, 10, 11]. Indeed, information about the occupancy or the activities could for instance determine the operation time of HVAC (Heating, Ventilation, and Air Conditioning) system in a given building [12–15].

Many approaches have been proposed in the past for occupancy estimation and activities recognition using data harvested from buildings' sensors (e.g. PIR, CO2, etc.) [16]. Machine learning (ML) techniques are mainly fueled by the recent tremendous growth of Internet of Things (IoT) [17–19] and the recent remarkable efforts to go "energy-efficient" [20–26]. Identifying and detecting a given activity depend on sensors' information processing [27, 28]. Most of researches have focused on developing activity recognition solutions using inertial signals [29–34] in the recent years. In order to reach their full potential, machine learning techniques need a large amount of balanced training data. Unfortunately, in the majority of real-life scenarios only a small training set is available which prevents using data hungry techniques. Such large and balanced training data are hard and expensive to obtain notably in applications related to smart buildings where only inhabitants, whose involvements is generally hard to obtain (because of privacy concerns, for instance), can provide reliable labels [35, 36]. The approaches have been proposed in the past to tackle that problem by mainly increasing the number of labeled data using techniques such as active learning or by deploying training data from other domains using transfer learning. For instance, the authors in [37] propose a cluster based active learning model based on K-Means and the authors in [38] propose a transfer learning approach based on a hierarchical Bayesian method.

Unlike previous approaches, we propose a learning technique based on a data-driven predictive distribution [39,40]. The main motivation of considering the predictive distribution, as compared to the commonly used point estimate methods [41–43] (e.g. maximum a posteriori, variational Bayes, expectation propagation), is that it leads to more reliable results when calculating the predictive likelihood of unseen data especially when the training data is small. Indeed, it is well-known that when training data is small, point estimate leads to estimated parameters with large variance and then cause uncertainty and unreliability [44]. The predictive density of a new vector $\mathbf{x}$ given the training data $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_N]$ is [1] [48, 49]

$$f(\mathbf{x}|\mathbf{X}) = \int f(\mathbf{x}|\boldsymbol{\theta}) f(\boldsymbol{\theta}|\mathbf{X}) d\boldsymbol{\theta} \tag{1.1}$$

where $\boldsymbol{\theta}$, the parameters of both likelihood function $f(\mathbf{x}|\boldsymbol{\theta})$ and posterior distribution $f(\boldsymbol{\theta}|\mathbf{X})$, should be learned from $\mathbf{X}$ via Bayesian inference. A crucial problem, in predictive modeling, is the choice of the statistical distribution to model the data. The Gaussian mixture model has been widely used in many recent applications (see, for instance, [40, 50–53]. However, not all real-life data satisfy the Gaussian assumption which should be dedicated mainly to model symmetric patterns [54]. This is especially true for features extracted from smart buildings sensors and which are generally semi-bounded [55]. In this context, mixtures of Dirichlet family distributions have been extensively studied recently and have shown state of the art results in a variety of applications from different domains (e.g. computer vision, pattern recognition, health

[1] It is noteworthy that when the training set become sufficiently large, the posterior distribution variance decreases and then the predictive distribution, which can be viewed as an average over model's parameters [45], could be approximated by $f(\mathbf{x}|\hat{\theta})$, where $\hat{\theta}$ is a point estimate (obtained via maximum a posteriori or expectation propagation or variational Bayes, for instance) [46, 47] .

informatics, image processing, etc.). Thus, we consider these mixture models, as likelihood functions, in our work to model the data that shall be used for prediction.

1.2 Contributions

The main objective of this thesis is to study predictive distributions based on Dirichlet family mixture models. Indeed, we shall see that this approach can be easily integrated with other frameworks and modules to improve further their performance. The contributions are listed as follows:

1. **A Generalized Inverted Dirichlet-Based Predictive Model for Activity Recognition using Small Training Data**
   We present an elegant principled statistical framework for predictive modeling based on the generalized inverted Dirichlet (GID) distribution [56,57] and a local variational inference. It demonstrates a good efficiency in classification when only small training data are provided. Activity recognition as a challenging application is considered to validate the effectiveness of the framework.

2. **Occupancy Estimation in Smart Buildings using Predictive Modeling in Imbalanced Domains**
   We introduce a model based on the predictive distribution of the inverted Dirichlet (ID) mixture model [58] to tackle the challenging problem of occupancy estimation in smart buildings. This model is mainly motivated by its prediction ability when only small training data are available. In order to tackle the imbalance problem in occupancy estimation, we develop a data pre-processing approach based on over-sampling via the generation of new synthetic examples. We are mainly motivated by the fact that synthesizing new data has well documented advantages such as reducing over-fitting risks and improving generalization capabilities [59, 60].

3. **A Hybrid of Interactive Learning and Predictive Modeling For Occupancy Estimation in Smart Buildings**
   We develop an accurate approximation to the predictive distribution of the generalized Dirichlet (GD) mixture model [61–63] as classifier. With the support of interactive learning module, the novel framework can label data automatically,

which increases the precision of the learning model by expending the training set and provides good results even when starting with a very small training data set.

1.3 Thesis Overview

- Chapter 1 introduces the motivations of our research work and contributions.
- Chapter 2 presents the details of the GID based predictive distribution and shows how the model works for activity recognition under small training data.
- In chapter 3, a predictive distribution of ID mixture model is presented with over-sampling module. According to our experiments that concerns occupancy estimation, this framework performs well with imbalanced training data and also provides a good efficiency on small training data.
- In chapter 4, a framework of GD based predictive distribution and interactive learning is discussed and validated via occupancy estimation application.
- Chapter 5 briefly concludes our contributions and discusses future works.

Chapter 2

A Generalized Inverted Dirichlet-Based Predictive Model for Activity Recognition using Small Training Data

In this chapter, we propose a predictive model based on generalized inverted Dirichlet (GID) mixture model. We show via extensive experiments that this model has a good performance for activity recognition under small training data.

2.1 Predictive Model

In this section, we first present briefly the generalized inverted Dirichlet mixture model. Then, its predictive distribution will be approximated by variational inference.

2.1.1 Generalized Inverted Dirichlet Mixture Model

Let M denotes the number of different components. Assume that a D-dimensional positive vector $\vec{Y} = (Y_1, \cdots, Y_D)$ follows a finite mixture model of generalized Inverted Dirichlet (GID) Distributions denoted by a common probability density function $p(\vec{Y} \mid \vec{\pi}, \vec{\alpha}, \vec{\beta})$ such that

$$p(\vec{Y} \mid \vec{\pi}, \vec{\alpha}, \vec{\beta}) = \sum_{j=1}^{M} \pi_j \mathrm{GID}\left(\vec{Y} \mid \vec{\alpha}_j, \vec{\beta}_j\right) \tag{2.1}$$

where $\vec{\alpha} = \{\vec{\alpha}_1, \ldots, \vec{\alpha}_M\}$, $\vec{\beta} = \left\{\vec{\beta}_1, \ldots, \vec{\beta}_M\right\}$, $\vec{\alpha}_j$ and $\vec{\beta}_j$ are the parameters of the GID distribution representing component j, where $\vec{\alpha}_j = (\alpha_{j1}, \cdots, \alpha_{jD})$ and $\vec{\beta}_j = (\beta_{j1}, \cdots, \beta_{jD})$. $\vec{\pi} = (\pi_1, \cdots, \pi_M)$ denotes the mixing weights, such that$\sum_{j=1}^{M} \pi_j = 1$ and $\pi_j > 0$, $j = 1, \ldots, M$. $\text{GID}\left(\vec{Y} \mid \vec{\alpha}_j, \vec{\beta}_j\right)$ is a GID distribution representing component j with parameters $\vec{\alpha}_j$ and $\vec{\beta}_j$ and is defined by [64]

$$\text{GID}\left(\vec{Y} \mid \vec{\alpha}_j, \vec{\beta}_j\right) = \prod_{l=1}^{D} \frac{\Gamma\left(\alpha_{jl} + \beta_{jl}\right)}{\Gamma\left(\alpha_{jl}\right)\Gamma\left(\beta_{jl}\right)} \frac{Y_l^{\alpha_{jl}-1}}{\left(1 + \sum_{k=1}^{l} Y_k\right)^{\gamma_{jl}}} \tag{2.2}$$

where $\alpha_{jl} > 0$, $\beta_{jl} > 0$.

Let $\mathcal{Y} = \left\{\vec{Y}_1, \ldots, \vec{Y}_N\right\}$ be a set of N independent identically distributed vectors taken from our mixture model. According to the Bayes' theorem, the probability that the vector $\vec{Y}_i$ is from component j (also called *responsibilities* of each mixture component j in generating each data sample $\vec{Y}_i$) can be written as

$$p\left(j \mid \vec{Y}_i\right) \propto \pi_j GID\left(\vec{Y}_i \mid \vec{\alpha}_j, \vec{\beta}_j\right) \tag{2.3}$$

We define $\gamma_{jl} = \beta_{jl} + \alpha_{jl+1} - \beta_{jl+1}$ for $l = 1, \cdots, D$, with $\beta_{jD+1} = 0$. After some mathematical manipulations [65], the responsibilities can be factorized as

$$p\left(j \mid \vec{Y}_i\right) \propto \pi_j \prod_{l=1}^{D} \text{iBeta}\left(X_{il} \mid \alpha_{jl}, \beta_{jl}\right) \tag{2.4}$$

where $X_{i1} = Y_{i1}$ and $X_{il} = Y_{il}/(1 + \sum_{k=1}^{l-1} Y_{ik})$ for $l > 1$ and $\text{iBeta}\left(X_{il} \mid \alpha_{jl}, \beta_{jl}\right)$ is an inverted Beta distribution with parameters $(\alpha_{jl}, \beta_{jl})$:

$$\text{iBeta}\left(X_{il} \mid \alpha_{jl}, \beta_{jl}\right) = \frac{\Gamma(\alpha_{jl} + \beta_{jl})}{\Gamma(\alpha_{jl})\Gamma(\beta_{jl})} X_{il}^{\alpha_{jl}-1}(1 + X_{il})^{-(\alpha_{jl}+\beta_{jl})} \tag{2.5}$$

where $\Gamma(z) = \int_0^\infty x^{z-1}e^{-x}dx$. Thus, the mixture model of the finite GID distribution underlying dataset $\mathcal{Y}$ is the same as that underlying $\mathcal{X} = \left\{\vec{X}_1, \ldots, \vec{X}_N\right\}$, where $\vec{X}_i = \left\{\vec{X}_{i1}, \ldots, \vec{X}_{iD}\right\}, i = 1, \ldots, N$, using the following clustering structure with conditionally independent features

$$p\left(\vec{X}_i \mid \vec{\pi}, \vec{\alpha}, \vec{\beta}\right) = \sum_{j=1}^{M} \pi_j \prod_{l=1}^{D} \mathrm{iBeta}\left(X_{il} \mid \alpha_{jl}, \beta_{jl}\right) \tag{2.6}$$

The formal conjugate prior distribution of the Beta distribution [49] does not have a closed form. Thus, we consider in our work a tractable approximation to the conjugate prior using a product of two independent Gamma distributions as previously used in [65] in the context of a global variational inference (GVI) framework

$$\begin{aligned} f(\alpha_{jl}, \beta_{jl}) &\approx \mathrm{Gam}\left(\alpha_{jl}; a_0, b_0\right) \times \mathrm{Gam}\left(\beta_{jl}; c_0, d_0\right) \\ &= \frac{b_0^{a_0}}{\Gamma\left(a_0\right)} \alpha_{jl}^{a_0-1} e^{-b_0\alpha_{jl}} \times \frac{d_0^{c_0}}{\Gamma\left(c_0\right)} \beta_{jl}^{c_0-1} e^{-d_0\beta_{jl}} \end{aligned} \tag{2.7}$$

With available data $\mathcal{X}_l = \{X_{1l}, X_{2l}, \ldots, X_{Nl}\}$, the hyperparameters ($a^*, b^*, c^*$ and d^*) of the posterior distribution can be easily obtained by variational Bayes estimation as detailed in [65]. The posterior distribution could be approximated by a product of two independent Gamma distribution as

$$\begin{aligned} f(\alpha_{jl}, \beta_{jl} \mid \mathcal{X}_l) &\approx \mathrm{Gam}\left(\alpha_{jl}; a^*, b^*\right) \times \mathrm{Gam}\left(\beta_{jl}; c^*, d^*\right) \\ &= \frac{{b^*}^{a^*}}{\Gamma\left(a^*\right)} \alpha_{jl}^{a^*-1} e^{-b^*\alpha_{jl}} \times \frac{{d^*}^{c^*}}{\Gamma\left(c^*\right)} \beta_{jl}^{c^*-1} e^{-d^*\beta_{jl}} \end{aligned} \tag{2.8}$$

2.1.2 Predictive Distribution of the Mixture Model

The predictive distribution can assess the uncertainty of a new coming observation with respect to the existing dataset. Let $\vec{Y}_i$ be that new observation independent from the existing $\mathcal{Y}$ which is assumed to be generated from $GID\left(\vec{Y}_i \mid \vec{\alpha}_j, \vec{\beta}_j\right)$. Using the transformation presented after Eq.2.4, the obtained new observation $\vec{X}_i$ follows a product of inverted Beta distributions. The predictive distribution of $\vec{X}_i$ given $\mathcal{X}$ is

$$f(\vec{X}_i \mid \mathcal{X}) = \int_0^\infty \int_0^\infty \prod_{l=1}^{D} \left[\mathrm{iBeta}(X_{il} \mid \alpha_{jl}, \beta_{jl}) f(\alpha_{jl}, \beta_{jl} \mid \mathcal{X}_l)\right] d\vec{\alpha}_j d\vec{\beta}_j \tag{2.9}$$

With the analytically tractable posterior distribution in Eq.2.8 and the function of the inverted Beta distribution (Eq.2.5), we can extend this predictive distribution as

$$\begin{aligned} f(\vec{X}_i \mid \mathcal{X}) &\approx \int_0^\infty \int_0^\infty \prod_{l=1}^{D} \left[\text{iBeta}(X_{il} \mid \alpha_{jl}, \beta_{jl}) \times \frac{{b^*}^{a^*}}{\Gamma\left(a^*\right)} \alpha_{jl}^{a^*-1} e^{-b^* \alpha_{jl}} \frac{{d^*}^{c^*}}{\Gamma\left(c^*\right)} \beta_{jl}^{c^*-1} e^{-d^* \beta_{jl}} \right] d\vec{\alpha}_j d\vec{\beta}_j \\ &= \prod_{l=1}^{D} \left[\frac{1}{X_{il}} \frac{{b^*}^{a^*}}{\Gamma\left(a^*\right)} \frac{{d^*}^{c^*}}{\Gamma\left(c^*\right)} \right] \\ &\quad \times \int_0^\infty \int_0^\infty \prod_{l=1}^{D} \left[\frac{\Gamma(\alpha_{jl} + \beta_{jl})}{\Gamma(\alpha_{jl})\Gamma(\beta_{jl})} X_{il}^{\alpha_{jl}} (1 + X_{il})^{-(\alpha_{jl}+\beta_{jl})} \alpha_{jl}^{a^*-1} e^{-b^* \alpha_{jl}} \beta_{jl}^{c^*-1} e^{-d^* \beta_{jl}} \right] d\vec{\alpha}_j d\vec{\beta}_j \end{aligned} \tag{2.10}$$

$$\begin{aligned} f(\vec{X}_i \mid \mathcal{X}) &\leq f_{\text{upp}}(\vec{X}_i \mid \mathcal{X}) \\ &= \prod_{l=1}^{D} \left[\frac{1}{X_{il}} \frac{b^{*a^*}}{\Gamma\left(a^*\right)} \frac{d^{*c^*}}{\Gamma\left(c^*\right)} \frac{\Gamma\left(\alpha_0 + \beta_0\right)}{\Gamma\left(\alpha_0\right)\Gamma\left(\beta_0\right)} e^{-\alpha_0[\psi(\alpha_0+\beta_0) - \psi(\alpha_0)] - \beta_0[\psi(\alpha_0+\beta_0) - \psi(\beta_0)]} \right] \\ &\quad \times \int_0^\infty \int_0^\infty \prod_{l=1}^{D} \Big[e^{\alpha_{jl}[\psi(\alpha_0+\beta_0) - \psi(\alpha_0)] + \beta_{jl}[\psi(\alpha_0+\beta_0) - \psi(\beta_0)]} \\ &\qquad\qquad X_{il}^{\alpha_{jl}} (1 + X_{il})^{-\alpha_{jl}} (1 + X_{il})^{-\beta_{jl}} \alpha_{jl}^{a^*-1} e^{-b^* \alpha_{jl}} \beta_{jl}^{c^*-1} e^{-d^* \beta_{jl}} \Big] d\vec{\alpha}_j d\vec{\beta}_j \\ &= \prod_{l=1}^{D} \left[\frac{1}{X_{il}} \frac{b^{*a^*}}{\Gamma\left(a^*\right)} \frac{d^{*c^*}}{\Gamma\left(c^*\right)} \frac{\Gamma\left(\alpha_0 + \beta_0\right)}{\Gamma\left(\alpha_0\right)\Gamma\left(\beta_0\right)} e^{-\alpha_0[\psi(\alpha_0+\beta_0) - \psi(\alpha_0)] - \beta_0[\psi(\alpha_0+\beta_0) - \psi(\beta_0)]} \right] \\ &\quad \times \prod_{l=1}^{D} \left\{ \int_0^\infty e^{-\alpha_{jl} \underbrace{\left[b^* - \ln X_{il} + \ln(1 + X_{il}) - \psi\left(\alpha_0 + \beta_0\right) + \psi\left(\alpha_0\right)\right]}_{g(X_{il}, \alpha_0, \beta_0)}} \alpha_{jl}^{a^*-1} d\alpha_{jl} \right\} \\ &\quad \times \prod_{l=1}^{D} \left\{ \int_0^\infty e^{-\beta_{jl} \underbrace{\left[d^* + \ln(1 + X_{il}) - \psi\left(\alpha_0 + \beta_0\right) + \psi\left(\beta_0\right)\right]}_{h(X_{il}, \alpha_0, \beta_0)}} \beta_{jl}^{c^*-1} d\beta_{jl} \right\} \end{aligned} \tag{2.11}$$

Eq.2.10 involves the Inverse Beta function which logarithm has been proved to be concave [66]. Using that concavity property and first order Taylor expansion, we can obtain the following inequality [67]

$$\frac{\Gamma(\alpha + \beta)}{\Gamma(\alpha)\Gamma(\beta)} \leq \frac{\Gamma\left(\alpha_0 + \beta_0\right)}{\Gamma\left(\alpha_0\right)\Gamma\left(\beta_0\right)} \times e^{[\psi(\alpha_0+\beta_0) - \psi(\alpha_0)](\alpha - \alpha_0) + [\psi(\alpha_0+\beta_0) - \psi(\beta_0)](\beta - \beta_0)} \tag{2.12}$$

where $\psi(\cdot)$ is the digamma function defined as $\psi(\cdot) = \partial \ln \Gamma(x) / \partial x$.

Using Eq.2.12 in Eq.2.10, we can find an upper bound for the predictive distribution as shown in Eq.2.11 by a local variational inference (LVI) method [67]. Compared with the global variational inference [65] which approximates all the model's variables, LVI is considered as a ’local’ approach to approximate a subset of variables [49]. In Eq.2.11, the integrand in each integration is a Gamma distribution. To simplify the predictive distribution, these integrations can be replaced by

$$\begin{aligned} &\int_0^\infty e^{-\alpha_{jl} g(X_{il},\alpha_0,\beta_0)} \alpha_{jl}^{a^*-1} d\alpha_{jl} \\ &\quad = \begin{cases} \frac{\Gamma(a^*)}{g(X_{il},\alpha_0,\beta_0)^{a^*}} & g\left(X_{il},\alpha_0,\beta_0\right) > 0 \\ \infty & g\left(X_{il},\alpha_0,\beta_0\right) \leq 0 \end{cases} \\ &\int_0^\infty e^{-\beta_{jl} h(X_{il},\alpha_0,\beta_0)} \beta_{jl}^{c^*-1} d\beta_{jl} \\ &\quad = \begin{cases} \frac{\Gamma(c^*)}{h(X_{il},\alpha_0,\beta_0)^{c^*}} & h\left(X_{il},\alpha_0,\beta_0\right) > 0 \\ \infty & h\left(X_{il},\alpha_0,\beta_0\right) \leq 0 \end{cases} \end{aligned} \tag{2.13}$$

If $g\left(X_{il},\alpha_0,\beta_0\right) > 0$ and $h\left(X_{il},\alpha_0,\beta_0\right) > 0$, we obtain a closed-form upper bound for the predictive distribution:

$$\begin{aligned} &f_{\text{upp}}(\vec{X}_i \mid \mathcal{X}) \\ &\quad = \prod_{l=1}^{D} \left[\frac{1}{X_{il}} \left[\frac{b^*}{g\left(X_{il},\alpha_0,\beta_0\right)} \right]^{a^*} \left[\frac{d^*}{h\left(X_{il},\alpha_0,\beta_0\right)} \right]^{c^*} \right. \\ &\quad \left. \times \frac{\Gamma\left(\alpha_0+\beta_0\right)}{\Gamma\left(\alpha_0\right)\Gamma\left(\beta_0\right)} e^{-\alpha_0[\psi(\alpha_0+\beta_0)-\psi(\alpha_0)]-\beta_0[\psi(\alpha_0+\beta_0)-\psi(\beta_0)]} \right] \end{aligned} \tag{2.14}$$

The upper bound is just a function of α_0, β_0 after being given $\mathcal{X}$, which can be rewritten as

$$f_{\text{upp}}(\vec{X}_i \mid \mathcal{X}) = \prod_{l=1}^{D} \left[\frac{b^{*a^*} d^{*c^*}}{X_{il}} \times F(X_{il},\alpha_0,\beta_0) \right] \tag{2.15}$$

where $F(X_{il},\alpha_0,\beta_0)$ can be straightforwardly deduced from Eq.2.14.

The means $\mathbf{E}(\alpha)$ and $\mathbf{E}(\beta)$ are the most representative values of α_0 and β_0, respectively, which can be taken to approximate the optimal solution (α_0^*, β_0^*). Besides, the means calculated by the observations in $\mathcal{X}$ are independent of X_{il}. To facilitate the calculation, the minimum of the upper bound can be approximated as

$$
\begin{aligned}
\min_{\alpha_0,\beta_0} & f_{\text{upp}}(\vec{X}_i \mid \mathcal{X}) \\
&= \prod_{l=1}^{D} \left[\frac{b^{*a^*} d^{*c^*}}{X_{il}} \times \min_{\alpha_0,\beta_0} F(X_{il}, \mathbf{E}(\alpha), \mathbf{E}(\beta)) \right] \\
&\approx \prod_{l=1}^{D} \left[\frac{b^{*a^*} d^{*c^*}}{X_{il}} \times F(X_{il}, \mathbf{E}(\alpha), \mathbf{E}(\beta)) \right]
\end{aligned}
\tag{2.16}
$$

Since $\min_{\alpha_0,\beta_0} f_{upp}(\vec{X}_i \mid \mathcal{X})$ is unnormalized, we need to calculate the normalization factor:

$$
C_{upp} = \int_0^{\infty} \min_{\alpha_0,\beta_0} f_{upp}(\vec{X}_i \mid \mathcal{X}) d\vec{X}_i \tag{2.17}
$$

The approximation to the mixture predictive distribution is finally obtained as

$$
\begin{aligned}
f(\vec{X}_i \mid \mathcal{X}) &\approx f_{appx}^{LVI}(\vec{X}_i \mid \mathcal{X}) \\
&= \sum_{j=1}^{M} \frac{\pi_j}{C_{uppj}} \prod_{l=1}^{D} \left[\frac{{b_{jl}^*}^{a_{jl}^*} {d_{jl}^*}^{c_{jl}^*}}{X_{il}} \times F(X_{il}, \mathbf{E}(\alpha_{jl}), \mathbf{E}(\beta_{jl})) \right]
\end{aligned}
\tag{2.18}
$$

2.2 Experimental Results

The predictive distribution of GID (Eq.2.18) is based on LVI as we mentioned previously. In this section, we first generate different training data to compare the performance of out LVI approach with GVI. GVI is the approximation using the posterior mean as the point estimates:

$$
\begin{aligned}
f(\vec{X}_i \mid \mathcal{X}) &\approx f_{appx}^{GVI}(\vec{X}_i \mid \mathcal{X}) \\
&= \sum_{j=1}^{M} \pi_j \prod_{l=1}^{D} \text{iBeta}\left(X_{il} \mid \mathbf{E}(\alpha), \mathbf{E}(\beta)\right)
\end{aligned}
\tag{2.19}
$$

In the second experiment, we apply our model for activity recognition.

2.2.1 Synthetic Data

We generate data from a known GID distribution to test our model in this section. The parameters of GVI and LVI are trained separately under the different training data. And then, all the test data will be used in GVI and LVI frameworks to obtain the predictive distributions. Finally, Kullback-Leibler (KL) divergence is used to judge which obtained predictive distribution is better.

$$\mathbf{KL}(f_{true}||f_{appx}^{GVI}) = \int f_{true}(x) \ln \frac{f_{true}(x)}{f_{appx}^{GVI}} dx \tag{2.20}$$

$$\mathbf{KL}(f_{true}||f_{appx}^{LVI}) = \int f_{true}(x) \ln \frac{f_{true}(x)}{f_{appx}^{LVI}} dx \tag{2.21}$$

where f_{true} is the true probability distribution, f_{appx}^{GVI} and f_{appx}^{LVI} are the GVI and LVI predictive distributions, respectively. Table 4.1 summarizes the average results over 10 random experiments. According to this table we can see clearly that our LVI approach provides better results than the previously proposed GVI one.

Table 2.1: Comparison of the KL divergences ($\times 10^2$)
(The results are averaged over 10 random experiments)

Distribution	KL divergences	N = 10	N = 20	N = 50	N = 100	N = 200
$iBeta(x; 3, 4)$	$\mathbf{KL}(f\|\|f_{appx}^{GVI})$	27.59	21.27	2.43	0.80	0.62
	$\mathbf{KL}(f\|\|f_{appx}^{LVI})$	**17.52**	**13.64**	**2.18**	**0.93**	**0.62**
$iBeta(x; 2, 7)$	$\mathbf{KL}(f\|\|f_{appx}^{GVI})$	16.78	7.31	2.59	1.42	0.65
	$\mathbf{KL}(f\|\|f_{appx}^{LVI})$	**12.26**	**6.08**	**2.31**	**1.39**	**0.70**
$iBeta(x; 3, 8) * 0.2$ $iBeta(x; 5, 5) * 0.2$ $iBeta(x; 4, 6) * 0.6$	$\mathbf{KL}(f\|\|f_{appx}^{GVI})$	31.63	10.17	3.48	2.34	1.58
	$\mathbf{KL}(f\|\|f_{appx}^{LVI})$	**15.63**	**7.98**	**2.74**	**2.03**	**1.62**
$iBeta(x; 1, 6) * 0.1$ $iBeta(x; 3, 2) * 0.1$ $iBeta(x; 5, 2) * 0.8$	$\mathbf{KL}(f\|\|f_{appx}^{GVI})$	155.13	204.81	90.39	11.90	6.97
	$\mathbf{KL}(f\|\|f_{appx}^{LVI})$	**131.65**	**175.37**	**137.26**	**135.38**	**155.78**
$iBeta(x; 1, 5) * 0.2$ $iBeta(x; 7, 2) * 0.3$ $iBeta(x; 2, 6) * 0.5$	$\mathbf{KL}(f\|\|f_{appx}^{GVI})$	40.72	22.07	13.54	9.83	8.08
	$\mathbf{KL}(f\|\|f_{appx}^{LVI})$	**36.91**	**21.40**	**13.51**	**10.62**	**9.23**

2.2.2 Activity Recognition

In this subsection, our statistical framework is applied to tackle the activity recognition problem. First, the considered dataset is described. Some important procedures about data preprocessing are also presented. Second, the recognition results are given and analysed.

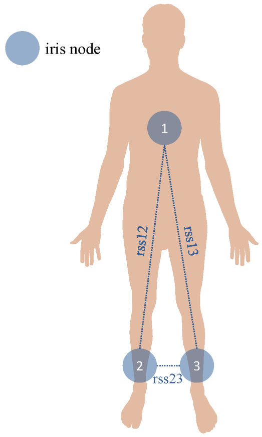


Figure 2.1: The sensors placement [1]

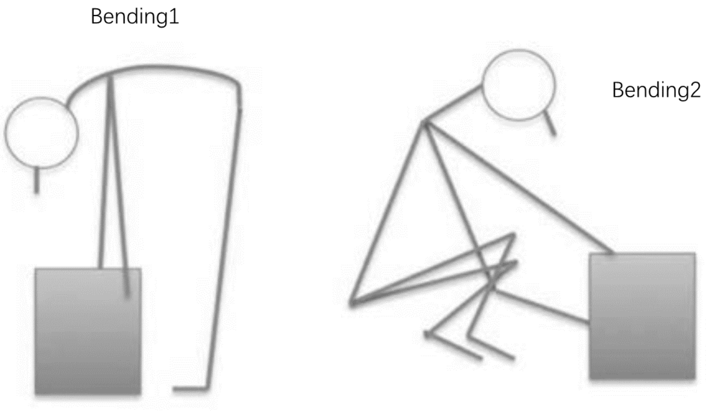


Figure 2.2: The two types of bending activity. [1]

Dataset Description and Preprocessing Procedure

The dataset represents a real-life benchmark [1]. Three wireless sensors (IRIS node [68]) are placed on a user to collect Received Signal Strength (RSS) as Fig.2.1 shows. In the collected dataset, seven activities are defined: Lying, Sitting, Standing, Walking, two types of Bending (Keeping the legs straight and folded), and Cycling (see Fig. 2.2). Sensors data were captured as pair (rss12, rss13 and rss 23) with a frequency of 20hz. The final dataset consist of avg_rss12, var_rss12, avg_rss13, var_rss13, avg_rss23, var_rss23 where avg and var are the mean and variance values over 250ms of data, respectively.

The normalization for the original data is essential because of the magnitude difference and zero appearance. In this experiment, we proportionally normalized the original data between 0.1 and 1. Finally, 3360 original samples were considered in total with 480 samples in each activity class. Before each experiment, 20 samples are randomly selected from each activity class as test data (140 samples totally). The synthetic data section has indicated that our model has high prediction performance for small training data. Therefore, the amount of training data is incremented dynamically, which is convenient to observe and compare the performance of the prediction model under different amounts of training data. We tested six different amounts of training data namely 5, 10, 20, 50, 100 and 200 samples for each activity class.

Table 2.2: Average accuracy of predictive models with different training data sizes. *(N: The amount of training data in each activity class)*

Type of predictive model	N=5	N=10	N=20	N=50	N=100	N=200
PD	0.44	0.480714	0.537143	0.645	0.697857	0.775
GVI	0.292143	0.372143	0.523571	0.674286	0.745	0.799286

Results Analysis

For each of the six different amounts of training data, we have applied our framework 10 times to take into account experimental uncertainty. Random selection of initialization parameters is considered to ensure the consistency of the model. Table 2.2 displays the average accuracy of our 10 simulations (seed 0-9) considering the seven activities. Fig. 2.3 shows the average accuracy for each activity type as a function of the training data size. In the line chart, x-axis represents the amount of training data for each type of activity. We obtained the results for 5, 10, 20, 50, 100 and 200 training data which are enough to display the general accuracy change for the recognition of the different activities. The y-axis is divided into seven different intervals (one for each activity) meaning 0%-100% in each interval. There are two lines for each type of activity, which are the accuracy fluctuation of our model and the accuracy fluctuation of GVI. We can clearly observe the changes of two accuracies for each activity. Let us focus on the case when the amount of training data is 5 in Fig.2.3. The accuracy of our model is higher than that of GVI in almost all activities except Lying activity. This trend is still being maintained when the amount of training data increases to 10. Only the accuracy of GVI in Bending2 and Standing are slightly better than that of our model. Therefore, the average accuracy of our model is obviously higher than that of GVI when the number of training data equals 5 and 10 as shown in Table.2.2. Yet, the comparison begins to become difficult when the amount of training data in each type of activity increases to 20. Both average accuracies for N=20 are slightly similar, about 0.53. When the number of training data becomes higher than 20, the average accuracy of GVI tends to be continually higher than that of our model. And the gap between the two average accuracies shows a trend of first expanding and then decreasing. Finally, both accuracies approach 0.8.

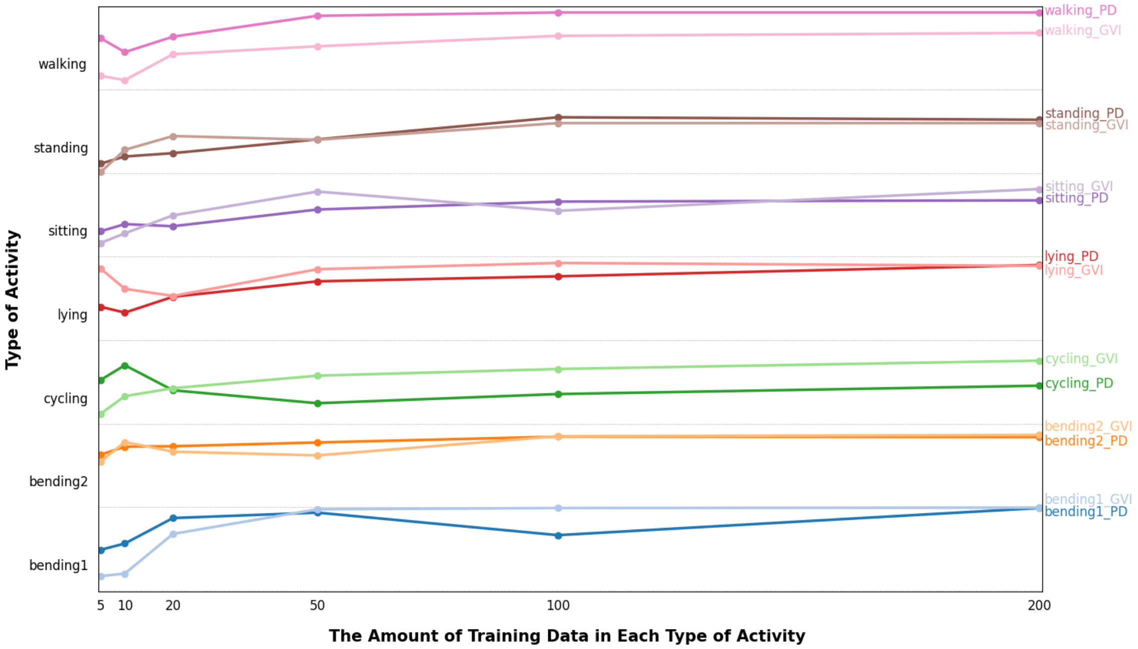


Figure 2.3: The accuracy fluctuation when increasing the training data size.

Chapter 3

Occupancy Estimation in Smart Buildings using Predictive Modeling in Imbalanced Domains

In this chapter, we build a predictive model for occupancy estimation. This model still performs well in imbalanced domains via over-sampling method.

3.1 Predictive Model

3.1.1 Inverted Dirichlet Mixture Model

Assume that a D-dimensional positive vector $\mathbf{x}_n = (x_{n1}, ..., x_{nD})$, representing sensors outputs for instance, is sampled from a finite inverted Dirichlet mixture model with I components, then we have:

$$p(\mathbf{x}_n|\boldsymbol{\pi}, \boldsymbol{\alpha}) = \sum_{i=1}^{I} \boldsymbol{\pi}_i \mathcal{ID}(\mathbf{x}_n|\boldsymbol{\alpha}_i) \tag{3.1}$$

where $\boldsymbol{\alpha} = (\boldsymbol{\alpha}_1, ..., \boldsymbol{\alpha}_I)$ and $\boldsymbol{\pi} = (\pi_1, ..., \pi_I)$ denotes the mixing coefficients with the constraints that are positive and sum to one. $\mathcal{ID}(\mathbf{x}_n|\boldsymbol{\alpha}_i)$ is an inverted Dirichlet distribution representing component i with parameter $\boldsymbol{\alpha}_i$ and is defined by [69]

$$\mathcal{ID}(\mathbf{x}_n|\boldsymbol{\alpha}_i) = \frac{\Gamma(\sum_{d=1}^{D+1} \alpha_{id})}{\prod_{d=1}^{D+1} \Gamma(\alpha_{id})} \prod_{d=1}^{D} x_{nd}^{\alpha_{id}-1} (1 + \sum_{d=1}^{D} x_{nd})^{-\sum_{d=1}^{D+1} \alpha_{id}} \tag{3.2}$$

where $0 < x_{nd} < \infty$ for $d = 1, ..., D$. In addition, $\boldsymbol{\alpha}_i = (\alpha_{i1}, ..., \alpha_{iD+1})$, such that $\alpha_{id} > 0$ for $d = 1, ..., D+1$.

3.1.2 Priors

An important property of the exponential family of distributions is that a formal conjugate prior could be developed [49]. The inverted Dirichlet is part of the exponential family as shown in [70] and as such, its conjugate prior is given by

$$\begin{aligned} f(\boldsymbol{\alpha}_i; \boldsymbol{\beta}_0, v_0) =& \frac{1}{\mathbf{C}(\boldsymbol{\beta}_0, v_0)} \left[\frac{\Gamma(\sum_{d=1}^{D+1} \alpha_{id})}{\prod_{d=1}^{D+1} \Gamma(\alpha_{id})} \right]^{v_0} \\ & \times e^{\ln \boldsymbol{\beta}_0 (\boldsymbol{\alpha}_i - \mathbf{1}_D)^T - (\sum_{d=1}^{D+1} \alpha_{id}) \ln(1 + \sum_{d=1}^{D} \beta_{0_d})} \end{aligned} \tag{3.3}$$

where $\boldsymbol{\beta}_0 = [\beta_{0_1}, ..., \beta_{0_D}]$ and v_0 are the prior hyperparameters, $\frac{1}{\mathbf{C}(\boldsymbol{\beta}_0, v_0)}$ is a normalization coefficient, $\mathbf{1}_D$ is a D-dimensional vector all whose elements are equal to one. Having the prior in hand, the posterior is given by

$$\begin{aligned} & f(\boldsymbol{\alpha}_i | \mathbf{X}; \boldsymbol{\beta}_N, v_N) \\ & = \frac{\mathcal{ID}(\mathbf{X}|\boldsymbol{\alpha}_i) f(\boldsymbol{\alpha}_i; \boldsymbol{\beta}_0, v_0)}{\int \mathcal{ID}(\mathbf{X}|\boldsymbol{\alpha}_i) f(\boldsymbol{\alpha}_i; \boldsymbol{\beta}_0, v_0) d\boldsymbol{\alpha}} \\ & = \frac{1}{\mathbf{C}(\boldsymbol{\beta}_N, v_N)} \left[\frac{\Gamma(\sum_{d=1}^{D+1} \alpha_{id})}{\prod_{d=1}^{D+1} \Gamma(\alpha_{id})} \right]^{v_N} \\ & \quad \times e^{\ln \boldsymbol{\beta}_N (\boldsymbol{\alpha}_i - \mathbf{1}_D)^T - (\sum_{d=1}^{D+1} \alpha_{id}) \ln(1 + \sum_{l=1}^{D} \beta_{N_l})} \end{aligned} \tag{3.4}$$

where $\boldsymbol{\beta}_N = \boldsymbol{\beta}_0 + \ln \mathbf{X} \times \mathbf{1}_N$ and $v_N = v_0 + N$ are the hyperparameters of the posterior distribution. We can see clearly that the prior in Eq.3.3 and the posterior have the same form which confirms the conjugacy property. Unfortunately working with the above conjugate prior is complex because of the normalizing function which cannot allow to obtain, for instance, the mean and the covariance matrix which makes the use of the obtained posterior distribution intractable.

A better approach to find a tractable conjugate prior and posterior distributions has been based on a global variational inference (GVI) framework as proposed in [71], where the prior was approximated as a product of Gamma distributions which give the following posterior

$$f(\boldsymbol{\alpha}|\mathbf{X};\boldsymbol{\beta}_N,v_N)\approx f(\boldsymbol{\alpha}|\mathbf{X};\mathbf{u}^*,\mathbf{v}^*)=\prod_{d=1}^{D+1}\mathbf{Gam}(\alpha_d|\mathbf{X};u_d^*,v_d^*) \tag{3.5}$$

where $\mathbf{u}^*=(u_1^*,\ldots,u_{D+1}^*)$, $\mathbf{v}^*=(v_1^*,\ldots,v_{D+1}^*)$, u_d^* and v_d^* are the hyperparameters.

3.1.3 Predictive Distribution of the Mixture Model

We start by supposing that the likelihood function in Eq.1.1 is a single inverted Dirichlet distribution. By using the approximated conjugate prior, we obtain the following according to the obtained posterior in Eq.3.5

$$\begin{aligned} f(\mathbf{x}|\mathbf{X}) &= \int \mathcal{ID}(\mathbf{x}|\boldsymbol{\alpha})f(\boldsymbol{\alpha}|\mathbf{X};\mathbf{u}^*,\mathbf{v}^*)d\boldsymbol{\alpha} \\ &= \int \frac{\Gamma(\sum_{d=1}^{D+1}\alpha_d)}{\prod_{d=1}^{D+1}\Gamma(\alpha_d)}\prod_{d=1}^{D}x_d^{\alpha_d-1}(1+\sum_{d=1}^{D}x_d)^{-\sum_{d=1}^{D+1}\alpha_d} \\ &\quad\times\prod_{d=1}^{D}\frac{(v_d^*)^{u_d^*}}{\Gamma(u_d^*)}\alpha_d^{u_d^*-1}e^{-v_d^*\alpha_d}d\boldsymbol{\alpha} \end{aligned} \tag{3.6}$$

It is clear that the predictive distribution in Eq.3.6 has an intractable form. Fortunately, this problem can be solved by a local variational inference (LVI) method [44] which is based on finding an upper-bound of the predictive distribution. In the following, we explain the idea behind LVI.

Unlike global variational inference used in [71] and which deploys a bound that approximates the variational objective function in terms of all the model's variables, LVI focuses on a subset of the variables to find an approximation [49]. A final approximation is obtained then after applying in turn multiple local approximations [49]. Indeed, suppose that the original intractable integral is

$$F=\int f(x)g(x)dx \tag{3.7}$$

where $f(x)$ is a probability density function of x and $g(x)$ is a function of x. We can find a function $h(x,\sigma)$, which is an analytically tractable upper-bound to $g(x)$, which allows to upper-bound the intractable integral as

$$F \leq G(\sigma) = \int f(x)h(x, \sigma)dx \tag{3.8}$$

In order to make $G(\sigma)$, expressed in closed form, approach F as much as possible, we need to find an optimal σ^*, which minimizes $G(\sigma)$ as

$$\sigma^* = \arg\min_{\sigma} G(\sigma) \tag{3.9}$$

Obviously, the optimized value σ^* depends on x, and $G(\sigma^*)$ is just an optimal convenient approximation to the integral [67]. An upper-bound for the predictive distribution of the inverted Dirichlet is given by (see Appendix A)

$$\begin{aligned} f_{upp}(\mathbf{x}|\mathbf{X}) =& \frac{\Gamma(\sum_{d=1}^{D+1} \tilde{\alpha}_d)}{\prod_{d=1}^{D+1} \Gamma(\tilde{\alpha}_d)} \times e^{-\sum_{d=1}^{D+1} \tilde{\alpha}_d \left[\psi\left(\sum_{d=1}^{D+1} \tilde{\alpha}_d\right) - \psi(\tilde{\alpha}_d)\right]} \\ & \times \prod_{d=1}^{D+1} \frac{(v_d^*)^{u_d^*}}{x_d \left[\mathbf{G}(x_d, \tilde{\boldsymbol{\alpha}})\right]^{u_d^*}} \end{aligned} \tag{3.10}$$

where $v_d^*, u_d^*, d = 1, 2, ..., D+1$ are the hyperparameters of the predictive distribution, which are obtained and fixed after the end of variational Bayes estimation. Therefore, the upper-bound is only a function of $\tilde{\boldsymbol{\alpha}}$ for any data vector $\mathbf{x}$. It is noteworthy that the result obtained in Eq.3.10 is the same as the one obtained in the case of the Dirichlet distribution [44], thus a good approximation to the predictive distribution of the inverted Dirichlet could be given, by analogy to the Dirichlet one, as [44]

$$\begin{aligned} f_{upp}(\mathbf{x}|\mathbf{X}) &\approx f_{appx}^{post.}(\mathbf{x}|\mathbf{X}) \\ &= \frac{1}{\mathbf{C}} \times \mathbf{P}(\mathbf{x}, \bar{\boldsymbol{\alpha}}) \times \prod_{d=1}^{D+1} \frac{(v_d^*)^{u_d^*}}{x_d} \end{aligned} \tag{3.11}$$

where $\mathbf{C}$ is a normalization factor and

$$
\begin{aligned}
\mathbf{P}(\mathbf{x}, \tilde{\boldsymbol{\alpha}}) =& \frac{\Gamma(\sum_{d=1}^{D+1} \tilde{\alpha}_d)}{\prod_{d=1}^{D+1} \Gamma(\tilde{\alpha}_d)} \times e^{-\sum_{d=1}^{D+1} \tilde{\alpha}_d \left[\psi\left(\sum_{d=1}^{D+1} \tilde{\alpha}_d\right) - \psi(\tilde{\alpha}_d)\right]} \\
& \times \prod_{d=1}^{D+1} \frac{1}{\mathbf{G}(x_d, \tilde{\boldsymbol{\alpha}})^{u_d^*}}
\end{aligned}
\tag{3.12}
$$

The posterior distribution of the IDMM can be obtained by the variational Bayes estimation method proposed in [71]. The predictive likelihood of a new vector given an estimated IDMM is given by

$$
f_{upp}(\mathbf{x}|\mathbf{X}) = \sum_{i=1}^{I} \pi_i \int \mathcal{ID}(\mathbf{x}|\boldsymbol{\alpha}_i) f(\boldsymbol{\alpha}_i|\mathbf{X}) d\alpha_i \tag{3.13}
$$

Using the approximation derived in Eq.3.11, the predictive distribution of the IDMM can be approximated as

$$
\begin{aligned}
f_{upp}(\mathbf{x}|\mathbf{X}) &\approx f_{appx}^{LVI}(\mathbf{x}|\mathbf{X}) \\
&= \sum_{i=1}^{I} \pi_i \left[\frac{1}{\mathbf{C}_i} \times \mathbf{P}(\mathbf{x}, \bar{\boldsymbol{\alpha}}_{\boldsymbol{i}}) \times \prod_{d=1}^{D+1} \frac{(v_{di}^*)^{u_{di}^*}}{x_d} \right]
\end{aligned}
\tag{3.14}
$$

3.2 Over-Sampling Via Data Generation and Complete Algorithm

In this section we tackle the imbalanced data problem via a data pre-processing strategy. Pre-processing techniques consist of approaches that use the actual dataset considering the user preference biases. These techniques can be grouped into two main categories [72]: 1) Distribution change approaches, and 2) Data space weighting techniques. Approaches in the first family change the data distribution to address the issue of poor representativeness of some classes. The second family of approaches modifies the distribution of the training set to avoid costly classification errors (i.e. by deploying information concerning misclassification costs) [73]. Here, we develop an effective solution that belongs to the first category which can be divided itself into three types of approaches [72]: stratified sampling, generating new data, or combination of both. While stratified sampling consists of removing and/or adding real examples to the original data set, data synthesis involve the generation of new artificially generated samples added also to the original data set. Our developed approach that we describe in the following can be viewed as an over-sampling technique based on data generation [1].

The over-sampling approach that we propose is based on generating new samples for the under-represented classes from a set of existing samples in these classes (i.e. by calculating the average of these randomly selected samples). It is noteworthy that this over-sampling plays a crucial role for the predictive modeling technique that we developed in the previous section. Indeed, the normalization coefficients ($\mathbf{C}_i$) in Eq.3.14 need to integrate all the predictive possibilities of training data that should be balanced to have the most possible accurate prediction for the new test data $\mathbf{x}$. The whole prediction algorithm with the proposed over-sampling approach can be summarized as follows.

1. Start with an initial training dataset $\mathbf{X}$ with know labels $\mathbf{Y}$. Assuming there are I possible labels in the training dataset.

2. Calculate the gap between the amount of samples in each group $(N_1, N_2, ...N_I)$ and the maximum amount of samples (N_{max}). And then, set the number of iterations (J) and the number of new data to be synthesized for each group $(K_1, K_2, ...K_I)$.
   $K_1, K_2, ...K_I = (N_{max} - (N_1, N_2, ...N_I))/J$

3. Randomly choose S samples from class $i, i = 1, \dots, I$ in the current training dataset $\mathbf{X}$ to calculate their mean which will be considered as the new data x_{ki} for group i. This process continues until the number of new introduced data for the different groups satisfies each K_i.
   $\mathbf{X} = \mathbf{x} + \mathbf{X}$

4. $J = J - 1$.
   Repeat Steps 3-4 until $J = 0$.

5. Run the variational Bayes estimation method proposed in [71] to train the

[1] Data generation techniques themselves can be categorized into two groups [72]. The first group of approaches introduces perturbations (i.e. producing noisy replicates of existing data). The second group is based on interpolating existing data. Our approach belongs to the second group.

inverted Dirichlet statistical model using the available training data $\mathbf{X}$ and $\mathbf{Y}$, get the weight π of each component, and the hyperparameters u^* and v^*.

6. For each mixture component, calculate the posterior means $\bar{\boldsymbol{\alpha}} = \mathbf{u}/\mathbf{v}$

7. Calculate the normalization coefficient $\mathbf{C}_i$ for the mixture components.

8. For any upcoming $\mathbf{x}$, the predictive distribution for each possible label is calculated by Eq.3.14

3.3 Experimental Results

The goal of this section is to validate the predictive distribution of IDMM with over-sampling using both synthetic data and real data extracted in the context of a challenging application namely occupancy estimation in smart buildings. In the synthetic data part, we firstly generate a dataset from a known IDMM model. Then, the LVI and GVI based predictive distributions performances are compared. The distributions comparison is based on the Kullback-Leibler (KL) divergence which is widely used in the literature (see, for instance, [53]). The occupancy estimation task is extensively detailed and different scenarios are discussed.

3.3.1 Synthetic Data

As mentioned previously, there are two main variational approaches to approximate the predictive distribution namely GVI and LVI. The goal of this experiment is to compare both approaches. The LVI approach for IDMM has been presented in previous section and the conventional GVI one was developed in [71] and is mainly based on the point estimate plug-in method by using the posterior mean as the point estimate:

$$f_{upp}(\mathbf{x}|\mathbf{X}) \approx f_{appx}^{GVI}(\mathbf{x}|\mathbf{X}) = \sum_{i=1}^{I} \pi_i \mathcal{ID}(\mathbf{x}; \bar{\boldsymbol{\alpha}}_i^{GVI}) \tag{3.15}$$

The synthetic data were generated from a two-component mixture of inverted Dirichlet distributions. The two components have the following $\boldsymbol{\alpha}$ parameters: [3,25,12] and [21,5,15]. Their weights $\boldsymbol{\pi}$ are [0.7, 0.3]. Each time, we extract different numbers

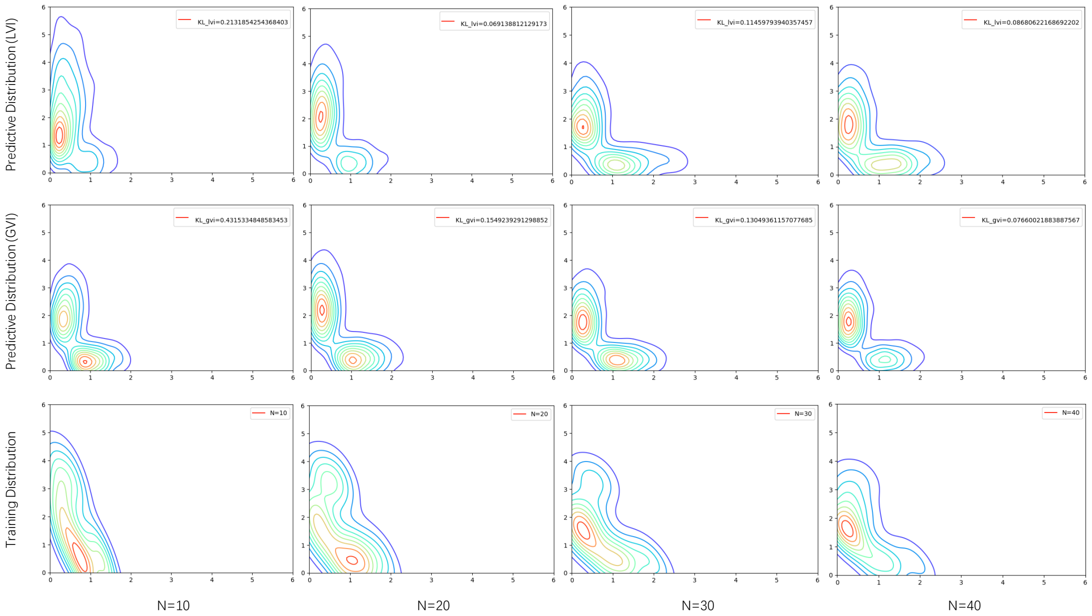


Figure 3.1: Comparisons of predictive distributions obtained using LVI and GVI when varying the training dataset size.

of samples from the true distribution as training dataset and then we use GVI and LVI methods to estimate the density functions. The obtained estimation results using both variational approaches when varying the size of the training data are shown in Fig.3.1.

To compare the LVI-based method (Eq.3.14) with the GVI-based one (Eq.3.15), we also evaluated the KL divergences between the true predictive distribution (See Fig.3.2) and the approximated ones. According to our results, we can notice that the predictive distributions obtained using both methods become gradually closer to the true distribution by increasing the training dataset size (See Fig.3.3). Indeed, we can clearly see that the KL values when using both LVI and GVI decrease with the growth of training dataset size and that the trend is similar for both methods which can be explained by the fact that LVI is partly based on the parameters obtained via GVI. When the amount of training data is small, the performance of LVI is better than GVI. However, the KL divergence when using GVI keeps a stable and better level when the amount of training data increases.

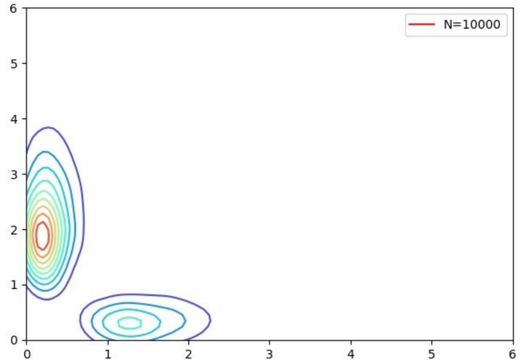


Figure 3.2: The true distribution based on generating 10000 points from a two-component mixture model with parameters $\boldsymbol{\alpha} = [[3, 25, 12], [21, 5, 15]], \boldsymbol{\pi} = [0.7, 0.3]$

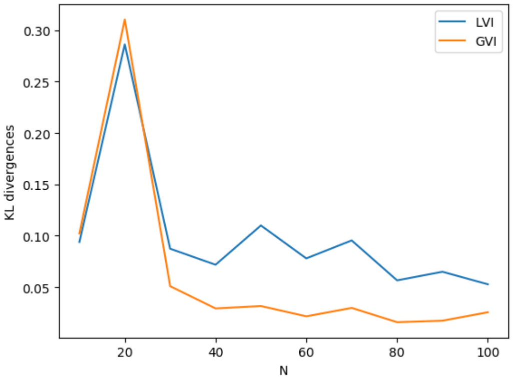


Figure 3.3: Comparison of KL divergences the true distribution and approximated ones obtained using GVI and LVI methods. The KL values are averaged over 5 simulation rounds.

3.3.2 Occupancy Estimation

In this section, we apply our statistical framework to tackle occupancy estimation problem in different situations and scenarios to verify its validity. The data used are extracted from a real-life heterogenous sensor environment as we explain in the following.

Dataset Description and Preprocessing Procedure

The real data used in this section were collected from a test bed (Fig.3.4) in Grenoble Institute of Technology. The test bed is an office usually used by a professor and 3 PhD students. It also houses frequently visitors for meetings and presentations.

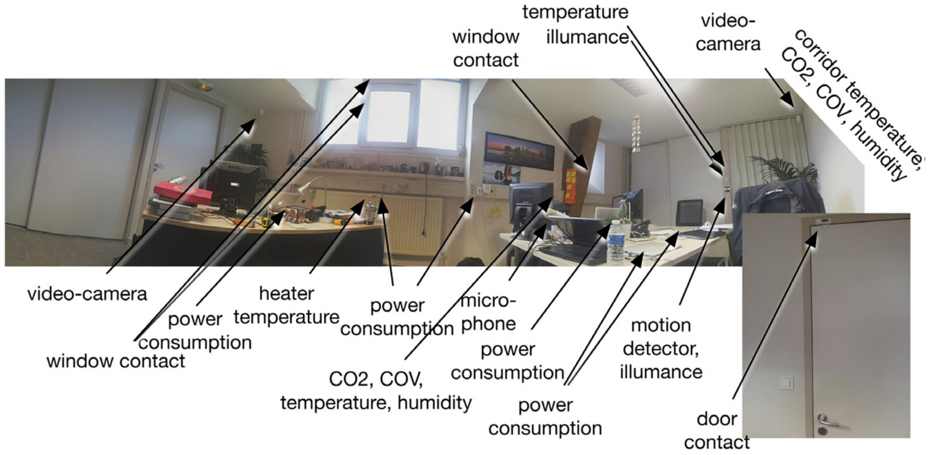


Figure 3.4: Sensor test bed at Grenoble INP [2]

Many sensors are installed in the office (e.g. temperature, relative humidity, motions, C02, power consumption, door and window positions, acoustic pressure from microphone, etc.). Moreover, two video cameras were installed to record real occupancy numbers and activities.

More details about the test bed and data collection and processing can be found in a previous work [2] where it was shown that the most relevant features that should be considered for occupancy estimation are motion, power consumption, acoustic pressure (microphone) and door opening. Thus, the data set that we consider in the following simulations is composed of 718 four-dimensional labeled vectors. The data directly obtained from sensors have generally different degrees of magnitude. Thus, we have used the following activation function in our experiments:

$$A(x) = \tanh(x * 0.1 + 0.3) \tag{3.16}$$

The 0.1 keeps the data from crowding around the edge of *tanh* and enlarges data features. The 0.3 is the factor to avoid the appearance of zero in data. If zero appears in the denominator, the result will be singular. In addition, cross-validation is essential and helpful for avoiding over-fitting since the dataset is small. Thus, the different data classes are divided into training data and test data in a ratio of 0.7:0.3.

Occupancy Estimation with Original Training Data

The original training data is imbalanced as we can see from Table 3.1. We considered these data to test our model with or without over-sampling. The purposes is to verify the necessity of synthesizing new data to improve the estimation accuracy.

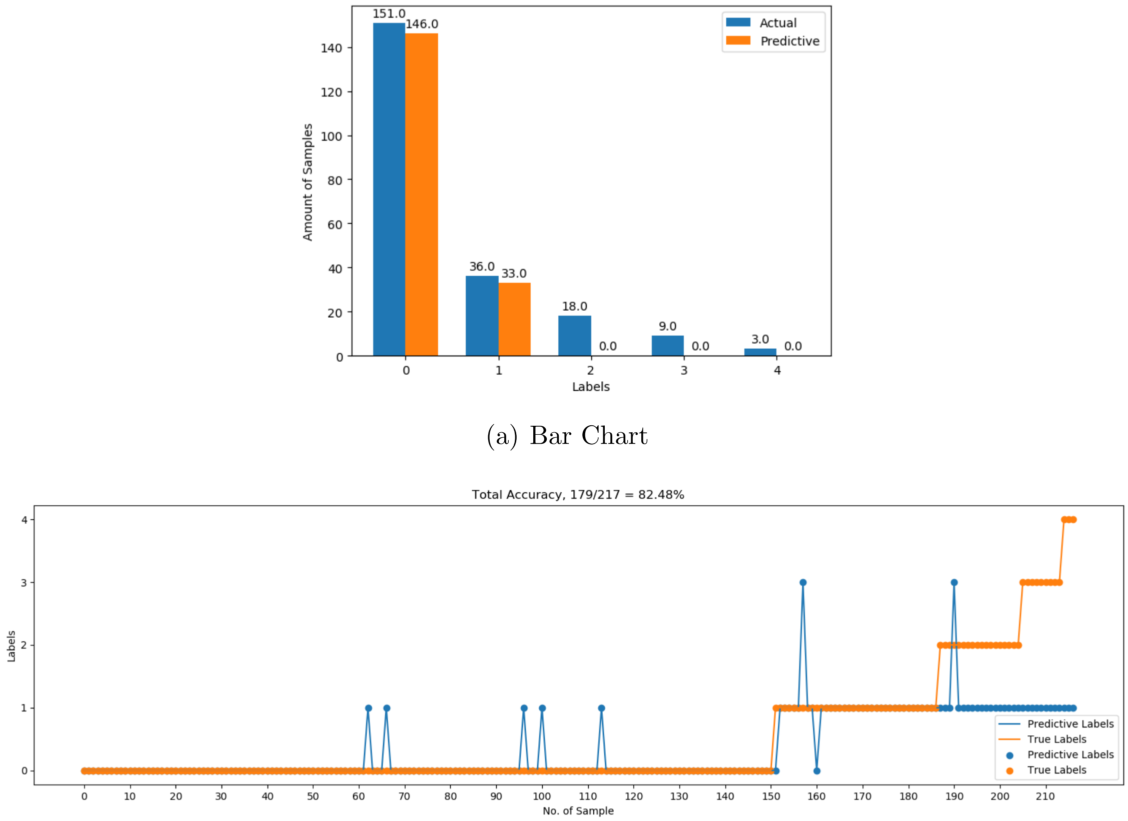
Table 3.1: Size of the classes in the original training data.

Label	Number of samples
0	352
1	82
2	42
3	18
4	6

The results of the model without over-sampling are shown in Fig.3.5. The bar chart shows how many samples from each class are predicted correctly. And the detailed prediction can be seen in the line graph. The blue line and points represent the actual situation. On the contrary, the orange line and point represents the predicted situation. The labels on the x-axis represent the prediction accuracy for different numbers of people in the office. Fig.3.5 represents just one simulation of the experiments using training data without over-sampling. After 10 simulations, the total accuracy stays around 82.94%. Yet, the distribution of the correct labels predictions is extremely imbalanced. There are few or no correct predictions for labels 2,3,4. This can be explained by following two points:

- The data in classes 1 and 2 are too much similar. It is difficult for the model to distinguish the two classes when the training data is so small.
- The sizes of classes 3 and 4 are too small and because our model is based on the integral normalization coefficient, the densities for classes 3 and 4 are too sparse.

The model is then reinforced after adding over-sampling module. Assuming that the number of iterations for over-sampling module is N, the growth of data size in each class in each iteration is calculated according to the following equation:



(a) Bar Chart

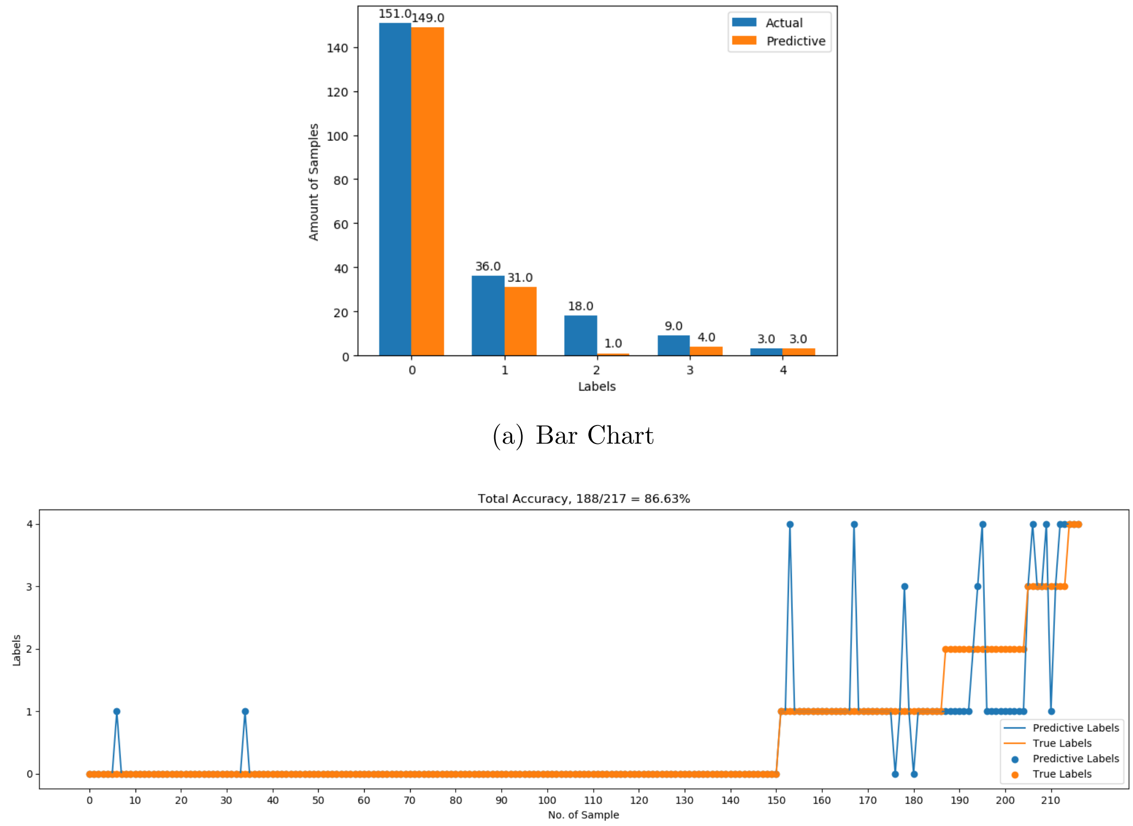
(b) Line Graph

Figure 3.5: The results using original training data without over-sampling

$$\mathbf{S} = \mathbf{S} + (\max(\mathbf{S}) - \mathbf{S})/N \tag{3.17}$$

where $\mathbf{S} = [352, 82, 42, 18, 6]$ represents the size of the different classes in the original data set. It is noteworthy that the hyperparameter u^* and v^* are updated after each iteration. As the training data increases, these parameters will be optimized step by step based on last iteration. Moreover, the synthesized data of each iteration are based on the training data of last iteration. This way of data generation makes the distribution of training data close to the real distribution.

Fig.3.6 shows the results of the model after over-sampling. Obviously, the distribution of the correct predictions is better than Fig.3.5. Even though the accuracy just increases from 82.48% to 86.63%, the prediction accuracy regarding classes 4 and 5 observed significant improvement. Thus, the average accuracy in terms of labels has been improved significantly. After 10 simulations with and without over-sampling, we



(a) Bar Chart

(b) Line Graph

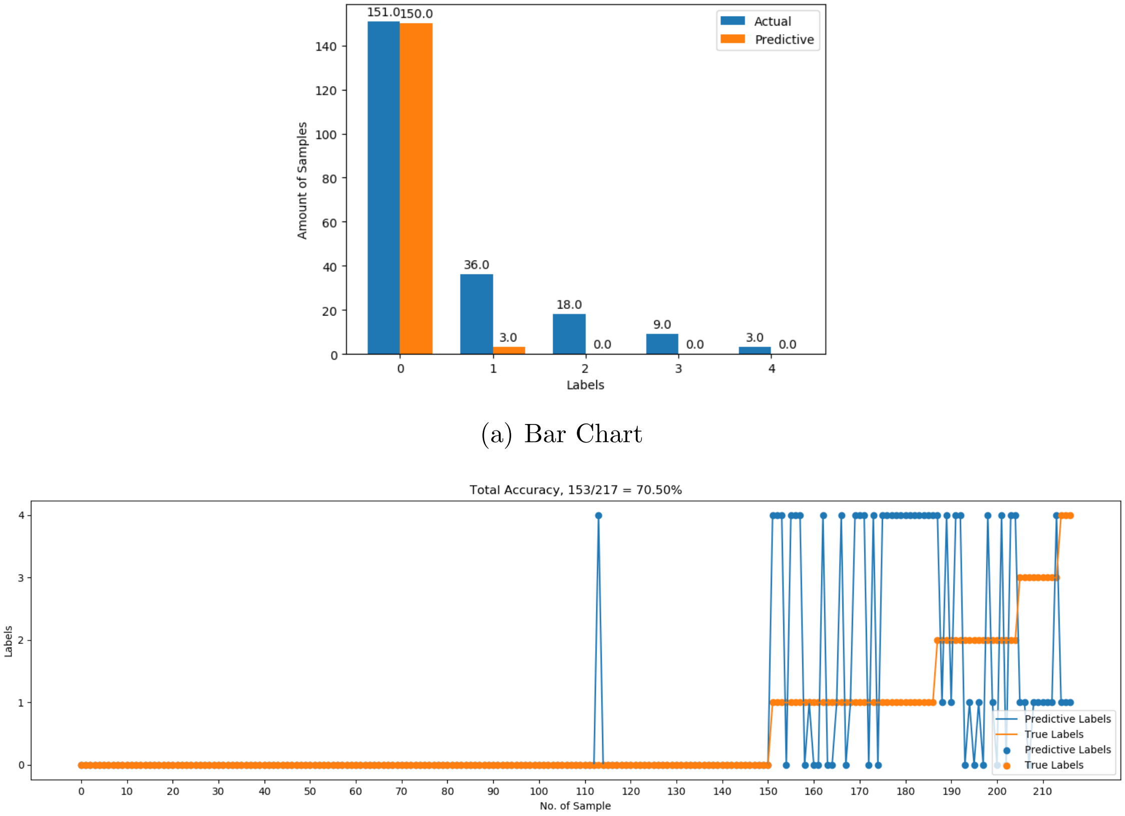
Figure 3.6: The results with over-sampling.

observed that the average accuracy in terms of labels had increased from 36.32% to 65.96%. In the line graph of Fig.3.6, we can still see that the majority of predictions that should be "label 2" are "label 1".

Occupancy Estimation with Extremely Small Training Data

When training data is extremely small, the KL divergences between predictive distribution based-LVI and the true distribution is better than the GVI-based variational inference learning. Thus, it is valuable to test the performance of the model with over-sampling.

The smallest class size in the original training data is 6. Thus, the sizes of the rest of classes are reduced to 6 samples (i.e.under-sampling) before the experiment. We can see the result of our prediction model without data generation in Fig.3.7. Almost all the predictions except those for class 0 are wrong, though the predictive



(a) Bar Chart

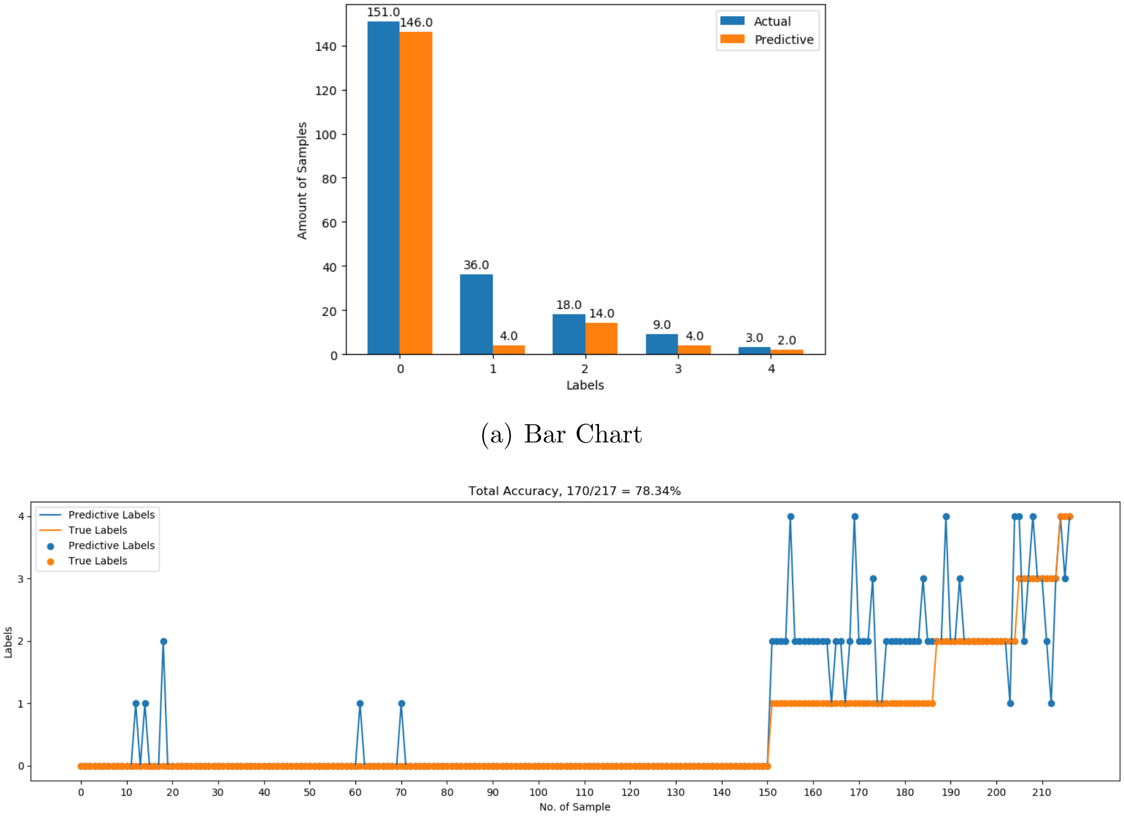
(b) Line Graph

Figure 3.7: The results when using 6 training data in each class without over-sampling.

distribution based on LVI is better than the one reached by GVI.

In the second experiment we introduced generated data. We set the number of iterations to 10 and generate 10 samples for each label in each iteration which gave us 106 training samples in each class. The prediction results are shown in Fig.3.8. After 10 simulations, the total accuracy increased from 69.76% to 79.72%. In addition, the average accuracy in terms of labels significantly increased from 20.32% to 52.98% which is close to the average accuracy of the model with original training data.

All the results are summarized in Tables 3.2 and 3.3. It is clear that the performance of the model is always improved with the support of the over-sampling module. It is interesting to observe also that the model with over-sampling performed well when we started with 6 samples in each class which makes our framework and attractive alternative when there are not too many training data at the beginning of



(a) Bar Chart

(b) Line Graph

Figure 3.8: The results wen using 6 training data in each class with over-sampling.

the estimation task.

Table 3.2: The total Accuracy.

Over-sampling Training Data	Without	With
Full original Data	82.48%	86.63%
6 Samples in each class	69.76%	79.72%

Table 3.3: The Average accuracy in terms of classes.

Over-sampling Training Data	Without	With
Full original Data	36.32%	65.96%
6 Samples in each class	20.32%	52.98%

Chapter 4

A Hybrid of Interactive Learning and Predictive Modeling For Occupancy Estimation in Smart Buildings

In this chapter, we present an occupancy estimation framework by combining the predictive model with interactive learning. This framework performs well for both small and imbalanced data.

4.1 Approximation of the Predictive Distribution of the Generalized Dirichlet Mixture Model

In this section, we briefly present the GD mixture model that we have previously proposed in [74], then we develop an approximation to its predictive distribution.

4.1.1 Generalized Dirichlet Mixture Model

Assume that a D-dimensional positive vector $\vec{Y} = (Y_1, \cdots, Y_D)$ is sampled from a finite mixture model of GD Distributions with M components, then:

$$p(\vec{Y} \mid \vec{\pi}, \vec{\alpha}, \vec{\beta}) = \sum_{j=1}^{M} \pi_j \mathrm{GD}\left(\vec{Y} \mid \vec{\alpha}_j, \vec{\beta}_j\right) \tag{4.1}$$

where $\vec{\alpha} = \{\vec{\alpha}_1, \ldots, \vec{\alpha}_M\}$, $\vec{\beta} = \left\{\vec{\beta}_1, \ldots, \vec{\beta}_M\right\}$, $\vec{\alpha}_j$ and $\vec{\beta}_j$ are the parameters of the GD distribution representing component j, where $\vec{\alpha}_j = (\alpha_{j1}, \cdots, \alpha_{jD})$ and $\vec{\beta}_j = (\beta_{j1}, \cdots, \beta_{jD})$. $\vec{\pi} = (\pi_1, \cdots, \pi_M)$ denotes the mixing coefficients with the constraints that are positive and sum to one. GD $\left(\vec{Y} \mid \vec{\alpha}_j, \vec{\beta}_j\right)$ is a GD distribution representing component j with parameters $\vec{\alpha}_j$ and $\vec{\beta}_j$ and is defined by [74]

$$\mathrm{GD}\left(\vec{Y} \mid \vec{\alpha}_j, \vec{\beta}_j\right) = \prod_{l=1}^{D} \frac{\Gamma\left(\alpha_{jl} + \beta_{jl}\right)}{\Gamma\left(\alpha_{jl}\right)\Gamma\left(\beta_{jl}\right)} Y_l^{\alpha_{jl}-1} \left(1 - \sum_{k=1}^{l} Y_k\right)^{\gamma_{jl}} \tag{4.2}$$

where $\sum_{l=1}^{D} Y_l < 1$ and $0 < Y_l < 1$ for $l = 1, \cdots, D$, $\alpha_{jl} > 0$, $\beta_{jl} > 0$, $\gamma_{jl} = \beta_{jl} - \alpha_{jl+1} - \beta_{jl+1}$ for $l = 1, \cdots, D-1$, and $\gamma_{jD} = \beta_{jD} - 1$.

Consider that a set of independent identically distributed vectors $\mathcal{Y} = \left\{\vec{Y}_1, \ldots, \vec{Y}_N\right\}$ follow a finite GD mixture model. According to the Bayes' theorem, the probability that the vector i is from component j conditional on the observed $\vec{Y}_i$ (also called as *responsibilities*) can be written as

$$p\left(j \mid \vec{Y}_i\right) \propto \pi_j GD\left(\vec{Y}_i \mid \vec{\alpha}_j, \vec{\beta}_j\right) \tag{4.3}$$

Because of an interesting mathematical property of the GD distribution [74], the responsibilities can be redefined as

$$p\left(j \mid \vec{Y}_i\right) \propto \pi_j \prod_{l=1}^{D} \mathrm{Beta}\left(X_{il} \mid \alpha_{jl}, \beta_{jl}\right) \tag{4.4}$$

where $X_{i1} = Y_{i1}$ and $X_{il} = Y_{il}/(1 - \sum_{k=1}^{l-1} Y_{ik})$ for $l > 1$ and $\mathrm{Beta}\left(X_{il} \mid \alpha_{jl}, \beta_{jl}\right)$ is a Beta distribution with parameters $(\alpha_{jl}, \beta_{jl})$:

$$\mathrm{Beta}\left(X_{il} \mid \alpha_{jl}, \beta_{jl}\right) = \frac{\Gamma(\alpha_{jl} + \beta_{jl})}{\Gamma(\alpha_{jl})\Gamma(\beta_{jl})} X_{il}^{\alpha_{jl}-1}(1 - X_{il})^{\beta_{jl}-1} \tag{4.5}$$

where $\Gamma(z) = \int_0^\infty x^{z-1} e^{-x} dx$. The Beta distribution has been widely deployed in various industrial application such, smart wireless indoor localization [75], thanks to its flexibility and ease of use.

Now, the mixture model of the finite GD distribution underlying dataset $\mathcal{Y}$ can be redefined by a new dataset $\mathcal{X} = \left\{\vec{X}_1, \ldots, \vec{X}_N\right\}$ using the following clustering structure with conditionally independent features

$$p\left(\vec{X}_i \mid \vec{\pi}, \vec{\alpha}, \vec{\beta}\right) = \sum_{j=1}^{M} \pi_j \prod_{l=1}^{D} \text{Beta}\left(X_{il} \mid \alpha_{jl}, \beta_{jl}\right) \tag{4.6}$$

The formal conjugate prior distribution of the Beta distribution [49] does not have a closed form because of the need to approximate the normalization factor. A solution to find a tractable approximation to the conjugate prior, based on a global variational inference (GVI) framework, has been proposed in [76], using a product of two independent Gamma distributions as

$$\begin{aligned} f(\alpha_{jl}, \beta_{jl}) &\approx \text{Gam}\left(\alpha_{jl}; a_0, b_0\right) \times \text{Gam}\left(\beta_{jl}; c_0, d_0\right) \\ &= \frac{b_0^{a_0}}{\Gamma\left(a_0\right)} \alpha_{jl}^{a_0-1} e^{-b_0 \alpha_{jl}} \times \frac{d_0^{c_0}}{\Gamma\left(c_0\right)} \beta_{jl}^{c_0-1} e^{-d_0 \beta_{jl}} \end{aligned} \tag{4.7}$$

With enough data $\mathcal{X}_l = \{X_{1l}, X_{2l}, \ldots, X_{Nl}\}$, the posterior distribution could approximated by a product of two independent gamma distribution as [76]

$$\begin{aligned} f(\alpha_{jl}, \beta_{jl} \mid \mathcal{X}_l) &\approx \text{Gam}\left(\alpha_{jl}; a^*, b^*\right) \times \text{Gam}\left(\beta_{jl}; c^*, d^*\right) \\ &= \frac{{b^*}^{a^*}}{\Gamma\left(a^*\right)} \alpha_{jl}^{a^*-1} e^{-b^* \alpha_{jl}} \times \frac{{d^*}^{c^*}}{\Gamma\left(c^*\right)} \beta_{jl}^{c^*-1} e^{-d^* \beta_{jl}} \end{aligned} \tag{4.8}$$

where a^*, b^*, c^* and d^* are the hyperparameters of the posterior distribution, which are obtained by variational Bayes estimation as detailed in [76].

4.1.2 Predictive Distribution of the Mixture Model

The predictive distribution can assess the uncertainty of a new coming observation with respect to the existing dataset. Let $\vec{Y}_i$ be that new observation independent from the existing $\mathcal{Y}$ which is assumed to be generated from $GD\left(\vec{Y}_i \mid \vec{\alpha}_j, \vec{\beta}_j\right)$. The redefined new observation $\vec{X}_i$ obtained using the transformation presented after Eq.4.4 follows a product of Beta distributions. The predictive distribution of $\vec{X}_i$ given $\mathcal{X}$ is

$$f(\vec{X}_i \mid \mathcal{X}) = \int_0^\infty \int_0^\infty \prod_{l=1}^{D} \left[\text{Beta}(X_{il} \mid \alpha_{jl}, \beta_{jl}) f(\alpha_{jl}, \beta_{jl} \mid \mathcal{X}_l)\right] d\vec{\alpha}_j d\vec{\beta}_j \tag{4.9}$$

$$
\begin{aligned}
f(\vec{X}_i \mid \mathcal{X}) &\leq f_{\mathrm{upp}}(\vec{X}_i \mid \mathcal{X}) \\
&= \prod_{l=1}^{D} \left[\frac{1}{X_{il}(1-X_{il})} \frac{b^{*a^*}}{\Gamma(a^*)} \frac{d^{*c^*}}{\Gamma(c^*)} \frac{\Gamma(\alpha_0+\beta_0)}{\Gamma(\alpha_0)\Gamma(\beta_0)} e^{-\alpha_0[\psi(\alpha_0+\beta_0)-\psi(\alpha_0)]-\beta_0[\psi(\alpha_0+\beta_0)-\psi(\beta_0)]} \right] \\
&\quad \times \int_0^\infty \int_0^\infty \prod_{l=1}^{D} \Big[e^{\alpha_{jl}[\psi(\alpha_0+\beta_0)-\psi(\alpha_0)]+\beta_{jl}[\psi(\alpha_0+\beta_0)-\psi(\beta_0)]} \\
&\qquad\qquad X_{il}^{\alpha_{jl}}(1-X_{il})^{\beta_{jl}} \alpha_{jl}^{a^*-1} e^{-b^*\alpha_{jl}} \beta_{jl}^{c^*-1} e^{-d^*\beta_{jl}} \Big] d\vec{\alpha}_j d\vec{\beta}_j \\
&= \prod_{l=1}^{D} \left[\frac{1}{X_{il}(1-X_{il})} \frac{b^{*a^*}}{\Gamma(a^*)} \frac{d^{*c^*}}{\Gamma(c^*)} \frac{\Gamma(\alpha_0+\beta_0)}{\Gamma(\alpha_0)\Gamma(\beta_0)} e^{-\alpha_0[\psi(\alpha_0+\beta_0)-\psi(\alpha_0)]-\beta_0[\psi(\alpha_0+\beta_0)-\psi(\beta_0)]} \right] \\
&\quad \times \prod_{l=1}^{D} \left\{ \int_0^\infty e^{-\alpha_{jl} \underbrace{[b^* - \ln X_{il} - \psi(\alpha_0+\beta_0) + \psi(\alpha_0)]}_{g(X_{il},\alpha_0,\beta_0)}} \alpha_{jl}^{a^*-1} d\alpha_{jl} \right\} \\
&\quad \times \prod_{l=1}^{D} \left\{ \int_0^\infty e^{-\beta_{jl} \underbrace{[d^* - \ln(1-X_{il}) - \psi(\alpha_0+\beta_0) + \psi(\beta_0)]}_{h(X_{il},\alpha_0,\beta_0)}} \beta_{jl}^{c^*-1} d\beta_{jl} \right\}
\end{aligned}
\tag{4.10}
$$

With the analytically tractable posterior distribution in Eq.4.8, we can approximate this predictive distribution as

$$
\begin{aligned}
&f(\vec{X}_i \mid \mathcal{X}) \\
&\quad \approx \int_0^\infty \int_0^\infty \prod_{l=1}^{D} \left[\mathrm{Beta}(X_{il} \mid \alpha_{jl}, \beta_{jl}) \times \frac{b^{*^{a^*}}}{\Gamma(a^*)} \alpha_{jl}^{a^*-1} e^{-b^*\alpha_{jl}} \frac{d^{*^{c^*}}}{\Gamma(c^*)} \beta_{jl}^{c^*-1} e^{-d^*\beta_{jl}} \right] d\vec{\alpha}_j d\vec{\beta}_j \\
&\quad = \prod_{l=1}^{D} \left[\frac{1}{X_{il}(1-X_{il})} \frac{b^{*^{a^*}}}{\Gamma(a^*)} \frac{d^{*^{c^*}}}{\Gamma(c^*)} \right] \\
&\qquad \times \int_0^\infty \int_0^\infty \prod_{l=1}^{D} \left[\frac{\Gamma(\alpha_{jl}+\beta_{jl})}{\Gamma(\alpha_{jl})\Gamma(\beta_{jl})} X_{il}^{\alpha_{jl}} (1-X_{il})^{\beta_{jl}} \alpha_{jl}^{a^*-1} e^{-b^*\alpha_{jl}} \beta_{jl}^{c^*-1} e^{-d^*\beta_{jl}} \right] d\vec{\alpha}_j d\vec{\beta}_j
\end{aligned}
\tag{4.11}
$$

The previous equation involves the Inverse Beta function which logarithm has been proved to be concave [66]. Using that concavity property, the following inequality can be easily obtained by first order Taylor expansion

$$\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} \leq \frac{\Gamma\left(\alpha_0+\beta_0\right)}{\Gamma\left(\alpha_0\right)\Gamma\left(\beta_0\right)} \times e^{[\psi(\alpha_0+\beta_0)-\psi(\alpha_0)](\alpha-\alpha_0)+[\psi(\alpha_0+\beta_0)-\psi(\beta_0)](\beta-\beta_0)} \tag{4.12}$$

where $\psi(\cdot)$ is the digamma function defined as $\psi(\cdot) = \partial \ln \Gamma(x)/\partial x$. Using Eq. 4.12 and a local variational inference (LVI) method [67] we can find an upper bound for the predictive distribution as shown in Eq.4.10. Compared with the global variational inference [76] which approximates all the model's variables, LVI is considered as a 'local' approach to approximate a subset of variables [49].

In Eq.4.10, the integrand in each integration is a Gamma distribution. To simplify the predictive distribution, these integrations can be replaced by

$$\begin{aligned}
&\textstyle\int_0^\infty e^{-\alpha_{jl} g(X_{il},\alpha_0,\beta_0)} \alpha_{jl}^{a^*-1} d\alpha_{jl} \\
&\quad = \begin{cases} \frac{\Gamma(a^*)}{g(X_{il},\alpha_0,\beta_0)^{a^*}} & g\left(X_{il},\alpha_0,\beta_0\right) > 0 \\ \infty & g\left(X_{il},\alpha_0,\beta_0\right) \leq 0 \end{cases} \\
&\textstyle\int_0^\infty e^{-\beta_{jl} h(X_{il},\alpha_0,\beta_0)} \beta_{jl}^{c^*-1} d\beta_{jl} \\
&\quad = \begin{cases} \frac{\Gamma(c^*)}{h(X_{il},\alpha_0,\beta_0)^{c^*}} & h\left(X_{il},\alpha_0,\beta_0\right) > 0 \\ \infty & h\left(X_{il},\alpha_0,\beta_0\right) \leq 0 \end{cases}
\end{aligned} \tag{4.13}$$

Using the previous equation, by obviously considering that $g\left(X_{il},\alpha_0,\beta_0\right) > 0$ and $h\left(X_{il},\alpha_0,\beta_0\right) > 0$, we obtain a closed-form upper bound for the predictive distribution:

$$\begin{aligned}
&f_{\text{upp}}(\vec{X}_i \mid \mathcal{X}) \\
&\quad = \prod_{l=1}^{D} \left[\frac{1}{X_{il}(1-X_{il})} \left[\frac{b^*}{g\left(X_{il},\alpha_0,\beta_0\right)}\right]^{a^*} \left[\frac{d^*}{h\left(X_{il},\alpha_0,\beta_0\right)}\right]^{c^*} \right. \\
&\quad \left. \times \frac{\Gamma\left(\alpha_0+\beta_0\right)}{\Gamma\left(\alpha_0\right)\Gamma\left(\beta_0\right)} e^{-\alpha_0[\psi(\alpha_0+\beta_0)-\psi(\alpha_0)]-\beta_0[\psi(\alpha_0+\beta_0)-\psi(\beta_0)]} \right]
\end{aligned} \tag{4.14}$$

The upper bound is just a function of α_0, β_0 after being given X_{il}, which can be rewritten as

$$f_{\text{upp}}(\vec{X}_i \mid \mathcal{X}) = \prod_{l=1}^{D} \left[\frac{{b^*}^{a^*} {d^*}^{c^*}}{X_{il}(1-X_{il})} \times F(X_{il},\alpha_0,\beta_0) \right] \tag{4.15}$$

where $F(X_{il}, \alpha_0, \beta_0)$ can be straightforwardly deduced from Eq. 4.14. The means $\mathbf{E}(\alpha)$ and $\mathbf{E}(\beta)$ are the most representative values of α_0 and β_0, respectively, which can be taken to approximate the optimal solution (α_0^*, β_0^*). Besides, the means calculated by the observations in $\mathcal{X}$ are independent of X_{il}. To facilitate the calculation, the minimum of the upper bound can be approximated as

$$\begin{aligned} &\min_{\alpha_0,\beta_0} f_{\text{upp}}(\vec{X}_i \mid \mathcal{X}) \\ &\qquad = \prod_{l=1}^{D} \left[\frac{{b^*}^{a^*} {d^*}^{c^*}}{X_{il}(1 - X_{il})} \times \min_{\alpha_0,\beta_0} F(X_{il}, \mathbf{E}(\alpha), \mathbf{E}(\beta)) \right] \\ &\qquad \approx \prod_{l=1}^{D} \left[\frac{{b^*}^{a^*} {d^*}^{c^*}}{X_{il}(1 - X_{il})} \times F(X_{il}, \mathbf{E}(\alpha), \mathbf{E}(\beta)) \right] \end{aligned} \tag{4.16}$$

Since $\min_{\alpha_0,\beta_0} f_{upp}(\vec{X}_i \mid \mathcal{X})$ is unnormalized, we can calculate the normalization factor:

$$C_{upp} = \int_0^1 \min_{\alpha_0,\beta_0} f_{upp}(\vec{X}_i \mid \mathcal{X}) d\vec{X}_i \tag{4.17}$$

The approximation to the mixture predictive distribution is finally obtained as

$$\begin{aligned} &f(\vec{X}_i \mid \mathcal{X}) \approx f_{appx}^{LVI}(\vec{X}_i \mid \mathcal{X}) \\ &= \sum_{j=1}^{M} \frac{\pi_j}{C_{uppj}} \prod_{l=1}^{D} \left[\frac{{b^*_{jl}}^{a^*_{jl}} {d^*_{jl}}^{c^*_{jl}}}{X_{il}(1 - X_{il})} \times F(X_{il}, \mathbf{E}(\alpha_{jl}), \mathbf{E}(\beta_{jl})) \right] \end{aligned} \tag{4.18}$$

4.2 Occupancy Estimation Framework

In this section, we introduce our occupancy estimation framework. The framework summarized in Fig. 4.1 is composed of two main modules: 1) an interactive learning module [36, 77] that allows to get the labels directly from the users and that we summarize briefly in this section, and 2) a predictive modeling one that we have developed in the last section. While interactive learning will allow to ensure to collect a set of training data of good quality with a minimal interaction of the user and to a certain extent self labeling to reduce further the involvement of the user and to retrain the model's hyperparameters, predictive modeling will allow continuous simultaneous occupancy estimation.

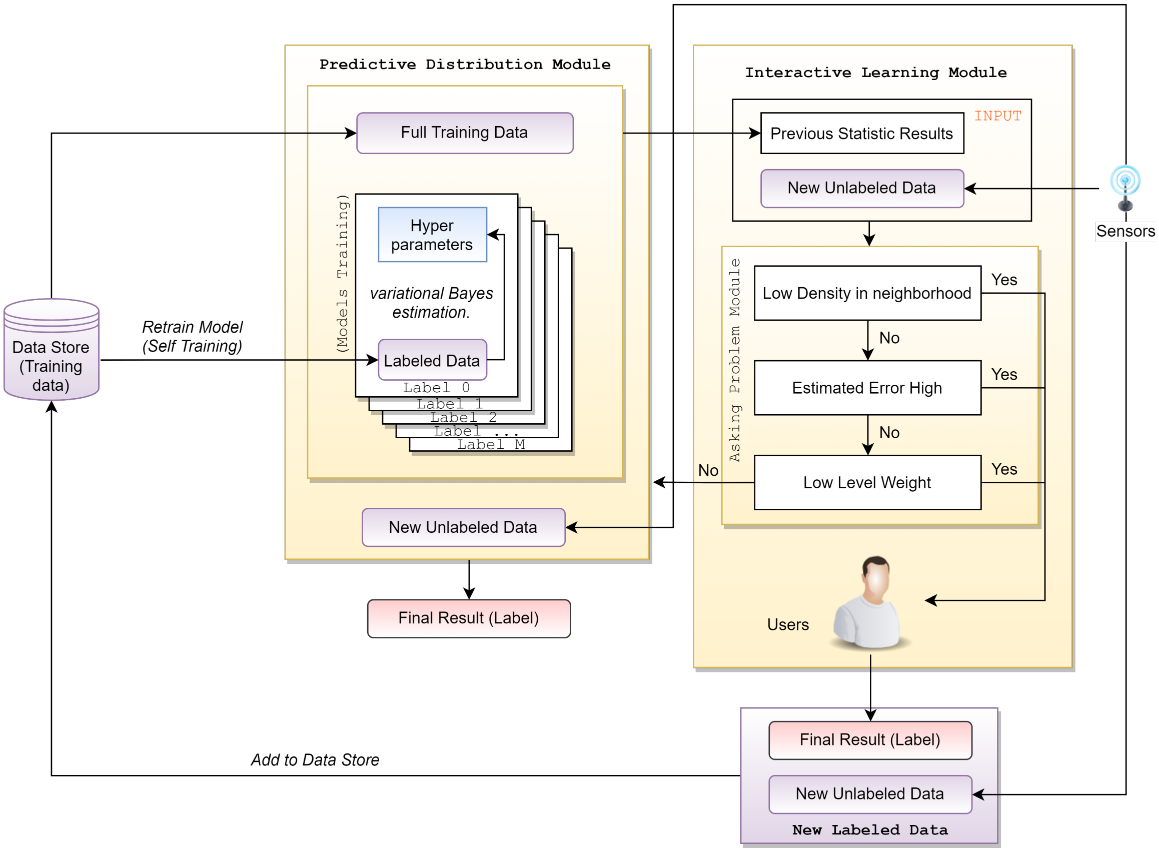


Figure 4.1: Occupancy estimation framework.

The so-called interactive learning is mainly motivated by its proven effectiveness for occupancy estimation as shown in [36, 77]. Interactive learning is a process that allows the exchange of information with users in buildings when needed and it is used here with the ultimate goal to reduce inner product of estimation errors. It could be obviously used at the beginning to start collecting training data to be able to build the predictive statistical model and learn its parameters. It is important to distinguish between interactive learning and active learning [78] in our context. Indeed, while active learning assumes that there is an ”oracle” such as a human expert to get ground-truth labels for selected unlabeled instances (i..e the outputs of the sensors such as motion detection, power consumption or CO_2 concentration) which is very difficult in our case, interactive learning extends supervised learning by collecting the labels directly from the occupants themselves.

The most important problem in interactive learning is determining when to ask the occupants to get their exact number. If the frequency is too high, occupants will feel bored and will not give timely feedback. Besides, the high frequency of inquiry is also the reflection of the weak prediction ability of the model itself. On the opposite side, the inner product of errors will be too high if the interactive learning is not activated when the prediction result of the original model is not satisfactory. In [36], three criteria had been designed to determine when an interaction, called an *ask*, with the user is required: 1) the density of the neighborhood of the data point to label (i.e. the added data point should increase the density), 2) the estimated error in the neighborhood of the potential data point to add to the training data (i..e the added data point should reduce the error), 3) the low level weight: if the weight of a given class is too small and training data is not enough, the prediction ability of the model for the small classes will also be insufficient. So, a minimum number of data points is set as a condition for an *ask*. The three criteria will be checked for each new data point in turn as we can see in Fig.4.1. If any criterion is validated, occupant response has to be taken into account and added to training dataset with the new data point. Furthermore, the hyper parameters of the predictive model that we shall develop in the next section need to be updated using the new training data. For more details about these three criteria and the interactive learning part, the interested reader is referred to [77].

4.3 Experimental Results

In this section, the predictive distribution of GD model is validated using synthetic data. Besides, the predictive distribution model with interactive learning is applied on real-world data that we have collected. We will illustrate our results from three perspectives. First, occupancy estimation with original training data. This is a regular experiment to test the model performance when the full training data is given. Second, occupancy estimation with extremely small training data. Predictive distribution has relatively good performance under small training data. The goal of this perspective is to validate that the predictive distribution of GD mixture model with interactive learning is also good enough when there is only a small supply of training data. Finally, a comprehensive comparative analysis of the above two aspects is extensively detailed and different scenarios are discussed in the third part.

4.3.1 Synthetic Data

In this section, we will use two main variational approaches to approximate the predictive distribution namely GVI and LVI. As we mentioned previously, Eq.4.18 is the predictive distribution of GD based on LVI. The GVI one is the approximation using the posterior mean as the point estimates:

$$\begin{aligned} f(\vec{X}_i \mid \mathcal{X}) &\approx f_{appx}^{GVI}(\vec{X}_i \mid \mathcal{X}) \\ &= \sum_{j=1}^{M} \pi_j \prod_{l=1}^{D} \text{Beta}\left(X_{il} \mid \mathbf{E}(\alpha), \mathbf{E}(\beta)\right) \end{aligned} \tag{4.19}$$

Table 4.1: Comparison of the KL divergences ($\times 10^{-2}$)

Distribution	KL divergences	N = 10	N = 20	N = 50	N = 200	N = 500
$Beta(x; 3, 4)$	$\mathbf{KL}(f\|\|f_{appx}^{GVI})$	36.24	8.60	1.75	0.56	0.20
	$\mathbf{KL}(f\|\|f_{appx}^{LVI})$	**25.80**	**7.59**	**1.43**	**0.54**	**0.25**
$Beta(x; 4, 5) * 0.2$	$\mathbf{KL}(f\|\|f_{appx}^{GVI})$	27.35	12.53	5.25	1.36	0.28
$Beta(x; 2, 8) * 0.8$	$\mathbf{KL}(f\|\|f_{appx}^{LVI})$	**20.69**	**11.76**	**7.03**	**4.45**	**4.20**
$Beta(x; 3, 8) * 0.4$	$\mathbf{KL}(f\|\|f_{appx}^{GVI})$	23.77	14.61	6.82	1.25	0.5005
$Beta(x; 5, 5) * 0.6$	$\mathbf{KL}(f\|\|f_{appx}^{LVI})$	**18.52**	**11.27**	**5.20**	**0.91**	**0.5081**

Kullback-Leibler (KL) divergence was used for the comparison of distributions [53]. The result of KL is smaller when the two distributions being compared are more similar. Because the synthetic data are generated from a known GD distribution, the true distribution (f) is compared with GVI (f_{appx}^{GVI}) and LVI (f_{appx}^{LVI}) distribution to calculate two KL values as in Table 4.1. One single distribution ($Beta(x; 3, 4)$) and two mixture distributions ($Beta(x; 4, 5) * 0.2, Beta(x; 2, 8) * 0.8$ and $Beta(x; 3, 8) * 0.4, Beta(x; 5, 5) * 0.6$) are considered to extract the synthetic data from 10 rounds of simulations. Each round, the synthetic data is grouped and tested from small to large according to the size of data. Table 4.1 shows the mean of KL values. As we can see, the KL values of both GVI and LVI decrease with the increase of training data size, which indicates that the performance of predictive distribution is better when more training data can be given. It is noteworthy that KL divergence of LVI is stably smaller when the amount of training data is under 50, and the performance of GVI is better when the size of training data reaches 500.

4.3.2 Occupancy Estimation

In this section, our statistical framework is applied to tackle occupancy estimation problem with different scenarios. The data used are extracted from a real-life heterogeneous sensor environment. In order to evaluate the performance of the framework, we considered four commonly used metrics: *Precision*, *Recall*, *F-measure* and *Accuracy*:

$$\mathit{Precision}(P) = \frac{TP}{TP + FP} \tag{4.20}$$

$$\mathit{Recall}(R) = \frac{TP}{TP + FN} \tag{4.21}$$

$$\mathit{F\text{-}measure}(F1) = \frac{2 \cdot P \cdot R}{P + R} \tag{4.22}$$

$$\mathit{Accuracy}(\mathit{Acc.}) = \frac{TP + FN}{TP + FP + TN + FN} \tag{4.23}$$

where *TP*, *TN*, *FP*, and *FN* are true-positives, true-negatives, false-positives and false-negatives, respectively. These metrics except accuracy are calculated for each label because of the imbalanced nature of the training data as we shall see later, and find their average weighted by the number of true instances for each label. Two kinds of accuracy are considered. The first is the total accuracy on the full training dataset ($\mathit{Acc.T}$) and the second is the average accuracy in terms of classes ($\mathit{Acc.A}$).

Dataset Description, Feature Selection and Preprocessing Procedure

The same real data used in this section is also collected from the test bed (Fig.3.4), in Grenoble Institute of Technology, described in [2] where the authors considered an office which is monitored by many sensors (e.g. temperature, relative humidity, motions, CO2, power consumption, door and window positions, acoustic pressure from microphone, etc.). Two cameras are used to record the true occupancy numbers. During the data capture period, a professor and 3 PhD students used the office, and the number of people in the room was uncertain.

After calculation of information gains and the analysis of decision trees by the authors, the top 4 most important features were selected and shall be used in this experiment to test the framework proposed in this paper. These features are Motion, Power Consumption, Acoustic Pressure (the microphone just provides the amplitude of the sound in dB) and Door Opening. Each data point in the dataset is based on an interval of 30 mins (referred to as 1 quantum). Motion counter is a PIR sensor which outputs a binary value and reports 1 whenever some motions are detected. The number of motions is the value of Motion in 1 quantum. There are 4 sensors connected to inhabitant laptops in the office. They are also binary sensors set to report 1 when the voltage is higher than 15W. The value of Power Consumption is how many laptops' voltage exceeded 15W. Acoustic Pressure is the root mean square (or average) of the amplitude of a sound. Door Opening is a state quantity: 0 means the door was always closed and 1 was always opened. The fourth feature corresponds then to the time ratio of the door opened during time quantum.

The full dataset is composed of 718 four-dimensional labeled vectors $\langle f_1, f_2, f_3, f_4; y\rangle$. Besides, the data directly obtained from sensors have generally different degrees of magnitude. We have used L2-norm to independently normalize each sample. Finally, we used cross-validation. The different data classes were divided into training data and test data in a ratio of 0.7:0.3.

Occupancy Estimation with Original Training Data

The original training data is imbalanced on labels as we can see from Table 3.1 (Label 0: 352, Label 1: 82, Label 2: 42, Label 3: 18, Label 4: 6). In this section, we considered this data to test our model with and without interactive learning. The results are shown in Fig.4.2. The experimental uncertainty can influence the numerical comparison. Random seeds are helpful to ensure the consistency of random selection between the cases with and without interactive learning. Fig.4.2 displays the average result of 10 experiments (seed 1-10) using real data. In the bar chart, x-axis represents the different labels meaning the occupancy number in the test bed. For a single label, the blue bar is the total amount of test data. The orange and green bars represent how many samples are predicted correctly using the model with and without interactive learning.

The total accuracy without interactive learning is 78.52%, and the single accuracy of label 0 is much better than that of labels 1, 2, 3 and 4. This can be explained by

Table 4.2: Comparison table of occupancy estimation using different training datasets and frameworks, where **6I** and **6** mean that the training uses 6 samples in each class with and without interactive learning. **FI** and **F** mean that the training uses all samples with and without interactive learning.

Training Dataset and Framework	*Acc.T*	*Acc.A*	*P*	*R*	*F1*
6I	84.10%	76.73%	90.34%	84.10%	85.46%
6	63.92%	42.91%	78.40%	63.92%	64.41%
FI	87.51%	75.05%	92.58%	87.51%	88.45%
F	78.52%	51.34%	84.70%	78.52%	79.78%

the higher amount of label 0 in the training data. The model is then reinforced after adding interactive learning module. The total accuracy has been improved to 87.51% with 36.3 asks. The accuracy of all classes has been improved in varying degrees especially labels 1, 2, 3, and 4. The average accuracy in terms of classes is increased from 51.34% to 75.05%. The trends of *Precision*, *Recall* and *F-measure* are similar to the *Accuracy* trend.

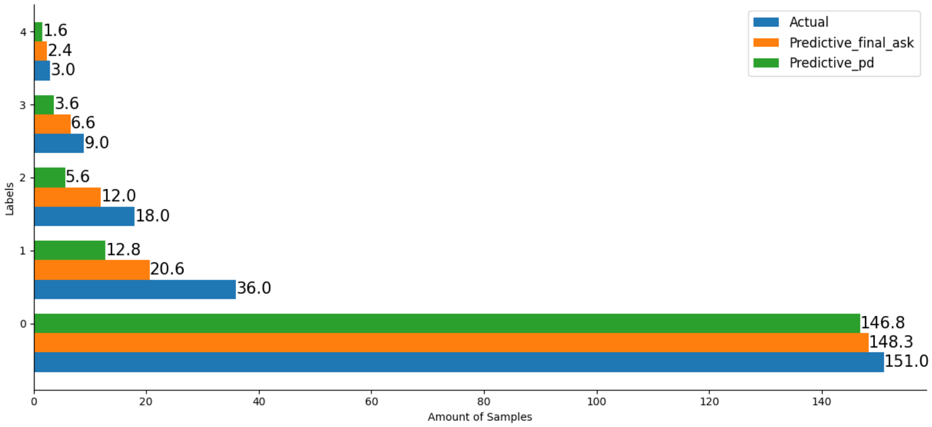


Figure 4.2: The results using original training data with and without interactive learning (average values obtained from 10 experiments).

Occupancy Estimation with Extremely Small Training Data

With synthetic data, the results prove that KL divergence between LVI-based predictive distribution and true one is better than GVI-based variational learning when training data is extremely small. Thus, it is valuable to test the performance of the model with interactive learning. The smallest class in original training data contains 6 samples. The rest of classes are all reduced to 6 samples before the experiments.

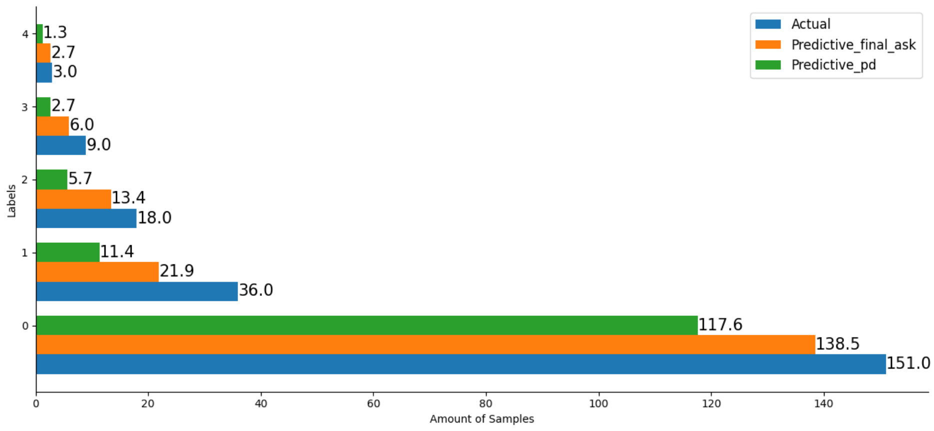


Figure 4.3: The results using 6 training data in each class with and without interactive learning (average values obtained from 10 experiments).

Fig.4.3 displays the results in this scenario. The total accuracy of the model using 6 samples in each class improved from 63.92% to 84.10% after combining the interactive learning with 64.2 asks. In addition, the average accuracy in terms of labels significantly increased from 42.91% to 76.73% which is beyond the average accuracy of the model with original training data. Even though the total accuracy under the small training data is not better than that using original training data, 84.10% is an encouraging accuracy for occupancy estimation when there are only few labeled data for training. From Table 4.2, it is clear that the performance of the model using an extremely small training data without interactive learning is far worse than that using original training data, which is expected since real data are more complex and uncertain as compared with synthetic data. However, the small weighted class sometimes performs better after reducing all classes to 6 samples. For example, the result of Label 2 (5.7) in Fig. 4.3 is higher than that (5.6) in Fig. 4.2

without interactive learning module. The balanced training dataset of Fig. 4.3 is the reason for this situation. As in Table 4.2, the difference of all metrics between small training data and original training data are obviously decreasing after using interactive learning. Consequently, interactive learning is an essential module when there are not too many training data at the beginning of the estimation task.

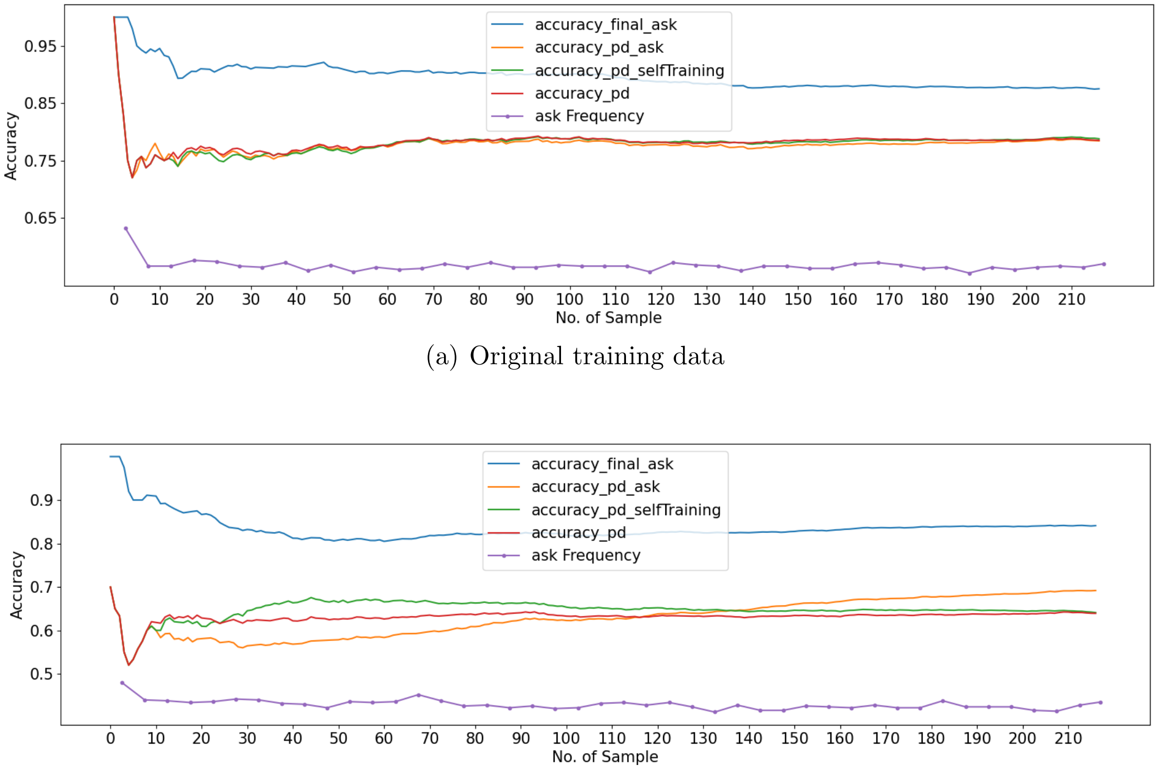
Comparative Analysis

The authors in [3] proposed a variational learning of a shifted scaled Dirichlet mixture model with component splitting approach (VSSDMM) to estimate occupancy using the same real data. They also assessed the experimental results using *Accuracy*, *Precision*, *Recall* and *F-measure* and performed a comparison with the results of variational learning of Dirichlet mixture model (VDMM) and variational learning of Gaussian mixture model (VGMM). The results are in Table 4.3, in which VSSDMM model is shown to be always better than VDMM and VGMM. It is noteworthy that, for all four metrics, our framework with interactive learning provided better results than VSSDMM, especially for *Precision*.

Table 4.3: Comparison Table of Occupancy Estimation using VSSDMM, VDMM and VGMM [3].

Framework	*Acc.*	*P*	*R*	*F1*
VSSDMM	81.72%	79.87%	81.72%	79.44%
VDMM	79.14%	77.31%	79.14%	78.21%
VGMM	78.03%	76.11%	78.03%	77.06%

In this section, we also considered the change of accuracy with the increase of ask times to compare the performance of the model under original training data and 6 training data in each class. In Fig. 4.4, 4 kinds of accuracy and ask frequency are averaged over 10 experiments and fluctuate with the increase of test samples. The results in Fig.4.2 and Fig.4.3 are the comparison of **accuracy_pd** (red line) and **accuracy_final_ask** (blue line) in Fig. 4.4.a and Fig. 4.4.b. The accuracy of the starting point (test 0) is actually just 0 or 1. But all the accuracy of lines in Fig. 4.4 are the average values of 10 experiments using the real data and the starting points are maybe different. **accuracy_final_ask** and **accuracy_pd** are obtained from the



(a) Original training data

(b) 6 training data in each class

Figure 4.4: The fluctuation of accuracy with the increase of ask times.

predictive distribution of GD model with and without interactive learning. However, the **accuracy_final_ask** is the revised result by interactive learning. It is difficult to see the changes that the interactive learning module brings to the result of the predictive distribution model. Thus, **accuracy_pd_ask** (orange line) records the unrevised result. The comparison between **accuracy_pd** and **accuracy_pd_ask** can show how the interactive learning improves the accuracy of the predictive distribution model by adding new data from interaction to training data and updating hyper parameters. The final accuracy is **accuracy_pd_selfTraining**. The result of each test is considered as a new labeled vector added to the training data. The y-axis of Fig. 4.4 does not match the **ask Frequency** (purple line), which is just a reference for observing the impact of ask frequency on accuracy. The **ask Frequency** is the average value of ask times at every five samples after 10 experiments.

With original training data, the **accuracy_pd_ask**, **accuracy_pd_selfTraining** and **accuracy_pd** are similar in the experiment. But, the **accuracy_pd_ask** exceeds the **accuracy_pd** and obviously has a better accuracy at the end of experiments with 6 training data in each class. With the rapid accumulation of training data, **accuracy_pd_selfTraining** is always better than **accuracy_pd** and **accuracy_pd_ask** before sample 130 in Fig.4.4.b, but gradually decreases and equals **accuracy_pd** because of the inner product of errors. No matter the size of training data, **accuracy_final_ask** always maintained an advantage of almost 10 percentage points compared with the other accuracy, and **ask Frequency** gradually decreases along with the experimental training and self labeling using the results of interactive learning, which satisfies the requirement of practical application. As we can see from Fig.4.4, while the interactive learning module is not helpful when enough training data (original training data) is given, it is essential under extremely small training data condition.

Chapter 5

Conclusion

In this thesis, we developed different mixture models based predictive distributions for activity recognition and occupancy estimation. Over-sampling and interactive learning as extra modules are introduced to improve the performance of predictive modeling.

In chapter 2, we presented an elegant principled statistical framework for predictive modeling based on the GID distribution and a local variational inference. The proposed approach is motivated first by the flexibility of the GID distribution when modeling semi-bounded positive vectors that are naturally generated by several applications involving sensors outputs and second by the efficiency of local variational inference when the amount of training data is limited. Extensive simulations based on synthetic data as well as a challenging real application that concerns activity recognition have shown the merits of our approach.

In chapter 3, we introduced a model based on the predictive distribution of the inverted Dirichlet mixture model to tackle the challenging problem of occupancy estimation in smart buildings. This model is mainly motivated by its prediction ability when only small training data are available. Moreover, an over-sampling approach is introduced to handle imbalanced data that we generally face in occupancy estimation.The model is shown to be promising and to provide impressive performance under small training data.

Finally, in chapter 4, we developed an occupancy estimation methodology in smart buildings in the commonly faced critical barriers for a practical implementation related to: 1) the knowledge of the stochasticity of occupancy and its modeling, 2) the non-Gaussianity of the generated data from the deployed sensors, 3) the necessity to have a large amount of labeled data to obtain a proper fit of the occupancy distribution, 4) the difficulty to obtain such labeled data since the labelling must be performed by the inhabitants. To tackle the first two challenges a mixture of generalized Dirichlet distributions is deployed. The third challenge is approached by developing the generalized Dirichlet mixture predictive distribution to allow reliable labeling of the data while interacting in an efficient manner with the users to get their output in order to face the fourth challenge. The proposed approach can be viewed as an elegant principled statistical framework for estimating occupancy in smart buildings using a set of easy-to-install, low-cost and small sensors. Our results suggest that our approach, which can be perceived as less invasive than camera-based ones, provides promising results without requiring a strong supervision information (i.e. fully ground-truth labels) which is difficult to get due to its high cost in terms of users involvement.

Potential future works could be devoted to developing a principled approach for the hyperparameters optimization like the one proposed in [79] within our occupancy estimation framework. The results of this research will be useful to tackle a future problem related to activities recognition and prediction with the ultimate goal to integrate all the work for a practical implementation of a smart energy management system to automatically control energy consumption that puts constantly the users in the loop without affecting their comfort nor their privacy.

Appendix A

Proof of Equation 3.10

The logarithm of the Multivariate-Inverse-Beta has been proved to be concave [44]. Thus, the following inequality can be easily obtained by first order Taylor expansion

$$\begin{aligned}\ln \frac{\Gamma(\sum_{d=1}^{D+1}\alpha_d)}{\prod_{d=1}^{D+1}\Gamma(\alpha_d)} \leq{} & \ln \frac{\Gamma(\sum_{d=1}^{D+1}\tilde{\alpha}_d)}{\prod_{d=1}^{D+1}\Gamma(\tilde{\alpha}_d)} \\ & + \sum_{d=1}^{D+1}\left[\psi\left(\sum_{d=1}^{D+1}\tilde{\alpha}_d\right) - \psi(\tilde{\alpha}_d)\right](\alpha_d - \tilde{\alpha}_d)\end{aligned} \tag{A.1}$$

where $\tilde{\alpha}_d, k = 1, 2, ..., D+1$ is any point from the posterior distribution. Taking the exponential of both sides, we have

$$\begin{aligned}\frac{\Gamma(\sum_{d=1}^{D+1}\alpha_d)}{\prod_{d=1}^{D+1}\Gamma(\alpha_d)} \leq{} & \frac{\Gamma(\sum_{d=1}^{D+1}\tilde{\alpha}_d)}{\prod_{d=1}^{D+1}\Gamma(\tilde{\alpha}_d)} \\ & \times e^{\sum_{d=1}^{D+1}\left[\psi\left(\sum_{d=1}^{D+1}\tilde{\alpha}_d\right) - \psi(\tilde{\alpha}_d)\right](\alpha_d - \tilde{\alpha}_d)}\end{aligned} \tag{A.2}$$

By substituting (A.2) into (3.6) and with some mathematical manipulations, we can obtain the following upper-bound

$$
\begin{aligned}
f(\mathbf{x}|\mathbf{X}) \leq & \int \cdots \int \frac{\Gamma(\sum_{d=1}^{D+1}\tilde{\alpha}_d)}{\prod_{d=1}^{D+1}\Gamma(\tilde{\alpha}_d)} \\
& \times e^{\sum_{d=1}^{D+1}\left[\psi\left(\sum_{d=1}^{D+1}\tilde{\alpha}_d\right)-\psi(\tilde{\alpha}_d)\right](\alpha_d-\tilde{\alpha}_d)} \\
& \times x_1^{\alpha_1-1}\frac{(v_1^*)^{u_1^*}}{\Gamma(u_1^*)}\alpha_1^{u_1^*-1}e^{-v_1^*\alpha_1}(1+\sum_{d=1}^{D}x_d)^{-\alpha_1} \\
& \cdots \\
& \times x_{D+1}^{\alpha_{D+1}-1}\frac{(v_{D+1}^*)^{u_{D+1}^*}}{\Gamma(u_{D+1}^*)}\alpha_{D+1}^{u_{D+1}^*-1} \\
& \quad e^{-v_{D+1}^*\alpha_{D+1}}(1+\sum_{d=1}^{D}x_k)^{-\alpha_{D+1}}d\alpha_1 ... d\alpha_{D+1} \qquad \text{(A.3)} \\
= & \frac{\Gamma(\sum_{d=1}^{D+1}\tilde{\alpha}_d)}{\prod_{d=1}^{D+1}\Gamma(\tilde{\alpha}_d)} \times e^{-\sum_{d=1}^{D+1}\tilde{\alpha}_d\left[\psi\left(\sum_{d=1}^{D+1}\tilde{\alpha}_d\right)-\psi(\tilde{\alpha}_d)\right]} \\
& \times \prod_{d=1}^{D+1}\frac{(v_d^*)^{u_d^*}}{x_d\Gamma(u_d^*)} \\
& \int e^{-\alpha_d\left[v_d^*-\ln x_d-\psi\left(\sum_{d=1}^{D+1}\tilde{\alpha}_d\right)+\psi(\tilde{\alpha}_d)+\ln\left(1+\sum_{d=1}^{D}x_d\right)\right]} \\
& \times \alpha_d^{u_d^*-1}d\alpha_d \\
\approx & f_{upp}(\mathbf{x}|\mathbf{X})
\end{aligned}
$$

For simplicity let's denote

$$
\mathbf{G}(x_d,\tilde{\alpha}) = v_d^* - \ln x_d - \psi\left(\sum_{d=1}^{D+1}\tilde{\alpha}_d\right) + \psi(\tilde{\alpha}_d) + \ln\left(1+\sum_{d=1}^{D}x_d\right) \qquad \text{(A.4)}
$$

where $d = 1, 2, ..., D+1$. Thus, the integration in Eq.A.3 has a same form as Gamma function and could be reduced to

$$
\int e^{-\alpha_d\mathbf{G}(x_d,\tilde{\alpha})}\alpha_d^{u_d^*-1}d\alpha_d = \begin{cases} \dfrac{\Gamma(u_d^*)}{[\mathbf{G}(x_d,\tilde{\alpha})]^{u_d^*}} & \mathbf{G}(x_d,\tilde{\boldsymbol{\alpha}}) > 0 \\ \infty & \mathbf{G}(x_d,\tilde{\boldsymbol{\alpha}}) \leq 0 \end{cases} \qquad \text{(A.5)}
$$

Here, we attempt $\mathbf{G}(x_d,\tilde{\alpha}) > 0$ for any d because the situation of $\mathbf{G}(x_d,\tilde{\alpha}) \leq 0$ is unsolvable. Finally, the analytically tractable form of finite upper-bound of the predictive distribution is

$$
\begin{aligned}
f_{upp}(\mathbf{x}|\mathbf{X}) = & \frac{\Gamma(\sum_{d=1}^{D+1} \tilde{\alpha}_d)}{\prod_{d=1}^{D+1} \Gamma(\tilde{\alpha}_d)} \times e^{-\sum_{d=1}^{D+1} \tilde{\alpha}_d \left[\psi\left(\sum_{d=1}^{D+1} \tilde{\alpha}_d\right) - \psi(\tilde{\alpha}_d)\right]} \\
& \times \prod_{d=1}^{D+1} \frac{(v_d^*)^{u_d^*}}{x_d \left[\mathbf{G}(x_d, \tilde{\boldsymbol{\alpha}})\right]^{u_d^*}}
\end{aligned}
\tag{A.6}
$$

Bibliography

[1] Filippo Palumbo, Claudio Gallicchio, Rita Pucci, and Alessio Micheli. Human activity recognition using multisensor data fusion based on reservoir computing. *Journal of Ambient Intelligence and Smart Environments*, 8(2):87–107, 2016.

[2] Manar Amayri, Abhay Arora, Stephane Ploix, Sanghamitra Bandhyopadyay, Quoc-Dung Ngo, and Venkata Ramana Badarla. Estimating occupancy in heterogeneous sensor environment. *Energy and Buildings*, 129:46–58, 2016.

[3] Narges Manouchehri, Oumayma Dalhoumi, Manar Amayri, and Nizar Bouguila. Variational learning of a shifted scaled dirichlet model with component splitting approach. In *IEEE International Conference on Artificial Intelligence for Industries (AI4I)*, pages 75–78. IEEE, 2020.

[4] A. Ouammi, Y. Achour, D. Zejli, and H. Dagdougui. Supervisory model predictive control for optimal energy management of networked smart greenhouses integrated microgrid. *IEEE Transactions on Automation Science and Engineering*, 17(1):117–128, 2020.

[5] Z. Nie, F. Gao, C. Yan, and X. Guan. Multi-timescale decision and optimization for hvac control systems with consistency goals. *IEEE Transactions on Automation Science and Engineering*, 17(1):296–309, 2020.

[6] D. Li, Y. Zhou, G. Hu, and C. J. Spanos. Handling incomplete sensor measurements in fault detection and diagnosis for building hvac systems. *IEEE Transactions on Automation Science and Engineering*, 17(2):833–846, 2020.

[7] R. Carli and M. Dotoli. A dynamic programming approach for the decentralized control of energy retrofit in large-scale street lighting systems. *IEEE Transactions on Automation Science and Engineering*, 17(3):1140–1157, 2020.

[8] L. Li, C. Li, Y. Tang, L. Li, and X. Chen. An integrated solution to minimize the energy consumption of a resource-constrained machining system. *IEEE Transactions on Automation Science and Engineering*, 17(3):1158–1175, 2020.

[9] Simona D'Oca, Tianzhen Hong, and Jared Langevin. The human dimensions of energy use in buildings: A review. *Renewable and Sustainable Energy Reviews*, 81:731 – 742, 2018.

[10] K. Viard, M. P. Fanti, G. Faraut, and J. J. Lesage. Human activity discovery and recognition using probabilistic finite-state automata. *IEEE Transactions on Automation Science and Engineering*, 17(4):2085–2096, 2020.

[11] Asma Benmansour, Abdelhamid Bouchachia, and Mohammed Feham. Multioccupant activity recognition in pervasive smart home environments. *ACM Comput. Surv.*, 48(3):34:1–34:36, 2016.

[12] Y. Yan, P. B. Luh, and K. R. Pattipati. Fault prognosis of key components in hvac air-handling systems at component and system levels. *IEEE Transactions on Automation Science and Engineering*, 17(4):2145–2153, 2020.

[13] Frauke Oldewurtel, David Sturzenegger, and Manfred Morari. Importance of occupancy information for building climate control. *Applied Energy*, 101:521 – 532, 2013.

[14] Tuan Anh Nguyen and Marco Aiello. Energy intelligent buildings based on user activity: A survey. *Energy and Buildings*, 56:244 – 257, 2013.

[15] Y. Yang, G. Hu, and C. J. Spanos. Hvac energy cost optimization for a multizone building via a decentralized approach. *IEEE Transactions on Automation Science and Engineering*, 17(4):1950–1960, 2020.

[16] T. Li, Y. Chien, C. Chou, C. Liao, W. Cheah, L. Fu, C. C. Chen, C. Chou, and I. Chen. A fast and low-cost repetitive movement pattern indicator for massive dementia screening. *IEEE Transactions on Automation Science and Engineering*, 17(2):771–783, 2020.

[17] A.R. Al-Ali, Imran A. Zualkernan, Mohammed Rashid, Ragini Gupta, and Mazin Alikarar. A smart home energy management system using iot and big data

analytics approach. *IEEE Transactions on Consumer Electronics*, 63(4):426–434, 2017.

[18] Ramón Alcarria, Borja Bordel, Diego Martín, and Diego Sánchez De Rivera. Rule-based monitoring and coordination of resource consumption in smart communities. *IEEE Transactions on Consumer Electronics*, 63(2):191–199, 2017.

[19] Y. Huang, X. Guan, H. Chen, Y. Liang, S. Yuan, and T. Ohtsuki. Risk assessment of private information inference for motion sensor embedded iot devices. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 4(3):265–275, 2020.

[20] Jinsoo Han, Chang-sic Choi, Wan-ki Park, Ilwoo Lee, and Sang-ha Kim. Smart home energy management system including renewable energy based on zigbee and plc. *IEEE Transactions on Consumer Electronics*, 60(2):198–202, 2014.

[21] Y. Tang, C. Li, A. Matta, and Q. Chang. Special issue on intelligent energy solutions to sustainable production and service automation. *IEEE Transactions on Automation Science and Engineering*, 18(2):615–617, 2021.

[22] S. M. Hosseini, R. Carli, and M. Dotoli. Robust optimal energy management of a residential microgrid under uncertainties on demand and renewable power generation. *IEEE Transactions on Automation Science and Engineering*, 18(2):618–637, 2021.

[23] H. Yuan, J. Bi, M. Zhou, Q. Liu, and A. C. Ammari. Biobjective task scheduling for distributed green data centers. *IEEE Transactions on Automation Science and Engineering*, 18(2):731–742, 2021.

[24] Guest editorial: Special issue on computational intelligence for smart energy applications to smart cities. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 3(3):173–176, 2019.

[25] B. Rajasekhar, N. Pindoriya, W. Tushar, and C. Yuen. Collaborative energy management for a residential community: A non-cooperative and evolutionary approach. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 3(3):177–192, June 2019.

[26] Dae-man Han and Jae-hyun Lim. Smart home energy management system using ieee 802.15.4 and zigbee. *IEEE Transactions on Consumer Electronics*, 56(3):1403–1410, 2010.

[27] Zawar Hussain, Quan Z Sheng, and Wei Emma Zhang. A review and categorization of techniques on device-free human activity recognition. *Journal of Network and Computer Applications*, 167:102738, 2020.

[28] Faisal Hussain, Muhammad Basit Umair, Muhammad Ehatisham-ul Haq, Ivan Miguel Pires, Tânia Valente, Nuno M Garcia, and Nuno Pombo. An efficient machine learning-based elderly fall detection algorithm. *arXiv preprint arXiv:1911.11976*, 2019.

[29] Wen Qi, Hang Su, Fei Chen, Xuanyi Zhou, Yan Shi, Giancarlo Ferrigno, and Elena De Momi. Depth vision guided human activity recognition in surgical procedure using wearable multisensor. In *2020 5th International Conference on Advanced Robotics and Mechatronics (ICARM)*, pages 431–436. IEEE, 2020.

[30] Daohua Pan, Hongwei Liu, Dongming Qu, and Zhan Zhang. Human falling detection algorithm based on multisensor data fusion with svm. *Mobile Information Systems*, 2020, 2020.

[31] Tianna-Kaye AE Woodstock. *Multisensor Fusion for Occupancy Detection and Activity Recognition in a Smart Room*. Rensselaer Polytechnic Institute, 2020.

[32] Shaoxiong Sun, Amos A Folarin, Yatharth Ranjan, Zulqarnain Rashid, Pauline Conde, Callum Stewart, Nicholas Cummins, Faith Matcham, Gloria Dalla Costa, Sara Simblett, et al. Using smartphones and wearable devices to monitor behavioral changes during covid-19. *Journal of Medical Internet Research*, 22(9):e19992, 2020.

[33] Debadyuti Mukherjee, Riktim Mondal, Pawan Kumar Singh, Ram Sarkar, and Debotosh Bhattacharjee. Ensemconvnet: a deep learning approach for human activity recognition using smartphone sensors for healthcare applications. *Multimedia Tools and Applications*, 79(41):31663–31690, 2020.

[34] Sarvesh Kumar Swarnakar, Harshit Agrawal, and Ankita Goel. Smartphone inertial sensors-based human activity detection using support vector machine. In *Soft Computing: Theories and Applications*, pages 231–241. Springer, 2021.

[35] Djamel Djenouri, Roufaida Laidi, Youcef Djenouri, and Ilangko Balasingham. Machine learning for smart building applications: Review and taxonomy. *ACM Comput. Surv.*, 52(2), March 2019.

[36] Manar Amayri, Stéphane Ploix, Nizar Bouguila, and Frédéric Wurtz. Estimating occupancy using interactive learning with a sensor environment: Real-time experiments. *IEEE Access*, 7:53932–53944, 2019.

[37] H M Sajjad Hossain, Md Abdullah Al Hafiz Khan, and Nirmalya Roy. Active learning enabled activity recognition. *Pervasive and Mobile Computing*, 38:312 – 330, 2017. Special Issue IEEE International Conference on Pervasive Computing and Communications (PerCom) 2016.

[38] Tom Diethe, Niall Twomey, and Peter A Flach. Active transfer learning for activity recognition. In *ESANN*, 2016.

[39] M. Li, P. Zhou, Y. Liu, and H. Wang. Data-driven predictive probability density function control of fiber length stochastic distribution shaping in refining process. *IEEE Transactions on Automation Science and Engineering*, 17(2):633–645, 2020.

[40] W. Shao, Z. Ge, L. Yao, and Z. Song. Bayesian nonlinear gaussian mixture regression and its application to virtual sensing for multimode industrial processes. *IEEE Transactions on Automation Science and Engineering*, 17(2):871–885, 2020.

[41] Sabri Boutemedjet, Djemel Ziou, and Nizar Bouguila. Unsupervised feature selection for accurate recommendation of high-dimensional image data. In John C. Platt, Daphne Koller, Yoram Singer, and Sam T. Roweis, editors, *Advances in Neural Information Processing Systems 20, Proceedings of the Twenty-First Annual Conference on Neural Information Processing Systems, Vancouver, British Columbia, Canada, December 3-6, 2007*, pages 177–184. Curran Associates, Inc., 2007.

[42] Wentao Fan, Nizar Bouguila, and Djemel Ziou. Variational learning of finite dirichlet mixture models using component splitting. *Neurocomputing*, 129:3–16, 2014.

[43] Wentao Fan, Nizar Bouguila, and Xin Liu. A nonparametric bayesian learning model using accelerated variational inference and feature selection. *Pattern Anal. Appl.*, 22(1):63–74, 2019.

[44] Zhanyu Ma, Arne Leijon, Zheng-Hua Tan, and Sheng Gao. Predictive distribution of the dirichlet mixture model by local variational inference. *Journal of Signal Processing Systems*, 74(3):359–374, 2014.

[45] Edward Snelson and Zoubin Ghahramani. Compact approximations to bayesian predictive distributions. In *Proceedings of the 22nd International Conference on Machine Learning*, ICML '05, pages 840–847, New York, NY, USA, 2005. Association for Computing Machinery.

[46] Andrew Gelman, Xiao li Meng, and Hal Stern. Posterior predictive assessment of model fitness via realized discrepancies. *Statistica Sinica*, pages 733–807, 1996.

[47] Sandip Sinharay and Hal S Stern. Posterior predictive model checking in hierarchical models. *Journal of Statistical Planning and Inference*, 111(1):209 – 221, 2003.

[48] Jan F. Bjornstad. Predictive likelihood: A review. *Statistical Science*, 5(2):242–254, 1990.

[49] Christopher M Bishop. *Pattern recognition and machine learning*. springer, 2006.

[50] Z. Min, J. Wang, and M. Q. . Meng. Robust generalized point cloud registration with orientational data based on expectation maximization. *IEEE Transactions on Automation Science and Engineering*, 17(1):207–221, 2020.

[51] Z. Min, J. Wang, and M. Q. . Meng. Joint rigid registration of multiple generalized point sets with hybrid mixture models. *IEEE Transactions on Automation Science and Engineering*, 17(1):334–347, 2020.

[52] F. Chu, B. Dai, X. Ma, F. Wang, and B. Ye. A minimum-cost modeling method for nonlinear industrial process based on multimodel migration and bayesian

model averaging method. *IEEE Transactions on Automation Science and Engineering*, 17(2):947–956, 2020.

[53] J. Wang and C. Zhao. A gaussian feature analytics-based dissim method for fine-grained non-gaussian process monitoring. *IEEE Transactions on Automation Science and Engineering*, 17(4):2175–2181, 2020.

[54] Sabri Boutemedjet, Djemel Ziou, and Nizar Bouguila. Model-based subspace clustering of non-gaussian data. *Neurocomputing*, 73(10-12):1730–1739, 2010.

[55] Rim Nasfi, Manar Amayri, and Nizar Bouguila. A novel approach for modeling positive vectors with inverted dirichlet-based hidden markov models. *Knowledge-Based Systems*, 192:105335, 2020.

[56] Sami Bourouis, Mohamed Al Mashrgy, and Nizar Bouguila. Bayesian learning of finite generalized inverted dirichlet mixtures: Application to object classification and forgery detection. *Expert Syst. Appl.*, 41(5):2329–2336, 2014.

[57] Mohamed Al Mashrgy, Taoufik Bdiri, and Nizar Bouguila. Robust simultaneous positive data clustering and unsupervised feature selection using generalized inverted dirichlet mixture models. *Knowl. Based Syst.*, 59:182–195, 2014.

[58] Taoufik Bdiri and Nizar Bouguila. Learning inverted dirichlet mixtures for positive data clustering. In Sergei O. Kuznetsov, Dominik Slezak, Daryl H. Hepting, and Boris G. Mirkin, editors, *Rough Sets, Fuzzy Sets, Data Mining and Granular Computing - 13th International Conference, RSFDGrC 2011, Moscow, Russia, June 25-27, 2011. Proceedings*, volume 6743 of *Lecture Notes in Computer Science*, pages 265–272. Springer, 2011.

[59] Nitesh V. Chawla, Kevin W. Bowyer, Lawrence O. Hall, and W. Philip Kegelmeyer. Smote: Synthetic minority over-sampling technique. *J. Artif. Int. Res.*, 16(1):321?357, June 2002.

[60] Giovanna Menardi and Nicola Torelli. Training and assessing classification rules with imbalanced data. *Data Min. Knowl. Discov.*, 28(1):92–122, 2014.

[61] Wentao Fan, Hassen Sallay, and Nizar Bouguila. Online learning of hierarchical pitman-yor process mixture of generalized dirichlet distributions with feature selection. *IEEE Trans. Neural Networks Learn. Syst.*, 28(9):2048–2061, 2017.

[62] Tarek Elguebaly and Nizar Bouguila. Simultaneous bayesian clustering and feature selection using rjmcmc-based learning of finite generalized dirichlet mixture models. *Signal Process.*, 93(6):1531–1546, 2013.

[63] Wentao Fan and Nizar Bouguila. Variational learning of dirichlet process mixtures of generalized dirichlet distributions and its applications. In Shuigeng Zhou, Songmao Zhang, and George Karypis, editors, *Advanced Data Mining and Applications, 8th International Conference, ADMA 2012, Nanjing, China, December 15-18, 2012. Proceedings*, volume 7713 of *Lecture Notes in Computer Science*, pages 199–213. Springer, 2012.

[64] Sami Bourouis, Mohamed Al Mashrgy, and Nizar Bouguila. Bayesian learning of finite generalized inverted dirichlet mixtures: Application to object classification and forgery detection. *Expert Systems with Applications*, 41(5):2329–2336, 2014.

[65] Taoufik Bdiri, Nizar Bouguila, and Djemel Ziou. Variational bayesian inference for infinite generalized inverted dirichlet mixtures with feature selection and its application to clustering. *Applied Intelligence*, 44(3):507–525, 2016.

[66] Stephen Boyd, Stephen P Boyd, and Lieven Vandenberghe. *Convex optimization.* Cambridge university press, 2004.

[67] Zhanyu Ma and Arne Leijon. Approximating the predictive distribution of the beta distribution with the local variational method. In *2011 IEEE International Workshop on Machine Learning for Signal Processing*, pages 1–6. IEEE, 2011.

[68] Filippo Palumbo, Paolo Barsocchi, Claudio Gallicchio, Stefano Chessa, and Alessio Micheli. Multisensor data fusion for activity recognition based on reservoir computing. In *International competition on evaluating AAL systems through competitive benchmarking*, pages 24–35. Springer, 2013.

[69] George G. Tiao and Irwin Cuttman. The inverted dirichlet distribution with applications. *Journal of the American Statistical Association*, 60(311):793–805, 1965.

[70] Taoufik Bdiri and Nizar Bouguila. Bayesian learning of inverted dirichlet mixtures for SVM kernels generation. *Neural Comput. Appl.*, 23(5):1443–1458, 2013.

[71] Parisa Tirdad, Nizar Bouguila, and Djemel Ziou. Variational learning of finite inverted dirichlet mixture models and applications. In Yacine Laalaoui and Nizar Bouguila, editors, *Artificial Intelligence Applications in Information and Communication Technologies*, volume 607 of *Studies in Computational Intelligence*, pages 119–145. Springer, 2015.

[72] Paula Branco, Luís Torgo, and Rita P. Ribeiro. A survey of predictive modeling on imbalanced domains. *ACM Comput. Surv.*, 49(2), August 2016.

[73] B. Zadrozny, J. Langford, and N. Abe. Cost-sensitive learning by cost-proportionate example weighting. In *Third IEEE International Conference on Data Mining*, pages 435–442, 2003.

[74] Nizar Bouguila and Djemel Ziou. A hybrid sem algorithm for high-dimensional unsupervised learning using a finite generalized dirichlet mixture. *IEEE Transactions on Image Processing*, 15(9):2657–2668, 2006.

[75] L. Li, X. Guo, and N. Ansari. Smartloc: Smart wireless indoor localization empowered by machine learning. *IEEE Transactions on Industrial Electronics*, 67(8):6883–6893, 2020.

[76] Wentao Fan and Nizar Bouguila. Variational learning of a dirichlet process of generalized dirichlet distributions for simultaneous clustering and feature selection. *Pattern Recognition*, 46(10):2754–2769, 2013.

[77] Manar Amayri, Stephane Ploix, Nizar Bouguila, and Frederic Wurtz. Database quality assessment for interactive learning: application to occupancy estimation. *Energy and Buildings*, 209:109578, 2020.

[78] HM Sajjad Hossain, Md Abdullah Al Hafiz Khan, and Nirmalya Roy. Active learning enabled activity recognition. *Pervasive and Mobile Computing*, 38:312–330, 2017.

[79] Y. Li, G. Liu, G. Lu, L. Jiao, N. Marturi, and R. Shang. Hyper-parameter optimization using mars surrogate for machine-learning algorithms. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 4(3):287–297, 2020.