December 15, 2016

RT0976
Computer Science; Mathematics 6 pages

Research Report

Bayesian Regression Selecting Valuable Subset from Mixed Bag Training Data

Takayuki Katsuki, Masato Inoue

IBM Research - Tokyo
IBM Japan, Ltd.
19-21, Nihonbashi Hakozaki-cho
Chuo-ku, Tokyo 103-8510 Japan

Bayesian Regression Selecting Valuable Subset from Mixed Bag Training Data

Takayuki Katsuki
IBM Research – Tokyo
Tokyo, Japan 103–8510
Email: kats@jp.ibm.com

Masato Inoue
School of Advanced Science and Engineering
Waseda University
Tokyo, Japan 169–0072
Email: masato.inoue@eb.waseda.ac.jp

***Abstract*—This paper addresses a problem in which we learn a regression model from sets of training data. Each of the sets has an only single label, and only one of the training data in the set reflects the label. This is particularly the case when the label is attached to a group of data, such as time-series data. The label is not attached to the point of the sequence but rather attached to particular time window of the sequence. As such, a small part of the time window likely reflects the label, whereas the other larger part of the time window likely does not reflect it. We design an algorithm for estimating which of the training data in each of the sets corresponds to the label, as well as for training the regression model on the basis of Bayesian modeling and posterior inference with variational Bayes. Our experimental results show that our approach perform better than baseline methods on an artificial dataset and on a real-world dataset.**

I. Introduction

Supervised learning is a fundamental task in pattern recognition and machine learning [1], [2]. A focus of such a task is in learning a model representing the relationship between data and a corresponding label, wherein the learned model can be used for assigning labels to new unlabeled data. The task is performed by using labeled training data that consists of the pairs of data and labels.

The quantity and quality of labeled training data has a huge influence on the quality of the learned model. Recently, the cost of preparing a huge amount of labeled training data has come down thanks to growth in crowdsourcing services, social networking services, and sensor networks [3]. We can learn a model by using the huge amount of the data labeled in these ways. However, the training data acquired by these means may often contain wrong labels and be likely a mixed bag. When the quality of the training data is expected to be low, we would traditionally use a robust method such as one based on a heavy-tailed distribution [4]–[7]. Also, when we use the crowdsourcing for performing the labeling, we should use one of many approaches which can handle the label quality in such situation [8]–[10]. Most of the existing methods comprising these approaches explicitly or implicitly rely on the assumption that the proportion of correct labels in the training data is higher than that of wrong labels. Using weighting techniques for assessing the noise strength, ability of crowd workers, and instance difficulty, they learn a model by majority rule of the labels.

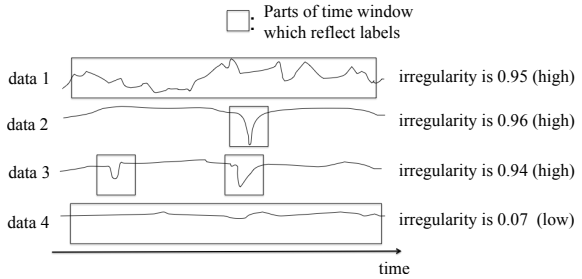

Fig. 1. Data 1, 2, and 3 are quite different time sequences but they are attached similar high irregularity label because of the existence of parts having high irregularity. On the other hand, data 4 is similar to data 2 and 3 in most part of the window, but is attached low irregularity label. Note that the irregularity is defined in the interval $[0, 1]$.

In particular, in the case of supervised learning on time-sequential data, it is possible that the proportion of the correct labels is *lower than* that of the wrong labels. Here, a label is not attached to a point of the sequence but rather attached to a time window of the sequence. In this case, only a small part of the data in this window likely reflects the label, and the remaining part does not reflect it, as shown in Figure 1. In this case, since the proportion of correct labels may be lower than that of the wrong ones, we can not use the majority rule in the robust methods. Also, since the feature vector from the whole window may not reflect the label, other existing methods cannot use such data to learn the model. They require the part of the sequence reflecting the label to be selected from the sequence.

In this paper, we focus on a regression problem using such mixed bag data. We formulate a problem in which we learn the regression model from sets of training data. Each of the sets has an only single label and only one of the training data in the set reflects the label. We propose a model to select valuable data from each of the sets for learning the desired regression model. Our model has hidden variables representing which of the training data in the set corresponds to the label. Based on fully Bayesian framework, we can simultaneously estimate the hidden variables and parameters of the regression model.

This paper is organized as follows. We formulate the task in Section II and propose the model in Section III. In Section IV, we describe a variational Bayes approach for learning the model. In Section V, we evaluate our method using artificial

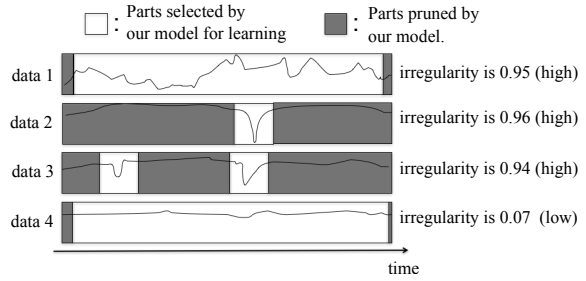

Fig. 2. Our model selects useful parts for learning the model.

and real-world datasets. In Section VI, we discuss the work related to this paper. We conclude in Section VII.

II. Regression Problem from Sets of Training Data

Suppose we are given N sets of training samples, $\{\boldsymbol{X}^{(n)}\}_{n=1}^N$, and the n-th set $\boldsymbol{X}^{(n)}$ has K training samples as $\boldsymbol{X}^{(n)} \equiv \{\boldsymbol{x}_1^{(n)}, \boldsymbol{x}_2^{(n)}, \ldots, \boldsymbol{x}_K^{(n)}\}$, where $\boldsymbol{x}_k^{(n)} \in \mathbb{R}^D$ is a D-dimensional feature vector for the k-th sample in the n-th set. A single label $y^{(n)} \in \mathbb{R}$ is attached to the n-th set. Then the N sets of the labels can be represented as $\boldsymbol{y} \equiv \{y^{(1)}, y^{(2)}, \ldots, y^{(N)}\}$. One of the K training samples in the n-th set corresponds to the label $y^{(n)}$, but we do not know which of them it is.

Our goal is to learn the relationship between the feature vector $\boldsymbol{x}$ and the label y by using the given data $\{\boldsymbol{X}^{(n)}\}_{n=1}^N$ and $\boldsymbol{y}$, and we use the relationship for making the prediction.

III. Bayesian Regression Model for Selecting a Valuable Subset

We design a regression model by introducing hidden variables $\boldsymbol{h}^{(n)} \in \{0,1\}^K$, $\sum_{k=1}^K h_k^{(n)} = 1$ that represent which of the K training samples in the n-th set corresponds to the n-th label $y^{(n)}$ in the 1-of-K notation, as shown in Figure 2. For example, if $\boldsymbol{h}^{(n)} = [1,0,0,0,...]$, the 1-st training sample $\boldsymbol{x}_1^{(n)}$ in the n-th set corresponds to the n-th label $y^{(n)}$. If $\boldsymbol{h}^{(n+1)} = [0,0,1,0,...]$, the 3-rd training sample $\boldsymbol{x}_3^{(n+1)}$ in the $n+1$-th set corresponds to the $n+1$-th label $y^{(n+1)}$. The N set of hidden variables is represented as $\boldsymbol{H} \equiv \{\boldsymbol{h}^{(n)}\}_{n=1}^N$ Although we use the 1-of-K notation for $\boldsymbol{h}^{(n)}$, our learning procedure estimates $\boldsymbol{h}^{(n)}$ probabilistically. Thus, we can represent a situation in which multiple samples in the n-th set correspond to the n-th label $y^{(n)}$ with specific weights, such as $[0.3, 0.1, 0.6, 0, ...]$.

Next, we define the regression model for $\boldsymbol{X}$ and $\boldsymbol{y}$ when $h_k = 1$ as

$$p(y|\boldsymbol{X}, h_k = 1, \boldsymbol{w}, \beta) \equiv \mathcal{N}(y|\boldsymbol{w}^\top \boldsymbol{x}_k, \beta^{-1}), \tag{1}$$

where $\mathcal{N}(x|\bullet, \beta^{-1})$ denotes a Gaussian distribution with mean $\bullet$ and precision β (see the Appendix-B for the explicit definition). The parameters $\boldsymbol{w} \equiv [w_1, w_2, \ldots, w_D] \in \mathbb{R}^D$ and $\beta > 0$ are model parameters to be learned. In particular, $\boldsymbol{w}$ represents the regression coefficients, and the d-th element in $\boldsymbol{w}$ is that for the d-th feature in $\boldsymbol{x}$.

Since we do not know which of the K training samples in $\boldsymbol{X}$ corresponds to the label $\boldsymbol{y}$, an arbitrary element in $\boldsymbol{h}$ can become one. Thus, our model has K mixture components such that

$$\begin{aligned} p(y|\boldsymbol{X}, \boldsymbol{h}, \boldsymbol{w}, \beta) &\equiv \prod_{k=1}^K \mathcal{N}\left(y|\boldsymbol{w}^\top \boldsymbol{x}_k, \beta^{-1}\right)^{h_k} \\ &= \frac{\exp\left(-\frac{\beta}{2}\sum_{k=1}^K h_k (y - \boldsymbol{w}^\top \boldsymbol{x}_k)^2\right)}{(2\pi\beta^{-1})^{\frac{1}{2}}}. \end{aligned} \tag{2}$$

Through the estimation of $\boldsymbol{h}$ in this model for the n-th set, we can select valuable training samples from the n-th set for learning the regression model.

Note that we can straightforwardly extend the model in Eq. (2) so that it can handle a non-linear relationship between $\boldsymbol{x}$ and y by using the basis function or kernel function, $\boldsymbol{\phi}(\boldsymbol{x}_k)$, as follows:

$$p(y|\boldsymbol{X}, \boldsymbol{h}, \boldsymbol{w}, \beta) \equiv \prod_{k=1}^K \mathcal{N}\left(y|\boldsymbol{w}^\top \boldsymbol{\phi}(\boldsymbol{x}_k), \beta^{-1}\right)^{h_k}. \tag{3}$$

For simplicity, we will continue to use the expression in Eq. (2) hereafter.

We can also extend our model so that it can be applied to classification tasks with a specific link function and distribution. Similarly, we may use other noise models, such as the t-distribution. Such investigations will be our future work.

IV. Learning Algorithm

Here, we design a learning algorithm for simultaneously estimating the hidden variables $\boldsymbol{H}$ for the N sets and the parameters $\boldsymbol{w}$ and β of the proposed model from the training data, $\{\boldsymbol{X}^{(n)}\}_{n=1}^N$ and $\boldsymbol{y}$. In the probabilistic formulation, the goal is to find the posterior distributions $p(\boldsymbol{H}|\{\boldsymbol{X}^{(n)}\}_{n=1}^N, \boldsymbol{y})$, $p(\boldsymbol{w}|\{\boldsymbol{X}^{(n)}\}_{n=1}^N, \boldsymbol{y})$, and $p(\beta|\{\boldsymbol{X}^{(n)}\}_{n=1}^N, \boldsymbol{y})$, which represent the probability distributions for $\boldsymbol{H}$, $\boldsymbol{w}$, and β given the training data.

A. Joint Distribution

For deriving the posterior distributions for $\boldsymbol{H}$, $\boldsymbol{w}$, and β, we first write down the joint distribution or the complete likelihood with our model in Eq.(2) and all of the random variables as

$$\begin{aligned} &p(\boldsymbol{y}, \boldsymbol{H}, \boldsymbol{w}, \beta|\{\boldsymbol{X}^{(n)}\}_{n=1}^N) \\ &\quad\equiv \prod_{n=1}^N p(y^{(n)}|\boldsymbol{X}^{(n)}, \boldsymbol{h}^{(n)}, \boldsymbol{w}, \beta) p(\boldsymbol{h}^{(n)}) p(\boldsymbol{w}) p(\beta), \end{aligned} \tag{4}$$

where, since we have no prior knowledge on the model parameters, we just introduce conjugate priors which are chosen based on the forms of our model: $p(\boldsymbol{h})$ is a categorical distribution, $p(\boldsymbol{w})$ is a Gaussian distribution, and $p(\beta)$ is a gamma distribution; these priors are set to be as non-informative as possible and to have a quite flat distribution. The explicit definitions are given in the Appendix-A.

All marginal and conditional distributions including the posteriors $p(\boldsymbol{H}|\{\boldsymbol{X}^{(n)}\}_{n=1}^N, \boldsymbol{y})$, $p(\boldsymbol{w}|\{\boldsymbol{X}^{(n)}\}_{n=1}^N, \boldsymbol{y})$ and

TABLE I
ESTIMATED PARAMETERS FOR EACH CASE OF $K = \{2, 5, 10\}$ WITH NORMAL NOISE (NOISE PRECISION $\beta = 10$).

	value of each element in $\boldsymbol{w}$						β
true	1.5	-2	0.5	0	0	0	10
estimated ($K = 2$)	1.5	-2.0	0.50	0.0	0.0	0.0	10
estimated ($K = 5$)	1.5	-2.0	0.50	0.0	0.0	0.0	10
estimated ($K = 10$)	1.5	-2.0	0.51	0.0	0.0	0.0	10

TABLE II
ESTIMATED PARAMETERS FOR EACH CASE OF $K = \{2, 5, 10\}$ WITH HIGH NOISE (NOISE PRECISION $\beta = 1$).

	value of each element in $\boldsymbol{w}$						β
true	1.5	-2	0.5	0	0	0	1
estimated ($K = 2$)	1.6	-2.1	0.53	0.0	0.0	0.0	1.0
estimated ($K = 5$)	1.7	-2.3	0.60	0.0	0.0	0.0	0.94
estimated ($K = 10$)	2.7	-3.6	1.2	0.0	0.0	0.0	0.33

$p(\beta|\{\boldsymbol{X}^{(n)}\}_{n=1}^N, \boldsymbol{y})$ can be derived in terms of this joint distribution.

B. Variational Bayes Algorithm

Here, it is not possible to obtain an exact analytical solution for the posteriors. Instead, we will derive an approximate solution by using the variational Bayes (VB) method [11]. The VB approach approximately finds the posterior distribution over the set of unobserved variables, $p(\boldsymbol{H}, \boldsymbol{w}, \beta|\{\boldsymbol{X}^{(n)}\}_{n=1}^N, \boldsymbol{y})$, in a factorized form:

$$p(\boldsymbol{H}, \boldsymbol{w}, \beta|\{\boldsymbol{X}^{(n)}\}_{n=1}^N, \boldsymbol{y}) \simeq q(\boldsymbol{H}, \boldsymbol{w}, \beta) \equiv q(\boldsymbol{H})q(\boldsymbol{w})q(\beta). \tag{5}$$

We identify the optimal approximate distribution that minimizes the Kullback-Leibler (KL) divergence [12] between the approximate distribution $q(\boldsymbol{H}, \boldsymbol{w}, \beta)$ and the true posterior distribution $p(\boldsymbol{H}, \boldsymbol{w}, \beta|\{\boldsymbol{X}^{(n)}\}_{n=1}^N, \boldsymbol{y})$ as the best approximation of the true distribution. In a popular VB approach [13], we solve the following iterative updating equations:

$$q(\boldsymbol{H}) \propto \exp\left[\left\langle \ln p(\boldsymbol{y}, \boldsymbol{H}, \boldsymbol{w}, \beta|\{\boldsymbol{X}^{(n)}\}_{n=1}^N)\right\rangle_{\boldsymbol{w},\beta}\right], \tag{6}$$

$$q(\boldsymbol{w}) \propto \exp\left[\left\langle \ln p(\boldsymbol{y}, \boldsymbol{H}, \boldsymbol{w}, \beta|\{\boldsymbol{X}^{(n)}\}_{n=1}^N)\right\rangle_{\boldsymbol{H},\beta}\right], \tag{7}$$

and (8)

$$q(\beta) \propto \exp\left[\left\langle \ln p(\boldsymbol{y}, \boldsymbol{H}, \boldsymbol{w}, \beta|\{\boldsymbol{X}^{(n)}\}_{n=1}^N)\right\rangle_{\boldsymbol{H},\boldsymbol{w}}\right],$$

where $\langle\cdot\rangle_*$ represents the expectation with regard to the distribution $q(*)$ of the random variables $*$. Thanks to the use of the conjugate prior distributions, we can compute the above expectations analytically as

$$q(\boldsymbol{H}) = \prod_{n=1}^N \mathrm{Categorical}\left(\boldsymbol{h}^{(n)}\middle|\boldsymbol{\xi}_{\boldsymbol{h}^{(n)}}\right), \tag{9}$$

$$q(\boldsymbol{w}) = \mathcal{N}\left(\boldsymbol{w}\middle|\boldsymbol{\mu}_{\boldsymbol{w}}, \boldsymbol{\Sigma}_{\boldsymbol{w}}\right), \quad \text{and} \tag{10}$$

$$q(\beta) = \mathrm{Gamma}\left(\beta\middle|a_\beta, b_\beta\right), \tag{11}$$

where Categorical is the categorical distribution and Gamma is the gamma distribution (see the Appendix-B for their definitions). The specific equations of the parameters $\boldsymbol{\xi}_{\boldsymbol{h}^{(n)}}$, $\boldsymbol{\mu}_{\boldsymbol{w}}$, $\boldsymbol{\Sigma}_{\boldsymbol{w}}$, a_β, and b_β are omitted here due to space limitations.

We can iteratively update q by simply computing only the parameters of these distributions in Eqs. (9) to (11). For the initial values of the parameters, we can use the same values as those of the corresponding priors. In practice, we stop the VB iterations when the relative differences between the current values of the variables, $\boldsymbol{z}_c$, and the previous values of the variables, $\boldsymbol{z}_p$, are sufficiently low:

$$\frac{\|\boldsymbol{z}_c - \boldsymbol{z}_p\|_2^2}{\|\boldsymbol{z}_p\|_2^2} < 10^{-5}. \tag{12}$$

After the above stopping condition is satisfied, we obtain the final outcome $q(\boldsymbol{H})$, $q(\boldsymbol{w})$ and $q(\beta)$ directly, which corresponds to an approximation of the learned posteriors, $p(\boldsymbol{H}|\{\boldsymbol{X}^{(n)}\}_{n=1}^N, \boldsymbol{y})$, $p(\boldsymbol{w}|\{\boldsymbol{X}^{(n)}\}_{n=1}^N, \boldsymbol{y})$ and $p(\beta|\{\boldsymbol{X}^{(n)}\}_{n=1}^N, \boldsymbol{y})$, since the distribution of $\boldsymbol{H}$, $\boldsymbol{w}$, and β has been factorized as shown in Eq. (5).

Using the learned $\boldsymbol{\mu}_{\boldsymbol{w}}$, we can predict the label for the new data as follows:

$$y \equiv \boldsymbol{\mu}_{\boldsymbol{w}}^\top \boldsymbol{x}. \tag{13}$$

Note that we can directly estimate the predictive posterior distribution $p(y|\boldsymbol{X}, \{\boldsymbol{X}^{(n)}\}_{n=1}^N, \boldsymbol{y})$ by using the VB method. However, since the predictive posterior distribution requires VB iterations for each new data, it is quite costly to compute. Instead, we use Eq. (13) as an approximation of the predictive posterior mean of y that can be computed with a much lower computational cost. This approximation corresponds to the predictive posterior mean of y when we assume that $q(\boldsymbol{w})$ is the true posterior of $\boldsymbol{w}$ and β is fixed by its mean value over $q(\beta)$, $\frac{a_\beta}{b_\beta}$.

C. Extension for Sparse Solution of Coefficients

For pruning irrelevant features in the feature vector $\boldsymbol{x}$, we can use the automatic relevance determination (ARD) prior [14]–[16] as the prior of the coefficients $\boldsymbol{w}$:

$$p(\boldsymbol{w}|\boldsymbol{\alpha}) \equiv \prod_{d=1}^D \mathcal{N}(w_d|0, \alpha_d). \tag{14}$$

Similarly to [17], since this is also the conjugate prior distribution for $\boldsymbol{w}$, we can straightforwardly apply the above result for VB in Eq. (10) to the model in which we use the ARD prior for $\boldsymbol{w}$. The parameter $\boldsymbol{\alpha}$ can be also estimated in the VB framework. Using the ARD prior, we can get a sparse solution for $\boldsymbol{w}$: many of their estimated coefficients are zero. In the experiments described in the next section, we used an implementation with the ARD prior.

V. EXPERIMENTAL RESULTS

We assessed the effectiveness of our approach in numerical experiments. First, we artificially generated datasets to study

TABLE III
ESTIMATION ACCURACY OF HIDDEN VARIABLES IN EACH CASE OF $K = \{2, 5, 10\}$ AND NOISE PRECISION $\beta = \{10, 1\}$. CHANCE LEVEL IS THE ACCURACY THAT WOULD BE EXPECTED BY RANDOM CHOICES.

	$K = 2$	$K = 5$	$K = 10$
chance level	0.5	0.2	0.1
proposed (noise precision $\beta = 10$)	0.95	0.81	0.65
peoposed (noise precision $\beta = 1$)	0.84	0.55	0.22

the performance of our algorithm (Section V-A). We then applied it to real-world time-sequential data from the UCI machine learning repository [18] (Section V-B).

A. Experiment on Artificial Dataset

We studied the validity of our algorithm by simultaneously estimating $\boldsymbol{H}$, $\boldsymbol{w}$, and β from the artificial validation dataset. In preparing the artificial validation dataset, we randomly generated $N \times K$ training samples, $\{\boldsymbol{X}^{(n)}\}_{n=1}^{N}$, from the standard Gaussian distribution $\mathcal{N}(\boldsymbol{x}|0, \boldsymbol{I})$, where the number of dimensions of $\boldsymbol{x}$ was 6 and $\boldsymbol{I}$ is the identity matrix. Then, using $\{\boldsymbol{X}^{(n)}\}_{n=1}^{N}$, we generated the corresponding N sets of labels $\boldsymbol{y}$ from the distribution in Eq. (1), where we randomly selected one of the K training samples in each n-th set from a uniform distribution, a limited number of the coefficients, $\boldsymbol{w}$, had non-zero values, *i.e.*, $\boldsymbol{w} = \{1.5, -2.0, 0.5, 0, 0, 0\}$. We repeatedly evaluated the proposed method for each of the following settings: the noise precision $\beta = \{10, 1\}$, which correspond normal and high noise settings, and number of training samples in each training set $K = \{2, 5, 10\}$. In the case of $K = 10$, only 10 percent of the data correctly corresponds to labels. In general, it is quite hard to learn regression models using such data. In this experiment, the number of training sets was $N = 10000$.

Tables I and II compares the estimated $\boldsymbol{w}$ and β with the true ones. Also, Table III shows the estimation accuracy of $\boldsymbol{h}$, which is defined as the proportion of indexes in which the maximum value in the estimated $\boldsymbol{h}$ is exactly the same as the true one selected in generating the data, where 1 is the best and 0 is the worst. The result confirms that our method can simultaneously estimate all of the parameters and hidden variables well except for the most difficult setting in which $K = 10$ and $\beta = 1$. Note that we can get a sparse solution for the coefficient thanks to the ARD prior described in Section IV-C.

Finally, Figure 3 compares our approach with common regression methods, which are t-regression [5], [19]–[22] with L1-regularization [23], relevance vector machine (RVM) with a linear kernel [16], [17], and random forest [24]. Since these baseline methods are not able to select valuable samples, in the training of these models, they select one of the K training samples in the n-th set from the same uniform distribution used to generate the data. We evaluated the results with regard to the mean absolute error (MAE) over M test samples, which were generated from the same distribution as the training

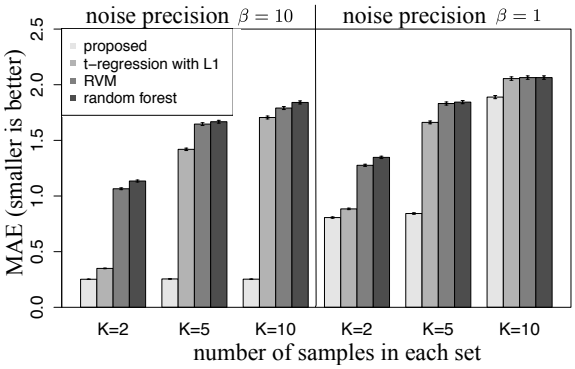

Fig. 3. Comparison of the proposed method and several regression methods in terms of MAE (smaller is better) on an artificial dataset. Error bars represent the standard error.

samples, and the number of test samples M was $M = 10000$. MAE is defined as

$$\mathrm{MAE} \equiv \frac{1}{M} \sum_{m=1}^{M} \left| y_{\mathrm{true}}^{(m)} - y_{\mathrm{estimate}}^{(m)} \right|, \tag{15}$$

where $y_{\mathrm{true}}^{(m)}$ is the true y in the m-th test sample, and $y_{\mathrm{estimate}}^{(m)}$ is the estimated y for the m-th test sample. We computed the standard error of the absolute error (the error bars in Fig. 3). From Fig. 3, we can see that the overall performance of our method is significantly better than those of the alternatives. The t-regression with L1-regularization, which is the well-known robust regression method, achieved a good result in the case of $K = 2$, but it did not work in the case of $K = 5$ or $K = 10$. Our method can select the valuable samples from each n-th set in $\{\boldsymbol{X}^{(n)}\}_{n=1}^{N}$ and achieved the best performance in all of the settings.

B. Experiment using UCI Dataset

We evaluated the proposed method in the prediction task for indoor temperatures from temporal sequences of sensor outputs in a house [25]. The dataset for this task was a real-world dataset collected from the publicly available UCI machine learning repository [18].

The dataset consisted of 4137 samples and each sample had 24 number of attributes. Regarding the feature vector, we used all the attributes except for non numeric attributes and attributes always taking 0. In this problem setting, y is the indoor temperature at a future timestamp, which is standardized by subtracting its mean and dividing by its standard deviation, and $\boldsymbol{X}$ is the set of $K = 4$ number of training samples $\boldsymbol{x}$ which are computed from four different time windows in an hour before the timestamp; in particular, we use the features in the first fifteen minutes as $\boldsymbol{x}_1$, the features in the second fifteen minutes as $\boldsymbol{x}_2$, the features in the third fifteen minutes as $\boldsymbol{x}_3$, and the features in the last fifteen minutes as $\boldsymbol{x}_4$. Our model prunes ones corrupted by noise and outliers and selects valuable ones in the time windows for training the prediction model.

Table IV compares our approach with RVM [16], [17] with a linear kernel. Since the baseline method does not have the

TABLE IV
COMPARISON OF THE PROPOSED METHOD AND BASELINE METHOD IN TERMS OF MAE (SMALLER IS BETTER) ON THE UCI DATASET.

method	MAE (smaller is better)
RVM	0.36 ± 0.0058
peoposed	$\mathbf{0.32} \pm 0.0055$

TABLE V
EXAMPLES OF ESTIMATION RESULTS OF HIDDEN VARIABLES $\boldsymbol{h}$.

data index	$h_1^{(n)}$	$h_2^{(n)}$	$h_3^{(n)}$	$h_4^{(n)}$
$n = 118$	0.0025	0.12	0.51	0.37
$n = 119$	0.65	0.26	0.07	0.02
$n = 120$	0.25	0.28	0.23	0.24
$n = 121$	0.12	0.25	0.22	0.41

ability to select valuable samples, in the training of the model, it selects one of the K training samples in the n-th set from the uniform distribution. In prediction, we always use the features in the last fifteen minutes, $\boldsymbol{x}_4$, for both of our method and the baseline method. We evaluated the results with regard to the mean absolute error (MAE) in 5-fold cross validation using the dataset and also computed the standard error of the absolute error. From Table IV, we can see that the MAE of our method is 10% better than that of the baseline method.

Finally, Table V shows a typical examples of the estimation results of $\boldsymbol{h}$ that represent which of the K training samples in the n-th set corresponds to the n-th label. We can see that the estimation results of $\boldsymbol{h}$ are significantly different from each other. It suggests that the ability to select valuable training samples in each n-th set is important for the prediction accuracy even in real-world case.

VI. RELATED WORK

There have been prior studies on handling the uncertainty of labels. The majority of the prior work has been on robust estimations, such as an estimation based on heavy-tailed distributions [4]–[7]. The t-regression, which is based on the student's t-distribution, is one of the most common robust regression methods [5], [19]–[22]. The L1-based estimator, which is related to median-based methods, is also commonly used [6], [26]. Most of these methods weight each of the training data based on its noise level and prune the data to which a lot of noise was added during the learning of the regression model. The literature on crowdsourcing has studies on explicitly handling the uncertainty of manual labeling [8]–[10]. These approaches learn the regression and classification models robustly by improving the quality of labels, where they obtain multiple labels for each training data from multiple labelers and weight them based on the ability of labelers and difficulties of labeling examples. It would be interesting to apply our fully Bayesian approach to a Multiple Instance Learning (MIL) problem [27]–[30] in future work. Since our problem setting is different from the problem setting of the MIL which requires to handle the mixed bag or the multiple instance even in prediction, it will be required to modify Eq. (13) for the MIL problem setting.

In Bayesian inference, we can evaluate the posterior of the estimation result [2], [31], [32]. This property is quite useful in the case of that the confidence of the estimation result, which is computed using the posterior, is important, such as in Bayesian optimization [33], [34], Bayesian active learning [35], [36], Bayesian reinforcement learning [37]–[40]. We use it in the VB updates. By using the confidence of the estimation result of each of the variables at each step of VB, we can tune the update width properly on the estimations of the variables at each step and can get a stable final estimation result in a situation in which there are many variables to be learned.

VII. CONCLUSION

We formulated a regression problem selecting a valuable subset from each set of the mixed bag training data using Bayesian modeling with hidden variables. For the proposed model, we designed an efficient learning algorithm by using VB. Our method does not have any parameters that require careful tuning, thanks to its fully Bayesian modeling. Experimental results show that our approach performed better than baseline methods on an artificial dataset and on a real-world dataset. Our method can achieve robust regression result even in the case of that only 10 percent of the data correctly corresponds to labels.

In the future, we plan to apply our approach to other learning tasks, such as classification problems.

ACKNOWLEDGMENTS

This research was supported by CREST, JST. The authors thank Y. Shinohara, S. Hara, T. Morimura, and T. Yoshizumi for their helpful discussions.

APPENDIX

A. Conjugate Priors for Model Parameters

For the prior distributions of $\boldsymbol{H}$, $\boldsymbol{w}$ and β, we simply use the conjugate priors:

$$p(\boldsymbol{H}) \equiv \prod_{n=1}^{N} \mathrm{Categorical}\left(\boldsymbol{h}^{(n)} \middle| \boldsymbol{\xi}_{\boldsymbol{h}}^{(0)}\right), \tag{16}$$

$$p(\boldsymbol{w}) \equiv \mathcal{N}\left(\boldsymbol{w} \middle| 0, \Sigma_{\boldsymbol{w}}^{(0)}\right), \quad \text{and} \tag{17}$$

$$p(\beta) \equiv \mathrm{Gamma}\left(\beta \middle| a_\beta^{(0)}, b_\beta^{(0)}\right), \tag{18}$$

where the parameters $\boldsymbol{\xi}_{\boldsymbol{h}}^{(0)}$, $\boldsymbol{\Sigma}_{\boldsymbol{w}}^{(0)}$, $a_\beta^{(0)}$, and $b_\beta^{(0)}$ are treated as input parameters given as part of the model. We chose the hyperparameter values in Eqs. (16) to (18) to be as non-informative as possible and to have a quite flat distribution: $\boldsymbol{\xi}_{\boldsymbol{h}}^{(0)} = 10^{-10} \times \boldsymbol{i}$, $a_\beta^{(0)}/N = b_\beta^{(0)}/N = 10^{-10}$ and $\boldsymbol{\Sigma}_{\boldsymbol{w}}^{(0)} = 10^{10} \times \boldsymbol{I}$, where $\boldsymbol{i}$ represents a vector of all ones.

In addition, for the model in Eq. (14), we define hyperprior distributions for $\boldsymbol{\alpha}$ using the conjugate priors:

$$p(\boldsymbol{\alpha}) \equiv \prod_{d=1}^{D} \mathrm{Gamma}\left(\alpha_d \middle| a_\alpha^{(0)}, b_\alpha^{(0)}\right), \tag{19}$$

where the hyperparameter values in Eq. (19) are also non-informative: $a_\alpha^{(0)} = b_\alpha^{(0)} = 10^{-10}$.

B. Probability Distributions

Here, we give the definitions of the gamma, Gaussian, and categorical distributions:

$$\mathrm{Gamma}(x|a,b) \equiv \frac{b^a}{\Gamma(a)} x^{a-1}\mathrm{e}^{-bx} \qquad (x>0),$$

$$\mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu},\boldsymbol{\Sigma}) \equiv |2\pi\boldsymbol{\Sigma}|^{-\frac{1}{2}}\mathrm{e}^{-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^\top\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})} \quad (\boldsymbol{x}\in\mathbb{R}^N), \quad \text{and}$$

$$\mathrm{Categorical}(\boldsymbol{x}|\boldsymbol{\xi}) \equiv \prod_{k=1}^{K}\xi_k^{x_k} \qquad (x_k\in\{0,1\},\ \sum_{k=1}^{K}x_k=1),$$

where Γ denotes the gamma function, $|\bullet|$ denotes the determinant of the given matrix $\bullet$, and the parameters are such that $a>0$, $b>0$, $\boldsymbol{\mu}\in\mathbb{R}^N$, $\boldsymbol{\Sigma}\in\mathbb{R}^{N\times N}$, $0\leq\xi_k\leq 1$, and $\sum_{k=1}^{K}\xi_k=1$. The variables in these definitions are not related to the variables that appear in the main text.

References

[1] A. K. Jain, R. P. Duin, and J. Mao, "Statistical pattern recognition: A review," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 22, no. 1, pp. 4–37, 2000.

[2] C. M. Bishop *et al.*, *Pattern recognition and machine learning*. springer New York, 2006, vol. 1.

[3] H. Jagadish, J. Gehrke, A. Labrinidis, Y. Papakonstantinou, J. M. Patel, R. Ramakrishnan, and C. Shahabi, "Big data and its technical challenges," *Communications of the ACM*, vol. 57, no. 7, pp. 86–94, 2014.

[4] A. O'Hagan, "On outlier rejection phenomena in bayes inference," *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 358–367, 1979.

[5] M. West, "Outlier models and prior distributions in bayesian linear regression," *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 431–439, 1984.

[6] R. R. Wilcox, *Introduction to robust estimation and hypothesis testing*. Academic Press, 2012.

[7] M. Feng, L. Y. Loy, K. Sim, C. Phua, F. Zhang, and C. Guan, "Artifact correction with robust statistics for non-stationary intracranial pressure signal monitoring," in *Proceedings of the 21st International Conference on Pattern Recognition (ICPR)*. IEEE, 2012, pp. 557–560.

[8] V. S. Sheng, F. Provost, and P. G. Ipeirotis, "Get another label? improving data quality and data mining using multiple, noisy labelers," in *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2008, pp. 614–622.

[9] J. Whitehill, T.-f. Wu, J. Bergsma, J. R. Movellan, and P. L. Ruvolo, "Whose vote should count more: Optimal integration of labels from labelers of unknown expertise," in *Advances in neural information processing systems*, 2009, pp. 2035–2043.

[10] V. C. Raykar, S. Yu, L. H. Zhao, G. H. Valadez, C. Florin, L. Bogoni, and L. Moy, "Learning from crowds," *The Journal of Machine Learning Research*, vol. 11, pp. 1297–1322, 2010.

[11] H. Attias and L. W. Ar, "Inferring parameters and structure of latent variable models by variational Bayes," in *Proc. of the Fifteenth Conference on Uncertainty in Artificial Intelligence*. Morgan Kaufmann Publishers, 1999, pp. 21–30.

[12] S. Kullback and R. A. Leibler, "On information and sufficiency," *The annals of mathematical statistics*, vol. 22, no. 1, pp. 79–86, 1951.

[13] C. Bishop, D. Spiegelhalter, and J. Winn, "VIBES: A variational inference engine for Bayesian networks," in *Advances in Neural Information Processing Systems 15*. MIT Press, 2003, pp. 777–784.

[14] D. J. C. Mackay, "Bayesian non-linear modelling for the prediction competition," *ASHRAE Transactions*, vol. 100, pp. 1053–1062, 1994.

[15] R. M. Neal, "Bayesian learning for neural networks," 1996.

[16] M. E. Tipping, "Sparse bayesian learning and the relevance vector machine," *The journal of machine learning research*, vol. 1, pp. 211–244, 2001.

[17] C. M. Bishop and M. E. Tipping, "Variational relevance vector machines," in *Proceedings of the Sixteenth conference on Uncertainty in artificial intelligence*. Morgan Kaufmann Publishers Inc., 2000, pp. 46–53.

[18] M. Lichman, "UCI machine learning repository," 2013. [Online]. Available: http://archive.ics.uci.edu/ml

[19] K. L. Lange, R. J. Little, and J. M. Taylor, "Robust statistical modeling using the t distribution," *Journal of the American Statistical Association*, vol. 84, no. 408, pp. 881–896, 1989.

[20] J. Geweke, "Bayesian treatment of the independent student-t linear model," *Journal of Applied Econometrics*, vol. 8, no. S1, pp. S19–S40, 1993.

[21] C. Liu and D. B. Rubin, "Ml estimation of the t distribution using em and its extensions, ecm and ecme," *Statistica Sinica*, pp. 19–39, 1995.

[22] G. McLachlan and T. Krishnan, *The EM algorithm and extensions*. John Wiley & Sons, 2007, vol. 382.

[23] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 267–288, 1996.

[24] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.

[25] F. Zamora-Martínez, P. Romeu, P. Botella-Rocamora, and J. Pardo, "On-line learning of indoor temperature forecasting models towards energy efficiency," *Energy and Buildings*, vol. 83, pp. 162–172, 2014.

[26] S. C. Narula and J. F. Wellington, "The minimum sum of absolute errors regression: A state of the art survey," *International Statistical Review/Revue Internationale de Statistique*, pp. 317–326, 1982.

[27] T. G. Dietterich, R. H. Lathrop, and T. Lozano-Pérez, "Solving the multiple instance problem with axis-parallel rectangles," *Artificial intelligence*, vol. 89, no. 1, pp. 31–71, 1997.

[28] S. Ray and D. Page, "Multiple instance regression," in *Proceedings of the 18th International Conference on Machine Learning*. Morgan Kaufmann Publishers Inc., 2001, pp. 425–432.

[29] D. R. Dooly, Q. Zhang, S. A. Goldman, and R. A. Amar, "Multiple-instance learning of real-valued data," *Journal of Machine Learning Research*, vol. 3, pp. 651–678, 2002.

[30] Z. Wang, L. Lan, and S. Vucetic, "Mixture model for multiple instance regression and applications in remote sensing," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 50, no. 6, pp. 2226–2237, 2012.

[31] R. M. Neal, *Bayesian learning for neural networks*. Springer Science & Business Media, 2012, vol. 118.

[32] C. Williams and C. Rasmussen, "Gaussian processes for regression," in *Advances in Neural Information Processing Systems*. MIT Press, 1996.

[33] D. R. Jones, "A taxonomy of global optimization methods based on response surfaces," *Journal of global optimization*, vol. 21, no. 4, pp. 345–383, 2001.

[34] J. Snoek, H. Larochelle, and R. P. Adams, "Practical bayesian optimization of machine learning algorithms," in *Advances in neural information processing systems*, 2012, pp. 2951–2959.

[35] K. Chaloner and I. Verdinelli, "Bayesian experimental design: A review," *Statistical Science*, pp. 273–304, 1995.

[36] G. Kumar and V. Govindaraju, "Bayesian active learning for keyword spotting in handwritten documents," in *22nd International Conference on Pattern Recognition (ICPR)*. IEEE, 2014, pp. 2041–2046.

[37] W. R. Thompson, "On the likelihood that one unknown probability exceeds another in view of the evidence of two samples," *Biometrika*, vol. 25, no. 3/4, pp. 285–294, 1933.

[38] M. Strens, "A bayesian framework for reinforcement learning," in *ICML*, 2000, pp. 943–950.

[39] A. Wilson, A. Fern, S. Ray, and P. Tadepalli, "Multi-task reinforcement learning: a hierarchical bayesian approach," in *Proceedings of the 24th international conference on Machine learning*. ACM, 2007, pp. 1015–1022.

[40] J. Asmuth, L. Li, M. L. Littman, A. Nouri, and D. Wingate, "A bayesian sampling approach to exploration in reinforcement learning," in *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*. AUAI Press, 2009, pp. 19–26.