Available online at www.sciencedirect.com
SCIENCE DIRECT®
Pattern Recognition Letters 26 (2005) 399–410
Pattern Recognition Letters
www.elsevier.com/locate/patrec

A privacy-sensitive approach to distributed clustering

Srujana Merugu *, Joydeep Ghosh

Department of Electrical and Computer Engineering, University of Texas at Austin, Austin, TX, USA

Available online 1 October 2004

Abstract

While data mining algorithms are often designed to operate on centralized data, in practice data is often acquired and stored in a distributed manner. Centralization of such data before analysis may not be desirable, and often not possible due to a variety of real-life constraints such as security, privacy and communication costs. This paper presents a general framework for distributed clustering that takes into account privacy requirements. It is based on building probabilistic models of the data at each local site, whose parameters are then transmitted to a central location. We mathematically show that the best representative of all the local models is a certain "mean" model, and empirically show that this model can be approximated quite well by generating artificial samples from the local models using sampling techniques, and then fitting a global model of a chosen parametric form to these samples. We also propose a new measure that quantifies privacy based on information theoretic concepts, and show that decreasing privacy improves the quality of the global model and vice versa. Empirical results are provided on different kinds of data to highlight the generality of our framework. The results show that high quality global clusters can be achieved with little loss of privacy.

Keywords: Privacy; Distributed clustering; Generative models

1. Introduction

Extracting useful knowledge from large, distributed data repositories can be a very difficult task when such data cannot be directly centralized or unified as a single file or database due to a variety of constraints. As in much of parallel processing, early work on distributed data mining mostly focused on technical constraints such as limited communication bandwidth or central storage. More recently, there has been an emphasis on obtaining high quality information from distributed sources while simultaneously adhering to restrictions on the *nature* of the data to be shared, due to data ownership or privacy issues. Much of this work is appearing under the moniker of "privacy-preserving data mining". In the clustering context, a prototypical privacy-sensitive application scenario

* Corresponding author.
E-mail address: merugu@ece.utexas.edu (S. Merugu).

0167-8655/$ - see front matter © 2004 Elsevier B.V. All rights reserved.
doi:10.1016/j.patrec.2004.08.003

is one in which there are multiple parties with confidential databases and the goal is to cluster the entire distributed data, without actually first pooling this data. For example, the parties can be a group of banks, with their own sets of customers, who would like to have a better insight into the behavior of the entire customer population without compromising the privacy of their individual customers.

Data mining techniques that focus on privacy have largely taken one of three approaches: (i) query restriction to solve the inference problem in databases (Farkas and Jajodia, 2002), (ii) subjecting individual records or attributes to a "privacy preserving" randomization operation and subsequently recovering the original data (Agrawal and Aggarwal, 2001), (iii) using cryptographic techniques for two-party or multi-party communications (Pinkas, 2002). The first method is difficult and manually intensive, while the latter two approaches are largely restricted to vector data and involve high communication costs. Moreover, recent research (Kargupta et al., 2003) shows that randomization operations do not necessarily preserve privacy as the original data can be substantially recovered using spectral filtering techniques.

There has been some work on distributed clustering for *vertically partitioned data*, wherein different sites contain different attributes/features of a common set of records/objects (Johnson and Kargupta, 1999), and on parallelizing clustering algorithms for *horizontally partitioned data*, i.e, the objects are distributed among the sites, which record the same set of features for each object (Dhillon and Modha, 1999, Tasolis and Vrahatis, 2004). Of these, the distributed clustering techniques proposed earlier do not specifically address privacy issues whereas the recently proposed privacy-sensitive clustering techniques are based on privacy requirements with limited practical applicability. In particular, the privacy-preserving clustering technique proposed in (Vaidya and Clifton, 2003) is based on the secure multi-party computation notion of privacy and requires high communication costs besides being vulnerable to collusion. Similarly, in the cluster ensembles framework (Strehl and Ghosh, 2002), the fundamental privacy constraint is that the local sites participating in the distributed clustering can share only the local cluster labels and the identifiers of the individual objects, which is useful for concealing proprietary algorithms, but not necessarily the individual objects themselves. Another recently proposed technique based on sampling local density estimates (Klusch et al., 2003) focuses on a privacy requirement that involves minimizing the number of data samples shared by the local sites.

In this paper, we present a general framework for clustering horizontally distributed data under an information theoretic privacy constraint, where neither the cluster labels nor a subset of the individual records can be shared. The basic motivation is that there is an (unknown) underlying distribution that represents the commonalities among the different data sources and identifying this distribution can provide useful information that is validated by all the data sources. Note that this underlying distribution is not necessarily a good descriptor of a specific contributing source, since each data source may have a different bias. The fundamental idea proposed in this work is that it is possible to learn the global underlying distribution by combining high-level information from the different sources instead of sharing individual records. A global model built in this manner can then be transmitted to each of the local sites and used for partitioning the local data.

We make three main contributions. First, we introduce a privacy preserving framework for distributed clustering that is applicable to a wide variety of data types and learning algorithms, so long as they can provide a generative model (Ghosh, 2003). In this framework, the parties owning the individual data sources independently train generative models on the local data and send the model parameters to a central combiner that integrates the models. This limits the amount of interactions between the data sources and the combiner and enables us to formulate the distributed clustering problem in a general as well as tractable form. Second, we present the idea that it is possible to obtain efficient solutions to optimization problems based on generative models by formulating approximate versions of the problems using sampling techniques, which can in turn be solved using

existing learning algorithms. We apply this idea to the specific problem of distributed clustering to develop EM based algorithms that are guaranteed to asymptotically converge to a locally optimal global model. Finally, we propose a measure for quantifying privacy based on ideas from information theory, which allows us to formalize the problem of obtaining a local model given the privacy constraints.

A word about the notation: Sets such as $\{x_1, \ldots, x_n\}$ are enumerated as $\{x_i\}_{i=1}^n$. Probability density function of a parametric model λ is denoted by p_λ. Expectation of functions of a random variable x following a distribution p are denoted by $\boldsymbol{E}_{x\sim p}[\cdot]$ or $\boldsymbol{E}_p[\cdot]$.

2. Distributed model-based clustering

Consider a scenario where the over-riding privacy constraint is that information about individual objects or "records" such as the feature values or the cluster labels, cannot be shared with another site. It is, therefore, necessary to describe the data by modeling the feature distributions across multiple records in such a way that the specifics of a particular record are obscured. To make this problem tractable, we consider the case where the records have the same sets of features at each site. This suggests an approach of building models locally and then combining them at a central location to obtain a more accurate model (Chan et al., 1996, Yamanishi, 1998). The advantage of this approach is that it enables easy analysis of privacy and communication costs in terms of the local model that is shared with the central location. The key is to characterize the data at each site using a suitable probabilistic (generative) model, and transmit only the model parameters to a central site, where "virtual samples" can be now generated using Monte Carlo Markov Chain (MCMC) sampling techniques and used to form a combined model. Since generative models are available for a wide range of data types, from vectors to variable length sequences and graphs (Cadez et al., 2000, Zhong and Ghosh, 2003), this approach is quite general and applicable to complex data. This also distinguishes our work from techniques that are applicable only to vector data, for example, those that combine multiple k-means solutions (Fayyad et al., 1998, Fred and Jain, 2002).

Since an important goal of data mining is to obtain highly interpretable results, we restrict our search for the optimal global model to the set of all mixture models based on a given parametric family (e.g., mixture of Gaussians). We call the resulting search problem of finding the highest quality global model within this family of models the *Distributed Model-based Clustering* (DMC) problem (Ghosh and Merugu, 2003, Merugu and Ghosh, 2003) and state it more formally below.

Let $\{\mathscr{X}_i\}_{i=1}^n$ be n horizontally partitioned data sources generated by a common underlying model λ^0. Let $\{\lambda_i\}_{i=1}^n$ be the local models obtained by applying clustering algorithms to these data sources and $\{v_i\}_{i=1}^n$ be non-negative weights associated with the local models based on their importance or on the size of the corresponding data sources. The objective of the DMC problem is to obtain the optimal global model λ_c^* belonging to a given family of models $\mathscr{F}$, i.e., $\lambda_c^* = \arg\min_{\lambda_c \in \mathscr{F}} Q(\lambda_c)$, where $Q(\cdot)$ is the model quality cost defined in terms of the local models and their weights.

2.1. Model representation and quality

We represent the *clustering models*, i.e., generative models produced by the clustering algorithms in terms of their probability density functions, i.e., the model λ is specified by $p_\lambda(x) = \sum_{h=1}^k \pi_\lambda^h p_\lambda(x|h)$, where $p_\lambda(x)$ is the probability density function, $\{\pi_\lambda^h\}_{h=1}^k$ are the cluster priors, $\{p_\lambda(x|h)\}_{h=1}^k$ are the cluster densities and k is the number of *components* or clusters (which could vary for each clustering model). This leads to a systematic approach for combining the models that is independent of the local clustering algorithms.

A natural definition for the quality cost, $Q_I(\cdot)$, for a global model, is simply the "distance" from the underlying true model λ^0, i.e., $Q_I(\lambda_c) = D(\lambda^0, \lambda_c)$, where $D(\cdot,\cdot)$ is a suitable distance measure for models. Since λ^0 is not known, we instead consider the different local models $\{\lambda_i\}_{i=1}^n$ as estimators of λ^0 with weights $\{v_i\}_{i=1}^n$ and define the

quality cost function in terms of the average distance from the local models, i.e., $Q(\lambda_c) = \sum_{i=1}^{n} v_i D(\lambda_i, \lambda_c)$, where $\sum_{i=1}^{n} v_i = 1$.

Metrics based on the norms of density functions such as the L_1 distance and the squared L_2 distance and KL-divergence are commonly used for comparing a pair of generative models. In particular, the KL-divergence between models λ_1 and λ_2 is given by

$$\begin{aligned} D_{\mathrm{KL}}(\lambda_1, \lambda_2) &= \mathrm{KL}(p_{\lambda_1} \| p_{\lambda_2}) \\ &= E_{p_{\lambda_1}} \log \left(\frac{p_{\lambda_1}(x)}{p_{\lambda_2}(x)} \right) \\ &= H(\lambda_1) - E_{p_{\lambda_1}} \log p_{\lambda_2}(x) \end{aligned} \quad (1)$$

where $H(\lambda_1)$ is the entropy of the distribution p_{λ_1} and the second term corresponds to the average log-likelihood of data generated from the distribution p_{λ_1} with respect to p_{λ_2}. Due to this linear relationship with the average log-likelihood,KL-divergence can be considered as the most natural comparison measure for generative models. It is also a well-behaved, differentiable function of the model parameters and has better convergence properties compared to the other measures. Hence, we optimize the quality cost based on KL-divergence and use the other measures only for secondary evaluation of the experimental results.

2.2. DMC algorithm

We first pose the DMC problem as an optimization problem, present an approximation using sampling techniques and then, propose a practical algorithm to efficiently address this approximate problem. The objective of the DMC problem is to obtain a global model λ_c belonging to a particular parametric family $\mathscr{F}$ such that the quality cost function $Q(\cdot)$ based on KL-divergence is minimized, i.e.,

$$\begin{aligned} \lambda_c^* &= \arg\min_{\lambda_c \in \mathscr{F}} Q(\lambda_c) \\ &= \arg\min_{\lambda_c \in \mathscr{F}} \sum_{i=1}^{n} v_i D_{\mathrm{KL}}(\lambda_i, \lambda_c), \end{aligned} \quad (2)$$

Algorithm 1. DMC Algorithm

Input: Set of models $\{\lambda_i\}_{i=1}^n$ with weights $\{v_i\}_{i=1}^n$ summing to 1, Mixture model family $\mathscr{F}$

Output: $\lambda_c^a \simeq \arg\min_{\lambda_c \in \mathscr{F}} \sum_{i=1}^{n} v_i D_{\mathrm{KL}}(\lambda_i, \lambda_c)$

Method:

1. Obtain mean model $\bar{\lambda}$ such that
$$p_{\bar{\lambda}}(x) = \sum_{i=1}^{n} v_i p_{\lambda_i}(x).$$
2. Generate $\bar{\mathscr{X}} = \{x_j\}_{j=1}^m$ from mean model, $\bar{\lambda}$ using MCMC sampling.
3. Apply EM algorithm to obtain the optimal model, λ_c^a, such that
$$\begin{aligned} \lambda_c^a &= \arg\max_{\lambda_c \in \mathscr{F}} L(\bar{\mathscr{X}}, \lambda_c) \\ &= \arg\max_{\lambda_c \in \mathscr{F}} \frac{1}{m} \sum_{j=1}^{m} \log(p_{\lambda_c}(x_j)). \end{aligned}$$

where $\{\lambda_i\}_{i=1}^n$ are the local clustering models. This problem can be simplified using the following result.

Theorem 1. [1] *Given a set of models $\{\lambda_i\}_{i=1}^n$ with weights $\{v_i\}_{i=1}^n$ summing to* 1, *then for any model λ_c, whose support set includes the support sets of $\{\lambda_i\}_{i=1}^n$* [2]

$$\begin{aligned} \sum_{i=1}^{n} v_i \mathrm{KL}(p_{\lambda_i} \| p_{\lambda_c}) &= \sum_{i=1}^{n} v_i \mathrm{KL}(p_{\lambda_i} \| p_{\bar{\lambda}}) \\ &\quad + \mathrm{KL}(p_{\bar{\lambda}} \| p_{\lambda_c}), \end{aligned}$$

where $\bar{\lambda}$ is such that $p_{\bar{\lambda}}(x) = \sum_{i=1}^{n} v_i p_{\lambda_i}(x)$.

Applying the above theorem, we see that the cost function in (2) is equal to $\sum_{i=1}^{n} v_i D_{\mathrm{KL}}(\lambda_i, \bar{\lambda}) + D_{\mathrm{KL}}(\bar{\lambda}, \lambda_c)$. The first term is independent of λ_c and hence, optimizing the cost function in (2) is

[1] This result is true for a class of functions called Bregman divergences (Azoury and Warmuth, 2001; Banerjee et al., 2004) of which KL-divergence and squared L_2 distance are particular cases.

[2] This ensures that the KL-divergence measure is well defined.

equivalent to minimizing KL-divergence with respect to the mean model $\bar{\lambda}$. In the absence of constraints on λ_c, the optimal solution is just the mean model $\bar{\lambda}$, as KL-divergence is always non-negative and zero only when both the arguments are equal.

The mean model also has the following nice property, which follows from Jensen's inequality.

Theorem 2. *Given a set of models* $\{\lambda_i\}_{i=1}^n$ *with weights* $\{v_i\}_{i=1}^n$ *summing to* 1 *and the true* [3] *model* λ^0,

$$D(\lambda^0, \bar{\lambda}) \leqslant \sum_{i=1}^{n} v_i D(\lambda^0, \lambda_i),$$

where $\bar{\lambda}$ *is such that* $p_{\bar{\lambda}}(x) = \sum_{i=1}^n v_i p_{\lambda_i}(x)$ *and* $D(\cdot,\cdot)$ *is any distance function* [4] *that is convex in the density function of the second model.*

Since the true model λ^0 is unknown, it is not possible to find out which of the models $\{\lambda_i\}_{i=1}^n$ is more accurate in terms of the ideal quality cost function $Q_I(\cdot)$. However, from the above theorem, one can guarantee that the mean model will always provide an improvement over the average quality of the available models. The mean model is thus a good choice in terms of both $Q(\cdot)$ and $Q_I(\cdot)$, but it might not be a very interpretable model as it could have a large number of overlapping components. For example, if there are five local models, each being a mixture of three Gaussians, then the mean model will consist of 15 possibly overlapping components, which might not be desirable when we need a smaller number of disjoint clusters, as is usually the case. In general, it is more appropriate to require the combined model to belong to a specified parametric family $\mathscr{F}$, e.g., the family of all mixtures of three Gaussians. Therefore, we find the model in $\mathscr{F}$ that is closest to the mean model in terms of KL-divergence. From Theorem 2, this is also the exact solution to the DMC problem (2), i.e.,

$$\lambda_c^* = \arg\min_{\lambda_c \in \mathscr{F}} D_{\mathrm{KL}}(\bar{\lambda}, \lambda_c) \tag{3}$$

The new optimization problem (3) is difficult to solve directly using gradient descent techniques since closed form expressions of the objective function do not exist for most generative models. Therefore, we pose an approximate version of the above problem and solve it *via* Expectation–Maximization (Dempster et al., 1977). Let $\overline{\mathscr{X}} = \{x_j\}_{j=1}^m$ be a dataset obtained by sampling from the mean model. Consider the problem of finding the model $\lambda_c^a \in \mathscr{F}$ that maximizes the average log-likelihood of the dataset $\overline{\mathscr{X}}$, i.e.,

$$\max_{\lambda_c \in \mathscr{F}} L(\overline{\mathscr{X}}, \lambda_c) = \max_{\lambda_c \in \mathscr{F}} \frac{1}{m} \sum_{j=1}^{m} \log(p_{\lambda_c}(x_j)), \tag{4}$$

where $L(\overline{\mathscr{X}}, \lambda_c)$ is the average log-likelihood of $\overline{\mathscr{X}}$ with respect to λ_c. As the size of the dataset $\overline{\mathscr{X}}$ goes to ∞, the average log-likelihood converges to the cross entropy between the densities $p_{\bar{\lambda}}$ and p_{λ_c}, i.e., $Lim_{m\to\infty} L(\overline{\mathscr{X}}, \lambda_c) = Lim_{m\to\infty} \boldsymbol{E}_{x\in\overline{\mathscr{X}}}[\log(p_{\lambda_c}(x))] = \boldsymbol{E}_{x\sim p_{\bar{\lambda}}}[\log(p_{\lambda_c}(x))]$. Now, the cross entropy between any two densities is linearly related to the KL-divergence between them, i.e., $\boldsymbol{E}_{x\sim p_{\bar{\lambda}}}[\log(p_{\lambda_c}(x))] = H(\bar{\lambda}) - D_{\mathrm{KL}}(\bar{\lambda}, \lambda_c)$, where $H(\bar{\lambda})$ is the entropy of the mean model and is independent of λ_c. Hence, maximizing the cross entropy with respect to the mean model is equivalent to minimizing the KL-divergence with respect to the mean model. The approximate problem (4), therefore, converges to the DMC problem (3) as the size of $\overline{\mathscr{X}}$ goes to ∞.

Viewing (4) as a maximum-likelihood parameter estimation problem leads to Algorithm 1. The main idea is to first generate a dataset $\overline{\mathscr{X}}$ following the mean model $\bar{\lambda}$, using MCMC sampling techniques (Neal, 1993) such as the Gibbs sampling method and then, apply the expectation maximization (EM) algorithm (Ghosh, 2003) for mixture estimation to this dataset to obtain the clustering model $\lambda_c^a \in \mathscr{F}$ that maximizes its likelihood of being observed. The resulting model λ_c^a is a local minimizer of the approximate problem and not necessarily the same as the solution λ_c^* of the original unsupervised DMC problem (2). However, it is guaranteed to asymptotically converge to a locally optimal solution as the size of $\overline{\mathscr{X}}$ goes to ∞. In practice, one can use multiple runs of the EM algorithm and pick

[3] This result is true for any model λ^0 and is proved in the general form in the Appendices A and B.

[4] Examples of distance functions that are convex in the density function of the second argument include KL-divergence, L_1 distance and squared L_2 distance.

the best solution so that the obtained model is close to the globally optimal model.

3. Privacy costs

In this section, we quantify the privacy cost using ideas from information theory and also show that there is an inverse relation between the privacy of the local models and the quality of the mean model. We propose that the privacy, $\mathscr{P}(x,\lambda)$ of an object x given a model λ be defined in terms of the probability of generating the data object from the model. The higher the probability, the lower the privacy. More specifically, noting that the reciprocal of the probability is related to uncertainty (Cover and Thomas, 1991), we have $\mathscr{P}(x,\lambda) = (p_\lambda(x))^{-1}$, where p_λ is the probability density function of the generative model λ so that privacy corresponds to the uncertainty in predicting the original data from the model.

For vector data, $\mathscr{P}(x,\lambda) = 1$ implies that x can be predicted with the same accuracy as a random variable with a uniform distribution on a ball of unit volume. Further, $\mathscr{P}(x,\lambda) = 0$, (i.e., $p_\lambda(x) \to \infty$) and $\mathscr{P}(x,\lambda) \to \infty$ (i.e., $p_\lambda(x) = 0$) correspond to the situations where we have perfect prediction accuracy and perfect privacy respectively. We can now define the privacy, $\mathscr{P}(\mathscr{X},\lambda)$ of a dataset $\mathscr{X}$ with respect to the model as some function of the privacy of the individual data objects. The geometric mean has a nice interpretation as the reciprocal of the average likelihood of the dataset being generated by the model, assuming that the individual samples are i.i.d., i.e.,

$$\mathscr{P}(\mathscr{X},\lambda) = \left(\prod_{x\in\mathscr{X}} p_\lambda(x)\right)^{\frac{-1}{|\mathscr{X}|}} = 2^{\left(-\frac{1}{|\mathscr{X}|}\sum_{x\in\mathscr{X}}\log_2 p_\lambda(x)\right)}.$$

A higher likelihood of generating the dataset from the model implies a lower level of privacy. For example, consider vector space data being modeled by a mixture of Gaussians. A highly detailed model with Gaussians of vanishing variance, centered at each of the data objects gives away the entire dataset and has no privacy. This is to be expected as the probability density $p_\lambda(x)$ goes to ∞, for all data objects $x \in \mathscr{X}$ making the privacy measure go to 0^+. On the other hand, a very coarse model, say with a single Gaussian of high variance has a low likelihood of generating the data and hence, has a high privacy.

Intuitively, if the local models are more detailed, the combined model can be improved at the cost of decreased privacy. In particular, there is an asymptotic linear relation between the average logarithm of privacy (log-privacy) of the local models and the quality of the optimal mean model. Consider the local datasets $\mathscr{X}_i = \{x_{ij}\}_{j=1}^{m_i}$, $1 \leqslant i \leqslant n$. The data objects $\{x_{ij}\}_{j=1}^{m_i}$ can be considered to be i.i.d. random variables following the unknown true model λ^0. Hence, the log-privacy values of the objects w.r.t. the corresponding local model $h_{ij} = \log \mathscr{P}(x_{ij},\lambda_i) = -\log p_{\lambda_i}(x_{ij})$ $1 \leqslant j \leqslant m_i$ are also i.i.d. random variables with mean, $\mu = \boldsymbol{E}_{x\sim p_{\lambda^0}}[-\log(p_{\lambda_i}(x))]$, i.e., negative cross entropy of λ_i w.r.t. λ^0. By definition, the log-privacy of the local dataset $\mathscr{X}_i$ w.r.t. the corresponding local model λ_i denoted by $\bar{h}_i$ is just the mean or the empirical average of the log-privacy values of the individual objects. From the weak law of large numbers and Chebyshev inequality (Papoulis, 1984), the empirical average $\bar{h}_i$ (i.e., log-privacy of $\mathscr{X}_i$ w.r.t. λ_i) converges to the mean μ (i.e., cross entropy of λ_i w.r.t. λ_0) in probability when the size of the local dataset m_i tends to ∞. Applying the same argument for each data source, we find that the average log-privacy of the datasets w.r.t. the corresponding local models converges to the average cross entropy between the local models and λ_0. Further, this average cross entropy is identical to the cross entropy between the mean model and λ_0, which in turn is linearly related to the KL-divergence between them, i.e., the quality of the mean model. Hence, as the privacy of the local models increases, the quality of the mean model, which is the optimal unconstrained model, also goes up. On the other hand, when the privacy of the local models decreases, the mean model tends to be more accurate.

4. Experimental evaluation

In this section, we provide empirical evidence that for a reasonable global sample size and privacy

level, the global model obtained through the DMC algorithm is better than the best local model for different types of data not only in terms of KL-divergence, but also for other distance measures. We also present results that show how the privacy and quality costs vary with the local model resolution.

4.1. Datasets and learning algorithms

We performed experiments on four different types of data (shown in Table 1), which include four artificial datasets and one real data set (`pendigits`) from the UCI repository (Blake and Merz, 1998). Artificial data was preferred since the true generative model is known, unlike in the case of real data, and one can perform controlled experiments to better understand algorithmic properties. However, for the sake of stress testing, we also evaluated our method on real data by estimating the quality with respect to a centralized model learned after pooling all the data. The artificial datasets were generated from an appropriate mixture model by sampling independently using MCMC techniques. Each of these datasets was then divided equally among five local sites and local clustering models were trained using the individual partitions. In the case of `pendigits` dataset, we created two different kinds of partitionings. In the first partitioning (`pendigitsl`), each site has a dataset with equal class distribution (10% for each class) whereas in the second partitioning (`pendigits2`), each site has a dataset with unequal class distribution (20% for two classes and 7.5% each for the rest). The datasets can be downloaded from `http://www.lans.ece.utexas.edu/~srujana/gencl/data`.

To perform clustering, we use the appropriate EM based mixture estimation algorithms. In particular, the EM algorithms employed for the Gaussian models, von Mises–Fisher models and Hidden Markov models are based on (Dempster et al., 1977; Banerjee et al., 2003; Smyth, 1997) respectively. The EM algorithms at both the local and global level were run multiple times and the best solution was chosen so as to reduce the probability of getting stuck in local minima.

4.2. Performance metrics

For each setting, we computed the privacy costs of the local models and the ideal quality costs based on the various distance measures mentioned in Section 2. For the artificial datasets (Figs. 1, 4 and 5), the distance measures are with respect to the true generative model whereas for the `pendigits` dataset (Figs. 1 and 2), the distance measures are with respect to the centralized model. We compare the performance of our method (`global`) with the average (`average`), best (`minimum`) and worst (`maximum`) of the various local models, the mean of the local models (`mean`), and the centralized model (`centralized`).

4.3. Results and discussion

We first studied the performance of our distributed clustering algorithms on the Euclidean vector datasets for different choices of global sample size and local model resolution. Based on these experiments, we chose good values for the global sample

Table 1
Details of generative models and datasets

Data type	Model type	#Dim/sequence length	Total data size	#Sites	#Clusters
Vector	Gaussian	8	5000	5	5
Vector	`pendigitsl–2`				
	Spherical Gaussian	16	10,992	5	10
Directional vector	von Mises–Fisher	100	5000	5	5
Discrete sequence	Discrete HMM				
	5 states, 4 symbols	30	1000	5	5
Continuous sequence	Cont. HMM				
	5 states, 4 mixtures	30	600	3	5

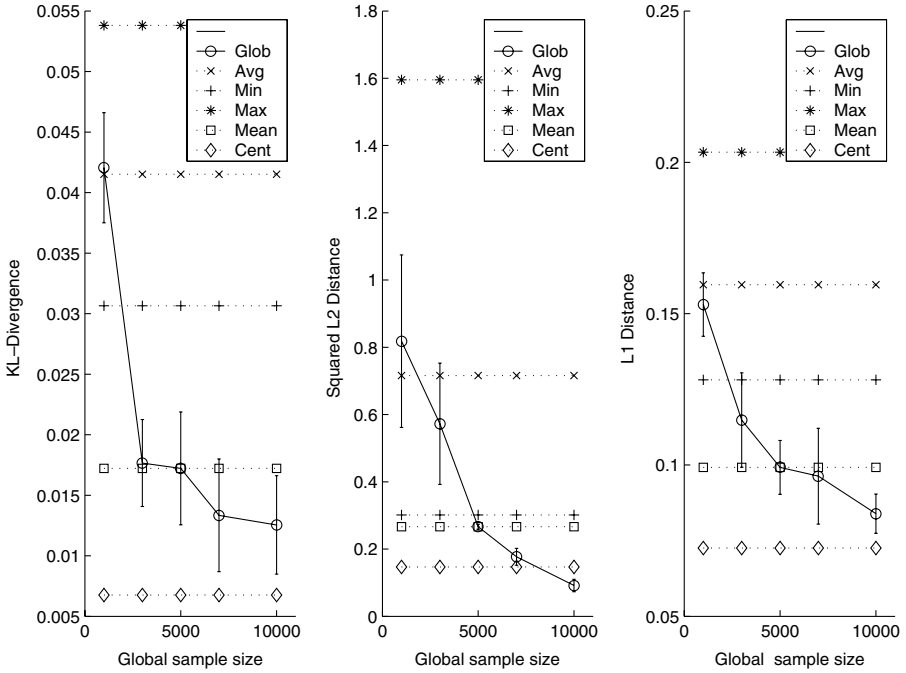

Fig. 1. Variation of global model quality with sample size for artificial Gaussian data. The error bars indicate the std-deviation over 10 trials.

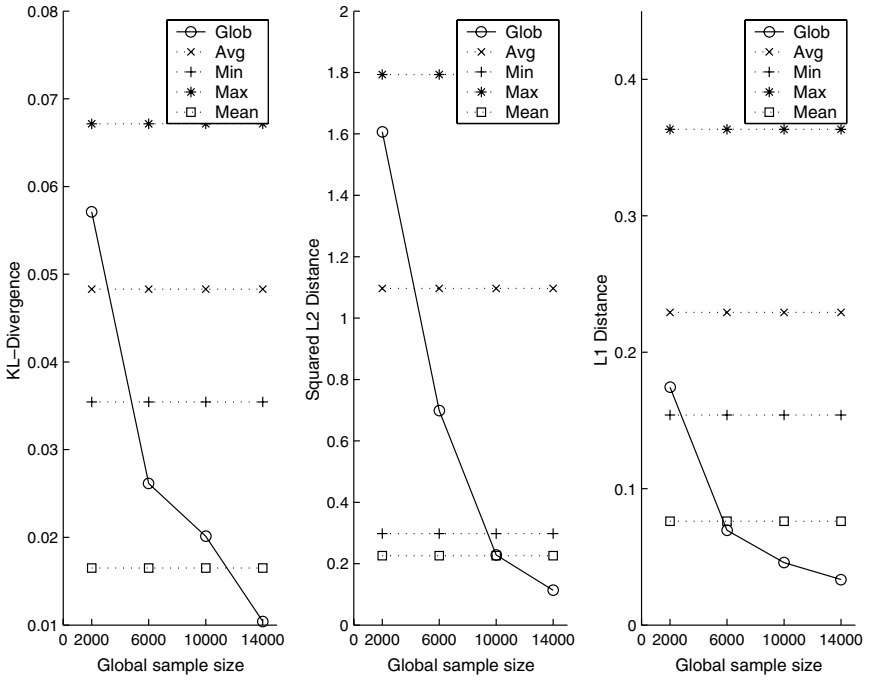

Fig. 2. Variation of global model quality (w.r.t. centralized model) with sample size for `pendigitsl`.

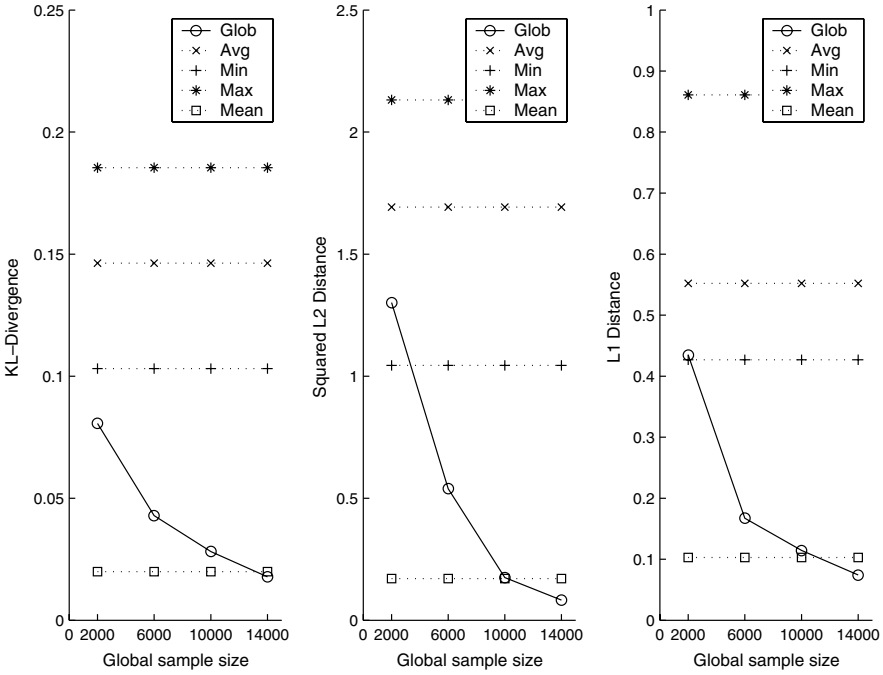

Fig. 3. Variation of global model quality (w.r.t. centralized model) with sample size for `pendigits2`.

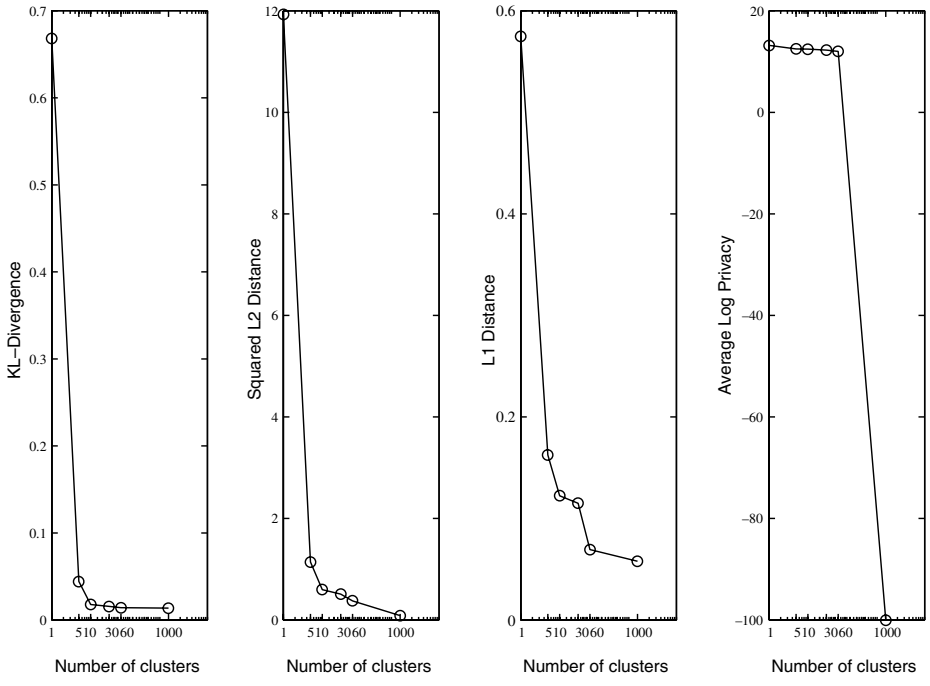

Fig. 4. Variation of privacy and global model quality w.r.t. base model resolution.

size and model resolution and applied our algorithms to different data types.

4.3.1. Variation of global model quality with sample size

An important step in our model-based learning approach is choosing the global MCMC sample size. Theoretical results indicate that the quality of model tends to improve as the sample size increases to ∞. In order to test this hypothesis, we ran our algorithm multiple times on the Euclidean vector datasets (artificial and `pendigitsl`) changing only the global sample size. Figs. 1 and 2 show how the quality of the different models varies with the sample size. The quality of the global model steadily improves with the number of global samples. When the sample size increases to that of the combined size of all the data sources, the global model is better than even the best of the local models.

4.3.2. Global model quality with varying local model bias

Fig. 3 shows the performance of our method on the `pendigits2` datasets where each local site has a different bias. In this case, all the models (i.e., local models, global model and mean model) have higher quality costs compared to Fig. 2 where all the local sites have the same class distribution. However, the performance gains of the global and the mean model over the individual local models are significantly higher than in Fig. 2, which suggests that our technique would perform better with increase in the diversity of the local datasets.

4.3.3. Variation of privacy and quality cost with model resolution

Another significant aspect of our framework is the trade-off between privacy restrictions and the quality of the combined model obtained. This

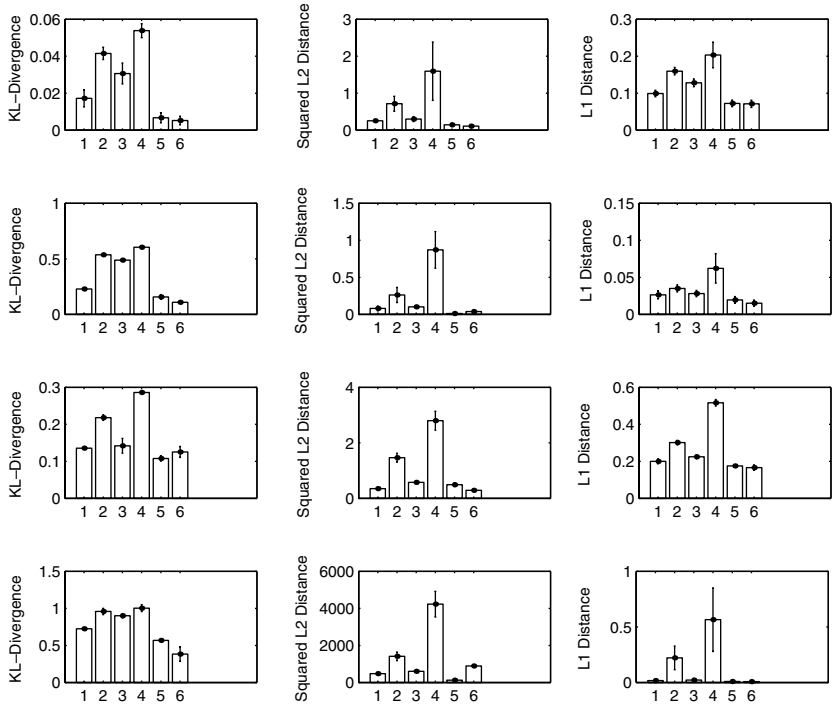

Fig. 5. Global model quality for different types of data. The rows 1–4 correspond to the results on the artificial vector, directional, discrete and continuous sequence data respectively. The white bar represents the average value and the error bars represents the standard deviation over 10 trials.

trade-off can be controlled by picking a suitable model resolution, e.g., number of clusters. Fig. 4 shows the variation of the average log-privacy and quality cost with the number of clusters in the local models for artificial Euclidean vector datasets. From the plots, we note that the average log-privacy as well as the quality costs decrease as the number of clusters increases. At a thousand clusters/location (i.e. one cluster per point) there is maximum loss of privacy, but because of the natural clusters in the data, comparable cluster quality can be obtained much before this limiting value, i.e., at a much smaller privacy cost.

4.3.4. Quality of global model for different data types

We also applied our learning algorithms to different data types to illustrate the generality of our approach. For a fair comparison, we chose the global sample size to be equal to the combined size of all the data sources and the model resolution of the local models to be the same as that of the true model. Fig. 5 shows the quality of the different models for all four data types. In all the cases, the global model performs better than the best local model. Moreover, the global model quality is in general closer to the quality of the centralized model than the average quality of the local models.

5. Concluding remarks

We presented a privacy preserving framework for distributed clustering that is applicable to a wide variety of data types and algorithms, so long as they can provide a generative model. Our approach is based on obtaining a global model from "virtual samples' generated from the local models using MCMC sampling techniques. We also proposed practical algorithms for distributed clustering based on this approach. Surprisingly good results are obtained with low communication overhead even though the sharing restrictions are rather severe. Further studies of distributed data mining for a wider range of data analysis goals/procedures and information sharing restrictions are warranted in order to unearth the full potential of this emerging pattern recognition area.

Acknowledgment

We would like to acknowledge support from the NSF under grants IIS-0307792 and IIS-0312471.

Appendix A. Proof of Theorem 1

$$\begin{aligned}
\sum_{i=1}^{n} v_i \mathrm{KL}(p_{\lambda_i} \| p_{\lambda_c}) &= \sum_{i=1}^{n} v_i E_{p_{\lambda_i}} \left[\log \left(\frac{p_{\lambda_i}(x)}{p_{\lambda_c}(x)} \right) \right] \\
&= \sum_{i=1}^{n} v_i E_{p_{\lambda_i}} \left[\log \left(\frac{p_{\lambda_i}(x) p_{\bar{\lambda}}(x)}{p_{\bar{\lambda}}(x) p_{\lambda_c}(x)} \right) \right] \\
&= \sum_{i=1}^{n} v_i E_{p_{\lambda_i}} \left[\log \left(\frac{p_{\lambda_i}(x)}{p_{\bar{\lambda}}(x)} \right) \right] \\
&\quad + E_v \left[E_{p_{\lambda_i}} \left[\log \left(\frac{p_{\bar{\lambda}}(x)}{p_{\lambda_c}(x)} \right) \right] \right] \\
&= \sum_{i=1}^{n} v_i \mathrm{KL}(p_{\lambda_i} \| p_{\bar{\lambda}}) \\
&\quad + E_{p_{\bar{\lambda}}} \left[\log \left(\frac{p_{\bar{\lambda}}(x)}{p_{\lambda_c}(x)} \right) \right] \\
&= \sum_{i=1}^{n} v_i \mathrm{KL}(p_{\lambda_i} \| p_{\bar{\lambda}}) \\
&\quad + \mathrm{KL}(p_{\bar{\lambda}} \| p_{\lambda_c}).
\end{aligned}$$

Appendix B. Proof of Theorem 2

From Jensen's inequality, we have for any convex function $f(\cdot)$ and random variable Y, $f(\boldsymbol{E}[Y]) \leqslant \boldsymbol{E}[f(Y)]$. Assuming $D(\lambda^0, \lambda)$ to be convex function in the first argument, i.e., $D(\lambda^0, \lambda) = f(p_\lambda)$, where $f(\cdot)$ is a convex function. Now, consider a random variable Y over the set $\{p_{\lambda_i}\}_{i=1}^n$ following a discrete distribution $\{v_i\}_{i=1}^n$, then using Jensen's inequality, we obtain

$$\begin{aligned}
f\left(\sum_{i=1}^{n} v_i p_{\lambda_i} \right) &\leqslant \sum_{i=1}^{n} v_i f(p_{\lambda_i}) \Longleftrightarrow f(p_{\bar{\lambda}}) \\
&\leqslant \sum_{i=1}^{n} v_i f(p_{\lambda_i}) \Longleftrightarrow D(\lambda^0, \bar{\lambda}) \\
&\leqslant \sum_{i=1}^{n} v_i D(\lambda^0, \lambda_i).
\end{aligned}$$

References

Agrawal, D., Aggarwal, C.C., 2001. On the design and quantification of privacy preserving data mining algorithms. In: Proc. Symp. on Principles of Database Systems (PODS), pp. 247–255.

Azoury, K.S., Warmuth, M.K., 2001. Relative loss bounds for on-line density estimation with the exponential family of distributions. Machine Learn. 43 (3), 211–246.

Banerjee, A., Dhillon, I., Ghosh, J., Sra, S., 2003. Generative model-based clustering of directional data. In: Proc. 9th Internat. Conf. on Knowledge Discovery and Data Mining (KDD), pp. 19–28.

Banerjee, A., Merugu, S., Dhillon, I., Ghosh, J., 2004. Clustering with bregman divergences. In: Proc. SIAM Internat. Conf. on Data Mining, pp. 234–245.

Blake, C.L., Merz, C.J., 1998. UCI repository of machine learning databases. Available from: <http://www.ics.uci.edu/~mlearn/MLRepository.html>.

Cadez, I.V., Gaffney, S., Smyth, P., 2000. A general probabilistic framework for clustering individuals and objects. In: Proc. 8th Internat. Conf. on Knowledge Discovery and Databases (KDD), pp. 140–149.

Chan, P., Stolfo, S., Wolpert, D., 1996. Integrating multiple learned models for improving and scaling machine learning algorithms. Machine Learn. 36 (1–2).

Cover, T.M., Thomas, J.A., 1991. Elements of Information Theory. Wiley.

Dempster, A.P., Laird, N.M., Rubin, D.B., 1977. Maximum likelihood from incomplete data via the EM algorithm. J. Roy. Statist. Soc., Series B (Methodological) 39 (1), 1–38.

Dhillon, I.S., Modha, D.S., 1999. A data-clustering algorithm on distributed memory multiprocessors. In: Proc. 7th Internat. Conf. on Knowledge Discovery and Data Mining (KDD), pp. 245–260.

Farkas, C., Jajodia, S., 2002. The inference problem: A survey. SIGKDD Explorations 4(2), 6–11.

Fayyad, U.M., Reina, C., Bradley, P.S., 1998. Initialization of iterative refinement clustering algorithms. In: Proc. Internat. Conf. on Machine Learning (ICML), pp. 194–198.

Fred, A.L.N., Jain, A.K., 2002. Data clustering using evidence accumulation. In: Proc. Internat. Conf. on Pattern Recognition (ICPR), pp. 276–280.

Ghosh, J., 2003. Scalable clustering methods for data mining. In: Ye, N. (Ed.), Handbook of Data Mining. Lawrence Erlbaum, pp. 247–277.

Ghosh, J., Merugu, S., 2003. Distributed data mining with limited knowledge sharing. In: Proc. 5th Internat. Conf. on Advances in Pattern Recognition (ICAPR), pp. 48–53.

Johnson, E., Kargupta, H., 1999. Collective, hierarchical clustering from distributed, heterogeneous data. In: Zaki, M., Ho, C. (Eds.), Large-Scale Parallel KDD Systems, LNCS, vol. 1759, pp. 221–244.

Kargupta, H., Dutta, S., Wang, Q., Sivakumar, M., 2003. Random data perturbation techniques and privacy preserving data mining. In: Proc. IEEE Internat. Conf. on Data Mining (ICDM), pp. 99–106.

Klusch, M., Lodi, S., Moro, G., 2003. Distributed clustering based on sampling local density estimates. In: Proc. Internat. Joint Conf. on Artificial Intelligence (IJCAI), pp. 485–490.

Merugu, S., Ghosh, J., 2003. A probabilistic approach to privacy-sensitive distributed data mining. In: Proc. 6th Internat. Conf. on Information Technology (CIT), pp. 405–410.

Neal, R.M., 1993. Probabilistic inference using Markov Chain Monte Carlo methods. Tech. Rep. CRG-TR-93-1, Department of Computer Science, University of Toronto.

Papoulis, A., 1984. Probability, Random Variables, and Stochastic Processes. McGraw-Hill, New York.

Pinkas, B., 2002. Cryptographic techniques for privacy-preserving data mining. SIGKDD Explorations 4 (2), 12–19.

Smyth, P., 1997. Clustering sequences with Hidden Markov models. In: Advances in Neural Information Processing Systems, pp. 648–654.

Strehl, A., Ghosh, J., 2002. Cluster ensembles—a knowledge reuse framework for combining partitionings. J. Machine Learn. Res. 3, 583–617.

Tasolis, D., Vrahatis, M., 2004. Unsupervised distributed clustering. In: Proc. IASTED Internat. Conf. on Parallel and Distributed Computing and Networks.

Vaidya, J., Clifton, C., 2003. Privacy-perserving *k*-means clustering over vertically partitioned data. In: Proc. 11th Internat. Conf. on Knowledge Discovery and Data Mining (KDD), pp. 206–215.

Yamanishi, K., 1998. Distributed cooperative Bayesian learning strategies. Inform. Comput. 150, 22–56.

Zhong, S., Ghosh, J., 2003. A unified framework for model-based clustering. J. Machine Learn. Res. 4, 1001–1037.