Building Predictors from Vertically Distributed Data

Sabine McConnell David B. Skillicorn

School of Computing
Queen's University
Kingston, Canada
{mcconnell,skill}@cs.queensu.ca

Abstract

Due in part to the large volume of data available today, but more importantly to privacy concerns, data are often distributed across institutional, geographical and organizational boundaries rather than being stored in a centralized location. Data can be distributed by separating objects or attributes: in the homogeneous case, sites contain subsets of objects with all attributes, while in the heterogeneous case sites contain subsets of attributes for all objects. Ensemble approaches combine the results obtained from a number of classifiers to obtain a final classification. In this paper, we present a novel ensemble approach, in which data is partitioned by attributes. We show that this method can successfully be applied to a wide range of data and can even produce an increase in classification accuracy compared to a centralized technique. As an ensemble approach, our technique exchanges models or classification results instead of raw data, which makes it suitable for privacy preserving data mining. In addition, both final model size and runtime are typically reduced compared to a centralized model. The proposed technique is evaluated using a decision tree, a variety of datasets, and several voting schemes. This approach is suitable for physically distributed data as well as privacy preserving data mining.

1 Introduction

Data mining is concerned with the extraction of non-trivial, novel and potentially useful knowledge from large databases. Sequential data mining techniques have been applied successfully to a wide range of areas such as customer relationship management, web mining, science, engineering and medicine. However, a need for distributed and parallel data mining techniques has emerged over the past years.

The motivation for parallel and distributed data mining is at least twofold. Perhaps the most obvious reason is the sheer volume of data available today. For example, data available on the world wide web roughly doubles every nine months while scientific data from surveys and simulations are generated faster than scientists can analyze it. One example is NASA's Earth Observing System (EOS), which produces roughly one terabyte of data each night. Clearly, data mining, which is typically an interactive process, results in prohibitively large runtimes for such massive datasets. Second, the introduction of deadlines emphasizes the need for parallel and distributed data mining. In tasks such as credit card fraud and intrusion detection, a quick response time is crucial. Third, data are considered an asset and cannot readily be shared across organizational and institutional boundaries. For example, financial institutions are not willing (or able) to share confidential transaction information with each other, but can still benefit from exchanging models and results rather than raw data to improve on the overall prediction accuracy. Finally, to a lesser extent, certain distributed

data mining techniques can also be used to improve prediction accuracy as compared to a centralized technique.

In this paper, we investigate an ensemble approach for vertically partitioned data. Predictors are built from datasets containing single attribute values only and then combined using simple voting schemes. We evaluate our ensemble approach using a decision tree algorithm and a variety of datasets.

The remainder of this paper is organized as follows. Section 2 gives background definitions while Section 3 introduces related work. The proposed method is presented in Section 4. Section 5 shows the results of the experimental evaluation of the technique. These results are discussed in Section 6 and we conclude with Section 7.

2 Background and Definitions

Throughout this paper, we assume that data is represented in matrix form, where rows of the matrix contain objects and columns the attributes known about the objects. Data can then be distributed across the sites involved in the computation in two main ways. *Horizontally distributed* data contains all attribute values for a subset of the objects at each site. In contrast, *vertically distributed* data contains a subset of the attributes for all objects at each site. Horizontally and vertically partitioned data are also referred to as *homogeneous* and *heterogenous* data. Figure 1 shows a homogeneous data partitioning over two sites, while Figure 2 show a corresponding heterogenous case. Note that in the heterogeneous setting, a common key is assumed to match subsets of attributes to objects. Combinations of homogeneous and heterogeneous partitions as well as overlap among sites are also possible.

Data mining techniques, which aim at extracting new and potentially useful information from the data, need to be adjusted to reflect the distribution of the data. In the more common homogeneous setting, techniques such as collections of classifiers, or ensembles, and parallelization of sequential algorithms have been utilized. In contrast to the ensemble technique presented in this paper, existing work for heterogeneous data partitions focuses either on constructing a single classifier using secure protocols or, as in the *Collective Data Mining framework* (CDM) introduced by Kargupta *et al.* [12], transformations of the functions to be learned.

The proposed technique for collections of classifiers over heterogeneous data is closely related to ensembles over horizontally partitioned data. Ensembles are collections of base classifiers, which collectively determine the outcome for the data instances, for example in techniques such as bagging and boosting. These base classifiers are constructed by either applying the same algorithm to different subsets of the data, by changing input parameters such as

ID	Age	Income	Married	Children
1	24	21000	yes	1
2	35	34000	no	3
3	61	41000	yes	0
4	22	19000	no	1

(a) Site 1

ID	Age	Income	Married	Children
5	19	27000	no	0
6	43	30000	yes	2

(b) Site 2

Figure 1: Homogeneous partitioning

ID	Age	Income	Married
1	24	21000	yes
2	35	34000	no
3	61	41000	yes
4	22	19000	no
5	19	27000	no
6	43	30000	yes

(a) Site 1

ID	age	Children
1	24	1
2	35	3
3	61	0
4	22	1
5	19	0
6	43	2

(b) Site 2

Figure 2: Heterogeneous partitioning

the number of hidden nodes in a neural network for example or by using different classifiers on the same data. In general, ensemble techniques are applied to *weak* classifiers, for which the results are dependent on the underlying datasets. Examples for such weak learners are neural networks and decision trees. Ensemble techniques were originally introduced to increase the accuracy of the final classification, but can also be utilized for distributed data mining by partitioning datasets and algorithms over a number of processors and combining the results in a central location. Such methods have been successfully applied to horizontally distributed data, because real life data typically exhibit a large number of repetitions where the same or similar samples are represented multiple times. This is not the case for a vertically partitioned setting. Therefore, ensembles do not trivially extend to the heterogeneous case.

In this paper, we restrict our technique to a supervised setting. In this case, the predictors are built from a labeled training set for which the classifications are known. The performance of these predictors is then evaluated using a previously unseen test set. The particular technique used in this paper is a decision tree algorithm, which builds tree-like structures from the training sets. Leaf nodes are associated with target classes, while internal nodes are labeled with attributes and represent tests to be performed on the data. The outcome of the test is determined by the value of the attribute used to label the node. Initially, all data is associated with the root. A splitting criterion is then used to separate the data, with the disjoint subsets now associated with the children of the root. The procedure is then applied recursively until the nodes contain mainly samples of one class. To classify previously unseen data, the test associated with the root is applied to a data instance, which subsequently follows a path down the tree until it reaches a leaf node. The path a datum follows, and therefore the leaf node it is assigned to, depends on the attribute values of the datum.

3 Related Work

Originally developed and utilized to increase the overall classification accuracy, ensemble techniques have been applied to a wide range of techniques such as decision trees and neural networks. Hansen and Salamon [8] show that the classification accuracy of an ensemble can surpass that of the single best classifier in the collection, providing that the classifiers are both accurate and diverse. Approaches such as bagging, boosting and stacking as well as random forests fall into this category. A survey of various approaches to combine ensembles is presented in Bahler and Navarro [1]. Meta-learning, which can be regarded as an extension of the ensemble approach, is loosely defined as learning about learning or from learned knowledge. The use of a second level for learning was originally suggested by Wolpert [16] and used in a supervised setting by Chan and Stolfo [3] [4]. Outputs from a set of classifiers are used as input to another classifier, which then produces the final prediction.

Both approaches for combining classifiers, ensembles and meta-learning, are complements to the data mining techniques. As such, they are independent of the underlying algorithms, thus rendering them applicable to a wide range of learning techniques without changes to the original code. Advantages of these techniques are the potential increase in accuracy, their scalability to larger datasets as well as their portability. In contrast, drawbacks include the increased need for runtime resources and the data dependency of the results. Even though ensembles and meta-learning have been applied extensively to horizontally partitioned data, in which sites contain subsets of objects for all attributes, to our best knowledge no approaches to vertically partitioned data are reported in the distributed data mining literature.

In contrast to ensembles, which produce a final prediction from a collection of classifiers, alternative data mining techniques for vertically partitioned data are the Collective Data Mining framework and the construction of a single classifier from the distributed data in privacy preserving data mining.

The collective data mining (CDM) framework was specifically designed for heteroge-

neous data. It is based on the fact that any function (such as the decision tree to be learned for example) can be expressed as the sum of a set of a possibly infinite number of basis functions.

$$f(x) = \sum_{k \in \Xi_I} w_k \Psi_k(x) \quad (1)$$

where Ξ is a set of basis functions, the k-th basis function is denoted by Ψ_k and the corresponding coefficient by w_k. Ξ_I denotes the set of indices of the basis functions. Rather than learning the original function f through the original representation of the data, CDM aims at learning the function f using an alternate basis by extracting the coefficients for the basis functions from the distributed datasites. To ensure computation polynomial in the number of samples, the sum in Equation 1 has to be finite, which imposes a bound on the accuracy that can be achieved.

After choosing an appropriate representation for the function to be represented, the coefficients of the basis functions are computed for each of the local datasets. These coefficients, in some cases along with a small representative sample of the local datasets, are then communicated to a central site, where the final model is generated. Through the use of small representative samples or the coefficients alone, the need of communication of large amounts of data is eliminated. This comes at the cost of a reduction in accuracy, since the original functions are approximated. This implies that, contrary to ensemble techniques such as the one introduced in this paper, and meta-learning approaches, the achieved accuracy in the CDM framework will always be lower than that of a predictor computed from a centralized dataset. In addition, the CDM framework relies on the communication of data samples to a central site for most techniques, which limits its use for privacy preserving data mining.

In addition to the fact that a large number of coefficients need to be zero or negligible as determined by a threshold to allow for polynomial computation of the coefficients, the coefficients also have to be approximated. Considering the need for linear time algorithms, this polynomial complexity represents a major limitation for this approach, in essence rendering it not scalable to larger datasets. Additionally, since the representation is approximate, the framework might not be suitable for all techniques, especially for those aimed at outlier detection.

The Collective Data Mining Framework encompasses a wide range of techniques. Hershberger and Kargupta established the use of regression within the CDM framework [9]. In this approach, the coefficients representing cross terms are estimated directly from the local coefficients sent to a central site, eliminating the need for transmitting raw data based on a wavelet representation. For this particular technique, the communication cost is independent of the sample size, since only the basis coefficients are communicated. In the same paper, the authors use this approach to include linear discriminant analysis in their framework. CDM has also been extended to clustering using Principal Component Analysis [11], genetic algorithms [10] and Bayesian networks [5].

In contrast to the Collective Data Mining Framework, which builds on a transformation of the function to be learned, privacy preserving techniques focus on secure sum and secure product computation from vertically partitioned data. The construction of a single classifier from vertically distributed data has been applied to a number of techniques, for example decision trees [7] and k-means clustering [14]. In this approach, the focus lies on secure computation, where as little as possible about local data is disclosed to other sites. Secure sum and secure product computation increase the computational complexity and communication requirements of the technique, while, at the same time, the accuracy is bounded by that of a single sequential classifier applied to a centralized dataset.

4 Ensembles for Vertically Partitioned Data

In our ensemble technique for vertically partitioned data, local predictors are built from training sets, which are partitioned by attributes. The predictions obtained from those local classifiers for the global test set are then combined through a voting scheme.

We constructed training sets of the size of the original datasets by drawing randomly with

Algorithm 1 Vertical Ensembles

Construct a training set
Construct an out-of-bag test set
Partition the training set by attributes
for all partitions of the training set
Distribute one partition of the training set to each of the local sites, along with the test set
end for
for all local sites
Build a sequential predictor (decision tree) from the local partitions of the training set
Obtain classifications for each test instance by applying the local decision tree to the test set
Communicate the classification of the test instances to a central site
end for
Combine the predictions of the classifiers over the test set

replacement. The training set consisted then of those samples not chosen to be included in the training set, for an out-of-bag estimate of the accuracy [2]. The training sets were initially partitioned by splitting them vertically into subsets containing single attributes for each object contained in the database to simulate a worst case scenario. In subsequent runs, increasing numbers of attribute values were grouped together in the default order of their appearance in the database to investigate the effect on the accuracy with decreasing number of sites. This was repeated with the number of sites varying from 2 to 11, depending on the dataset. Of course, in a real world setting, the data is already partitioned, typically with a wide range of possible subsets.

For each of the vertically partitioned subsets, we build separate decision tree classifiers, which were subsequently combined using two different voting schemes. For the simpler scheme, each classifier was given the same vote and the test datum was assigned to the class with the highest number of votes. In a more sophisticated voting scheme, each vote was weighted with the probability that the decision classifier gave to the class for that specific sample. The final prediction is tested on the out-of-bag samples.

For our experiments, we used the J48 decision tree implemented in WEKA, which is based on the C4.5 decision tree algorithm. WEKA is an open source data mining package for tasks such as regression and visualization and contains a wide range of machine learning algorithms. WEKA is available for download at *http://www.cs.waikato.ac.nz/ml/weka/*.

Name	Samples	Targets	Attr.	Type
Balance	625	3	4	numeric
Iris	150	3	4	numeric
Parity	100	3	9	binary
LED	1024	10	7	binary
Haberman	306	2	3	numeric
Contr.	1473	4	9	all
Pima	768	2	8	numeric
Macho	10970	3	12	numeric
Star	4192	2	14	numeric
Mushroom	8125	2	21	numeric

Table 1: Overview of datasets

Even though the vertical voting approach was initially implemented utilizing a decision tree technique, it can be applied to a wide range of data mining techniques, including rule-based approaches, neural networks and genetic algorithms.

5 Experimental Results

We selected a wide variety of sample datasets to evaluate the technique. With the exception of the astronomical data the MACHO dataset and the Star/Galaxy dataset, all datasets were taken from the UCI repository. The MACHO data was taken from the from the catalogue of Variable Stars in the Large Magellanic Clouds (MACHO), available through the VizierR database of astronomical catalogues. The Star/Galaxy dataset was used by Odewahn in [13] to illustrate the use of neural networks for the separation of stars and galaxies in data obtained from the Palomar Sky Survey.

The number of samples in the datasets ranges from 100 to 10970, while the number of attributes fall between 3 and 22, with a mixture of binary and numeric attributes. The number of target classes varies between 2 and 10. Table 1 shows the number of samples, attributes and target classes as well as the attribute types of the 10 datasets used for the experimental evaluation. The distributed execution was

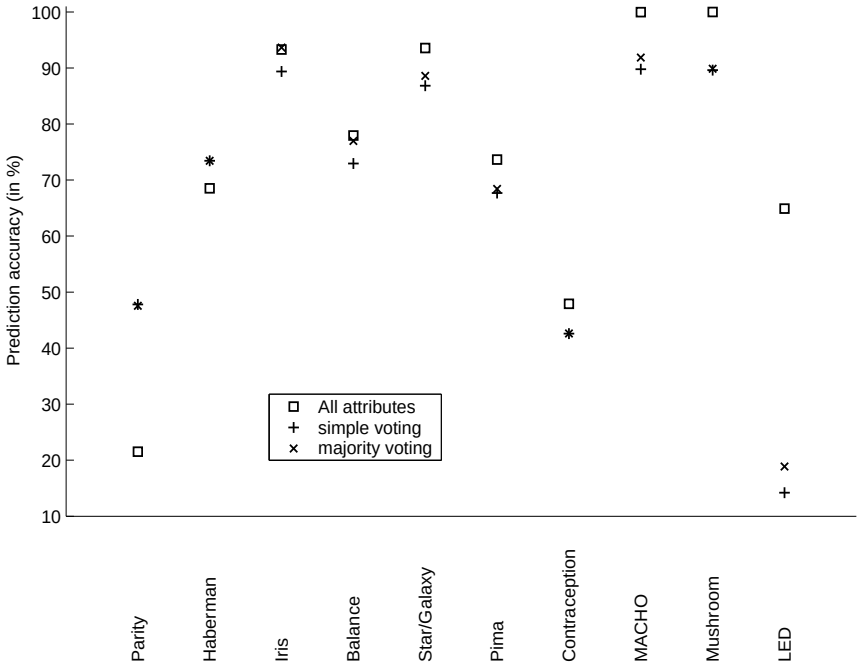


Figure 3: Accuracy comparisons: sequential vs. single attribute per partition

simulated on a Sun Ultra 1 processor. We measured the achieved accuracies, model sizes and time to build the decision tree in both the distributed and centralized approaches and all results were averaged over 50 trials.

Figure 3 gives the results for the accuracies achieved for all datasets in the worst case scenario where each local site contains the values of one attributes besides a common key. In this case, correlations between attributes are completely ignored when building the individual classifier, since no exchange of raw data, models or results takes place at this stage. Note that this makes it suitable for a case in which the communication of raw data is not desirable due to cost of communication or privacy issues, when parties cannot or do no want to share their local data. The results are shown for both the simple and weighted voting scheme and compared to the accuracy achieved from a centralized dataset. As expected, the more sophisticated voting approach, where each vote is weighted by the probability the classifiers assign to specific samples and classes, is at least as good as that of the simple voting scheme, which assigns equal weights to each classifier.

For two of the datasets (Haberman and Parity) the accuracies for both voting schemes in the distributed case are higher than the accuracies achieved from a centralized dataset. In two additional cases, the more sophisticated majority voting scheme is equivalent to the centralized approach (Iris and Balance). The performance of the distributed approach is slightly worse than the centralized approach for the majority of the remaining cases, and fails badly in the LED dataset. For comparison, Hershberger and Kargupta achieved a 90.3 % accurate classification for the IRIS dataset using 3-fold cross validation in the Collective Data Mining framework [9]. The fact that the accuracy is higher when seeing only single attributes of the data and then combining the prediction could possibly be due to the fact that each of the attributes is a good predictor on a large number of objects, combined with a minimal overlap of these ranges for different attributes.

Figure 4 gives the accuracies for each dataset with varying numbers of sites. Shown are the results for majority voting, where the results obtained from a centralized dataset (shown as boxed values) are included for comparison. As

expected, we see the general pattern of increasing accuracies with increasing numbers of variables at the sites, with the exception of the Parity and Haberman datasets, for which the overall accuracy is less in a centralized approach.

More experiments are underway to determine the reason for the peaks and dips, for example in the Balance dataset for 2 variables and the Pima dataset for 4 variables per site. We suspect that this is due to the ordering of the variables that were grouped together, two uncorrelated attributes placed with each other might be responsible for the dips. In general, a substantial increase in accuracy is achieved for small groupings of attributes compared to the worst case of single attributes at each site. For example, accuracies in the Pima and Contraception datasets for two variables at each of the site is comparable to that of the centralized approach, while for the Balance dataset, the accuracy is even higher.

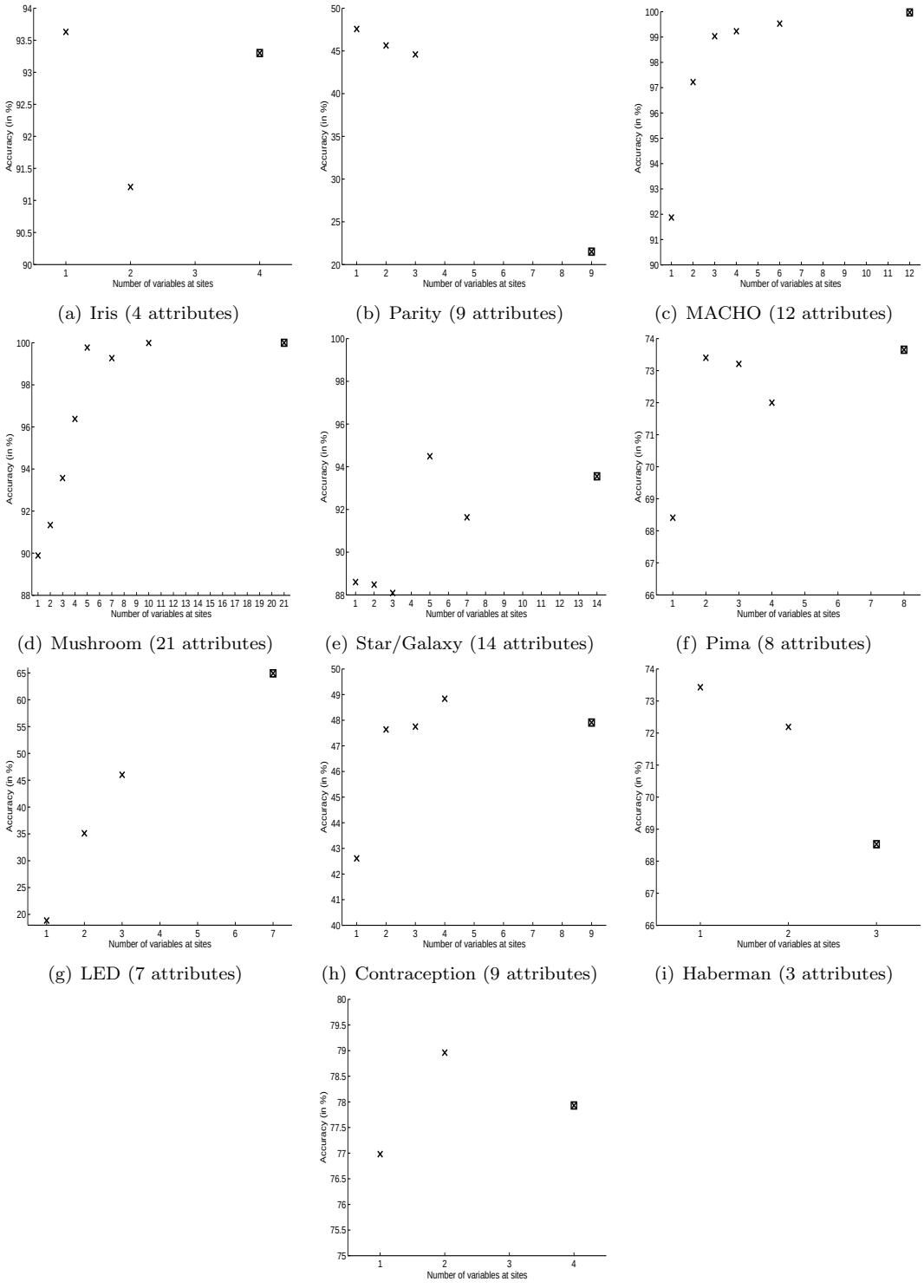
Figure 5 shows the model sizes for the different datasets with varying numbers of sites. The average size of the decision tree in the centralized approach and the average model size at each of the sites are shown. With the exception of the Macho dataset, the model sizes increase as more complex models are build with an increase of number of attributes at each site. Incidentally, the highest increase in model size with increasing number of attributes per site occurs for the Parity dataset, for which the resulting accuracy in the distributed case is higher than the accuracy achieved by a centralized approach. The times to build the decision trees, both in the distributed and centralized approach are given in Figure 6. Parallel to the increase in model size, the time to build the tree for the Parity dataset increases sharply for a centralized approach compared to the use of two sites. Similar results are also observed for the Haberman dataset.

From these results, it is clear that our approach has the potential of an increase in accuracy combined with a reduction in model size and runtime as compared to a centralized approach. In the most extreme case, an increase in accuracy was achieved with both a reduction in model size and runtime for a worst case scenario where each site contains a single attribute only. Furthermore, for those datasets where the centralized approach had a higher accuracy, the results for the distributed case improve dramatically when increasing the number of attributes per site to two, with the exception of the Star/Galaxy dataset only. For each of the datasets except the LED data, we observed at least one distributed case with an accuracy comparable to (within one percentage point) or higher than a centralized approach. Moreover, a small loss in accuracy as seen in the worst case scenario for a subset of the data can be acceptable if deadlines are involved and/or the size of the models are of concern.

The proposed technique can be deployed in a number of ways depending on the data mining needs. If privacy of data is of main concern, the classifiers can be built locally by each of the parties involved without the exchange of raw data or even models. In such a scenario, each party can classify a previously unseen dataset and only transmit the class predicted by each classifier to a central site. Such an approach requires the distribution of samples to be classified to each of the local sites, for a communication requirement of order $O(np)$. Here, there is a tradeoff between preserving the privacy of the data and the achieved accuracy.

Alternatively, the predictors, or models, themselves can be transmitted to a central datasite and then applied to new datasets to be classified. This trades the communication of the data to the local sites with the communication of the classifiers, which only has to happen initially. If we assume that the number of attributes is smaller than the number of samples, which is a typical real world scenario, then the communication cost of our approach is equivalent to the communication cost given by Vaidya and Clifton [15] for privacy preserving association rule mining.

Furthermore, the technique potentially reduces the overall model size and runtime, which can be crucial in the presence of deadlines and extremely large datasets: most data mining techniques assume the model to be stored in main memory. Promising results are also achieved by the application of our technique to data that is both horizontally and vertically distributed and for which no distributed data mining techniques are known. Preliminary results indicate that the loss of classification ac-



(a) Iris (4 attributes)

(b) Parity (9 attributes)

(c) MACHO (12 attributes)

(d) Mushroom (21 attributes)

(e) Star/Galaxy (14 attributes)

(f) Pima (8 attributes)

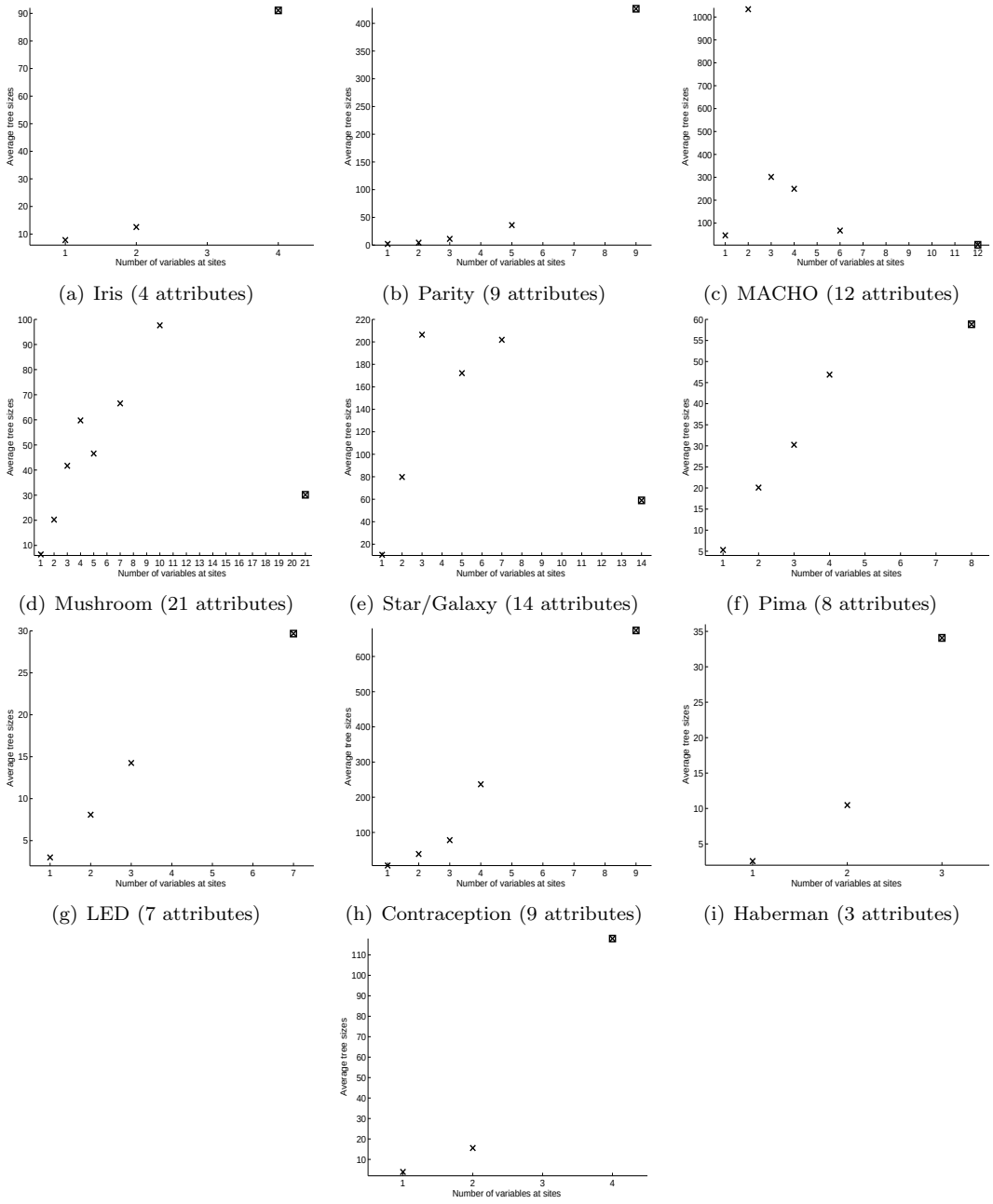
(g) LED (7 attributes)

(h) Contraception (9 attributes)

(i) Haberman (3 attributes)

(j) Balance (4 attributes)

Figure 4: Accuracies for increasing number of attributes per site: We observe an increase, with the exception of the Parity and Haberman datasets in 4(b) and 4(i)



(a) Iris (4 attributes)

(b) Parity (9 attributes)

(c) MACHO (12 attributes)

(d) Mushroom (21 attributes)

(e) Star/Galaxy (14 attributes)

(f) Pima (8 attributes)

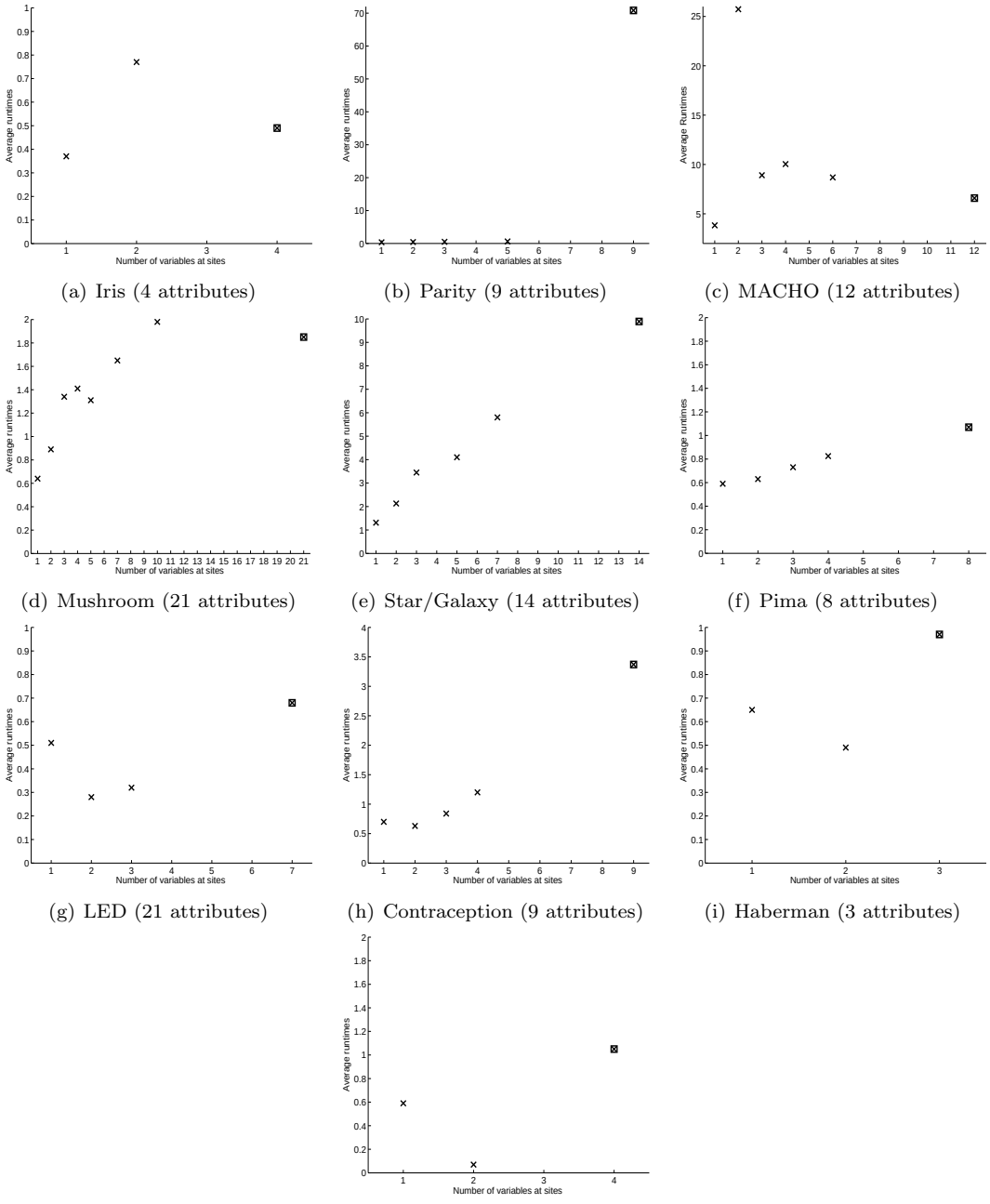
(g) LED (7 attributes)

(h) Contraception (9 attributes)

(i) Haberman (3 attributes)

(j) Balance (4 attributes)

Figure 5: Model sizes for increasing number of attributes per site: With the exception of the Macho dataset, all data show an increase in model size as the number of attributes at each site increases.



(a) Iris (4 attributes)

(b) Parity (9 attributes)

(c) MACHO (12 attributes)

(d) Mushroom (21 attributes)

(e) Star/Galaxy (14 attributes)

(f) Pima (8 attributes)

(g) LED (21 attributes)

(h) Contraception (9 attributes)

(i) Haberman (3 attributes)

(j) Balance (4 attributes)

Figure 6: Time to build decision trees for increasing number of attributes per site: All datasets exhibit an increase in time needed to build the decision tree as the number of attributes per site increases.

curacy stemming from the independence assumption of the attributes is offset by the gains from an ensemble approach.

6 Theoretical Justification

Due to the fact that vertically partitioned data does not exhibit the repetitiveness of horizontally partitioned data, the success of a vertical voting approach such as the one presented in this paper comes as somewhat of a surprise on a first glance. However, the effectiveness of the technique, when applied to the majority of the datasets investigated, is related to the success of the Simple Bayesian Classifier as shown by Domingos and Pazzani [6].

Bayes' Theorem expresses $p(H_i \mid x)$, the posterior probability of a hypothesis H_i given a sample x, as follows:

$$p(H_i \mid x) = \frac{p(x \mid H_i) \times p(H_i)}{p(x)} \qquad (2)$$

where $p(x \mid H_i)$ is the posterior probability of sample x given a Hypothesis H_i, $p(H_i)$ is the prior probability of H_i and $p(x)$ is the prior probability of x. All terms on the right hand side are estimated from training data, where the difficulty lies in the fact that the estimation of the first term is exponential in the number of variables. One solution to this problem is to assume that the p variables are conditionally independent, so that the first term can be factorized as

$$p(x \mid H_i) = \prod_{j=1}^{p} p(x_j \mid H_i) \qquad (3)$$

which is known as simple, naive or first order Bayes approach. This has the advantage that now the computation is linear in both the number of samples and attributes, which is optimal for any data mining algorithm that is not based on sampling from the original data. Naive Bayes techniques are robust with respect to irrelevant attributes, and experience has shown that this approach works surprisingly well for a wide range of problems and that adding terms relating to combinations of variables does not necessarily mean large improvements. Domingos and Pazzani show that this is due to the fact that although the probabilities are estimated wrongly, their ranking is preserved, i.e. the class that should be assigned to with the highest probability is still the highest under the naive Bayes assumption. The authors also show that the naive Bayes approach is optimal for disjunction and conjunction. By presenting empirical evidence, Domingos and Pazzani also show that the there is a small correlation between the difference in accuracy achieved by the Naive Bayes approach compared to other techniques they used and the attribute dependence found in the datasets. By distributing the attributes across single sites in our distributed voting approach, we are essentially assuming independence amongst attributes, where terms corresponding to the conditional dependence of the attributes are ignored. Similar to the first order Bayes assumption, our vertical voting scheme classifies a large number of samples correctly because the ranking of the class assignments are preserved to a large extent.

7 Conclusions

We presented an ensemble approach for vertically partitioned data. Evaluation using a decision tree technique on a variety of datasets show that we achieve only small losses of prediction accuracy in the worst case when each site learns from a single attribute. Moreover, in some cases an increase in prediction accuracy is observed. The fact that only models or votes are exchanged, combined with a reduction in model sizes and time required to build the classifiers, makes the technique suitable for cases where data is physically distributed, or for privacy preserving data mining.

Acknowledgements

The author would like to thank the Natural Sciences and Engineering Research Council of Canada and the Canadian Space Agency for support of this project.

About the Authors

Sabine McConnell is a Ph. D. candidate in the School of Computing at Queen's University in Kingston, Canada. She is currently investigating distributed data mining techniques with a particular focus on applications in astrophysics.

David Skillicorn is a Professor in the School of Computing at Queen's University in Canada. His research is in smart information management, both the problems of extracting and sharing useful knowledge from data, and the problems of accessing and computing with large datasets that are geographically distributed.

References

[1] D. Bahler and L. Navarro. Methods for combining heterogeneous sets of classifiers. Technical report, Artificial Intelligence Laboratory, Department of Computer Science, North Carolina State University, Raleigh, NC.

[2] L. Breiman. Out-of-bag estimation. Technical report, Statistics Department, University of California, Berkeley, Ca., 1996.

[3] P. Chan and S. Stolfo. Experiments in multistrategy learning by meta-learning. In *Proceedings of the Second International Conference on Information and Knowledge Management*, pages 314–323, Washington, DC, 1993.

[4] P. Chan and S. Stolfo. Toward parallel and distributed learning by meta-learning. In *Working Notes AAAI Workshop, Knowledge Discovery in Databases*, pages 227–240, 1993.

[5] R. Chen, K. Sivakumar, and H. Kargupta. Learning Bayesian network structure from distributed data. In *Proceedings of the SIAM International Data Mining Conference, San Francisco, USA*, 2003.

[6] P. Domingos and M. Pazzani. Beyond Independence: Conditions for the Optimality of the Simple Bayesian Classifier. In *In Machine Learning: Proceedings of the Thirteenth International Conference on Machine Learning. Morgan Kaufmann*, 1996.

[7] W. Du and Z. Zhan. Building decision tree classifier on private data. In Chris Clifton and Vladimir Estivill-Castro, editors, *IEEE ICDM Workshop on Privacy, Security and Data Mining*, volume 14 of *Conferences in Research and Practice in Information Technology*, pages 1–8, Maebashi City, Japan, 2002. ACS.

[8] L. Hansen and P. Salamon. Neural Network Ensembles. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 993–1001, 1990.

[9] D. Hershberger and H. Kargupta. Distributed multivariate regression using wavelet-based collective data mining. *Journal of Parallel and Distributed Computing*, 61(3):372–400, 2001.

[10] H. Kargupta. Gene expression and fast construction of distributed evolutionary representation. *Evolutionary Computation*, 9, 8 2000.

[11] H. Kargupta, W. Huang, K. Sivakumar, and E. Johnson. Distributed clustering using collective principal component analysis. Under consideration for publication in Knowledge and Information Systems.

[12] H. Kargupta, B. Park, D. Hershberger, and E. Johnson. Collective data mining: A new perspective toward distributed data mining. In *Advances in Distributed Data Mining, Eds: Hillol Kargupta and Philip Chan, AAAI/MIT Press*. 1999.

[13] S. C. Odewahn, E. B. Stockwell, R. L. Pennington, R. M. Humphreys, and W. A. Zumach. Automated star/galaxy discrimination with neural networks. *The Astronomical Journal*, 103:318–331, January 1992.

[14] J. Vaidya and C. Clifton. Privacy-preserving k-means clustering over vertically partitioned data. In *Conference on Knowledge Discovery in Data, Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 206–215.

[15] J. Vaidya and C. Clifton. Privacy preserving association rule mining in vertically partitioned data. In *Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2002.

[16] D. Wolpert. Stacked generalization. Technical Report LA-UR-90-3460, Complex Systems Group, Theoretical Divison, and Center for Non-linear Studies, Los Alamos, NM, 1990.