Report on Preliminary Experiments with Data Grid Models in the Agnostic Learning vs. Prior Knowledge Challenge

Marc Boullé

Abstract— **This paper introduces a new method[1] to automatically, rapidly and reliably evaluate the class conditional information of any subset of variables in supervised learning. It is based on a partitioning of each input variable, in intervals in the numerical case and in groups of values in the categorical case. The cross-product of the univariate partitions forms a multivariate partition of the input representation space into a set of cells. This multivariate partition, called data grid, allows to evaluate the correlation between the input variables and the output variable. The best data grid is searched owing to a Bayesian model selection approach and to combinatorial algorithms.**

Three classification techniques exploiting data grids differently are presented and evaluated in the Agnostic Learning vs. Prior Knowledge Challenge. These preliminary experiments demonstrate the interest of using data grid in machine learning tasks.

I. Introduction

Univariate partitioning methods have been studied extensively in the past, mainly in the context of decision trees [1], [2], [3], [4]. Supervised discretization methods split the numerical domain into a set of intervals and supervised value grouping methods partition the input values into groups. Fine grained partitions allow an accurate discrimination of the output values, whereas coarse grain partitions tend to be more reliable. When the size of the partition is a free parameter, the trade-off between information and reliability is an issue. In the MODL approach, supervised discretization [5] (or value grouping [6]) is considered as a non-parametric model of dependence between the input and output variables. The best partition is found using a Bayesian model selection approach.

In this paper, we describe an extension of the MODL approach to the bivariate case for pairs of numerical input variables [7], and introduce its generalization to any subset of variables of any types, numerical, categorical or mixed types. Each input variable is partitioned, in intervals in the numerical case and in groups of values in the categorical case. This joint partitioning defines a distribution of the instances in a multi-dimensional input data grid. The correlation between the cells of this data grid and the output values allows to quantify the joint predictive information. The tradeoff between information and reliability is established using a Bayesian model selection approach.

Sophisticated algorithms are necessary to explore the search space of data grid. They have to strike a balance between the quality of the optimization and the computation time. Several optimization heuristics, including greedy search, meta-heuristic and post-optimization are introduced to efficiently search the best possible data grid.

The paper is organized as follows. Section II summarizes the MODL method in the univariate discretization case. Section III describes the extension of the approach to the bivariate discretization case. Section IV presents an overview of the optimization algorithms in the bivariate case. Section V introduces the extension of data grids to the multivariate case, for supervised and unsupervised learning. Section VI describes three ways of building classifiers from data grids and section VII evaluates these classifiers on the Agnostic Learning vs. Prior Knowledge Challenge datasets. Finally, section VIII gives a summary and discusses future work.

II. The MODL discretization method

This section summarizes the MODL approach for supervised discretization, fully detailed in [5].

The objective of supervised discretization is to induce a list of intervals which splits the numerical domain of a continuous input variable, while keeping the information relative to the output variable. A compromise must be found between information quality (homogeneous intervals in regard to the output variable) and statistical quality (sufficient sample size in every interval to ensure generalization).

In the MODL approach, the discretization is turned into a model selection problem. First, a space of discretization models is defined. The parameters of a specific discretization are the number of intervals, the bounds of the intervals and the frequencies of the output values in each interval. Then, a prior distribution is proposed on this model space. This prior exploits the hierarchy of the parameters: the number of intervals is first chosen, then the bounds of the intervals and finally the frequencies of the output values. The choice is uniform at each stage of the hierarchy. Finally, we assume that the multinomial distributions of the output values in each interval are independent from each other. A Bayesian approach is applied to select the best discretization model, which is found by maximizing the probability $p(Model|Data)$ of the model given the data. Using the Bayes rule and since the probability $p(Data)$ is constant under varying the model, this is equivalent to maximizing $p(Model)p(Data|Model)$.

Let N be the number of instances, J the number of output values, I the number of intervals for the input domain. $N_{i.}$ denotes the number of instances in the interval i, and N_{ij} the number of instances of output value j in the interval i. In the context of supervised classification, the number of instances N and the number of classes J are supposed to be known.

M. Boullé is with France Telecom R&D, 2, avenue Pierre Marzin, 22307 Lannion Cedex, France (e-mail: marc.boulle@orange-ftgroup.com)

[1]French patent N° 06 01499

A discretization model M is then defined by the parameter set $\left\{I, \{N_{i.}\}_{1\leq i\leq I}, \{N_{ij}\}_{1\leq i\leq I, 1\leq j\leq J}\right\}$.

Owing to the definition of the model space and its prior distribution, the Bayes formula is applicable to exactly calculate the prior probabilities of the models and the probability of the data given a model. Taking the negative log of the probabilities, this provides the evaluation criterion given in formula 1.

$$c(M) = \log N + \log \binom{N+I-1}{I-1} + \sum_{i=1}^{I} \log \binom{N_{i.}+J-1}{J-1} + \sum_{i=1}^{I} \log \frac{N_{i.}!}{N_{i1}!N_{i2}!\ldots N_{iJ}!} \quad (1)$$

The first term of the criterion stands for the choice of the number of intervals and the second term for the choice of the bounds of the intervals. The third term corresponds to the choice of the output distribution in each interval and the last term represents the conditional likelihood of the data given the model. Therefore "complex" models with large numbers of intervals are penalized.

Once the optimality of the evaluation criterion is established, the problem is to design a search algorithm in order to find a discretization model that minimizes the criterion. In [5], a standard greedy bottom-up heuristic is used to find a good discretization. In order to further improve the quality of the solution, the MODL algorithm performs post-optimizations based on hill-climbing search in the neighborhood of a discretization. The neighbors of a discretization are defined with combinations of interval splits and interval merges. Overall, the time complexity of the algorithm is $O(JN \log N)$.

The MODL discretization method for classification provides the most probable discretization given the data sample. Extensive comparative experiments report high quality performance.

III. Extension to bivariate discretization

In this section, we describe the extension of the MODL approach to the supervised bivariate discretization of input variables [7]. We first introduce the approach using an illustrative example and then present the bivariate evaluation criterion in the case of two numerical variables.

A. Interest of the joint partitioning of two input variables

Figure 1 draws the multiple scatter plot (per class value) of the input variables V1 and V7 of the Wine dataset [8]. This diagram allows to visualize the conditional probability of the output values given the pair of input variables. The V1 variable taken alone cannot separate Class 1 from Class 3 for input values greater than 13. Similarly, the V7 variable is a mixture of Class 1 and Class 2 for input values greater than 2. Taken jointly, the two input variables allow a better separation of the class values.

Extending the univariate case, we partition the dataset on the cross-product of the input variables to quantify the relationship between the input and output variables. Each input variable is partitioned into a set of *parts* (intervals in the numerical case and groups of values in the categorical case). The cross-product of the univariate input partitions defines a *data grid*, which partitions the instances into a set of *data cells*. Each data cell is defined by a pair of parts. The connection between the input variables and the output variable is evaluated owing to the distribution of the output values in each cell of the data grid. It is noteworthy that the considered partitions can be factorized on the input variables.

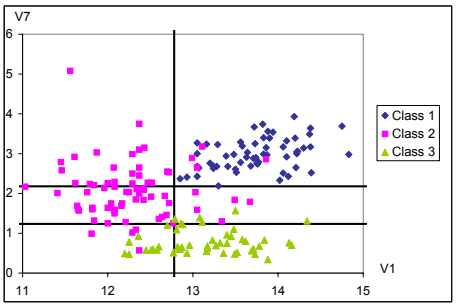

Fig. 1. Multiple scatterplot (per class value) of the input variables V1 and V7 of the Wine dataset. The optimal MODL supervised bivariate partition of the input variables is drawn on the multiple scatterplot

For instance in Figure 1, the V1 variable is discretized into 2 intervals (one bound 12.78) and the V7 variable into 3 intervals (two bounds 1.235 and 2.18). The instances of the dataset are distributed in the resulting bidimensional data grid. In each cell of the grid, the distribution of the output values can be estimated by counting. For example, the cell defined by the intervals $]12.78, +\infty[$ on V1 and $]2.18, +\infty[$ on V7 contains 63 instances. These 63 instances are distributed on 59 instances for Class 1 and 4 instances for Class 3.

Coarse grain data grids tend to be reliable, whereas fine grain data grids allow a better separation of the output values. In our example, the MODL optimal data grid is drawn on the multiple scatter plot on Figure 1.

B. Evaluation criterion for pairs of numerical variables

We extend the MODL approach to find the best tradeoff between information and reliability. We introduce in Definition 1 a family of bivariate partitioning models and select the best model owing to a Bayesian model selection approach.

Definition 1: *A data grid model is a bivariate partitioning model defined by a partition of each input variable in a set of intervals and by a multinomial distribution of the output values in each cell of the data grid resulting from the cross-product of the univariate partitions.*

Notation.

- Y: output variable,
- X_1, X_2: input variables,
- N: number of instances,
- J: number of output values,
- I_1, I_2: number of intervals for each input variable,

- $N_{i_1..}$: number of instances in the interval i_1 of X_1,
- $N_{.i_2.}$: number of instances in the interval i_2 of X_2,
- $N_{i_1 i_2.}$: number of instances in the input data cell (i_1, i_2),
- $N_{i_1 i_2 j}$: number of instances of output value j in the input data cell (i_1, i_2).

A data grid model describes the distribution of the output values given the input values. It is completely defined by the numbers of intervals I_1 and I_2, the bounds of the intervals $\{N_{i_1..}\}$ and $\{N_{.i_2.}\}$ and the distribution of the output values $\{N_{i_1 i_2 j}\}$ in each cell (i_1, i_2) of the data grid . It is noteworthy that the numbers of instances per cell $\{N_{i_1 i_2.}\}$ do not belong to the parameters of the data grid models: they are derived from the definition of the two univariate partitions and from the dataset.

Any input information is used to define the family of the model. The bounds of the univariate partition come from the input values and the frequencies of the input data cells come from the dataset. In that sense, the data grid models are data dependent. What is described in the model is the connection between the input variables and the output variable.
We now introduce in Definition 2 a prior distribution on the parameters of the data grid models. This prior exploits the hierarchy of the parameters and is uniform at each stage of this hierarchy.

Definition 2: *The hierarchical prior of the data grid models is defined as follows:*
- *the numbers of input intervals are independent from each other, and uniformly distributed between* 1 *and* N*,*
- *for each input variable and for a given number of intervals, every partition in intervals is equiprobable,*
- *for each cell of the data grid, every distribution of the output values is equiprobable,*
- *the distributions of the output values in each cell are independent from each other.*

We apply the Bayesian model selection approach and obtain the evaluation criterion of a data grid model M in formula 2.

$$\begin{aligned} c(M) &= \log N + \log \binom{N + I_1 - 1}{I_1 - 1} \\ &+ \log N + \log \binom{N + I_2 - 1}{I_2 - 1} \\ &+ \sum_{i_1=1}^{I_1} \sum_{i_2=1}^{I_2} \log \binom{N_{i_1 i_2.} + J - 1}{J - 1} \\ &+ \sum_{i_1=1}^{I_1} \sum_{i_2=1}^{I_2} \log \frac{N_{i_1 i_2.}!}{N_{i_1 i_2 1}! N_{i_1 i_2 2}! \dots N_{i_1 i_2 J}!} \end{aligned} \tag{2}$$

As in the case of univariate discretization (formula 1), the two first terms correspond to the prior probability of the parameters (number of intervals and choice of the bounds) of the discretization of the input variable X_1. Similarly, the two following terms correspond to the prior probability of the discretization of the input variable X_2. The binomial term in the first double sum represents the choice of the multinomial distribution of the output values in each cell. The multinomial term in the last double sum represents the conditional likelihood of the output values given the data grid model.

IV. Bivariate Optimization Algorithms

The space of data grid models is so large that straightforward algorithms almost surely fail to obtain good solutions within a practicable computational time. Given that the MODL criterion is optimal w.r.t the prior asumptions, the design of sophisticated optimization algorithms is both necessary and meaningful. In this section, we give an overview of the data grid optimization algorithms in the case of supervised bivariate discretization. They finely exploit the sparseness of the data grids and the additivity of the MODL criterion, and allow a deep search in the space of data grid models with $\mathrm{O}(N)$ memory complexity and a $\mathrm{O}(N \log N)$ time complexity.

The optimization of a data grid is a combinatorial problem. For each input variable X_1 and X_2, there are 2^N possible univariate discretizations, which represents $\left(2^N\right)^2$ possible bivariate discretizations. An exhaustive search through the whole space of models is unrealistic. We exploit a greedy bottom up merge heuristic (GBUM) to optimize the data grids. The method starts with the maximum data grid M_{Max}, which corresponds to the finest possible univariate discretizations, with N single value intervals. It evaluates all the merges between adjacent intervals, and performs the best merge if the evaluation criterion decreases after the merge. The process is reiterated until no further merge decreases the criterion.

Each evaluation of a data grid requires $\mathrm{O}(N^2)$ time, since the initial data grid model M_{Max} contains N^2 cells. Each step of the algorithm relies on $\mathrm{O}(N)$ evaluations of interval merges, and there are at most $\mathrm{O}(N)$ steps, since the data grid becomes equal to the null model $M_\emptyset$ once all the possible merges have been performed. Overall, the time complexity of the algorithm is $\mathrm{O}(N^4)$ using a straightforward implementation of the algorithm. However, the method can be optimized in $\mathrm{O}(N \log N)$ time. The optimized algorithm mainly exploits the sparseness of the data and the additivity of the evaluation criterion. Although a data grid may contain $\mathrm{O}(N^2)$ cells, at most N cells are non empty. Thus, each evaluation of a data grid can be performed in $\mathrm{O}(N)$ owing to a specific algorithmic data structure. The additivity of the evaluation criterion means that the criterion can be decomposed on the hierarchy of the components of the data grid: variables, parts and cells. Using this additivity property, all the merges between adjacent parts can be evaluated in $\mathrm{O}(N)$ time. Furthermore, when the best merge is performed, the only impacted merges that need to be reevaluated for the next optimization step are the merges that share instances with the best merge. Since the data grid is sparse, the number of reevaluations of data grids is small on average. Sophisticated algorithmic data structures and algorithms are necessary to exploit these optimization principles and guarantee a time

complexity of $O(N \log N)$.

The optimized version of the greedy heuristic is time efficient, but it may fall into a local optimum. First, the greedy heuristic may stop too soon and produce too many intervals for each input variable. Second, the boundaries of the intervals may be sub-optimal since the merge decisions of the greedy heuristic are never rejected. The post-optimization algorithms described in [5] in the case of univariate discretization are applied alternatively to each input variable, for a frozen partition of the other input variable.

While post-optimizations may help to refine a good solution, the main heuristic may be unable to obtain such an initial good solution. This problem is tackled using the Variable Neighborhood Search (VNS) meta-heuristic [9], which mainly benefits from multiple runs of the algorithms with different random initial solutions.

V. Data Grids for any Subset of Variables

In this section, we present an overview of the extension of data grid models to any subset of variables, in the supervised case then in the unsupervised case.

A. Supervised data grid models

The MODL approach has been studied in the case of univariate supervised partitioning for numerical variables [5] and categorical variables [6]. The extension to the multivariate case applies the same principles as those described in section III. Each input variable is partitioned, in intervals in the numerical case and in groups of values in the categorical case. Taking the cross-product of the univariate partitions, we obtain a data grid of input cells, the content of which allows to characterize the distribution of the output values.

The space of multivariate data grid models is very large and prone to overfitting. A Bayesian model selection approach is employed to find the best data grid model given the data. The parameters of the data grid models are precisely defined, and a prior is proposed that exploits the hierarchy of the parameters, is uniform at each stage of the hierarchy, and assumes the independence of the output distribution within each cell. We then obtain an analytic formula that evaluates the posterior probability of each data grid model, and exploit extensions of the algorithms summarized in section IV to efficiently search the space of data grid models.

B. Unsupervised data grid models

Data grid models are generalized to unsupervised learning. Each variable is partitioned into a set of intervals (or groups of values), and the cross-product of the univariate partitions forms a data grid of cells. The instances are distributed in the cells of the grid according to a multinomial distribution. Such models describe the joint distribution between the variables. For example, in case of independent variables, the distribution of the instances in the cells is homogeneous (w.r.t. the variable ranks, not their values), where as it is unbalanced in the case of correlated variables.

Applying the MODL approach, a prior is defined on the model parameters, and the MAP Data Grid is optimized using the same search algorithms as in the supervised case.

VI. Building classifiers from data grid models

In this section, we describe three ways of building classifiers from data grid models. This is preliminary work and we expect that the evaluation of these approaches on the Agnostic Learning vs. Prior Knowledge Challenge [10] will guide future research.

A. Data grid

In this evaluation of data grid models, we consider one single data grid, the MAP one. We build a classifier from a data grid model by first retrieving the cell related to a test instance, and predicting the output conditional probabilities of the retrieved cell. For empty cells, the conditional probability used for the prediction is that of the entire grid.

Data grid models can be considered as a feature selection methods, since the input variables whose partition reduces to a single part can be ignored. The purpose of this experiment is to focus on understandable models and evaluate the balance between the number of selected variables and the predictive performance.

B. Data grid ensemble

In this evaluation, we focus on the predictive performance rather than on understandability, by the mean of averaging the prediction of a large number of classifiers. This principle was successfully exploited in Bagging [11] using multiple classifiers trained from re-sampled datasets. This was generalized in Random Forests [12], where the subsets of variables are randomized as well. In these approaches, the averaged classifier uses a voting rule to classify new instances. Unlike this approach where each classifier has the same weight, the Bayesian Model Averaging (BMA) approach [13] weights the classifier according to their posterior probability. The BMA approach has stronger theoretical foundations, but it requires both to be able to evaluate the posterior probability of classifiers and to sample their posterior distribution.

In the case of data grid models, the posterior probability of each model is given by an analytic criterion. Concerning the problem of sampling the posterior distribution of data grid models, we have to strike a balance between the quality of the sampling and the computation time. We adopt a pragmatic choice by just collecting all the data grids evaluated during training, using the optimization algorithm introduced in section IV. We keep all the local optima encountered in the VNS meta-heuristic and eliminate the duplicates. An inspection of the collected data grids reveals that their posterior distribution is so sharply peaked that averaging them according to the BMA approach almost reduces to the MAP model. In this situation, averaging is useless. The same problem has been noticed in [14] in the case of averaging Selective Naive Bayes models. To find a trade-off between equal weights as in bagging and extremely unbalanced weights as in the BMA approach, we exploit a logarithmic smoothing of the posterior distribution called compression-based model averaging (CMA), like that introduced in [14].

To summarize, we collect the data grid models encountered during the data grid optimization algorithm and weight

them according to a logarithmic smoothing of their posterior probability to build a Data Grid Ensemble classifier.

C. Coclustering of instances and variables

We first introduce the application of unsupervised data grids to the coclustering problem, then describe how to build a classifier on the basis of coclustering.

1) Coclustering: A coclustering [15] is the simultaneous clustering of the rows and columns of a matrix. In case of binary sparse datasets, coclustering is an appealing data preparation technique to identify correlation between clusters of instances and clusters of variables. Let us notice that continuous variables can be transformed into binary variables according to whether their value is null or non null.

Let us consider a sparse dataset with N instances, K variables and V non-null values. A sparse dataset can be represented in tabular format, with two columns and V raws. This corresponds to a new *dataset* with two *variables* named "Instance ID" and "Variable ID" where each *instance* is a couple of values (Instance ID, Variable ID). Bivariate unsupervised data grid models are applied to form groups of instances IDs and groups of variable IDS, so as to maximize the correlation between instances and variables. We expect to find "natural" patterns both in the space of instances and in the space of variables. It is noteworthy that the clusters retrieved by data grid models are non-overlapping, since they form a partition of the whole dataset.

2) Application to supervised learning: We apply a semi-supervised learning approach [16] to exploit all the data from the train, validation and test datasets. In a first step, all the instances are processed without any output label to identify the "natural" clusters of instances owing to the data grid coclustering technique. In a second step, the available labeled instances are used to describe the output distribution in each cluster of instances. The label of a test instance is then predicted according to the output distribution of its cluster.

Preprocessing the data with semi-supervised coclustering makes sense under the assumption that the "natural" clusters are correlated with the output values (predefined clusters). We expect that this assumption is true for some datasets, especially in the pattern recognition domain.

VII. Evaluation of Data Grid Models

In the section, we first summarize the evaluation protocol of the challenge, then describe which data grid models are used to build classifiers, and finally report the results.

A. The Agnostic Learning vs. Prior Knowledge Challenge

The purpose of the challenge [10], [17] is to assess the real added value of prior domain knowledge in supervised learning tasks. Five datasets coming from different domains are selected to evaluate the performance of agnostic classifiers vs. prior knowledge classifiers. These datasets come into two formats, as shown in Table I. In the agnostic format, all the input variables are numerical. In the prior knowledge format, the input variables are both categorical and numerical for three datasets and have a special format in the two other datasets: chemical structure or text.

TABLE I
Challenge Datasets

Name	Domain	Num. ex. (train)	Prior features	Agnostic features
Ada	Marketing	4147	14	48
Gina	Digit reco.	3153	784	970
Hiva	Drug discovery	3845	Struct.	1617
Nova	Text classif.	1754	Text	16969
Sylva	Ecology	13086	108	216

The size of each train dataset is reported in Table I. The validation datasets are ten times smaller than the train datasets, which is too small for efficient model selection. The test datasets are ten times larger than the train datasets, which is large enough for reliable performance evaluation. The predictive performance is evaluated using the Balanced Error Rate (BER) criterion.

B. Data grid models used for classification

We use all the datasets in their agnostic format and only three of them in their prior format (we have neither domain knowledge nor time to exploit the chemical structure in the Hiva dataset or the native text format in the Nova dataset).

We exploit directly the native format of the datasets, numerical only in the agnostic case, either numerical or categorical in the prior case. We apply only one transformation to the representation space, in the case of the Sylva dataset in its prior format. We replace each subset of 40 binary "SoilType" variables by one single categorical variable with 40 values. The resulting dataset has only 30 variables instead of 108.

We apply the three classification techniques based on multivariate data grid models introduced in section VI : data grid (DG), data grid ensemble (DGE) and semi-supervised coclustering (DGCC). The classifiers are trained using the train and validation labeled instances. The DGCC methods exploits all the available unlabeled data in its preprocessing. It is applied on the three sparse datasets: Gina in its Prior Knowledge format (PK), Hiva and Nova in their Agnostic Learning format (AL).

All these techniques are able to predict the output conditional probabilities for each test instance. When the evaluation criterion is the classification accuracy, predicting the class with the highest conditional probability is optimal. This is not the case for the BER criterion used in the challenge. We post-process each trained classifier by optimizing the probability threshold in order to maximize the BER. This optimization is performed directly on the train dataset.

C. Evaluation results

We report in Table II the BER obtained by our methods on the challenge datasets. The BER is evaluated using a stratified ten-cross validation on the train+validation datasets. The missing results correspond to the datasets with complex

TABLE II
CHALLENGE BER RESULTS

Name	Prior		Mixed	Agnostic		Rank	
	DGE	DG	DGCC	DGE	DG	PK	AL
Ada	0.192	0.213		0.192	0.225	1	2
Gina	0.140	0.184	0.052 (PK)	0.147	0.182	4	10
Hiva			0.320 (AL)	0.310	0.340	-	8
Nova			0.075 (AL)	0.135	0.243	-	3
Sylva	0.008	0.009		0.022	0.021	4	6

prior format (Hiva and Nova), and to the datasets with dense representation in the case of the coclustering method.

We also report the rank of our best submission in the Agnostic Learning (AL) track and Prior Knowledge (PK) track, available on the challenge web site [10]. The final results will be published after August 1^{st} 2007.

The Data Grid classifiers obtain good results on the Ada and Sylva datasets, especially on the prior track. Let us now focus on understandability and inspect the number of selected variables in each trained Data Grid model. In the agnostic track, the MAP data grid exploits only 5 variables for Ada, 5 for Gina, 4 for Hiva, 8 for Nova and 8 for Sylva. In the prior track, the MAP data grid exploits 6 variables for Ada, 7 for Gina and 3 for Sylva. These numbers of variables are remarkably small w.r.t the BER performance of the models.

The Data Grid Ensemble classifiers confirm the benefits of compression-based model averaging. They obtain a very significant improvement of the BER criterion compared to the Data Grid classifiers. This focus on predictive performance is realized at the expense of understandability, since each trained Data Grid Ensemble averages several hundreds of elementary Data Grid models.

The Data Grid semi-supervized coclustering classifier obtain significantly better results on the Gina and Nova datasets than the Data Grid Ensemble classifiers. The assumption that the "natural" patterns are correlated with the classes looks true in the problems of digit recognition and text classification. For the Hiva dataset, the BER result is not much competitive. In this dataset, the "natural" patterns in the representation space of the challenge does not seem highly correlated with the classes.

Although its predictive performance is still far from the best results, the coclustering technique carries valuable insights on datasets. For example, in the case of text classification (Nova), about 1000 clusters of words (themes) and 300 clusters of texts ("natural" classes) are identified. In the case of digit recognition (Gina), about 100 clusters of pixels (regions) and 500 clusters of images ("natural" patterns) are identified. Further investigation is necessary to inspect these clusters and evaluate their impact on data understanding, data reduction or modeling.

Apart from their use of the type of each input variable (categorical or numerical), data grid models are agnostic learners. It is noteworthy that in this challenge, our BER results are always better in the prior track than in the agnostic track, which tends to confirm the assumption that prior knowledge and specialized representation space makes the learning task easier.

VIII. CONCLUSION

The data grid models introduced in this paper are based on a partitioning model of each input variables, in intervals for numerical variables and in groups of values for categorical variables. The cross-product of the univariate partitions, called a data grid, allows to quantify the conditional information relative to the output variable. We have detailed this technique in the case of bivariate numerical variables and presented an overview for the extension of data grids to the multivariate case, both for supervised and unsupervised learning.

We have introduced three ways of building classifiers from data grids and experimented them on the Agnostic Learning vs. Prior Knowledge challenge. This preliminary evaluation looks promising and provides usefull informations to drive our future research.

REFERENCES

[1] G. Kass, "An exploratory technique for investigating large quantities of categorical data," *Applied Statistics*, vol. 29, no. 2, pp. 119–127, 1980.

[2] L. Breiman, J. Friedman, R. Olshen, and C. Stone, *Classification and Regression Trees*. California: Wadsworth International, 1984.

[3] J. Quinlan, *C4.5: Programs for Machine Learning*. Morgan Kaufmann, 1993.

[4] D. Zighed and R. Rakotomalala, *Graphes d'induction*. France: Hermes, 2000.

[5] M. Boullé, "MODL: a Bayes optimal discretization method for continuous attributes," *Machine Learning*, vol. 65, no. 1, pp. 131–165, 2006.

[6] M. Boullé, "A Bayes optimal approach for partitioning the values of categorical attributes," *Journal of Machine Learning Research*, vol. 6, pp. 1431–1452, 2005.

[7] M. Boullé, "Une méthode optimale d'évaluation bivariée pour la classification supervisée," in *Extraction et gestion des connaissances (EGC'2007)*, 2007, pp. 461–472.

[8] C. Blake and C. Merz, "UCI repository of machine learning databases," 1996, http://www.ics.uci.edu/mlearn/MLRepository.html.

[9] P. Hansen and N. Mladenovic, "Variable neighborhood search: principles and applications," *European Journal of Operational Research*, vol. 130, pp. 449–467, 2001.

[10] I. Guyon, "Agnostic learning vs. prior knowledge challenge," 2007, http://clopinet.com/isabelle/Projects/agnostic/.

[11] L. Breiman, "Bagging predictors," *Machine Learning*, vol. 24, no. 2, pp. 123–140, 1996.

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

[13] J. Hoeting, D. Madigan, A. Raftery, and C. Volinsky, "Bayesian model averaging: A tutorial," *Statistical Science*, vol. 14, no. 4, pp. 382–417, 1999.

[14] M. Boullé, "Regularization and averaging of the selective naïve Bayes classifier," in *International Joint Conference on Neural Networks*, 2006, pp. 2989–2997.

[15] J. Hartigan, "Direct clustering of a data matrix," *Journal of the American Statistical Association*, vol. 67, no. 337, pp. 123–129, 1972.

[16] O. Chapelle, B. Schölkopf, and A. Zien, *Semi-Supervised Learning*. MIT Press, Cambridge, MA, 2006, (in press).

[17] I. Guyon, A. Saffari, G. Dror, and G. Cawley, "Agnostic learning vs. prior knowledge challenge," in *International Joint Conference on Neural Networks*, 2007, in press.