Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 46 (2015) 476 – 482

Procedia
Computer Science

International Conference on Information and Communication Technologies (ICICT 2014)

Bayesian classifier structure-learning using several general algorithms

Heni Bouhamed[a,]*, Afif Masmoudi[b], Ahmed Rebai[a]

[a]Bioinformatics Unit, Biotechnology Center of Sfax BP 1177 , Sfax 3018, Tunisia
[b]Faculty of Sciences of Sfax, Soukra Road km 3.5 B.P 802, Sfax 3000, Tunisia

Abstract

The use of Bayesian Networks (BNs) as classifiers in different fields of application has recently witnessed a noticeable growth. Yet, the Naïve Bayes application, and even the augmented Naïve Bayes, to classifier-structure learning, has been vulnerable to certain limits, which explains the practitioners resort to other more sophisticated types of algorithms. Consequently, the use of such algorithms has paved the way for raising the problem of super-exponential increase in computational complexity of the Bayesian classifier learning structure, with the increasing number of descriptive variables. In this context, the present work's major objective lies in setting up a further solution whereby a remedy can be conceived for the intricate algorithmic complexity imposed during the learning of Bayesian classifiers structure with the use of sophisticated algorithms.

Keywords: Bayesian classifier; Clustering; Algorithmic complexity.

1. Introduction

It is worth noting that efficient classifiers can be reached through the use of Bayesian networks[1, 2, 3]. In fact, a Bayesian classifier relative to a problem with p variables is characterized by the distinction of having p+1 nodes. Indeed, all Bayesian classifiers model the fact of belonging to a certain class by means of a discrete node dubbed

* Corresponding author. Tel.: +216 25290881; fax: +216 74875818.
E-mail address: heni_bouhamed@yahoo.fr

doi:10.1016/j.procs.2015.02.046

"class node". This node is discrete and multinomial having k modality. Regarding the other p variables, which we call descriptive variables, they are denoted X_i (i from 1 to p). The Bayesian classifier with the simplest structure is the naïve Bayesian network (RBN)[7], also called Naïve Bayes classifier. Nevertheless, no correlations between the attributes are taken into account with respect to the Naïve Bayes, where all features contribute to the classification in the same way. The classification node takes advantage of the information provided by each attribute independently of the information provided by other features-still; this may not be optimal for the classification task. Hence, various proposals have been suggested in a bid to enrich the Naive Bayesian Network structure to make it account for correlations between different attributes. In[2], for instance, the authors have proposed a Tree-Augmented Naïve Bayes (TAN) approach to enrich the Naïve Bayes structure. According to this approach, a tree structure is applied for the classification to be achieved[15, 5]. The tree structure has the advantage of having a low degree of complexity, along with the ability to avoid over fitting problems. However, it restricts the number of parents, other than the classification node, to exactly one single parent for each node, which turns out to be a strong constraint. So, the resulting structure appears to neglect the case where a variable is correlated with several other variables. Besides, it outlooks the case where a variable is conditionally independent of all other variables within the classification node. In which case, the node representing that variable only needs the class node as a parent. The addition of another parent only adds unnecessary complexity and increases the number of network parameters. Consequently, other authors[4, 5, 6, 18, 19, 20, 21, 22, 23] have proposed the use of more general and sophisticated methods, without specific restriction, to overcome these shortcomings, among which are: the K2 algorithms[8], the PC algorithm (causality search algorithm)[9], the BN-PC-B algorithm[25], the Greedy Search[24] and the Greedy Equivalent Search (GES)[26]. Although these algorithms have actually managed to attain performant Bayesian network, their application has resulted in the frequently and commonly encountered problem of structure-learning computational complexity owing to the increase in the number of descriptive variables.

Hence, a new approach has been proposed through this research work based on a structure learning upstream clustering, which can be jointly used with the cited above algorithms pertinent to the structure learning of Bayesian Classifiers. The envisaged aim behind this framework proposal is to reduce the computational complexity and, consequently, the execution time without engendering a loss of information, in comparison to the use of only the classic algorithms.

The remainder of this paper has been arranged as follows. In the upcoming section, we are going to put forward a new approach which we shall test upon an Asia, a Car-diagnosis, a Lymphography diagnosis and a Mushroom classification databases. Finally, we will close up our work by concluding and paving the way for certain potential perspectives for future researches.

2. A new clustering-based approach (BCCA): procedure and applied methodologies

The idea lying behind our conceived procedure lies in the rapid super-exponential surge of algorithmic complexity of learning the Bayesian classifier structure from data with respect to the rise in the number of variables. To remedy this problem, our idea consists in subdividing the variables into subsets (or clusters), by learning the structure of each cluster separately, while looking for a convenient procedure whereby the different structures could be assembled into a final structure. In this regard, it has been noticed that in the case of a Bayesian classifier learning structure, there exists one single central variable of a global interest called "class" variable. In this respect, we reckon to execute the processing of each cluster learning structure with the class variable, then, proceed by assembling the different various structures around this class variable as a next step.

2.1. The variables' clustering

Clustering is the most frequently used and widespread technique among the data-analysis and data-mining descriptive techniques. It is often used when we have a huge amount of data, within which we intend to distinguish some homogeneous subsets suitable for processing and for differential analyses[10].

Actually, there exist two major well-known clustering families of algorithms in the literature, namely: the partition methods and the ascending hierarchical-clustering. The advantage of the ascending-hierarchical methods, as compared to the partitioning ones, lies in the fact that they able to choose, appropriately, the optimum number of clusters. Nevertheless, the partitioning criterion is not global; it exclusively depends on the already-obtained clusters, since two variables placed in different clusters could by no means be compared any more. Contrary to the hierarchical methods, the partitioning algorithms might perpetually improve the clusters quality[10], in addition to the fact that their algorithmic complexities are linear. Regarding our present work, we have chosen to use the K-means algorithm, as it is the most popular, added to fact that its algorithmic complexity is linear (O(n))[11]. Besides, we reckon to apply a hierarchical clustering algorithm along with the bootstrap technique to obtain the optimal number of clusters suitable for the K-means algorithm. To note, the databases that will be applied to test our approach, in the experimentation section, consist of some categorical variables, and regarding the performance of clustering we will use the package ClustOfVar with R language[12]. In particular, we will use the variant K-means[13] along with the linkage-likelihood analysis[14] (hierarchical clustering algorithm) for categorical variables.

2.2. Structure learning

A structure learning has been performed for each cluster of variables including the class variable. The ultimate structure would be the assembling of the *n* structures obtained from each cluster around the class variable.

We will perform our tests, firstly, via the K2 algorithm with, as input, the order obtained by applying the algorithm MWST (for the MWST algorithm, the initial node will be the class variable)[16], secondly, via the BN-PC-B algorithm, thirdly, via the Greedy Search, fourthly, via the Greedy Equivalent Search and finally via the Greedy Search with, as input, the obtained tree by applying the algorithm MWST (for the MWST algorithm, the initial node will be the class variable) . In our study case, we would rather try to prove that the joint use of our approach together with the above cited algorithms can be beneficial in reducing the computational complexity without losing information.

Note that in our work, we will use the BNT toolbox[17] running on the Matlab software (2010 version) to apply all the tested algorithms to structure learning. We will also apply the BNT toolbox for parameters learning and inference.

3. Experimentation procedures

3.1. Data-bases

We first test our approach, on a famous Asia database (it comes from the diagnosis of dyspnea introduced by Lauritzen and Spiegelhalter[27]). It has 8 variables, among which we will assume that the variable called “Lung Cancer” is the class variable. Among the 5000 instances of data, 200 have been left aside for the references’ testing phase. We second test our approach, on a car diagnosis database (Car Diagnosis 2). It has 18 variables, among which is a status variable called “Car starts (ST)”, the class variable. The parameters’ generating file of this database is available on-line at: http://www.norsys.com/downloads/netlib/. According to these parameters, we have been able to generate 10.000 examples, among which 200 have been left aside for the references’ testing phase. We third apply our approach to a Lymphography diagnosis database (Lymphography). It is made up of 19 variables, among which is a status variable called “Diagnosis”, the class variable. This lymphography domain has been obtained from the University Medical Centre, Institute of Oncology, Ljubljana, Yugoslavia (available on request on-line at: http://archive.ics.uci.edu/ml/datasets/Lymphography). Among the 143 instances of data, only 36 have been left aside for the references’ testing phase (accordingly to the limited number of examples). Ultimately, we apply our approach to a Mushroom classification database. It is made up of 21 variables (we have eliminated two variables, one is uniform and the other has a missing data), among which is a status variable called “classes”, the class variable. This Mushroom domain has been available on-line at: https://archive.ics.uci.edu/ml/datasets/Mushroom. Among the 8124 instances of data, 200 have been left aside for the references’ testing phase.

3.2. Bayesian classifier learning structure using K2+MWST, BN-PC-B, Greedy Search, GES and Greedy Search+MWST compared to our new approach jointly used with the cited above algorithms

Results are presented in tables form (see Table 1, 2, 3, 4, 5), each table present the difference, in terms of CPU execution time, between using an algorithm tested with and without our approach. (the machine used for running the entire experimentation procedure is a Personal Computer with core 2 Duo processor and 3 gigabytes of RAM memory).

Table 1. Summary of execution times using K2+MWST.

	Asia	Car Diagnosis 2	Lymphography	Mushroom
K2+MWST	0,156 Seconds CPU	3,961 Seconds CPU	1,731 Seconds CPU	51,215 Seconds CPU
Our approach using K2+MWST	0,048 Seconds CPU	0,776 Seconds CPU	0,253 Seconds CPU	0,529 Seconds CPU

Table 2. Summary of execution times using BN-PC-B.

	Asia	Car Diagnosis 2	Lymphography	Mushroom
BN-PC-B	2,772 Seconds CPU	61,433 Seconds CPU	12,402 Seconds CPU	1526,659 Seconds CPU
Our approach using BN-PC-B	1,981 Seconds CPU	9,81 Seconds CPU	1,245 Seconds CPU	193,168 Seconds CPU

Table 3. Summary of execution times using Greedy Search.

	Asia	Car Diagnosis 2	Lymphography	Mushroom
Greedy Search	13,078 Seconds CPU	673,836 Seconds CPU	189,603 Seconds CPU	1483,25 Seconds CPU
Our approach using Greedy Search	0,542 Seconds CPU	64,053 Seconds CPU	9,861 Seconds CPU	12,878 Seconds CPU

Table 4. Summary of execution times using GES.

	Asia	Car Diagnosis 2	Lymphography	Mushroom
GES	9,396 Seconds CPU	319,256 Seconds CPU	115,284 Seconds CPU	961,129 Seconds CPU
Our approach using GES	0,35 Seconds CPU	33,5 Seconds CPU	8,718 Seconds CPU	11,677 Seconds CPU

Table 5. Summary of execution times using Greedy Search+MWST.

	Asia	Car Diagnosis 2	Lymphography	Mushroom
Greedy Search+MWST	3,365 Seconds CPU	357,367 Seconds CPU	210,460 Seconds CPU	782,157 Seconds CPU
Our approach using Greedy Search+MWST	0,525 Seconds CPU	35,074 Seconds CPU	10,027 Seconds CPU	8,256 Seconds CPU

3.3. Attained structures' relevant inferences and result comparisons

Our approach favors the preservation of data for the sake of the class variable, in the aim to have good classification results. We will learn the parameters of the two structures found for each of the databases and for each of the algorithms studied (the structure found after learning all the variables simultaneously and the one found after assembling the various structures of the clusters around the class variables). For the class variable (for each of the databases), we are going to calculate the probability of its first instance given the state of the networks other nodes in respect of the two obtained Bayesian classifiers structures (for each of the algorithms studied). Thus, a 200-example database will be used to experiment the class variables of the "Asia", "Car Diagnosis2" and "Mushroom Classification" databases. However, only a 36-example database will be used to experiment the class variables of the

"Lymphography" database (accordingly to the limited number of examples). Naturally, the experimentation examples were excluded during the learning of structures.

The four tested class variables are "Lung cancer (L)" of the "Asia" database, "Car starts (ST)" of the "Car Diagnosis 2" database, "Diagnosis" of the "Lymphography" database and "Class" of the "Mushroom Classification" database. The correct classification percentage of all experimentation are presented in Table 6. The presented results are for each of the databases and the studied algorithms.

Table 6. Correct classification percentage.

	Asia	Car Diagnosis 2	Lymphography	Mushroom
K2+MWST	99,5	81	94,44	100
Our approach using K2+MWST	99	81,5	94,44	100
BN-PC-B	99,5	81,5	94,44	100
Our approach using BN-PC-B	99	81,5	91,66	100
Greedy Search	99,5	81	91,66	100
Our approach using Greedy Search	99	81	91,66	99
GES	99,5	81	91,66	99
Our approach using GES	99	81	88,88	100
Greedy Search+MWST	99,5	82	88,88	100
Our approach using Greedy Search+MWST	99	82	88,88	100

3.4. Discussion

Based on the achieved experimental results, the gain in terms of execution time is certain. It is sometimes very large, reaching up to 96 times reduction in contribution to the execution time of learning all the variables simultaneously (see Table 1). We still have to answer this question, 'what impact does our approach have on the operating results of the eventual models builds?

The probability pairs pertaining to each class variables are very similar and in many cases they are even identical. In terms of the classification effectiveness, the results are very similar and there are even better results found with the use of our approach (see Table 6). Similarly, the class variable (for each of the databases) instance probabilities, found with the use of our approach (jointly with each of the studied algorithms), are in many cases the most accurate compared to the class variable original instance value.

Therefore, it can be deduced that the inference results, regarding both learning-structure approaches, are very similar. Thus, through our approach, we have managed to reduce considerably the algorithmic complexity of the Bayesian classifier structure learning without any significant loss of information, especially with regard to the class variable. The clustering constancy and trustworthiness play a determining role in the accuracy of the resulting structure. In fact, the more independent the obtained clusters are, the more the number of inter-cluster edges to be lost would shrink. Consequently, the more independent the clusters are, the more negligible the lost information would be.

Throughout the present study, a new approach has been proposed. It has been based on the implementation of class variable as a linking variable among the subsets' different structures. Through our proper experimentation procedure, we have proved that by implementing this undertaking, we can be immune to the information loss problem while achieving a considerable gain in terms of execution time. Our original solution has been improved because no criterion has been defined for the applicability of our approach on a certain database (possibility of having clusters sufficiently independent to avoid losing information). Secondly, the method applied for determining the optimal number of clusters is known to be greedy in computational complexity (in the order of O(n^3)). Therefore, a heuristic, less complicated yet effective, would be among our aims in future research. Inversely, however, with the help of our newly-devised concept, new large-scale horizons have been opened, paving the way for other more global solutions and taking advantage of the fact that the possible number of Direct Acyclic Graphs decreases incredibly by treating variables' subsets during the Bayesian classifier or the general BN structure learning from database.

4. Conclusion

Within the scope of the present work, we have set up a new well-defined approach for the Bayesian classifier structure learning from data-base, so useful that it can be jointly applied with the K2+MWST, BN-PC-B, Greedy Search, Greedy Equivalent Search and Greedy Search+MWST algorithms. As a first step, a specially-devised approach has been proposed to perform a Bayesian classifier. As a second step, through a specially-conducted experimentation administered over a "Asia", a "Car Diagnosis 2", a "Lymphography" and a "Mushroom Classification" databases, we have proved that loss in information turns out to be so negligible that it does not affect the extracted Bayesian classifier stemming results during the inference stage, while saving a great deal of execution time.

In a potential future research, we reckon to make a serious attempt to investigate other possible alternatives useful and fit to exploit the considerable reduction of algorithmic complexity during the BN structure learning by examining and treating variables' sub-sets, developing some structure-retrieving oriented heuristics encompassing the already achieved sub-structures, a framework that would be the closest possible to the discovered structure, while simultaneously treating the whole set of variables in their entirety.

References

1. Langley P, Sage S, Induction of Selective Bayesian Classifiers, *in Proceedings of the Tenth Conference on Uncertainty in Artificial Intelligence* 1994; p399-406.
2. Friedman N, Geiger D, Goldszmid M, Bayesian Network classifiers. *Machine Learning* 1997; p131-163.
3. Pernkopf F, Bayesian network classifiers versus selective k-NN classifier, *Pattern Recognition*, 2005; p1-10.
4. Stuart M, Yulan H, Kecheng L, Choosing the best Bayesian classifier : An empirical study. *IAENG International Journal of Computer Science* 2009; p1-10.
5. Madden MG, A New Bayesian Network Structure for Classification Tasks, *in Proceedings of 13th Irish Conference on Artificial Intelligence & Cognitive Science*, 2002; p203-208.
6. Lerner B, Malka R, Investigation of the K2 algorithm in learning Bayesian Network Classifiers, *Applied Artificial Intelligence*, 2011; p74-96.
7. Domingos P, Pazzani M, On the optimality of the simple Bayesian classifier under zero-one loss. *MachineLearning*, 1997; p103-130.
8. Cooper G, Hersovits E, A Bayesian method for the induction of probabilistic networks from data, *Machine learning*. 1992; p309-347.
9. Spirtes P, Glymour C., Scheines R, Causation, Prediction, and Search. *The MIT Press*, 2nd edition, 2000.
10. Tufféry S, Data mining et statistique décisionnelle: l'intelligence des données, *Editions TECHNIP*, 2010.
11. Jain AK, Data clustering: 50 years beyond K-means, *Pattern Recognition Letters*. 2010; p 651-666.
12. Chavent M, Kuentz V, Liquet B, Saracco J, ClustOfVar: an R package for the clustering of variables. *The R user conference, University of Warwick Coventry UK*. 2011; p44.
13. Chavent M, Kuentz V, Saracco J, A partitioning method for the clustering of categorical variables. *In classification as a tool for Research, Herman locarek-Junge, claus Weihs (Eds), Springer, in Proceedings of the IFCS* 2009.
14. Lerman IC, Likelihood linkage analysis (LLA) classification method : An example treated by hand, *Biochimie*, 1993; p379-397.
15. Chow C, Liu C, Approximating discrete probability distributions with dependence trees. *IEEE Transactions on Information Theory*. 1968; p462-467.
16. Francois O, Leray P, Evaluation d'algorithmes d'apprentissage de structure pour les réseaux bayésiens, *In Proceedings of 14ème Congrès Francophone Reconnaissance des Formes et Intelligence Artificielle*, 2004; p1453-1460.
17. Murphy K, The BayesNet Toolbox for Matlab, Computing Science and Statistics: *Proceedings of Interface*. 33 2001; http ://www.ai.mit.edu/~murphyk/Software/BNT/bnt.html.
18. Ezawa K, Singh M, Norton S, Learning goal oriented Bayesian networks for telecommunications risk management, *In Proceedings of the Thirteenth International Conference on Machine Learning* 1996; p139-147.
19. Porwal A, Carranza E, Hale M, Bayesian network classifiers for mineral potential mapping, *Computers & Geosciences* 2006; p1–16.
20. Malka R, Lerner B, Classification of fluorescence in situ hybridization images using belief networks, *Pattern Recognition Letters* 2004; p1777-1785.
21. Estevam R, Hruschka J, Ebecken N, Towards efficient variables ordering for Bayesian networks classifier, *Data & Knowledge Engineering* 2007; p258-269.
22. Carta JA, Velázquez S, Matías JM, Use of Bayesian networks classifiers for long-term mean wind turbine energy output estimation at a potential wind energy conversion site, *Energy Conversion and Management* 2011; p1137-1149.

23. Kelner R, Lerner B, Learning Bayesian network classifiers by risk minimization, *International Journal of Approximate Reasoning*, 2012; p248-272.
24. Chickering D, Geiger D, Heckerman D. Learning bayesian net-works : Search methods and experimental results. *In Proceedings of Fifth Conference on Artificial Intelligence and Statistics*, 1995; p112-128.
25. Cheng J, Greiner R, Kelly J, Bell D, Liu W, Learning Bayesian networks from data: An information-theory based approach. *Artificial Intelligence* 2002; p43-90.
26. Chickering DM, Optimal structure identification with greedy search. *Journal of Machine Learning Research*, 2002; p507-554.
27. Lauritzen S, Speigelhalter D. Local computations with probabilities on graphical structures and their application to expert systems. *Royal statistical Society* 1988, p157-224.