Fast Committee-Based Structure Learning

Ernest Mwebaze EMWEBAZE@CIT.MAK.AC.UG
Faculty of Computing & I.T.
Makerere University
Kampala, Uganda

John A. Quinn JQUINN@CIT.MAK.AC.UG
Faculty of Computing & I.T.
Makerere University
Kampala, Uganda

Editor: Isabelle Guyon, Dominik Janzing and Bernhard Schölkopf

Abstract

Current methods for causal structure learning tend to be computationally intensive or intractable for large datasets. Some recent approaches have speeded up the process by first making hard decisions about the set of parents and children for each variable, in order to break large-scale problems into sets of tractable local neighbourhoods. We use this principle in order to apply a structure learning committee for orientating edges between variables. We find that a combination of weak structure learners can be effective in recovering causal dependencies. Though such a formulation would be intractable for large problems at the global level, we show that it can run quickly when processing local neighbourhoods in turn. Experimental results show that this localized, committee-based approach has advantages over standard causal discovery algorithms both in terms of speed and accuracy.

Keywords: Bayesian Network, feature ranking, relevance learning, committee method

1. Introduction

Current methods for causal structure learning tend to be computationally intensive or intractable for large datasets. Most approaches towards causal structure learning can be categorized into two classes: constraint-based approaches that use independence tests and score-based techniques that search for Bayesian networks. The former are slow because independence has to be tested between variables under many different conditioning sets. The latter are slow because of the possible number of Bayesian networks; a naïve scoring with just 10 variables would have to consider around 10^{18} configurations (Robinson, 1977).

Some recent approaches have speeded up the process by finding the network skeleton first and then doing local neighborhood learning to orient the skeleton edges, such as MMHC (Tsamardinos et al., 2006). Building the skeleton of a network is an easier task than orientating the edges as we only look at associations between variables and not the causal relationships between them.

In this paper we propose a method for fast structure learning based on finding the set of parents and children for each variable, and then applying a committee of structure learners to make a joint decision about edge orientation. Some of the structure learning methods we use would be intractible when applied globally to a dataset with many variables, but can run rapidly at neighbourhood level. When the structure learners are based on different principles (e.g. a mixture of constraint-based and score-based) it is significant when they agree with each other, and in particular we find that this strategy gives good worst-case accuracy.

The contributions of this paper are:

- We generalise previous work on restricting the search space to speed up structure learning;
- We present a novel local structure learning algorithm, EPC, specifically intended for analysing a target variable and its immediate neighbourhood;
- We show how different structure learners can be combined in a committee to give results with better consistency.

The rest of the paper is organized as follows. In section 2 we discuss the initialization step for finding the skeleton of a network of variables. Section 3 discusses the causal discovery committee. We present experimental evidence in section 4, and summarise our findings in section 5.

2. Skeleton Discovery

The overall aim of skeleton discovery is to consider each variable in a dataset and find the set of directly neighbouring variables. To find the neighbourhood of one variable, we begin by considering all variables as potential neighbours and then filtering down this set in two phases. We first employ Relevance Learning Vector Quantization (RLVQ), a fast prototype-based classification method, to do an initial feature selection for each variable. The variables found to have low relevance during this stage are removed from the estimated set of neighbours. We then apply the HITON algorithm on the resulting variables to narrow down this set.

LVQ and RLVQ are prototype-based classification methods applied in supervised learning. They employ a distance measure (typically Manhattan distance or quadratic Euclidean distance) that quantifies the similarity of a given feature vector with a prototype (representative) of any particular class. The distance measure (Manhattan distance) for two arbitrary vectors $x, y \in \mathbb{R}^N$ can be defined as:

$$d(x,y) = \sum_{j=1}^{N} \left| x_j - y_j \right| . \qquad (1)$$

Because the features have varied meanings and magnitudes in the data, quantifying their similarity by a uniform distance measure tends to be problematic. These differences are accounted by relevance learning schemes like RLVQ that employ adaptive scaling factors that scale the features based on their relevance for classification. This takes the form

$$d_{\lambda}^{i}\left(w^{i}, \xi\right) = \sum_{j=1}^{N} \lambda_{j}^{i} \left| w_{j}^{i} - \xi_{j} \right| \qquad (2)$$

where w denotes a prototype or representative vector of a particular class, ξ denotes a data vector and the adaptive relevance factors λ_j^i are restricted to non-negative values and obey the normalization $\sum_{j=1}^{N} \lambda_j^i = 1$. The special case $\lambda_j^i = 1/N$ for all $j = 1, \ldots N$ is analogous to the original LVQ measure. The RLVQ adapts the prototypes and the relevance factors for each training run through the data until the error rate is at a minimum. Further details on LVQ and RLVQ can be obtained from Bojer et al. (2001).

These methods have been used is several applications because on top of being intuitively easy to understand they are easy to implement and their complexity is controlled by the user. For our purposes however we draw from the fact that they are fast and have been shown to give high accuracy in identifying relevant features for classification (Biehl et al., 2007).

HITON is a standard algorithm for feature selection that, assuming the joint data distribution is faithful to a Bayesian Network, carries out statistical tests on the data to determine the Markov boundary and the Markov blanket of a target variable. HITON has been proven to accrue two main advantages over other feature selection algorithms: 1) it reduces the number of variables in the prediction models roughly by three orders of magnitude relative to the original variable set while improving or maintaining accuracy, and 2) it outperforms the baseline algorithms by selecting smaller variable sets than the baselines (Aliferis et al., 2003). Because HITON takes several hours to run for datasets with hundreds or thousands of variables, the RLVQ preprocessing step is useful to speed the process of obtaining Markov boundaries for each variable.

To summarise, for each variable in the local neighbourhood a set of features relevant for its classification are obtained using RLVQ (phase 1). For each of these sets of relevant features, the HITON algorithm is used to further narrow down the set of parents and children of the variable under consideration. Given this skeleton of undirected edges between variables, a committee of structure learning methods is then used to vote on the causes (parents) and effects (children), as described in the next section.

3. Causal Discovery Committees

Once a skeleton of the network is found we apply a structure learning committee for orientating edges between variables. We find that a combination of weak structure learners can be effective in recovering causal dependencies. Though such a formulation would be intractable for large problems at the global level, we show that it can run quickly when processing local neighbourhoods in turn.

The structure learning committee method takes the neighbourhood of each variable and applies different algorithms to determine whether each neighbour of that variable is a cause or an effect. If the majority of the algorithms determine that a given neighbour is a cause, then we classify it as a cause. Effects are classified in the same way. We do not apply any conflict resolution at the moment; our method might return bi-directional causes. Algorithm 1 shows the committee voting method.

Algorithm 1 Localized causal discovery committee.

```
input: $\mathbf{c}_1 \ldots \mathbf{c}_N$, data vectors for variables $C_1, \ldots, C_N$
       $PC\,(i)$, set of parents and children for each variable $C_i$.
for each variable $C_i$, $i = 1 \ldots N$ do
    for each algorithm $Algo_j$ do /* PC, EPC, GES, MWST, LiNGAM, K2 */
        $causes(C_i, j), effects(C_i, j) \leftarrow Algo_j\,(C_i, PC\,(i))$
    $\mathbf{C_i} \leftarrow$ majorityVote($causes(C_i, :)$)
    $\mathbf{E_i} \leftarrow$ majorityVote($effects(C_i, :)$)
return: $\{\mathbf{C},\mathbf{E}\}$, causes and effects of variables $C_1, \ldots, C_N$.
```

In the remainder of this section, we first introduce a novel structure learning algorithm, EPC, and then list the other committee members.

3.1 Expected Partial Correlation (EPC) Method

EPC is a simple local neighborhood structure discovery algorithm. Given the set of parents and children of a target variable, it returns a probability of each neighbourhood variable being either a cause or an effect. It is based on partial correlation as a measure of conditional independence, which is true in certain cases such as binary or linear Gaussian networks (Baba et al., 2004). We denote the Pearson correlation coefficient between A and B as ρ_{AB}, and the partial correlation between A and B conditioned on C as $\rho_{AB\cdot C}$.

The algorithm works by considering different three-variable subsets of the target and neighbourhood. We divide 3-variable acyclic connected models into three interesting classes: the collider or V-structure ($A \rightarrow C \leftarrow B$); the chain ($A \rightarrow C \rightarrow B, A \leftarrow C \leftarrow B$); and the fork ($A \leftarrow C \rightarrow B$), as shown in Figure 1(i-iii). The chain and the fork have the same conditional independency $A \perp\!\!\!\perp B \mid C$, while the V-structure has the unique property $A \perp\!\!\!\perp B$ but $A \not\perp\!\!\!\perp B \mid C$. We only consider variables B which are not directly connected to A, as this would imply a cycle, which we cannot make any inferences about.

Given a particular sample size and type of distribution, we can work out what distribution of empirical correlation and partial correlation we expect from each different class. We show histograms of correlation and partial correlation in simulated networks in Figure 2. 10,000 binary models in each class (collider, chain, fork) were randomly created, with conditional probability tables sampled from the uniform distribution. We can see that the V-structure is the only case where conditioning on the variable C increases the scale of the correlation between A and B, from the distribution of $|\rho_{AB\cdot C}| - |\rho_{AB}|$ in Figure 2 (right).

The histogram in Figure 2(right) therefore gives us a probability distribution on the likelihood $P\left(\delta_{ABC} | class(A,B,C)\right)$, where $\delta_{ABC} = |\rho_{AB\cdot C}| - |\rho_{AB}|$ and $class(A,B,C)$ can be "collider" or "chain/fork" as in Figure 1(i-iii). By specifying priors on $P(class(A,B,C))$ we can then calculate the probability that A is a cause of C, using the assumption that in the "collider" class A is always a cause of C, whereas in the "chain/fork" class, there are 3 possible orientations, in only one of which A is a cause of C. While trying to calculate whether A is a cause or an effect, we incorporate evidence from each of the B_i's in the neighbourhood and obtain $P\left(Cause(A,C) | \delta_{AB_1C}, \delta_{AB_2C}, \ldots, \delta_{AB_{N-1}C}\right)$ for a neighbourhood of size N and threshold at 0.5 to determine whether A is a cause or an effect. Algorithm 2 shows the steps of this calculation.

The algorithm is limited to certain distributions, such as binary or Gaussian networks, where partial correlation is a measure of conditional independence. The method would fail in non-linear relationships between variables such as an XOR function. We also do not have an ana-

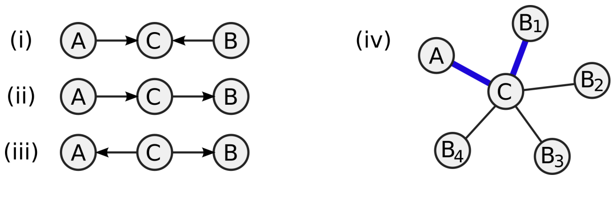

Figure 1: Possible 3-variable structures: (i) collider, (ii) chain, (iii) fork. Panel (iv) shows an example local neighbourhood for a variable C. The EPC algorithm orientates the edge AC by looking at the supporting evidence from each of the B_i's.

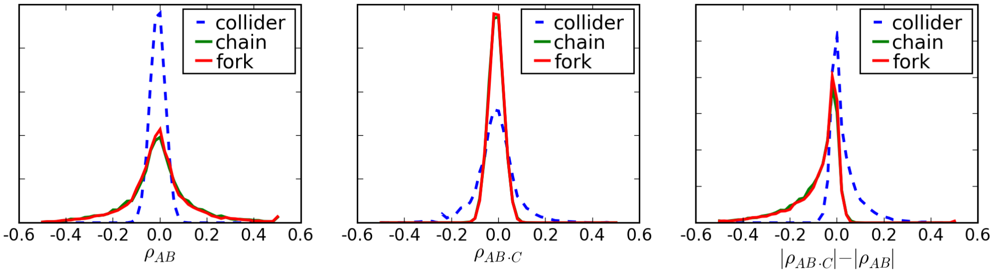

Figure 2: Histograms of correlation and partial correlation from 10,000 simulated 3-variable binary networks of each class, with 1000 samples drawn from each. Chains and forks have indistinguishable correlation distributions.

lytical form for the likelihood function; we currently have to estimate the distribution through simulations.

However the advantages of the algorithm are as follows. First, it is cheap to run: $O(N^2)$ in the neighbourhood size and $O(M)$ in the sample size. Second, it provides probabilities rather than categorical outputs – most methods based on CI constraints simply accept or reject a causal hypothesis. Third, we have the ability to incorporate prior beliefs about the orientations of edges. Fourth, it is useful as a committee member, as it gives high confidence when there is a V-structure and low confidence otherwise.

The performance of the EPC algorithm in recovering true causes and effects is evaluated in section 4.

Algorithm 2 EPC Algorithm to distinguish between local causes and effects.

```
input: $\mathbf{c}, \mathbf{b_1}, \ldots, \mathbf{b_N}$, data vectors for target variable $C$ and
        set of parents/children $B_1, \ldots, B_N$.
    $P(Cause(B_i, C))$ for all $i$, priors for each $B_i$ being a cause of $C$.
for each variable $B_i$ do
    for each variable $B_{j \neq i}$ ($B_i$ not a neighbour of $B_j$) do
        $\delta_{ij} \leftarrow |\rho_{B_i B_j \cdot C}| - |\rho_{B_i B_j}|$
        Compute likelihoods $L(\delta_{ij} | class(B_i, B_j, C))$
            where $class(B_i, B_j, C) \in \{$“collider’, “chain/fork”$\}$
    $causeodds(i) \leftarrow P(Cause(B_i, C)) \prod_{i \neq j} (L(\delta_{ij}|\text{collider}) + L(\delta_{ij}|\text{chain/fork}))$
    $effectodds(i) \leftarrow (1 - P(Cause(B_i, C))) \prod_{i \neq j} 2L(\delta_{ij}|\text{chain/fork})$
    $P(Cause(B_i, C) | \mathbf{c}, \mathbf{b_1}, \ldots, \mathbf{b_N}) \leftarrow \frac{causeodds(i)}{causeodds(i) + effectodds(i)}$
return: $P(Cause(B_i, C) | \mathbf{c}, \mathbf{b_1}, \ldots, \mathbf{b_N})$ for each $i$, posterior probabilities that
        each $B_i$ is a cause of $C$.
```

3.2 Other Committee Members

Standard algorithms were used in conjunction with EPC to form the structure learning committee. The strength of the committee method is derived from applying each of these methods to the same skeleton obtained from Section 2 for each dataset. These methods were selected to include methods based on different principles. These methods were as follows[1].

[1] Implementations from four packages were used: BNT (http://www.ai.mit.edu/~murphyk/Software/BNT/bnt.html), BNT-SLT (http://bnt.insa-rouen.fr/), LiNGAM (http:

PC : This is a common benchmark constraint-based causal discovery algorithm, introduced by Spirtes et al. (1993). A confidence level of 0.05 was used with this method.
MWST : The MWST algorithm as introduced by Chow and Liu (1968) is based on the maximum weight spanning tree. It essentially associates a weight with each edge obtained according to some similarity criterion (mutual information between variables or BDeu score) and then builds the maximum spanning tree of the obtained graph. For our experiments we used the mutual information between variables as a measure of (conditional) dependence.
GES : The greedy search (GS) algorithm is an implementation of a standard optimization heuristic. Greedy Equivalent Search is an extension of the GS algorithm that optimizes searching the DAG space by searching in the Markov equivalent space. This method initially starts with an empty graph, adds arcs until the score cannot be improved then tries to suppress some irrelevant arcs (Munteanu and Bendou, 2002). For our experiments we used the Bayesian Information Criterion (BIC) as our scoring function with an instantiation cache of 300.
K2 : The K2 algorithm (Cooper and Herskovits, 1992) is a probabilistic algorithm that maximizes structure probability given the data. It defines the Bayesian measure(BIC/BDeu) which is a quality measure of the network given the data. We use it in the committee to vote on whether a feature is an effect of the target variable only and not a cause because it is easier to specify a node order for the former. For our experiments we used the Bayesian Score (BIC) as our scoring function.
LiNGAM : LiNGAM (Shimizu et al., 2006) is a more specific technique that attempts to discover the causal structure in linear non-Gaussian acyclic models. We include it in the committee because it provides a relatively different technique form the rest of the committee members and hence can account for certain distributions on which the other members may produce poor results. For our experiments default settings were used, as provided in the author's implementation.

4. Experiments

We test our methods on several standard datasets with known generating structures. The causal structures found by our methods are evaluated using the same edit distance score based evaluation method used to evaluate the NIPS 2008 causality challenge entries. In this method, a confusion matrix C_{ij} that specifies the number of relatives confused for another type of relative is computed. It evaluates the 14 types of relatives in a depth-3 network. A cost matrix A_{ij} is also computed to account for the edit distance between the relatives. The edit distance specifies the number of substitutions, insertions, or deletions to go from one string to another. A score for a particular structure is then computed as $S = \sum_{ij} A_{ij}C_{ij}$. The results are summarised in Table 1 for the different datasets. In this table, we first show the performance of different standard methods and the EPC algorithm when applied in a local neighbourhood setting. Each of the algorithms in Table 1 are applied on the same skeleton (for each dataset) obtained using feature selection/reduction techniques discussed in Section 2. We then show the performance of all methods when combined in committee, using a majority voting scheme. As a benchmark we then show performance of the PC algorithm when applied as standard to the whole datasets (no localization).

Table 2 (a-f) shows confusion matrices for the committee output of three datasets. The confusion matrices give an idea of the recall and precision rates of the method. An ideal confusion matrix would be a diagonal matrix indicating the true positives and true negatives. The figures that are not on the diagonal represent the numbers of false positives (spurious causes, bottom

//www.cs.helsinki.fi/group/neuroinf/lingam/) and Causal Explorer (http://discover.mc.vanderbilt.edu/discover/public/causal_explorer/)

left) and the numbers of false negatives (spurious independencies, top right).The performance of our committee method in the LOCANET challenge is given in Table 2 (g), compared to other participants. Execution time for HAILFINDER using the standard PC algorithm for example was 4452.4 seconds, while for the committee this time was 190 seconds for obtaining the skeleton and 13.3 seconds for obtaining the local graph from the committee.

Table 3 shows other performance metrics used for datasets where we know the ground truth. We calculate precision (ratio of true causes found to total causes found), recall (proportion of true causes found to actual number of causes), and Fmeasure $= \frac{2\times\text{Precision}\times\text{Recall}}{\text{Precision}+\text{Recall}}$.

Method	LUCAS (2000)	LUCAP (2000)	ALARM (5000)	ASIA (2000)	INSURANCE (2000)	HAILFINDER (20000)
PC	1.91	2.14	2.43	2.08	2.81	1.79
EPC	0.91	1.81	0.57	2.94	2.2	2.2
GES	1.86	2.14	1.5	2.96	3.38	2.58
MWST	2.86	2.46	2.21	1.7	2.7	1.68
K2	2.18	1.95	2.1	1.78	2.15	1.79
LiNGAM	1.73	3.08	1.93	1.38	2.81	1.43
Committee (M)	1.65	1.9	1.07	2.86	2.46	2.39
PC‡	2.91	3.38	2.72	3.29	2.81	2.73

Table 1: Evaluation of edit distances for various algorithms with known networks (sample size in brackets). Committee (M) denotes the committee decision with *Majority Voting*. PC‡ represents results obtained on running the standard PC on the whole dataset.

5. Discussion

From the challenge results we can see that our method gives performance comparable to other entries , while employing a method designed to give fast inference time. We provide two scores

	→	x
→	66	123
x	89	20458

(a) Lucap-EPC

	→	x
→	78	236
x	178	20244

(b) Lucap-Committee

	→	x
→	10	22
x	74	1263

(c) Alarm-EPC

	→	x
→	18	18
x	131	1202

(d) Alarm-Committee

	→	x
→	2	27
x	28	3079

(e) LiNGAM

	→	x
→	7	27
x	45	3057

(f) Committee

Dataset	Score	Others(range)
CINA	2.32	1.70 - 3.31
REGED	0.22/0.439	0.27 - 0.50
SIDO	3.46	3.31 - 3.48

(g) Results on challenge datasets

Table 2: (a-g) Confusion matrices showing the algorithms with the winning edit distance and the corresponding majority committee decisions in terms of true causes found (top left), spurious causes (bottom left), true independencies (bottom right) and spurious independencies (top right) for Lucap (a-b), Alarm (c-d), and Hailfinder (e-f). (g) LOCANET challenge results.

Dataset/Method	PRECISION	RECALL	FMEASURE
Lucap-EPC	0.43	0.54	0.47
Lucap-Committee	0.30	0.33	0.32
Alarm-EPC	0.12	0.45	0.19
Alarm-Committee	0.12	1.00	0.22
Hailfinder-LiNGAM	0.07	0.07	0.07
Hailfinder-Committee	0.13	0.26	0.18

Table 3: Evaluation of Precision, Recall and Fmeasure scores for three of the datasets contrasting different individual algorithms with the committee

for the REGED dataset; the first one represents the initial submission where our method failed to find causes but due to the bias in the scoring statistic, this did quite well, the second represents a score obtained by the committee with the voting threshold decreased. For the benchmark datasets we find that the quality of the committee decisions is close to the best committee member in each case. Results obtained for applying PC to whole datasets without localization generally indicate a lower accuracy rate than either PC with localization or the causal discovery committee. In principle to increase precision (at the expense of recall) we can increase the voting threshold upwards towards the unanimous voting level. Conversely it is also possible to increase the recall rate by altering the voting threshold in the opposite fashion. For applications where a causal relationship needs to be established with high precision, a unanimous voting scheme may be used though we have not so far analysed the accuracy of this approach.

The confusion matrices in Table 2 indicate that the committee generally obtains more true positives (higher recall - Table 3) than the corresponding committee member with the best average edit distance score. However the committee also generally obtains more false positives (lower precision) which accounts for the committee score not being as good as that of the best algorithm in each case.

Currently our method does not explicitly handle conflict of orientation so it is possible to have a situation where we find that $P \rightarrow Q$ and also $Q \leftarrow P$. The output is therefore not a DAG. We conjecture that finding bi-directional causes $P \leftrightarrow Q$ may indicate the presence of a hidden variable which influences both P and Q.

We have used our local committee framework with particular structure learning algorithms, but anticipate that other algorithms can be used in future work. Future research will also look at weighting the committee members based on derived properties of the dataset.

Acknowledgments

We would like to thank Michael Biehl for helpful discussions on relevance learning. The work was supported in part by the Dutch NUFFIC NPT project.

References

C. F. Aliferis, I. Tsamardinos, and A. Statnikov. HITON, A Novel Markov Blanket Algorithm for Optimal Variable Selection. In *Proc. of the 2003 American Medical Informatics Association (AMIA) Annual Symposium*, pages 21–25, 2003.

K. Baba, R. Shibata, and M. Sibuya. Partial Correlation and Conditional Correlation as Measures of Conditional Independence. *Australian and New Zealand Journal of Statistics*, 46(4): 657–664, 2004.

M. Biehl, R. Breitling, and Y. Li. Analysis of Tiling Microarray Data by Learning Vector Quantization and Relevance Learning. In *Proc. of the 2007 IDEAL*, 2007.

T. Bojer, B. Hammer, D. Schunk, and Tluk von Toschanowitz. Relevance determination in learning vector quantization. In Verleysen M, editor, *European Symposium on Artificial Neural Networks*, pages 271–276. d-facto publications, 2001.

C. K. Chow and C. N. Liu. Approximating discrete probability distribution with dependence trees. *IEEE Transactions on Information Theory*, 14(3):462–467, 1968.

G. F. Cooper and E. H. Herskovits. The induction of probabilistic networks from data. *Machine Learning*, 9(4):309–347, 1992.

P. Munteanu and M. Bendou. The EQ framework for learning equivalence classes of Bayesian networks. In *First IEEE International Conference on Data Mining (IEEE ICDM)*, San Jose, 2002.

R. W. Robinson. Counting unlabeled acyclic digraphs. In C.H.C. Little, editor, *Combinatorial Mathematics V*, volume 622 of *Lecture Notes in Mathematics*. Springer, Berlin, 1977.

S. Shimizu, P. O. Hoyer, A. Hyvarinen, and A. Kerminen. A Linear Non-Gaussian Acyclic Model for Causal Discovery. *Machine Learning Research*, 7:2003–2030, 2006.

P. Spirtes, C. Glymour, and R. Scheines. *Causation, Prediction and Search*, volume 81. Springer Verlag, Berlin, 1993.

I. Tsamardinos, L.E Brown, and C.F. Aliferis. The max-min hill-climbing Bayesian network structure learning algorithm. *Machine Learning*, 65:31–78, 2006.

Appendix A. Pot-luck challenge: FACT SHEET.

(for a task solved)

Title: LOCANET

Ernest Mwebaze & John A. Quinn
Faculty of Computing & IT
Makerere University
(emwebaze, jquinn)@cit.mak.ac.ug

Task(s) solved: Local Structure Discovery

Method:

Our method uses a relevance learning algorithm (RLVQ) and the HITON algorithm to reduce the feature set to parents and children of each feature. A novel partial correlation algorithm in a committee of standard structure learning algorithms then votes on which of the features are parents and which are children for each Markov boundary obtained from the feature reduction step. Because we employ feature reduction initially, the method is fast and because a committee votes on the edge directions, the method yields high accuracy.

- Preprocessing : None
- Causal discovery : Use of novel probabilistic partial correlation algorithm in committee of standard structure learning algorithms ; PC, GES, MWST, K2 and LiNGAM.
- Feature selection : Use of Relevance Learning Vector Quantization and HITON algorithms
- Classification : None
- Model selection/hyperparameter selection : Majority vote of committee of algorithms

Results:

Dataset/Task	Score 1
CINA	2.32
REGED	0.22/0.439
SIDO	3.46

Table 4: Result table.

Advantages:

- Quantitative advantages : Our method employs feature selection techniques to obtain relevant features for classification from which we can obtain relevant features for causality. This reduces the processing time as indicated in our paper.
- Qualitative advantages : We employ a novel method, Expected Partial Correlation (EPC), that offers comparable results when compared with other standard algorithms on known datasets as illustrated in Table 1 in the paper.

Method Implementation:

We implemented our method in matlab on an Intel Duo CPU T7100 laptop computer with 1024 MB or RAM.

The standard algorithms were implemented using standard packages from different individuals/organizations these included :-

- Bayesian Network Toolkit (http://www.ai.mit.edu/m̃urphyk/Software/BNT/bnt.html),
- Structure Learning Toolkit (http://bnt.insa-rouen.fr/),
- LiNGAM package (http://www.cs.helsinki.fi/group/neuroinf/lingam/)
- Causal Explorer (http://discover.mc.vanderbilt.edu/discover/public/causal_explorer/).

The whole application is built up into two modules, one that does the feature reduction and the other that does the structure learning based on the novel algorithm EPC and a host of standard algorithms including PC, MWST, GES, LiNGAM and K2.