

# **Identifying Markov Blankets with Decision Tree Induction**

Lewis Frey, Douglas Fisher

Ioannis Tsamardinos, Constantin F. Aliferis,  
Alexander Statnikov

# Outline

- Introduction
- Markov Blankets
- Algorithms
  - PC
  - C5.0
  - C5C
- Data Sets
- Results
- Discussion
- Limitations
- Future Work
- Conclusion

# Introduction

We have developed a method for identifying Markov Blanket variables.

The method, C5C, is a simple augmentation to a widely used machine learning application C5.0.

## Key Points

1. Easy to use & Accessible
2. Computationally efficient
3. Scales to large data sets
4. Performance is equivalent or better than PC

# Markov Blanket (MB)

The Markov Blanket is the minimum conditioning set that makes all other variables independent for a particular target.

# Applications of the MB

- Feature Selection/Reduction
  - clinical diagnosis
  - text categorization
  - gene expression
  - web analysis
- Causal Discovery
  - Guide experimental tests for direct causes of a target variable
- Bayesian Network Construction
  - Guide for Bayesian Network learning (Margaritis & Thrun)

# Identifying MB

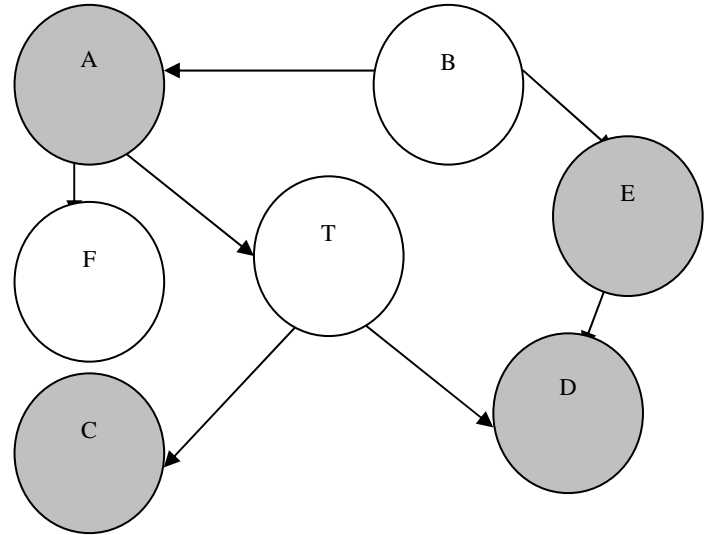
- Some of the algorithms used for identifying Markov Blankets
  - PC: Global Bayesian Network identification
    - limited to a few hundred variables
  - Grow-Shrink: Local Markov Blanket Identification
  - Koller-Sahami: Ranked list of Markov Blanket variables
- What is needed is an algorithm for identifying Markov Blankets that is
  - computationally efficient (low cost) & accessible
  - scalable to large data sets
  - scalable to large Markov Blankets

# Motivation

- Evaluate the ability of C5.0 to identify Markov Blankets
  - C5.0 an inexpensive, efficient, off-the-shelf decision tree induction engine
  - Can it identify MB variables?
- Markov Blankets and feature relevance
  - Decision tree induction feature selection (Cardie; Aluallim & Dietterich)
    - Mixed findings
- C5C is a simple modification of C5.0
- Common principles of feature relevance that underlie induction of classifiers and Bayesian Networks.

# Bayesian Network Graph

- In Bayesian Networks the union of parents and children of  $T$ , and parents of children (spouses) of  $T$  is equivalent to the Markov Blanket.
- In Figure, the Markov Blanket for  $T$  is  $\{A, C, D, E\}$ . This means that variables  $B$  and  $F$  are independent of  $T$  conditioned on  $\{A, C, D, E\}$ .



# PC: Algorithm for MB Identification

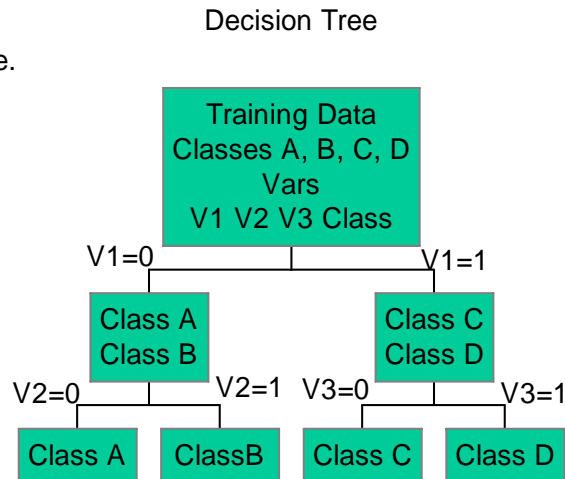
- Initial Bayesian Network graph is fully connected and unoriented
- phase I
  - eliminates edges: This is achieved by using the criterion that variable  $A$  has a direct edge to variable  $B$  if, and only if, for all subsets of features there is no subset  $S$ , such that  $A$  is independent of  $B$  conditioned on  $S$ .
- phases II and III
  - orients the edges by performing global constraint propagation
  - If not able to orient some edges, the output class of structurally equivalent BNs
- Significance thresholds based
  - $G^2$  statistic
  - Fisher's  $z$ -test (linear domains )

# PC Scalability

- Intractable on large densely connected data sets
- Complexity is the number of variables  $V$  raised to the maximal degree,  $d$ , (i.e.,  $O(V^d)$ )
- Search and score Bayesian methods
  - Difficulties in scaling
- Distributional assumption (“monotone restriction”) Cheng et al.
  - improves the complexity (to  $O(V^4)$ )
  - properties currently being explored

# Decision Trees

C5.0 is a decision tree induction engine.



## Conjunctive Rules

$V1=0$  and  $V2=0$  -> Class A

$V1=0$  and  $V2=1$  -> Class B

$V1=1$  and  $V3=0$  -> Class C

$V1=1$  and  $V3=1$  -> Class D

# C5.0

- Greedy algorithm that recursively partitions the data set into a tree based on variables that give the largest reduction in entropy.
- Classes  $C_1, \dots, C_N$  in data set  $S$  where  $P(S_c)$  is the probability of class  $C$  occurring in the data set  $S$ :

$$E(S) = - \sum_{c=1}^N P(S_c) * \log_2 P(S_c) \quad \text{Gain}(S, V) = E(S) - \sum_{v \in \text{values}(V)} \frac{|S_v|}{|S|} * E(S_v)$$

- A final decision tree is changed to a set of rules by converting the paths into conjunctive rules and pruning them to improve classification accuracy.

# C5C

- **Hypothesis is that frequently occurring features in C5.0 production rules provide a good approximation of the  $MB(T)$**
- This is tested through a simple augmentation of C5.0.
- C5C uses a simple script that counts the occurrence of the variables in the C5.0 rules output and produces a ranking of the variables using frequency.
- If the hypothesis is correct, the Markov Blanket variables should be in the set of the most frequent variables.
- Consequently, a threshold is needed to distinguish between Markov and non-Markov Blanket variables.

# Data Sets

- In order to test the accuracy in identifying Markov Blanket variables, the Markov Blanket must be known. Bayesian Networks are used to generate data sets with known Markov Blankets.
- **Data generated from Bayesian Networks**
  - Alarm Network (37 variables) medical monitoring network
  - Hailfinder Network (56 variables) severe weather forecasting
  - Insurance Network (27 variables) claim costs for insurance policies
  - Mildew Network (35 variables) amount fungicides to use on wheat
  - Barley Network (48 variables) yield & quality of barley without pesticides
- **Artificial Bayesian Network**
  - Explore number of variables and the sample size upon the algorithms' ability to find the Markov Blanket for one variable.
  - The variable has three parents, two children and one parent of a child for a total of six Markov Blanket variables.

# Measures

- *Sensitivity* is the ratio of correctly predicted MB variables over true MB variables.
- *Specificity* is the ratio of correctly predicted (i.e., excluded) non-MB variables over true non-MB variables.
- $$dist = \sqrt{(1 - sen)^2 + (1 - spec)^2}$$

# Testing C5C

Four methods of identifying the Markov Blanket variables are to be examined.

- The first, called the **oracle** test, chooses the best frequency threshold given **knowledge** of the true MB. For this test C5C is compared to C5.0 and this test is intended as a best-case analysis.
- The second strategy finds the frequency threshold that gives the best accuracy on a test set. In this test, C5C is compared to C5.0.
- The third strategy employs the  $G^2$ -test to test for independence of the top  $k$  C5C variables from target given conditioning set of top  $n$  C5C variables (where  $k > n$ ).
- The fourth compares area under ROC for C5C and PC.

# Average Over Targets

- Table 1. Average over target variables of sensitivity (sen), specificity (spec) and distance (dist) for C5.0 and the **oracle** thresholds for C5C. Data sets have 20,000 instances. The asterisk (\*) denotes the mean distance for C5C is significantly different from C5.0 by the paired Wilcoxon signed rank test of the equality of means ( $p < 0.05$ ).

Data Set	C5.0			C5C		
	Sen	Spec	Dist	Sen	Spec	Dist
Alarm	0.83	0.84	0.32	0.82	0.99	0.18*
Hailfinder	0.81	0.27	0.89	0.80	0.98	0.22*
Insurance	0.89	0.46	0.64	0.78	0.93	0.25*
Mildew	0.94	0.11	0.95	0.80	0.90	0.26*
Barley	0.84	0.43	0.67	0.76	0.94	0.28*

# Closer to MB

- Table 2. Number of target variables (var) out of the total for each data set that the distance of C5C's **oracle** predicted MB is closer to the true MB than C5.0 otherwise it is equivalent to C5.0.

DATA SET	FREQ THAT C5C IS CLOSER TO TRUE MB THAN C5.0	TOTAL VAR
ALARM	15	37
HAILFINDER	46	56
INSURANCE	22	27
MILDEW	34	35
BARLEY	39	48

# Average MB Size

- Table 3. Average Markov Blanket size and average distance to the true MB for C5.0 and **oracle** threshold for C5C. The asterisk (\*) denotes significance by the paired Wilcoxon signed rank test of the equality of means ( $p < 0.05$ ) in comparing C5.0 and C5C distance.

Data Set	Avg MB size		Distance	
	C5.0	C5C	C5.0	C5C
Alarm	16	3	0.40	0.05*
Hailfinder	49	4	0.93	0.12*
Insurance	19	6	0.67	0.20*
Mildew	31	6	0.98	0.27*
Barley	34	7	0.73	0.22*

# Accuracy & G<sup>2</sup>-Test

- Table 4. Average over target variables of sensitivity (sen), specificity (spec) and distance (dist) for C5C with the C5.0 decision tree test set accuracy determining threshold and the G<sup>2</sup>-test identifying the MB. For the test accuracy the training set is 16,000 instances and the test set is 4,000 instances. The G<sup>2</sup>-test uses 20,000 instances. The asterisk (\*) and plus (+) denote the mean distance for the method is significantly different from C5.0 (Table 1) by the paired Wilcoxon signed rank test of the equality of means at  $p < .05$  and  $p < 0.1$ , respectively.

Data Set	C5C – Test Acc.			C5C – G <sup>2</sup>		
	Sen	Spec	Dist	Sen	Spec	Dist
Alarm	0.79	0.97	0.23	0.83	0.97	0.20 <sup>+</sup>
Hailfinder	0.76	0.95	0.27*	0.79	0.88	0.29*
Insurance	0.74	0.80	0.42*	0.76	0.88	0.31*
Mildew	0.75	0.85	0.38*	0.78	0.80	0.35*
Barley	0.71	0.91	0.36*	0.76	0.87	0.31*

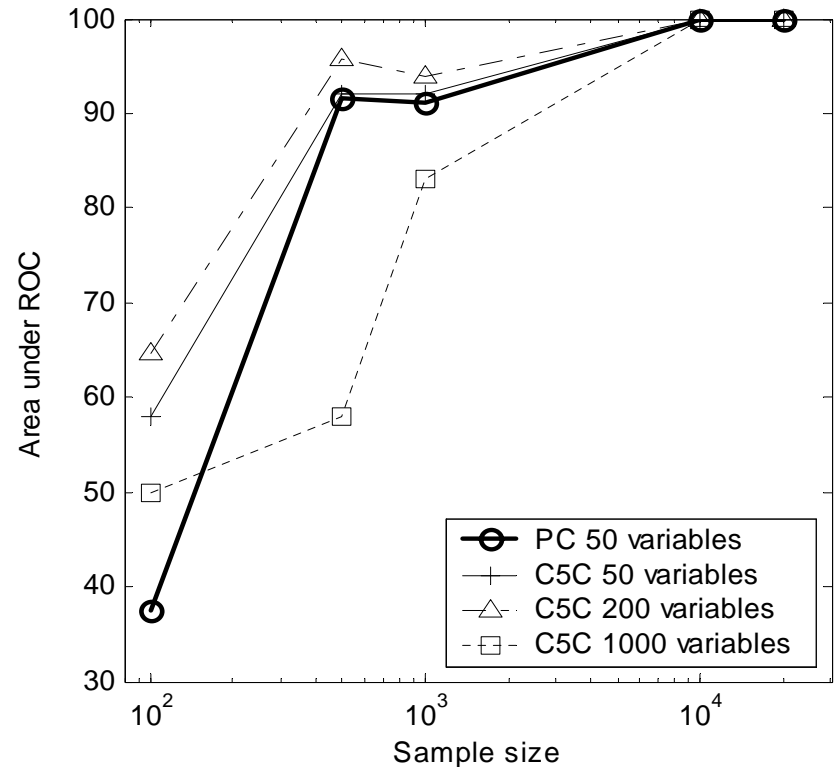
# Average ROC

- Table 5. Average ROC over all variables in Network: PC & C5C with 6 thresholds (T'holds) and C5C with 101 thresholds. Data sets have 20,000 instances. The asterisk (\*) denotes the mean ROC for C5C as significantly different from PC by the paired Wilcoxon signed rank test ( $p < 0.05$ ).

Data Set	Total Targets	PC	C5C	
# of T'holds		6	6	101
Alarm	37	96.3	91.1	91.4
Hailfinder	56	91.7	87.5	88.9
Insurance	27	82.0	86.3	88.0
Mildew	35	64.3	80.0*	88.0*
Barley	48	50.0	81.6*	85.9*

# Number of Variables

- Figure 2. Area under the ROC for MB size 6 for PC with 50 variables and C5C with 50, 200 and 1,000 variables. The artificial Bayesian Network sample sizes are 100, 500, 1,000, 10,000 and 20,000 examples.



# Discussion

- In the oracle test, C5C provides a better approximation of the MB than C5.0 via a distance measure across all five data sets.
- The accuracy test and  $G^2$ -test determined thresholds, C5C compared to C5.0 provided a better MB approximation for four of the five data sets.
- For area under the ROC, C5C offers a better estimate of the MB compared to PC on two data sets. C5C and PC are equivalent on the remaining three data sets.
- C5C performed well with large numbers of variables and limited sample sizes.

# Limitations

- The C5C algorithm is not able to find the Markov Blanket for target variables that occur a disproportionate number of times in one class.
- The C5C algorithm is not able to predict the Markov Blanket when one variable predicts the target variable without error.
  - However, this is an unfaithful distribution because there are deterministic relationships in the data set.

# Future Work

- C5C is applicable to the area of feature selection in machine learning and data mining.
  - Compare to other feature selection methods
- The threshold methods examined are preliminary.
  - Explore a range of threshold determining methods
- Expansion of C5C
  - explore pruning levels in C5.0
  - examine weighting importance of features instead of straight counting.
    - size of rule, accuracy of rule, coverage
- Use the ranked C5C variables to identify Markov Blanket variables on real world data sets..

# Conclusion

- C5C performs simple post-processing of C5.0 rules.
- C5C algorithm performs as well as or better than C5.0 and PC in identifying Markov Blanket variables on generated data sets.
- C5C scales to large data sets.