Supplementary File

A Novel Feature Selection Method via Mining Markov Blanket

Waqar Khan, Lingfu Kong, Sohail M.Noman and Brekhna Brekhna

———————*———————

This is a supplementary file for the paper entitled "A Novel Feature Selection Method via Mining Markov Blanket" in Journal of Intelligent Information Systems. Additional Section, Figures, and Tables are kept into this file and cited inside the original paper.

S1. Conceptual Background

Theorem S1(1) *If G is faithful to P, if X and Y are adjoining in G, then becomes two kinds of situations: 1) X and Y are conditionally dependent given every set of features that does not contain X and Y, $\forall S \subseteq R \setminus \{X, Y\}$, $X \not\perp\!\!\!\perp Y \mid S$. 2) Since features (variables) X, Y, and Z into the G. If features (variables) X and Z are adjoining, Y is adjoining of Z, and Y is not adjoining of X. Then they build a V-structure$(X \rightarrow Z \leftarrow Y)$ with Y as a collider feature iff $X \not\perp\!\!\!\perp Z \mid S$, $\forall S$, $(X, Z) \notin S$ and $Y \in S$.*

Definition S1 (D-separation) A path D between a feature (variable) X and feature (variable) Y is blocked by conditioning on a set of features S if each one of the following situations is met, as shown in Figure S1: Situation 1. S has a non-collider feature (variable) on path D, which is Z. Situation 2. Although Z is a collider feature (variable) on path D, while Z and any of its successors are not included in S. Path D is unblocked among features (variables) X and Y otherwise. Two features (variables) X and Y are d-separated given S, if and only if, every path among feature (variables) X and Y are jammed by S.

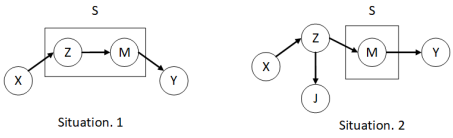

Fig. S1 Representation of D-Separation

Proposition S1(1) *Under the faithfulness assumption, the parents-child (PC) set combines the true PC of target T and some false-positive F.*

Proof First, demonstrate that PC_T has all true positives features and non-false negatives, i.e. $PC_T \subseteq PC$. Since feature X is a true PC of target T, both X and T are neighbors in the causal structure (CS). There is no set $Z \subseteq R$ exists that d-separates feature X from target T. Based on the first portion of **Theorem S1(1)**, $X \not\perp\!\!\!\perp T \mid Z$ will continuously dominant correct, provided the assumption of faithfulness. As a result, there is no way to exclude feature X from the equation. As a result, $PC_T \subseteq PC$ shows by discrepancies that PC set can contain some false-positives $PC \cup F$. Suppose that false-positives do not occur in PC_T, look at the situation depicted in Figure 1 (Can be find in the article of this Supplementary File), where node (feature) D depends on Target T if node C is present. Since S and R hold a sep-set A of size 1 and D's smallest sep-set. $C \cup S$, node S and R will be excluded from PC_T before D, and D will remain in PC_T since no sep-sets of D occur in PC_T. On the other hand, D is neither T's parent nor his child, contradicting the assumption. As a result, some false-positives are still present in PC_T. □

Proposition S1(2) *In a BN, R is a feature set, so by assuming that X is adjoining to Y, Y is adjoining to T, and X is not adjoining to T, e.g,* $(X \to Y \leftarrow T)$*. Two situations exist The first for finding true spouse, and the second is for finding a false-positive spouse:*

- *if* $Z \subseteq R \backslash \{X, Y, T\}$*, s.t.* $X \perp\!\!\!\perp T | Z$ *and* $X \not\perp\!\!\!\perp T | \{Z \cup Y\}$ *hold, X is a spouse of T associated with Y, i.e.,* $X \in SP_T\{Y\}$*.*
- *Once feature M enters* $SP_T\{Y\}$*, for any existing feature X in* $SP_T\{Y\} \backslash M$*,* $\exists Z \subseteq PC_T \cup SP_T\{Y\} \setminus X$*,* $X \perp\!\!\!\perp T \mid \{Z \cup Y\}$*, then X is false-positive and removed from* $SP_T\{Y\}$*.*

S2. Supplementary framework

Algorithm S1 : FSMB's framework

1. Initialization

   - Tmp = ∅;
   - Candidate features set $CPC_T = \varnothing$;
   - Non-parents-child features set Non-PC$_T = \varnothing$;
   - $PC_T = \varnothing$;
   - $SP_T = \varnothing$;
   - $FPF = \varnothing$;

2. Getting PC dependent on target T.

   - If X_i in Tmp maximizes association with target T, add X_i in CPC_T through **Proposition 1**. Otherwise, add X_i in Non-PC$_T$, and do not consider it again.
   - Until no more features are left in Tmp.

3. Find a spouse and get rid of any descendants who are not children.

   - According to **Proposition 2** and **3** if satisfied, for each $Y \in PC_T$, block path from T to $X \in R \setminus PC_T$, if not found, Remove Y, else X is a spouse according **Proposition 4(1)** and Y is a child.

4. Remove false-positive from the spouse set.

   - According to **Proposition 4(2)** if satisfied, for each $X \in SP_T\{Y\}$, trail-set = $PC_T \cup SP_T\{Y\} \setminus X$, if it finds X independent of T conditioning on trail-set, then remove X from $SP_T\{Y\}$.

5. Non-MB decedent in PC set.

   - According to **Proposition 2** and **3**, for each $X \in PC_T$, if X is conditional independent of T, given on the union of PC_T and $SP_T\{X\}$, then remove X from PC_T.
   - $PC_T = PC_T \setminus X$

6. Output.

   - $MB_T = PC_T \cup SP_T$

S3. Evaluation Metrics

S3.1. Evaluation Metrics for Benchmark BN

- **Precision:** This measure is calculated by dividing the number of true positives in an algorithm's output (for example, features (variables) belonging

to the true MB_T of a target feature (variable) in a directed acyclic graph) by the number of features (variables) in the algorithm's output.

- **Recall:** The true MB of a target feature in a directed acyclic graph is computed by dividing the number of true positives in output by the number of true positives in an input.
- **F1**-score $= 2 * \left[\frac{Precision * Recall}{Precision + Recall}\right]$: In the best-case scenario, F1 = 1 if precision and recall are both excellent. In the worst-case scenario, F1 = 0.

Table S1 Benchmark BN's summary

Bayesian Network	**Num-Features**	**Num-Edges**	**Maximum In/Out Degree**	**Min/Max** $\lvert PCset \rvert$
Child	20	25	2/7	1/8
Child3	60	79	3/7	1/8
Child10	200	126	2/7	1/9
Alarm	37	46	4/5	1/6
Insurance	27	52	3/7	1/9
Pig	441	592	2/39	1/41
Gene	801	972	4/10	0/11
Hailfinder	56	66	4/16	1/17
Barley	48	84	4/5	1/8
Mildew	35	46	3/3	1/5

S3.2. Evaluation Metrics for Real-world Datasets

- **Productiveness:** Num-Features represents the number of features that have been chosen. The prediction performance is based on the Cosine KNN and Linear SVM classifier and the mean accuracy (percentage) of the accurately categorized testing data that were previously unseen. The prediction accuracy of the two classifiers is being used to evaluate the efficacy of the various methods.
- **Efficiency:** The running time is presented as a metric for evaluating the efficiency of various algorithms.

Table S2 Benchmark real-world datasets summary

Dataset	**Num-Features**	**Num-Samples**
spect	22	267
wdbc	30	569
sylva	216	13086
colon	2000	62
reged1	999	500
bankruptcy	147	7063
lung	3312	203
medelon	500	2600
dexter	20,000	300