Songklanakarin J. Sci. Technol.
43 (6), 1823-1830, Nov. - Dec. 2021

http://www.sjst.psu.ac.th

Original Article

Floating search and conditional independence testing for causal feature selection♣

Rakkrit Duangsoithong* and Yuying Zhao

Department of Electrical Engineering, Faculty of Engineering,
Prince of Songkla University, Hat Yai, Songkhla, 90110 Thailand

Received: 22 December 2020; Revised: 1 November 2021; Accepted: 1 December 2021

Abstract

The curse of dimensionality and over-fitting problems are usually associated with high-dimensional data. Feature selection is one method that can overcome these problems. This paper proposes floating search and conditional independence testing as a causal feature selection algorithm (FSCI). FSCI uses mutual information with floating search strategy to eliminate irrelevant features and removes redundant features using conditional independence testing. The experimental demonstration is based on 8 datasets and the results are evaluated by number of selected features, classification accuracy, and complexity of the algorithm. The results are compared with the non-causal feature selection algorithms FCBF, ReliefF, and with the causal feature selection algorithms MMPC, IAMB, FBED and MMMB. The overall results show that the average number of features selected by the proposed FSCI algorithm (12.8) is below those with ReliefF (16.5) and MMMB (13) algorithms. According to the classification tests, FSCI algorithm provided the highest average accuracy (87.40%) among the feature selection methods tested. Moreover, FSCI can infer causality with less complexity.

Keywords: causal feature selection, floating search, redundant features, conditional independence testing, classification accuracy

1. Introduction

Feature selection selects a subset of features from the original features for classification, prediction, or data understanding (Lee & Jun, 2015). As the dimensionality of data increases feature selection becomes more important, while such high-dimensional data sets have become ubiquitous in various applications. Examples include various types of trade transaction data, gene expression data, WEB usage data, multimedia data, etc. (Wang, Irani & Pu, 2012). High dimensionality not only leads to higher computational costs and memory usage, but also reduces classification accuracy and easily causes overfitting.

Therefore, many feature selection methods have been proposed, and they fall into three categorie: filters, wrappers, and embedded methods (Kumar & Minz, 2014). Filter methods select a subset of features independently of any classifier. Wrapper methods use classifiers to evaluate selected feature subsets. The embedded methods automatically perform feature selection during training of the classifier. This research focuses on a filter method because it can preserve the original dataset, and is useful for discovering causality from the original data. Feature selection consists of two elements: a search strategy for feature subset generation, and an evaluation criterion for measuring relevance of the features.

The objective of a feature selection algorithm is to find a near optimal feature subset of the original features, without retaining irrelevant and redundant features (Yu & Liu, 2004). While many feature selection algorithms can eliminate both irrelevant features and redundant features, it is difficult to remove redundant features in some approaches, as with a causal feature selection algorithm.

♣Peer-reviewed paper selected from The 9th International Conference on Engineering and Technology (ICET-2021)
*Corresponding author
Email address: rakkrit.d@psu.ac.th

1.1 Search strategy of feature selection

In the feature selection algorithm, subsets of features are generated based on a search strategy. There are many ways of searching, such as complete search, heuristic search, and random search (Liu & Yu, 2005). In its dependence on data dimensionality, the complexity of complete search can be exponential, while for heuristic search the time complexity might be quadratic. The complexity of random search can be linearly related to the number of iterations (Liu & Setiono, 1996).

Complete search strategy finds the globally optimal subset of the original feature set, but is useful only when the feature dimensionality is low: when the original data has 10,000 features, the count of possible subsets is 2^{10000}, making a complete search impossible with such high-dimensional data. With dimensionality as indicator of problem size, complete search is NP-hard (Davies & Russell, 1994).

Heuristic search approach is more effective and feasible than a complete search. Depending on starting point of the search, it can be one of three types: forward search, backward search, and bidirectional search. Forward search starts from an empty set, and every round adds a feature that is optimal according to an evaluation criterion. The disadvantage is that features can be only added but not removed. Backward search is the opposite and starts from the full set of features, so that each time one feature is removed from the feature set, until the evaluation function value is optimized. The disadvantage is that the candidate feature subset only decreases without increasing. In bidirectional search the ideas of forward and backward searches are combined (Pudil, Novovičová & Kittler, 1994).

Random search strategy is different from complete and heuristic searches. It selects features randomly. A common shortcoming of random search is that it relies on random factors, and is difficult to reproduce experimentally. Commonly used random search approaches are Random Generation plus Sequential Selection (RGSS), Simulated Annealing (SA), Genetic Algorithms (GA), etc. (Wutzl *et al.*, 2007). Considering the advantages and disadvantages of the search algorithm types, this article focuses on using a heuristic search.

1.2 Non-causal feature selection

In the past decades, different feature selection methods have been proposed, such as methods based on distance (Kira & Rendell, 1992), correlation (Hall, 1999) and Chi-squared (Miller & Siegmund, 1982). Usually, Pearson correlation coefficient is used to calculate the correlation of a feature with a class. For a pair of variables (X, Y), the linear correlation coefficient 'r' is given by Equation 1:

$$r = \frac{\sum(X_i-\overline{X_i})(Y_i-\overline{Y_i})}{\sqrt{\sum(X_i-\overline{X_i})^2}\sqrt{\sum(Y_i-\overline{Y_i})^2}} \quad (1)$$

In 2004, Yu and Liu proposed Fast Correlation-Based Filter (FCBF) method. FCBF has relevance analysis and redundancy analysis. FCBF uses symmetrical uncertainty (SU) to remove irrelevant features. The *SU* equation is as follows:

$$\mathrm{SU(X,Y)} = 2\left[\frac{IG(X|Y)}{H(X)+H(Y)}\right] \quad (2)$$

SU has a range from 0 to 1. A larger *SU* means higher relevance of a feature to a class. The $IG(X|Y)$ denotes Information Gain of *X* after observing variable *Y*. *H(X)* is the entropy of variable *X*, and $H(X|Y)$ is the conditional entropy of *X* after observing values of another variable *Y*.

Then, FCBF removes a redundant feature F_i by the condition shown in Equation 3:

$$SU_{i,c} \leq SU_{j,c} \text{ and } SU_{i,c} \leq SU_{i,j} \quad (3)$$

Here $SU_{i,c}$ is the correlation between a feature F_i and the class C. $SU_{i,j}$ is the correlation between a pair of features F_i and F_j ($i \neq j$).

ReliefF algorithm only focuses on removing irrelevant features (Robnik Šikonja & Kononenko, 1997). ReliefF searches for its nearest neighbors according to the degree of discrimination of different class and weight features.

1.3 Causal feature selection

As an emerging filtering method, causal feature selection has attracted a lot of attention in recent years. By using causality, causal feature selection can naturally provide causal explanations about the relationship between features, or between features and classes, so as to better understand the mechanisms behind the data.

a) Peter-Clark (PC) algorithm is implemented in two stages: the skeleton learning stage and the direction inference stage. The PC algorithm starts from the fully connected graph, and the independence detection method is used to determine whether there is d separation. Then the redundant edges are removed one by one, and finally an undirected graph is formed. The direction of certain edges can be determined through the V structure or other local direction inference method (Spirtes & Glymour, 1991).

b) Max-Min Parents and Children (MMPC) algorithm was first described by Brown, Tsamardinos and Aliferis, (2004). Parents and Children refer to the fact that the algorithm identifies the parents and children of the class, assuming a Bayesian Network for all observed data. It will not recover the spouses of the children.

c) Incremental Association Markov Blanket (IAMB) was proposed by Tsamardinos and others in 2003. IAMB has two stages. In the first growth phase, nodes determined to depend on the target node are added to the Markov Blanket (MB) through independence testing. In the next shrinking phase, any node in the MB determined to be unrelated to the target node is deleted from the MB. Tsamardinos *et al.* proved that IAMB satisfies rationality under the assumption of faithfulness.

d) Forward-Backward with Early Dropping (FBED) algorithm is a recently proposed algorithm. The key element of FBED that makes it scalable to high dimensional data is that it removes insignificant features at every step. FBED 's final step includes a backward selection to remove falsely selected features. FBED algorithm is able to identify the full MB of the class (Borboudakis & Tsamardinos, 2019).

e) The Maximum and Minimum Markov Blanket (MMMB) algorithm attempts to overcome the data inefficiency problem of IAM, but it is still scalable under the premise of faithfulness. MMMB is also divided into two stages, but it uses a divide-and-conquer method, which is different from IAMB in using topology information. In the first stage (called MMPC), identify the parent and child nodes of a class, and then find the spouse nodes of class in the MMMB stage. Although not all nodes are determined to depend on the target node in the test, not all nodes can be included in the MB (Tsamardinos, Aliferis & Statnikov, 2003).

The main objective of proposed FSCI algorithm in this paper is to solve the problem of threshold setting and reduce the complexity of causal feature selection algorithm. In Section 2, proposed FSCI algorithm and redundant features definition are given. In Section 3, this paper introduces datasets and experimental set-up. Section 4 reports the experimental results and discusses their possible reasons. In Section 5, this paper concludes by summarizing experimental observations.

2. Theory and Proposed FSCI Algorithm

The block diagram of the proposed FSCI algorithm is shown in Figure 1. FSCI first uses mutual information (MI) as objective function to adopt SFFS search strategy for removing irrelevant features from the original dataset. Then, FSCI uses CI testing to remove redundant features. Among them, SFFS does not need to set a threshold and the algorithm complexity is lower than of complete search.

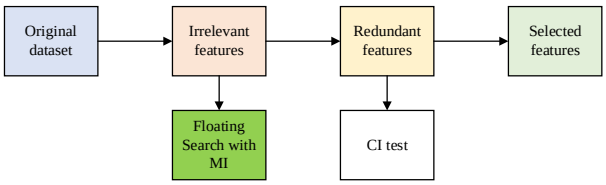

Figure 1. Block diagram of FSCI algorithm

As shown in Figure 2(a) flowchart, FSCI starts from an empty set, selects a feature F_i from the not yet selected features in each round, and optimizes the objective function MI after adding the feature F_i until no more features can be added. Next, FSCI removes a feature F_j from the selected features, and continues to eliminate from the subset until the mutual information is optimized. After that, the FSCI algorithm removes redundant features as described below. The flowchart of redundancy analysis is shown in Figure 2(b).

Proposed definition of redundant feature: a feature F_i or F_j is redundant to the class C, if and only if $MI(F_j, C|S) = MI(F_i, C|S)$ and $CI(F_i, C|S)$ or $CI(F_j, C|S)$ is independent. Here S is the current selected feature subset.

Definition of CI test: Let F_i, F_j and Z form a set of random features. Given Z, do a CI test between F_i and F_j, denoted as $F_i \perp F_j \mid Z$. That is, test whether conditional on given Z, F_i and F_j are independent.

3. Experiment Set-Up

3.1 Datasets

The experiment uses 8 binary class datasets collected from UCI machine learning repository (Dua, D., & Graff, C. ,2017).

- Lucas)LUng CAncer Simple set(has 2000 samples, 11 features and 2 classes. The class indicates having lung cancer or not. Lucas contains toy data generated artificially by causal Bayesian networks with binary variables.
- Abalone dataset contains 4177 samples and 8 features, and is divided into classes depending on whether the “number of rings” is greater than 10 or not.
- Spambase dataset is a classic spam email dataset from the UCI Machine Learning Repository. It has 4601 samples, 57 features and 2 classes.
- Sonar dataset contains 208 samples and 60 features. The dataset is divided into two classes: "R" if the object is rock; and "M" if it is mine.
- Ionosphere dataset needs to predict the atmospheric structure according to the radar echo of free electrons in the ionosphere. It contains 351 samples, 34 features and 2 classes)g for good, B for bad(.
- Hepatitis dataset contains 155 samples, 19 features, with “live” and “die” classes, and comes from the UCI Machine Learning Repository.
- Parkinson dataset has 197 samples and 23 features with two classes)healthy and Parkinson’s patient(.
- Lucap)LUng CAncer set with Probes(is LUCAS dataset with probes. LUCAP has 143 features, 2000 samples, and binary classes.

Each dataset is divided into 80% for training and 20% for testing, and the experiment is randomly repeated 5 times to calculate the average classification performance. The results are compared with 2 non-causal feature selection algorithms: FCBF and ReliefF; and with 4 causal feature selection algorithms: MMPC, IAMB, FBED and MMMB. Experiment uses CI test with significance value to provide output as Markov Blanket of the classes.

3.2 Evaluation criteria

The objective of this section is to describe evaluation of the algorithm. There are 4 evaluation criteria used in this paper: causal graph score compared with ground truth, number of selected features, classification accuracy, and algorithm complexity.

a) Causal graph score compared with ground truth

To reveal the causal graph of dataset, the Recall, Precision, and F1 score were selected as evaluation criteria. The definitions of these evaluation criteria are as follows:

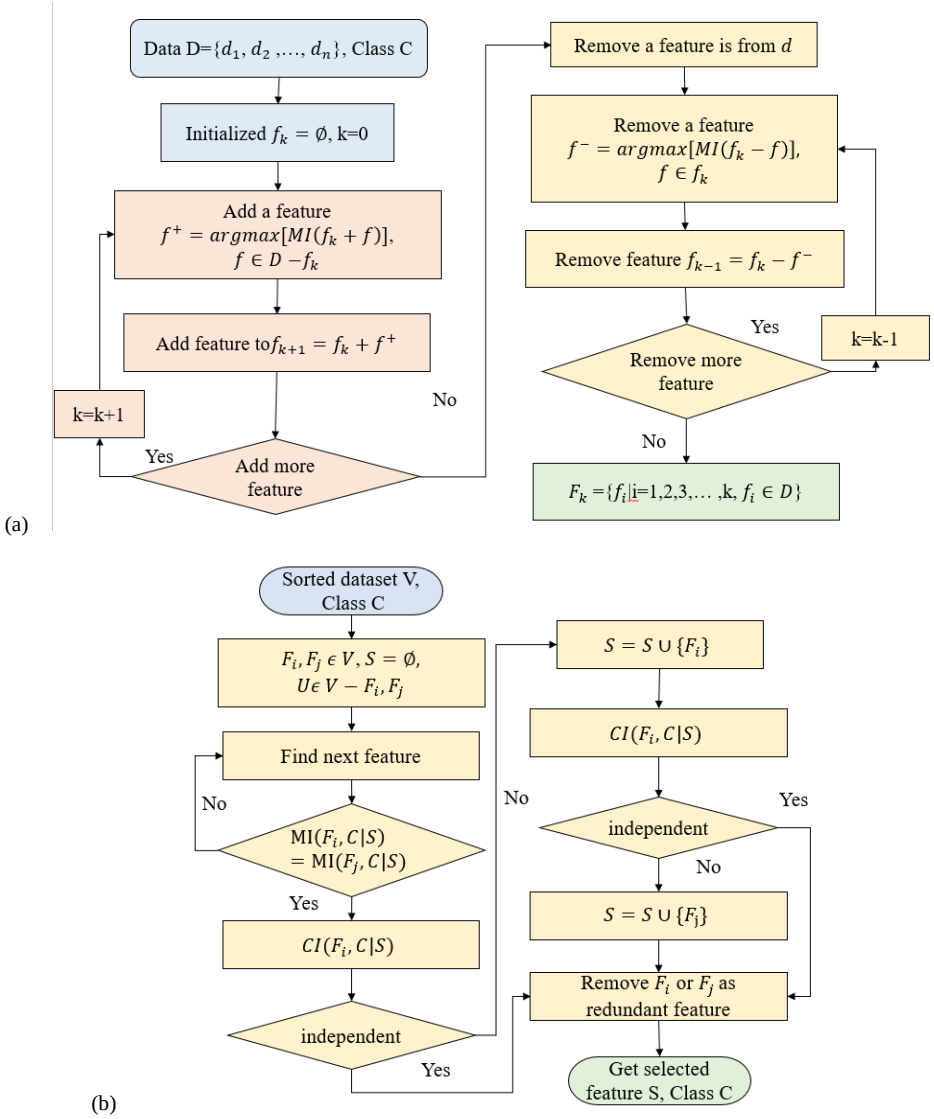

Figure 2. (a) Flowchart of FSCI algorithm: removing irrelevant features (b) Flowchart of FSCI algorithm: removing redundant features

$$Recall = \frac{Causal\ Inference\ Node \cap True\ Causal\ Node}{True\ causal\ node} \quad (4)$$

$$Precision = \frac{Causal\ Inference\ Node \cap True\ Causal\ Node}{Causal\ Inference\ Node} \quad (5)$$

$$F1 = \frac{2 * Recall * Precision}{Recall + Precision} \quad (6)$$

The real causal node is the causal relationship between each node in the real causal network, and the causal inference node is to infer the causal relationship between each node of the causal network through an algorithm. Recall refers to the ratio between the directionally correct edges in the inferred causal network and the actual causal network, and Precision refers to the ratio between the direction of the correct edge in the inferred causal network and the inferred network. *F1* is a combination of recall rate and accuracy, and is a standard evaluation measure. Generally, there are not many datasets that provide the ground truth causal graph. From the datasets of this experiment, only Lucas and Abalone datasets provide the ground truth causal graph. Therefore, the causal graph scores in the experiment are analyzed based on only these two datasets.

b) Number of selected features

This article summarizes and compares the number of features selected after each algorithm.

c) Classification accuracy

From the perspective of prediction capability, this paper compares the classification performances of the selected feature subsets of the different feature selection algorithms. Specifically, accuracy measure is examined, which can be calculated from a confusion matrix, as shown in Equations 7 and 8. Three well-known classifiers (k-nearest neighbors (kNN), Naive Bayes (NB) and Decision Tree (DT)) are used in the experiment. The set parameters for classification are as follows; kNN uses k = 5 with Euclidean distance, NB uses Gaussian Naïve Bayes algorithm, and DT forms splits by

using entropy. All the classifiers are implemented using sklearn library for Python language programs.

Confusion matrix =

Actual / predicted	Positive	Negative
Positive	a	b
Negative	c	d

(7)

$$Accuracy = \frac{a+d}{a+b+c+d} \quad (8)$$

d) Algorithm complexity

Proposed FSCI algorithm uses heuristic search first to remove irrelevant features, and the complexity is $O(N*n)$. The overall complexity of the algorithm is: $O(N*2^n*logN)$.

4. Results and Discussion

4.1 Results

a) Causal graph score compared with ground truth

The FSCI algorithm proposed in this paper is compared with PC algorithm as regards discovery of the causal graph of the dataset. The experimental results are shown in Table 1. The No. of node and True causal node in the table are observed from dataset ground truth graph.

From the results in Table 1, it can be seen that Recall, Precision and F1 score of the proposed FSCI algorithm are higher than those for the PC algorithm, with the Abalone dataset. The F1 value of the FSCI algorithm is 0.50, while the F1 value of the PC algorithm is only 0.09. However, in the Lucas dataset PC performed better than the proposed FSCI algorithm. The experimental results show that the FSCI algorithm can discover a causal graph from original dataset. In Abalone dataset the FSCI algorithm is better than the PC algorithm. Due to content limit, we only give an example causal graph comparison for the Lucas dataset in Figure 3.

b(Number of selected features

Table 2 shows the numbers of selected features as ultimately selected by each algorithm. To facilitate analysing the results, the averages were calculated. The bold number is the minimum number of features selected by each algorithm. The experimental results show that the number of features selected by FSCI algorithm is below those of ReliefF and MMMB algorithms. In causal and non-causal algorithms, causal feature selection gave the least features.

c(Classification accuracy

Classification accuracy results on 8 datasets are shown in Table 3. With kNN classifier the FSCI algorithm only achieved 76.16% classification accuracy on the Abalone dataset. The result is poorer than the classification accuracy using the original dataset. At the same time, other causal and

Table 1. Causal graph scores compared with ground truth

Dataset	No. of node	Ground truth node	Recall		Precision		F1	
			PC	FSCI	PC	FSCI	PC	FSCI
Lucas	12	12	**0.75**	0.33	**0.75**	0.57	**0.75**	0.35
Abalone	9	3	0.33	**0.66**	0.05	**0.4**	0.09	**0.5**

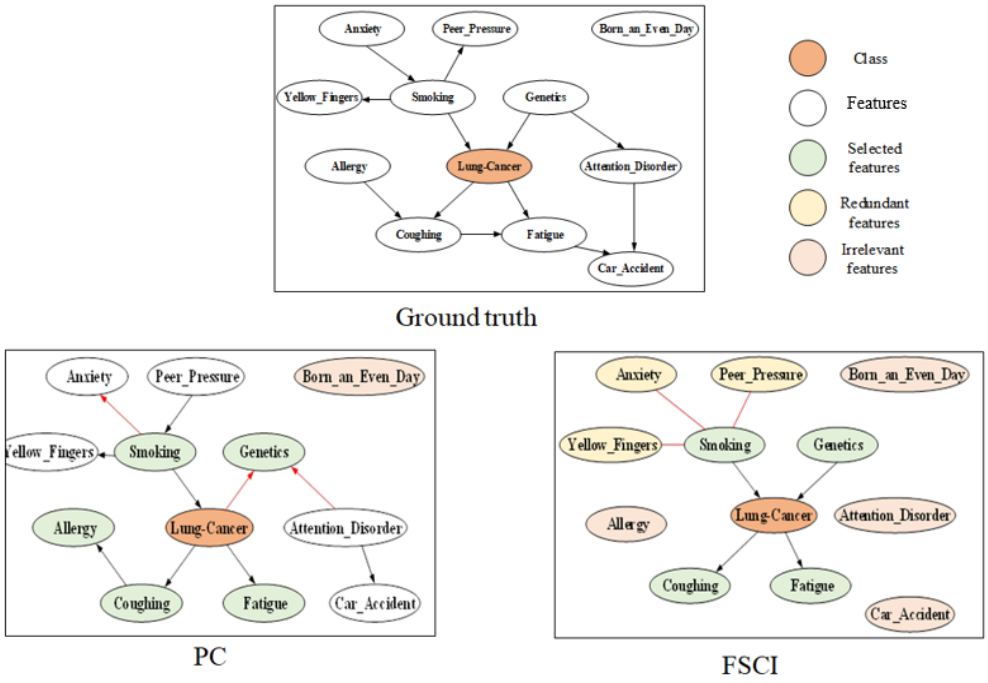

Figure 3. The causal graph comparison for Lucas dataset

Table 2. Numbers of selected features

Dataset	Original	Non-causal		Causal				Proposed
		FCBF	ReliefF	MMPC	IAMB	FBED	MMMB	FSCI
Lucas	11	6	5	5	6	6	8	**4**
Abalone	8	**4**	**4**	6	5	5	5	5
Spambase	57	**20**	34	23	27	30	22	29
Sonar	60	15	21	14	**11**	13	15	13
Ionosphere	34	9	15	**6**	7	8	7	7
Hepatitis	19	5	7	6	5	**3**	7	6
Parkinson	23	8	12	8	**4**	**4**	9	7
Lucap	143	23	34	27	23	**19**	31	32
Average	44.38	11.25	16.50	11.88	**11.00**	**11.00**	13.00	12.88

Table 3. Classification accuracies (%) for 3 classifiers on 8 datasets

Classifier	Algorithm		Lucas	Abalone	Spambase	Sonar	Ionosphere	Hepatitis	Parkinson	Lucap	Average
kNN	Original		91.61	77.17	93.25	70.00	85.00	89.08	**87.30**	79.07	84.06
	Non-causal	FCBF	**98.84**	88.97	95.60	76.50	87.25	81.27	76.90	80.09	85.68
		ReliefF	89.01	79.00	**97.58**	86.65	90.83	78.61	78.54	90.20	86.30
	Causal	MMPC	83.61	76.20	88.92	80.18	87.81	87.69	75.71	86.91	83.38
		IAMB	96.79	82.73	81.98	89.28	91.53	**90.98**	74.38	81.41	86.14
		FBED	91.97	81.11	85.64	92.06	82.91	75.76	87.09	**91.53**	86.01
		MMMB	81.13	**89.35**	80.97	**93.52**	**93.03**	83.39	86.00	87.22	86.83
	Proposed	FSCI	97.27	76.16	96.00	83.98	92.35	81.15	87.22	81.68	**86.98**
NB	Original		91.19	88.94	85.82	**89.66**	72.12	92.38	82.82	85.85	86.10
	Non-causal	FCBF	**98.74**	87.11	93.66	88.57	**94.50**	83.43	86.07	91.06	**90.39**
		ReliefF	95.46	86.27	**94.73**	79.18	91.67	77.20	**91.39**	88.48	88.05
	Causal	MMPC	91.69	85.12	90.66	82.12	83.22	**92.88**	85.22	83.91	86.85
		IAMB	95.25	89.92	92.22	82.11	76.65	80.41	90.06	91.75	87.30
		FBED	82.15	93.19	92.72	82.65	86.90	85.34	75.71	85.85	85.56
		MMMB	88.29	89.48	78.71	81.03	94.47	92.58	87.47	83.53	86.95
	Proposed	FSCI	92.76	**93.27**	92.89	82.66	89.18	75.21	86.85	**93.08**	88.24
DT	Original		92.15	84.71	87.75	82.84	79.21	82.72	82.28	81.99	84.21
	Non-causal	FCBF	**98.69**	82.05	83.64	70.80	80.97	82.12	86.34	79.28	82.99
		ReliefF	96.08	86.18	83.52	78.30	84.24	**92.43**	75.92	79.50	84.52
	Causal	MMPC	89.42	91.25	77.34	90.44	76.32	88.52	82.53	86.98	85.35
		IAMB	94.25	80.28	88.69	**93.60**	**89.71**	88.17	88.49	80.61	**87.98**
		FBED	85.90	**94.85**	83.01	81.64	84.67	75.49	87.90	**92.77**	85.78
		MMMB	80.70	85.15	85.89	75.65	79.53	87.06	**90.47**	81.97	83.30
	Proposed	FSCI	92.33	87.22	**94.56**	88.82	85.24	81.11	90.44	76.06	86.97

non-causal feature selection algorithms also gave low classification accuracies on these data. On the Ionosphere dataset, compared with the original data, the FSCI algorithm achieved the best results. On the Spambase and Sonar datasets, compared with the original data, FSCI did not improve the classification accuracy with NB classifier. The classification accuracy of the FSCI algorithm on the Abalone dataset is 93.27%, which is better than the classification accuracy using original data.

Both non-causal feature selection algorithm and causal feature selection algorithms can effectively improve the classification accuracy of the original dataset. The proposed FSCI algorithm is slightly inferior to non-causal feature selection in KNN classification results, but this algorithm is superior to the other causal feature selection algorithms. In NB classification results, the proposed FSCI algorithm is second only to iamb algorithm and superior to other causal feature selection algorithms. In the classification results of DT classifier, the FSCI algorithm proposed in this paper was also better than using the original dataset. In addition to MMMB algorithm, other non-causal and causal feature selection algorithms can effectively improve the classification accuracy of data.

Figure 4 shows the average results of the three classifiers in Table 4. As can be seen from Figure 4, in these 8 data sets, both non-causal and causal feature selection algorithms effectively improved the accuracy of the original classification. The proposed FSCI algorithm is superior to the other non-causal and causal feature selection algorithms tested.

d) Algorithm complexity

The algorithm complexity comparison is shown in Table 5. FCBF performs forward and backward heuristic search and its complexity is $O(M*N*logN)$. ReliefF algorithm complexity is linear. As for causal feature selection algorithms, the complexity depends on the number of features

Table 4. Overall classification accuracies for 3 classifiers on 8 datasets (%)

Classifier		kNN	NB	DT	Average
Original		84.06	86.10	84.21	84.79
Non-causal	FCBF	85.68	**90.39**	82.99	86.35
	ReliefF	86.30	88.05	84.52	86.29
Causal	MMPC	83.38	86.85	85.35	85.19
	IAMB	86.14	87.30	**87.98**	87.14
	FBED	86.01	85.56	85.78	85.78
	MMMB	86.83	86.95	83.30	85.69
Proposed	FSCI	**86.98**	88.24	86.97	**87.40**

Table 5. Algorithm complexity comparison of tested algorithms

Algorithm		Complexity	Average time from 8 datasets (min)	Remark
Non-causal	FCBF	$O(M*N*logN)$	9.71	M = number of samples
	ReliefF	$O(M*N*n)$	8.28	N = number of original features
Causal	MMMB	$O(N*2^{MB})$	27.57	n = number of selected features
	MMPC	$O(N*MB^{l})$	22.29	MB = number of features in Markov Blanket of Class
	IAMB	$O(N*2^{MB})$	26.86	k = number of iteration
	FBED	$O(N*MB^{k})$	26.59	l = number of condition feature set size
Proposed	FSCI	Relevancy: $O(N*n)$ Redundancy: $O(N*2^{n}*logN)O$	21.35	

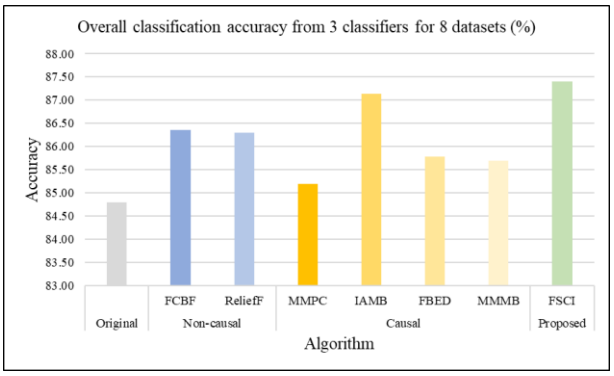

Figure 4. Overall classification accuracy using 3 classifiers on 8 datasets (%)

in MB. In order to compare the algorithm complexity, time is tested in the feature selection phase. The experiment was performed on Windows 10 operating system with a laptop computer that has Intel i5 CPU, 8GB RAM and as hard drive a 256GB SSD. The complexity comparison results in both Big O notation and computational time for each algorithm are shown in Table 5.

4.2 Discussion

4.2.1 Feature selection discussion

FSCI chooses more features than some other algorithms because FSCI uses a heuristic search strategy. When irrelevant features are deleted, the algorithm may stop after other algorithms. Among the causal and non-causal algorithms, the non-causal ones chose more features, because ReliefF algorithms only analyses the relevance of features. In experimental results based on 8 datasets, the difference in number of selected features between Spambase and Sonar datasets affected the average result. The reasons are that Spambase dataset is a sparse dataset, and the Sonar dataset is a highly correlated dataset.

4.2.2 Classification results discussion

With KNN classifier the FSCI algorithm achieved the best classification accuracy. This is because FSCI is the most stable in the 8 datasets. Although the other algorithms can get the best classification result in a single dataset, the white line of different algorithms is different. For example, sonar and ionosphere datasets achieved better classification results with MMMB algorithm, but the experimental results show that the proposed FSCI algorithm had better universality or consistency in performance.

With NB classifier, the results of sonar and Spambase datasets differ. One reason is that the sonar data has high similarity in elements. MI is the standard function in the experiment. These features will affect the NB classification accuracy of FSCI algorithm. Another reason is that Spambase dataset is a sparse dataset. In the process of removing irrelevant features, the differences in dataset will affect the analysis of the second part of redundant features.

For DT classifier, the proposed FSCI algorithm effectively improved the original results according to the experimental results on 8 datasets. The reason is that in the process of DT training, feature selection is carried out.

5. Conclusions

This paper proposed a causal feature selection algorithm named FSCI. FSCI uses heuristic search strategy to avoid setting a threshold and this reduces the complexity of the algorithm. From the experimental results on 8 datasets, the FSCI algorithm can reveal causality while effectively

reducing the number of features and improving classification accuracy. Moreover, the proposed FSCI algorithm also provided better average accuracy compared with both non-causal and causal prior feature selection algorithms.

Acknowledgements

This research was supported by grants Thailand Science Research and Innovation (TSRI). The fundamental fund 2564, ref: ENG6405014S and Scholarship Contract for Engineering Graduate Students Graduate Study Level Master Degree, The Department of Electrical Engineering, Faculty of Engineering, Prince of Songkla University, PSU-Master Scholarship.

References

Borboudakis, G., & Tsamardinos, I. (2019). Forward-backward selection with early dropping. *The Journal of Machine Learning Research, 20*(1), 276-314. doi: 10.5555/3322706.3322714

Brown, L. E., Tsamardinos, I., & Aliferis, C. F. (2004, September). A novel algorithm for scalable and accurate Bayesian network learning. *Medinfo*, 711-715. doi: 10.3233/978-1-60750-949-3-711

Davies, S., & Russell, S. (1994, November). NP-completeness of searches for smallest possible feature sets. *AAAI Symposium on Intelligent Relevance*, 37-39. doi:10.1.1.57.3452

Duangsoithong, R., & Windeatt, T. (2011). Hybrid correlation and causal feature selection for ensemble classifiers. In Ensembles in Machine Learning Applications (pp. 97-115). Berlin, Heidelberg, Germany: Springer. doi:10.1007/978-3-642-22910-7_6

Dua, D., & Graff, C. (2017). UCI Machine learning repository. Retrieved from http://archive.ics.uci.edu/ml.

Hall, M. A. (1999). Correlation-based feature selection for machine learning. Retrieved from https://www.cs.waikato.ac.nz/~mhall/thesis.pdf

Kumar, V., & Minz, S. (2014). Feature selection: A literature review. *SmartCR, 4*(3), 211-229. doi:10.6029/smartcr.2014.03.007

Kira, K., & Rendell, L. A. (1992). A practical approach to feature selection. *Machine Learning Proceedings*, 249-256. doi:10.1016/B978-1-55860-247-2.50037-1

Lee, J., & Jun, C. H. (2015). Classification of high dimensionality data through feature selection using Markov blanket. *Industrial Engineering and Management Systems, 14*(2), 210-219. doi:10.7232/iems.2015.14.2.210

Liu, H., & Yu, L. (2005). Toward integrating feature selection algorithms for classification and clustering. *IEEE Transactions on Knowledge and Data Engineering, 17*(4), 491-502. doi:10.1109/TKDE.2005.66

Liu, H., & Setiono, R. (1996, July). A probabilistic approach to feature selection-a filter solution. *ICML, 96*, 319-327. doi:10.1.1.36.1270

Miller, R., & Siegmund, D. (1982). Maximally selected chi square statistics. *Biometrics*, 1011-1016.

Pudil, P., Novovičová, J., & Kittler, J. (1994). Floating search methods in feature selection. *Pattern Recognition Letters, 15*(11), 1119-1125. doi:10.1016/0167-8655(94)90127-9

Robnik-Šikonja, M., & Kononenko, I. (2003). Theoretical and empirical analysis of ReliefF and RReliefF. *Machine learning, 53*(1), 23-69. doi:10.1023/A:1025667309714.

Spirtes, P., & Glymour, C. (1991). An algorithm for fast recovery of sparse causal graphs. *Social Science Computer Review, 9*(1), 62-72. doi:10.1177/089443939100900106

Tsamardinos, I., Aliferis, C. F., Statnikov, A. R., & Statnikov, E. (2003, May). Algorithms for large scale Markov blanket discovery. *FLAIRS Conference, 2*, 376-380. doi:10.1.1.13.5894

Tsamardinos, l., Brown, L. E., Aliferis, C. F., & Moore, A. W. (2006). The Max-Min Hill-Climbing Bayesian network structure learning algorithm. *Machine Learning, 65*(1), 31–78. doi:10.1007/s10994-006-6889-7

Tsamardinos, I., Aliferis, C. F., & Statnikov, A. (2003, August). Time and sample efficient discovery of Markov blankets and direct causal relations. *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 673-678. doi:10.1145/956750.956838

Wang, D., Irani, D., & Pu, C. (2012, October). Evolutionary study of web spam: Webb spam corpus 2011 versus webb spam corpus 2006. *Proceeding of the 8th International Conference on Collaborative Computing: Networking, Applications and Worksharing (CollaborateCom) (pp. 40-49), IEEE*. doi:10.4108/icst.collaboratecom.2012.250689

Wutzl, B., Leibnitz, K., Rattay, F., Kronbichler, M., Murata, M., & Golaszewski, S. M. (2019). Genetic algorithms for feature selection when classifying severe chronic disorders of consciousness. *PloS one, 14*(7), e0219683. doi:10.1371/journal.pone.0219683

Yu, L., & Liu, H. (2004). Efficient feature selection via analysis of relevance and redundancy. *Journal of Machine Learning Research, 5*(Oct), 1205-1224. doi: 10.1.1.71.5055