# Graph-based Information-Theoretic Approach For Unsupervised Learning

A project report submitted for the partial fulfillment of the
Bachelor of Technology Degree in
**Computer Science & Engineering under**
**Maulana Abul Kalam Azad University of Technology by**

**Sagarika Saroj Kundu**
Roll No: 10400115135, Registration Number: 151040115135
**&**
**Pritika Sarkar**
Roll No: 10400115119, Registration Number: 151040115119

Academic Session: 2015-2019

Under the Supervision of
Prof. Amit Kumar Das


**Department of Computer Science and Engineering Institute of Engineering & Management**
Y-12, Salt Lake, Sector 5, Kolkata, Pin 700091, West Bengal, India

Affiliated To


**Maulana Abul Kalam Azad University of Technology**
BF 142, BF Block, Sector 1, Kolkata, West Bengal 700064

**May 2019**

INSTITUTE
OF ENGINEERING & MANAGEMENT
श्रद्धावान लभते ज्ञानम्
Salt Lake Electronics Complex, Kolkata - 700091, WB, INDIA

Phone : (033) 2357 2969/2059/2995
(033) 2357 8189/8908/5389
Fax : 91 33 2357 8302
E-mail : director@iemcal.com
Website : www.iemcal.com

# CERTIFICATE

## TO WHOM IT MAY CONCERN

This is to certify that the project report titled **"Graph-based Information-Theoretic Approach For Unsupervised Learning"**, submitted by **Sagarika Saroj Kundu, Roll No: 10400115135, Registration Number: 151040115135**, **Pritika Sarkar, Roll No: 10400115119, Registration Number: 151040115119**, students of **Institute of Engineering & Management** in partial fulfillment of requirements for the award of the degree of **Bachelor of Technology in Computer Science & Engineering**, is a bonafide work carried out under the supervision of **Prof. Amit Kumar Das** during the final year of the academic session of 2015-2019. The content of this report has not been submitted to any other university or institute for the award of any other degree.

It is further certified that the work is entirely original and the performance has been found to be satisfactory.

**Prof. Amit Kumar Das**
Assistant Professor
Department of Computer Science and Engineering
Institute of Engineering & Management

**Prof.(Dr.) Himadri Nath Saha**
H.O.D.
Department of Computer Science and Engineering
Institute of Engineering & Management

**Prof.(Dr.) Amlan Kusum Nayak**
Principal
Institute of Engineering & Management

# INSTITUTE OF ENGINEERING & MANAGEMENT


## DECLARATION
## FOR NON-COMMITMENT OF PLAGIARISM

We, Sagarika Saroj Kundu, Pritika Sarkar, students of B.Tech in the Department of Computer Science and Engineering, Institute of Engineering & Management have submitted the project report in partial fulfillment of the requirements to obtain the above noted degree. We declare that we have not committed plagiarism in any form or violated copyright while writing the report and have acknowledged the sources and/or the credit of other authors wherever applicable. If subsequently it is found that we have committed plagiarism or violated copyright, then the authority has full right to cancel/reject/revoke our degree.

Name of the Student: SAGARIKA SAROJ KUNDU

Full Signature: _____

Name of the Student: PRITIKA SARKAR

Full Signature: _____

Date: _____

# Contents

**1. Abstract…….. 1**

**2. Acknowledgement…….. 2**

**3. Keywords…… 3**

**4. List of figures……... 4**

**5. List of tables…….. 5**

**6. Introduction…….. ..6-7**

**7. Related Work……..8-9**

**8. Basic Underlying Concept**

**8.1 Future Relevance…… 9-10**

**8.2 Future Redundancy…… 10-11**

**9. Proposed Approach……... ..11-13**

**10. Algorithm……14-16**

**11. Illustration…….. 17-19**

**12. Graph produced by GITAUFS on UCI data sets……20-22**

**13. Experiments and Outcome**

**13.1 Summary of Outcome……... 23**

**13.1.1 Comparison of Silhouette width value…… 23-24**

**13.1.2 Comparison of Feature Reduction…….. 24-26**

**13.1.3 Comparison of Execution Time……... 26-27**

**13.2 Overall Comparison Performance…….. 27-28**

**14. Conclusion…….. 29**

**15. References…… 30-31**

# 1. Abstract

There is a critical need for feature selection today with the increase in the number of high dimensional data sets. Selecting a subset consisting of important features not only reduces the execution time, but also increases the predictive ability of the machine learning model. In this paper, a novel graph-based feature selection algorithm for unsupervised learning has been proposed. Unlike many of the feature selection algorithms which use correlation as a measure of dependency between features, the proposed algorithm derives feature dependency using information-theoretic approach. The proposed algorithm Graph-based Information Theoretic Approach for Unsupervised Feature Selection (GITAUFS) generates multiple minimal vertex covers (MVC) of the feature graph and evaluates them to find the most optimal one in context of the learning task. In our experimental setup comprising 13 benchmark data sets, GITAUFS has shown a 10% increase in the silhouette width value along with a significant feature reduction of 90.62% compared to the next best performing algorithm.

# 2. Acknowledgements

We must not forget to acknowledge everyone who has provided constant support to us during our B.Tech course. First and foremost, we would like to express sincere gratitude to our supervisor **Prof. Amit Kumar Das** for his continuous support and motivation in fueling the pursuance of carrying out this project endeavor. Without his guidance and persistent encouragement, this project work would not have been possible. He has been a tremendous mentor for us throughout this academic journey. Many of his academic advises about our career growth have been priceless.

We would like to convey sincere gratitude to **Prof. Himadri Nath Saha** for providing us constant inspiration to stand firm against several setbacks throughout the course. Additionally, we would like to thank all the technical, non-technical and office staffs of our department for extending facilitating cooperation wherever required. We also express gratitude to all of our friends in the department for providing the friendly environment to work on the project work.

We would also like to thank our Director **Prof. Satyajit Chakraborti** for providing us an outstanding platform in order to develop our academic career. In addition, we also preserve a very special thankful feeling about our Principal **Prof. Amlan Kusum Nayak** for being a constant source of inspiration.

A special thank is due to our family. Words cannot express how grateful we are to our parents for all the sacrifices that they have made while giving us necessary strength to stand on our own feet.

Finally, we would like to thank everybody who has provided assistance, in whatever little form, towards successful realization of this project but with an apology that we could not mention everybody's name individually.

# 3. Keywords

| Keywords | Descriptions |
| --- | --- |
| GITAUFS | Graph-based Information-Theoretic Approach For Unsupervised Feature Selection. |
| Feature Selection | The process of selecting a subset of relevant features for use in model construction. |
| Mutual Information | A quantity that measures how much one random variable tells us about other. |
| MVC (Minimal Vertex Cover) | A vertex cover of an undirected graph is a subset of its vertices such that for every edge(u,v) of the graph either ‘u’ or ‘v’ is in vertex cover. |

# 4. List of Figures


Fig 1: Illustration for ‘mfeat’ dataset.
Fig 2: Flowchart of the proposed algorithm.
Fig 3: Illustration for GITAUFS.
Fig 4: Wins/Ties for silhouette width value.
Fig 5: Illustration of ‘apndcts’ dataset.
Fig 6: Illustration of ‘btissue’ dataset.
Fig 7: Illustration of ‘cleave’ dataset.
Fig 8: Illustration of ‘ecoli’ dataset.
Fig 9: Illustration of ‘glass’ dataset.
Fig 10: Illustration of ‘ILPD’ dataset.
Fig 11: Illustration of ‘pima’ dataset.
Fig 12: Illustration of ‘mfeat’ dataset.
Fig 13: Illustration of ‘sonar’ dataset.
Fig 14: Illustration of ‘wine’ dataset.
Fig 15: Illustration of ‘wbdc’ dataset.
Fig 16: Illustration of ‘vehicle’ dataset.
Fig 17: Illustration of ‘wiscon’ dataset.
Fig 18: Performance for silhouette width.
Fig 19: Feature Reduction.

# 5. List of Tables

Table 1: Description of UCI data sets.
Table 2: Performance of silhouette width.
Table 3: Percentage feature reduction.
Table 4: Execution time (in seconds).
Table 5: Comparison of different algorithms.
Table 6: Summary of performance (Silhouette width).

# 6. Introduction

## Motivation & objective of the thesis

Feature selection is a critical area of research focus, especially in domains having large number of attributes. Such domains include processing of internet documents, customer review analysis and interpretation of data from genomic project to name a few. It is advantageous as it allows us to design cost-effective machine-learning models as well as reduce model execution time in high-dimensional data sets. An unsupervised machine learning algorithm draws inferences from data without having any known labeled responses. In unsupervised learning, the grouping of unlabeled data instances needs to be done based on some specific set of statistical measures. In this context, feature selection is a combinatorial optimization problem where the objective is to find an optimal feature subset from the entire feature set such that no information is effectively lost from the data in question. The features need to be selected on the basis of how informative they are and contribute to the specific unsupervised learning task. Also, it is important to consider how redundant the features are based on their similarity with other features. In general, approaches adopted for feature selection include wrapper approach, filter approach, embedded approach and hybrid approach. The wrapper approach algorithmically learns and determines the optimal subset of features based on prediction accuracy. It is very accurate but is prone to over fitting. On the contrary, in filter approach, statistical measures are used in place of learning algorithms, making it suitable for high-dimensional data sets. The embedded approach chooses the optimal feature subset during training. The hybrid approach exploits the benefits of both filter and wrapper approaches. In feature selection, it is important to adopt a suitable similarity measure in order to evaluate inter-feature similarity. It is also important to establish feature relevance in order to decide which features should be selected as part of the final subset. A feature is considered irrelevant if it contributes almost no information and is thus insignificant for tasks such as clustering of

given data instances; it can be removed if it does not significantly contribute to the learning task. Feature relevance can be measured using measures like joint mutual information, symmetrical relevance, entropy. Features whose contribution is nearly same as one or more other features are considered to be potentially redundant. Such similarities can be measured using different measures like Fisher score, Pearson's correlation, mutual information, etc. The algorithm proposed in this paper uses graph-theoretic approach to represent the combinatorial relationship between different features of an input data set. This allows visualizing the degree of inter-feature similarity and hence featuring redundancy. It also derives feature subset by using graph-theoretic principles of finding sub-graphs from graphs. In our algorithm, after rejecting features based on their entropy, the features are represented as graph based on their mutual information statistics. Then the optimal subset of features is obtained by using the 2-approximation algorithm of minimal vertex cover. The use of mutual information to determine the association between features is supported by its ability to measure the general dependence between features and obtain a complete characterization of symbolic as well as numeric sequences and features, as opposed to classical methods like PCA (for dimensionality reduction), KMeans or measures like Pearson's correlation that are able to capture linear relations at best. Furthermore, unlike mutual information, these classical methods are sensitive to scale effects and necessitate the use of pre-processing measures like normalization prior to designing models.

# 7. Related work

Feature selection is a topic of great interest for research. Various methods have been adopted by researchers for feature selection. However, very few papers have proposed graph-based feature selection approach. In most of the works in this area features are represented as vertices of a graph and the edges represent the relationship between features like inter-feature similarity, etc. Graph theoretic principle for deriving sub-graphs of graphs is used in graph-based feature selection.

In a related work, features have been mapped as vertices and weighted edges represent inter-feature mutual information. In this paper, the first subset of features is selected to minimize redundancy by selecting the densest sub graph. In the next stage, final feature subset is obtained by feature clustering around the non-redundant features.

In another work, community detection algorithm has been used. The features represented as vertices in graph are clustered using community detection algorithm. This reduces the feature subset. Then features from each cluster are selected iteratively if they have value more than a threshold until there exists no feature having value more than the threshold.

Lu et al. proposed a feature selection algorithm where the subsets of the available features are chosen using the same criteria as PCA and applied it to tasks like face-tracking and content based image retrieval. In another feature selection algorithm, a Laplacian score is calculated which determines the locality pre-serving power. The local structure of data space is the focus rather than the global structure. A nearest neighbor graph is plotted to evaluate the local geometric structure. Feature association mapping has been used as an underlying concept in which applies to both supervised and unsupervised learning. The approach uses maximal independent set and graph theoretic approach of minimal vertex cover to derive the final set of features. Another work has used an ant colony based search process, a nature inspired optimization algorithm. The

approach also uses graph-based modeling of the feature set. Hill-climbing based approach with graphical representation of the input data set.

In a recent work, a graph encoder based technique for feature selection has been adopted. Different works involving graph-based feature selection involve finding the feature-to-class or feature to feature relation. The information contributed by a feature for the learning task has been a focus in the work by. The approach uses multidimensional interaction information (MII) as the selection criteria for features into final feature subset. MII is measured based on the mutual information between the feature subset selected and the target class. Dominant set clustering is used to group the feature vectors. The final feature subset is made by selecting the features from the dominant-set using the MII criteria. This ensures a final reduced feature set having maximum about the class.

# 8. Basic underlying concepts

In this section, the concepts required for GITAUFS have been outlined. In the proposed algorithm, there are three stages. First, the irrelevant features are excluded from candidate feature list. Then the associations among selected features are measured through the use of well-established algorithms and experimentally chosen threshold values to determine the ones which are redundant and need to be excluded.

## 8.1 Feature relevance

The entropy of a feature allows us to quantify the average information contributed by it through measurement of the unpredictability of the state; lower probability implies higher information content. This information I is expressed as:

$$I(x_i) = log_2 \frac{1}{P(x_i)} = -log_2 P(x_i) \qquad (1)$$

According to Shannon, entropy H of feature $X = \{x_i\}_{i=1...n}$ is defined as:

$$H(X) = \sum_{i=1}^{n} P(x_i)I(x_i) = -\sum_{i=1}^{n} P(x_i)log_2P(x_i) \qquad (2)$$

Entropy value is used to remove features that carry little information relevant to learning. Since this information measure depends only on the probability distribution of a random variable rather than on its actual values, it has been widely used in feature selection. When $P(X)$ is uniformly distributed, the entropy of X is maximal, meaning that it has highest level of unpredictability, and, thus, the maximal information content. Since entropy reflects the amount of disorder of a system, many methods employ some form of such a measure in the objective function of clustering. For each feature x, the information $\overline{I_{\varepsilon-\{x\}}}$ contributed by the entire set of features is measured i.e. $\varepsilon$ minus x using Eqn.2. All the features are thus ranked according to the metric $\bar{I}$ and among these ones having higher value are considered to be potentially irrelevant.

## 8.2 Feature redundancy

GITAUFS uses mutual information of candidate features as the starting component of a pruning algorithm to strip away redundant features from the initial subset of available ones. While measures like correlation allow us to understand association between different features and dimensionality reduction techniques like PCA are sufficient for simple distributions of patterns belonging to different classes, feature selection using these fail in the case of classification or clustering tasks with complex decision boundaries, since they consider only linear relations between features. On the other hand, mutual information can measure arbitrary relations between variables and is independent of the transformation done on them and is suitable for assessing their "information content" for a robust estimation of redundancy. Hence, if two features $f_1$ and $f_2$ are strongly similar, or one contributes significant amount of information about the other, then their MI is large. If both $f_1$ and $f_2$ kept in the feature set while clustering, then the results obtained will not be different from the result obtained when either of these features is used.

The mutual information M for a pair of features(X,Y), is defined as

$$M(X,Y) = \sum_{x \epsilon X} \sum_{y \epsilon Y} P(x,y) log \frac{P(x,y)}{P(x)P(y)} \qquad (3)$$

Here $P(x, y)$ represents the joint probability. In particular, Eqn. 3 measures how much information is communicated, on average, in one random variable about another. Features having mutual information of 0 are considered statistically independent. Based on this, matrix MI is constructed.

# 9. Proposed Approach

The proposed approach represents each feature in the data set as a vertex in the graph. The main stages of GITAUFS algorithm include selecting the relevant features, identifying the potentially redundant features and finding the most optimal feature subset based on silhouette width value after evaluating $\gamma$ minimal vertex covers derived from the potentially redundant features.

- Step 1: Highlighting the irrelevant (or least informative) features
  Entropy can determine the information contribution of feature in the data set for a learning task. High entropy for a particular feature signifies high information contribution by the feature for the clustering of data. Thus, features with high entropy need to be considered for further evaluation as they are potential candidates of relevant features. In this stage, the features having high entropy are marked "green" and the rest are marked "red" as they are irrelevant (due to low entropy or low information contribution). The irrelevant features are thus highlighted in this stage.

- Step 2: Highlighting potentially redundant features.
  The features marked "green" in the previous stage are evaluated in this stage for potential similarity between them. Mutual Information between features is calculated. The similarity matrix ($MI_{mat}$) generated holds each cell corresponding to the Mutual Information value between features respective to the column and row for that cell. The

mean of the similarity matrix is used as the threshold to indicate how high or low is the similarity between two features. Feature pairs having mutual information value greater than the mean are considered similar and below or equal to the mean are considered as dissimilar.

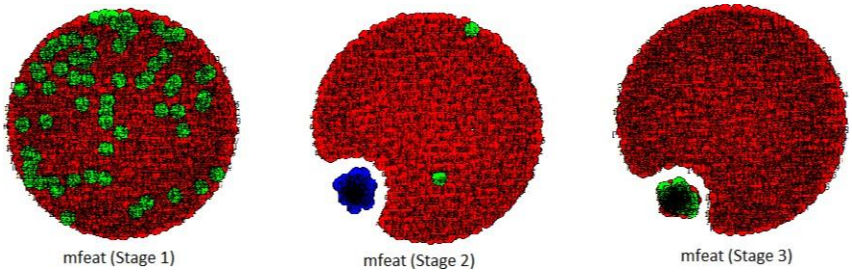


Fig. 1: Illustration for 'mfeat' dataset

The adjacency matrix is then made using the matrix having value of mutual information between features and the threshold value (mean of the matrix). The cells having mutual information value lesser than or equal to the mean are made 0, and rest are made 1. The leading diagonal of the matrix is made 0 as it represents the mutual information of a feature to itself. The similar features are represented in the graph using the adjacency matrix. These features are colored "blue" in the feature graph and they contain the potentially redundant features. The rest of the features initially marked "green" are candidates for final subset of features without further evaluation because they have no similarity with any other feature as well as high information contribution (due to high entropy). These features are hence kept "green" in color. This stage thus marks the similar features with edges.

- Step 3: Highlighting the final subset of features. In this stage, a subset of features is selected from the connected features marked "blue" as a representative of the whole set. This ensures that the information content of the features remains same even though the numbers of features reduce. Minimal Vertex Cover (MVC) algorithm has been used for this purpose. This algorithm determines a subset of vertices whose edges are incident to at least one the vertices select in the subset. MVC is a 2-approximation minimizing algorithm and hence the number of features in the subset will be

minimal. However the minimum subset is not always determined by the algorithm. Finding the minimum subset of features is a NP hard problem. The proposed approach however tries to find the best subset among multiple minimal vertex covers given by MVC algorithm. The subsets are ranked based on the entropy (or information contribution) of the features. The top $\gamma$ subsets having higher entropy value are selected. These subsets are then evaluated based on the silhouette width value. The subset giving the highest silhouette value is the final subset of selected features. The features in this subset are then marked "green" and the rest of the "blue" vertices are colored "red".

By the conclusion of Step 3, the final set of features GITAUFS algorithm for the data set is ready. The features represented by "red" colored vertices are the rejected features and the final subset of selected features is represented in "green". All the three stages of GITAUFS generate graph which convey important information about the features in the respective data set.

# 10. Algorithm

**Input:** N-dimensional data set $D_N$ having original feature set F = $f_1$, $f_2$, ...., $f_N$ .
$\alpha$ - Relevance threshold.
$\gamma$ - Number of MVCs to be compared for final feature set.
**Output:** Optimal feature subset $F_{opt}$ .

**Begin**
/*Stage 1: Entopy of the features is calculated and only top contributing features are coloured green.*/
1: For i = 1 to N
2: $E_i$ = ENTROPY($f_i$)
3: Next
4: SORT(E)
5: For i = 1 to (α%)N
6: color($f_i$) = "green"
7: Next
8: For i = ((α%)N)+1 to N
9: color($f_i$) = "red"
10: Next
11: $g_1$ = generateGITAUFS(F)

/*Stage 2: The similar features among the possible optimal feature set (which were marked "green" in the previous stage) will be coloured "blue" in this stage.*/
12: F' = {x: x ⊆ F and color(x) = "green"}
13: $MI_{mat}$ = mutual-information($D_N$[F'])
14: For I = 1 to |F'|
15: For j = 1 to |F'|
16: If($MI_{mat}$ > mean($MI_{mat}$)) & (i ≠ j)) then
17: add-edge($F_i$, $F_j$, $g_1$)
18: color($F_i$) = "blue"
19: color($F_j$) = "blue"
20: End If
21: Next
22: Next

/*Stage 3: The Minimal vertex cover algorithm gives all possible minimal set of features from features marked “blue”. Top $\gamma$ subsets based on entropy ranking are further evaluated for silhouette width value and the subset with highest silhouette width value is declared as the final feature subset by GITAUFS and is marked by “green” colour.*/

23: $F'' = \{x: x \subseteq F \text{ and color}(x) = \text{“blue”}\}$

24: $V = \{x: x \in \text{Minimal-Vertex-Covers}(F'')\}$

25: For i = 1 to length(V)

26: $S = \sum (\text{Entropy}(f_j)), \{f_j: f_i \in V_i, V_i = \text{Minimal-Vertex-Covers}_i(F'')\}$

27: Next

28: SORT(S)

29: For i = 1 to $\gamma$

30: opt = max(silhouette-width-value($S_i$))

31: color($V_{opt}$) = “green”

32: color($F'' - V_{opt}$) = “red”

33: $F_{opt} = \{x: x \subseteq F \text{ and color}(x) = \text{“green”}\}$

**End**

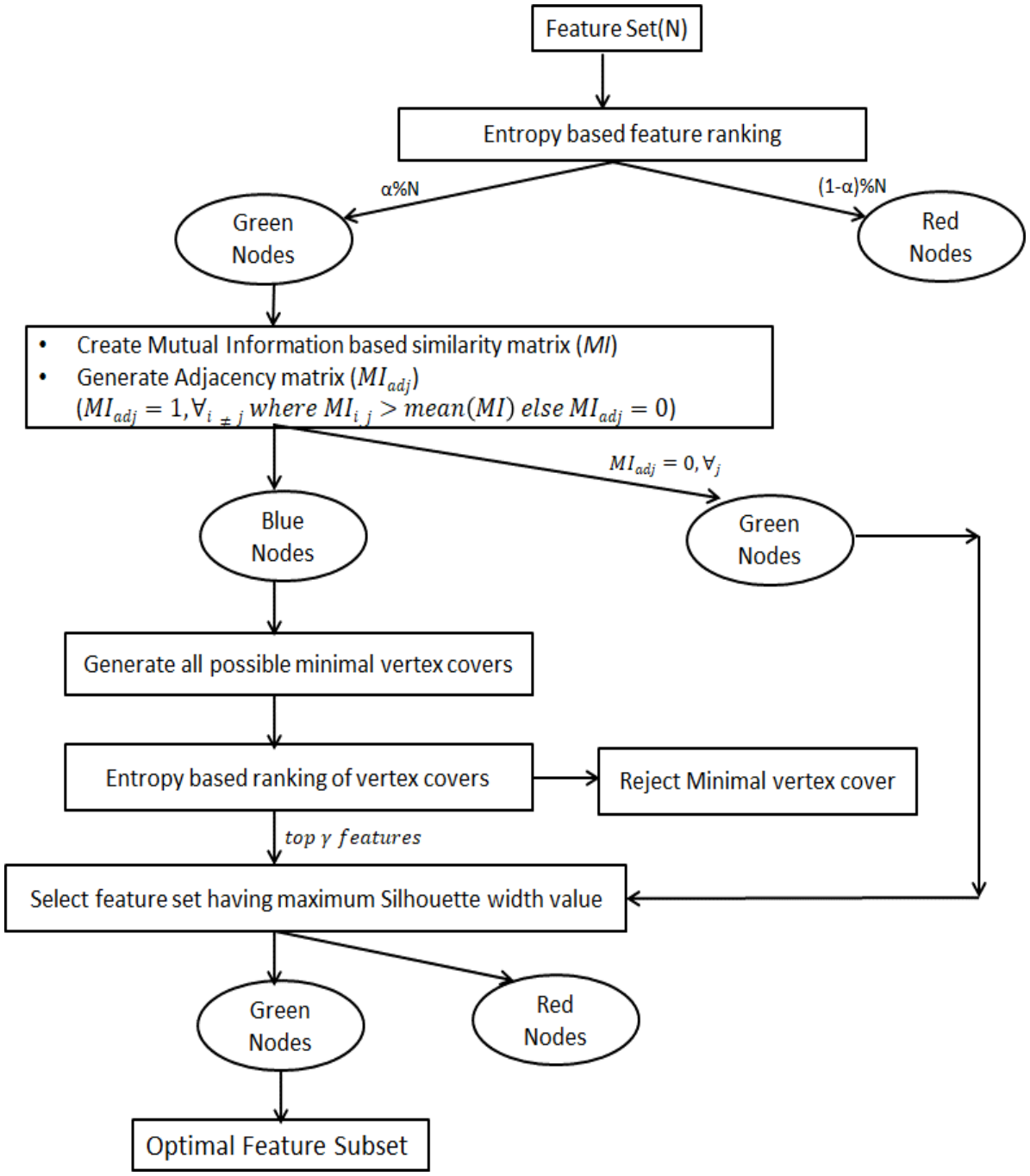


Fig 2: Flowchart for proposed algorithm

# 11. Illustration

In this section, the three stages of GITAUFS approach have been illustrated with the help of generated graphs with color codes representing the selected and rejected vertices after each pruning stage.

- Stage-1: The initial filtering is carried out according to the entropy $E$ of each attribute using Eqn. 2 as described in Section 4. The top $\sigma$% of attributes are chosen for the next stages (they are colored "green") and the rest are discarded (colored "red"). In Fig. 3(a), "At2" and "At4" are filtered out from the list of seven attributes.

- Stage-2: The mutual information (MI) of each of the "green" attributes is calculated using Eqn. 3 mentioned earlier. The mean $\overline{MI}$ of the resulting similarity matrix is used as threshold to distinguish potentially redundant features from others. An adjacency matrix $MI_{ad\ j} = 1, \forall_{i \neq j}$ if $MI_{i,j} > \overline{MI}$ and $MI_{ad\ j} = 0, \forall_{i=j}$. The features having a higher than average mutual information is grouped into a subset (colored "blue"). "At5" is not a redundant attribute and is selected as a final attribute (colored "green") for Stage 3, as shown in Fig 3(b). After this, the attributes having mutual information $M_{i,j} > \overline{MI}$ are connected by edges to form the graph also depicted in the figure.

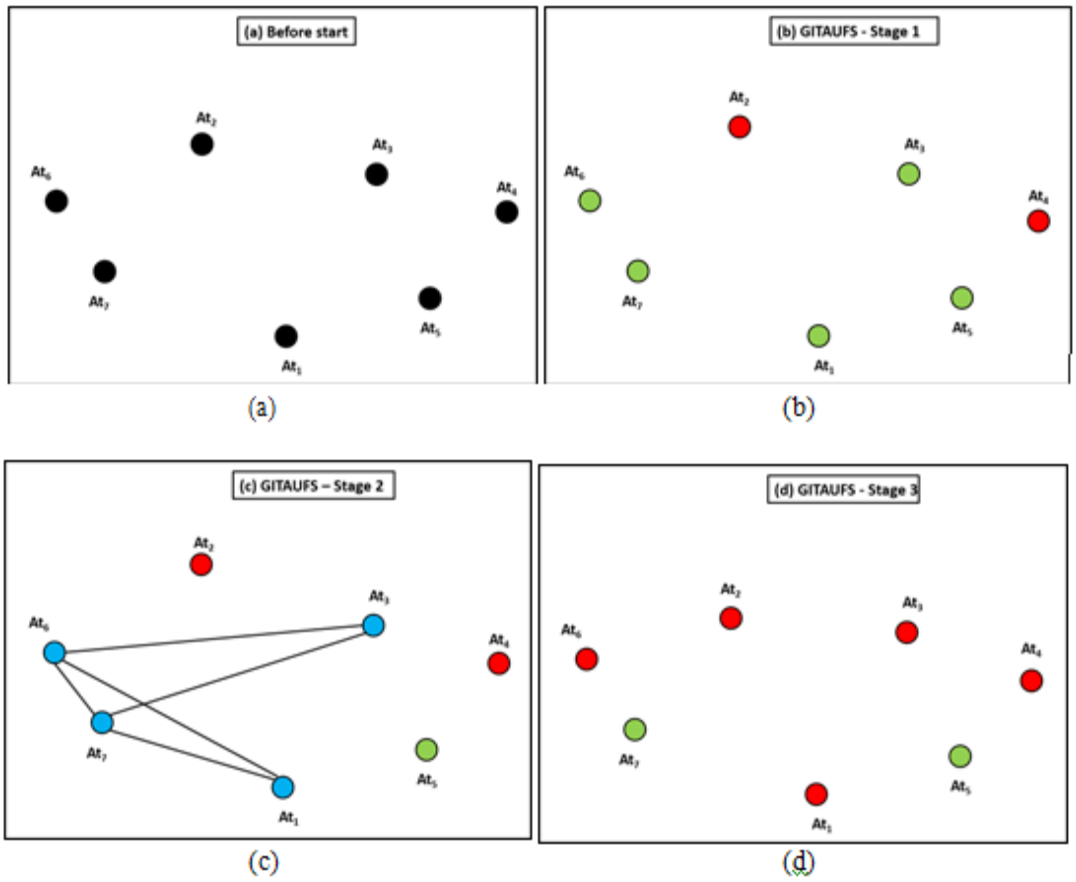


Fig. 3: Illustration for GITAUFS

$$\text{MI} = \begin{array}{c|ccccc} & At_1 & At_3 & At_5 & At_6 & At_7 \\ At_1 & 1.34 & 0.12 & 0.24 & 0.69 & 0.56 \\ At_3 & 0.12 & 1.37 & 0.15 & 0.54 & 0.58 \\ At_5 & 0.24 & 0.15 & 1.38 & 0.32 & 0.17 \\ At_6 & 0.69 & 0.54 & 0.32 & 1.38 & 0.68 \\ At_7 & 0.56 & 0.58 & 0.17 & 0.68 & 1.37 \end{array} , \overline{MI} = 0.46$$

$$\text{MI}_{adj} = \begin{array}{c|ccccc} & At_1 & At_3 & At_5 & At_6 & At_7 \\ At_1 & 1.34 & 0.12 & 0.24 & 0.69 & 0.56 \\ At_3 & 0.12 & 1.37 & 0.15 & 0.54 & 0.58 \\ At_5 & 0.24 & 0.15 & 1.38 & 0.32 & 0.17 \\ At_6 & 0.69 & 0.54 & 0.32 & 1.38 & 0.68 \\ At_7 & 0.56 & 0.58 & 0.17 & 0.68 & 1.37 \end{array}$$

- Stage-3: The final selection of vertices is done from the "blue" vertices. The vertices/attributes marked "green" in Stage 2 are directly selected into the final feature subset. Minimal vertex

cover (MVC) is run on features marked 'blue' in Stage 2. The minimal vertex algorithm returns all possible minimal vertex covers which the top $\gamma$ minimal vertex covers having high entropy are further evaluated along with features marked 'green' in Stage 2 on the basis of silhouette value. The subset having highest silhouette width value is the most final feature subset derived by GITAUFS. These features are marked 'green' such as "At7", is chosen (colored "green") while the others, such as "At1", "At3" and "At6" are rejected (colored "red"), as shown in Fig. 3(c).

| **Data set** | **# of Features** | **# of Instances** |
|---|---|---|
| Apndcts | 7 | 106 |
| Btissue | 9 | 106 |
| Cleave | 13 | 297 |
| Ecoli | 7 | 336 |
| Glass | 9 | 214 |
| ILPD | 10 | 579 |
| Mfeat | 649 | 2000 |
| Pima | 8 | 768 |
| Sonar | 60 | 208 |
| Vehicle | 18 | 846 |
| Wbdc | 30 | 569 |
| Wine | 13 | 178 |
| Wiscon | 9 | 682 |

Table 1: Description of UCI data sets

# 12. Graphs produced by GITAUFS on UCI data sets

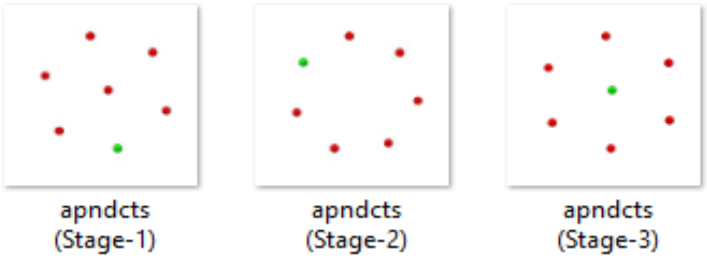


***Fig 4: Illustration of 'apndcts' dataset***

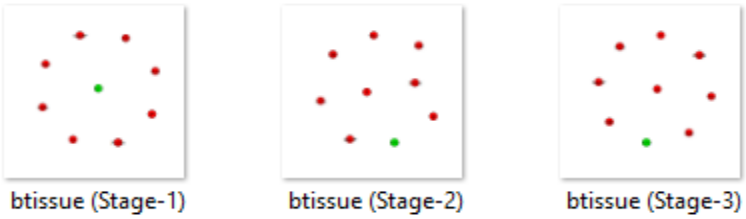


***Fig 5: Illustration of 'btissue' dataset***

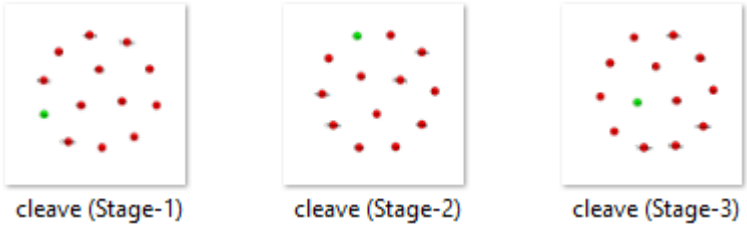


***Fig 6: Illustration of 'cleave' dataset***

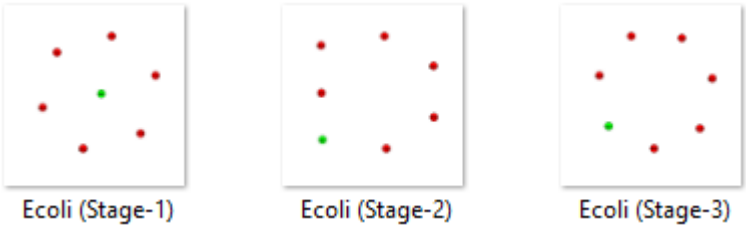


***Fig 7: Illustration of 'Ecoli' dataset***

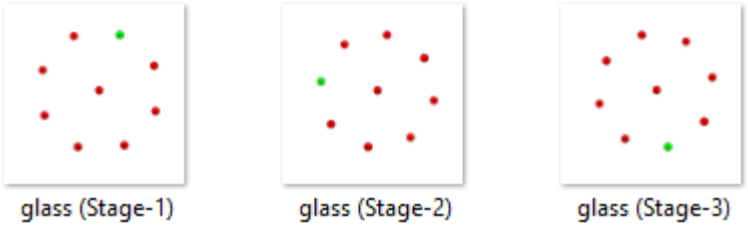


**Fig 8:** ***Illustration of 'glass' dataset***

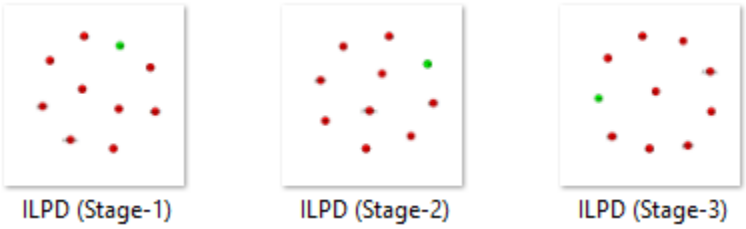


**Fig 9:** ***Illustration of 'ILPD' dataset***

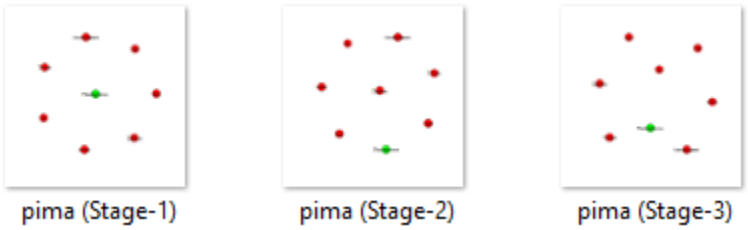


***Fig 10: Illustration of 'pima' dataset***


***Fig 11: Illustration of 'mfeat' dataset***

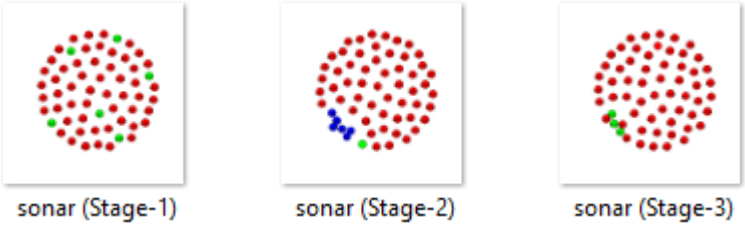


***Fig 12: Illustration of 'sonar' dataset***

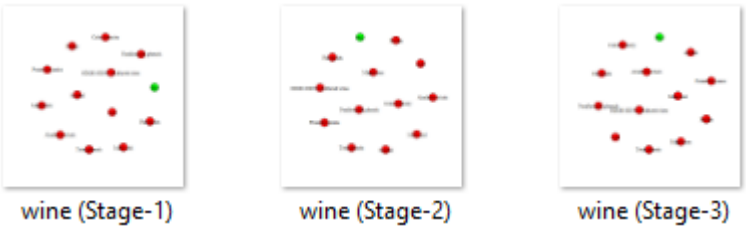


***Fig 13: Illustration of 'wine' dataset***

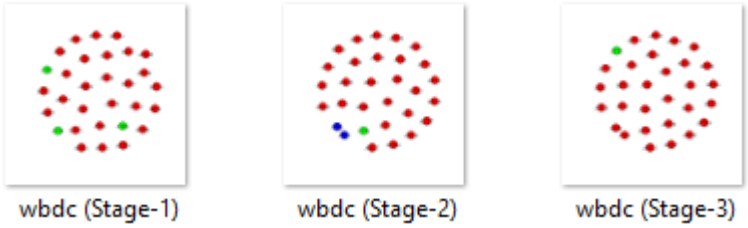


***Fig 14: Illustration of 'wbdc' dataset***

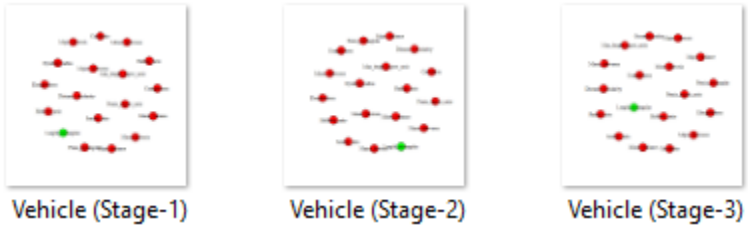


***Fig 15: Illustration of 'Vehicle' dataset***

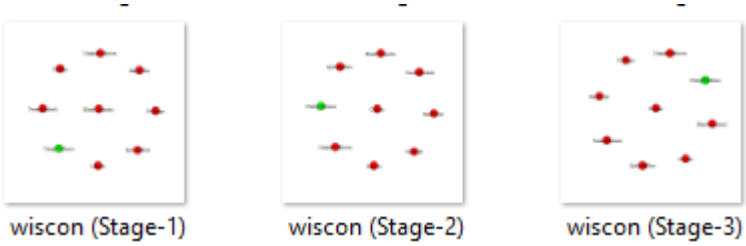


***Fig 16: Illustration of 'wiscon' dataset***

# 13. Experiments and Outcome

Our approach has been benchmarked on data sets obtained from the UCI Machine Learning repository. Our graphs have been generated using the 'matplotlib' library in Python. Table 1 depicts the summary of the characteristics of the data sets used for experimenting. The value of $\alpha$ is 10% in this experiment. GITAUFS has been compared with two graph-based feature selection algorithms - DSUB and UFAM and two benchmark feature selection algorithms - Laplacian and Principal Feature Analysis (PFA).

## 13.1 Summary of Outcome

The proposed algorithm (GITAUFS) has been evaluated for its performance by com- paring it with other competing algorithms based on three main aspects - silhouette width value, percentage of feature reduction and execution time. The following sub-sections describe the comparative results obtained in each of these aspects.

### 13.1.1 Comparison of Silhouette Width Value

| Dataset | GITAUFS | UFAM | ALL | LAPLACIAN | PFA | DSUB |
|---|---|---|---|---|---|---|
| apndcts | **0.66** | 0.6 | 0.46 | 0.44 | 0.48 | 0.43 |
| Btissue | **0.65** | 0.62 | 0.58 | 0.56 | 0.53 | 0.58 |
| Cleave | 0.55 | 0.31 | 0.27 | 0.57 | **0.58** | 0.27 |
| Ecoli | 0.54 | 0.54 | 0.44 | **0.56** | 0.47 | 0.44 |
| Glass | 0.59 | **0.61** | 0.52 | 0.54 | 0.53 | 0.52 |
| ILPD | **0.86** | 0.71 | 0.49 | 0.72 | 0.72 | 0.49 |
| mfeat | 0.20 | 0.2 | **0.24** | 0.18 | 0.14 | 0.19 |
| Pima | 0.56 | 0.27 | 0.51 | 0.27 | 0.27 | 0.24 |
| Sonar | 0.41 | 0.17 | **0.47** | 0.19 | 0.12 | 0.39 |
| Vehicle | **0.62** | 0.52 | 0.32 | 0.43 | 0.38 | 0.47 |
| wbdc | 0.55 | 0.44 | **0.65** | 0.49 | 0.29 | 0.4 |
| Wine | **0.57** | 0.4 | 0.002 | 0.35 | 0.31 | 0.32 |
| wiscon | 0.63 | **0.69** | 0.19 | 0.66 | 0.65 | 0.65 |

Table 2: Performance of silhouette width

The silhouette width values for the proposed algorithm GITAUFS along with other competing algorithms UFAM, LAPLACIAN, PFA, and DSUB have been recorded in Table 2. The column corresponding to ALL contains silhouette width values all features of the dataset.

A graphical comparison of the performance of silhouette width value has been presented in Fig. 4. From Table 2 and Fig. 4 it is evident that GITAUFS has outperformed the benchmark algorithms. GITAUFS has recorded the highest or the near highest silhouette width value for all the data sets used. The summary of the comparison based on silhouette width value is given below.

GITAUFS has the highest silhouette width value for 6 out of 13 data sets used in the experiment. For the other data sets where GITAUFS does not have highest silhouette width value, it is giving value very close to the best value. GITAUFS has outperformed both the benchmark algorithms - Laplacian and PFA with respect to silhouette width value is given below-

- GITAUFS has the highest silhouette width value for 6 out of 13 data sets used in the experiment.
- For the other data sets where GITAUFS does not have highest silhouette width value, it is giving value very close to the best value.
- GITAUFS has outperformed both the benchmark algorithms - Laplacian and PFA with respect to silhouette width value.

### 13.1.2 Comparison of feature reduction

The percentage of feature reduction for GITAUFS and other competing algorithms has been presented in Table 3. Graphical representation of the comparison has been shown in Fig. 5.
GITAUFS has shown a very high percentage of feature reduction for all the data sets when compared to the benchmark algorithms. An overview of the comparison for feature reduction is given below-

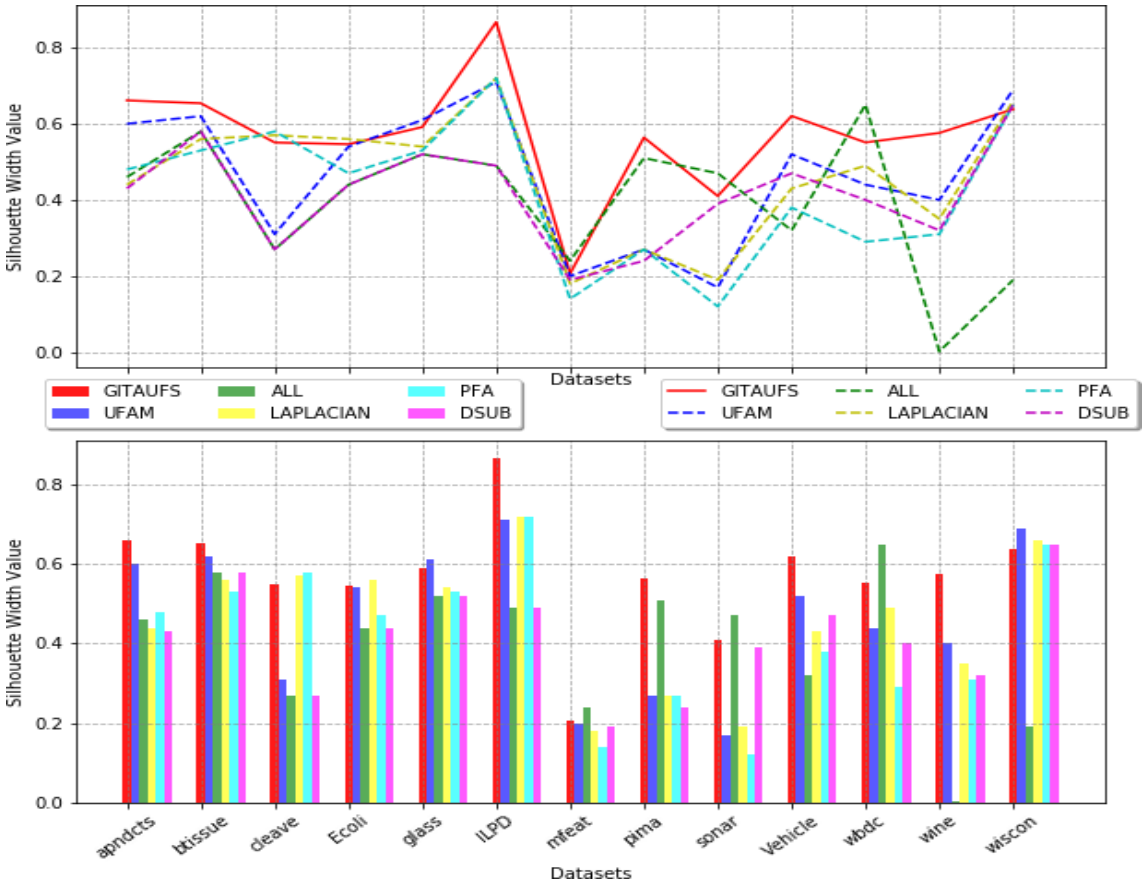


Fig 17: Performance of Silhouette Width

| Dataset | GITAUFS | UFAM | PFA | LAPLACIAN | DSUB |
|---|---|---|---|---|---|
| Apndcts | **85.71** | 50 | 57.14 | 57.14 | 71.43 |
| Btissue | **88.89** | 62.5 | 55.56 | 55.56 | 66.67 |
| Cleave | **92.31** | 25 | 23.08 | 23.08 | 76.92 |
| Ecoli | **85.71** | 33.33 | 28.57 | 28.57 | 71.43 |
| Glass | **88.89** | 37.5 | 33.33 | 33.33 | 77.78 |
| ILPD | **90** | 44 | 30 | 30 | 70 |
| mfeat | 91.68 | 29.78 | 85.82 | 85.82 | **99.69** |
| Pima | **87.5** | 14.29 | 12.5 | 12.5 | 75 |
| Sonar | **95** | 44.07 | 63.33 | 63.33 | 68.33 |
| Vehicle | **94.44** | 35.29 | 72.22 | 72.22 | 88.89 |
| wbdc | **96.67** | 31.03 | 76.67 | 76.67 | 93.33 |
| wine | **92.31** | 25 | 38.46 | 38.46 | 84.62 |
| wiscon | **88.89** | 37.5 | 33.33 | 33.33 | 77.78 |

Table 3: Percentage feature reduction

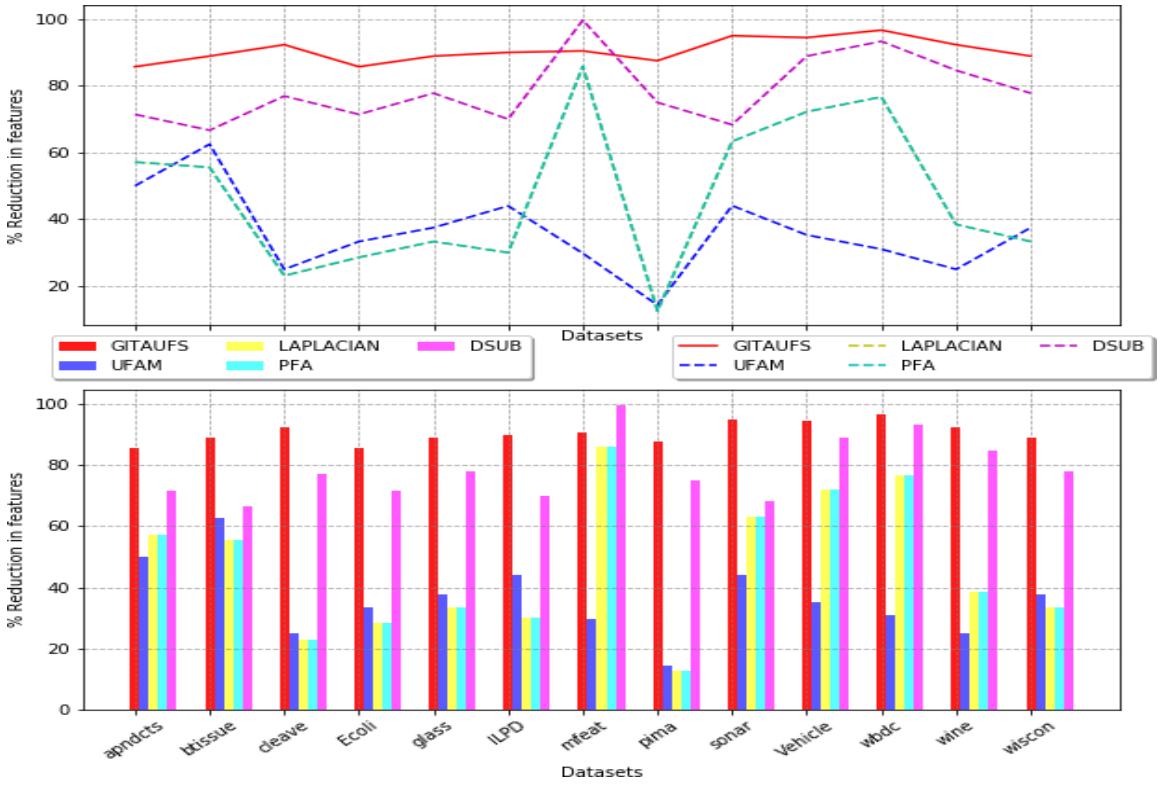


Fig 18: Feature reduction

- GITAUFS has shown the highest reduction in the number of features compared to the competing algorithms for all the data sets except 'mfeat' where DSUB has a higher reduction in the number of features. However, GITAUFS has a better silhouette width value than DSUB for 'mfeat' justifying its efficiency.
- GITAUFS has the highest feature reduction as high as 96.67% and the lowest being 85.71% which is also very high compared to the other competing algorithms.
- GITAUFS has performed extremely good for high dimensional data set giving a reduction in features exceeding 91%.

### 13.1.3 Comparison of execution time

The values representing the execution time for GITAUFS and other competing algorithm is shown in Table 4. Execution time for GITAUFS is very close to benchmark algorithms. A summary of the conclusions drawn on comparing the execution time is given below-

| Dataset | GITAUFS | UFAM | PFA | LAPLACIAN | DSUB |
|---|---|---|---|---|---|
| Apndcts | 0.17 | 0.39 | 0.15 | **0.12** | 0.58 |
| Btissue | 0.25 | 0.4 | 0.18 | 0.15 | **0.09** |
| Cleave | 0.90 | 0.4 | 1.74 | 0.17 | **0.16** |
| Ecoli | 1.37 | 0.41 | 1.27 | 0.1 | **0.09** |
| Glass | 0.61 | 0.41 | 0.46 | 0.12 | **0.11** |
| ILPD | 1.09 | 0.39 | 9.58 | 0.2 | **0.11** |
| mfeat | 854.13 | **2.63** | 41,034.06 | 25.99 | 1,531.69 |
| Pima | 1.17 | 0.39 | 23.9 | 0.14 | **0.1** |
| Sonar | 0.96 | **0.02** | 2.63 | 0.75 | 2.63 |
| Vehicle | 2.74 | 0.47 | 65.5 | **0.14** | 0.3 |
| wbdc | 1.03 | **0.01** | 32.4 | 0.27 | 0.52 |
| Wine | 0.34 | 0.38 | 0.62 | 0.16 | 0.14 |
| wiscon | 0.67 | 0.4 | 17.78 | 0.13 | 0.11 |

Table 4: Execution time(in seconds)

- The execution time for GITAUFS is almost similar to that of benchmark algorithms.
- For high dimensional data sets like 'mfeat', GITAUFS has a fairly acceptable execution time. DSUB has almost double execution time for 'mfeat' and the execution time of PFA is almost 48 times higher than GITAUFS

## 13.2 Overall comparison of performance

An overall comparison of GITAUFS has been done in Table 5 with benchmark algorithms. Summary of the silhouette width performance is given in Table 6. A graphical representation of mean rank for various competing algorithms based on silhouette width value is shown in Fig. 6.

| Algorithm | Mean Silhouette Width | Mean Execution Time(s) | Mean Feature Reduction(%) |
|---|---|---|---|
| GITAUFS | **0.57** | 66.58 | **90.62** |
| UFAM | 0.47 | **0.52** | 36.1 |
| LAPLACIAN | 0.46 | 2.19 | 46.92 |
| PFA | 0.42 | 168.48 | 46.92 |
| DSUB | 0.41 | 118.2 | 78.61 |

Table 5: Comparison of different algorithms

Observations made from the Table 5, Table 6 and Fig. 6 are described below-

| Algorithm | Mean Rank | Number of Wins/Ties |
|---|---|---|
| HITAUFS | 1.85 | 6 |
| UFAM | 2.77 | 2 |
| ALL | 3.85 | 3 |
| LAPLACIAN | 3.23 | 1 |
| PFA | 4.15 | 1 |
| DSUB | 4.38 | 0 |

Table 6: Summary of Performance (Silhouette Width)

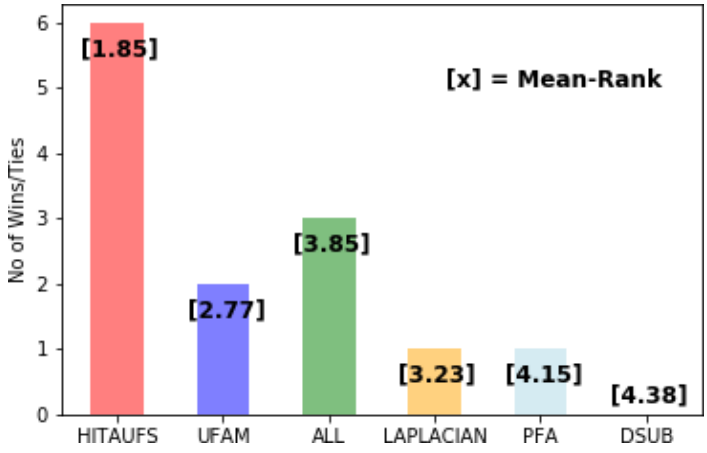


Fig. 19: Wins/Ties for silhouette width value

- GITAUFS has higher silhouette width value than all the other competing algorithms. The mean rank for GITAUFS is the lowest which indicates that GITAUFS has the best silhouette value for most of the data sets used. GITAUFS has the highest number of wins based on its performance of silhouette width.
- Feature reduction is the highest for GITAUFS. GITAUFS has a mean feature reduction of 90.62% which is very high in general and specially high when compared with other competing algorithms.
- The execution time for GITAUFS is similar to other competing algorithms as seen in Table 4. For high dimensional data sets UFAM has best execution time. However GITAUFS also has better than average execution time for high dimensional data sets.

# 14. Conclusion

The proposed approach, GITAUFS, has shown significant feature reduction of 90.62% which is 12% higher than the next best performing algorithm. GITAUFS also gives a high silhouette width value of 57% with the lowest mean rank among other competing algorithms and stands out as the best performer. GITAUFS is an information theoretic approach which captures the general dependency between features. GITAUFS also addresses the challenge of finding the minimal vertex covers being an NP hard problem by evaluating multiple minimum vertex cover. GITAUFS has an added advantage of being a graph-based approach which provides the visualization of all the stages of the algorithm. The graphical representation of features as vertices of a graph gives a visual understanding of the relevance of features and similarity between features. GITAUFS has overall best performance when compared to the benchmark algorithms for the experimental set-up.

# 15. References


1. Bache, K., Lichman, M.: Uci machine learning repository [http://archive. ics. uci. edu/ml]. irvine, ca: University of california. School of information and computer science **28** (2013)
2. Bandyopadhyay, S., Bhadra, T., Mitra, P., Maulik, U.: Integration of dense subgraph finding with feature clustering for unsupervised feature selection. Pattern Recognition Letters **40**, 104–112 (2014)
3. Battiti, R.: Using mutual information for selecting features in supervised neural net learning. IEEE Transactions on neural networks **5**(4), 537–550 (1994)
4. Bennasar, M., Hicks, Y., Setchi, R.: Feature selection using joint mutual information maximi- sation. Expert Systems with Applications **42**(22), 8520–8532 (2015)
5. Brown, G., Pocock, A., Zhao, M.J., Luj án, M.: Conditional
6. likelihood maximisation: a uni- fying framework for information theoretic feature selection. Journal of machine learning re- search **13**(Jan), 27–66 (2012)
7. Das, A.K., Goswami, S., Chakrabarti, A., Chakraborty, B.: A new hybrid feature selection approach using feature association map for supervised and unsupervised classification. Expert Systems with Applications **88**, 81–94 (2017)
8. Das, A.K., Goswami, S., Chakraborty, B., Chakrabarti, A.: A graph-theoretic approach for vi- sualization of data set feature association. In: Advanced Computing and Systems for Security, pp. 109–124. Springer (2017)
9. Dey Sarkar, S., Goswami, S., Agarwal, A., Aktar, J.: A novel feature selection technique for text classification using naive bayes. International scholarly research notices **2014** (2014) 9.Ding, C., Peng, H.: Minimum redundancy feature selection from microarray gene expression data. Journal of bioinformatics and computational biology **3**(02), 185–205 (2005)
10. Feng, S., Duarte, M.F.: Graph autoencoder-based unsupervised feature selection with broad and local data structure preservation. Neurocomputing **312**, 310–323 (2018)
11. Goswami, S., Das, A.K., Guha, P., Tarafdar, A., Chakraborty, S., Chakrabarti, A., Chakraborty, B.: An approach of feature selection using graph-theoretic heuristic and hill climbing. Pattern Analysis and Applications pp. 1–17 (2017)
12. Gu, Q., Li, Z., Han, J.: Generalized fisher score for feature selection. arXiv preprint arXiv:1202.3725 (2012)

13. Hall, M.A.: Correlation-based feature selection for machine learning (1999)
14. He, X., Cai, D., Niyogi, P.: Laplacian score for feature selection. In: Advances in neural information processing systems, pp. 507–514 (2006)
15. Hua, J., Tembe, W.D., Dougherty, E.R.: Performance of feature-selection methods in the clas- sification of high-dimension data. Pattern Recognition **42**(3), 409–424 (2009)
16. Lewis, D.D.: Feature selection and feature extraction for text categorization. In: Proceedings of the workshop on Speech and Natural Language, pp. 212–217. Association for Computa- tional Linguistics (1992)
17. Lu, Y., Cohen, I., Zhou, X.S., Tian, Q.: Feature selection using principal feature analysis. In: Proceedings of the 15th ACM international conference on Multimedia, pp. 301–304. ACM (2007)
18. Meyer, P.E., Schretter, C., Bontempi, G.: Information-theoretic feature selection in microarray data using variable complementarity. IEEE Journal of Selected Topics in Signal Processing **2**(3), 261–274 (2008)
19. Moghaddam, B., Pentland, A.: Probabilistic visual learning for object detection. In: Proceed- ings of IEEE international conference on computer vision, pp. 786–793. IEEE (1995)
20. Moradi, P., Rostami, M.: A graph theoretic approach for unsupervised feature selection. En- gineering Applications of Artificial Intelligence **44**, 33–45 (2015)
21. Moradi, P., Rostami, M.: Integration of graph clustering with ant colony optimization for fea- ture selection. Knowledge-Based Systems **84**, 144–161 (2015)
22. Murphy, K., Torralba, A., Eaton, D., Freeman, W.: Object detection and localization using lo- cal and global features. In: Toward Category-Level Object Recognition, pp. 382–400. Springer (2006)
23. Ng, K., Liu, H.: Customer retention via data mining. Artificial Intelligence Review **14**(6), 569–590 (2000)
24. Quinlan, J.R.: C4. 5: programs for machine learning. Elsevier (2014)
25. Xing, E.P., Jordan, M.I., Karp, R.M.: Feature selection for high-dimensional genomic microar- ray data. In: ICML, vol. 1, pp. 601–608. Citeseer (2001)
26. Yang, H.H., Moody, J.: Data visualization and feature selection: New algorithms for nongaus- sian data. In: Advances in neural information processing systems, pp. 687–693 (2000) 27.Zhang, Z., Hancock, E.R.: A graph-based approach to feature selection. In: International Workshop on Graph-Based Representations in Pattern Recognition, pp. 205–214. Springer (2011)