Feature Selection via Weighted Independent Domination

M. R. Benabid*, F. N. Abu-Khzam*
* Department of Computer Science and Mathematic, Lebanese American University
manelroumaissa.benabid@lau.edu

Abstract—**Feature selection is an important step in machine learning tasks that involves choosing a subset of relevant features from the original set of features. This process is performed to improve model performance, reduce overfitting, enhance interpretability, and decrease computational complexity. There are supervised and unsupervised feature selection techniques such as wrapper and filter (supervised) and Principle component Analysis (unsupervised). Features can be ranked by importance by various methods such as tree-based approaches (e.g. Random Forest). Modeling the pairwise relation between columns using graphs and selecting a minimum set of columns i.e., (features) that form an independent dominating set have been proposed more than a decade ago. This latter approach seems to have been neglected in the literature. Moreover, the corresponding independent dominating set model was solved via an ILP formulation, which takes exponential time. Observing that certain features can be favored over others based on a novel 'diversity criterion', we propose a weighted independent dominating set approach that can better model the feature selection and ranking process. The most common greedy heuristic for domination problems, namely the classic Chvatal's *maximum-utility* heuristic, is adapted in this paper to be used for a weighted variant equipped with the proposed diversity measure. Preliminary experimental results show great improvements over the known feature selection models.**

Index Terms—**Feature selection, Minimum Independent Dominating Set, Classification, Diversity.**

I. Introduction

The primary goal of feature selection is to carefully choose a subset of variables from the input that effectively characterizes the input data. This process aims to minimize the impact of noise or irrelevant variables, ensuring that the selected subset maintains the ability to yield accurate prediction results [1]. The standardized gene expression data may encompass numerous variables, and a considerable portion of them might exhibit high correlations with each other. For instance, if two features are perfectly correlated, it implies that only one feature is necessary to capture and describe the underlying data. [1]. Dependent variables, devoid of class-relevant information, act as noise in prediction. Removing them reduces data volume and improves classification performance by focusing on distinctive features. In some cases, uncorrelated variables may introduce bias, affecting accuracy. Feature selection enhances insights, streamlining computation, and boosting predictive precision. [1]. In high-dimensional data, where the number of features far exceeds the number of samples, identifying the optimal feature subset becomes a challenging task [2]. Feature selection has proven successful in medical applications, serving not only to diminish dimensionality but also aiding in comprehending the underlying causes of diseases. Machine learning models generally demand an ample number of samples to prevent overfitting and enhance generalization. In contrast, a large number of features is not necessary due to concerns related to the curse of dimensionality [3]. Dimensionality reduction techniques can be categorized into feature selection and feature extraction methods. The key distinction lies in the fact that feature extraction amalgamates the original features to generate a new set, whereas feature selection picks a subset from the original features [3]. Feature selection methods employ either individual evaluation (feature ranking) or subset evaluation of features based on the desired output. In individual evaluation, features are assessed individually, each assigned a weight reflecting its relevance. In subset evaluation, candidate feature subsets are evaluated using a specified measure to select the most effective features [3]. Feature selection methods can be classified into supervised, semi-supervised, and unsupervised [4]. Supervised feature selection methods can be categorized into three main categories [4]:

1) Filter:
   a) Univariate filter methods: or ranking-based UFS. Evaluate features to come up with a ranking list. These methods are good at weeding out irrelevant features. However, they cannot remove the redundant ones because they do not consider dependent features.
   b) Multivariate filter methods: evaluate features together instead of each one individually. They can handle redundancy in features and treat the irrelevant ones.

They have proven to be much better than univariate filter methods.

2) Wrapper
    a) Sequential methods: features get added/removed in a sequential manner.
    b) Bio-inspired: use randomness to avoid local optima
    c) Iterative: they cast the UFS problem as an estimation problem to avoid combinatorial search.
3) Hybrid/embedded: features are selected/ranked by implementing a measure that is based on intrinsic properties of the data.

The work of Kou et al. [5] evaluated methods of feature selection using multiple criteria decision-making (MCDM)- based methods for text classification. They experimented on 10 classification datasets where the number of features is much greater than the number of observations (curse of dimensionality). They concluded that no feature selection method outperformed others on all criteria, regardless of features or classifier. Therefore, employing multiple performance measures was necessary for a thorough evaluation.

The problem of Independent Dominating Set (IDS), which is a combinatorial optimization problem [6] is defined as follows: ”An independent dominating set of a graph G is a subset D of $V(G)$ such that every vertex not in D is adjacent to at least one vertex of D and no two vertices in D are adjacent.” IDS seeks to find a set with minimum cardinality [7] which we call the Minimum Independent Dominating Set (MIDS). Moreover, the problem of a weighted independent dominating set (WIDS) asks to find the IDS with the smallest weight such that the weight of the set D is the sum of the weights of all vertices in D. MIDS is NP-hard by a trivial reduction from Minimum Dominating Set (MDS). The work of [8] consisted of devising a branching algorithm to compute the MIDS, they achieved a running time of $O*(1.3351^n)$.

In our approach, we aim to assign weights to features according to their diversity, i.e. if the feature contains a variety of distinct values with respect to the class value then it’s diverse. The authors of [9] explored this in their work, they proposed a relevance-diversity-based feature selection technique to optimize features and reduce feature selection search time. We will not be utilizing their diversity measure to assign weights to our features in such a way that the most diverse features obtained from the MIDS get higher weights than less diverse features.

II. Methodology

In our proposed approach, we first start by finding the minimum independent dominating set (MIDS) from the set of features. Similar to the work of [10] our approach consists of representing features as vertices in an undirected graph (Step1), two vertices are connected through an edge if they are correlated. The correlation measure is the Pearson correlation coefficient and we say that two features are correlated if their Pearson correlation measure is greater than a certain threshold T. Step 2 consists of computing the diversity measure and assigning that value as a weight to each vertex. The measure we will be using to calculate the diversity of a feature is the coefficient of variation (CV). We are using the CV because our data consists of continuous values, note that if the data is discrete values then entropy is better at measuring diversity in the data. The equation of the CV is shown in Fig 1. Step 3 consists of computing the weighted minimum independent dominating set (WMIDS) of the graph. The vertices of the WMIDS represent the features that are not correlated/dependent and every other feature not the WMIDS is dependent on at least one feature in the WMIDS. This allows us to cover all information in the data using the least number of features without losing any valuable information. Step 4 consist of performing feature selection, two feature selection methods were used: first is Lasso, and the second is SVM-RFE. Finally, in step 5 we use the obtained features to run our classification algorithm using Random Forest and Support Vector Machines.

$$CV\ (\%) = \left(\frac{Standard\ deviation}{Mean}\right) \times 100$$

Fig. 1. Coefficient of Variation (CV) equation

A. Computing the WMIDS (Algorithm)

Algorithm 1 shows the code for the heuristic used to compute the WMIDS. the maxUtil function returns the active vertex with the maximum utility value while the delVertex is designed to delete the specified vertex x and update the utility values of its neighboring vertices.

The max-utility heuristic picks the vertex with the highest sum of utility of its neighbors. To analyze the running time of the WMIDS function, let’s break down the main components:

1. **maxUtil Function:** The ‘maxUtil‘ function iterates over all vertices (‘n‘ vertices) to find the vertex with the maximum utility. The time complexity is O(n).

2. **While Loop in WMIDS heuristic:** The ‘WMIDS‘ function contains a ‘while‘ loop that repeatedly calls ‘maxUtil‘ and ‘delVertex‘. In each iteration, it finds the vertex with the maximum utility and deletes its neighbors.

Inside the loop:

- ‘maxUtil‘: O(n)
- ‘delVertex‘: The complexity depends on the number of neighbors of the current vertex. Let’s denote the maximum degree in the graph as maxDegree. The worst-case complexity is O(maxDegree).

The ‘while‘ loop iterates at most n times (once for each vertex). Therefore, the worst-case time complexity of the ‘while‘ loop is $O(n \times \text{maxDegree})$.

3. **Final Loop in WMIDS Function:** After the ‘while‘ loop, there is a final loop over all vertices to handle isolated vertices (vertices with status 1 that are not covered by the previous iterations). The time complexity of this loop is O(n).

Overall, the worst-case time complexity of the ‘WMIDS‘ function is dominated by the ‘while‘ loop, resulting in $O(n \times \text{maxDegree})$.

B. Feature Selection: Lasso & SVM-RFE

A common problem we face when training data is overfitting, which is a phenomenon that occurs when the model gets too specific to the training data and cannot generalize over unseen data. When this happens, the model performs well on the training data but poorly on unseen data.

Regularization helps deal with overfitting by reducing the complexity of the model through a penalty term on the model’s parameters i.e. it restricts the magnitude of the coefficients. By doing so, the model is less likely to fit the noise in the training data and will be able to generalize better. The idea behind reducing the complexity is that it is theorized that models that overfit the data are complex, in other words, they have many parameters.

Overfitting means the model has high variance, regularization helps develop a better model that does not fit the data “too well” by increasing the bias. By increasing the bias, we reduce the variance of the model (Bias-Variance trade-off). One of the most common techniques of regularizing linear models also called shrinkage methods is L1 regularization or Lasso. The optimal model requires us to define a loss function to describe how well the model fits the data. The model needs to minimize this function by penalizing it by adding a complexity term that would increase the loss when the model complexity increases.

Algorithm 1: Weighted Minimum Idependent Dominating Set

```
Data: $d \geqslant 0$, $sts \leftarrow \{0, 1\}$, $util \geqslant 0$
$v \leftarrow 0$;
$w \leftarrow 0$;
$i \leftarrow 0$;
$count \leftarrow 0$;
$r \leftarrow 1$;
$v \leftarrow maxUtil(d, sts, util)$;
while $v \neq -1$ do
    $sts[v] \leftarrow 2$;
    $rank[v] \leftarrow r++$;
    $count++$;
    for $i = 0$ to $d[v]$ do
        $w = adjList[v][i]$;
        if $sts[w] == 1$ then
            $delVertex(d, sts, util, w)$;
        end
    end
    $v = maxUtil(d, sts, util)$;
end
for $i = 0$ to $n$ do
    if $sts[w] == 1$ then
        $sts[i] = 2$;
        $rank[i] = r++$;
        $count++$;
    end
end
return $count$
```

In the case that we are using least squares to fit the data, the function minimizes the sum of the square residuals. Lasso seeks to :

minimize (sum of the sum square residuals) + lambda * abs(weight)

We can increase or decrease the value of lambda, for models with large lambda highly complex models are excluded. For small lambda values, models that have high training errors are excluded.

Lasso adds a penalty for non-zero coefficients but (unlike L2 Ridge) it penalizes the sum of their absolute value i.e. for large lambda value, coefficients can be shrunk to zero, which is never the case with ridge, where coefficients can get very small and close to zero but never zero. The most important part is the

Datasets	Gene	Instances	Class
Colon	2000	60	2
Leukemia	7129	72	2

TABLE I
DATASETS

choice of lambda, which can be done through cross-validation.

On a second note, Support Vector Machine Recursive Feature Elimination (SVM-RFE) is a feature selection technique that combines the principles of Support Vector Machines (SVM) and recursive feature elimination to identify the most relevant features in a dataset. SVM-RFE works in the following way, it recursively removes the least important features from the dataset. The process involves training the model, ranking the features based on their importance, and eliminating the least important feature. This process is repeated until the desired number of features is reached.

C. Classification Models

After the best features are chosen, we proceed to the final step, classification. The models that will be used for classification are Random Forest and Support Vector Machines (SVM) with linear kernels. We will also be implementing hyperparameter tuning in the hopes of getting better classification accuracy.

III. EXPERIMENTAL RESULTS

Table I shows the datasets used in our experiments, the datasets were downloaded from [11].

We first computed the coefficient of variation for each of the features and used the absolute value as a weight for the graph vertices. The correlation matrix of each pair of genes was also computed. As mentioned before, we say that two vertices are connected if the features are "correlated"; two features are correlated if their Pearson correlation is treated than a certain threshold. We experimented with three different thresholds: 0.7,0.75, and 0.8. The vertices, their weights, and edges were then given to the heuristic to compute the weighted MIDS.

A. Colon dataset results

Table II shows the results of the weighted MIDS and the features chosen using Lasso and SVM-RFE for the three threshold values. We also ran Lasso and SVM-RFE feature extraction on the full dataset and we obtained 29 features using Lasso and 24 features using SVM-RFE.

Threshold	WMIDS	Lasso Features	SVM-RFE features
0.7	255	11	22
0.75	344	8	30
0.8	548	11	31

TABLE II
COLON FEATURE EXTRACTION

Threshold	WMIDS	Lasso Features	SVM-RFE features
0.7	6664	48	50
0.75	6913	46	50
0.8	7025	45	50

TABLE III
LEUKEMIA FEATURE EXTRACTION

B. Leukemia dataset results

Table III shows the results of the weighted MIDS and the features chosen using Lasso and SVM-RFE for the three threshold values. We also ran Lasso and SVM-RFE feature extraction on the full dataset and we obtained 5 features using Lasso and ? features using SVM-RFE.

C. Classification results

Tables IV and V show the classification results of the SVM and Fandom forest Model respectively. We noticed that the SVM model performed better than the Random Forest with the Leukemia dataset whereas it performed as good as Random Forest on the Colon dataset. As for the contribution of weighted MIDS to the classification, we notice that both models achieved a better accuracy with features obtained from the weighted MIDs than those obtained without performing the weighted MIDS. We can conclude that computing the MIDS of the features is a good approach to finding the best minimum number of features to represent the data better without loss of significant information. Moreover, obtaining the MIDS features significantly improved the interpretability of the model, i.e. we can tell which are the features that significantly contribute to the prediction without interfering with other features. However, compared to the results by [10], their model performed far better in terms of classification accuracy. They achieved an accuracy of 0.94 on the Colon dataset compared to 0.85 achieved by our model, and an accuracy of 0.99 with 14 features only compared to 0.93 with 50 features by our model. The significant difference between our model and the model of [10] is the use of max-utlity heuristic for the weighted MIDS while they computed the exact MIDS using the ILP formulation, our utility was the diversity of the feature where we claimed that diverse features are more important than less diverse features and applied that approach on genetic expression data for tumor/cancer classification. We can hypothesize that in the context of classification, genetic diversity can play a role in determining the susceptibility of individuals to diseases like tumors. Higher genetic diversity within a population might contribute to a more varied response to tumor development, potentially influencing factors such as tumor growth rates or responses to treatment. However, we theorize that the relationship between genetic diversity and tumor classification is complex, as it depends on the specific genes involved, the nature of the tumors, and the environmental factors influencing their development. In some cases, certain genetic variations may be associated with an increased risk of tumors, while others may confer resistance or resilience. We conclude that understanding the interplay between genetic diversity and tumor classification is crucial for developing effective diagnostic, and that the diversity of certain genes does not necessarily mean that it is significant for our diagnosis.

IV. Conclusion

This paper introduces an innovative approach to feature selection, presenting a framework centered around the concept of weighted minimal independent dominating sets. The framework identifies weighted minimum independent dominating sets that maintain the original feature space's functionality while accounting for feature diversity. By eliminating redundant features, our approach enhances the performance of cutting-edge feature selection algorithms. Empirical studies demonstrate that our framework's selected features result in improved prediction accuracy compared to applying feature selection algorithms directly to all input features.

References

1 Chandrashekar, G. and Sahin, F., "A survey on feature selection methods," *Computers Electrical Engineering*, vol. 40, no. 1, pp. 16–28, 2014, 40th-year commemorative issue. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0045790613003066

2 Khaire, U. M. and Dhanalakshmi, R., "Stability of feature selection algorithm: A review," *Journal of King Saud University - Computer and Information Sciences*, vol. 34, no. 4, pp. 1060–1073, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1319157819304379

3 Remeseiro, B. and Bolon-Canedo, V., "A review of feature selection methods in medical applications," *Computers in Biology and Medicine*, vol. 112, p. 103375, 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0010482519302525

4 Solorio-Fernández, S. and , J. A. M.-T. J. F., "A review of unsupervised feature selection methods," *Artificial Intelligence Review*, 2020.

5 Kou, G., Yang, P., Peng, Y., Xiao, F., Chen, Y., and Alsaadi, F. E., "Evaluation of feature selection methods for text classification with small datasets using multiple criteria decision-making methods," *Applied Soft Computing*, vol. 86, p. 105836, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1568494619306179

6 Wang, Y., Li, C., and Yin, M., "A two phase removing algorithm for minimum independent dominating set problem," *Applied Soft Computing*, vol. 88, p. 105949, 2020.

7 Liu, C.-H., Poon, S.-H., and Lin, J.-Y., "Independent dominating set problem revisited," *Theoretical Computer Science*, vol. 562, pp. 1–22, 2015.

8 Bourgeois, N., Della Croce, F., Escoffier, B., and Paschos, V., "Fast algorithms for min independent dominating set," *Discrete Applied Mathematics*, vol. 161, no. 4, pp. 558–572, 2013, seventh International Conference on Graphs and Optimization 2010. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0166218X1200011X

9 Shaheen, M., Naheed, N., and Ahsan, A., "Relevance-diversity algorithm for feature selection and modified bayes for prediction," *Alexandria Engineering Journal*, vol. 66, pp. 329–342, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1110016822007359

10 Shu, L., Ma, T., and Latecki, L. J., "Stable feature selection with minimal independent dominating sets," in *Proceedings of the International Conference on Bioinformatics, Computational Biology and Biomedical Informatics*, ser. BCB'13. New York, NY, USA: Association for Computing Machinery, 2013, p. 450–457. [Online]. Available: https://doi.org/10.1145/2506583.2506600

11 Zhu, Z., Ong, Y.-S., and Dash, M., "Markov blanket-embedded genetic algorithm for gene selection," *Pattern Recogn.*, vol. 40, no. 11, p. 3236–3248, nov 2007. [Online]. Available: https://doi.org/10.1016/j.patcog.2007.02.007

Datasets	SVM-RFE		LASSO	
	Without MIDS	With MIDS	Without MIDS	With MIDS
Colon	0.69	0.77	0.85	**0.85**
Leukemia	0.93	0.93	0.87	**0.93**

TABLE IV
SVM classification results- Accuracy

Datasets	SVM-RFE		LASSO	
	Without MIDS	With MIDS	Without MIDS	With MIDS
Colon	0.69	0.77	0.85	**0.85**
Leukemia	0.62	0.77	0.62	**0.77**

TABLE V
Random Forest classification results- Accuracy